

ROZPRAWA DOKTORSKA

**Równania różniczkowe cząstkowe  
w problemach filtracji obrazów  
i trajektorii przestrzennych**

AUTOR:  
mgr inż. Bartosz Jabłoński

PROMOTOR:  
prof. dr hab. inż. Ewaryst Rafajłowicz

SŁOWA KLUCZOWE:  
- przetwarzanie sygnałów  
- filtracja obrazów  
- trajektorie przestrzenne  
- równania różniczkowe cząstkowe  
- przestrzeń rotacji  
- kwaterniony

WROCLAW 2007

## STRESZCZENIE

Przetwarzanie ruchu jest rozwijającą się dziedziną, która jest intensywnie wykorzystywana w różnych zastosowaniach (przemysłowych, medycznych, edukacyjnych, rozrywkowych, itp.). Z punktu widzenia praktycznego opisu modeli obiektów poruszających się istotne jest rozważenie trajektorii danych reprezentujących rotacje w przestrzeni. Rotacje takie mogą być opisane przy użyciu różnych parametryzacji – w szczególności często używana jest parametryzacja oparta na kwaternionach. W pracy rozważany jest problem filtracji trajektorii reprezentujących ruch w przestrzeni rotacji. Badane są metody filtracji oparte na dyskretnej realizacji równania różniczkowego o pochodnych cząstkowych dla dyfuzji ciepła.

W pierwszej części pracy przedstawiono propozycję metody filtracji obrazów uwzględniającej dodatkową wiedzę o strukturze geometrycznej obrazu. Metody te zostały następnie uogólnione do przetwarzania trajektorii w przestrzeni rotacji. Zaproponowano algorytm uwzględniający specyfikę przestrzeni kwaternionów oraz zachowujący kształt przetwarzanej trajektorii. Dla zaproponowanej metody udowodniono istotne własności teoretyczne oraz wykonano serię eksperymentów numerycznych. Badania przeprowadzono zarówno dla trajektorii syntetycznych, jak i rzeczywistych danych reprezentujących ruch ludzki. Otrzymane wyniki wskazują na znaczną poprawę jakości otrzymywanych wyników w porównaniu do istniejących metod filtracji.

## ABSTRACT

Motion processing is the rapidly developing area, which is also intensively exploited in many practical applications (for industrial, medical, educational, entertainment and other purposes). From the point of view of models of moving objects, it is important to consider data trajectory, which represent rotations in space. Rotations can be described using different kinds of parametrization of the space. One of commonly used parametrization is based on quaternions. In the presented thesis the problem of rotational trajectory filtering is being considered. Different methods based on discrete realization of partial differential equation describing heat diffusion are presented and investigated.

In the first part of the thesis a method for image filtering, which considers additional knowledge about geometric structure of the image. These kinds of methods have been consequently generalized for trajectory processing in rotational space. A new algorithm has been proposed, which takes into consideration specific features of quaternion space and preserves the shape of processed trajectory. For the proposed method some important theoretical properties have been proved. Moreover a set of numerical experiments have been performed. The experiments were carried out both for synthetical test trajectories and for real sequences representing human motion. Obtained results show significant improvement in quality of filtering results in comparison to the classical algorithms.

## PODZIĘKOWANIA

Do powstania tej pracy przyczyniło się bardzo wielu wspaniałych ludzi, bez których trudno byłoby mi doprowadzić wszystkie badania do końca. Przede wszystkim bardzo dziękuję Panu prof. Ewarystowi Rafajłowiczowi – za niezliczone dyskusje, pomysły oraz opiekę naukową przez ostatnie 4 lata. Dziękuję też Pani dr hab. Ewie Skubalskiej Rafajłowicz za wszystkie podpowiedzi i sugestie.

Dziękuję Panu dr Ryszardowi Klempousowi za inspirację tematyką przetwarzania ruchu i moralne wsparcie prowadzonych badań. Badania dotyczące analizy ruchu ludzkiego rozpocząłem za Jego namową w ramach projektu TRAF – dziękuję wszystkim jego obecnym i poprzednim uczestnikom, w szczególności: dr Jackowi Segenowi, Markowi Kulbackiemu i Jackowi Rawickiemu. Dziękuję także Andrzejowi Rusieckiemu, Krzyszłowi Halawie oraz pozostałym kolegom z naszego Zakładu za wspólne zmaganie się na studiach doktoranckich.

Od chwili, gdy rozpocząłem badania związane z doktoratem, wiele się zmieniło. Dziękuję mojej wspaniałej żonie Małgosi za jej obecność i pomoc we wszystkich małych oraz wielkich sprawach. Nie byłbym w stanie skończyć tej pracy, gdyby nie Jej wsparcie w każdej trudnej chwili. Dziękuję moim Rodzicom, Babci i Siostrze za wszystkie lata spędzone razem, za troskę, pomoc i wiele ciepłych słów.

Dziękuję wszystkim tym, których nie wymieniłem, a którzy udzielali mi wiele cennych rad i wskazówek oraz na różne sposoby przyczynili się do powstania tego doktoratu.

Badania prowadzone w ramach pracy doktorskiej możliwe były dzięki finansowemu wsparciu, które otrzymałem od:

- Prorektora ds. Badań Naukowych i Współpracy z Gospodarką – prof. dr hab. inż. Tadeusza Więckowskiego w ramach wewnętrznego grantu promorskiego
- Fundacji na Rzecz Nauki Polskiej
- Ministra Nauki i Szkolnictwa Wyższego w ramach promotorskiego projektu badawczego

## Spis treści

<b>1. WPROWADZENIE .....</b>	<b>8</b>
<b>2. PORÓWNANIE METOD FILTRACJI DLA OBRAZÓW DWUWYMIAROWYCH .....</b>	<b>14</b>
2.1 WPROWADZENIE DO SPOSOBÓW REPREZENTACJI DANYCH .....	15
2.2 MATEMATYCZNY OPIS PROCESÓW DYFUZYJNYCH .....	17
2.3 ALGORYTMY NUMERYCZNEGO ROZWIĄZYWANIA RÓWNAŃ RÓŻNICZKOWYCH CZĄSTKOWYCH JAKO FILTRY CYFROWE .....	20
2.4 PRZESTRZENNE METODY FILTRACJI OPARTE NA PROCESIE DYFUZJI .....	23
2.5 PRZEGLĄD NIELINIOWYCH METOD FILTRACJI.....	28
2.5.1 <i>Model Perona-Malik</i> .....	29
2.5.2 <i>Dyskretyzacja nieliniowego modelu Perona-Malik</i> .....	32
2.5.3 <i>Regularyzacja modelu nieliniowego</i> .....	33
2.5.4 <i>Anizotropowy model nieliniowej filtracji</i> .....	34
2.6 PRZYKŁADY ZASTOSOWANIA FILTRÓW OPARTYCH NA RÓWNIANIACH RÓŻNICZKOWYCH .....	37
2.6.1 <i>Generacja przestrzeni skal na podstawie obrazu oryginalnego</i> .....	38
2.6.2 <i>Skuteczność filtracji realizującej liniowe równanie dyfuzji – obrazy syntetyczne</i> .....	41
2.6.3 <i>Skuteczność filtracji realizującej nieliniowe równanie dyfuzji – obrazy syntetyczne</i> .....	43
2.6.4 <i>Porównanie skuteczności filtracji obrazów rzeczywistych</i> .....	49
2.7 PODSUMOWANIE METOD FILTRACJI OBRAZÓW.....	53
<b>3. PROPOZYCJA UWZGLĘDNIENIA W PROCESIE FILTRACJI GLOBALNEJ WIEDZY O STRUKTURZE GEOMETRYCZNEJ OBRAZU.....</b>	<b>55</b>
3.1 PRZEGLĄD ISTNIEJĄCYCH METOD UWZGLĘDNIAJĄCYCH DODATKOWE INFORMACJE.....	55
3.1.1 <i>Uwzględnienie informacji z wielokanałowego toru pomiarowego</i> .....	56
3.1.2 <i>Uwzględnienie informacji dotyczących własności geometrycznych</i> .....	56
3.2 STANDARDOWA TRANSFORMACJA HOUGHGA .....	59
3.2.1 <i>Konstrukcja transformacji</i> .....	59
3.2.2 <i>Aspekty implementacyjne</i> .....	60
3.2.3 <i>Przykłady zastosowania</i> .....	62
3.3 PROPOZYCJA SPRĘŻENIA PROCESU DYFUZJI Z PRZESTRZENIĄ HOUGHGA.....	64
3.3.1 <i>Nieliniowy sprzężony proces dyfuzji</i> .....	66
3.3.2 <i>Adaptacyjne wyznaczanie współczynnika sprzężenia <math>h_f</math></i> .....	68
3.3.3 <i>Algorytm filtracji z uwzględnieniem wiedzy o strukturze geometrycznej</i> .....	71
3.4 BADANIE WŁASNOŚCI ALGORYTMU SPRĘŻONEGO .....	72
3.4.1 <i>Skuteczność filtracji dla współczynnika dyfuzji <math>c_1</math> (sprzężenie addytywne)</i> .....	73
3.4.2 <i>Skuteczność filtracji dla współczynnika dyfuzji <math>c_2</math> (sprzężenie multiplikatywno -addytywne)</i> ...	83
3.4.3 <i>Charakterystyka filtracji sprzężonej przy zastosowaniu wartości progowej dla przestrzeni</i>	

transformaty odwrotnej .....	84
3.5 DYSKUSJA I PODSUMOWANIE METOD FILTRACJI SPRZĘŻONEJ .....	88
<b>4. MODELE OPISU PRZESTRZENNEJ ORIENTACJI OBIEKTÓW .....</b>	<b>94</b>
4.1 PARAMETRIZACJA PRZESTRZENI TRANSLACJI I ORIENTACJI .....	96
4.1.1 Interpretacja macierzy rotacji – twierdzenie Eulera .....	98
4.1.2 Sposoby parametryzacji rotacji .....	99
4.1.3 Osobliwości parametryzacji orientacji .....	102
4.1.4 Wizualizacja danych w przestrzeni rotacji .....	103
4.2 ZASTOSOWANIE PRZESTRZENI KWATERNIONÓW .....	105
4.2.1 Podstawowe własności kwaternionów .....	106
4.2.2 Parametryzacja przestrzeni rotacji przez kwaterniony .....	107
4.2.3 Przestrzeń styczna .....	111
4.2.4 Interpolacja kwaternionowa .....	112
4.2.5 Miary odległości w przestrzeni $S^3$ .....	116
4.2.6 Odpowiedniki wielkości statystycznych w przestrzeni kwaternionów .....	117
4.2.7 Generacja wartości pseudolosowych w przestrzeni kwaternionów .....	122
4.3 PODSUMOWANIE SPOSOBÓW PARAMETRIZACJI PRZESTRZENI ORIENTACJI .....	124
4.4 MODEL DANYCH OPISUJĄCYCH ORIENTACJĘ UKŁADÓW .....	126
4.4.1 Model danych trajektorii .....	127
4.4.2 Miary odległości dla trajektorii .....	128
4.5 ADAPTACJA METODY DYNAMICZNEGO ZNIEKSZTAŁCANIA DZIEDZINY CZASU DLA KWATERNIONÓW .....	128
4.5.1 Podstawowy schemat algorytmu .....	129
4.5.2 Uogólnienie pojęcia lokalnej miary podobieństwa elementów trajektorii .....	131
4.5.3 Konstrukcja miary zaszumienia jakości sygnału opartej na algorytmie QDTW .....	133
4.5.4 Przykład działania algorytmu QDTW .....	134
4.5.5 Podsumowanie kwaternionowej wersji algorytmu DTW .....	136
4.6 REPREZENTACJA DANYCH DLA MODELI HIERARCHICZNYCH ŁAŃCUCHÓW KINEMATYCZNYCH .....	137
4.6.1 Zadanie kinematyki prostej i odwrotnej dla łańcuchów kinematycznych .....	138
4.6.2 Przykład złożonego łańcucha kinematycznego .....	141
<b>5. PROPOZYCJE PRZETWARZANIA TRAJEKTORII W PRZESTRZENIACH ROTACJI <math>S^3</math> ....</b>	<b>146</b>
5.1 PRZEGLĄD ISTNIEJĄCYCH METOD PRZETWARZANIA TRAJEKTORII W PRZESTRZENIACH ORIENTACJI ...	148
5.1.1 Metody oparte na liniowym przetwarzaniu szeregu czasowego w przestrzeni stycznej $R^3$ .....	148
5.1.2 Iteracyjne metody minimalizacji energii trajektorii .....	151
5.1.3 Pozostałe metody filtracji w przestrzeni kwaternionów .....	154
5.2 WYKAZANIE ZWIĄZKU FILTRÓW PRZETWARZAJĄCYCH ORIENTACJĘ UKŁADÓW Z ROZWIĄZANIEM	
LINIOWEGO RÓWNIANIA PROCESU DYFUZJI .....	156
5.2.1 Sformułowanie procesu dyfuzji .....	156
5.2.2 Dyskretyzacja modelu liniowego .....	158
5.2.3 Konstrukcja algorytmu filtracji i aspekty implementacyjne .....	160
5.2.4 Podstawowe własności procesu .....	161

5.3	UOGÓLNIENIE NIELINIOWEGO PROCESU DYFUZJI DLA PRZETWARZANIA ORIENTACJI UKŁADU W PRZESTRZENI $S^3$ .....	163
5.3.1	<i>Dyskretyzacja modelu nieliniowego</i> .....	165
5.3.2	<i>Konstrukcja algorytmu filtracji nieliniowej</i> .....	166
5.4	PROPOZYCJA ANIZOTROPOWEGO PROCESU DYFUZJI DLA TRAJEKTORII W PRZESTRZENI ORIENTACJI... ..	168
5.4.1	<i>Cel wprowadzenia współczynnika uwzględniającego kierunek</i> .....	169
5.4.2	<i>Sformułowanie procesu filtracji</i> .....	169
5.4.3	<i>Algorytm filtracji anizotropowej</i> .....	172
5.5	GEOMETRYCZNA INTERPRETACJA METOD OPARTYCH NA RÓWNANIACH DYFUZJI .....	175
5.5.1	<i>Algorytm filtracji liniowej LinQuat</i> .....	176
5.5.2	<i>Algorytm filtracji liniowej NonLinQuat</i> .....	178
5.5.3	<i>Algorytm filtracji anizotropowej AnisoQuat</i> .....	181
5.6	WŁASNOŚCI FILTRACJI DLA TRAJEKTORII W $S^3$ OPARTYCH NA RÓWNANIU RÓZNICZKOWYM DYFUZJI. ..	184
5.6.1	<i>Liniowość przekształcenia</i> .....	185
5.6.2	<i>Stacjonarność przekształcenia (ang. time-invariance)</i> .....	187
5.6.3	<i>Niezmienniczość względem transformacji (ang. coordinate-invariance)</i> .....	188
5.6.4	<i>Ograniczenie przestrzennego rozrzutu</i> .....	190
5.7	PODSUMOWANIE PROPOZYCJI FILTRÓW PRZETWARZAJĄCYCH TRAJEKTORIE W PRZESTRZENI $S^3$ .....	193
<b>6.</b>	<b>WYNIKI BADAŃ EKSPERYMENTALNYCH.....</b>	<b>195</b>
6.1	STOSOWANA PROCEDURA BADAWCZA .....	195
6.1.1	<i>Zastosowane trajektorie testowe</i> .....	196
6.1.2	<i>Złożone sekwencje testowe</i> .....	201
6.1.3	<i>Metody oceny wyników przetwarzania</i> .....	203
6.2	SKUTECZNOŚĆ ALGORYTMU LINQUAT DLA SYNTETYCZNYCH TRAJEKTORII TESTOWYCH .....	203
6.2.1	<i>Skuteczność filtracji LQ w zależności od współczynnika dyfuzji c</i> .....	204
6.2.2	<i>Skuteczność filtracji LQ w zależności od stopnia zaszumienia trajektorii</i> .....	208
6.2.3	<i>Skuteczność filtracji LQ w zależności od warunku brzegowego</i> .....	210
6.2.4	<i>Podsumowanie badań dla filtracji LinQuat</i> .....	212
6.3	SKUTECZNOŚĆ ALGORYTMU NONLINQUAT DLA SYNTETYCZNYCH TRAJEKTORII TESTOWYCH.....	212
6.3.1	<i>Skuteczność filtracji NLQ w zależności od współczynnika wrażliwości K</i> .....	213
6.3.2	<i>Skuteczność filtracji NLQ w zależności od współczynnika <math>\lambda</math></i> .....	218
6.3.3	<i>Podatność filtracji NLQ na poziom zaszumienia sygnału</i> .....	219
6.3.4	<i>Podsumowanie badań dla filtracji NonLinQuat</i> .....	220
6.4	SKUTECZNOŚĆ ALGORYTMU ANISOQUAT DLA SYNTETYCZNYCH TRAJEKTORII TESTOWYCH.....	220
6.4.1	<i>Skuteczność filtracji AQ w zależności od współczynników wrażliwości</i> .....	221
6.4.2	<i>Regulacja szybkości dyfuzji filtracji AQ</i> .....	228
6.4.3	<i>Rezultaty filtracji AQ w zależności od poziomu zaszumienia sygnału</i> .....	230
6.4.4	<i>Wpływ przyjętego warunku brzegowego na przebieg filtracji</i> .....	231
6.4.5	<i>Filtracja AQ dla skomplikowanej trajektorii testowej typu QuatMulti</i> .....	232
6.4.6	<i>Porównanie własności filtracji algorytmami LinQuat oraz AnisoQuat</i> .....	235

6.4.7 Podsumowanie badań dla filtracji AnisoQuat.....	241
6.5 SKUTECZNOŚĆ FILTRACJI DLA SEKWENCJI RUCHU REALIZUJĄCYCH ZŁOŻONY MODEL HIERARCHICZNY .....	242
<b>7. PODSUMOWANIE .....</b>	<b>248</b>
<b>BIBLIOGRAFIA.....</b>	<b>252</b>
<b>PUBLIKACJE WŁASNE.....</b>	<b>252</b>
<b>SPIS ILUSTRACJI.....</b>	<b>259</b>
<b>WYKAZ ALGORYTMÓW .....</b>	<b>263</b>
<b>WYKAZ OZNACZEŃ.....</b>	<b>264</b>

## **Rozdział 1**

### **Wprowadzenie**

Przetwarzanie wszelkiego rodzaju sygnałów jest bardzo intensywnie rozwijającym się działem nauki. Pod pojęciem przetwarzania rozumiane są najczęściej metody, które mają za zadanie filtrację, kompresję lub kodowanie informacji. Wiele metod jest częściami składowymi większych systemów przetwarzających, także realizujących działania w sposób automatyczny. W wyniku tego możliwe jest wydobycie z badanych sygnałów nowych, trudnych do pierwotnego odczytania informacji lub też uzyskanie pożądanej zmiany postaci danych. To z kolei pozwala na efektywne (pod względem szybkości, zwiększenia bezpieczeństwa, etc.) przesyłanie lub dalsze przetwarzanie różnego typu sygnałów.

Ze względu na zastosowania praktyczne widoczne jest coraz większe zapotrzebowanie na systemy przetwarzania informacji wizualnej. Dotyczy to zarówno pojedynczych obrazów statycznych, jak i sekwencji obrazów reprezentujących ruch. Przetwarzanie tego typu danych staje się coraz bardziej popularne także ze względu na rosnące możliwości techniczne i dostępną moc obliczeniową dzisiejszych komputerów.

Przetwarzanie obrazów ma duże znaczenie praktyczne w wielu współczesnych dziedzinach. Istnieje wiele zastosowań przemysłowych wykorzystujących automatyczne systemy analizy obrazów. Wśród nich można wymienić problematykę kontroli jakości produktów lub weryfikację poprawności przebiegu procesu. Analiza strumieni wideo staje także coraz bardziej istotna w automatycznych systemach bezpieczeństwa i kontroli dostępu. Równie powszechne staje się wykorzystywanie metod przetwarzania obrazów w przemyśle rozrywkowym, w szczególności na użytek produkcji filmowych lub przy tworzeniu gier komputerowych.

Początkowo przetwarzanie danych wizualnych było oparte wyłącznie na analizie klasycznego obrazu dwuwymiarowego. Dalszy rozwój techniki pozwolił na zapoczątkowanie badań nad dwuwymiarowym obrazem ruchomym. Dzięki temu możemy dzisiaj korzystać z wielu osiągnięć w dziedzinie skutecznej kompresji sekwencji obrazów w postaci filmów i animacji.

Od pewnego czasu obserwuje się nową tendencję w dziedzinie przetwarzania sygnałów, która rozpatruje informacje reprezentowane w wirtualnej trójwymiarowej przestrzeni modelującej przestrzeń rzeczywistą. Podejście takie jest wykorzystywane od dawna w dziedzinie automatyki i robotyki w problemach związanych przykładowo z planowaniem trasy ruchu manipulatora. W dziedzinie przetwarzania i analizy obrazu wizyjnego uwzględnienie trzech wymiarów oraz dodatkowego wymiaru czasowego jest intensywnie rozwijane.



## **Motywacja**

Problematyka przetwarzania sekwencji danych reprezentujących położenie i orientację obiektu w przestrzeni często związana jest z opisem ruchu obiektu modelowanego przez hierarchiczne połączenie elementów łańcucha kinematycznego. W wielu przypadkach składowa orientacji ma dominujące znaczenie w rozpatrywanym problemie. Metody opisu położenia w danej chwili czasowej tego typu elementów wykorzystują najczęściej przestrzeń rotacji przy zastosowaniu różnych sposobów parametryzacji. Jednym z ważniejszych sposobów parametryzacji przestrzeni rotacji układu są kwaterniony. Sekwencję danych reprezentującą rotacyjny ruch obiektu w przestrzeni można w związku z tym interpretować jako trajektorię opisaną w przestrzeni kwaternionów.

Należy zaznaczyć, że wybór kwaternionów, jako parametryzacji przestrzeni nie ogranicza rozważanego problemu tylko do tego typu trajektorii. Kwaterniony są wygodnym sposobem opisu rotacji, który może być zrealizowana także przy użyciu innej parametryzacji (np. macierze rotacji, kąty Eulera). Jednakże ze względu na interesujące własności tej parametryzacji będzie ona używana w prezentowanej pracy jako podstawowy sposób opisu rotacji. Szczegółowa zagadnienie analizowane jest w rozdziale 4.

Trajektorie punktów reprezentujących dane w przestrzeni rotacji (w szczególności w postaci reprezentacji kwaternionowej) mogą być otrzymywane przez analizę skorelowanych dwuwymiarowych sekwencji ruchu obserwowanego obiektu. W tego rodzaju torze przetwarzającym mogą wystąpić różnego rodzaju zakłócenia, które w efekcie powodują błędne określenie umiejscowienia punktów trajektorii. Wtedy niezbędne może się okazać zastosowanie algorytmu filtracji danych. Filtrację można przeprowadzać na dwóch etapach: w odniesieniu do obrazów lub w odniesieniu bezpośrednio do trajektorii.

Większość metod filtracji obrazów oparta jest na analizie lokalnej zawartości sąsiedztwa w przestrzeni jasności pikseli. Jednakże często można określić dodatkowe globalne informacje o strukturach geometrycznych znajdujących się na przetwarzanym obrazie. Sytuacja taka może mieć przykładowo miejsce także przy analizie obrazów dwuwymiarowych służącej przygotowaniu trajektorii w przestrzeni rotacji. Uwzględnienie dodatkowych globalnych informacji o strukturach geometrycznych obecnych na obrazie może być pomocne w przeprowadzeniu filtracji oraz określeniu poszukiwanych cech znaczeniowych. Pośrednią motywacją rozprawy jest przeprowadzenie badań nad możliwością uwzględnienia dodatkowych informacji geometrycznych w procesie filtracji obrazów.

Badanych jest wiele podejść, których celem jest prawidłowe odwzorowanie położenia punktu w przestrzeni na etapie analizy jednego lub wielu obrazów dwuwymiarowych. Dokładność odwzorowania można zwiększyć przez filtrację danych opisujących ruch przeprowadzoną bezpośrednio w przestrzeni rotacji. Pozwala to na uwzględnienie specyficznych własności wykonywanego ruchu bazując na zapisie trajektorii punktów. W takim przypadku oraz w dalszych rozważaniach dotyczących tej tematyki zakłada się, że przetwarzanie realizowane jest dla trajektorii, która już została przekształcona do przestrzeni rotacji.

Problematyka filtracji trajektorii w przestrzeni rotacji nie została jeszcze intensywnie zbadana. Istnieje kilka propozycji metod filtracji, które wykorzystują wybrane własności parametryzacji kwaternionowej. Przegląd istniejących metod zostanie zaprezentowany w rozdziale 5.1. Oparte są one w

głównej mierze na minimalizacji energii ruchu, rozpatrywanej jako miara krzywizny trajektorii. Podejście takie jest analogią do klasycznych metod przetwarzania danych w przestrzeni rzeczywistej. Jednakże opracowane na tej podstawie algorytmy nie wykorzystują w pełni specyfiki danych dla przestrzeni rotacji.

W wielu zastosowaniach praktycznych występuje potrzeba przeprowadzenia filtracji sygnałów o dużym stopniu zaszumienia. Opracowanie metody, która wykorzystuje specyficzne własności sygnału zapisanego w przestrzeni rotacji pozwoli na uzyskanie wysokiej jakości przefiltrowanych danych wynikowych. Metoda taka może być następnie zastosowana również w przypadku bardziej złożonej analizy ruchu, której celem jest uwypuklenie wybranych cech trajektorii. Motywacją pracy jest przeprowadzenie badań w celu opracowania i sprawdzenia własności tego typu metody ze szczególnym uwzględnieniem własności trajektorii w przestrzeni rotacji.

## Cel rozprawy

Intestywnie rozwijaną gałęzią dziedziny przetwarzania obrazów jest podejście oparte na wykorzystaniu rozwiązania równania różniczkowego o pochodnych cząstkowych opisującego proces dyfuzji ciepła. W tego typu filtracji przetwarzany obraz wykorzystuje się jako warunek początkowy dla rozwiązania równania wyznaczanego numerycznie. Numeryczne rozwiązanie podstawowego równania dyfuzji pozwala skonstruować filtr liniowy. Działanie filtracji liniowej jest równoważne wyznaczeniu splotu jądra gaussowskiego z przetwarzanym obrazem.

Filtracja liniowa ma jednak szereg wad, z których najistotniejsza polega na rozmyciu krawędzi, czyli zmniejszeniu zawartości znaczeniowej obrazu. Wprowadzenie nieliniowego, niejednostajnego czynnika do zmiennej sterującej procesem pozwoliło na skonstruowanie modelu nazwanego od nazwisk twórców *Perona-Malik* [79]. Podejście to uwzględnia lokalne występowanie istotnych informacji na przetwarzanym obrazie reprezentowanych przez krawędzie. Algorytm wykorzystujący ten model pozwala na realizację filtracji, która zamiast rozmywać, wyostrza krawędzie przy jednoczesnym usuwaniu zakłóceń z obrazu. Model ten jest znacząco różny od klasycznych filtrów, które najczęściej działają jednostronnie: albo usuwając zakłócenia albo wyostrzając obraz. Dalsze badania w tej dziedzinie pozwoliły na stworzenie w pełni anizotropowego modelu filtracji charakteryzującego się jeszcze lepszymi własnościami estymacji lokalnych cech obrazu [107]. Niezbadaną w znacznym stopniu gałęzią badań nad filtrami opartymi na równaniach różniczkowych jest podejście uwzględniające nie tylko lokalne cechy, ale także globalny opis zawartości obrazu. W literaturze nie występują również prace, które opierając się na filtracji wykorzystującej równanie dyfuzji ciepła, uogólniają metodę w celu konstrukcji algorytmu działającego na trajektoriach kwaternionów.

Podejście wykorzystywane w filtracji obrazów dwuwymiarowych oparte na rozwiązaniu równania dyfuzji będzie punktem wyjścia dla problematyki, której dotyczy prezentowana praca. Celem pracy jest zweryfikowanie tezy o możliwości rozwiązania następujących problemów:

- Konstrukcja metody filtracji uwzględniającej globalną (z góry określoną) wiedzę o strukturze geometrycznej obrazu.
- Uogólnienie metod filtracji opartej na równaniu różniczkowym dyfuzji dla trajektorii rotacji opisanej w przestrzeni kwaternionów

- Uwzględnienie w procesie filtracji trajektorii kwaternionów specyficznych cech decydujących o charakterze i zawartości znaczeniowej badanego sygnału.

## Schemat rozprawy

Prezentowana rozprawa podzielona została na dwie logiczne części. *Część pierwsza* dotyczy problematyki przetwarzania obrazów, w której zaprezentowano zarówno przegląd literaturowy, jak i propozycje własne autora. Należą do niej następujące rozdziały:

**Rozdział 2**, w którym zostanie zaprezentowany przegląd literaturowy metod filtracji obrazów dwuwymiarowych ze szczególnym uwzględnieniem modelu wykorzystującego równanie różniczkowe dyfuzji ciepła. W rozdziale tym zostaną również zaprezentowane przykładowe wyniki działania filtracji opartej na modelu *Perona-Malik*.

**Rozdział 3** zawiera propozycję konstrukcji metody filtracji, w której uwzględniona zostaje globalna wiedza o strukturze geometrycznej obrazu. Zostanie przedstawiona propozycja sprzężonego algorytmu filtracji oraz wyniki przeprowadzonych eksperymentów numerycznych.

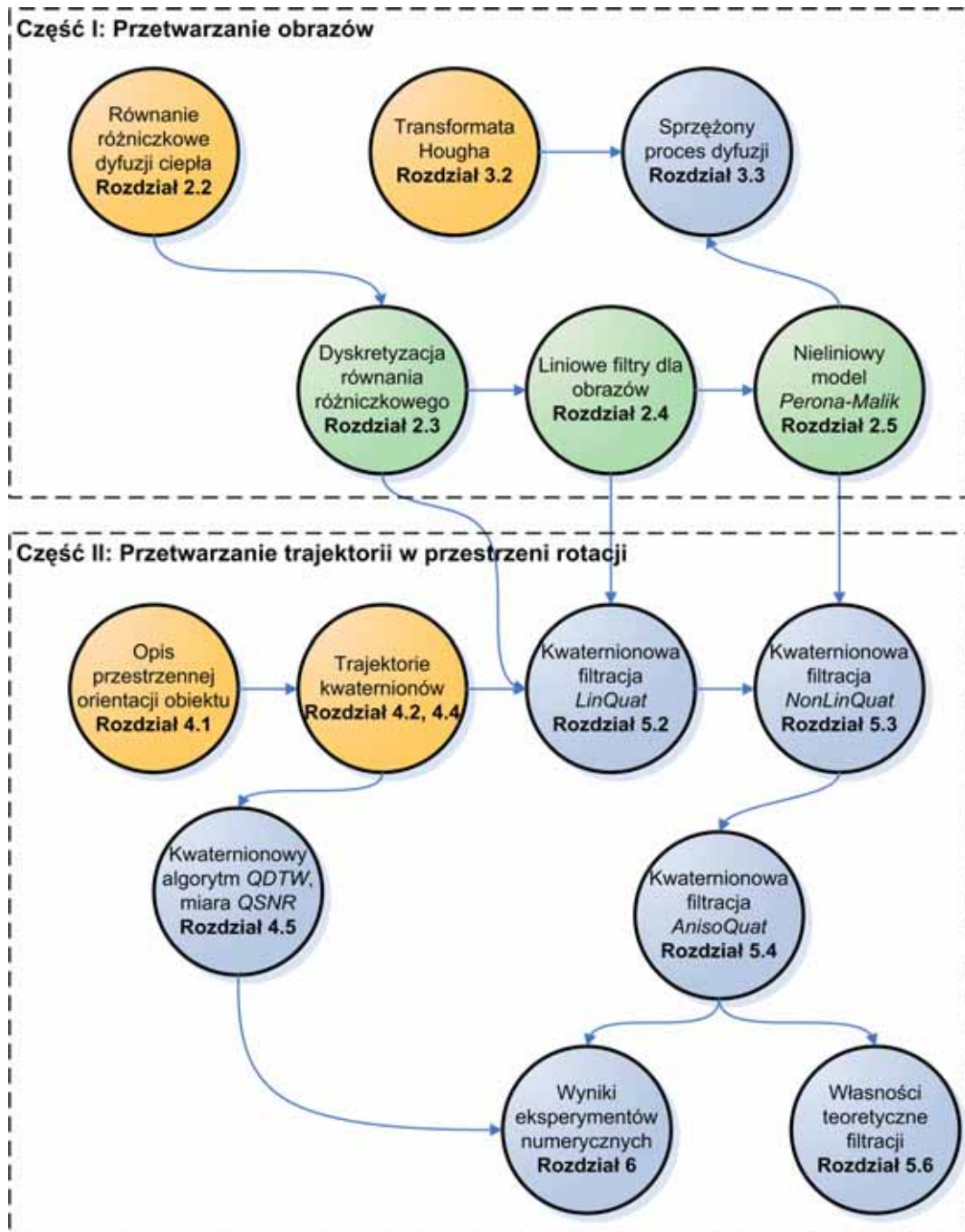
Pierwsza część rozprawy jest punktem wyjścia do dalszej analizy możliwości konstrukcji algorytmów dla trajektorii w przestrzeni rotacji. W *drugiej części* przystępuje się do uogólnienia metod zaprezentowanych w rozdziale 2. Składa się ona z następujących rozdziałów:

**Rozdział 4**, w którym zostaną przedstawione modele opisu orientacji układu w przestrzeni ze szczególnym uwzględnieniem możliwości oraz motywacji użycia do tego celu przestrzeni kwaternionów. Zostaną przedstawione podstawowe własności kwaternionów, które będą wykorzystywane w dalszej części pracy. W rozdziale zostanie również przedstawione nowatorskie uogólnienie metody porównywania trajektorii kwaternionów opartej na dynamicznym zniekształcaniu dziedzin czasu (ang. *Dynamic Time Warping*).

**Rozdział 5** dotyczy metod przetwarzania i filtracji trajektorii w przestrzeniach rotacji ze szczególnym wykorzystaniem własności przestrzeni kwaternionów. Zostaną przedstawione propozycje uogólnienia dyskretnej postaci liniowego i nieliniowego równania różniczkowego dyfuzji do przetwarzania trajektorii kwaternionowej. W rozdziale zostanie zaprezentowana propozycja konstrukcji algorytmu uwzględniającego specyficzne własności trajektorii kwaternionów takie, jak zmiana lokalnej krzywizny sygnału. Rozdział zakończony jest analizą podstawowych własności teoretycznych proponowanych algorytmów.

**Rozdział 6** zawiera opracowane wyniki eksperymentów numerycznych przeprowadzonych w celu weryfikacji i porównania właściwości algorytmów filtracji trajektorii kwaternionowych. Skuteczność algorytmów jest badana zarówno na podstawie syntetycznych trajektorii testowych, jak i na wybranych złożonych sygnałach zarejestrowanych dla ruchów wykonywanych przez człowieka w przestrzeni rzeczywistej.

Na rysunku 1-1 przedstawiony jest diagram reprezentujący strukturę oraz sposób organizacji prezentowanej pracy. Kolorem pomarańczowym zaznaczono elementy pracy, które stanowią podstawę teoretyczną wykorzystywaną w dalszych częściach. Kolorem zielonym zaznaczono składowe pierwszej części pracy, które odnoszą się do problematyki filtracji obrazów. Kolorem niebieskim oznaczone zostały elementy pracy reprezentujące najważniejsze cele uzyskane w ramach rozprawy. Szczegółowe omówienie otrzymanych rezultatów zostanie zebrane i przedstawione w rozdziale 7. W rozdziale tym zawarta zostanie również analiza możliwości dalszego rozwoju i wykorzystania zaprezentowanych metod.



Rysunek 1-1: Diagram przedstawiający schemat konstrukcji prezentowanej rozprawy.

# **Część I**

## **Przetwarzanie obrazów**

## **Rozdział 2**

# **Porównanie metod filtracji dla obrazów dwuwymiarowych**

Przetwarzanie obrazów może być bardzo szeroko rozumiane i obejmuje procesy przesyłania, zapisywania i modyfikowania informacji opisujących obraz. Rozwój techniki przetwarzania jest ściśle powiązany z wieloma innymi dyscyplinami naukowymi, a w jej początkach przede wszystkim z rozwojem optyki [108]. Znaczącym momentem dla rozwoju techniki przetwarzania obrazów są lata 30-te XIX wieku, kiedy rodziła się klasyczna fotografia. Przez kolejne dziesiątki lat przetwarzanie obrazów miało charakter typowo analogowy opierając się przede wszystkim na rozwoju elementów optycznych oraz doskonaleniu materiałów światłoczułych i procesu ich obróbki. Wiek XX przyniósł nowe wyzwania, a także nowe możliwości związane przede wszystkim z pojawieniem się telewizji oraz nowych sposobów pozyskiwania obrazów (rozwój mikroskopii, metod diagnostycznych itp.). Rozwój techniki obrazowej jest wielotorowy i obejmuje zarówno nowe, sformalizowane koncepcje dotyczące optyki, jak i coraz powszechniej wprowadzanie możliwości związanych z sygnałami wysokiej częstotliwości. Pojawienie się optoelektroniki spowodowało zmianę w podejściu na rzecz przetwarzania obrazów w postaci sygnałów elektronicznych. Jednocześnie bardzo intensywnie rozwijała się technika cyfrowa. Lata 80-te i 90-te stanowiły przełom także w technice przetwarzania obrazów na rzecz ich obróbki cyfrowej. W dzisiejszych czasach większość systemów przetwarzających obraz składa się z dwóch części:

- Analogowej odpowiedzialnej za właściwe przekształcenie obrazu (światła) do postaci cyfrowej. W tym fragmencie toru przetwarzania nadal bardzo istotną rolę spełniają odpowiednio przygotowane elementy optyczne oraz optoelektroniczne;
- Cyfrowej, która realizuje właściwą obróbkę obrazu, także najczęściej połączoną z jego kompresją i zapisem w postaci cyfrowej.

Zmniejszenie kosztów produkcji specjalizowanych układów elektronicznych przetwarzających sygnał wideo, powoduje w ostatnich latach upowszechnienie urządzeń do przetwarzania nie tylko pojedynczych obrazów, ale całych sekwencji rejestrowanych w postaci filmów. Rozwój technik numerycznych, w tym nowych metod kompresji, pozwala z kolei na coraz wydajniejsze zapisywanie i przesyłanie danych reprezentujących obraz.

Jednym z najczęściej wykorzystywanych sposobów przetwarzania obrazów jest filtracja obrazów. Jest ona coraz częściej stosowana jako składnik bardziej złożonych systemów przetwarzania informacji. System taki wykorzystuje algorytmy filtracji jako pierwszy stopień przetwarzający. W ten sposób analizowany obraz zostaje oczyszczony z niepożądanych informacji. Informacje takie pojawiają się najczęściej w wyniku zakłóceń związanych z przetwarzaniem lub przesyłaniem obrazu pomiędzy urządzeniami (np. w trakcie rejestracji obrazu przez elementy światłoczułe kamery rejestrującej lub w torze przesyłowym). Jednocześnie filtracja może być sposobem na wyodrębnienie informacji, które są najważniejsze w dalszych etapach przetwarzania (np. wyodrębnienie lub wyostrenie krawędzi, których kształt jest następnie rozpoznawany).

Cyfrowy sposób przetwarzania obrazów wymaga odpowiedniej dyskretnej formy opisu obrazu. W kolejnym podrozdziale zostaną krótko przedstawione sposoby reprezentacji obrazów, których użycie ma bezpośredni wpływ na konstrukcję algorytmów filtracji. W następnym podrozdziale zebrane zostaną liniowe i nieliniowe metody przetwarzania obrazów, których konstrukcja oparta jest na rozwiązaniu równania różniczkowego cząstkowego. W ostatnim podrozdziale zostaną podsumowane możliwości oraz zaproponowane sposoby wprowadzenia apriorycznej wiedzy do algorytmów filtrujących obraz.

## 2.1 Wprowadzenie do sposobów reprezentacji danych

Najczęściej obraz opisuje się jako funkcję  $f_I$  operującą na przestrzeni pikseli ( $P$ ) przekształcającą ją w przestrzeń intensywności (w ogólności przestrzeń kolorów) ( $C$ ) [93]:

$$f_I : (P) \rightarrow (C) \quad (2-1)$$

Najczęściej spotyka się obrazy 3-spektralne, w których kodowanie barwy opiera się na 3 barwach podstawowych  $R$ ,  $G$ ,  $B$ , z których każda próbkowana jest przy 256 poziomach kwantyzacji. Obrazy z odcieniami szarości (które często stanowią bazę działania wielu algorytmów przetwarzania) posiadają przestrzeń kolorów jednowymiarową o zadanej liczbie poziomów kwantyzacji (256 dla przetwarzania 8-bitowego). Przestrzeń kolorów obrazów binarnych jest dwuelementowa o wartościach  $\{0, 1\}$ .

Przestrzeń pikseli jest zazwyczaj reprezentowana, jako siatka  $n_1 \times n_2$  dyskretnych punktów o następujących własnościach:

- Prototyp piksela definiuje się, jako punkt  $PP = \{(0, 0)\}$
- Zbiór indeksów dla siatki prostokątnej  $IS = IS(0, 0; n_1 - 1, n_2 - 1)$
- Istnieje zdefiniowana izometria  $\alpha_{ij}(0,0) = (i, j)$  przekształcająca prototyp piksela w dowolny punkt z przestrzeni siatki
- W  $IS$  definiuje się relację sąsiedztwa przy użyciu dyskretnej funkcji odległości  $\rho : E_{ij}$  jest sąsiadem  $E_{i'j'}$ , wtedy i tylko wtedy, gdy  $\rho((i, j), (i', j')) = 1$ , np:

$$\circ \quad \rho((i, j), (i', j')) = |i - i'| + |j - j'| - \text{relacja czterosąsiedztwa,}$$

- $\rho((i, j), (i', j')) = \max(|i - i'|, |j - j'|)$  - relacja ośmiosąsiedztwa

W ogólności przestrzeń pikseli nie musi być zdefiniowana na siatce prostokątne, a prototyp piksela nie musi być punktem. Szczegółowy opis oraz właściwości różnych przestrzeni pikseli znajduje się w [93]. W dalszych częściach pracy używany będzie powyżej przedstawiony model, dla którego w przypadku relacji czterosąsiedztwa stosuje się następującą notację:

- N – sąsiad “północny” – od  $(i, j)$  do  $(i, j - 1)$
- S – sąsiad “południowy” – od  $(i, j)$  do  $(i, j + 1)$
- W – sąsiad “zachodni” – od  $(i, j)$  do  $(i - 1, j)$
- E – sąsiad “wschodni” od  $(i, j)$  do  $(i + 1, j)$

Jeżeli nie zaznaczono inaczej, w algorytmach przetwarzających obrazy stosować się będzie przegląd rastrowy, czyli sekwencyjne pobieranie kolejnych elementów obrazów w kolejnych wierszach. Sposób przeglądu obrazu może mieć znaczenie dla własności algorytmów przetwarzających. W szczególności w zagadnieniach kwantyzacji obrazów dobre efekty daje stosowanie przeglądu wzdłuż krzywych fraktalnych (np. Krzywa Hilberta, Peano, ghill, itp.). Szczegółowo zagadnienie własności krzywych wypełniających jest przedstawione w [93].

### Miary odległości dla pikseli

W przypadku porównywania dwóch obrazów niezbędne jest użycie zdefiniowanej miary bliskości obrazów. Miara taka wymaga natomiast określenia metryki odległości dla dwóch elementów obrazu (pikseli) w przestrzeni kolorów. Dla 3-spektralnych obrazów podstawowe metryki w przestrzeni kolorów zdefiniowane są następująco [93]:

- Odległość euklidesowa:  $d_E(C_1, C_2) = \left( \sum_{i=1}^3 w_i (C_1^{(i)} - C_2^{(i)})^2 \right)^{1/2}$ , (2 - 2)

- Odległość Manhattan:  $d_M(C_1, C_2) = \sum_{i=1}^3 w_i |C_1^{(i)} - C_2^{(i)}|$ , (2 - 3)

- Odległość maksimum:  $d_{\max}(C_1, C_2) = \max_{1 \leq i \leq 3} \{w_i |C_1^{(i)} - C_2^{(i)}|\}$ , (2 - 4)

gdzie  $C = (C^{(1)}, C^{(2)}, C^{(3)})^T$  jest wektorem jasności piksela, a  $w_i$  oznacza dodatnią wagę.

Jeżeli nie zaznaczono inaczej, w dalszej części pracy używana będzie odległość euklidesowa ze stałymi wagami o wartości 1 (przyjmuje się oznaczenie  $d = d_E$ ). Ponadto większość algorytmów zostanie zaprezentowana dla obrazów monochromatycznych, ze względu na prostszy opis, który ma służyć jako punkt wyjścia do konstrukcji metod dla danych reprezentujących ruch w przestrzeni.

### Miary odległości dla obrazów

Prezentowane w pracy algorytmy przetwarzające obrazy należą do grupy algorytmów rekonstrukcji danych oryginalnych na podstawie obserwacji. W praktyce testowanie algorytmów polega na zaszumieniu



obrazu testowego (w ogólności wprowadzeniu zakłóceń o określonym charakterze), uruchomieniu procedury rekonstruującej, a na końcu porównaniu otrzymanego wyniku z obrazem oryginalnym. Najczęściej celem rekonstrukcji obrazu jest otrzymanie wyniku, który będzie jak najbardziej zbliżony do oryginału. Do określenia podobieństwa obrazów konieczne jest zdefiniowanie miar bliskości pomiędzy nimi. Problem porównywania obrazów jest skomplikowany ze względu na sposób działania oka ludzkiego. Podstawowe miary odległości nie odnoszą się do zawartości obrazów, a jedynie do ich matematycznych reprezentacji. Na potrzebę indywidualnych zastosowań powstały bardziej złożone metody porównywania obrazów, które częściowo próbują uwzględnić specyficzną charakterystykę czułości oka na zawartość szumu w obrazach (np. wykresy Hosaka, H-wykresy).

W prezentowanej pracy tradycyjne miary podobieństwa wystarczą do przeanalizowania podstawowych własności metod filtracji. Będą również służyć do konstrukcji analogicznych miar dla porównywania sekwencji ruchów. Ze standardowych sposobów określania podobieństwa obrazów należy wymienić następujące (we wszystkich definicjach przyjmuje się, że oba obrazy mają jednakowy rozmiar, a  $N$  oznacza całkowitą liczbę pikseli każdego z nich):

- Błąd średniokwadratowy:

$$MSE(f_{I_0}, f_{I_1}) = \frac{1}{N} \sum_{(i,j) \in IS} d^2(f_{I_0}(i,j), f_{I_1}(i,j)) \quad (2-5)$$

- Znormalizowany błąd średniokwadratowy:

$$\bar{f}_I = \frac{1}{N} \sum_{(i,j) \in IS} f_I(i,j) \quad (\text{wartość średnia obrazu}) \quad (2-6)$$

$$\sigma_I^2 = \frac{1}{N} \sum_{(i,j) \in IS} d^2(f_I(i,j), \bar{f}_I) \quad (\text{wariancja obrazu}) \quad (2-7)$$

$$NMSE(f_{I_0}, f_{I_1}) = \frac{MSE(f_{I_0}, f_{I_1})}{\sigma_{I_0}^2} \quad (\text{znormalizowany błąd średniokwadratowy}) \quad (2-8)$$

- Stosunek sygnału do szumu (ang. *signal to noise ratio*):

$$SNR(f_{I_0}, f_{I_1}) = 10 \log_{10} \frac{\sigma_{I_0}^2}{MSE(f_{I_0}, f_{I_1})} = -10 \log_{10} NMSE(f_{I_0}, f_{I_1}) [dB] \quad (2-9)$$

Ostatnia miara ma interpretację energetyczną przy założeniu, że wariancję sygnału  $I_0$  określa energię sygnału, natomiast różnica  $(f_{I_0} - f_{I_1})$  jest sygnałem szumu. W prezentowanej pracy w eksperymentach najczęściej wykorzystywana będzie miara (2-9) w szczególności do określania jakości obrazów odfiltrowanych w odniesieniu do obrazów oryginalnych.

## 2.2 Matematyczny opis procesów dyfuzyjnych

Metody filtracji obrazów, które zostaną zaprezentowane w pracy są realizacją matematycznych zależności wynikających z właściwości procesu dyfuzji ciepła. Model oparty na parabolicznym równaniu

różniczkowym dyfuzji ciepła został wybrany ze względu na jego własności filtracyjne. W literaturze rozważane są także modele oparte na równaniach różniczkowych innego typu, jednakże równanie paraboliczne wykazuje najlepsze własności redukcji zakłóceń oraz znaczną stabilność dla problemów filtracji obrazów.

Jak to zostanie pokazane w rozdziale 2.4 zastosowanie modelu dyfuzyjnego pozwala na uzyskanie matematycznego opisu dla wielu metod realizowanych przez algorytmy filtracji liniowej. Ponadto dzięki takiemu podejściu analiza przebiegu filtracji może być przeprowadzona przez analizę własności otrzymanego rozwiązania równania różniczkowego.

Dzięki zdefiniowaniu modelu w postaci zagadnienia początkowego możliwa była konstrukcja nowych metod filtracji nieliniowych, które zaprezentowane zostaną w rozdziale 2.5. Metody te zostaną następnie uogólnione do konstrukcji algorytmów filtracji trajektorii przestrzennych w rozdziale 5.

Szczegółowy opis własności równań dyfuzji, a także ich fizyczna interpretacja znajduje się w [49]. W podrozdziale zostanie krótko przedstawiony opis procesu dyfuzji niezbędny do dalszej konstrukcji metod filtracji opartych na równaniach różniczkowych cząstkowych.

W dalszych przekształceniach używane będą następujące oznaczenia:

Gradient funkcji skalarnej  $f(x, y, z)$

$$\text{grad}(f) = \nabla f = \left[ \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right] \quad (2 - 10)$$

Dywergencja pola wektorowego  $\vec{F}$  dla współrzędnych kartezjańskich,

$$\text{div}(\vec{F}) = \nabla \cdot \vec{F} = \frac{\partial F_x}{\partial x} + \frac{\partial F_y}{\partial y} + \frac{\partial F_z}{\partial z} \quad (2 - 11)$$

Laplasjan funkcji skalarnej  $f(x, y, z)$  dla współrzędnych kartezjańskich

$$\text{div}(\text{grad}(f)) = \nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} \quad (2 - 12)$$

Obszar, na którym rozpatrywane będzie równanie różniczkowe oznaczany będzie przez  $\Omega$ , a powierzchnia ograniczająca przez  $\partial\Omega$ . Przy rozpatrywaniu procesu dyfuzji cieplnej, rozpatrywana jest funkcja  $\hat{T}$  pola temperatury na obszarze  $\Omega$ . W przypadku ogólnym rozpatrywane jest równanie różniczkowe dla funkcji  $u$  na obszarze  $\Omega$ .

Prawo Fouriera określa wektor  $\vec{Q}$  strumienia ciepła wynikającego z istniejącego pola temperatur:

$$\vec{Q} = -D\nabla\hat{T}, \quad (2 - 13)$$

gdzie  $\hat{T} = \hat{T}(P, t)$  określa temperaturę w punkcie  $P(x, y, z) \in \Omega$  dla chwili czasowej  $t$ , a  $D$  jest współczynnikiem przewodzenia ciepła. Postać współczynnika  $D$  będzie zależeć od rodzaju dyfuzji, która jest opisywana równaniem. Z bilansu mocy cieplnych można wyprowadzić równanie różniczkowe cząstkowe przewodzenia ciepła w postaci:

$$\text{div}(D\nabla\hat{T}) - C_w \delta \frac{\partial \hat{T}}{\partial t} = -\gamma(P, t), \quad (2 - 14)$$

gdzie  $C_w$  jest ciepłem właściwym środowiska,  $\delta$  jego gęstością, natomiast  $\gamma(P, t)$  określa wydajność przestrzennych źródeł ciepła. Jest to równanie różniczkowe typu parabolicznego.

W dalszych rozważaniach przyjmuje się, że w rozpatrywanym obszarze nie ma przestrzennych źródeł ciepła, czyli  $\gamma(P, t) = 0$ :

$$\operatorname{div}(D\nabla\hat{T}) = C_w\delta\frac{\partial\hat{T}}{\partial t}. \quad (2 - 15)$$

Dla przypadku izotropowego kierunki wektorów  $\vec{Q}$  oraz  $\nabla\hat{T}$  są zgodne, więc współczynnik dyfuzji  $D$  definiuje się, jako wartość skalarną. Zakładając, że współczynnik dyfuzji jest stały względem pola temperatur, możliwe jest przekształcenie równania (2 - 15) do postaci:

$$\nabla^2\hat{T} = \xi\frac{\partial\hat{T}}{\partial t}, \quad (2 - 16)$$

gdzie  $\xi = \frac{C_w\delta}{D}$ .

W ogólności wektory  $\vec{Q}$  oraz  $\nabla\hat{T}$  nie muszą być równoległe, co prowadzi do *anizotropowego* równania dyfuzji. W takim przypadku współczynnik dyfuzji  $D$  jest tensorem, czyli wielkością będącą uogólnieniem pojęcia wektora. Zastosowanie tensorowego współczynnika dyfuzji pozwala uwzględnić różne kierunki wektorów strumienia ciepła oraz gradientu temperatury. Oznacza to, że kierunek, w jakim zachodzi dyfuzja różni się od kierunku, jaki wyznacza lokalny gradient pola temperatury.

Jeżeli współczynnik dyfuzji  $D$  jest stały na całym obszarze, to mówi się o przypadku dyfuzji *jednorodnej*. Współczynnik  $D$ , który nie jest stały w obszarze definiuje dyfuzję *niejednorodną* – współczynnik może być zależny od pola temperatur w danej chwili czasowej. Prowadzi to do równań różniczkowych *nieliniowych*, dla których nie istnieje w ogólności rozwiązanie analityczne. Współczynnik, który takiej zależności nie ma, definiuje równanie *liniowe*.

W równaniach przewodzenia ciepła opisujących rzeczywiste procesy fizyczne nie spotyka się przypadków, kiedy występuje zależność czasowo-przestrzenna współczynnika  $D$ . Możliwe jest jednak użycie takiego równania do symulacji wirtualnego pola, co zostanie dokładniej opisane w ramach nieliniowych metod przetwarzania obrazów w rozdziale 2.5.

Zagadnienie graniczne dla problemu przewodzenia ciepła składa się z równania różniczkowego (2 - 11) wraz z warunkami granicznymi: warunkiem początkowym oraz warunkiem brzegowym. Warunek początkowy opisuje pole temperatury w rozpatrywanym obszarze  $\Omega$  w chwili  $t = 0$ . Warunek brzegowy określa stan cieplny powierzchni  $\partial\Omega$  ograniczającej obszar  $\Omega$  w chwilach  $t \geq 0$ . Warunek ten może być przedstawiony w różny sposób opisując, w jaki sposób realizowana jest wymiana ciepła z otoczeniem. Ogólnie warunek przedstawiający mieszany charakter wymiany ciepła na granicy obszaru ma postać [49]:

$$A\left(\frac{d\hat{T}}{d\vec{n}_\perp}\right)_P + Bf_1(\hat{T}) + Cf_2(P, t) = 0, \quad (2 - 17)$$

gdzie  $\left(\frac{d\hat{T}}{d\vec{n}_\perp}\right)_P$  jest pochodną w punkcie  $P \in \partial\Omega$  względem skierowanej na zewnątrz normalnej  $\vec{n}_\perp$  do powierzchni  $\partial\Omega$  ograniczającej rozważany obszar, natomiast parametry  $A, B, C$  oraz funkcje  $f_1, f_2$  stanowią o charakterze warunku brzegowego.

Najczęściej spotyka się następujące szczególne przypadki warunku (2 – 17):

- $A = 0, BC \neq 0$  - warunek brzegowy pierwszego rodzaju (warunek Dirchleta), którego szczególnym przypadkiem jest utrzymywanie stałej temperatury na brzegu obszaru  $\Omega$
- $AC \neq 0, B = 0$  - warunek brzegowy drugiego rodzaju (warunek Neumanna), którego szczególnym przypadkiem jest brak wymiany ciepła z otoczeniem  $\left(\frac{d\hat{T}}{d\vec{n}_\perp}\right)_P = 0$ ,
- $AB \neq 0, C = 0$  - warunek brzegowy trzeciego rodzaju odpowiadający wymianie ciepła ze środowiskiem według prawa Newtona.

### 2.3 Algorytmy numerycznego rozwiązywania równań różniczkowych cząstkowych jako filtry cyfrowe

Równanie różniczkowe (2 – 12) w ogólnym przypadku może być nieliniowe ze względu na tensor współczynnika dyfuzji  $D$ . W dodatku, jeżeli tensor ten zależy od aktualnej wartości funkcji  $\hat{T}$ , to nie istnieją metody analitycznego wyznaczenia rozwiązania danego zagadnienia początkowego. W takim przypadku trzeba zastosować numeryczne metody rozwiązywania równań różniczkowych.

W przedstawionych przykładach wykorzystujących równanie dyfuzji używane było naturalne oznaczenie na funkcję pola temperatury  $\hat{T}(P)$ . Dla podkreślenia ogólności rozpatrywanej problematyki, w dalszej części pracy przyjmuje się konwencję, że równanie różniczkowe (w przypadku ciągłym) rozpatrywane jest dla poszukiwanej funkcji  $u(x)$ .

Najczęściej stosowanymi metodami numerycznego wyznaczania rozwiązania zagadnień początkowych są metody oparte na schematach różnicowych oraz metody elementu skończonego. W szczególności te pierwsze okazały się dawać bardzo dobre rezultaty dla algorytmów filtracji obrazów. W dalszej części zostaną przedstawione podstawowe założenia oraz sposoby realizacji najczęściej spotykanych metod numerycznego rozwiązywania równań różniczkowych [59].

#### Metody różnic skończonych

Pierwszym krokiem w metodach różnic skończonych jest przygotowanie obszaru  $\Omega$ , na którym określone jest zagadnienie początkowe. Ciągły obszar  $\Omega$  zastępuje się tzw. funkcją siatkową będącą dyskretnym skończonym zbiorem węzłów oddalonych od siebie o zdefiniowany krok dyskretyzacji  $h_r$  (gdzie  $r$  jest indeksem reprezentującym kolejne wymiary w przestrzeni  $\Omega$ ). Przyjmując, że rozpatrywane

jest równanie różniczkowe dla jednowymiarowej funkcji  $u$ , dyskretyzacja przedziału  $x \in [0, L]$  za pomocą równooddalonych węzłów siatki tworzy zbiór punktów takich, że:

$$u(x_i) \equiv u(ih) \equiv u_i, \quad (2-18)$$

gdzie  $h$  oznacza krok dyskretyzacji<sup>1</sup>,  $M$  – liczbę węzłów siatki takich, że  $L = Mh$  oraz  $i = 0, 1, 2, \dots, M$ .

Tworzenie schematów różnicowych opiera się na rozwinięciach funkcji  $u$  w szeregi Taylora w punkcie  $x_i$ :

$$u(x_i + h) = u(x_i) + h \frac{\partial u}{\partial x} \Big|_{x=x_i} + \sum_{k=2}^{\infty} \frac{h^k}{k!} \frac{\partial^k u}{\partial x^k} \Big|_{x=x_i} \quad (2-19)$$

$$u(x_i - h) = u(x_i) - h \frac{\partial u}{\partial x} \Big|_{x=x_i} + \sum_{k=2}^{\infty} (-1)^k \frac{h^k}{k!} \frac{\partial^k u}{\partial x^k} \Big|_{x=x_i} \quad (2-20)$$

W przedstawionych równaniach celowo wyodrębniono liniowy element szeregu Taylora, aby łatwiejsze było zapisanie ich w następującej postaci:

$$\frac{\partial u}{\partial x} \Big|_{x=x_i} = \frac{u(x_i + h) - u(x_i)}{h} - \sum_{k=2}^{\infty} \frac{h^k}{k!} \frac{\partial^k u}{\partial x^k} \Big|_{x=x_i} \quad (2-21)$$

$$\frac{\partial u}{\partial x} \Big|_{x=x_i} = \frac{u(x_i) - u(x_i - h)}{h} + \sum_{k=2}^{\infty} \frac{h^k}{k!} \frac{\partial^k u}{\partial x^k} \Big|_{x=x_i} \quad (2-22)$$

Pomijając kolejne elementy szeregu otrzymuje się aproksymację pochodnej cząstkowej funkcji  $u$  w punkcie  $x_i$ :

$$\frac{\partial u}{\partial x} \Big|_{x=x_i} \approx \frac{u_{i+1} - u_i}{h} \quad (\text{pierwszy iloraz różnicowy przedni}) \quad (2-23)$$

$$\frac{\partial u}{\partial x} \Big|_{x=x_i} \approx \frac{u_i - u_{i-1}}{h} \quad (\text{pierwszy iloraz różnicowy wsteczny}) \quad (2-24)$$

Stosując przybliżenie (2-23) popełnia się błąd aproksymacji  $O(h)$ . Analogicznie można wyznaczyć:

$$\frac{\partial u}{\partial x} \Big|_{x=x_i} \approx \frac{u_{i+1} - u_{i-1}}{2h} \quad (\text{pierwszy iloraz różnicowy centralny}), \quad (2-25)$$

przy czym popełniany jest błąd aproksymacji  $O(h^2)$ . Dla drugiej pochodnej otrzymuje się:

$$\frac{\partial^2 u}{\partial x^2} \Big|_{x=x_i} \approx \frac{u_{i-1} - 2u_i + u_{i+1}}{h^2} \quad (\text{drugi iloraz różnicowy centralny}) \quad (2-26)$$

Laplasjan funkcji dwóch zmiennych  $\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}$  aproksymuje się następującą różnicą

skończoną [71]:

<sup>1</sup> Analogicznie można zdyskretyzować obszar dla funkcji dwuwymiarowej przyjmując długości kroków dyskretyzacji jako  $h_x, h_y$ . Często przyjmuje się jednakową dla obu wymiarów długość kroków dyskretyzacji  $h_x = h_y = h$ .

$$\begin{aligned}\nabla^2 u &\approx \frac{u_{i+1,j} - u_{i,j} + u_{i-1,j} - u_{i,j} + u_{i,j+1} - u_{i,j} + u_{i,j-1} - u_{i,j}}{h^2} = \\ &= \frac{1}{h^2} (u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j})\end{aligned}\quad (2-27)$$

Przy dyskretyzacji równań prezentowanych w kolejnych rozdziałach przydatny będzie również jawny schemat różnicowy dla nieliniowego równania różniczkowego, w którym również współczynnik dyfuzji  $c$  występujący przy funkcji  $u$  jest zależny od zmiennych  $x, y$ :

$$\frac{\partial u(x, y, t)}{\partial t} = \text{div}(c(x, y, t) \nabla u(x, y, t)) \quad (2-28)$$

Przyjmując oznaczenie  $\Phi$  na funkcję strumienia (oznaczenie jest używane w konkretnych zastosowaniach równania w rozdziale 2.5):

$$\Phi(\nabla u(x, t)) = c(x, t) \nabla u(x, t) \quad (2-29)$$

można w następujący sposób przedstawić dyskretyzację równania różniczkowego (2-28) dla przypadku jednowymiarowego [17]:

$$\begin{aligned}\text{div}[c(x, t) \cdot \nabla u(x, t)] &= \frac{\partial}{\partial x} \left( c(x, t) \cdot \frac{\partial u}{\partial x} \right) \approx \frac{\partial}{\partial x} \left[ c(x, t) \cdot \frac{1}{h} \left( u\left(x + \frac{h}{2}, t\right) - u\left(x - \frac{h}{2}, t\right) \right) \right] \\ &\approx \frac{1}{h^2} \left[ c\left(x + \frac{h}{2}, t\right) \cdot (u(x+h) - u(x)) - c\left(x - \frac{h}{2}, t\right) \cdot (u(x) - u(x-h)) \right] = \Phi_R - \Phi_L\end{aligned}\quad (2-30)$$

przy założeniu, że przyjmuje się krok dyskretyzacji  $h=1$ . Dla przypadku dwuwymiarowego otrzymuje się:

$$\begin{aligned}\text{div}[c(x, y, t) \cdot \nabla u(x, y, t)] &= \frac{\partial}{\partial x} \left( c(x, y, t) \cdot \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( c(x, y, t) \cdot \frac{\partial u}{\partial y} \right) \\ &= \frac{1}{h_x^2} \left[ c\left(x + \frac{h_x}{2}, y, t\right) \cdot (u(x+h_x, y) - u(x, y)) - c\left(x - \frac{h_x}{2}, y, t\right) \cdot (u(x, y) - u(x-h_x, y)) \right] + \\ &\quad \frac{1}{h_y^2} \left[ c\left(x, y + \frac{h_y}{2}, t\right) \cdot (u(x, y+h_y) - u(x, y)) - c\left(x, y - \frac{h_y}{2}, t\right) \cdot (u(x, y) - u(x, y-h_y)) \right] \\ &= \Phi_E - \Phi_W + \Phi_N - \Phi_S\end{aligned}\quad (2-31)$$

przy założeniu równych długości kroków dyskretyzacji  $h_x = h_y = 1$ . Indeksy funkcji  $\Phi$  odnoszą się odpowiednio do sąsiedztwa rozpatrywanego elementu:  $R, L$  – elementy następny, poprzedni (przypadek jednowymiarowy);  $N, S, W, E$  – elementy północny, południowy, zachodni, wschodni zgodnie z konwencją opisaną w rozdziale 2.1 (przypadek dwuwymiarowy).

Podobnie konstruuje się różnice skończone dla warstwy czasowej, gdzie czas jest traktowany, jako dodatkowy wymiar funkcji. W ten sposób można konstruować jawne i niejawne schematy różnicowe opierające się na uwzględnianiu odpowiednich różnic skończonych.

Jawne schematy różnicowe wyznaczają wartość poszukiwanej funkcji siatkowej w danej chwili czasowej jedynie przy użyciu wartości tej funkcji w poprzednich chwilach czasowych. W konstrukcji niejawnych schematów różnicowych występuje natomiast przynajmniej jeden węzeł z warstwy czasowej

funkcji, dla której wartość jest poszukiwana. Wymaga to większego nakładu obliczeniowego i wielokrotnego rozwiązania układu równań.

Metody różnic skończonych mają podstawową zaletę w zagadnieniu przetwarzania obrazu. Biorąc pod uwagę model obrazu, jako siatki pikseli o prototypie punktowym (rozdział 2.1), nie ma konieczności dodatkowego wyznaczania funkcji siatkowej z badanego obszaru. Funkcja taka jest wyznaczona automatycznie i zdeterminowana rozdzielczością analizowanego obrazu. Z drugiej strony nie ma bezpośredniej możliwości zwiększenia kroku kwantyzacji, więc popełniany błąd aproksymacji równania zależeć będzie tylko od przyjętego schematu różnicowego. Sposoby konstrukcji schematów różnicowych w algorytmach przetwarzania obrazów przedstawione są w rozdziałach 2.4, 2.5.

### **Metoda elementu skończonego**

Metoda ta pozwala na zwiększenie dokładności rozwiązania równania różniczkowego cząstkowego przez dokładniejszą aproksymację zadania wyjściowego. Obszar poszukiwanej funkcji  $u$  w zagadnieniu początkowym jest zastępowany przez podobszary nazywane elementami skończonymi. Przyjmuje się, że funkcja  $u$  jest reprezentowana wewnątrz elementów skończonych przez odpowiedni wielomian interpolacyjny, na który nałożone są określone wymagania dotyczące gładkości. Przybliżone rozwiązanie równania jest opisane jako:

$$u(\cdot) = u_0(\cdot) + \sum_{j=1}^N \alpha_j \phi_j(\cdot), \quad (2 - 32)$$

gdzie  $N$  jest skończone,  $u_0(\cdot)$  jest funkcją spełniającą warunek początkowy, a  $\phi_j(\cdot)$  są znanymi funkcjami analitycznymi (tzw. funkcje bazowe). Zadanie aproksymacji metodą elementu skończonego polega na takim doborze współczynników  $\alpha_j$ , aby zminimalizowana była odchyłka pomiędzy postacią przybliżoną (2 – 32), a rozwiązaniem oryginalnego równaniem różniczkowym.

Wśród różnych metod elementu skończonego można wyróżnić metodę Galerkina. Przy pomocy procedury Galerkina można również przekształcić równanie różniczkowe cząstkowe do układu równań różniczkowych zwyczajnych, których numeryczne rozwiązanie jest o wiele łatwiejsze. Realizacją metody elementu skończonego w sformułowaniu wariacyjnym jest metoda Ritza. Implementacja przedstawionych w pracy metod filtracji obrazów opiera się na metodach różnic skończonych, więc ta tematyka nie będzie dalej rozwijana. Szczegółowy opis metod elementu skończonego można znaleźć w [59], [71].

## **2.4 Przestrzenne metody filtracji oparte na procesie dyfuzji**

W poprzednich podrozdziałach zostało przedstawione matematyczne sformułowanie procesów dyfuzyjnych. Procesy opisane równaniami różniczkowymi cząstkowymi mogą być zaadaptowane do przetwarzania obrazów. Przedstawione zostanie, w jaki sposób proces taki może być użyty do filtracji oraz jego powiązanie z tzw. wygładzaniem (rozmywaniem) gaussowskim.

Wszelkie metody filtracji bazujące na równaniach różniczkowych cząstkowych są realizacją następującego modelu ogólnego [1]:

$$\begin{cases} \frac{\partial u(x,t)}{\partial t} + F(x, u(x,t), \nabla u(x,t), \nabla^2 u(x,t)) = 0 & \text{w } (0, T) \times \Omega \\ \frac{\partial u(x,t)}{\partial \vec{n}_\perp} = 0 & \text{na } (0, T) \times \partial\Omega, \\ u(x, 0) = u_0(x) & \text{dla } x \in R^2 \end{cases} \quad (2 - 33)$$

gdzie  $\Omega$  jest rozpatrywanym obszarem, natomiast  $\vec{n}_\perp$  wektorem normalnym do powierzchni ograniczającej obszar  $\partial\Omega$ .

W przedstawionym modelu  $F$  jest operatorem różniczkowym, którego wybór decyduje o właściwościach metody filtracji. Najczęściej przyjmuje się warunek brzegowy przedstawiony w powyższej formie, który jest szczególnym przypadkiem warunku Neumanna opisanego dokładniej równaniem (2 – 17). Tak przyjęty warunek oznacza idealną izolację procesu od otoczenia (dla równania przewodnictwa ciepła oznacza to, że ciało nie wymienia ciepła z otoczeniem).

Warunek początkowy jest najważniejszym elementem pozwalającym na filtrację przy użyciu procesu dyfuzji. Jako dwuwymiarową funkcję  $u_0(x)$  przyjmuje się obraz, który ma być poddany przetwarzaniu. Rozwiązanie zagadnienia początkowego dla określonego czasu  $t$  daje w wyniku filtrację obrazu, który został użyty jako warunek początkowy. Rozwiązując równanie dla kolejnych chwil czasowych  $t > 0$  uzyskuje się całą rodzinę obrazów, które zostały przekształcone z obrazu źródłowego będącego warunkiem początkowym. Każdy kolejny obraz (dla rosnących  $t$ ) jest uproszczoną (przy pewnych założeniach) wersją obrazu  $u_0(x)$ . W ten sposób tworzy się tzw. *przestrzeń skal* (ang. *scale-space*) względem zmiennej  $t$  [67]. Pożądane właściwości tak zdefiniowanej przestrzeni zostaną przedstawione w kolejnych podrozdziałach.

Przetwarzanie obrazów oparte na rozwiązywaniu zagadnienia początkowego (2 – 33) realizuje kilka podstawowych celów:

- minimalizacja zakłóceń znajdujących się na obrazie (które zazwyczaj występują w postaci szumu o określonym charakterze),
- wyostrenie cech charakterystycznych dla obrazu (które określone są przez krawędzie),
- generacja przestrzeni skal, które pozwalają opisać i analizować obraz na różnych poziomach szczegółowości [67].

Model przedstawiony w rozdziale 2.1 jest szczególnym przypadkiem zagadnienia (2 – 33). Jest to model liniowy ze względu na parametry występujące przy operatorze różniczkowych  $F$ . Jego zastosowanie do przetwarzania obrazów zostanie krótko zaprezentowane ze względu na dobrze zbadane własności teoretyczne oraz możliwości rozszerzenia modelu do przypadku nieliniowego. Na podstawie modelu do przetwarzania obrazów zostanie następnie (rozdział 5) zaproponowana metoda przetwarzania danych reprezentujący ruch ludzki.



## Zastosowanie liniowego równania przewodzenia ciepła

Równanie różniczkowe cząstkowe (2 – 15) dla funkcji dwóch zmiennych ( $x \in R^2$ ) może zostać zapisane w następującej formie (przy założeniu, że jest to równanie izotropowe, współczynnik  $D$  jest stały, bez utraty ogólności można przyjąć współczynniki  $\frac{C_w \delta}{D} = 1$ ):

$$\operatorname{div}(\nabla u(x, t)) = \frac{\partial u(x, t)}{\partial t} \quad (2 - 34)$$

W dalszej części pracy do oznaczenia funkcji niewiadomej w równaniu różniczkowym używane będzie oznaczenie  $u$  (dla odróżnienia od pola temperatury  $\hat{T}$ ). Zagadnienie początkowe przyjmuje postać:

$$\begin{cases} \frac{\partial u(x, t)}{\partial t} = \Delta u(x, t) \\ u(x, 0) = u_0(x) \end{cases} \quad (2 - 35)$$

Rozwiązaniem tego zagadnienia początkowego jest następujące wyrażenie:

$$u(x, t) = \begin{cases} u_0(x) & \text{dla } t = 0 \\ (K_{\sqrt{2t}} * u_0)(x) & \text{dla } t > 0 \end{cases} \quad (2 - 36)$$

gdzie  $*$  oznacza operację splotu, a  $K$  jest dwuwymiarowym jądrem rozkładu Gaussa o określonym odchyleniu standardowym  $\sigma = \sqrt{2t}$  i średniej  $\mu = 0$  wyrażone jako:

$$K_\sigma(x) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{\|x\|^2}{2\sigma^2}\right) \quad (2 - 37)$$

Otrzymuje się rozwiązanie, które jest zależne tylko od warunku początkowego (funkcja  $u_0$ ) oraz parametru  $t$ . Parametr czasowy pozwala na wyznaczenie obrazu w odpowiedniej skali z utworzonej przestrzeni skal.

W dziedzinie przetwarzania sygnałów istnieje wiele metod, które oparte są na wyznaczeniu splotu sygnału z inną funkcją reprezentowaną przez maskę o skończonym rozmiarze. Przyjmuje się, że obraz opisany przekształceniem (2 – 1) jest funkcją odwzorowującą jednoznacznie każdy element (piksel) w przestrzeń liczb rzeczywistych (nasycenie). Dla przypadku obrazów monochromatycznych jest to odwzorowanie przestrzeni dwuwymiarowej w jednowymiarową (przestrzeń intensywności). Przykładem filtracji opartej na wyznaczeniu splotu funkcji jest rozmywanie Gaussa (ang. *Gaussian blur*), które jest opisane przez wyrażenie:

$$(K_\sigma * u_0)(x) = \int_{R^2} K_\sigma(x - y) u_0(y) dy \quad (2 - 38)$$

przy określonym parametrze  $\sigma$  decydującym o stopniu wygładzania. Analogicznie do poprzednio stosowanych oznaczeń funkcja  $u_0$  jest początkowym obrazem, który jest poddawany filtracji.

W zależności od użycia rodzaju jądra splotu  $K$  można otrzymać filtry o różnych własnościach (niekoniecznie musi to być jądro gaussowskie). Przy użyciu jądra przedstawionego w wyrażeniu (2 – 38)

otrzymuje się dolnoprzepustowy filtr, który wygładza obraz (a co za tym idzie rozmywa również krawędzie). Implementacja algorytmu oparta na równaniu (2 – 38) jest więc w praktyce wyznaczeniem rozwiązania zagadnienia początkowego (2 – 35) przy założeniu, że funkcja  $u_0(x)$  reprezentująca obraz jest warunkiem początkowym. Rozwiązanie zagadnienia (2 – 35) jest odpowiednikiem wyznaczenia spłotu funkcji (2 – 38) dla chwili czasowej  $t = \frac{\sigma^2}{2}$ .

W praktyce wygładzanie gaussowskie wyznaczane jest jako dyskretny spłot obrazu oraz zdefiniowanej maski realizującej jądro Gaussa. Przyjmując maskę kwadratową o wymiarach  $(2m + 1) \times (2m + 1)$  o elementach  $w(i, j)$  operację dyskretnego spłotu dla pojedynczego piksela wybranego z jednorodnej siatki można zapisać jako:

$$f^{(t+1)}(x, y) = \frac{1}{S} \sum_{i=-m}^m \sum_{j=-m}^m (f^{(t)}(x, y) + f^{(t)}(x + i, y + j)w(i + m, j + m)), \quad (2 - 39)$$

gdzie  $S = \sum_{i=0}^{2m} \sum_{j=0}^{2m} w(i, j)$ . (2 – 40)

Spłot dla pojedynczego elementu jest więc realizacją ważonej interpolacji na dwuwymiarowym obszarze. Obliczenie spłotu dla całego obrazu wymaga wykonania przeglądu wszystkich pikseli według dowolnej krzywej wypełniającej. Rodzaj krzywej w tym przypadku nie ma wpływu na działanie tego algorytmu, gdyż w każdej iteracji brana jest pod uwagę wartość funkcji w poprzednim kroku. Implementacja powinna uwzględniać zapisywanie nowych wartości do nowej tablicy reprezentującej obraz. Taka realizacja filtracji dolnoprzepustowej będzie istotna z punktu widzenia metody przetwarzania danych ruchu przedstawionej w rozdziale 4.

Dyskretny spłot przedstawiony wyrażeniem (2 – 39) jest ogólną metodą przetwarzania obrazu za pomocą zdefiniowanej maski o wagach  $w(i, j)$ . W zależności od przyjętych wag można otrzymać algorytm wygładzający obraz, wyostrzający krawędzie czy też służący do ich detekcji. Aby przybliżyć dwuwymiarowy rozkład Gaussa stosuje się przykładowo następującą maskę (dla  $\sigma=0.48$ ) [29]:

$$\begin{bmatrix} 0.009 & 0.0789 & 0.009 \\ 0.0789 & 0.6908 & 0.0789 \\ 0.009 & 0.0789 & 0.009 \end{bmatrix} \quad (2 - 41)$$

Konstrukcja maski oraz jej rozmiar zależą od odchylenia standardowego  $\sigma$ , o jakim rozkład ma być zrealizowany. Najczęściej przyjmuje się, że rozmiar macierzy reprezentującej maskę powinien być wystarczająco duży, aby zostały uwzględnione wagi mieszczące się w granicy  $3\sigma$ . Opierając się na centralnym twierdzeniu granicznym można pokazać, że aproksymacja spłotu procedurą (2 – 36) jest zbieżna do rozwiązania (2 – 37). W praktyce konstruuje się maskę tak, aby jej zakres obejmował dwuwymiarowy obszar  $[6\sigma + 1, 6\sigma + 1]$ .

Różne metody filtracji wykorzystują wyznaczenie nowej wartości elementu centralnego na podstawie określonej maski. Przykładowo ważona interpolacja w równaniu (2 – 39) może przyjmować jednakowe wagi

dla wszystkich uwzględnianych elementów. Działanie takie jest filtrem uśredniającym wartości naokoło wybranego piksela. Efekt działania tej procedury jest wizualnie podobny do wygładzania gaussowskiego. Jednak w tym drugim przypadku krawędzie są rozmywane w mniejszym stopniu. Poza tym jedynie wygładzanie gaussowskie ma swój odpowiednik jako rozwiązanie równania różniczkowego cząstkowego. Należy zauważyć, że jawny schemat różnicowy zastosowany do zagadnienia początkowego (2 – 36) miałby w granicy realizację zbliżoną do wyznaczania ważonej średniej przy użyciu określonej siatki reprezentującej jądro gaussowskie.



**Rysunek 2-1: Obraz testowy nr 1 (kwiat).**

Na rysunku 2 – 2 przedstawiono kolejne iteracje (50, 100, 150) filtru realizującego splot obrazu testowego (rysunek 2 – 1) z jądrem Gaussa dla odchylenia standardowego  $\sigma=0.48$  (dyskretna maska o rozmiarze  $3 \times 3$ ). Dla porównania pokazano wynik filtracji typu uśredniającego o identycznym rozmiarze okna  $3 \times 3$  i jednostkowych wagach dla każdego elementu (rysunek 2 – 3).



**Rysunek 2-2: Wynik działania filtracji opartej na liniowym równaniu dyfuzji,  $\sigma=0.48$  (odpowiednio 50, 100, 150 iteracji).**



**Rysunek 2-3: Wynik działania filtracji uśredniającej, rozmiar okna  $3 \times 3$  (odpowiednio 50, 100, 150 iteracji).**

Na przykładowych rysunkach widoczne jest, że filtracja uśredniająca przy tym samym rozmiarze okna znacznie szybciej rozmywa obraz. Jednocześnie również krawędzie obiektów na obrazie szybciej

zostają zatarte. Pożądaną cechą jest, aby filtr usuwał niepożądane cechy na obrazie (szum), a pozostawiał cechy charakteryzujące zawartość znaczeniową. Najczęściej przyjmuje się, że to krawędzie są rozróżniającym elementem pozwalającym na późniejszą identyfikację obiektów. W rozdziale 2.5 zostanie pokazana metoda filtracji pozwalająca na minimalizację rozmycia krawędzi.

### Własności częstotliwościowe filtracji liniowej

Jak to zostało przedstawione, filtr oparty na rozwiązaniu zagadnienia początkowego przewodnictwa ciepła ma właściwości identyczne z wygładzaniem gaussowskim. Kolejne iteracje (dla kolejnych chwil czasowych) powodują, że obraz zawiera coraz mniejszą liczbę szczegółów (szumu, krawędzi). Oznacza to, że energia sygnału monotonicznie maleje. Można to również sprawdzić analitycznie wyznaczając transformatę Fouriera dla wyrażenia (2 – 37). Transformata Fouriera zdefiniowana jest jako:

$$F[f(x)](\omega) = \int_{\mathbb{R}^2} f(x) \exp(-i(\omega \cdot x)) dx, \quad (2 - 42)$$

gdzie  $\omega \cdot x$  jest iloczynem skalarnym wektorów.

Jednocześnie wiadomo, że transformata spłotu funkcji jest iloczynem transformat:

$$F[f * g] = F[f] \cdot F[g] \quad (2 - 43)$$

Biorąc pod uwagę rozwiązanie zagadnienia początkowego wyrażone przez (2 – 32) otrzymuje się:

$$F[(K_\sigma * u_0)(x)](\omega) = F[K_\sigma](\omega) \cdot F[u_0](\omega) \quad (2 - 44)$$

$$F[K_\sigma](\omega) = \exp\left(-\frac{\|\omega\|^2}{2/\sigma^2}\right) \quad (2 - 45)$$

Charakterystyka częstotliwościowa filtracji opartej na równaniu różniczkowym przewodzenia ciepła jest więc również funkcją Gaussa. Realizowane są w ten sposób założenia o uproszczeniu obrazu dla kolejnych chwil czasowych  $t$  (co odpowiada zwiększającemu się parametrowi  $\sigma$  w przypadku filtracji opartej na wygładzaniu Gaussa przez spłot). Gładka charakterystyka częstotliwościowa Gaussa zapewnia, że otrzymany filtr jest dolnoprzepustowy. Charakterystyka ta ma istotne własności w przypadku używania tego rodzaju filtracji do obrazów, gdzie będzie przeprowadzana detekcja krawędzi. Dodatkowo filtracja oparta na jądrze Gaussa ma lepsze właściwości częstotliwościowe niż uśrednianie realizowane metodą (2 – 38) przy użyciu stałych wag dla całej maski. W przypadku uśredniania charakterystyka częstotliwościowa może mieć dodatkowe podbicia w zakresie średnich częstotliwości.

## 2.5 Przegląd nieliniowych metod filtracji

Przedstawiona w poprzednim rozdziale metoda filtracji oparta na rozwiązaniu zagadnienia początkowego opisującego równanie przewodnictwa ciepła jest dobrze opisana w teorii, ma dobre właściwości w odniesieniu do tworzonej przestrzeni skal oraz nie sprawia trudności w zastosowaniu praktycznym. Jednak przy usuwaniu zakłóceń z przetwarzanego obrazu cechy charakterystyczne (takie jak krawędzie) również są rozmywane. Dodatkowo położenie krawędzi również ulega zmianie, co jest znaczną wadą przy śledzeniu

krawędzi w przestrzeni skal (od skali największej do najmniejszej). Aby temu zapobiec autorzy pracy [79] zaproponowali modyfikację równania przewodnictwa i wprowadzenia czynnika nieliniowego.

### 2.5.1 Model Perona-Malik

Do równania przekazywania ciepła (2 – 15) autorzy pracy [79] proponują wprowadzić zamiast stałego współczynnika dyfuzji  $D$  czynnik  $c = c(x, y, u, t)$  zależny od umiejscowienia na obrazie oraz wartości funkcji  $u$  w danym miejscu. Pozostałe składowe zagadnienia początkowego (2 – 33) pozostają bez zmian:

$$\begin{cases} \frac{\partial u(x, t)}{\partial t} = \operatorname{div}(c(x, y, u, t)\nabla u) & \text{na } (0, T) \times \Omega \\ \frac{\partial u(x, t)}{\partial \vec{n}_\perp} = 0 & \text{na } (0, T) \times \partial\Omega \\ u(x, 0) = u_0(x) & \text{na } \Omega, \text{ dla } x \in R^2 \end{cases}, \quad (2 - 46)$$

gdzie  $\Omega$  jest rozpatrywanym obszarem, a  $\vec{n}_\perp$  wektorem normalnym do powierzchni ograniczającej obszar  $\partial\Omega$ .

Współczynnik dyfuzji nie jest już stały, więc musi być brany pod uwagę przy operacji różniczkowania, co prowadzi do następującej postaci:

$$\frac{\partial u}{\partial t} = \operatorname{div}(c(x, y, u, t)\nabla u) = c(x, y, t, u)\nabla^2 u + \nabla c \cdot \nabla u, \quad (2 - 47)$$

gdzie  $\nabla^2 u$  oznacza laplasjan (2 – 12) funkcji  $u$ , natomiast  $\nabla c \cdot \nabla u$  jest iloczynem skalarnym.

Celem wprowadzenia zmiennego współczynnika dyfuzji jest takie sterowanie procesem, aby był on przeprowadzany intensywnie w obszarze jednolitym, natomiast zahamowany w pobliżu krawędzi. W praktyce może to zostać zrealizowane przez ustalenie współczynnika  $c = 1$ , gdy przetwarzany jest obszar jednolity oraz  $c = 0$ , gdy przetwarzanie dotyczy brzegów. W ten sposób zostaje usunięta podstawowa wada liniowego, izotropowego równania różniczkowego przewodnictwa ciepła, którego użycie wiązało się z rozmyciem krawędzi. Podejście takie wymaga jednak znajomości położenia krawędzi, co jest celem przetwarzania, a nie danymi wejściowymi. Konieczne jest więc użycie pewnego oszacowania występowania krawędzi. W pracy [79] proponuje się, aby estymator krawędzi był zdefiniowany jako następująca funkcja:

- $E(x, y, t) = 0$  - wewnątrz każdego obszaru jednolitego
- $E(x, y, t) = Ke(x, y, t)$  - w punkcie należącym do krawędzi, gdzie  $e$  jest jednostkowym wektorem normalnym do krawędzi, a  $K$  jest zmienną reprezentującą kontrast w punkcie rozumiany jako różnica jasności pomiędzy krawędzią, a otoczeniem

Następnie jako współczynnik  $c$  przyjmuje się funkcję  $g$  zależną od funkcji estymatora krawędzi taką, że w obszarze jednolitym przyjmuje wartość  $g(0) = 1$  oraz jest monotonicznie malejąca:

$$c(x, y, u, t) = g(\|E\|) \quad (2 - 48)$$

W oryginalnym modelu *Perona-Malik* przyjęto, że funkcją estymatora krawędzi jest gradient jasności obrazu w punkcie zdefiniowany jako:

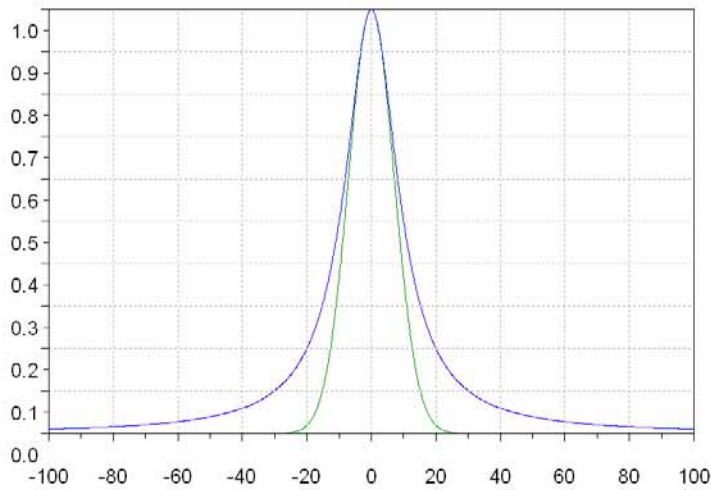
$$E(x, y, t) = \nabla u(x, y, t) \quad (2-49)$$

$$c(x, y, u, t) = g(\|\nabla u(x, y, t)\|) \quad (2-50)$$

Razem z nieliniowym modelem w cytowanej pracy zostały zaproponowane dwie postacie funkcji  $g$  pełniącej rolę zmiennego współczynnika dyfuzji (wykresy funkcji pokazane są na rysunku 2-4):

$$g_1(s) = \exp\left(-\left(\frac{s}{K}\right)^2\right) \quad (2-51)$$

$$g_2(s) = \frac{1}{1 + \left(\frac{s}{K}\right)^2} \quad (2-52)$$



Rysunek 2-4: Funkcje  $g_1$  (linia niebieska),  $g_2$  (linia zielona) dla parametru  $K=10$ .

Parametr  $K > 0$  decyduje o poziomie kontrastu, jaki jest potrzebny do zakwalifikowania wybranego punktu jako krawędź lub jako obszaru jednolitego. Rola tego parametru jest dobrze widoczna po przekształceniu funkcji  $g$  do funkcji strumienia  $\Phi(s) = sg(s)$ :

$$\Phi_1(s) = s \exp\left(-\left(\frac{s}{K}\right)^2\right) \quad (2-53)$$

$$\Phi_2(s) = \frac{s}{1 + \left(\frac{s}{K}\right)^2} \quad (2-54)$$

Po wyznaczeniu pochodnych funkcji strumienia widoczne jest, że dla  $|s| \leq K$  pochodna jest nieujemna  $\Phi'(s) \geq 0$ , co oznacza, że w tym rejonie równanie ma charakter paraboliczny o dyfuzji postępującej. Natomiast dla  $|s| > K$  otrzymuje się  $\Phi'(s) < 0$ , co prowadzi do wystąpienia dyfuzji

wstecznej. Może to prowadzić do procesu, który nie jest stabilny i powoduje wzmacnianie sygnału szumu. Jednak jak podają autorzy pracy [79] obszary, na których taka zależność występuje zostają szybko zredukowane przez proces wygładzania.

Równanie (2 – 47) może być przekształcone przy użyciu funkcji strumienia do postaci (przypadek jednowymiarowy):

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \Phi(u_x) = \Phi'(u_x) u_{xx} \quad (2 - 55)$$

a dla przypadku dwumiarowego:

$$\frac{\partial u}{\partial t} = \Phi'(\nabla u) u_{\eta\eta} + g(\nabla u) u_{\xi\xi} \quad (2 - 56)$$

W równaniu (2 – 56) zostały zamienione współrzędne na  $\xi, \eta$ , które oznaczają odpowiednio kierunek prostopadły oraz równoległy do  $\nabla u$ . Otrzymuje się więc dyfuzję postępującą wzdłuż kierunku prostopadłego do gradientu funkcji  $u$  oraz możliwość wystąpienia dyfuzji wstecznej dla kierunku zgodnego z gradientem funkcji.

Użycie funkcji  $g_1$  powoduje lepsze wyznaczenie krawędzi o wysokim kontraście, natomiast funkcja  $g_2$  lepiej wygładza rozległe jednolite obszary. Sposób określenia parametru  $K$  nie jest jednoznaczny. Najczęściej trzeba go wyznaczyć eksperymentalnie. Możliwe jest również uzależnienie go od histogramu szumu wyznaczonego z obrazu [79]. W eksperymentach przeprowadzanych w niniejszej pracy stosuje się między innymi następujący sposób adaptacyjnego sposobu określenia współczynnika  $K$ :

$$K(t) = \mathcal{G} \frac{1}{W(I)H(I)} \sum_{x=0}^{W(I)-1} \sum_{y=0}^{H(I)-1} |\nabla u(x, y, t)|, \quad (2 - 57)$$

gdzie  $W(I), H(I)$  są odpowiednio szerokością i wysokością obrazu, natomiast  $\mathcal{G}$  jest współczynnikiem skalującym, którego wartość na potrzeby eksperymentów przyjęto równą 0,9. Adaptacyjne wyznaczenie współczynnika  $K$  pozwala zmniejszyć wpływ funkcji stopującej  $g$  na początku procesu, kiedy obraz zawiera znaczną zawartość szumu. Duża wartość współczynnika  $K$  wynika z dużej wartości średniej normy gradientu obrazu, co powoduje, że współczynnik dyfuzji jest zbliżony do 1. Proces staje się podobny do liniowej dyfuzji, w której informacja o lokalnej wartości gradientu nie jest brana pod uwagę. Po początkowych iteracjach zaszumienie obrazu się zmniejsza, co powoduje spadek wartości współczynnika  $K$ . W ten sposób proces staje się bardziej czuły na lokalne zmiany przybliżone gradientem funkcji i realizowane jest zmniejszenie dyfuzji w obszarach, w których prawdopodobnie (z punktu widzenia oszacowania) występują krawędzie. W kolejnych rozdziałach zostanie pokazana w praktyce różnica w działaniu filtracji wykorzystujących stały oraz adaptacyjny współczynnik  $K$ .

W literaturze były badane różnego rodzaju funkcje  $g$  ze względu na możliwość użycia ich w roli funkcji stopującej we współczynniku dyfuzji. Między innymi w pracy [8] została przeprowadzona analiza filtracji opartej na nieliniowej dyfuzji w kontekście statystyk odpornych. Zaproponowane było użycie następujących funkcji  $g$  wynikających z odpowiednich statystyk:

$$g_3(x, \sigma) = \begin{cases} \frac{1}{2} \left( 1 - \left( \frac{x}{\sigma} \right)^2 \right)^2 & \text{dla } |x| \leq \sigma \\ 0 & \text{w przeciwnym wypadku} \end{cases} \quad (2-58)$$

$$g_4(x, \sigma) = \begin{cases} \frac{1}{\sigma} & \text{dla } |x| \leq \sigma \\ \text{sign}(x)/x & \text{w przeciwnym wypadku} \end{cases} \quad (2-59)$$

Funkcje te są wyznaczone na podstawie norm statystyk odpornych odpowiednio *Tukey's biweight* oraz *Huber's minmax*, jako  $g(x)x = \rho'(x)$ . Parametr  $\sigma$  określa skalę odporności, jest więc odpowiednikiem kryterium, czy dany element należy określić jako błąd gruby. Parametr ten może być wyznaczony także automatycznie, czego propozycja została podana w [8].

### 2.5.2 Dyskretyzacja nieliniowego modelu *Perona-Malik*

Razem z nieliniowym modelem, w pracy [79] została zaproponowana następująca dyskretyzacja nieliniowego równania dyfuzji oparta na jawnym schemacie różnicowym przedstawionym w rozdziale 2.3:

$$u_s^{t+1} = u_s^t + \frac{\lambda}{|\kappa_s|} \sum_{p \in \kappa_s} g(\nabla I_{s,p}) \nabla I_{s,p}, \quad (2-60)$$

gdzie  $s, p$  jest położeniem piksela na dwuwymiarowej siatce  $s = (x, y)$ ,  $\kappa_s$  oznacza przestrzenne sąsiedztwo piksela  $s$ , natomiast  $|\kappa_s|$  jest liczbą sąsiadów piksela  $s$ . Jeżeli przyjmuje się warunek początkowy typu Neumanna (adiabatyczny – brak wymiany ciepła z otoczeniem), to dla wszystkich pikseli oprócz pikseli brzegowych występuje relacja czterosąsiedztwa zdefiniowana w rozdziale 2.1. W takim wypadku dyskretyzację można zapisać używając notacji przestrzennej:

$$u_s^{t+1} = u_s^t + \frac{\lambda}{4} [c_N \cdot \nabla_N u + c_S \cdot \nabla_S u + c_E \cdot \nabla_E u + c_W \cdot \nabla_W u]_s^t \quad (2-61)$$

Parametr  $\lambda \in (0, 1]$  decyduje o szybkości realizowania dyfuzji wyznaczonej za pomocą tego schematu. Zamiast wyznaczać rzeczywistą wartość gradientu, proponuje się określenia go na podstawie czterech sąsiedztw w ustalonych ortogonalnych kierunkach:

$$\begin{aligned} \nabla_N u_{x,y} &= u_{x-1,y} - u_{x,y} \\ \nabla_S u_{x,y} &= u_{x+1,y} - u_{x,y} \\ \nabla_E u_{x,y} &= u_{x,y+1} - u_{x,y} \\ \nabla_W u_{x,y} &= u_{x,y-1} - u_{x,y} \end{aligned} \quad (2-62)$$



Również współczynniki dyfuzji  $c$  są wyznaczane w każdej iteracji na podstawie wyliczonych różnic:

$$\begin{aligned} c_N^t &= g\left(\|\nabla_N u_s^t\|\right) \\ c_S^t &= g\left(\|\nabla_S u_s^t\|\right) \\ c_E^t &= g\left(\|\nabla_T u_s^t\|\right) \\ c_W^t &= g\left(\|\nabla_W u_s^t\|\right) \end{aligned} \tag{2-63}$$

Jest to najprostszy ze sposobów dyskretyzacji równania nieliniowego. Prosta implementacja oraz brak problemów ze stabilnością numeryczną spowodowały, że przedstawiony schemat jest bardzo często stosowany w praktyce. Jednak dyskretyzacja (2-61) nie jest dokładnym odwzorowaniem równania (2-47). Równanie oryginalne jest ciągłe oraz uwzględnia gradient funkcji  $u$  liczony w danym punkcie. Przedstawiona dyskretyzacja zamiast gradientu wyznacza dyskretną pochodną cząstkową funkcji w ortogonalnych kierunkach  $x, y$ . W praktyce okazuje się jednak, że przyjęcie nawet tak prostego schematu numerycznego może dawać dobre efekty obliczeniowe. Dodatkową zaletą tego podejścia jest łatwa implementacja obliczeń równoległych w postaci sieci VLSI [78]. Przedstawiony model pozwala również na zastosowanie w sieciach neuronowych realizujących filtrację obrazów [16], [82].

### 2.5.3 Regularyzacja modelu nieliniowego

Przedstawiony w podrozdziale 2.5.2 model oparty na nieliniowym równaniu różniczkowym cząstkowym, może powodować niestabilność procesu ze względu na występowanie czynnika reprezentującego wsteczną dyfuzję. Użycie odpowiednich metod dyskretyzacji równania może samo w sobie stabilizować realizowany proces. Aby jednak zapobiec niestabilnym przypadkom można dodatkowo wprowadzić element regularyzacyjny do samego modelu.

Najbardziej popularnym sposobem regularyzacji przestrzennej jest zmiana sposobu wyznaczania wartości współczynnika dyfuzji przy obliczaniu funkcji  $g$ . Oryginalnie współczynnik dyfuzji jest funkcją gradientu obrazu  $u$ . Jednak przykładowo w przypadku znacznych zakłóceń znajdujących się na obrazie gradient ten może błędnie wskazywać na wystąpienie krawędzi w miejscu, gdzie jej nie ma. Według przedstawionego modelu zakłócenia takie zamiast zostać usunięte, zostaną wyostrzone, co jest zdecydowanie niepożądanym efektem. W związku z tym proponuje się modyfikację równania dyfuzji w następujący sposób [15]:

$$\frac{\partial u}{\partial t} = \text{div}(g(|\nabla u_\sigma|) \nabla u) \tag{2-64}$$

$$u_\sigma = K_\sigma * u \tag{2-65}$$

Do wyznaczenia funkcji  $g$  używa się więc informacji z obrazu, który został wcześniej poddanych splotowi z jądrem gaussowskim. Jest to dwustopniowa procedura, która pomocniczo wykonuje liniową filtrację określoną standardowym równaniem dyfuzji (rozdział 2.4). Można pokazać, że takie podejście wprowadza wystarczającą (ze względu na stabilność) regularyzację do modelu.

W innym podejściu pokazanym w pracy [30] dyskretyzację zaproponowaną oryginalnie przez *Perona-Malik* proponuje się rozszerzyć przez uwzględnienie relacji ośmiosąsiedztwa dla piksela centralnego:

$$\rho((i, j), (i', j')) = \max(|i - i'|, |j - j'|) \quad (2 - 66)$$

(według oznaczeń przyjętych w rozdziale 2.1).

Dla pikseli sąsiadujących diagonalnie przyjmuje się wtedy krok dyskretyzacji o długości  $\sqrt{2}$ . Wyniki przedstawione w cytowanej pracy pokazują, że takie podejście pozytywnie wpływa zarówno na regularyzację, jak i na dokładność otrzymywanych wyników.

Wprowadzenie regulacji czasowej zaproponowane zostało w pracy [17] przez użycie systemu sprzężonych równań różniczkowych:

$$\frac{\partial u}{\partial t} - \operatorname{div}(D\nabla u) = 0 \quad (2 - 67)$$

$$\frac{dD}{dt} + \frac{1}{\alpha} D = \frac{1}{\alpha} F(\nabla u), \quad (2 - 68)$$

gdzie  $\tau$  jest parametrem realizującym opóźnienie czasowe, natomiast  $F(\nabla u)$  wyznacza rzut w kierunku ortogonalnym do  $\nabla u$ :

$$F(\nabla u) = \begin{cases} s^2 P_{\nabla u^\perp} & \text{dla } |\nabla u| > s \\ |\nabla u|^2 P_{\nabla u^\perp} + \frac{3}{2}(s^2 - |\nabla u|^2)I & \text{dla } |\nabla u| \leq s \end{cases} \quad (2 - 69)$$

$$P_{\nabla u^\perp} = \frac{1}{|\nabla u|^2} \begin{pmatrix} u_x^2 & -u_x u_y \\ -u_x u_y & u_y^2 \end{pmatrix} \quad (2 - 70)$$

Inne metody opierają się na wprowadzenia regularyzacji czasowo-przestrzennej [76] lub użyciu równania różniczkowego trzeciego rzędu [6]. W pracy [97] przedstawione jest podejście oparte na sprzężonych równaniach różniczkowych. Znaczne problemy ze stabilnością zostały przedstawione w [112], gdzie rozpatrywany jest model oparty na równaniach różniczkowych czwartego rzędu. Własności i stabilność tego modelu są następnie rozpatrywane w [23]. Zastosowanie równań różniczkowych czwartego rzędu zostało rozwinięte w pracy [68].

W każdym z tych przypadków regularyzacja nie tylko stabilizuje proces, ale również pozwala na zmianę charakteru filtracji otrzymywanej przy użyciu takiego modelu.

#### 2.5.4 Anizotropowy model nieliniowej filtracji

Modele filtracji przedstawione w poprzednich rozdziałach bazują na izotropowym równaniu dyfuzji. Oznacza to, że współczynnik przewodzenia  $c$  w równaniu różniczkowym dyfuzji jest skalarem – stałym dla równania liniowego, zmiennym i zależnym od funkcji  $u$  dla równania nieliniowego w modelu typu *Perona-*

Malik<sup>2</sup>. W modelu izotropowym strumień jest zawsze równoległy do kierunku gradientu określonego w danym punkcie. Takie podejście nie zawsze jest wystarczające w przypadku niektórych zastosowań (np. detekcja narożników, nie tylko krawędzi). W pracy [107] został zaproponowany w pełni anizotropowy model filtracji obrazów. Sformułowane zostały również ciągła, pół-dyskretna oraz dyskretna postać procesu opartym na tak określonym równaniu różniczkowym. Krótko podejście to zostanie przedstawione wraz z jego podstawowymi własnościami.

W pracach [106], [107] wprowadza się następujący ciągły model anizotropowej dyfuzji realizującej filtrację obrazów:

$$\frac{\partial u}{\partial t} = \operatorname{div}(D(\nabla u_\sigma)\nabla u) \quad \text{w } (0, \infty) \times \Omega \quad (2-71)$$

$$u(x, 0) = u_0(x) \quad \text{na } \Omega, \text{ dla } x \in R^2 \quad (2-72)$$

$$\langle D(\nabla u_\sigma)\nabla u, \vec{n}_\perp \rangle = 0 \quad \text{na } (0, \infty) \times \partial\Omega, \quad (2-73)$$

gdzie  $\vec{n}_\perp$  jest wektorem normalnym do powierzchni brzegowej  $\partial\Omega$ ,  $D \in R^{2 \times 2}$  jest tensorem zależnym od struktury obrazu. Aby proces był stabilny proponuje się uzależnić tensor  $D$  od funkcji  $u$  wstępnie przetworzonej w liniowym izotropowym procesie dyfuzji (zamiast bezpośrednio od funkcji  $u$ ):

$$u_\sigma(x, t) = (K_\sigma * \underline{u}(\cdot, t))(x) \quad (\sigma > 0), \quad (2-74)$$

gdzie  $\underline{u}$  oznacza funkcję będącą rozszerzeniem  $u$  z obszaru  $\Omega$  na przestrzeń  $R^2$  przez operację odbicia względem  $\partial\Omega$ . Jest to realizacja innego niż do tej pory rozpatrywano warunku brzegowego, kiedy przyjmowany był brak wymiany ciepła z otoczeniem.

### Dyfuzja uwydatniająca krawędzie

Aby proces filtracji uwydatniał krawędzie pożądane jest, by dyfuzja przebiegała wzdłuż krawędzi (prostopadle do wektora gradientu funkcji), a była zahamowana w kierunku prostopadłym do krawędzi. W tym celu konstruuje się odwzorowanie opisane równaniem (2-73) w którym wektory własne tensora dyfuzji spełniają następujące własności:

$$\vec{e}_1^{(D)} \parallel \nabla u_\sigma \quad (2-75)$$

$$\vec{e}_2^{(D)} \perp \nabla u_\sigma \quad (2-76)$$

Aby zwiększyć dyfuzję wzdłuż krawędzi wartości własne tensora  $D$  powinny spełniać:

$$\frac{\lambda_1^{(D)}(|\nabla u_\sigma|)}{\lambda_2^{(D)}(|\nabla u_\sigma|)} \rightarrow 0 \quad \text{dla } |\nabla u_\sigma| \rightarrow \infty \quad (2-77)$$

Przykładowo wartości własne można wybrać jako:

$$\lambda_1^{(D)} = g(|\nabla u_\sigma|^2) \quad (2-78)$$

<sup>2</sup> W literaturze występuje niejednoznaczność nazewnictwa. Autorzy pracy [79] nazywają model *Perona-Malik* filtracją anizotropową – anizotropia jest w tym przypadku uwzględniana pośrednio przez nieliniową funkcję  $g$  uwzględniającą lokalną normę gradientu. Model anizotropowy pokazany w [107] ma pod tym względem bardziej ogólny charakter.

$$\lambda_2^{(D)} = 1 \quad (2 - 79)$$

Takie podejście powoduje, że gradient  $\nabla u$  nie jest równoległy do tensora  $D$ , jeżeli tylko parametr regularyzacji  $\sigma > 0$ . Gdy parametr ten zmierza do 0, własności dyfuzji zblizają się do własności procesu *Perona-Malik* opisanego równaniem (2 – 43). Użycie wstępnie przetworzonej postaci funkcji  $u_\sigma$  pozwala nie tylko na regularyzację procesu, ale nadaje mu w ten sposób zupełnie nowy charakter.

### Dyfuzja uwydatniająca spójność

Nie w każdym przypadku najistotniejsze jest uwydatnienie krawędzi. Przykładowo dla analizy struktur na obrazie, w których informacja zapisana jest w postaci połączonych linii, ważniejsze jest, aby proces filtracji zachowywał i uwydatniał spójność elementów. Przykładem takiej analizy jest rozpoznawanie odcisków palców. W takim wypadku w pracach [104], [106] rozpatruje się tensor struktury (ang. *structure tensor*) geometrycznej obrazu  $J_\rho(\nabla u_\sigma)$ , który posłuży następnie do skonstruowania tensora dyfuzji  $D(J_\rho(\nabla u_\sigma))$ . W ogólności tensor struktury dla dwuwymiarowej funkcji  $u(x, y)$  określony jest przez następującą macierz:

$$J(u) = \begin{bmatrix} \frac{\partial^2 u}{\partial x^2} & \frac{\partial u}{\partial x} \frac{\partial u}{\partial y} \\ \frac{\partial u}{\partial x} \frac{\partial u}{\partial y} & \frac{\partial^2 u}{\partial y^2} \end{bmatrix} \quad (2 - 80)$$

Struktury, które są uwzględniane w procesie filtracji wynikają z krawędzi znajdujących się na obrazie. Stąd nieregularyzowaną wersję tensora struktury otrzymuje się jako następujący iloczyn tensorowy:

$$J_0(\nabla u_\sigma) = \nabla u_\sigma \otimes \nabla u_\sigma = \nabla u_\sigma \nabla u_\sigma^T \quad (2 - 81)$$

który spełnia warunki (2 – 71), (2 – 72). Wielkością, która uwzględnia zmianę wartości funkcji w interesujących kierunkach, a nie tylko w najbliższym położeniu jest następujący tensor:

$$J_\rho(\nabla u_\sigma) = K_\rho * (\nabla u_\sigma \otimes \nabla u_\sigma) \quad \text{dla } \rho \geq 0, \quad (2 - 82)$$

gdzie  $K_\rho$  jest jądrem gaussowskim o odchyleniu standardowym  $\rho$ . Dzięki przekształceniu (2 – 82) otrzymuje się regularyzowaną wersję tensora struktury, która bierze pod uwagę wartości funkcji z większego obszaru w zależności od szerokości przyjętego jądra Gaussowskiego.

Można pokazać [107], że symetryczna, dodatnio określona macierz  $J_\rho$  o elementach  $\begin{bmatrix} J_{11} & J_{12} \\ J_{21} & J_{22} \end{bmatrix}$

posiada ortonormalne wektory własne  $\bar{e}_1^{(J)}$ ,  $\bar{e}_2^{(J)}$  gdzie

$$\bar{e}_1^{(J)} \parallel \begin{pmatrix} 2J_{12} \\ J_{22} - J_{11} + \sqrt{(J_{11} - J_{22})^2 + 4J_{12}^2} \end{pmatrix} \quad (2 - 83)$$

oraz odpowiadające im wartości własne

$$\lambda_{1,2}^{(J)} = \frac{1}{2} \left( J_{11} + J_{22} \pm \sqrt{(J_{11} - J_{22})^2 + 4J_{12}^2} \right) \quad (2-84)$$

Parametr  $\rho$  służy do uśrednienia wartości funkcji w otoczeniu o wielkości  $O(\rho)$ , co pozwala na określenie kontrastu wzdłuż kierunku równoległego do wektorów własnych. Wielkość tego parametru zależy więc od wielkości struktur występujących na obrazie, które mają być uwzględniane. Parametr  $\sigma$  jest związany ze wstępną izotropową filtracją i powoduje, że anizotropowy proces jest niewrażliwy na szum występujący na obrazie. Jego wielkość powinna być uzależniona od wielkości szumu na obrazie.

Wartości własne mają następującą interpretację dla poszczególnych przypadków opisujących lokalną strukturę obrazu:

Obszar jednolity	$\lambda_1^{(J)} = \lambda_2^{(J)} = 0$
Krawędzie	$\lambda_1^{(J)} \gg \lambda_2^{(J)} = 0$
Narożniki	$\lambda_1^{(J)} > \lambda_2^{(J)} \gg 0$
Struktury anizotropowe	$(\lambda_1^{(J)} - \lambda_2^{(J)})^2$ rośnie

**Tabela 1:** Wartości własne tensora  $D$  w zależności od struktury na obrazie (na podstawie [107])

Wielkość  $(\lambda_1^{(J)} - \lambda_2^{(J)})^2$  jest miarą lokalnej spójności obrazu. Wektor własny  $\vec{e}_2^{(J)}$  określa kierunek spójności, wzdłuż którego powinna być przeprowadzona dyfuzja uwydatniająca spójność. Proces, który będzie uwzględniał to założenie może być zrealizowany przez użycie tensora  $D$ , który posiada takie same wektory własne, jak  $J_\rho$  oraz odpowiadające im wartości własne [106]:

$$\lambda_1^{(D)} = \alpha \quad (2-85)$$

$$\lambda_2^{(D)} = \begin{cases} \alpha & \text{dla } \lambda_1^{(J)} = \lambda_2^{(J)} \\ \alpha + (1 - \alpha) \exp\left(-\frac{C}{(\mu_1 - \mu_2)^{2m}}\right) & \text{dla } \lambda_1^{(J)} \neq \lambda_2^{(J)} \end{cases} \quad (2-86)$$

gdzie parametry  $C > 0, m \in \mathbb{N}, \alpha \in (0,1)$  pozwalają sterować szybkością procesu dyfuzji oraz zapewniają, że tensor dyfuzji jest dodatnio określony.

## 2.6 Przykłady zastosowania filtrów opartych na równaniach różniczkowych

Aby porównać sposoby działania liniowej i nieliniowej filtracji zostaną przedstawione wyniki skuteczności filtracji zarówno na obrazach naturalnych, jak i sztucznych. Badania dotyczą obrazów reprezentowanych w skali szarości, ze względu na analizę podstawowych własności procesów dyfuzyjnych wykorzystywanych przy konstrukcji filtrów w kolejnych rozdziałach. Liniowa dyfuzja obrazów kolorowych jest prostym uogólnieniem na przestrzeń wielowymiarową (przykład zastosowania w rozdziale 2.4). Zastosowanie nieliniowej dyfuzji (przykładowo modelu *Perona-Malik*) dla tego typu obrazów nie może być

realizowane przez niezależne procesy dla każdej składowej spektralnej. Takie podejście mogłoby powodować rozmycie krawędzi, na których opis składają się poszczególne składowe. Sformalizowane podejście modelujące krawędzie w przestrzeni obrazu kolorowego zostało zaproponowane w [87].

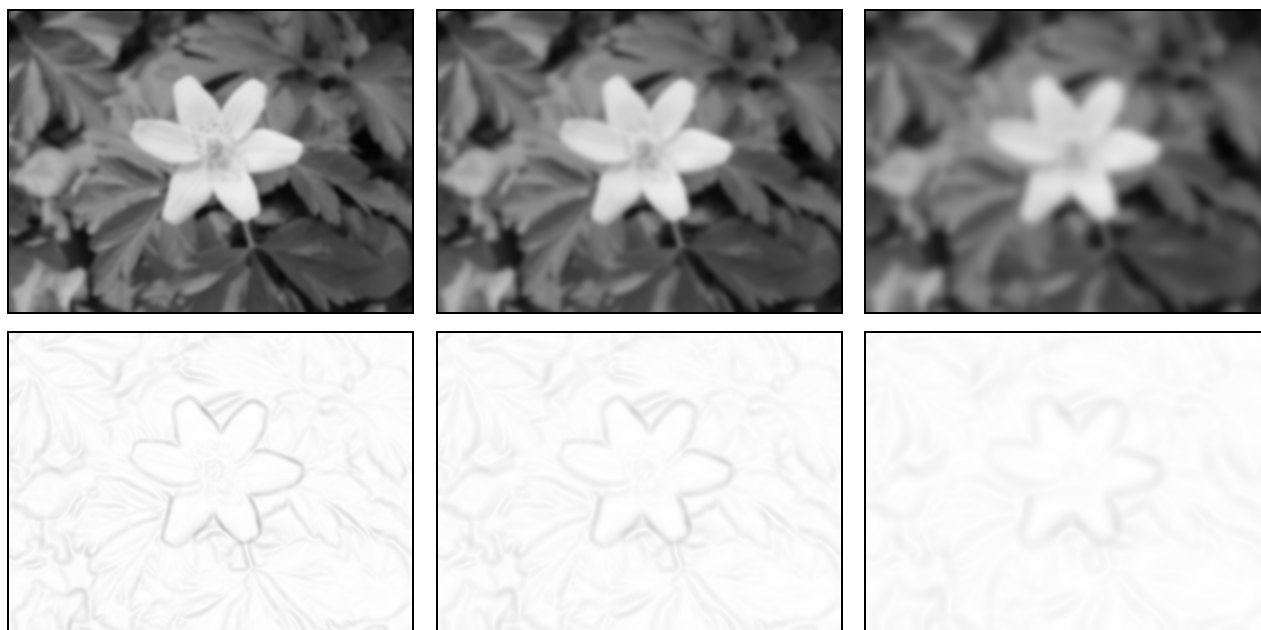
Zostaną przedstawione możliwości generacji przestrzeni skal przez wybrane algorytmy. Następnie zostanie porównana skuteczność algorytmów w przypadku obrazów zaszumionych.

### 2.6.1 Generacja przestrzeni skal na podstawie obrazu oryginalnego

Analizowane algorytmy filtracji obrazów generują przestrzeń skal, to znaczy takiego ciągu reprezentacji obrazu, dla którego kolejne elementy są uproszczoną wersją poprzednich. Na przykładzie testowego obrazu nr 1 (rysunek 2-5) pokazana zostanie generacja przestrzeni skal dla algorytmów opartych na liniowym procesie dyfuzyjny i nieliniowym procesie *Perona-Malik*.



Rysunek 2-5: Obraz testowy nr 1 (rozmiar 400x300), detekcja krawędzi, przebieg funkcji jednej linii obrazu ( $y=105$ , linia zaznaczona kolorem czerwonym).



Rysunek 2-6: Wynik działania filtru liniowego dla  $\sigma=0.8$ ; iteracje 5, 10, 30 (dolny rząd – detekcja krawędzi typu norma gradientu).

Detekcja krawędzi jest realizowana jako norma gradientu w każdym punkcie obrazu. Na rysunkach zawartych w pracy, obraz przedstawiający detekcję krawędzi jest pokazany w negatywie dla zwiększenia czytelności (krawędzie zaznaczone są kolorem czarnym, brak krawędzi reprezentuje kolor biały). Jeżeli nie zaznaczono inaczej, to układ współrzędny ma początek w lewym dolnym rogu obrazu.

Na rysunku 2-6 pokazane są wybrane iteracje filtracji liniowej przy parametrze  $\sigma=0.48$ . W drugim rzędzie przedstawiony jest wynik detekcji krawędzi.

Wynik filtracji realizującej nieliniowy model *Perona-Malik* przedstawiony jest na rysunku 2-7. Zastosowano parametr  $K=12$  oraz funkcję stopującą  $g_I$ . W tym badaniu oraz we wszystkich kolejnych, jeżeli nie zaznaczono inaczej przyjmuje się wartość parametru  $\lambda=0.25$ .



**Rysunek 2-7: Wynik działania filtru PM dla  $K=12$ ,  $g_I$ ; iteracje 5, 10, 30.**

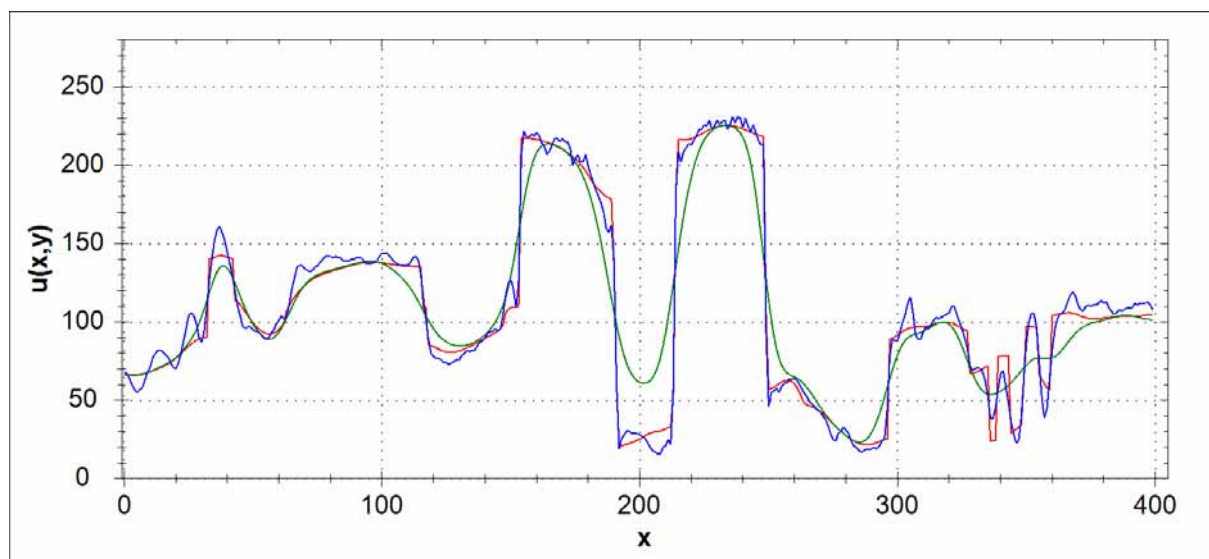
Porównując rysunki 2-6 oraz 2-7 można stwierdzić, że w obu przypadkach następuje generacja rodziny coraz bardziej uproszczonych obrazów. W przypadku filtracji realizującej liniowe równanie dyfuzji cały obraz jest w sposób jednolity upraszczany. Dotyczy to przede wszystkim szczegółów, które są źródłem wyższych częstotliwości w widmie. Jednocześnie obraz jest upraszczany przez rozmycie krawędzi. Dobrze jest to widoczne w dolnym rzędzie na rysunku 2-6, gdzie poszczególne krawędzie są coraz mniej widoczne. Proces ten prowadzi do stanu, w którym wszystkie punkty obrazu będą miały stałą wartość równą średniej jasności obrazu początkowego.

Generacja uproszczonej wersji w przypadku dyfuzji nieliniowej PM przebiega w odmienny sposób (rysunek 2-7). Filtracja powoduje, że drobne szczegóły (dobrze widoczne na liściach i płatkach kwiatu na obrazie oryginalnym) są rozmywane. Jednak bardziej znaczące brzegi nie ulegają rozmyciu, a nawet są wyostżane. Ostatnia para obrazów na rysunku 2-7 pokazuje, że proces ten szybciej przebiega w obszarach, gdzie gradient (krawędzie) są niewielkie. Dyfuzja prostopadła do krawędzi przez długi czas jest zahamowana. Rozróżnienie czy dany obszar jest jednolity czy występuje na nim krawędź zależy od zastosowanej funkcji stopującej  $g$  oraz od parametru  $K$ . W kolejnych podrozdziałach pokazany zostanie wpływ tego parametru na własności filtracji procesu PM.

Podobnie jak w przypadku filtracji liniowej, ten proces również dąży do stanu równowagi, w którym wszystkie punkty mają jednakową wartość. Wynika to z własności procesu, który nie może wprowadzać

nowych maksimum na obszarze. Stan równowagi jest znacznie później osiągnięty ze względu na zmniejszoną szybkość dyfuzji pomiędzy obszarami oddzielnymi krawędziami.

Na rysunku 2-8 przedstawiono wykres funkcji obrazu  $u(x, y)$  względem osi OX przy ustalonej wartości  $y=105$ . Na wykresie zaznaczono przebieg funkcji dla obrazu oryginalnego, obrazu przetworzonego filtrem liniowym oraz obrazu przetworzonego filtracją opartą na nieliniowym równaniu dyfuzji.



**Rysunek 2-8: Przykładowy przebieg funkcji  $u(x,y)$  dla jednej linii obrazu  $y=105$  (niebieska – obraz oryginalny; zielona – filtracja liniowa,  $it=30$ ; czerwona – filtracja PM,  $it=30$ ).**

Krótką analizę wykresu 2-8 wskazuje na zróżnicowany charakter działania obu tych filtrów. Filtracja liniowa (wykres zielony) całkowicie usuwa wszelkie wyższe częstotliwości występujące na obrazie. Otrzymana funkcja jest znacznie wygładzona, co powoduje jednocześnie zmniejszenie wyrazistości krawędzi. Ostra krawędź reprezentowana jest przez nagłą zmianę jasności na obrazie (przykładowo widoczne jest to w okolicach wartości  $x=210$ ). Na wykresie widać, że zarówno małe fluktuacje funkcji, jak i duże zmiany zostają złagodzone. Funkcja staje się gładka (także ze względu na istnienie pochodnych, co jest wykorzystywane przy regularyzacji procesów z użyciem splotu z jądrem Gaussa).

Filtracja oparta na modelu PM (wykres czerwony) powoduje inną modyfikację funkcji obrazu. Wyższe częstotliwości zostają usunięte, jednak krawędzie rozumiane jako nagłe skoki funkcji przez dłuższy czas nie są wygładzane. Na niektórych odcinkach następuje nawet wyostrenie krawędzi, co prowadzi do występowania jednolitych obszarów dobrze widocznych po 30 iteracjach na rysunku 2-7.

Własności upraszczające obraz są bezpośrednio wykorzystywane przy filtracji z niepożądanych zakłóceń. Krótko zostaną zaprezentowane wyniki otrzymane w wyniku zastosowania algorytmów do usuwania szumu o rozkładzie Gaussa.

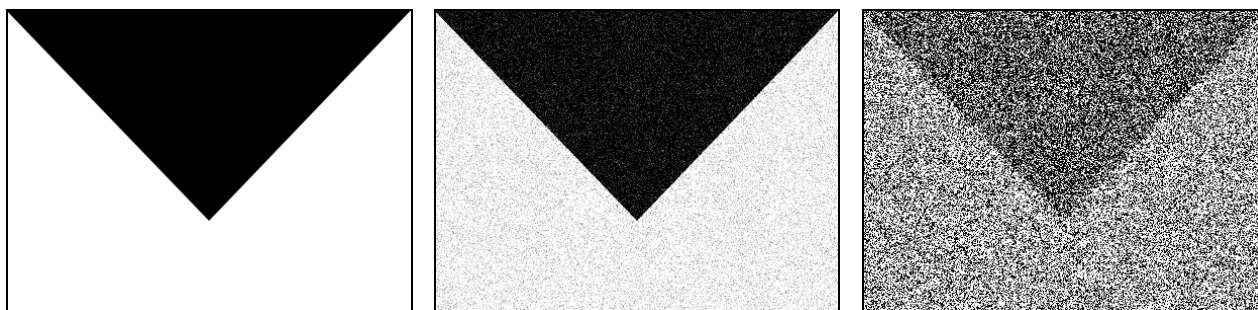


## 2.6.2 Skuteczność filtracji realizującej liniowe równanie dyfuzji – obrazy syntetyczne

Ogólna procedura testowa dla każdego rodzaju filtru realizowana jest według następującego schematu:

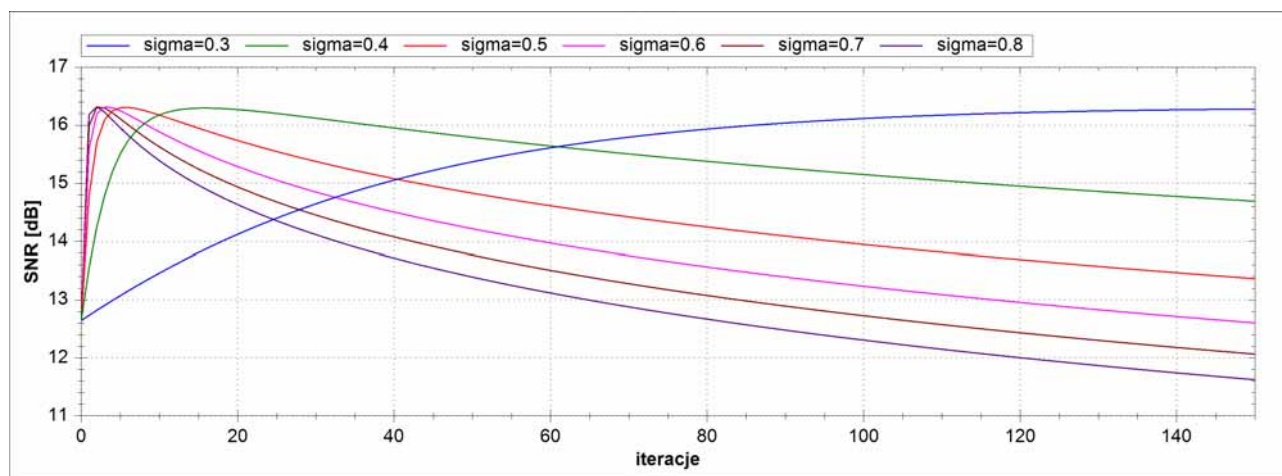
1. Przygotowanie obrazu testowego  $u_0$  na podstawie obrazu źródłowego  $u_{src}$  – wprowadzenie addytywnych zakłóceń o rozkładzie Gaussa i określonym odchyleniu standardowym  $\sigma_n$ .
2. Wykonanie  $maxIt$  iteracji dla obrazu  $u_0$  wybranego algorytmu filtracji.
3. Wyznaczenie współczynnika  $SNR$  względem  $u_{src}$  dla każdej iteracji algorytmu.

Na potrzeby testów skuteczności przygotowano syntetyczny obraz przedstawiający czarny trójkąt na białym tle (rysunek 2-9). Przekształcanie takiego rodzaju obrazu pozwala na ocenę poprawności filtracji dla dobrze początkowo zdefiniowanych krawędzi. Obraz testowy zostanie zakłócony szumem o rozkładzie Gaussa z odchyleniem standardowym odpowiednio równym  $\sigma_n = 40, 400$ .



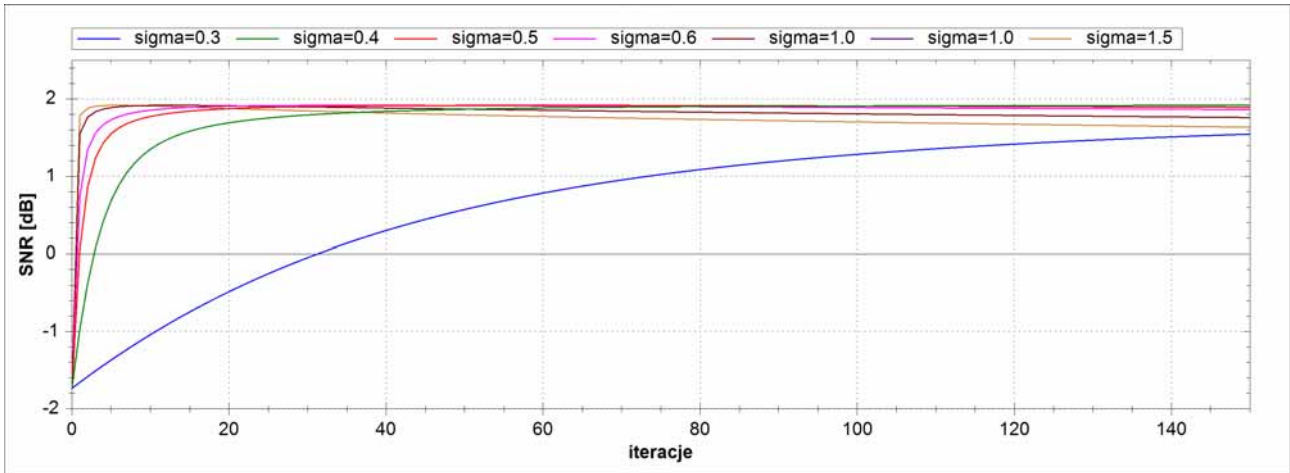
Rysunek 2-9: Syntetyczny obraz testowy: oryginalny, zaszumiony  $\sigma_n = 40$ , zaszumiony  $\sigma_n = 400$ .

Zaszumiony obraz podano działaniu filtru wykorzystującego liniowy proces dyfuzji. W tym celu sprawdzono działanie filtru dla różnych wartości parametru  $\sigma$  (określającego odchylenie standardowe rozkładu Gaussa użytego w rozwiązaniu równania). Na rysunku 2-10 przedstawiona jest wartość współczynnika  $SNR$  (ang. *signal-to-noise ratio*) po filtracji obrazu o zaszumieniu  $\sigma_n = 40$  dla poszczególnych iteracji. Analogiczny wykres dla obrazu o zaszumieniu  $\sigma_n = 400$  pokazany jest na rysunku 2-11.

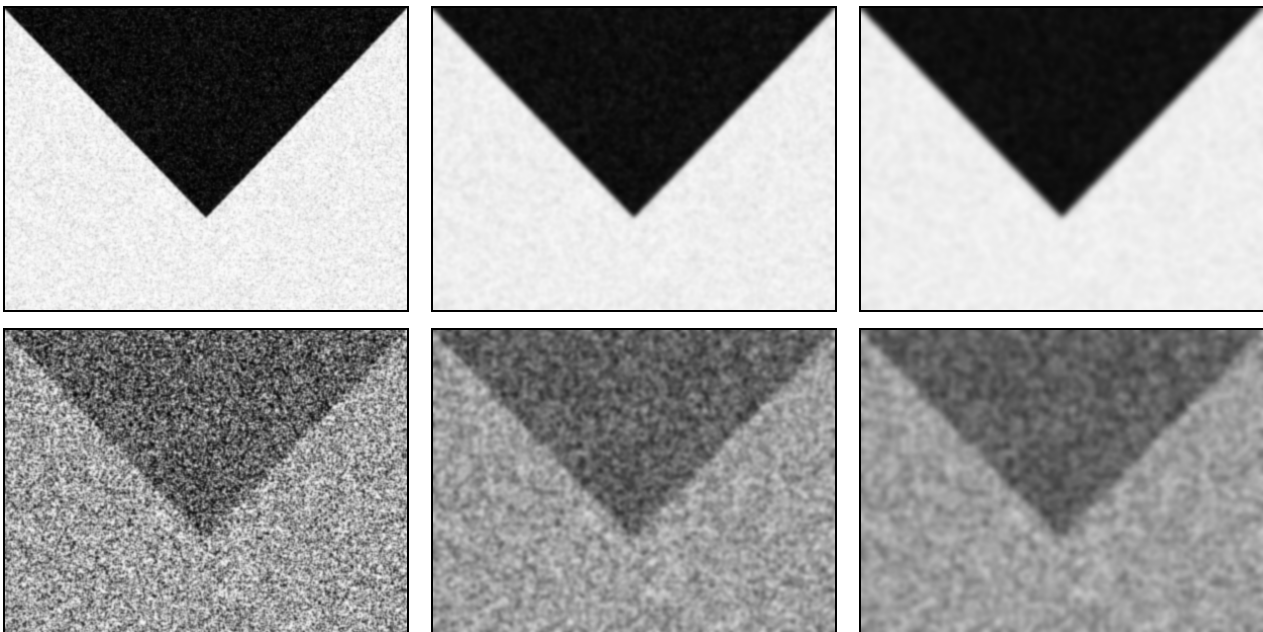


Rysunek 2-10: Miara  $SNR$  dla filtracji liniowej, zaszumienie obrazu testowego  $\sigma_n = 40$ .

Na wykresie widoczne jest, że po osiągnięciu maksymalnej wartości miary SNR, jakość otrzymanego obrazu pogarsza się dla kolejnych iteracji. Pogarszające się właściwości wyniku dla kolejnych iteracji związane są z coraz mocniejszym rozmyciem krawędzi na obrazie. Wartość maksymalna współczynnika SNR nie zmienia się znacząco dla różnych wartości parametru filtracji  $\sigma$  i wynosi w każdym przypadku około 16.2 dB. Maksimum reprezentuje iterację, dla której szum został optymalnie usunięty (przy danym parametrze  $\sigma$ ). Zwiększenie parametru  $\sigma$  powoduje, że maksimum będzie szybciej osiągnięte. Jednakże użycie zbyt dużej wartości  $\sigma$  może spowodować, że jakość otrzymanego obrazu będzie gorsza – jest to spowodowane jednoczesnym usunięciem szumu z rozmyciem istotnych krawędzi na obrazie.



Rysunek 2-11: Miara SNR dla filtracji liniowej, zaszumienie obrazu testowego  $\sigma_n = 400$ .



Rysunek 2-12: Wybrane iteracje (5, 20, 40) filtracji liniowej o parametrze  $\sigma = 0.5$  dla obrazów o zaszumieniu  $\sigma_n = 40, 400$ .

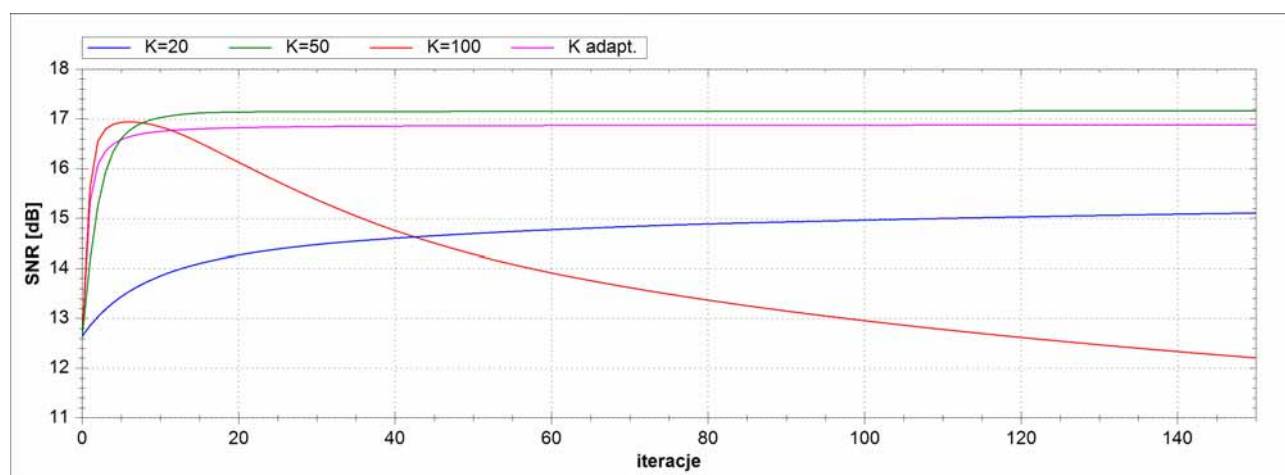
Dla obrazu, na którym szum występuje ze znacznie większym natężeniem filtracja jest trudniejsza. Wykresy na rysunku 2-11 pokazują, że po początkowym zwiększeniu współczynnika SNR, dalsza filtracja nie wprowadza poprawy. Wynika to z faktu, że początkowa zawartość szumu tak bardzo zniekształciła informację o obrazie, że trudno jest uzyskać znaczącą poprawę. Wszelkie próby usunięcia szumu są

związane z jednoczesną degradacją obrazu przez rozmycie istotnych krawędzi. Stąd niska wartość współczynnika dla wszystkich parametrów  $\sigma_n$  nawet przy znacznej liczbie iteracji.

Rysunek 2-12 przedstawia uzyskane wyniki filtracji dla wybranych iteracji (pierwszy rząd dotyczy filtracji obrazu o zaszumieniu  $\sigma_n = 40$ , drugi o zaszumieniu  $\sigma_n = 400$ ). Widoczne jest charakterystyczne dla tego rodzaju filtrów rozmycie krawędzi już po 40 iteracjach. Wizualnie otrzymany wynik w przypadku małego zaszumienia jest już bardzo dobry przy 20 iteracjach. Dla dużego zaszumienia zwiększenie liczby iteracji zmniejsza wariancję jasności punktów na obrazie, jednak jednocześnie krawędzie obiektu są tak bardzo rozmywane, że trudno jest poprawić całkowity współczynnik jakości.

### 2.6.3 Skuteczność filtracji realizującej nieliniowe równanie dyfuzji – obrazy syntetyczne

Analogiczne eksperymenty przeprowadzono dla filtru wykorzystującego proces dyfuzji według modelu PM. Testowano skuteczność filtracji dla kilku wybranych wartości parametru  $K$  oraz w zależności od używanej funkcji stopującej  $g$ . Dodatkowo przeprowadzono test skuteczności filtracji przy użyciu adaptacyjnego sposobu określania parametru  $K$ .



Rysunek 2-13: Miara SNR dla filtracji modelu PM, funkcja  $g$ , zaszumienie obrazu testowego  $\sigma_n = 40$ .

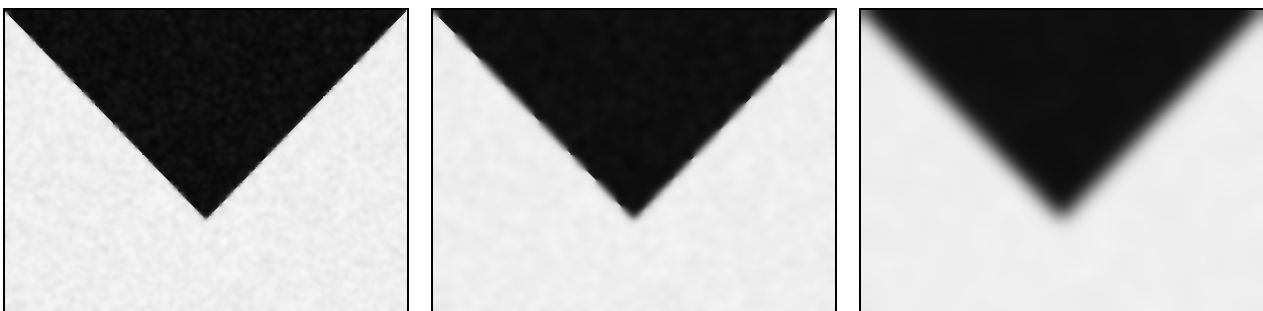
Na rysunku 2-13 widoczny jest charakterystyczny sposób działania filtracji według tego modelu. Dla zbyt małych (w porównaniu do zawartości szumu na obrazie) wartości parametru  $K$  wzrost jakości według SNR jest powolny, lecz stabilny (niebieska krzywa dla  $K=20$ ). Aby zachować czytelność wykresu skala została zawężona do 150 iteracji. Jednak nawet dla 500 iteracji filtracja przy  $K=20$  uzyskuje wartość SNR wynoszącą około 15 dB. Filtracja dla większego  $K$  (kolor zielony) pozwala na bardzo szybkie uzyskanie dobrego efektu filtracji, który utrzymuje się przez długi czas (brzezi pozostają stabilne nawet dla bardzo wielu iteracji). Użycie parametru  $K$  o zbyt dużej wartości (kolor czerwony) prowadzi do szybkiego zmniejszenia zakłóceń, co jednak po kolejnych kilku iteracjach skutkuje znacznym spadkiem jakości obrazu. Jest to spowodowane rozpoczęciem rozmywania brzegów. Należy zauważyć, że dla dużego  $K$  filtracja ma charakter zbliżony do procesu opartego na dyfuzji liniowej – funkcja stopująca nie odgrywa wtedy zbyt istotnej roli. Stąd znaczny spadek wartości SNR dla kolejnych iteracji.

Użycie adaptacyjnego sposobu wyznaczania parametru  $K$  jest bardzo praktycznym rozwiązaniem. Parametr ten zależy od zawartości szumu na obrazie, co jest mierzone przez moduł gradientu na całym obszarze. Na początku, gdy szum jest znaczny, parametr  $K$  również przyjmuje duże wartości (wykres fioletowy). Następnie jego wartość zmniejsza się, co powoduje, że dyfuzja w obszarach, gdzie występuje ciągle znaczny gradient, jest coraz bardziej hamowana. Ostatecznie filtracja taka uzyskuje nieznacznie gorsze efekty w porównaniu do odpowiednio dobranego stałego współczynnika  $K$ . Niemniej jednak podejście to pozwala na skuteczną automatyczną filtrację w przypadku zakłóceń, o których natężeniu nie ma dodatkowych informacji. Filtracja adaptacyjna również jest stabilna w czasie, to znaczy brzegi nie są rozmywane nawet dla bardzo dużej liczby iteracji.

Rysunek 2-14 przedstawia wynik filtracji obrazu o zaszumieniu  $\sigma_n = 40$  po 150 iteracjach przy użyciu funkcji stopującej  $g_I$  z parametrami  $K$  odpowiednio 20, 50 oraz wersję adaptacyjną. Dla zbyt małego współczynnika  $K$  filtracja pozostawia znaczące zakłócenia. Dla porównania na rysunku 2-15 przedstawiono wybrane iteracje procesu filtracji dla  $K=100$ . Już przy 10 iteracjach widoczny jest rozpoczynający się proces rozmywania brzegów. Skala tego efektu coraz szybciej się powiększa, co powoduje znaczny spadek miary SNR na wykresie 2-13.



Rysunek 2-14: Wyniki filtracji dla 150 iteracji z parametrem  $K=20, 50$  oraz dla wersji adaptacyjnej, funkcja  $g_I$ . Początkowe zaszumienie obrazu  $\sigma_n = 40$ .

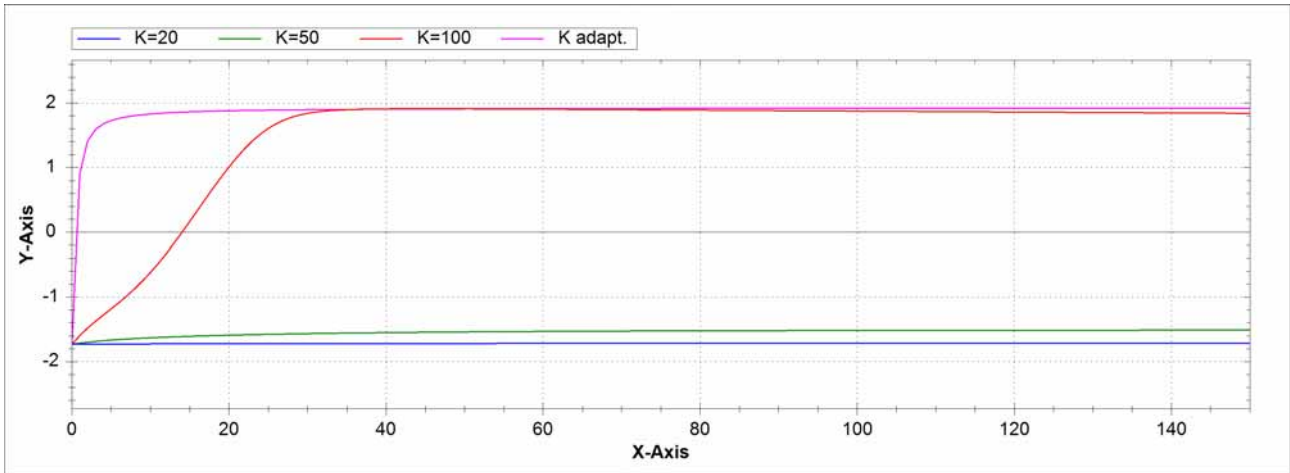


Rysunek 2-15: Wyniki filtracji dla wybranych iteracji (10, 30, 150) z parametrem  $K=100$ , funkcja  $g_I$ . Początkowe zaszumienie obrazu  $\sigma_n = 40$ .

Charakterystyka odszumiania zmienia się w przypadku filtracji obrazu o bardzo znacznym zaszumieniu (rysunek 2-16). W tym przypadku małe wartości współczynnika  $K$  nie dają wystarczających efektów – zaszumienie obrazu zostaje bardzo nieznacznie zmniejszane. Użycie współczynnika o dużej wartości powoduje szybki wzrost miary SNR, co jest jednak związane również ze znacznym rozmyciem krawędzi na obrazie. Wersja adaptacyjna algorytmu działa w tym przypadku znacznie lepiej w porównaniu do filtracji przy stałej wartości  $K$ . Jakkolwiek maksymalna miara SNR osiągnięta przez filtrację przy

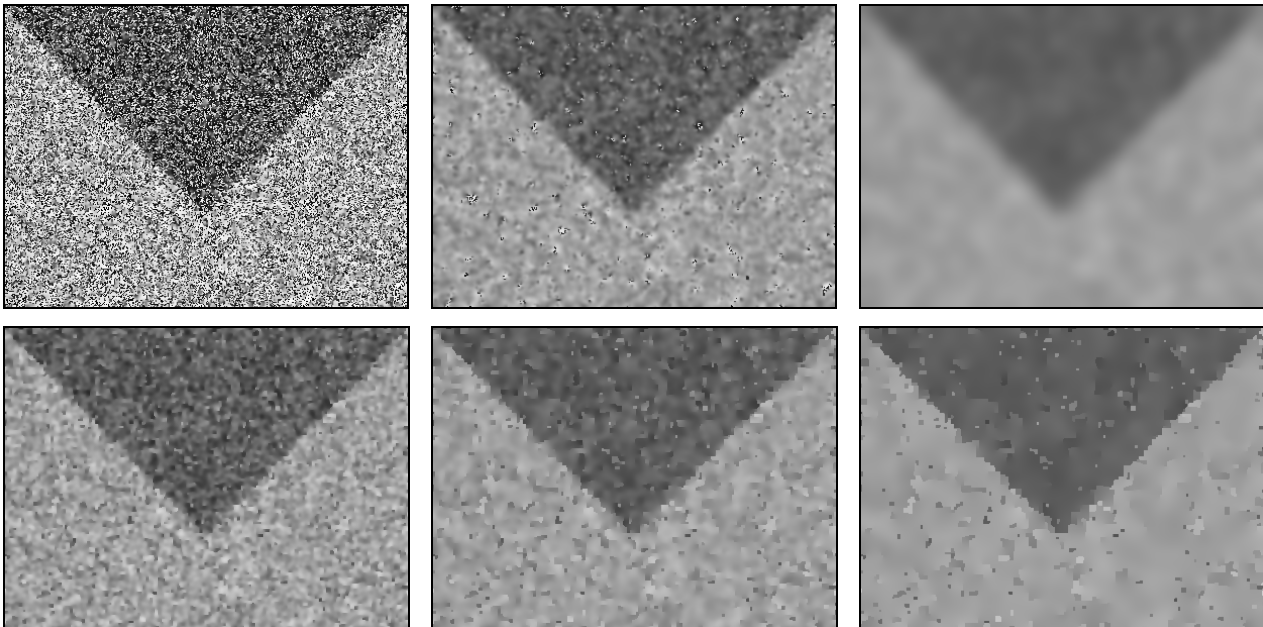
adaptacyjnym doborze  $K$  jest podobna jak w przypadku stałego dużego współczynników, to takie podejście wykazuje szereg zalet:

- początkowa filtracja jest szybsza,
- miara SNR utrzymuje się na zbliżonym poziomie przez bardzo wiele iteracji,
- brzegi obiektu na obrazie są wyostrene pomimo bardzo znacznych zakłóceń.



**Rysunek 2-16: Miara SNR dla filtracji modelu PM, funkcja  $g_I$ , zaszumienie obrazu testowego  $\sigma_n = 400$ .**

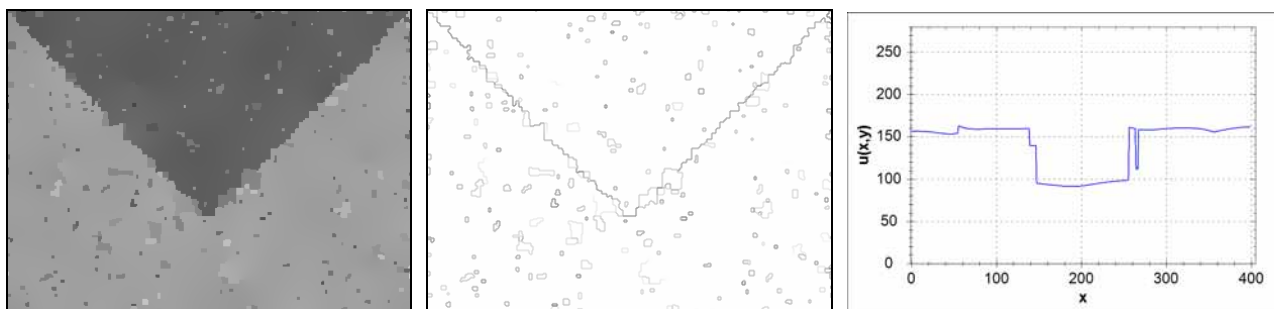
Ostatnia z wymienionych cech odróżnia charakter tej filtracji od filtracji liniowej. Brzegi w przypadku filtracji opartej na dyfuzji liniowej (rysunek 2-12) były za każdym razem rozmywane, podobnie jak to się dzieje w przypadku źle dobranej wartości współczynnika  $K$ . Dobrze zrealizowana filtracja nieliniowa dąży do wyostrenia znaczących brzegów na obrazie. Rysunek 2-17 przedstawia wybrane iteracje filtracji przy stałym  $K=100$  oraz dla  $K$  dobieranego adaptacyjnie.



**Rysunek 2-17: Wyniki filtracji dla wybranych iteracji (10, 30, 150) funkcja  $g_I$ ; górny rząd -  $K=100$ , dolny rząd -  $K$  adaptacyjne. Początkowe zaszumienie obrazu  $\sigma_n = 400$ .**

Filtracja dla  $K=100$  daje wizualnie nieznacznie lepsze efekty od filtracji liniowej przy tym samym zaszumieniu. Natomiast wersja adaptacyjna algorytmu pozwala na uzyskanie bardzo wyraźnego

rozgraniczenia pomiędzy obiektem, a otoczeniem. Miara SNR przyjmuje w tym przypadku podobną wartość, jednak brzegi obiektu są znacznie lepiej odtworzone. Jest to jeszcze lepiej widoczne dla wyniku filtracji przy 800 iteracjach przedstawionego na rysunku 2-18.



**Rysunek 2-18: Wyniki filtracji po 800 iteracjach,  $K$  adaptacyjne, funkcja  $g_1$ ; detekcja krawędzi; funkcja obrazu dla linii  $y=150$ . Początkowe zaszumienie obrazu  $\sigma_n = 400$ .**

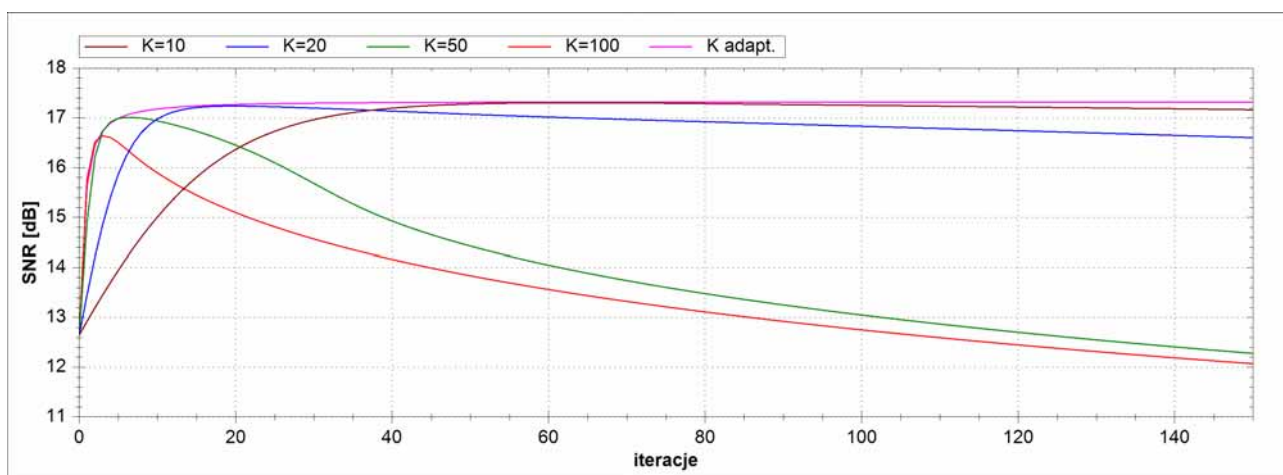
Zgodnie z teoretycznymi założeniami proces dyfuzji w nieskończoności dąży do jednolitej wartości funkcji na całym rozpatrywanym obszarze. Wartość ta jest średnią jasności pikseli obrazu początkowego, która pozostaje niezmienną po każdej iteracji filtru skonstruowanego jako realizacja procesu dyfuzyjnego. Rysunek 2-18 pokazuje, że przy użyciu nieliniowej wersji filtru możliwe jest uzyskanie dobrego rozróżnienia obiektu od tła nawet w przypadku bardzo dużej początkowej degradacji informacji. Wprowadzenie sygnału zakłócającego do obrazu spowodowało:

- Degradację krawędzi, których rekonstrukcja nie jest w pełni możliwa bez dodatkowej wiedzy. Należy zauważyć, że degradacja krawędzi w przypadku dyfuzji nieliniowej (rysunek 2-18) ma inny charakter niż w przypadku dyfuzji liniowej (rysunek 2-12). Z punktu widzenia segmentacji obrazów wynik uzyskany dla dyfuzji nieliniowej jest bardziej praktyczny.
- Zmianę średniej jasności obrazu. Wykres dla jednej linii obrazu na rysunku 2-18 pokazuje, że początkowe obszary o maksymalnej i minimalnej jasności ostatecznie zostały sprowadzone do jasności o mniejszych różnicach.
- Powstawanie nowych obszarów o jednolitej jasności – jest to specyficzny efekt wynikający z charakteru rozpatrywanej nieliniowej filtracji.

We wszystkich przeprowadzonych eksperymentach współczynnik  $K$  był wyznaczany za pomocą zależności (2 – 57). Za każdym razem przyjęto parametr skalujący  $\mathcal{G}$  na poziomie 0.9. Własności filtracji mogłyby nieznacznie ulec zmianie po przyjęciu innej wartości parametru  $\mathcal{G}$ . Z obserwacji w trakcie przeprowadzania eksperymentów można stwierdzić, że współczynnik skalujący w mniejszy sposób wpływa na jakość uzyskiwanych wyników. Stąd jego dobór jest łatwiejszy niż prawidłowy dobór współczynnika  $K$ .

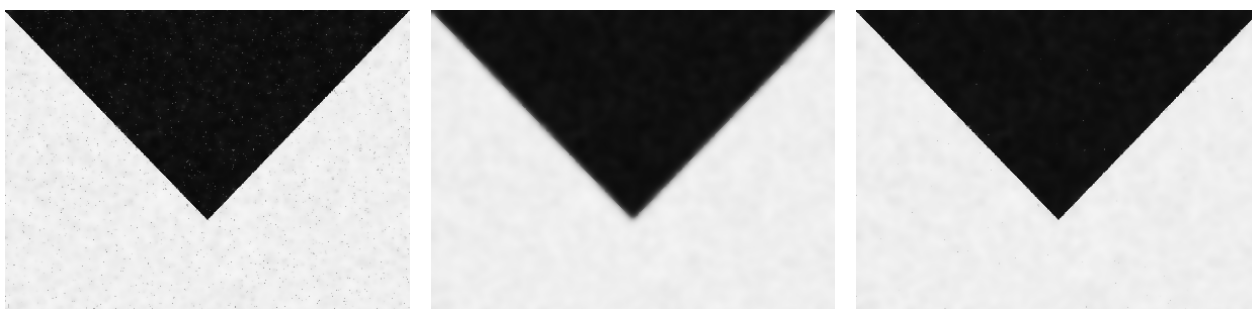
### Wyniki filtracji przy użyciu alternatywnej funkcji stopującej

Charakter filtracji zmienia się w przypadku użyciu funkcji  $g_2$ , jako funkcji stopującej. Wyniki skuteczności filtracji syntetycznego obrazu testowego o niewielkim zaszumieniu  $\sigma_n = 40$  przedstawione są na rysunku 2-19.

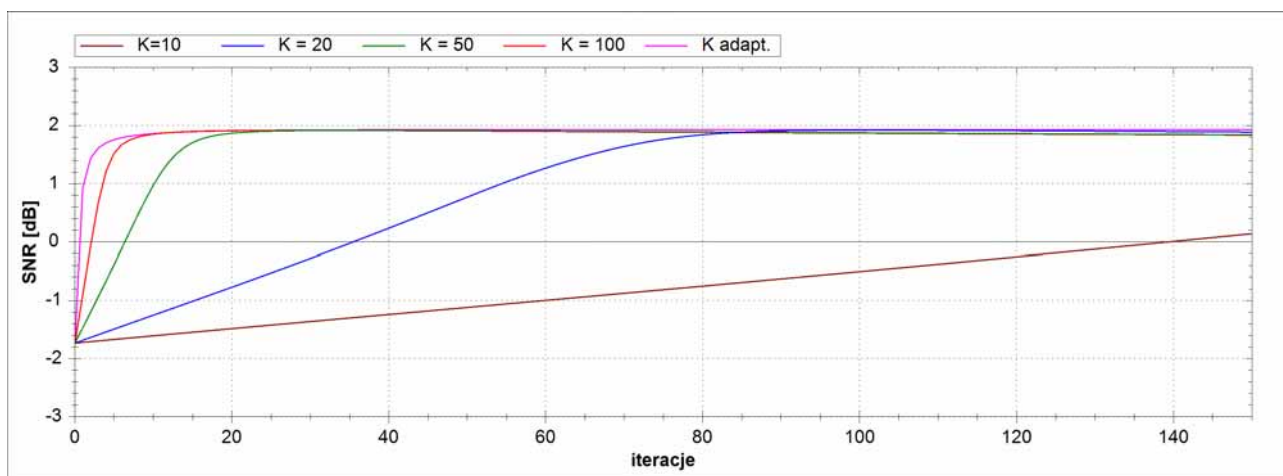


Rysunek 2-19: Miara SNR dla filtracji modelu PM, funkcja  $g_2$ , zaszumienie obrazu testowego  $\sigma_n = 40$ .

Dla funkcji  $g_2$  potrzebne są mniejsze wartości parametrów, aby uzyskać podobną szybkość filtracji, jak w przypadku funkcji  $g_1$  (patrz rysunek 2-4). Proces jest bardziej wrażliwy na użycie zbyt dużego parametru  $K$ , co może powodować znaczne rozmycie krawędzi na obrazie i szybki spadek miary SNR. Jednak w tym przypadku przyjęcie adaptacyjnej wersji algorytmu ze względu na parametr  $K$ , daje bardzo dobre efekty. Filtracja powoduje szybki wzrost miary SNR dla początkowych iteracji, a następnie stabilne zachowywanie brzegów na obrazie w kolejnych krokach.



Rysunek 2-20: Wyniki filtracji dla 30 iteracji z parametrem  $K=20, 50$  oraz dla wersji adaptacyjnej, funkcja  $g_2$ . Początkowe zaszumienie obrazu  $\sigma_n = 40$ .

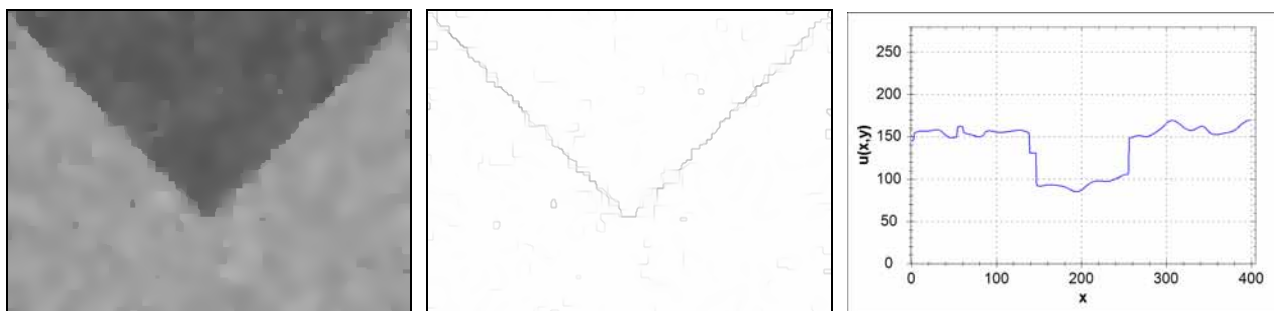


Rysunek 2-21: Miara SNR dla filtracji modelu PM, funkcja  $g_2$ , zaszumienie obrazu testowego  $\sigma_n = 400$

Rysunek 2-20 przedstawia wynik filtracji po 30 iteracjach dla wartości  $K=10, 50$  oraz wersji adaptacyjnej. Widoczne jest, że dla małego  $K$  taka liczba iteracji jest niewystarczająca do pełnego

odszumienia obrazu (patrz wykres 2-19). Większe  $K$  powoduje szybkie rozmywanie brzegów zaraz po usunięciu z obrazu większości szumu. Adaptacyjny dobór współczynnika  $K$  daje bardzo dobre efekty filtracji, które poprawiają się dla kolejnych iteracji.

Charakter filtracji przy użyciu funkcji  $g_2$  jest podobny przy przetwarzaniu obrazu testowego z dużym zaszumieniem  $\sigma_n = 400$  (rysunek 2-21). W tym przypadku nawet małe wartości parametru  $K$  powodują powolny, ale stabilny wzrost miary SNR. Użycie adaptacyjnej wersji algorytmu również w tym przypadku pozwoliło uzyskać dobrą jakość obrazu. Na rysunku 2-22 pokazany jest efekty filtracji mocno zaszumionego obrazu przy użyciu adaptacyjnej wersji algorytmu z funkcją  $g_2$  po 150 iteracjach.



**Rysunek 2-22: Wyniki filtracji dla 150 iteracji,  $K$  adaptacyjne, funkcja  $g_2$ ; detekcja krawędzi; funkcja obrazu dla linii  $y=150$ . Początkowe zaszumienie obrazu  $\sigma_n = 400$ .**

W porównaniu do analogicznej filtracji przy użyciu funkcji  $g_1$  (rysunek 2-18) uzyskuje się obraz, który ma usuniętą większość z zakłóceń kosztem specyficznej degradacji krawędzi. Obiekt na przedstawionym obrazie wynikowym ma delikatnie bardziej rozmyte krawędzie, jednak cały czas możliwe jest jego jednoznaczne rozróżnienie od tła. Również detekcja krawędzi pokazana na rysunku 2-22 wskazuje na możliwość dobrego wyodrębnienia poszukiwanego obiektu na obrazie. Wykres funkcji wybranej linii z obrazu wynikowego wskazuje na zmiękczenie krawędzi, które nadal jednak pozostają stabilne. Znaczną zaletą uzyskanego wyniku jest brak niewielkich jednolitych (pod względem jasności) obszarów powstałych w poprzednim przypadku na skutek błędnego wyostrenia obszarów zaszumionych.

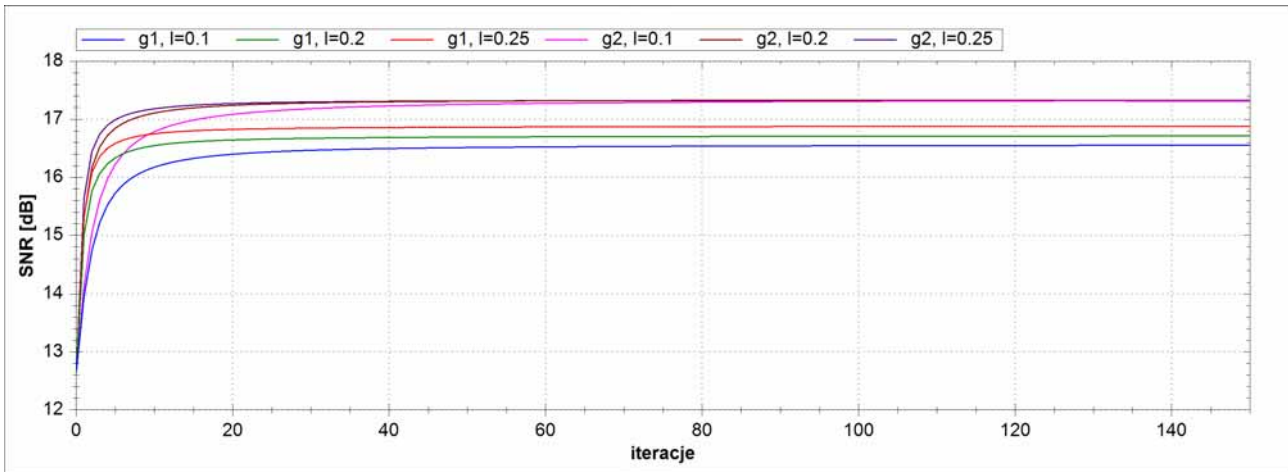
### Wyniki filtracji dla różnych wartości parametru $\lambda$

Filtry realizujące nieliniowe równanie dyfuzji parametryzowane są także współczynnikiem  $\lambda$ . Krótko zostaną zaprezentowane własności filtracji w zależności od wartości tego współczynnika.

Parametr  $\lambda$  w równaniu (2 – 60) nie wpływa znacząco na wyniki uzyskiwane przez zastosowanie filtracji PM. Decyduje on o szybkości dyfuzji przeprowadzanej w każdej iteracji. Aby została zachowana stabilność schematu dyskretyzacji równania różniczkowego, współczynnik ten nie może być większy niż 1. Na rysunku 2-23 przedstawiono miarę SNR w funkcji iteracji dla kilku wartości współczynnika  $l=\lambda/4$ . Przy użyciu funkcji stopującej  $g_1$  oraz  $g_2$ . Eksperyment wykonano dla obrazu testowego o zaszumieniu  $\sigma_n = 40$ .

W przypadku funkcji  $g_2$  filtracja współczynnik  $\lambda$  wpływa jedynie na początkowy przebieg filtracji. Na wykresie wyraźnie widoczny jest wolniejszy wzrost miary SNR dla początkowych iteracji w przypadku zastosowania  $l=0.1$ . Od około 80 iteracji wyniki filtracji (według przyjętej miary) nie zależą znacząco od przyjętej wartości  $\lambda$ .





**Rysunek 2-23: Miara SNR dla filtracji modelu PM, funkcje  $g_2, g_1$ , dla różnych wartości parametru  $l=\lambda/4$ . Zaszumienie obrazu testowego  $\sigma_n = 40$ .**

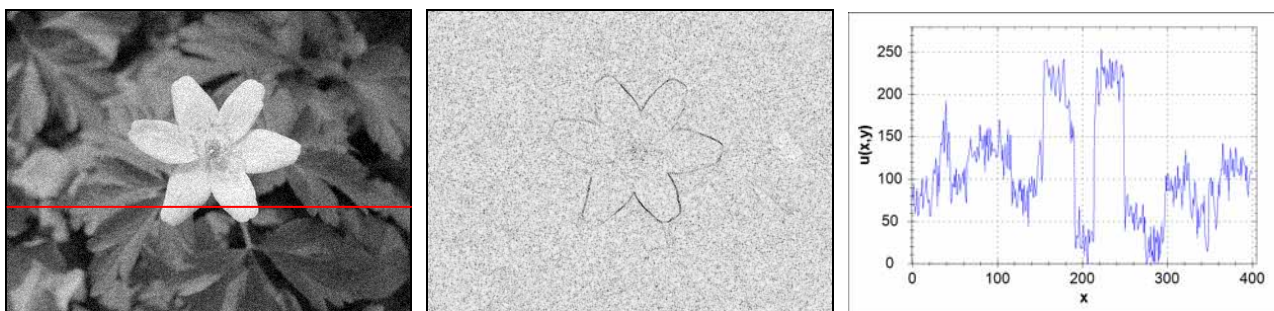
Charakter filtracji zmienia się w zależności od wartości  $\lambda$  przy użyciu funkcji  $g_l$ . W tym przypadku zmniejszenie  $\lambda$  wpływa nie tylko na początkową szybkość filtracji. Również dla bardzo wielu iteracji miara SNR została w znaczący sposób zmniejszona przy zastosowaniu mniejszego współczynnika. Wynika to prawdopodobnie z charakteru funkcji  $g_l$ , której użycie w poprzednich eksperymentach prowadziło do uzyskania wyników, które nie zostały całkowicie odsumione. Przy użyciu adaptacyjnego wyznaczenia parametru  $K$  w początkowej fazie filtracji zostaje usunięte najwięcej zakłóceń. Po automatycznym zmniejszeniu wartości  $K$ , szybkość dyfuzji jest globalnie zmniejszona. Mniejsza szybkość w początkowej fazie odsumiania (przez użycie parametru  $\lambda$  o mniejszej wartości) powoduje, że w chwili przejścia do fazy spowolnionej dyfuzji, obraz jest gorzej odsumiony. Stąd bardziej równoległe wykresy miary SNR dla funkcji  $g_l$ .

W praktyce najczęściej stosuje się maksymalną wartość współczynnika  $\lambda=0.25$ . Nie wyklucza to jednak specyficznych zastosowań, kiedy pożądane jest zmniejszenie szybkości dyfuzji przez mniejszą wartość tego współczynnika.

#### 2.6.4 Porównanie skuteczności filtracji obrazów rzeczywistych

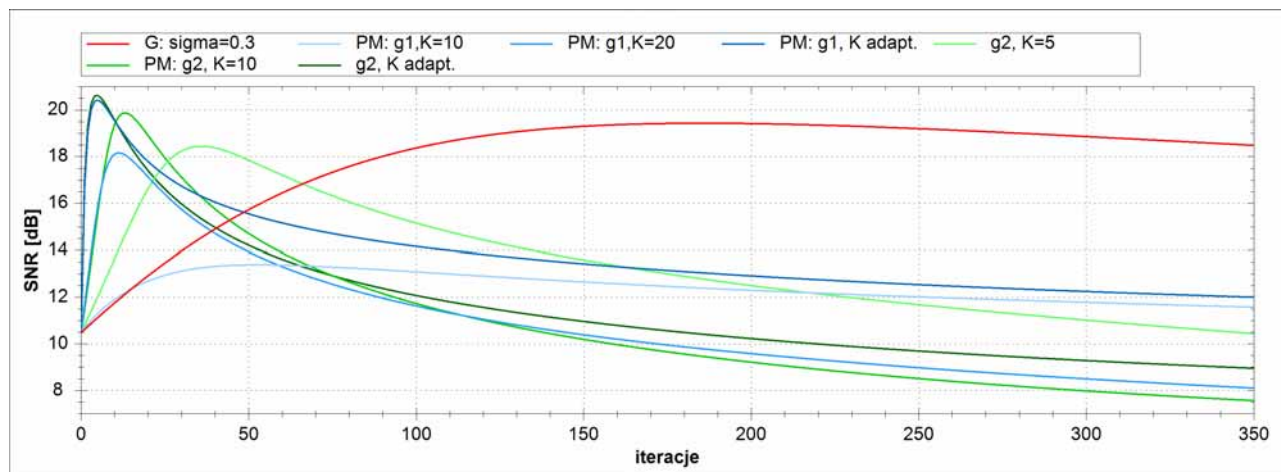
Przebieg filtracji dla obrazów rzeczywistych (na przykład fotografii) może być inny niż dla obrazu wygenerowanego syntetycznie. W poprzednim podrozdziale zaprezentowane zostały wyniki badania obrazu syntetycznego. Pozwoliło to na przybliżenie własności działania filtrów opartych na procesie dyfuzji opisanej równaniami różniczkowymi. Krótko zostanie zaprezentowane porównanie skuteczności działania filtrów na przykładzie rzeczywistego obrazu testowego (rysunek 2-24).

Porównanie obrazu na rysunkach 2-24 oraz oryginalnej wersji 2-5 pokazuje, że wprowadzony szum znacząco wpływa na jakość oraz zawartość znaczeniową. Na podstawie detekcji krawędzi (zrealizowanej jako norma gradientu) możliwe jest odróżnienie jedynie najbardziej istotnych kształtów.



Rysunek 2-24: Zaszumiony obraz testowy nr 1, detekcja krawędzi, przebieg funkcji wybranej linii obrazu ( $y=105$ ). Zaszumienie obrazu  $\sigma_n = 16$ .

Na rysunku 2-25 przedstawiono wykres miary SNR w funkcji kolejnych iteracji procesu przy zastosowaniu różnych wersji filtracji. Dla porównania umieszczono efekt działania filtru opartego na liniowym procesie dyfuzji (kolor czerwony). Ponieważ wprowadzony szum ma rozkład Gaussa, to można się spodziewać, że liniowa filtracja pozwoli na uzyskanie bardzo dobrych efektów. Umieszczono również miary SNR dla filtrów realizujących filtrację nieliniową przy użyciu stałych oraz adaptacyjnych parametrów  $K$  dla funkcji  $g_1$  (wykresy w odcieniach niebieskiego),  $g_2$  (wykresy w odcieniach zielonego).



Rysunek 2-25: Miara SNR dla różnego typu filtracji – rzeczywisty obraz testowy o zaszumieniu  $\sigma_n = 16$ .

Rodzaj filtracji	Nr iteracji	Max SNR [dB]
G: $\sigma=0.3$	186	19,43
PM: $g_1, K=10$	53	13,39
PM: $g_1, K=20$	11	18,16
PM: $g_1, K$ adapt.	5	20,43
PM: $g_2, K=5$	36	18,45
PM: $g_2, K=10$	13	19,88
PM: $g_2, K$ adapt.	5	20,63

Tabela 2: Maksymalna wartość SNR w zależności od zastosowanej filtracji.

Powoli postępująca filtracja liniowa prowadzi do uzyskania dobrych efektów. Po około 180 iteracjach miara SNR wzrosła do 19,43 dB. Użycie filtrów o większym odchyleniu standardowym powodowało szybszą filtrację, jednak po kolejnych iteracjach dawało gorsze efekty (analogicznie jak to było

w przypadku testów na obrazie syntetycznym). W tabeli zebrano maksymalne wartości miar SNR, które udało się uzyskać w przypadku określonego rodzaju filtracji.

Filtry oparte na nieliniowym procesie dyfuzji dają inną charakterystykę wyników niż w poprzednich eksperymentach. Większość z nich pozwala na początkowy znaczny wzrost miary SNR, jednak krótko po tym następuje znaczący spadek. Obserwuje się, że szybkość tego spadku jest większa dla sytuacji, gdy początkowo następowała szybka poprawa jakości obrazu. Ostatecznie spośród przebadanych konfiguracji parametrów jedynie adaptacyjne wersje filtrów oraz przypadek filtracji z funkcją  $g_2$  przy  $K=10$  pozwolił uzyskać wyniki lepsze niż filtracja liniowa. W tym przypadku miara SNR nie pozostawała na ustalonym poziomie, jak to było dla testów na obrazie syntetycznym.

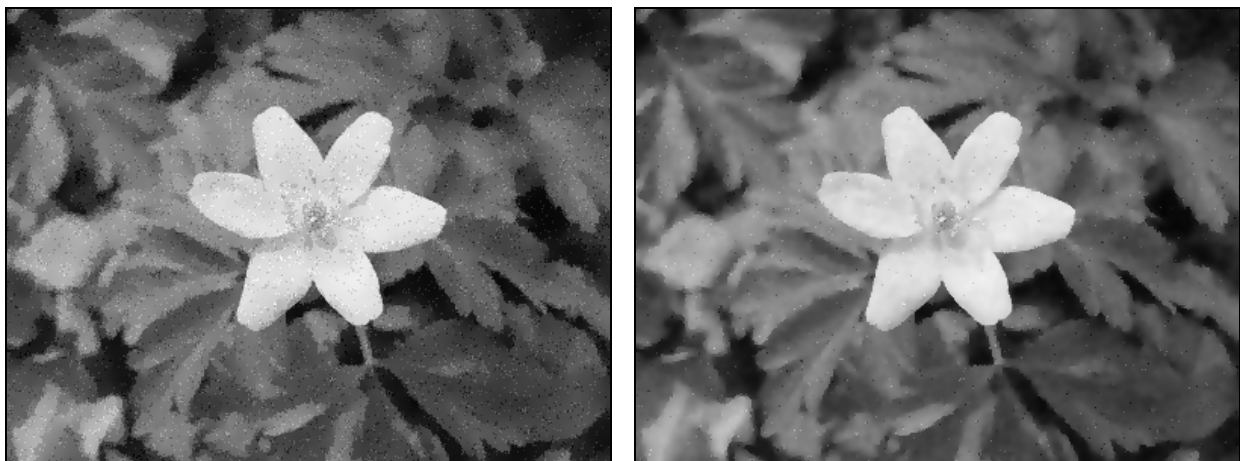
Znaczny spadek jakości dla filtrów nieliniowych wynika z dwóch czynników: specyfiki filtracji oraz zastosowanej miary porównawczej. Miara SNR nie uwzględnia jakości obrazu z punktu widzenia zachowania ostrości brzegów. Wyższa jakość może być uzyskana, jeżeli odpowiadające sobie wartości jasności punktów są zbliżone. Filtracja nieliniowa dąży natomiast do ujednoczenia obszarów, na których nie występują znaczące krawędzie. Może powodować to utratę informacji w przypadku obrazów, gdzie występują naturalne tekstury reprezentujące informacje o obiekcie. Filtracja liniowa rozmywa obszary tekstur w sposób jednostajny. Filtracja nieliniowa dąży do szybkiego ujednoczenia tych obszarów, na których norma gradientu jest zbyt mała. Ostatecznie w przypadku liniowej filtracji utrata informacji na obrazie w związku z rozmyciem krawędzi jest wolniejsza niż utrata informacji w związku z utratą tekstury przy filtracji nieliniowej.

W przypadku badania obrazu syntetycznego sytuacja była odwrotna. Istota informacji o obrazie zawarta była w krawędziach obiektu. Obiekt nie był pokryty teksturą, która niosłaby ze sobą dodatkową informację. Filtracja nieliniowa powodowała usuwanie zakłóceń z obrazu wraz z jednoczesną degradacją informacji o krawędziach. Filtracja nieliniowa usuwała zanieczyszczenia wyostrzając (do pewnego stopnia) krawędzie. W związku z tym nie występowała dodatkowa utrata informacji, więc przebieg procesu stabilizował się aż do chwili, w której krawędzie zaczynały być rozmywane. Stąd też wynikała inna charakterystyka dla miary SNR określającej jakość otrzymanego po filtracji obrazu.

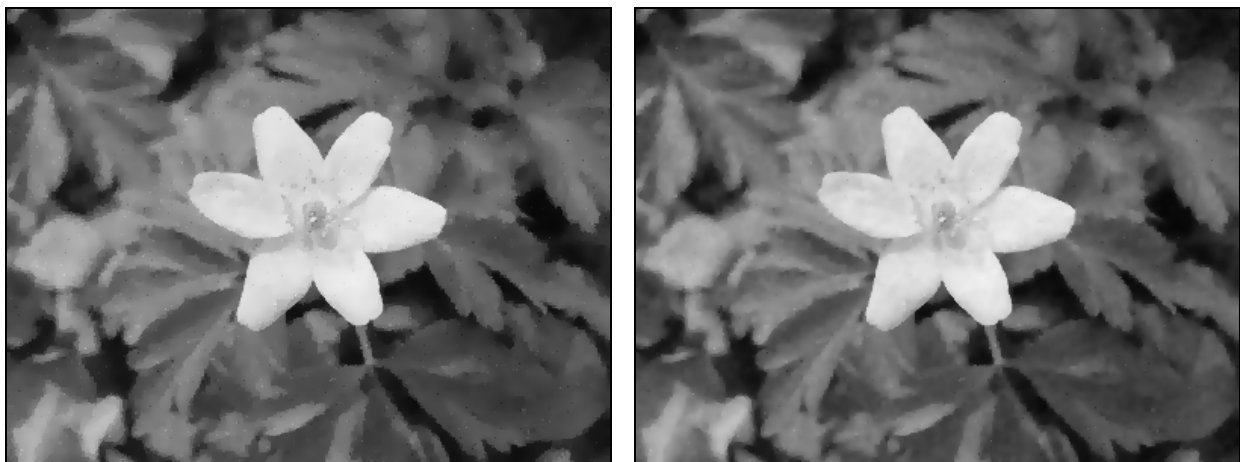
Na rysunku 2-26 przedstawiono obrazy wynikowe dla wybranych iteracji przy zastosowaniu filtracji liniowej oraz adaptacyjnych wersji filtracji nieliniowej. Do prezentacji wybrano te iteracje, dla których uzyskano maksymalną wartość SNR. W dolnym rzędzie znajduje się powiększony dwukrotnie wynik detekcji krawędzi. Należy zwrócić uwagę na rozmycie krawędzi w przypadku zastosowania filtru liniowego. Filtry nieliniowe powodują natomiast wyostrenie tych krawędzi, które charakteryzują się dużą normą gradientu. W przypadku drobnych detali (tekstura powierzchni liści itd.) informacje są rozmywane szybciej niż jest to w przypadku filtru liniowego. Efekt ten jest jeszcze lepiej widoczny dla filtracji nieliniowej przy ustalonym parametrze  $K$  – otrzymane wyniki przedstawione są na rysunkach 2-27 oraz 2-28.



Rysunek 2-26: Wynik działania filtrów: liniowego  $\sigma=0.8$  (iteracja 180), PM  $g_1$  adaptacyjny (iteracja 5), PM  $g_2$  adaptacyjny (iteracja 5); dolny rząd – detekcja krawędzi.



Rysunek 2-27: Wynik działania filtru nieliniowego  $g_1$ :  $K=10$ ,  $it=53$ ,  $SNR=13,39$  dB;  $K=20$ ,  $it=11$ ,  $SNR=18,16$  dB.



Rysunek 2-28: Wynik działania filtru nieliniowego  $g_2$ :  $K=5$ ,  $it=36$ ,  $SNR=18,45$  dB;  $K=10$ ,  $it=13$ ,  $SNR=19,88$  dB.

Na rysunku 2-27 widoczne są charakterystyczne pojedyncze punkty, które nie zostały usunięte przez proces filtracji wykorzystujący funkcję  $g_1$ . Punkty te charakteryzują się dużą wartością gradientu i przy ustalonym parametrze  $K$  są błędnie traktowane jako element krawędzi. Często pozostają widoczne nawet dla bardzo wielu iteracji. Efekt taki nie występuje w przypadku użyciu funkcji  $g_2$ . Szczególnie ostatni z przedstawionych eksperymentów dla  $K=10$  (rysunek 2-28) pozwolił na uzyskanie bardzo dobrego wizualnie efektu. Szum występujący na obrazie został w dużej części usunięty, krawędzie pozostały stabilne, a drobne szczegóły tekstury na liściach nie zostały znacząco zdegradowane.

Właściwy dobór współczynników procesu jest bardzo istotny z punktu widzenia jakości uzyskiwanych wyników. Równie istotne jest określenie liczby iteracji, które pozwalają na uzyskanie obrazu o pożądanej jakości. W praktyce określenie tych parametrów często zależy od rodzaju przetwarzanego obrazu oraz od stopnia jego zaszumienia.

## 2.7 Podsumowanie metod filtracji obrazów

W rozdziale został zaprezentowany przegląd metod przetwarzania obrazów opartych na procesach dyfuzyjnych opisanych równaniami różniczkowymi cząstkowymi. Przedstawiony został liniowy oraz nieliniowy model procesu [79] pozwalającego na filtrację obrazów. Pokazano w jaki sposób odpowiednia dyskretyzacja równania różniczkowego ([30], [59]) pozwala na konstrukcję filtru przestrzennego o określonych właściwościach. Przedstawiony został również przegląd podstawowych metod regularyzacji układów równań pozwalających na wyznaczenie stabilnych numerycznie obliczeń. Szczególnym rodzajem regularyzacji, znacząco wpływającej na charakter działania filtru jest wprowadzenie anizotropowego współczynnika dyfuzji, co zostało zaproponowane w [107].

Zastosowanie nieliniowego modelu typu *Perona-Malik* daje bardzo dobre wyniki. Przeprowadzono szereg doświadczeń, które pozwoliły na określenie praktycznych właściwości tak skonstruowanego filtru. Wyniki eksperymentów pokazują, że filtr tego typu pozwala na skuteczne odszumianie obrazów. Po odpowiednim doborze parametrów uzyskane wyniki były porównywalne, a wielokrotnie lepsze od filtru opartej na procesie liniowym (według miary SNR). Efekt wizualny zastosowania filtracji nieliniowej prowadzi do wyostrenia krawędzi na obrazie. Wynik filtracji liniowej i nieliniowej o porównywalnej mierze SNR znacząco różni się z punktu widzenia wyrazistości krawędzi. Dla modelu PM brzegi obiektów przez bardzo wiele iteracji pozostawały stabilne.

Otwartym problemem pozostaje kwestia doboru parametrów oraz liczby iteracji potrzebnych do uzyskania optymalnych wyników filtracji. Została przetestowana adaptacyjna wersja algorytmu filtracji PM wyznaczająca wartość współczynnika  $K$  na podstawie średniej normy gradientu (na podstawie propozycji przedstawionych w [79]). W wielu przypadkach wersja ta pozwala na uzyskanie bardzo dobrych wyników prowadzących do osiągnięcia stabilnego punktu minimum w procesie filtracji. Jednak w przypadku obrazów rzeczywistych (zdjęcia, obrazy z dużą liczbą szczegółów) proces PM jest bardzo wrażliwy na właściwy dobór parametru oraz liczbę iteracji. Za duża liczba kroków filtracji prowadzi do znacznej utraty jakości obrazu zarówno pod względem wartości miary SNR, jak i ostrości krawędzi.

Przeprowadzone eksperymenty pozwoliły na dokładną analizę metod filtracji opartych na procesach dyfuzji opisanych równaniami różniczkowymi cząstkowymi. Wynikiem tego jest propozycja uwzględnienia w algorytmie globalnej wiedzy o strukturze geometrycznej obrazu zaprezentowana w rozdziale 3. W rozdziale 5 zostanie zaprezentowane uogólnienie metod liniowej i nieliniowej filtracji na przestrzeń kwaternionów jednostkowych  $S^3$  z uwzględnieniem specyfiki parametryzacji przestrzeni rotacji poruszającego się obiektu.

## Rozdział 3

# Propozycja uwzględnienia w procesie filtracji globalnej wiedzy o strukturze geometrycznej obrazu

Przedstawione w rozdziale 2.5 modele procesu dyfuzji same w sobie realizują uwzględnienie dodatkowej wiedzy w procesie filtracji. Standardowy izotropowy model liniowy, którego rozwiązaniem jest splot funkcji  $u$  z jądrem Gaussa nie bierze pod uwagę żadnych dodatkowych czynników. Konstrukcja tego typu filtracji zakłada jedynie, że szum, który powinien być usunięty ma rozkład normalny o zerowej wartości średniej. Jednak często w praktyce struktura zakłóceń nie jest znana, więc ten sam filtr stosuje się do szumu o zróżnicowanym rozkładzie.

Model nieliniowy *Perona-Malik* w prosty sposób uwzględnia strukturę obrazu przez wprowadzenie estymatora krawędzi w postaci uzależnienia współczynnika dyfuzji od gradientu funkcji  $\nabla u$ . Anizotropowy model zaproponowany przez Weickerta [107] jeszcze bardziej rozbudowuje czynnik opisujący strukturę obrazu, co pozwala na polepszenie jakości filtracji dla pozostałych cech charakterystycznych (np. narożniki, struktury o wysokiej spójności, itd.).

W rozdziale 3 zostanie przedstawiona propozycja autora uwzględnienia w procesie filtracji informacji wynikających ze struktur geometrycznych występujących na obrazie opisanych globalnie (w przeciwieństwie do informacji lokalnych takich jak gradient lub tensor dyfuzji). Przedstawiona propozycja jest uogólnieniem metod opisanych w pracach [30], [85], [86], w których proces filtracji projektowany jest pod kątem specyficznego zastosowania. W podrozdziale 3.1 zostaną przedstawione metody zaprezentowane w cytowanych pracach, a następnie zaproponowany zostanie sposób sprzężenia nieliniowego procesu dyfuzji z przestrzenią Hougha. W podrozdziale 3.2 znajduje się krótkie wprowadzenie do konstrukcji transformaty Hougha wykorzystywanej w dalszej konstrukcji sprzężenia.

### **3.1 Przegląd istniejących metod uwzględniających dodatkowe informacje**

Podstawowym sposobem uwzględnienia wiedzy apriorycznej jest modyfikacja współczynnika dyfuzji  $c$  (lub  $D$  w przypadku realizacji filtracji anizotropowej). W zależności od tego, w jaki sposób współczynnik  $c$  jest modyfikowany można uzyskać modele o różnych właściwościach. Przedstawione

zostaną dwa podejścia, które występują w literaturze. Pierwsze z nich nie jest bezpośrednim uwzględnieniem wiedzy apriorycznej, jednak obrazuje, w jaki sposób informacje dodatkowe mogą wpływać na proces filtracji. Model ten służyć będzie jako podstawa do uogólnienia metody wykorzystującej własności geometryczne obiektów znajdujących się na obrazie.

### 3.1.1 Uwzględnienie informacji z wielokanałowego toru pomiarowego

W pracy [30] rozpatrywane jest przetwarzanie danych uzyskanych z badania metodą rezonansu magnetycznego MRI. Przedstawione są metody użyte do dyskretyzacji zagadnienia początkowego (2 – 43) dla przestrzeni trójwymiarowej (ponieważ dane uzyskane z badania obrazują obszary trójwymiarowe). Dodatkowo wprowadzona jest modyfikacja, która uwzględnia wielokanałowe przetwarzanie informacji. W trakcie pojedynczego rozpatrywanego badania MRI są zapisywane informacje uzyskane z dwóch kanałów pomiarowych. Oba kanały były skalibrowane tak, aby odczyty dotyczyły tych samych współrzędnych w przestrzeni 3D. Do późniejszego przetwarzania tak zgromadzonych danych proponowany jest następujący układ równań różniczkowych.

$$\frac{\partial u_1}{\partial t} = \operatorname{div}(c_1(x, y, z, u, t) \nabla u_1) \quad (3 - 1)$$

$$\frac{\partial u_2}{\partial t} = \operatorname{div}(c_2(x, y, z, u, t) \nabla u_2), \quad (3 - 2)$$

gdzie współczynniki  $c_1, c_2$  są określone podobnie, jak to było w przypadku standardowego modelu *Perona-Malik* w równaniu (2 – 50). Ponieważ wartości z obu kanałów odnoszą się do tego samego rzeczywistego obszaru, więc funkcje estymatorów krawędzi w obu tych przypadkach powinny być ze sobą skorelowane. W związku z tym zaproponowane zostało, aby w każdym kroku wyznaczony został wspólny współczynnik dyfuzji na podstawie informacji z obu kanałów:

$$c_1(x, y, z, u, t) = g(\|\nabla u_1\|) \quad (3 - 3)$$

$$c_2(x, y, z, u, t) = g(\|\nabla u_2\|) \quad (3 - 4)$$

$$c_c(x, y, z, u, t) = f(c_1, c_2) = f(\|\nabla u_1\|, \|\nabla u_2\|) = g\left(\sqrt{\|\nabla u_1\|^2 + \|\nabla u_2\|^2}\right) \quad (3 - 5)$$

### 3.1.2 Uwzględnienie informacji dotyczących własności geometrycznych

Praca [85], która następnie została zmodyfikowana i rozszerzona w [86] prezentuje sposób uwzględnienia geometrycznych własności obiektów umieszczonych na obrazie. Również w tym przypadku autorzy biorą pod uwagę obrazy otrzymywane w trakcie badania serca metodą rezonansu magnetycznego. Analizowane dane są czterowymiarowe: 3 wymiary dotyczą przestrzeni badanego obiektu, czwarty wymiar dotyczy czasu, w trakcie którego badanie było wykonywane. Używany model filtracji oparty jest na standardowym modelu *Perona-Malik* zaprezentowanym w poprzednich rozdziałach. Uogólnienie dla przypadku czterowymiarowego jest przeprowadzone bez dodatkowych modyfikacji. Uwzględnienie wiedzy apriorycznej jest realizowane przez zmianę współczynnika dyfuzji  $D$  podobną, jaka została zastosowana w



poprzednim przypadku. Proponowana w pracach filtracja opisana jest następującym równaniem różniczkowym:

$$\frac{\partial u(x, y, z, \tau, t)}{\partial t} = \operatorname{div}(D \nabla u), \quad (3 - 6)$$

gdzie współczynnik dyfuzji  $D$  jest iloczynem dwóch elementów: macierzy współczynnika  $D^{(G)}$  realizującej wielowymiarową funkcję stopującą odpowiadającą równaniu (2 – 47) oraz macierzy wag  $D^{(W)}$  uwzględniających wiedzę dodatkową.

$$D = D^{(G)}(u)W(x, y, z, \tau) \quad (3 - 7)$$

Macierz  $D^{(G)}$  jest macierzą diagonalną, której elementy  $D_{rr}^{(G)}$  reprezentują wartości współczynnika dyfuzji dla kolejnych wymiarów problemu:

$$D_{rr}^{(G)} = g_r \left( \|\nabla u\|_r \right) = \frac{1}{1 + \left( \frac{\|\nabla u\|_r}{k_r} \right)^2}, \quad (3 - 8)$$

gdzie  $r \in \{x, y, z, \tau\}$  jest indeksem wymiaru, natomiast  $g_r$  oznacza funkcję stopującą dla wymiaru  $m$ . Współczynniki  $k_r$  są obliczane analogicznie, jak to było zaproponowane w pracy [79], czyli w każdej iteracji na podstawie oszacowania szumu (0,9 wartości całki histogramu gradientu dla wybranego wymiaru). W ogólności macierz  $D_{rr}^{(G)}$  nie musi być diagonalna, dzięki czemu otrzymywany proces nabiera cech anizotropowych. Jednak w cytowanej pracy taka możliwość jest jedynie zasygnalizowana bez dalszej analizy.

Macierz współczynników  $D^{(W)}$  jest tworzona na podstawie informacji apriorycznej typowej dla danego procesu. Autorzy podają, że w przypadku przetwarzania danych z rozpatrywanego badania można określić następujące informacje użyteczne w procesie filtracji.

### Struktura geometryczna

- Obserwowany obiekt ma w przybliżeniu kształt cylindryczny
- W płaszczyźnie  $x$ - $y$  obiekt ma kształt okręgu (lub nieznacznie zdeformowanego okręgu), którego położenie nie zmienia się w trakcie całego badania (lub zmienia się w niewielkim stopniu).
- Promień obiektu wzdłuż osi  $z$  zmienia się w sposób ciągły – można przyjąć, że piksel na krawędzi obiektu nie ma bezpośredniego sąsiedztwa z krawędzią wzdłuż osi  $z$ .

Zakładając, że znany jest środek okręgu  $(x_0, y_0)$  reprezentującego obiekt badania określa się współczynniki wag, które powodują, że dyfuzja jest zatrzymana w kierunku prostopadłym do okręgu. Dzięki temu krawędzie obiektu nie zostaną rozmyte z otoczeniem w przypadku, gdyby kontrast był w tym przypadku zbyt mały. W cytowanej pracy proponowane jest przyjęcie następujących wag:

$$D_{xx}^{(W)}(x, y) = |\sin(\theta(x, y))| \quad (3 - 9)$$

$$D_{yy}^{(W)}(x, y) = |\cos(\theta(x, y))|, \quad (3-10)$$

$$\text{gdzie } \theta(x, y) = \arctan\left(\frac{y - y_0}{x - x_0}\right) \quad (3-11)$$

jest kątem określającym lokalnie zdefiniowany układ współrzędnych biegunowych.

Zmienność obiektu wzdłuż osi  $z$  modeluje się również za pomocą funkcji trygonometrycznej przy uwzględnieniu odpowiednio dobranych współczynników:

$$D_{zz}^{(W)}(z) = |\sin(\omega_z z + \phi_z)| \quad (3-12)$$

Parametry  $\omega_z, \phi_z$  są wyznaczane tak, aby współczynnik  $D_{zz}^{(W)}$  odpowiadał danym reprezentowanym na obrazie. Autorzy nie podają jednak metody na wyznaczenie tych parametrów, co pozwala przypuszczać, że były one określane apriorycznie metodą prób i błędów lub przy użyciu innych źródeł wiedzy.

### Dynamika obiektu

- Korelacja danych zmienia się w czasie i zależy od szybkości, z jaką następują fazy skurczowe oraz rozkurczowe serca.

Oznacza to, że dane, które zostały zarejestrowane w czasie szybkiej akcji serca są ze sobą słabiej skorelowane ze względu na dużą zmienność procesu w czasie. W sytuacji, gdy serce porusza się wolniej dane następujące po sobie (wzdłuż osi  $\tau$ ) są ze sobą bardziej skorelowane. W cytowanej pracy proponuje się przyjąć następujący współczynnik wagowy, który w przybliżony sposób opisuje rytm skurczowo-rozkurczowy:

$$D_{\tau\tau}^{(W)}(\tau) = |\cos(\omega_\tau z + \phi_\tau)| \quad (3-13)$$

Model ten jest znacznie uproszczony, gdyż nie uwzględnia asymetrycznego sposobu bicia serca (różne tempo skurczów i rozkurczów). Określenie parametrów  $\omega_\tau, \phi_\tau$  również wymaga dodatkowych obliczeń, dla których nie podano automatycznej procedury. Częstotliwość  $\omega_\tau$  proponuje się przyjąć jako:

$$\omega_\tau = \frac{2\Pi}{\tau_c}, \quad (3-14)$$

gdzie  $\tau_c$  jest okresem cyklu bicia serca (czasem pomiędzy kolejnymi skurczami). Fazę  $\phi_\tau$  można przyjąć równą 0, jeżeli badanie rozpoczęło się w chwili, gdy rozpoczynał się okres spoczynku.

Przedstawiony model obiektu jest uproszczony. Również diagonalna macierz współczynników wagowych  $D^{(W)}$  jest zrealizowana bez użycia dodatkowych korelacji między poszczególnymi wymiarami. Korelacje takie są możliwe przez uwzględnienie bardziej skomplikowanego modelu, który również na poziomie współczynników realizowałby anizotropowy charakter procesu. W takim przypadku macierz  $D^{(W)}$  nie byłaby diagonalna. W pracy [86] zostało pokazane, że przyjęcie tak zmodyfikowanego współczynnika dyfuzji pozwala na przeprowadzenie filtracji o nowych właściwościach. Proces miał ciekawe właściwości przy przetwarzaniu syntetycznych danych, które reprezentowały nie tylko okręgi, ale również koncentryczne

kwadraty. Współczynniki wagowe określone równaniami (3 – 7), (3 – 8) pozwoliły na wprowadzenie dodatkowej informacji o przetwarzanym obiekcie.

### 3.2 Standardowa transformacja Hougha

W niniejszej pracy proponowana jest metoda uogólnienia sposobu uwzględniania dodatkowej informacji w procesie filtracji. Jest ona oparta na próbie dodatkowego określenia kształtu (lub kształtów) struktury interesujących obiektów na przetwarzanym obrazie. Zmodyfikowana postać estymatora będzie oparta na przestrzeni Hougha, której podstawowe właściwości zostaną krótko zaprezentowane.

#### 3.2.1 Konstrukcja transformacji

Transformacja Hougha polega na wyznaczeniu wartości w przestrzeni parametrycznej (o postaci zależnej od przyjętej parametryzacji) odpowiadającym krzywym, które znajdują się na przetwarzanym obrazie. Oryginalna wersja transformacji realizowała wykrywanie linii prostych, co zostało opisane we wniosku patentowym złożonym w 1962 roku. Jej sformułowanie do wykrywania krzywych analitycznych podane zostało w [24]. Zbiór punktów należących do pewnej z góry określonej krzywej  $f$  (zdefiniowanej w przestrzeni  $R^2$  przez  $N$  parametrów  $a_1, \dots, a_N$ ) może być opisany jako [113]:

$$\Lambda_0 = \{(x, y) \in R^2 : f_{(a_1, \dots, a_N)}(x, y) = 0\} \quad (3 - 15)$$

Z punktu widzenia transformacji interesująca jest relacja opisująca projekcję wsteczną, czyli określającą zbiór parametrów krzywej  $f$ , które dla ustalonego punktu  $(x, y)$  spełniają równanie krzywej. W równaniu zostaje zamienione znaczenie ustalonych parametrów i zmiennych:

$$\Lambda_T = \{(a_1, \dots, a_N) \in R^N : f_{(x, y)}(a_1, \dots, a_N) = 0\} \quad (3 - 16)$$

Transformacja Hougha  $HT(a_1, \dots, a_N)$  dla obrazu  $u(x, y)$  określonego na  $\Omega$  jest określona jako:

$$HT(a_1, \dots, a_N) = \sum_{(x_i, y_i) \in \Omega} h_{(x_i, y_i)}(a_1, \dots, a_N), \quad (3 - 17)$$

gdzie

$$h_{(x_i, y_i)}(a_1, \dots, a_N) = \begin{cases} 1 & \text{dla } f_{(x_i, y_i)}(a_1, \dots, a_N) = 0 \\ 0 & \text{w przeciwnym wypadku} \end{cases} \quad (3 - 18)$$

Podstawowym zastosowaniem transformacji Hougha jest detekcja linii prostych, które mogą być opisane następującym równaniem:

$$x \cos(\alpha) + y \sin(\alpha) - r = 0 \quad (3 - 19)$$

Przestrzeń parametrów w tym przypadku jest dwuwymiarowa  $(r, \alpha)$ , gdzie  $r \in [-v, v]$ ,  $\alpha \in [0, \alpha_{\max}]$ . Zwykle przyjmuje się  $\alpha_{\max} = \Pi$ , natomiast wartość  $v$  jest uzależniona od wielkości obszaru  $\Omega$ , na którym określono obraz.

Transformacja Hougha dla linii prostych i obrazów binarnych jest szczególnym przypadkiem transformacji Radona określonej wzorem:

$$RT(r, \alpha) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} u(x, y) \delta(x \cos(\alpha) - y \sin(\alpha) - r) dx dy \quad (3 - 20)$$

Jako wynik działania transformaty otrzymuje się wartości punktów w przestrzeni parametrów (zwaną przestrzenią Hougha) zdefiniowanej przez przyjęcie odpowiedniej parametryzacji. Znalazienie maksymalnych wartości elementów przestrzeni pozwala na określenie najbardziej istotnych struktur występujących na obrazie. Metoda pozwala więc na znalezienie parametrów krzywych, które z największym prawdopodobieństwem występują na obrazie. Nie jest to tożsame z samymi krzywymi – detekcja konkretnego położenia krzywych wymaga dalszej analizy.

### 3.2.2 Aspekty implementacyjne

W praktyce wszystkie parametry przestrzeni Hougha podlegają kwantyzacji. Dla mniejszych kroków kwantyzacji uzyskuje się większą dokładność detekcji obiektu. Z drugiej strony należy pamiętać, że pociąga to za sobą wydłużenie czasów operacji, co przy dużej złożoności obliczeniowej całej transformaty może być znaczące.

Wyznaczenie przestrzeni Hougha realizowane jest na podstawie wyników detekcji krawędzi na obrazie. Tworzona jest tablica zwana akumulatorem reprezentująca przestrzeń parametrów. Rozmiar akumulatora zależy od kroku kwantyzacji dla każdego parametru.

Transformata dla linii prostych obrazu binarnego może być wyznaczona przez następującą bezpośrednią implementację:

---

#### Algorytm 1: Transformacja Hougha dla linii prostych na obrazach binarnych.

**Parametry:**  $s_\alpha, s_r$  - liczba kroków kwantyzacji odpowiednio dla parametrów  $\alpha, r$

**Oznaczenia:**  $A$  – tablica akumulatora przestrzeni Hougha;  $k_\alpha = \frac{180}{s_\alpha}$  - krok kwantyzacji parametru  $\alpha$ ;  $k_r$  - krok kwantyzacji parametru  $r$

**Inicjalizacja:** Przygotuj tablicę akumulatora  $A$  o rozmiarze  $[s_r, s_\alpha]$ ; początkowo tablica zawiera wartości 0

- krok 1:** Dla każdego piksela obrazu  $(x_i, y_i) \in \Omega$ , którego wartość  $u(x_i, y_i) = 1$ :
  - krok 2:** przypisz  $i = 0$
  - krok 3:** wyznacz  $\alpha' = i \cdot k_\alpha$
  - krok 4:** wyznacz  $r' = x_i \cos(\alpha') + y_i \sin(\alpha')$
  - krok 5:** zwiększ odpowiednią komórkę tablicy  $A(\lfloor r'/k_r, \alpha' \rfloor) = A(\lfloor r'/k_r, \alpha' \rfloor) + 1$
  - krok 6:** przypisz  $i \leftarrow i + 1$
  - krok 7:** jeżeli  $i < s_\alpha$ , wróć do kroku 4
-

Widać, że tak zastosowany algorytm ma dużą złożoność obliczeniową. Jednak z drugiej strony w prosty sposób można zastosować równoległe obliczenia pozwalające na sprzętową realizację przekształcenia [95]. Obecnie produkuje się wiele specjalizowanych układów wyznaczających tego rodzaju transformatę.

Przy rozpatrywaniu obrazów w skali szarości (zamiast binarnych) w praktyce stosuje się progowanie w warunku sprawdzanym w kroku 2 algorytmu, który przyjmuje postać:

$$u(x_i, y_i) \geq \xi, \quad (3 - 21)$$

gdzie  $\xi$  oznacza jasność piksela. W przypadku obrazów zaszumionych progowanie pozwala uzyskać mniej zakłóconą przestrzeń Hougha. Podejście jest często używane w przypadku detekcji kształtów na podstawie krawędzi. Przy takim zastosowaniu można również zamienić zliczanie pikseli w kroku 6 na sumowanie wartości:

$$A(\lfloor r' / k_r \rfloor, \alpha') = A(\lfloor r' / k_r \rfloor, \alpha') + u(x_i, y_i) \quad (3 - 22)$$

Wybór, która wersja obliczeń ma być stosowana zależy będzie od konkretnego zastosowania.

W przypadku, gdy na rozpatrywanym obrazie znajduje się duża liczba szczegółów może okazać się, że z otrzymanej tablicy akumulatorowej przestrzeni Hougha bardzo trudno jest wyznaczyć maksimum reprezentujące poszukiwany kształt. Przetworzenie akumulatora przez zastosowanie odpowiednio dobranego filtru przestrzennego (opartego na dyskretnym splocie opisanym w rozdziale 2.4) pozwala na uwypuklenie interesujących maksimumów. Można pokazać [113], że dla algorytmu realizującego wyznaczanie linii prostych na obrazie bardzo dobre efekty pozwala uzyskać zastosowanie tzw. filtru *butterfly*. W praktyce filtr ten realizuje się przy pomocy maski o następujących parametrach:

$$M_B = \begin{bmatrix} 0 & -2 & 0 \\ 1 & 2 & 1 \\ 0 & -2 & 0 \end{bmatrix} \quad (3 - 23)$$

Sposób filtracji parametrów zależy będzie od używanego wariantu metody. Maską  $M_B$  używana jest w przypadku transformacji dla linii prostych.

Powstało również bardzo wiele modyfikacji oryginalnego algorytmu. Wśród nich warto wspomnieć o kilku wyznaczających główne kierunki badań dotyczących tego typu transformacji:

- modyfikacje oparte na uogólnionej transformacji Hougha [5] pozwalające na detekcję dowolnych kształtów (nie tylko krzywych, które mają analityczny opis), np. zakładające nieregularność kształtów poszukiwanego obiektu [84]; optymalizacje i implementacje transformaty do przetwarzania danych w czasie rzeczywistym [102];
- hierarchiczna transformacja Hougha polegająca na wyznaczaniu transformaty dla linii prostych na podobszarach obrazu i dalszego łączenia ich na wyższych poziomach [109];
- adaptacyjna transformacja Hougha [44], gdzie automatycznie dobierany jest rozmiar akumulatora;

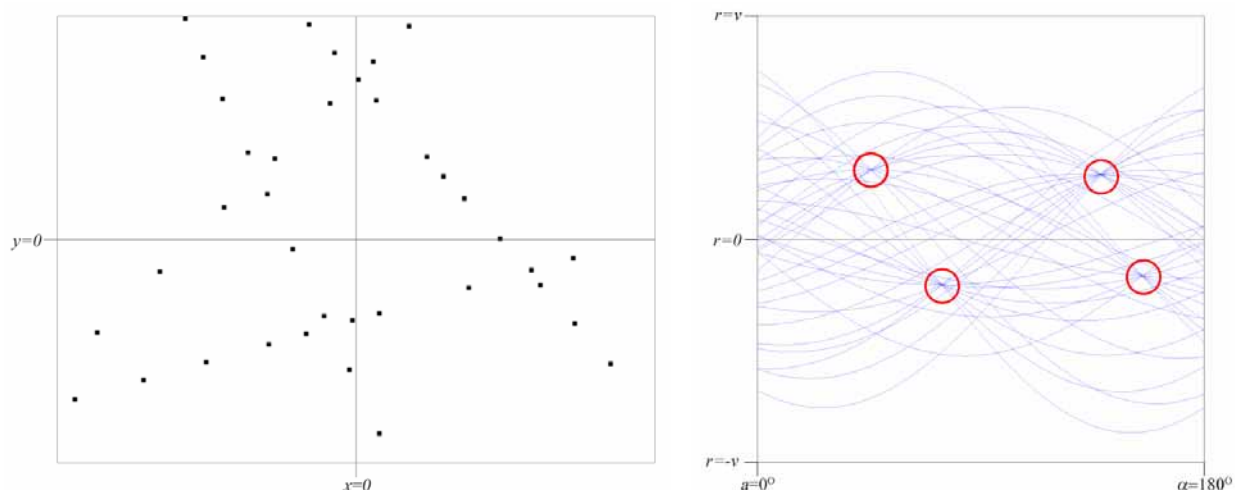
- szybka transformacja Hougha [36] oparta na dynamicznym zwiększaniu rozdzielczości przestrzeni parametrów w przypadku, gdy w danym obszarze może wystąpić poszukiwany kształt;
- probabilistyczna transformacja Hougha [58], [73] operująca na losowo wybranej podgrupie pikseli, co w praktyce często wystarcza do detekcji poszukiwanych kształtów;
- transformacja dla obiektów trójwymiarowych odwzorowanych na obrazie [37], [99].

Pomimo dużej złożoności obliczeniowej (zwłaszcza dla bardziej skomplikowanych obiektów) transformacja Hougha ma wiele zalet i są nad nią prowadzone intensywne badania. Wśród najważniejszych właściwości tego przekształcenia należy wymienić:

- możliwość prowadzenia obliczeń równoległych ze względu na niezależność przekształcenia dla każdego elementu obrazu,
- możliwość przetwarzania obrazów częściowo zniekształconych oraz zaszumionych – właściwość ta, która można by nazwać małą wrażliwością na sygnał szumu jest bezpośrednio wykorzystywana w prezentowanej metodzie w rozdziale 3.3.3,
- możliwość wykrywania różnego rodzaju obiektów – zarówno opisanych krzywymi analitycznymi, jak i kształtów o dowolnej orientacji i skali (uogólniona transformacja Hougha).

### 3.2.3 Przykłady zastosowania

W wersji transformacji dla linii prostych przestrzeń Hougha jest dwuwymiarowa. Każdy punkt obrazu poddany transformacji jest źródłem jednej krzywej opisanej równaniem parametrycznym (2 – 100). Na rysunku 3-1 pokazany jest przykładowy obraz zawierający tylko pojedyncze punkty oraz odpowiadająca mu przestrzeń akumulatora (obraz o wymiarach 400x300, na rysunkach pokazany jest w negatywie dla lepszej czytelności).



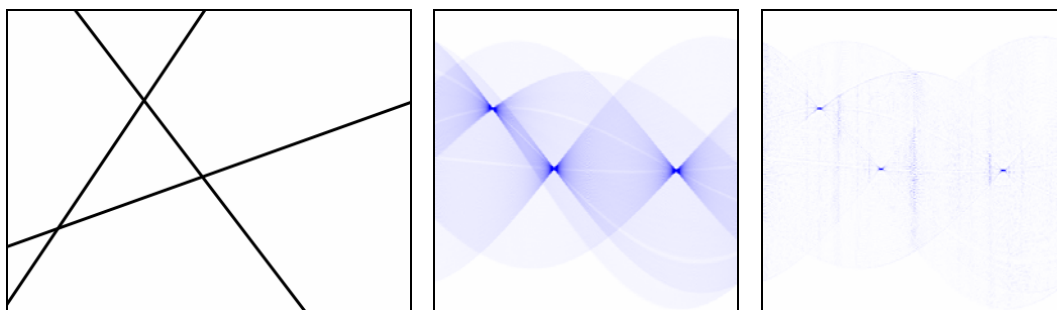
**Rysunek 3-1: Transformacja Hougha dla linii prostych – po lewej przykładowy obraz, po prawej przestrzeń Hougha z zaznaczonymi maksimami.**

Gdy kilka punktów jest współliniowych, to odpowiadające im krzywe w przestrzeni Hougha przecinają się w jednym punkcie. W ten sposób tworzą się lokalne maksima, które odpowiadają punktom

składającym się na proste na obrazie. Równanie parametryczne prostej, której dotyczy dane maksimum można odtworzyć na podstawie współrzędnych  $r$ ,  $\alpha$  w przestrzeni parametrów.

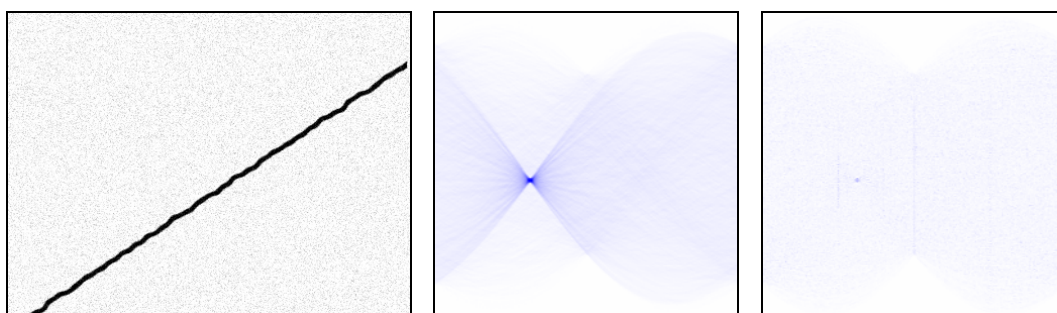
W przedstawionej implementacji przyjęto, że środek układu współrzędnych obrazu znajduje się w jego punkcie centralnym. Parametr  $r$  może przyjmować wartości dodatnie i ujemne w zależności od położenia względem początku lokalnego układu współrzędnych.

Występowanie szerokich odcinków na obrazie (o szerokości większej niż jeden piksel) powoduje utworzenie „płaskich” obszarów w przestrzeni Hougha o zbliżonych wartościach. W takim przypadku trudne jest wyznaczenie pojedynczego maksimum reprezentującego prostą na obrazie. Zadanie staje się jeszcze bardziej skomplikowane, gdy celem jest odnalezienie odcinków, które mają różną długość – odcinki krótsze powodują, że wartość lokalnego maksimum, które im odpowiada jest mniejsza niż wartości pozostałych maksimumów. Użycie filtru typu *butterfly* pozwala na uwydatnienie maksimumów tworzonych w przestrzeni Hougha dla linii prostych. Jego zastosowanie pokazano na rysunku 3-2.



**Rysunek 3-2: Przykładowy obraz, oryginalna przestrzeń Hougha oraz wynik działania filtru typu *butterfly*.**

Istotną cechą omawianej transformacji jest możliwość zastosowania jej do obrazów zaszumionych i częściowo zniekształconych. Na rysunku 3-3 pokazany jest przykład, w którym odcinek jest szerszy niż pojedynczy piksel, ma zniekształcony przebieg, a obraz został dodatkowo zakłóconym szumem o rozkładzie normalnym.



**Rysunek 3-3: Przykładowy obraz oraz ze zniekształconą linią, zaszumiony szumem  $\sigma_n = 20$ ; przestrzeń Hougha oryginalna oraz po filtracji typu *butterfly*.**

Transformacja Hougha w takim przypadku nadal pozwala na detekcję poszukiwanego kształtu. Zniekształcenie odcinka powoduje mniej regularny obraz akumulatora. Szum występujący na obrazie wprowadza występowanie nieregularnego tła w odpowiadającej mu przestrzeni Hougha. Pomimo tego detekcja maksimum akumulatora nadal jest możliwa, nawet bez zastosowania filtracji *butterfly*. Ze względu na to, że każdy punkt obrazu jest źródłem krzywej w przestrzeni Hougha niezależnie od pozostałych,

transformacja ta jest odporna do pewnego stopnia na zakłócenia i zniekształcenia. Cecha ta będzie bardzo istotna w konstrukcji proponowanego sprzężenia procesu dyfuzji z przestrzenią Hougha. Mała podatność transformacji na szum pozwala na poprawienie szybkości i jakości filtracji opartej na procesie dyfuzyjnym.

### 3.3 Propozycja sprzężenia procesu dyfuzji z przestrzenią Hougha

Wprowadzenie regularyzacji do procesu dyfuzji odbywa się zazwyczaj przez bezpośrednią modyfikację równania różniczkowego lub estymatora krawędzi (np. dodatkowa faza przekształcenia izotropowego). W proponowanej metodzie wprowadzona zostanie dodatkowa informacja do procesu przez użycie estymatora krawędzi opartego na poszukiwaniu globalnych struktur przy pomocy przestrzeni Hougha. Krótka analiza działania tego przekształcenia pokazuje, że jest ono mało wrażliwe na zakłócenia występujące na obrazie. Za pomocą transformacji Hougha możliwe jest rozpoznawanie kształtów nawet w przypadku informacji cząstkowych lub zniekształconych. Właściwość ta może być pomocna dla estymacji krawędzi w procesie odszumiania.

Istotą proponowanego sprzężenia jest możliwość określenia cech, które powinny zostać uwydatnione na obrazie. Cechy te mogą być opisane za pomocą krzywych analitycznych (standardowa transformacja Hougha) lub też dowolnych kształtów (uogólniona transformacja Hougha). Istotna w tym przypadku jest możliwość częściowa rekonstrukcja obrazu na podstawie przestrzeni Hougha. Dodatkowy czynnik filtrujący opiera się na częściowości rekonstrukcji realizowanej przez transformację odwrotną. Kształty, które okazują się zbyt mało uwydatnione (zbyt mało istotne z punktu widzenia znaczeniowego analizy obrazu) są ignorowane przy rekonstrukcji przez zastosowanie czynnika progującego  $\psi_H$ . Dodatkowo rekonstrukcja kształtu może bazować na wartościach zgromadzonych w akumulatorze tak, aby jasność odtworzonej struktury była proporcjonalna do odpowiadającej jej wartości w przestrzeni Hougha.

Alternatywnym podejściem jest określenie licznosci  $\eta_H$  struktur geometrycznych, które powinny się znaleźć na przetwarzanym obrazie. W takim przypadku odwrotna transformacja Hougha pozwala na odwzorowanie dokładnie  $\eta_H$  struktur. Podejście to pozwala na określenie dodatkowej wiedzy początkowej, która określa nie tylko rodzaj struktury (przez zastosowanie odpowiedniej wersji transformacji Hougha), ale też jej częstość występowania.

Ogólny schemat proponowanego podejścia pokazany został na rysunku 3-4.

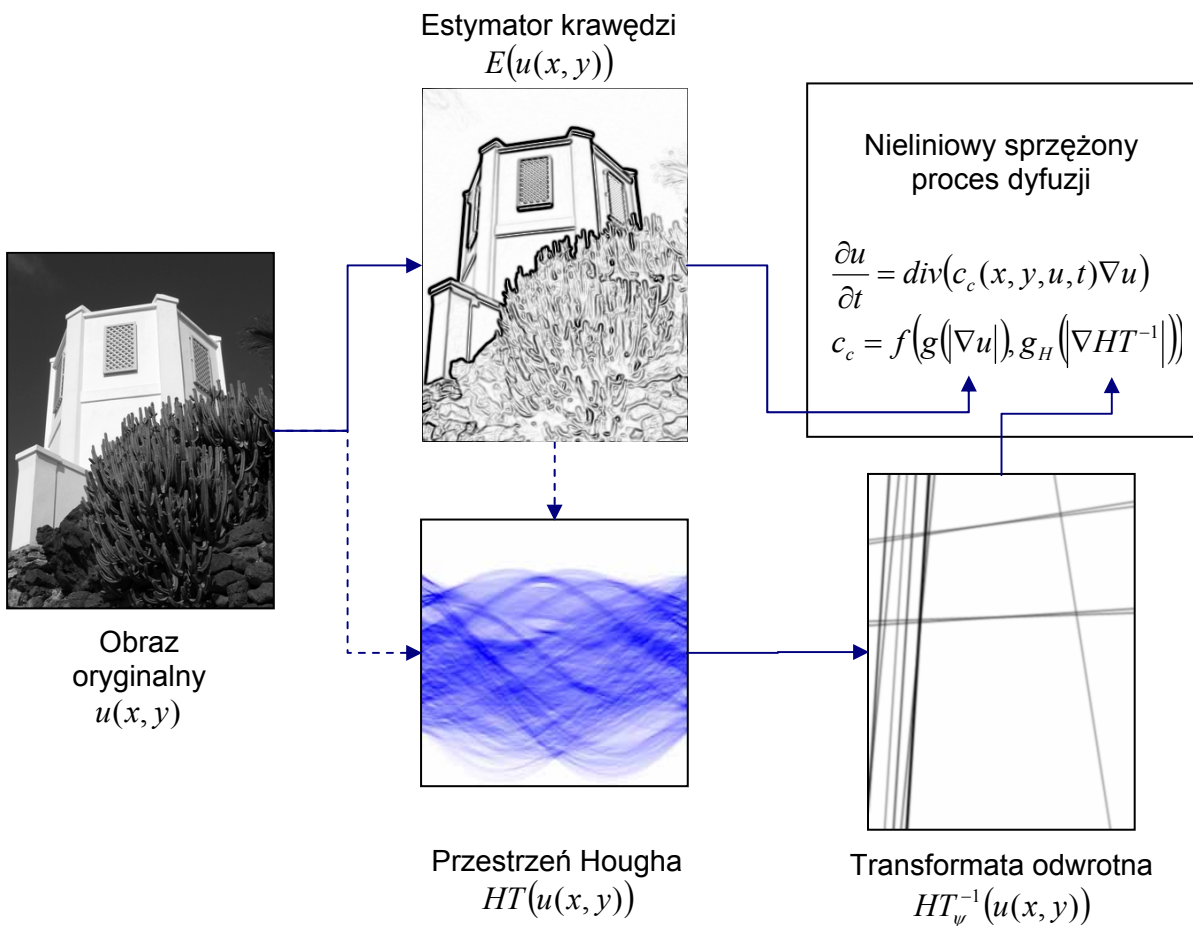
#### Oznaczenia

W dalszej części prezentowanego opisu proponowanej metody przyjmuje się następujące oznaczenia:

- $u(x, y)$  – funkcja reprezentująca obraz określona na obszarze  $\Omega$  - dla uproszczenia rozważany jest obraz w skali szarości,
- $E(u(x, y))$  – funkcja określona na obszarze  $\Omega$  uzyskana w wyniku operacji detekcji krawędzi na obrazie określonym przez  $u(x, y)$  – metoda detekcji krawędzi nie jest narzucona,



- $HT(u(x, y))$  – transformata Hougha obrazu  $u(x, y)$ ,
- $HT(u(x, y)) * M_B$  – splot przestrzeni Hougha z filtrem przestrzennym o jądrze  $M_B$  – dla transformaty linii prostych jest to filtr *butterfly* realizowany przez maskę (3 – 23),
- $HT_{\psi}^{-1}(u(x, y))$  – odwrotna transformata Hougha realizująca rekonstrukcję obrazu przy użyciu krzywych analitycznych (w ogólności kształtów) o parametrach, dla których funkcja  $H$  przyjmuje wartości większe od wartości progowej  $\psi_H$ ; odwrotna transformata Hougha jest funkcją zmiennych  $(x, y)$  określoną na obszarze  $\Omega$ ,
- $HT_{\eta}^{-1}(u(x, y))$  – odwrotna transformata Hougha realizująca rekonstrukcję obrazu przy użyciu dokładnie  $\eta_H$  krzywych analitycznych (w ogólności kształtów) określonych przez  $\eta_H$  maksimum w przestrzeni parametrów Hougha,
- $K_{\sigma} * HT_{\psi}^{-1}$  – splot odwrotnej transformaty Hougha z jądrem Gaussa o odchyleniu standardowym  $\sigma$ .



Rysunek 3-4: Ogólny schemat sprzężenia procesu dyfuzji z przestrzenią Hougha.

### 3.3.1 Nieliniowy sprzężony proces dyfuzji

Brany będzie pod uwagę proces dyfuzji opisany równaniem różniczkowym:

$$\frac{\partial u}{\partial t} = \operatorname{div}(c_c(x, y, u, t)\nabla u) \quad (3 - 24)$$

Współczynnik dyfuzji  $c_c$  jest sprzężonym estymatorem krawędzi występujących na obrazie. Proponuje się dwa odmienne podejścia definiujące postać tego współczynnika, które zostaną po kolei omówione. Postać sprzężonego współczynnika  $c_c$  zależy może od konkretnego zastosowania (także od detekowanych struktur tworzących krawędzie na obrazie).

#### Sprzężenie addytywne

$$c_1 = g\left(\left(1 - h_f\right) \cdot \|\nabla u(x, y)\| + h_f \cdot \left\|\nabla\left(K_\sigma * HT_\psi^{-1}\right)\right\|\right) \quad (3 - 25)$$

Rozpatrywane są dwie składowe współczynnika postaci  $c_1$ , których suma podawana jest jako argument funkcji  $g$ . Na funkcję  $g$  nałożone są te same wymagania, jak w przypadku modelu *Perona-Malik* (patrz równania (2 - 51), (2 - 52) w rozdziale 2.5.1). Także dyskretyzacja równania (3 - 26) jest realizowana analogicznie do postaci (2 - 63).

Pierwsza składowa współczynnika  $\|\nabla u(x, y)\|$  odpowiada standardowemu estymatorowi krawędzi opartemu na gradiencie funkcji  $u(x, y)$ . Druga składowa jest estymatorem struktury geometrycznej wyznaczanym na podstawie odwrotnej transformaty Hougha w postaci  $HT_\psi^{-1}$  lub  $HT_\eta^{-1}$ . Transformata ta jest poddawana dodatkowo splotowi z jądrem Gaussa aby proces był bardziej stabilny. Jest to istotne również z punktu widzenia dokładności wyznaczania transformaty i odwrotnej transformaty. W zależności od kwantyzacji w przestrzeni parametrów może się zdarzyć, że odwrotna transformata nie pokrywa się z oryginalną strukturą znajdującą się na obrazie. Zastosowanie splotu z funkcją Gaussa regularyzuje proces i pozwala na estymację krawędzi odtworzonej struktury, nawet jeżeli nie pokrywa się ona całkowicie z krawędzią występującą na obrazie, a jest umiejscowiona w otoczeniu oryginału.

Istotną rolę odgrywa współczynnik  $h_f$ , który decyduje o stosunku, w jakim określane jest całkowite oszacowanie stopnia występowania krawędzi i przekazywane do funkcji  $g$  na podstawie dwóch źródeł informacji. Współczynnik ten może przyjmować wartości w zakresie  $[0,1]$ . Dla  $h_f = 0$  filtracja jest realizowana według klasycznego modelu *Perona-Malik*. Natomiast dla  $h_f = 1$  jedynym źródłem dla estymacji krawędzi jest odwrotna transformata Hougha. W ogólności współczynnik  $h_f$  nie musi pozostawać stały. W dalszej części pracy zostanie pokazane, w jaki sposób może on być dynamicznie określany na podstawie oszacowania zawartości szumu na obrazie.

Wprowadzenie ograniczenia  $h_f \in [0,1]$  zapewnia, że również współczynnik  $c_1 \in [0,1]$ , co jest warunkiem koniecznym do zachowania stabilności procesu. Dzięki temu podstawowe własności filtracji przenoszą się na tak skonstruowany model.

Użycie współczynnika progowego  $\psi_H$  pozwala na określenie, które kształty powinny być brane pod uwagę. Na wartość tego parametru powinna wpływać przede wszystkim zawartość obrazu. Jest to dodatkowy parametr (poza samą metodą), który może posłużyć do wprowadzenia wiedzy apriorycznej – w zależności od intensywności i liczności występowania wybranych kształtów. W ten sposób otrzymuje się dodatkową możliwość filtracji tylko tych poszukiwanych cech, które są najbardziej istotne na obrazie. Współczynnik ten również może być określany adaptacyjnie na podstawie zawartości szumu na obrazie. Alternatywnie można stosować wersję transformaty odwrotnej, która wyznacza  $\eta_H$  najistotniejszych struktur znalezionych na obrazie.

Ten rodzaj sprzężenia jest przydatny w sytuacji, gdy znane jest z dużą dokładnością położenie krawędzi uzyskanych na podstawie oszacowania opartego na  $HT^{-1}$ . Gdy jedna ze składowych współczynnika rośnie – dyfuzja jest w danym rejonie hamowana. Jeżeli w obszarze została znaleziona struktura (określona w przestrzeni Hougha), to dyfuzja będzie zahamowana nawet, jeżeli krawędzi struktury są lokalnie zniekształcone przez szum. Jeżeli natomiast struktura nie została zdetekowana, to proces będzie realizowany według standardowego modelu, w którym główną rolę estymacji odgrywa gradient obrazu.

Użycie tej postaci sprzężenia pozwala na łagodniejszą filtrację w przypadku braku detekcji kształtów przez przekształcenie  $HT_{\psi}^{-1}$  (na przykład przy znacznym zaszumieniu obrazu). Jednak wymagane jest, aby odwrotna transformata chociaż w przybliżony sposób odwzorowywała położenie krawędzi na obrazie. Przykładowo dla transformaty Hougha użytej do poszukiwania okręgów o równaniu  $(x - x_s)^2 + (y - y_s)^2 - r^2 = 0$  proces pozwala na dokładną rekonstrukcję znalezionych kształtów. Transformacja do poszukiwaniu linii prostych (równanie (3–19)) nie pozwala wprost na dokładną rekonstrukcję krawędzi, jedynie prostych, w których krawędzie się zawierają (z dokładnością do kroku kwantyzacji). Rozwinięciem proponowanej metody mogłoby być uwzględnienie algorytmów odwrotnego przekształcenia Hougha przedstawionych w [52].

Aby użyć addytywnej postaci sprzężenia procesu z transformacją dla linii prostych, konieczne jest zastosowanie adaptacyjnego sposobu określania współczynnika  $h_f$ . W przypadku stałej wartości współczynnika lepsze efekty uzyskuje się przez sprzężenie multiplikatywne (zminimalizowany jest czynnik błędnej estymacji krawędzi, gdy znaleziony jest jedynie ich obraz w przestrzeni odwrotnej transformaty).

### Sprzężenie multiplikatywno-addytywne

$$c_2 = \frac{g(\|\nabla u(x, y)\|) \cdot (h_f g_H (\|\nabla(K_{\sigma} * HT_{\psi}^{-1})\|) + h_s)}{1 + h_s} \quad (3 - 26)$$

Druga postać sprzężonego współczynnika dyfuzji wykazuje inne właściwości. Konstrukcja jest analogiczna do podejścia opisanego równaniem (3 – 7).

Funkcja  $g_H$  w równaniu (3 – 26) może przyjmować tą samą postać, co funkcja  $g$ . Pełni ona rolę funkcji stopującej dla estymatora struktury geometrycznej na podstawie odwrotnej transformaty Hougha. Przy tej postaci sprzężenia możliwe jest użycie niezależnej postaci i parametrów dla funkcji  $g$  oraz  $g_H$ . Ze

względnie jednak na stosunkowo regularną strukturę przestrzeni  $HT^{-1}$  postać funkcji  $g_H$  nie wpływa znacząco na właściwości całego procesu.

Na współczynnik  $c_2$  składa się odpowiednio przeskalowany iloczyn dwóch niezależnych funkcji stopujących operujących na różnych estymatorach krawędzi. Analogicznie jak poprzednio pierwszy czynnik iloczynu oparty jest na standardowym estymatorze realizowanym przez gradient obrazu. Drugi czynnik realizuje funkcję stopującą w zależności od estymacji występowania struktury na podstawie odwrotnej transformaty Hougha (przy parametrze  $\psi_H$  lub  $\eta_H$  w zależności od wersji). Również w tym przypadku współczynnik  $h_f \in [0,1]$ . Wprowadza się dodatkowy współczynnik skalujący  $h_s$ , który pozwala na określenie, w jakim stopniu sprzężenie ma mieć charakter addytywny lub multiplikatywny. Dla  $h_s = 0$  sprzężenie ma charakter czysto multiplikatywny, co najczęściej nie ma praktycznego zastosowania. Dla  $h_s \rightarrow \infty$  proces jest coraz bliższy modelowi *Perona-Malik*, gdyż dominuje w nim składowa oszacowania występowania krawędzi opartego na gradientie obrazu oryginalnego.

Addytywna część wyrażenia (3 – 26) powoduje, że częściowo realizowana jest filtracja niezależna od estymacji struktury w przestrzeni Hougha. Jeżeli jednocześnie oszacowanie występowania struktury wykazuje brak krawędzi (wartość funkcji  $g_H$  równa w przybliżeniu 1), to dyfuzja na tym obszarze zostaje przyspieszona. Jeżeli natomiast struktura ta jest odnaleziona, to wartość całkowita współczynnika  $c_2$  zostaje dodatkowo zmniejszona, co powoduje szybsze zahamowanie dyfuzji.

Szczególnym przypadkiem jest sytuacja, gdy oszacowanie występowania krawędzi obrazu wykazuje ich brak (wartość funkcja  $g$  równa w przybliżeniu 1), a oszacowanie struktury na podstawie przestrzeni Hougha wskazuje na jej występowanie (wartość funkcji  $g_H$  zmierza do 0). Jest to możliwe w przypadku detekcji linii prostych, dla których podstawowy algorytm Hougha nie określa położenia linii, jedynie proste, w jakich się zawierają. Jednak ponieważ całkowity proces filtracji nie jest czysto multiplikatywny, to dyfuzja w takim obszarze nadal jest prowadzona pomimo hamującego wpływu funkcji  $g_H$ .

Ta wersja sprzężenia wykazuje dobre właściwości zarówno w przypadku krzywych, których położenie można dokładnie odwzorować (np. okrąg) jak też i w przypadku transformacji dla linii prostych. Tak przygotowany proces często pozwala na przeprowadzenie filtracji przy stałej wartości współczynnika  $h_f$ . Wartość współczynnika  $h_s$  można przyjąć w zależności od rodzaju struktur występujących na odszumianym obrazie. Dla struktur zamkniętych (np. okręgi) można zwiększyć addytywną składową procesu przez przyjęcie dużej wartości  $h_s$ . Dla struktur zawartych w ogólnych krzywych analitycznych (np. linie i proste) można w większym stopniu uwzględnić multiplikatywny charakter sprzężenia. Prawidłowe działanie procesu będzie jednak często zależeć od poziomu i natężenia sygnału szumu, który może powodować błędną detekcję krawędzi i struktur na obrazie.

### 3.3.2 Adaptacyjne wyznaczanie współczynnika sprzężenia $h_f$

Proponuje się, aby wartość współczynnika  $h_f$  zależała od zaszumienia występującego na obrazie. Podejście to powinno uwzględnić następujące fazy o zróżnicowanym charakterze filtracji:

1. Dla bardzo dużego zaszumienia możliwe jest, że detekcja struktur występujących na obrazie da fałszywe wyniki (pomimo tego, że transformacja Hougha jest w pewnym stopniu odporna na zakłócenia). Początkowo błędnie określona struktura może mieć bardzo negatywny wpływ na cały proces filtracji. W takim przypadku estymacja struktury dla odwrotnej transformaty Hougha powinna być brana z mniejszym współczynnikiem, ze względu na możliwość błędnego wyniku. Pierwsza faza filtracji powinna przypominać klasyczny model *Perona-Malik* –  $h_f$  przyjmuje małe wartości.
2. Gdy obraz zostanie częściowo odszumiony, prawdopodobieństwo prawidłowej identyfikacji struktury przy pomocy transformacji Hougha rośnie. Proponuje się, aby w drugiej fazie procesu oszacowanie występowania krawędzi wyznaczane było w znacznym stopniu na podstawie informacji o strukturach geometrycznych obliczonych przez odwrotną transformację Hougha –  $h_f$  przyjmuje duże wartości zbliżone lub równe 1.
3. W zależności od kwantyzacji przestrzeni parametrów Hougha identyfikacja struktur na obrazie może być zrealizowana ze skończoną dokładnością. Częściowym rozwiązaniem tego problemu jest użycie splotu transformaty odwrotnej z jądrem Gaussa, co powoduje, że krawędzie struktury są estymowane nie w pojedynczym punkcie, ale na pewnym obszarze wokół niego. Proponuje się, aby w adaptacyjnej wersji algorytmu w trakcie ostatniej fazy odszumiania (gdy większość szumu została już usunięta) proces ponownie miał charakter oparty na modelu *Perona-Malik*, a jedynie w niewielkiej części używał informacji z przestrzeni Hougha. Podejście takie jest szczególnie przydatne w przypadku transformacji dla linii prostych, która może powodować identyfikację struktury (linii) w miejscu, w którym taka nie występuje. Zmniejszenie współczynnika  $h_f$  w ostatniej fazie algorytmu powoduje, że zostają wygładzone obszary, dla których taka błędna identyfikacja mogła zajść.

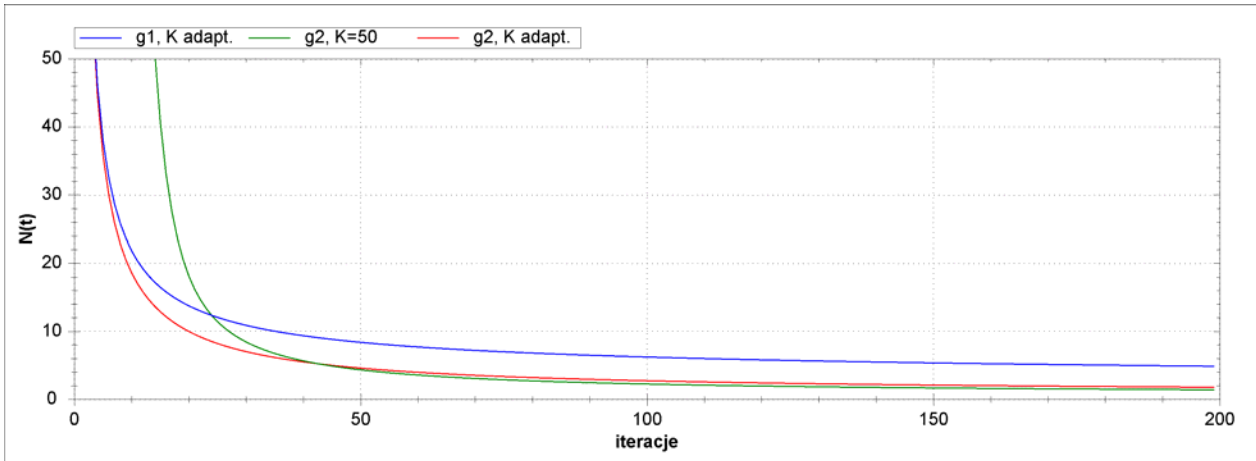
Proponuje się, aby uzależnić wartość współczynnika sprzężenia od oszacowania globalnego zaszumienia obrazu, podobnie jak to było realizowane dla współczynnika  $K$  opisanym równaniem (2 – 57) w nieliniowym procesie dyfuzji. Funkcja estymacji poziomu zaszumienia przyjmuje postać:

$$N(t) = \frac{1}{W(I)H(I)} \sum_{x=0}^{W(I)-1} \sum_{y=0}^{H(I)-1} |\nabla u(x, y, t)|, \quad (3 - 27)$$

gdzie  $W(I), H(I)$  oznaczają odpowiednio szerokość i wysokość obrazu w pikselach. Funkcja ta w praktyce jest monotonicznie malejąca dla kolejnych iteracji  $t$  procesu filtracji. Przykładowe przebiegi  $N(t)$  dla filtracji realizującej model *Perona-Malik* pokazane zostały na rysunku 3-5. Dla tego przykładu filtracji został poddany syntetyczny obraz testowy o zaszumieniu  $\sigma_n = 400$  pokazany na rysunku 2-9. Użyto zarówno stałej wartości  $K=50$  oraz adaptacyjnego doboru współczynnika  $K$  z parametrem skalującym  $\mathcal{G}=0.9$ , funkcji stopujących  $g_1, g_2$  oraz parametru  $\lambda=0.25$ . Oszacowanie  $N(t)$  dla początkowych iteracji wynosiło około 400 (wykres zawężono dla lepszej czytelności w kolejnych iteracjach).

Przebieg funkcji  $N(t)$  dla filtracji przy innym modelu (także sprzężonym) będzie oczywiście odmienny. Jednak istotne jest to, że funkcja ta w praktyce jest monotonicznie malejąca – szybciej w

początkowym etapie przetwarzania, wolniej po usunięciu znacznej części zaszumienia. Monotoniczny charakter tego współczynnika wynika z własności procesu filtracji, który nie wprowadza żadnych nowych maksimum, a jedynie może powodować zmniejszenie różnicy wartości pomiędzy sąsiadującymi pikselami. Dzięki temu zagwarantowane jest również to, że w kolejnych krokach wartość bezwzględna gradientu nie będzie rosła.



**Rysunek 3-5: Oszacowanie zaszumienia obrazu  $N(t)$  dla kolejnych kroków filtracji przy użyciu modelu *Perona-Malik* (obraz testowy nr 1, początkowe zaszumienie  $\sigma_n = 400$ ).**

Różne przebiegi otrzymane na rysunku 3-5 wynikają z faktu użycia tego estymatora, jako czynnika, od którego zależy parametr  $K$  (dla wersji adaptacyjnej). Widoczne jest, że przy stałym ustalonym  $K$  wartość funkcji  $N(t)$  maleje wolniej (dla ustalonego  $K$  przy użyciu funkcji  $g_1$  wartość ta nie spadła poniżej 300). Charakterystyka odszumiania z punktu widzenia oszacowania zawartości szumu powinna być zbliżona do modelu oryginalnego przy użyciu modelu sprzężonego. Wynika to z realizacji tego samego procesu dyfuzyjnego, na którym oparty jest oryginalny model – modyfikacji podlega jedynie sposób estymacji krawędzi obiektów. Własności procesu (takie jak nie wprowadzanie nowych ekstremów) będą zachowane.

Proponuje się, aby adaptacyjny sposób wyznaczania współczynnika sprzężenia realizował funkcję zbliżoną do funkcji Gaussa o niezerowej wartości oczekiwanej:

$$h_f(t) = h_{\max} \exp\left(\frac{-(N(t) - h_\mu)^2}{h_\sigma}\right), \quad (3 - 28)$$

gdzie  $h_{\max}$  jest współczynnikiem skalującym, który określa maksymalną wartość, jaką może przyjąć  $h_f$ ;  $h_\mu$  jest wartością średnią zaszumienia, dla której współczynnik sprzężenia przyjmie wartość maksymalną;  $h_\sigma$  jest odpowiednikiem odchylenia standardowego funkcji Gaussa i decyduje o przebiegu funkcji współczynnika  $h_f$  (dla małych wartości przebieg jest zawężony, dla wartości większych współczynnik wolniej narasta i wolniej zostaje wytłumiony).

W podrozdziale 3.4 zostanie zaprezentowany przykładowe wykresy wartości współczynnika sprzężenia  $h_f$  dla kolejnych iteracji.

### 3.3.3 Algorytm filtracji z uwzględnieniem wiedzy o strukturze geometrycznej

Zostanie przedstawiony i zbadany algorytm wykorzystujący nieliniowy proces dyfuzji uwzględniający dodatkową informację o filtrowanym obrazie. Istotne dla algorytmu jest określenie, z jaką częstotliwością należy wykonywać transformację Hougha. Jakkolwiek zawartość znaczeniowa obrazu się zmienia, to przestrzeń Hougha nie musi być aktualizowana w każdym kroku. Dlatego proponuje się wykonywać transformację Hougha co określoną liczbę iteracji z okresem  $T_H$ .

#### Algorytm 2: Nieliniowy proces dyfuzji sprzężony z przestrzenią Hougha.

Parametry:  $\sigma, \sigma'$  - poziom wstępnej filtracji (odchylenie standardowe);  $\psi_H / \eta_H$  - wartość progowa/liczba struktur branych pod uwagę dla odwrotnej transformaty Hougha;  $K, K_H$  - współczynniki dla funkcji  $g, g_H$ ;  $\lambda$  - współczynnik szybkości dyfuzji;  $T_H$  - okres wyznaczania transformaty Hougha i rekonstrukcji obrazu;  $h_f$  - współczynnik sprzężenia;  $h_{\max}, h_\mu, h_\sigma$  - parametry potrzebne do adaptacyjnego określania  $h_f$ ;  $t_{\max}$  - liczba iteracji;  $s_\alpha, s_r$  - kwantyzacja akumulatora

Oznaczenia:  $u^0$  - obraz oryginalny,  $u^t$  - obraz w iteracji  $t$ ,  $t$  - numer iteracji

**krok 1:** Przypisz  $t = 0$

**krok 2:** Wyznacz  $E_\sigma = K_\sigma * E(u_t)$  - detekcja krawędzi obrazu  $u_t$  poddanemu wstępnej filtracji jądrem Gaussa

**krok 3:** Wyznacz  $HT(E_\sigma)$  - przestrzeń Hougha dla funkcji opisującej krawędzie obrazu

**krok 4:** Wyznacz  $HT_\psi^{-1}(E_\sigma)$  lub  $HT_\eta^{-1}(E_\sigma)$  - rekonstrukcja obrazu  $u_t$  na podstawie przestrzeni Hougha przy parametrze  $\psi_H$  lub  $\eta_H$

**krok 5:** Jeżeli  $h_f$  adaptacyjne wyznacz  $N(t)$  oraz  $h_f(t)$

**krok 6:** Dla każdego piksela obrazu  $u_s^t$ , gdzie  $s = (x, y)$ :

**krok 7:** wyznacz wartości różnic na obrazie  $d(u)_{NSEW} = \|\nabla_{NSEW} u_s^t\|$

**krok 8:** wyznacz wartości różnic transformaty  $d(H^{-1})_{NSEW} = \|\nabla_{NSEW} (K_\sigma * HT_\psi^{-1}(E_\sigma))\|$

**krok 9:** wyznacz wartości współczynnika  $c_c$  (w zależności od typu sprzężenia)

$$c_{1,NSEW} = g((1 - h_f) \cdot d(u)_{NSEW} + h_f \cdot d(HT^{-1})_{NSEW})$$

$$c_{2,NSEW} = \frac{g(d(u)_{NSEW}) \cdot (h_f g_H(d(HT^{-1})_{NSEW}) + h_s)}{1 + h_s}$$

**krok 10:** wyznacz nową wartość piksela

$$u_s^{t+1} = u_s^t + \frac{\lambda}{4} [c_{c,N} \cdot \nabla_N u_s^t + c_{c,S} \cdot \nabla_S u_s^t + c_{c,E} \cdot \nabla_E u_s^t + c_{c,W} \cdot \nabla_W u_s^t]$$

**krok 11:** Przypisz  $t \leftarrow t + 1$

**krok 12:** Jeżeli  $(t \bmod T_H) = 0$ , wróć do kroku 2

**krok 13:** Jeżeli  $t < t_{\max}$ , wróć do kroku 5

W przedstawionym algorytmie przyjęto dyskretyzację równania (3 – 24) jawnym schematem różnicowym opisanym w rozdziale 2.5.2. W ogólności można użyć dowolnego sposobu dyskretyzacji zaproponowanego procesu dyfuzji. Detekcja krawędzi wykonywana w kroku 2 powoduje, że operacja wyznaczania transformaty Hougha działa efektywniej – pozwala na poszukiwanie właściwych kształtów w przestrzeni krawędzi. Możliwe jest pominięcie tego kroku, jeżeli obraz poddawany obróbce zawiera już te struktury, które powinny być uwydatnione. W przedstawionej implementacji przyjęto, że struktury te reprezentowane są przez krawędzie obiektów znajdujących się na obrazie. Do detekcji krawędzi może być użyta dowolna metoda wyznaczenia  $E(u_t)$ . Jednak ze względu na używaną w kroku 7 operację realizującą dyskretyzację gradientu funkcji  $u'_s$ , można również użyć tej samej metody i wyników w obu krokach.

Splot funkcji  $u'$  z jądrem Gaussa w kroku 2 pozwala na zmniejszenie wrażliwości metody na szum występujący na obrazie. Dzięki wstępnej liniowej filtracji zmniejsza się prawdopodobieństwa błędnej detekcji kształtu przy wyznaczaniu transformaty Hougha w kroku 3. Splot rekonstrukcji obrazu  $HT_{\psi}^{-1}(E_{\sigma})$  w kroku 8 pozwala natomiast na regularyzację procesu z punktu widzenia skończonej dokładności przestrzeni Hougha (problem doboru kroku kwantyzacji).

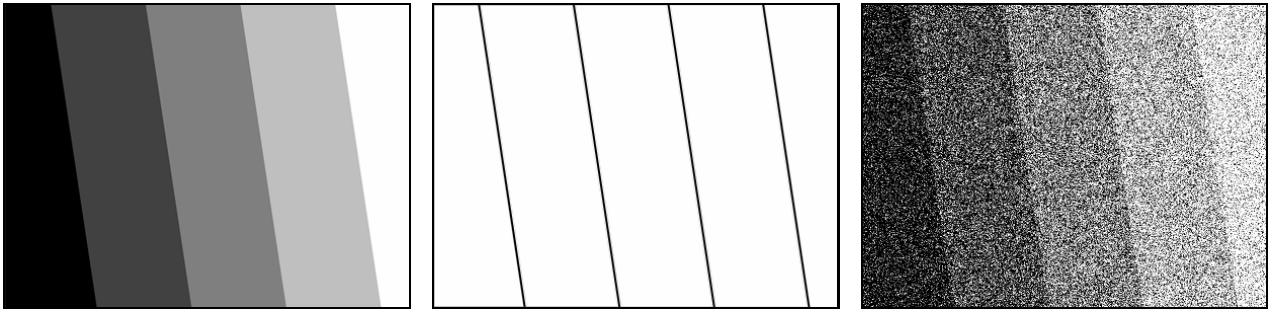
Należy zauważyć, że zaprezentowaną metodę można również rozszerzyć do sprzężonego anizotropowego procesu dyfuzji. Ze względu na wykorzystywanie informacji z dwóch przestrzeni określonych na wspólnym obszarze  $\Omega$ , można analogicznie jak to zostało pokazane zdefiniować sprzężony tensor  $D$  uwzględniających szybkość dyfuzji na podstawie obrazu oraz jego rekonstrukcji  $HT_{\psi}^{-1}(E_{\sigma})$ . Dodatkowo można również uwzględnić informację o kierunku krawędzi, np. przy realizacji transformacji Hougha dla linii prostych (informacja o współczynniku kierunkowym jest wyznaczana przy okazji obliczania transformaty).

W przeprowadzonych eksperymentach używana była standardowa transformacja Hougha dla linii prostych. Nie zmienia to ogólności wniosków dotyczących tej metody ze względu na użycie rekonstrukcji obrazu na podstawie transformaty odwrotnej. Również użycie różnych modyfikacji transformacji (szybka, hierarchiczna, adaptacyjna) może dać ciekawe wyniki. Jednak przebadanie możliwości zastosowania różnego rodzaju transformacji jest bardzo rozbudowane i wykracza poza ramy prezentowanej pracy.

### **3.4 Badanie własności algorytmu sprzężonego**

Filtracja oparta na sprzężeniu z przestrzenią Hougha zakłada, że na obrazie mają być uwydatnione określone struktury geometryczne. Rodzaj tych struktur będzie zależał od użytej transformacji Hougha. Przedstawione eksperymenty dotyczyć będą transformacji dla linii prostych, w związku z czym badania przeprowadzone będą na przygotowanych syntetycznie obrazach testowych, na których linie takie powinny zostać uwydatnione. Do tego celu używany będzie obraz testowy, którego filtracja badana była w 2.6 oraz dodatkowy obraz testowy zawierający większą liczbę krawędzi, które mogą być opisane linią prostą (rysunek 3-6).





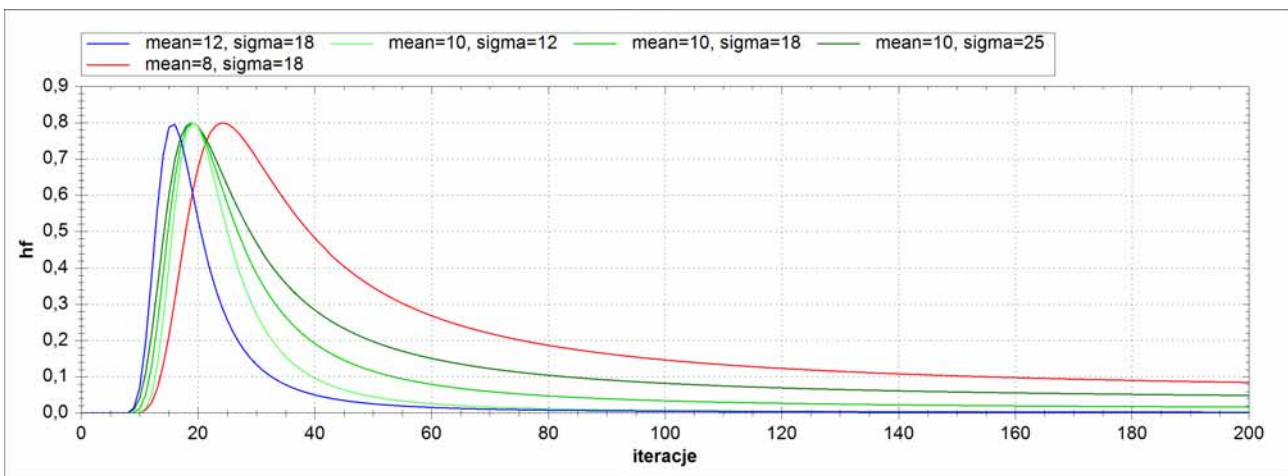
Rysunek 3-6: Syntetyczny obraz testowy nr 2 (paski); detekcja krawędzi; zaszumiony obraz testowy  $\sigma_n = 100$ .

### 3.4.1 Skuteczność filtracji dla współczynnika dyfuzji $c_1$ (sprężenie addytywne)

Użycie sprężenia addytywnego w postaci współczynnika dyfuzji wyrażonego wzorem (3 – 25) najczęściej będzie połączone z adaptacyjnym sposobem wyznaczania współczynnika sprężenia  $h_f$  (3 – 27). Taka sytuacja powinna mieć miejsce w przypadku obrazów, dla których kształty nie pokrywają się całkowicie ze strukturami opisanymi w przestrzeni Hougha (przykładem jest obraz testowy nr 1). Jeżeli natomiast kształty mogą się całkowicie pokryć z poszukiwaną krzywą, to użycie adaptacyjnego wyznaczania  $h_f$  nie jest konieczne. Zostaną zaprezentowane wyniki filtracji uzyskane przy adaptacyjnym, jak i stałym (dla obrazu nr 2) współczynniku sprężenia.

#### Adaptacyjne obliczanie współczynnika $h_f$

W zależności (3 – 27) zostało zaproponowane wyznaczenie współczynnika sprężenia od oszacowania całkowitego zaszumienia obrazu. Wykonano próbną serię filtracji, która miała na celu pokazanie przebiegu wartości współczynnika w zależności od parametrów użytych do jego obliczenia. Do obliczenia tego współczynnika potrzebne jest określenie dwóch parametrów decydujących o kształcie funkcji ( $h_\mu$  oraz  $h_\sigma$ ) oraz parametru skalującego  $h_{\max}$ . Na wykresie przedstawiono przykładową funkcję  $h_f$  dla kolejnych iteracji procesu filtracji obrazu testowego nr 1.



Rysunek 3-7: Wartość współczynnika  $h_f$  w zależności od parametrów  $h_\mu$ ,  $h_\sigma$  (oznaczonych odpowiednio *mean*, *sigma*). Obraz testowy nr 1, początkowe zaszumienie  $\sigma_n = 400$ .

Funkcja określająca współczynnik sprzężenia jest zbliżona w kształcie do rozkładu normalnego (przy zmodyfikowanym skalowaniu ze względu na konieczność zachowania wartości współczynnika w zakresie  $[0, 1]$ ). Zwiększenie odpowiednika wartości średniej zawęży przebieg  $h_f$  oraz powoduje, że wartość maksymalna  $h_{max}$  jest osiągana wcześniej. Zmniejszenie wartości  $h_\mu$  powoduje, że przebieg jest łagodniejszy – wolniej narasta oraz zdecydowanie wolniej opada (przy zachowaniu tej samej wartości  $h_\sigma$ ). Bardzo łagodny spadek wartości  $h_f$  nie jest zwykle korzystny, gdyż w tym obszarze filtracji pożądane jest, aby dodatkowo wygładzone zostały fragmenty, dla których filtracja była zahamowana przez sprzężenie z przestrzenią Hougha.

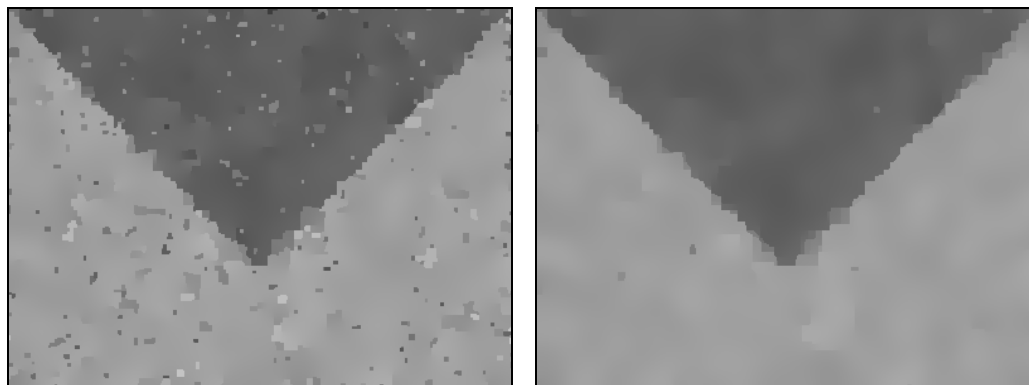
Modyfikacja odpowiednika odchylenia standardowego pozwala na sterowanie szybkością narastania oraz opadania  $h_f$  bez znaczącej zmiany momentu, dla którego osiągana jest wartość maksymalna. Stąd można wyciągnąć wniosek, że szybsze zmniejszenie zaszumienia obrazu jest realizowane przed i po okresie, gdy współczynnik  $h_f$  będzie odgrywał dominującą rolę.

Niesymetryczny przebieg tego współczynnika wynika także z charakteru filtracji realizowanej przy adaptacyjnym wyznaczaniu parametru  $K$ . Taki sposób powoduje, że w początkowej fazie filtracji, gdy całkowite zaszumienie obrazu jest duże, filtracja zbliżona jest do modelu liniowego. Dopiero potem współczynnik  $K$  maleje, co powoduje, że proces hamowania dyfuzji w okolicach krawędzi jest bardziej intensywny. Spadek zaszumienia powinien być powiązany ze wzrostem wartości współczynnika sprzężenia, co pozwoli na przyspieszenie filtracji przy jednoczesnym zmniejszeniu prawdopodobieństwa błędnego wykrycia poszukiwanych struktur na obrazie. Dalszy spadek zaszumienia jest wolniejszy niż w początkowej fazie. Dlatego zmniejszanie współczynnika  $h_f$  jest łagodniejsze. Pozwala to na usuwaniu zakłóceń, których filtracja mogła być do tej pory zahamowana ze względu na niedokładne odwzorowanie struktur w procesie odwrotnej transformacji Hougha. Współczynnik ten nie jest jednak zmniejszany do 0, dzięki czemu obszary, gdzie występują krawędzie i pokrywają się z odnalezioną strukturą na obrazie, mają znacznie mniejszą dyfuzyjność w porównaniu do pozostałych fragmentów obrazu.

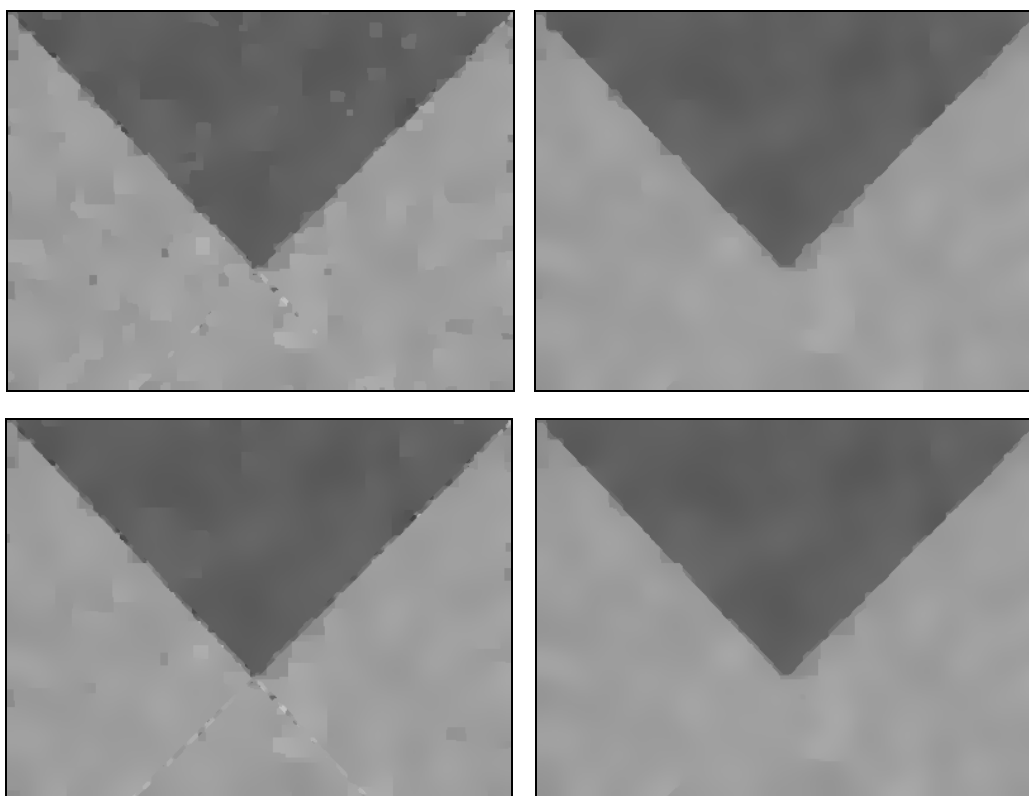
Na rysunku 3-9 zebrane zostały wyniki filtracji obrazu testowego numer 1 o zaszumieniu  $\sigma_n = 400$  po wykonaniu 300 iteracji algorytmu. W pierwszej kolumnie pokazane są wyniki filtracji przy użyciu funkcji stopującej  $g_1$ , a w drugiej kolumnie przy użyciu funkcji  $g_2$ . Wszystkie przypadki dotyczą użycia addytywnej postaci współczynnika sprzężonego  $c_1$  przy  $h_\mu = 10$ . Pierwszy rząd przedstawia wyniki dla  $h_\sigma = 10$ , drugi dla  $h_\sigma = 20$ . Na rysunku 3-8 umieszczone zostały wyniki filtracji według standardowego modelu *Perona-Malik*, w celu porównaniu wydajności filtracji z metodą bez sprzężenia. W każdym przypadku współczynnik  $K$  wyznaczany był adaptacyjnie według procedury przedstawionej w rozdziale 2.

Po zastosowaniu sprzężenia, niezależnie od użytej funkcji  $g$  widoczna jest znaczna poprawa jakości otrzymanej filtracji w porównaniu do modelu standardowego. Krawędzie obiektu przedstawionego na obrazie stają się bardziej wyraźne i regularne, co jest bezpośrednim wynikiem zastosowania modelu sprzężonego. Jednocześnie obszary jednolite są lepiej wygładzone i w każdym z przypadków lepiej usuwane są małe podobszary będące wynikiem zaszumienia obrazu początkowego. Jest to szczególnie wyraźnie widoczne przy użyciu funkcji  $g_1$ , dla której filtracja często powoduje pozostawienie obszarów o ostrych

krawędziach, które jednak nie posiadają swojego odpowiednika na obrazie oryginalnym, a są wynikiem zaszumienia. Sprzężenie z przestrzenią Hougha powoduje, że w tych rejonach, gdzie poszukiwane struktury na obrazie nie zostały określone, dyfuzja jest przeprowadzana bardziej intensywnie.



Rysunek 3-8: Wynik filtracji PM po 300 iteracjach dla obrazu testowego nr 1 (zaszumienie  $\sigma_n = 400$ ).  
Kolumna 1 –funkcja  $g_1$ , kolumna 2 –funkcja  $g_2$ .

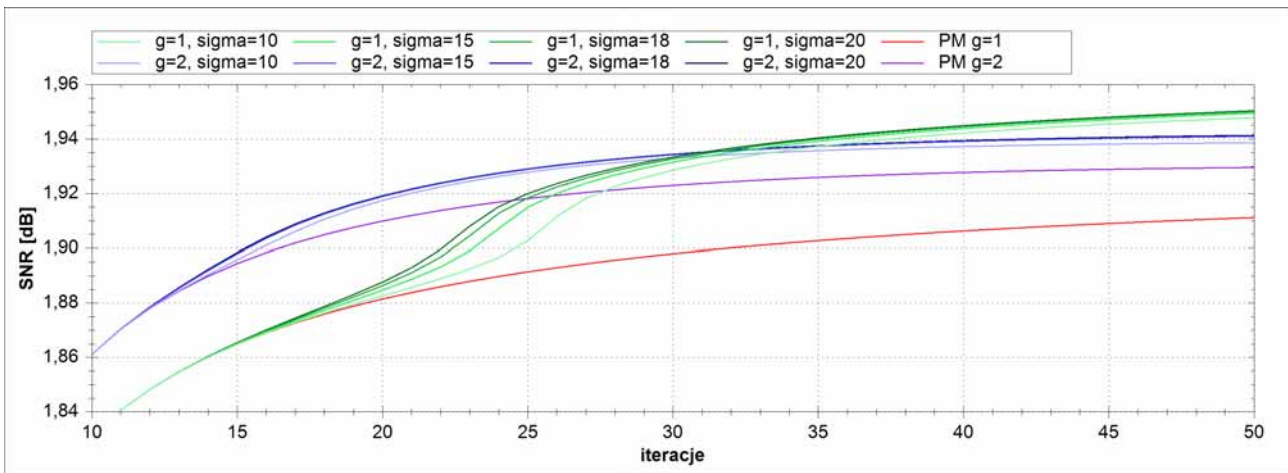


Rysunek 3-9: Wynik filtracji sprzężonej po 300 iteracjach dla obrazu testowego nr 1 (zaszumienie  $\sigma_n = 400$ ).  
Kolumna 1 –funkcja  $g_1$ , kolumna 2 –funkcja  $g_2$ . Wiersze odpowiednio:  $h_\sigma=10$ ;  $h_\sigma=20$ .

Efekt przyspieszenia dyfuzji jest w mniejszym stopniu zauważalny przy użyciu funkcji  $g_2$  ze względu na to, że użycie modelu standardowego pozwala na uzyskanie w przybliżeniu jednolitych obszarów bez krawędzi. Jednak również w tym przypadku użycie modelu sprzężonego pozwala na usunięcie pozostałości zaszumienia, a krawędzie obiektu stają się znacznie ostrzejsze i bardziej regularne w porównaniu do modelu niesprzężonego (rysunek 3-8).

W przypadku filtracji z funkcją  $g_1$  dobrze widoczny jest wpływ zmiany parametru  $h_\sigma$  do adaptacyjnego wyznaczania współczynnika sprzężenia. Użycie małej wartości  $h_\sigma$  powoduje, że informacja o strukturze geometrycznej obrazu uzyskana z przestrzeni Hougha jest brana pod uwagę tylko dla niewielkiej liczby kroków, dla których zaszumienie obrazu jest na średnim poziomie (wąski wykres na rysunku 3-7). Pozwala to na przyspieszenie dyfuzji, w obszarach, w których struktury te nie zostały odnalezione. Jednak przyspieszenie to jest krótkotrwałe, co powoduje, że pewna część zakłóceń nie zostanie usunięta.

Użycie parametru  $h_\sigma$  o zbyt dużej wartości (dolny rząd na rysunku 3-9) pozwala na szybsze usunięcie zakłóceń występujących poza zidentyfikowanymi strukturami. Jednak jednocześnie jest to powodem pojawienia się niewielkich obszarów, które należą do struktur wyznaczonych przez odwrotną transformację Hougha, a na rysunku oryginalnym nie są reprezentowane przez krawędzie obiektu. Jest to typowy przykład sytuacji, dla której sprzężenie o typie addytywnym było zbyt intensywne. Po właściwej filtracji wartość współczynnika sprzężenia powinna być zmniejszona, aby możliwe było wygładzenie struktur nienależących do właściwych krawędzi obiektów znajdujących się na obrazie. Dla dużych wartości  $h_\sigma$  końcowe wygładzenie nie jest możliwe ze względu na utrzymywanie się wartości współczynnika na podobnych poziomie przez bardzo wiele iteracji (rysunek 3-7).



**Rysunek 3-10: Jakość sygnału po filtracji według miary SNR dla różnych wartości parametru  $h_\sigma$  adaptacyjnego sprzężenia addytywnego.**

Fragmety wykresu przedstawiony na rysunku 3-10 pokazuje miarę SNR jakości obrazu po filtracji w zależności od parametru  $h_\sigma$ . Na wykresie umieszczono również miarę uzyskaną przy użyciu standardowych algorytmów bez sprzężenia. Przebiegi dla funkcji  $g_1$  wykazują charakterystyczną zmianę od 15 do około 30 iteracji. Filtracja przeprowadzana była przy użyciu adaptacyjnego wyznaczania  $h_f$ , który dla tych iteracji zaczyna być dominujący. Dzięki temu współczynnik jakości znacznie wzrasta w porównaniu do filtracji standardowej bez sprzężenia (czerwony wykres).

Przy użyciu funkcji  $g_2$  przebieg miary SNR jest odmienny. W okolicach 15 iteracji również następuje przyspieszenie filtracji, czyli szybszy wzrost miary jakości obrazu. Dalsze iteracje powodują jednak, że współczynnik stabilizuje się na poziomie ok. 1.94 dB. Również w tym przypadku otrzymane wyniki za pomocą algorytmu sprzężonego przewyższają (jednak mniej znacząco) wyniki otrzymane za pomocą filtracji standardowej (kolor fioletowy na wykresie).

Analiza przebiegów miary przy użyciu tego typu funkcji pokazuje, że proces jest w tym przypadku mniej wrażliwy na współczynnik  $h_\sigma$  (przebiegi o różnych odcieniach niebieskiego na wykresie). Jedynie dla bardzo małej wartości  $h_\sigma=10$  wykres wykazuje nieznacznie mniejszą wartość miary SNR. W pozostałych przypadkach przebiegi tej miary się pokrywają. Wynika to z charakterystyki nieliniowej filtracji używającej funkcji  $g_2$  (patrz rozdział 2.6.3). Filtracja działająca bez sprzężenia dawała rezultaty, które w przypadku obrazów syntetycznych pozwalało osiągnąć dobre rezultaty zarówno według miary SNR, jak i według oceny wizualnej. Użycie sprzężenia w tym przypadku pozwala przede wszystkim na wyostrenie krawędzi obiektu przedstawionego na rysunku – efekt przyspieszenia filtracji przez usuwanie zakłóceń poza właściwymi brzegami jest mniej widoczny w porównaniu do przypadku użycia funkcji  $g_1$ .

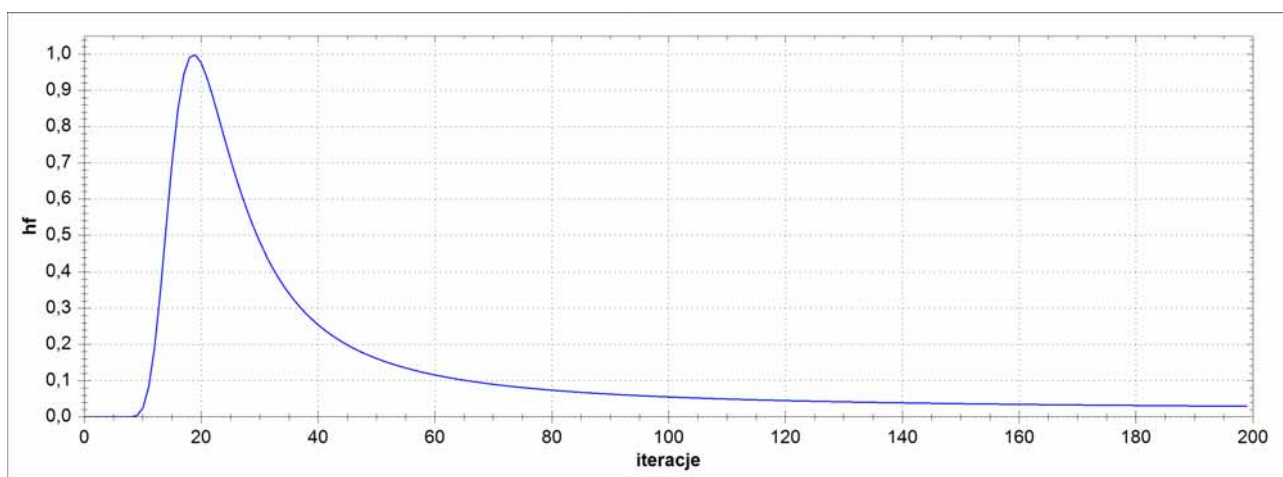
Porównując wykres z rysunku 3-10 z wynikami przedstawionymi na rysunku 3-9 należy zauważyć, że użycie funkcji  $g_2$  pozwala na uzyskanie wizualnie wyższej jakości. Według wykresu otrzymuje się jednak odmienne informacje, według których użycie funkcji  $g_1$  prowadzi do uzyskania lepszych rezultatów. Rozbieżność ta wynika przede wszystkim z tego, że przy ocenie wizualnej brane są pod uwagę przede wszystkim specyficzne cechy obrazu takie, jak krawędzie. Określenie jakości za pomocą SNR nie jest do końca miarodajne, gdyż nie uwzględnia charakterystycznych elementów, których wyodrębnienie jest celem tak przygotowanego procesu filtracji. Nie są również uwzględniane niepożądane efekty takie, jak obszary, które nie zostały odsumione ze względu na za dużą wartość sprzężenia przestrzeni Hougha, a które są znaczące przy ocenie wizualnej. Miara ta pozwala jednak na zbadanie ogólnej charakterystyki odsumiania w zależności od użytych parametrów.

### **Analiza wyników filtracji dla kolejnych iteracji algorytmu**

Na rysunku 3-12 zostały przedstawione wybrane iteracje z procesu filtracji przy pomocy nieliniowego procesu sprzężonego z przestrzenią Hougha (prawa kolumna obrazów). Dla porównania pokazano wyniki filtracji przy zastosowaniu nieliniowego procesu o tych samych parametrach bez użycia sprzężenia (lewa kolumna). W obu przypadkach przyjęto następujące parametry:  $K$  wyznaczenie adaptacyjnie, funkcja stopująca  $g_2$ ,  $\lambda=0,25$ . Dla procesu sprzężonego użyto dodatkowo następujących parametrów: sprzężenie addytywne  $c_1$ ;  $h_{max} = 1,0$ ;  $h_\sigma = 20$ ;  $h_\mu = 10$ ;  $T_H = 5$ ;  $\eta = 2$ . Na rysunku w kolejnych wierszach przedstawiono wyniki filtracji odpowiednio dla 20, 40, 100 i 200 iteracji. Wynik detekcji krawędzi typu Sobel dla tych samych iteracji znajduje się na rysunku 3-13.

Analiza przebiegu filtracji musi być powiązana z analizą wartości współczynnika sprzężenia  $h_f$  dla poszczególnych iteracji. Jego wykres znajduje się na rysunku 3-11.

Po pierwszych 20 iteracjach obserwuje się niewielką różnicę pomiędzy procesem standardowym i sprzężonym. Dotyczy ona przede wszystkim nieznacznie lepszego odsumienia obszarów poza krawędziami, co jest dobrze widoczne w pierwszym wierszu na rysunku 3-13. Podobieństwo przebiegu obu procesów wynika z prawie całkowitego zahamowania sprzężenia aż o około 10 iteracji (patrzy wykres na rysunku 3-11). Kolejne 10 iteracji powodowało stopniowe narastanie wartości współczynnika, co jest zbyt krótkim odcinkiem czasu do zaobserwowania znaczącego przyspieszenia procesu odsumiania.



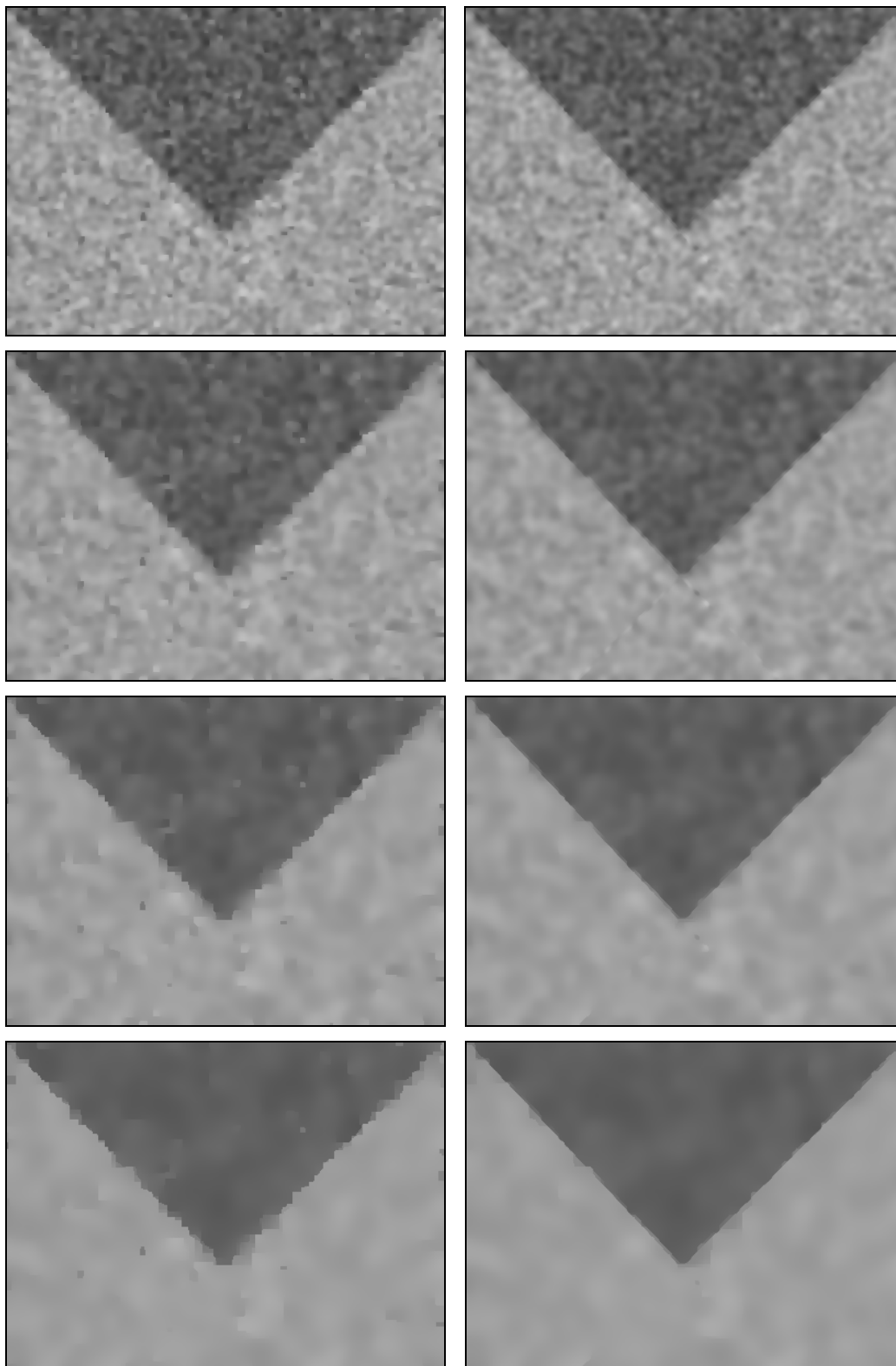
**Rysunek 3-11: Wartość współczynnika  $h_f$  dla analizowanego procesu filtracji sprzężonej. Obraz testowy nr 1, początkowe zaszumienie  $\sigma_n = 400$ .**

W ciągu kolejnych około 20 iteracji (wiersz 2 na rysunkach 3-12, 3-13) proces przebiega z dominującym czynnikiem sprzęgającym z przestrzenią Hougha. Dzięki temu zaszumienie obszarów jednolitych znacznie się zmniejsza – proces odzuszumiania jest wyraźnie szybszy niż w przypadku filtracji bez sprzężenia.

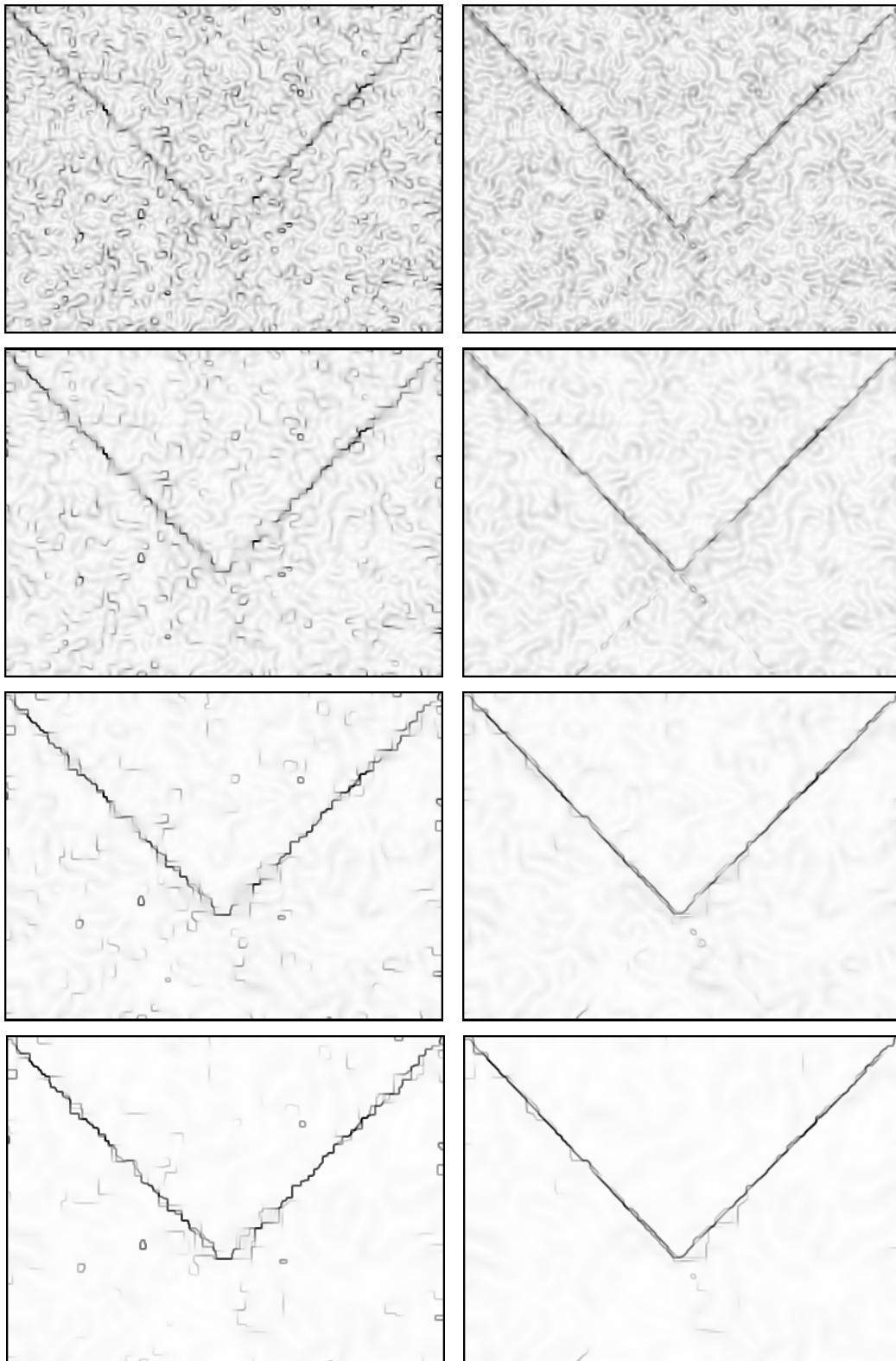
Filtracja z dominującym sprzężeniem powoduje dodatkowo, że krawędzie obiektu znajdującego się na obrazie stają się wyraźniejsze i bardziej regularne. Pojawia się również niepożądany efekt w postaci nowych krawędzi, które należą do obiektu opisane parametrami przestrzeni Hougha, a nie występują na obrazie oryginalnym. Jest to charakterystyczny efekt występujący w przypadku sprzężenia addytywnego oraz transformacji Hougha wyznaczającej krzywe analityczne zawierające (ale niepokrywające się) poszukiwaną strukturę. Efekt taki wynika ze zmniejszenia dyfuzji w obszarze, w którym krawędzie nie występują, co powoduje, że występujące tam zakłócenia formują strukturę, która wcześniej nie występowała. W praktyce żadna nowa struktura nie jest wprowadzana na obraz, lecz nieusunięte zakłócenia sprawiają wrażenie, że struktura taka powstała (w porównaniu do przefiltrowanego obszaru jednolitego). Adaptacyjna regulacja parametru sprzężenia ma również na celu eliminację tak wprowadzonych błędnych informacji.

Po 100 iteracjach wszystkie większość drobnych zakłóceń została usunięta. Jest to znaczna poprawa w porównaniu do filtracji bez sprzężenia, w przypadku którego zakłócenia te zamiast zostać usunięte, zostają uwydatnione tworząc małe jednolite podobszary o ostrych krawędziach. Zmniejszająca się wartość współczynnika  $h_f$  (poniżej 0.2) powoduje, że proces od 40 iteracji zbliża się w swoim charakterze do filtracji bez sprzężenia. Dzięki temu zostają również w znacznej części usunięte zakłócenia, które błędnie formowały strukturę po 40 iteracjach. Krawędzie obiektu na obrazie są po 100 iteracjach znacznie bardziej regularne niż przy filtracji bez sprzężenia.

Od 100 do 200 iteracji sprzężenia procesu jest bardzo niewielkie. Dzięki temu wszelkie błędne wprowadzone struktury zostają prawie całkowicie usunięte. Charakter filtracji jest więc zbliżony do modelu bez sprzężenia. Jednak uzyskiwane wyniki znacząco się różnią ze względu na wcześniej występujące sprzężenie. Dzięki sprzężeniu procesu nie zostały błędnie utworzone niewielkie obszary, które nie mogą być



**Rysunek 3-12:** Wynik nieliniowej filtracji standardowej (kolumna lewa) oraz sprzężonej (kolumna prawa) po odpowiednio 20, 40, 100, 200 iteracjach dla obrazu testowego nr 1 (zazumienie  $\sigma_n = 400$ ).



**Rysunek 3-13: Detekcja krawędzi typu Sobel dla rezultatów nieliniowej filtracji standardowej (kolumna lewa) oraz sprzężonej (kolumna prawa) po odpowiednio 20, 40, 100, 200 iteracjach. Obraz testowy nr 1 (zaszumienie  $\sigma_n = 400$ ).**



usunięte ze względu na zbyt ostre krawędzie. Krawędzie obiektu są znacznie wyraźniejsze i bardziej regularne. Obszary, na których krawędzie nie występują są w tym przypadku znacznie bardziej jednolite.

Podsumowując przebieg procesu filtracji z adaptacyjnym sprzężeniem z przestrzenią Hougha można wyróżnić kilka podstawowych cech:

- w początkowych iteracjach proces sprzężony i niesprzężony przebiegają podobnie, aby zmniejszyć wpływ błędnej detekcji struktury na bardzo zaszumionym obrazie,
- dominujący wpływ sprzężenia powoduje, że obraz jest znacznie szybciej odszumiany,
- nieprawidłowo określone struktury (ze względu na zastosowany rodzaj transformacji) zostają prawie całkowicie usunięte dzięki zastosowaniu współczynnika adaptacyjnego,
- uzyskuje się znacznie wyraźniejsze i bardziej regularne krawędzie w porównaniu do procesu niesprzężonego.

Podstawową własność algorytmu sprzężonego można sformułować następująco (która wynika z zachowania cech, które posiada nieliniowy proces filtracji):

### **Własność 3.1**

Filtracja algorytmem realizującym sprzężenie z przestrzenią Hougha nie wprowadza nowych wartości ekstremalnych w przestrzeni funkcji  $u(x, y)$ . Oznacza to, że kolejne iteracje filtracji dają w wyniku coraz bardziej uproszczoną wersję obrazu oryginalnego  $u(x, y)$  niezależnie od stopnia sprzężenia z przestrzenią Hougha.

W trakcie przeprowadzonych eksperymentów zaobserwowano, że w efekcie zastosowania sprzężenia otrzymuje się proces przyspieszony, ale o innej charakterystyce niż by to wynikało z prostego zwiększenia wartości współczynnika dyfuzji. Na tej podstawie można sformułować następującą własność.

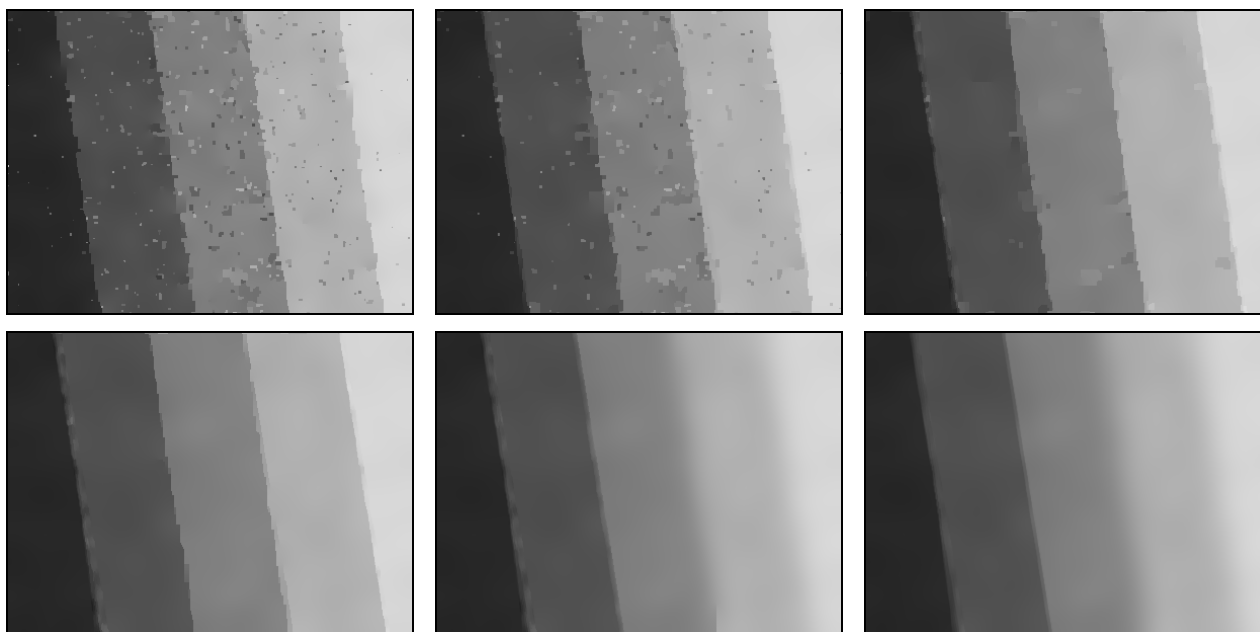
### **Własność 3.2**

Algorytm filtracji realizujący sprzężenie z przestrzenią Hougha powoduje szybsze usuwanie zakłóceń dzięki uwzględnianiu dwóch źródeł wiedzy o strukturze obrazu: estymatora krawędzi oraz filtrujących własności przestrzeni Hougha.

### **Filtracja przy stałym współczynniku $h_f$**

W przypadku sprzężenia addytywnego, gdy krawędzie na obrazie tworzą struktury, które całkowicie mogą być opisane przez parametry przestrzeni Hougha, nie ma konieczności stosowania adaptacyjnego sposobu wyznaczania współczynnika sprzężenia w celu usunięcia niepożądanych dodatkowych struktur. Przykładem takiego obrazu jest syntetyczny obraz testowy nr 2.

Na rysunku 3-14 przedstawiono wynik filtracji dla 300 iteracji w przypadku procesu bez sprzężenia (w celu porównania rezultatów) oraz dla różnych wartości współczynnika  $h_f$ . W procesie filtracji użyto funkcji  $g_r$ , parametrów  $\lambda=0.25$  oraz adaptacyjnie wyznaczaną wartość  $K$ . W procesie sprzężonym użyto parametrów:  $T_H = 5$ ,  $\eta = 4$ .



**Rysunek 3-14: Wynik nieliniowej filtracji po 300 iteracjach,  $g_1$ . Pierwszy wiersz: filtracja bez sprzężenia, sprzężenie  $h_f=0.1$ , sprzężenie  $h_f=0.3$ . Drugi wiersz: sprzężenie  $h_f=0.5$ , sprzężenie  $h_f=0.7$ , sprzężenie  $h_f=0.9$ .**

Analizując otrzymane wyniki filtracji można dojść do następujących wniosków:

- mały współczynnik sprzężenia  $h_f=0.1$  (pierwszy rząd, druga kolumna) nie wpływa znacząco na przebieg procesu filtracji – jego zastosowanie spowodowało jedynie nieznaczne przyspieszenie filtracji i usunięcie niektórych zakłóceń,
- zwiększenie współczynnika  $h_f$  do wartości 0.3 pozwala uzyskać znacznie lepsze rezultaty z punktu widzenia szybkości odszumiania – po 300 iteracjach znacznie więcej zakłóceń zostało usuniętych w porównaniu do wyniku filtracji bez sprzężenia,
- dalsze zwiększanie współczynnika powoduje, że struktury, które zostały prawidłowo określone przez transformację Hougha są ostrzejsze i znacznie bardziej regularne; jednak w przypadku, gdy struktura taka nie została prawidłowo rozpoznana (ze względu na duży stopień zakłóceń na obrazie początkowym) – krawędzie są mocno rozmywane i następuje znaczna utrata informacji (ze względu na zwiększenie dyfuzyjności w całym obszarze).

W przypadku sprzężenia typu addytywnego wymagane jest dobre dobranie wartości współczynnika  $h_f$ , aby otrzymać rozsądne rezultaty. Zbyt mała wartość współczynnika prowadzi do niewystarczającego uwzględnienia informacji dodatkowych pochodzących z przestrzeni Hougha. Zbyt duża wartość współczynnika powoduje znaczną utratę informacji: w pierwszej fazie filtracji właściwa detekcja struktury występującej na obrazie nie jest możliwa ze względu na szum, w kolejnej fazie filtracji detekcja taka nie jest możliwa ze względu na zbyt intensywne rozmycie krawędzi po nieudanej początkowej detekcji kształtów.

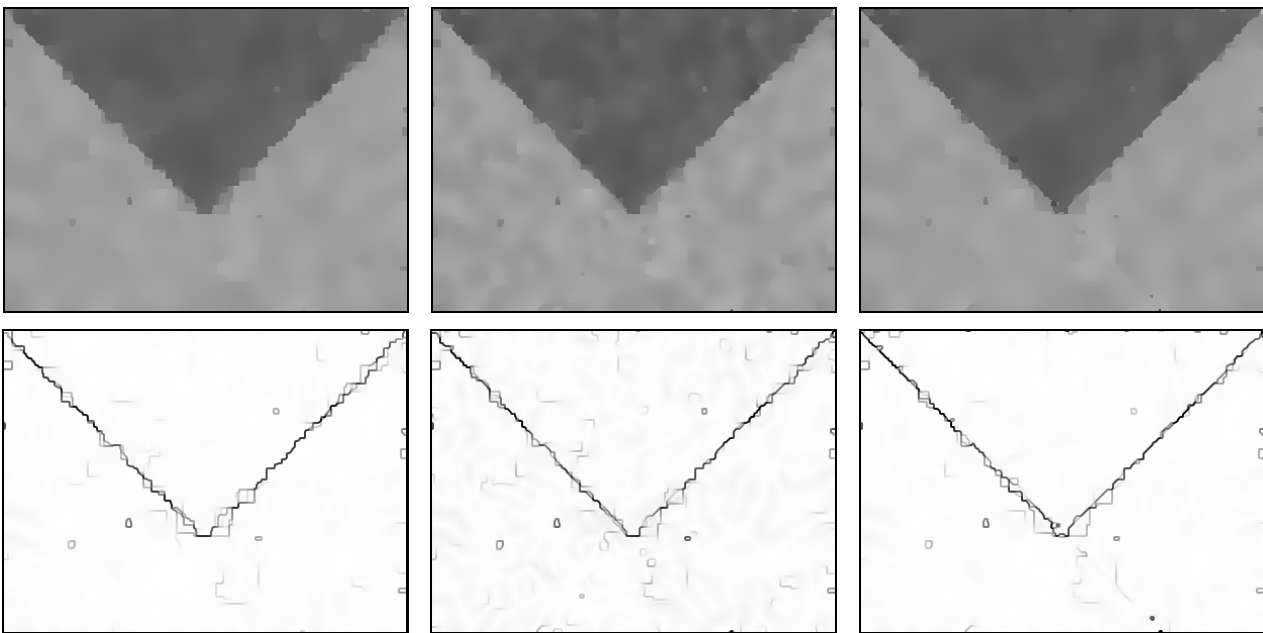
Z przedstawionych rezultatów wynika, że adaptacyjny sposób wyznaczania współczynnika sprzężenia może być przydatny także w sytuacji, gdy wszystkie struktury na obrazie mogą być całkowicie opisane w przestrzeni transformacji Hougha. Zmniejszenie sprzężenia w początkowej fazie odszumiania pozwala w takim przypadku na minimalizację utraty informacji wynikającej z błędnego rozpoznania usytuowania kształtów przez algorytm transformacji. W takim przypadku algorytm adaptacyjny może zostać

uruchomiony z parametrami, które nie zmniejszają w znaczący sposób wartość sprzężenia w końcowej fazie odsumiania. Może to być zrealizowane przez wybranie dużej wartości współczynnika  $h_\sigma$ .

### 3.4.2 Skuteczność filtracji dla współczynnika dyfuzji $c_2$ (sprężenie multiplikatywno-addytywne)

Dla porównania zostaną przedstawione rezultaty filtracji przy zastosowania sprzężenia multiplikatywno-addytywne. Jak to zostało zasygnalizowane w rozdziale 3.3.2, sprzężenie tego typu stosowane może być w przypadku obrazów, dla których transformata Hougha nie generuje krzywych pokrywających się całkowicie z poszukiwaną strukturą na obrazie oryginalnym. Celem takiego rodzaju współczynnika  $c$  jest zminimalizowanie wpływu sprzężenia w obszarach, gdzie wyznaczona struktura geometryczna nie pokrywa się z krawędziami występującymi na obrazie.

Na rysunku 3-15 zostały pokazane rezultaty filtracji obrazu testowego nr 1 algorytmem bez sprzężenia (pierwsza kolumna) oraz algorytmem sprzężonym w przypadku małej wartości współczynnika  $h_f=0.3$  (druga kolumna) i dużej wartości współczynnika  $h_f=0.9$ . Zastosowano funkcje stopujące  $g_2$  oraz  $g_{HI}$ . Przyjęto parametr skalujący  $h_s=0.5$ . Dla pozostałych parametrów zastosowano takie same wartości, jak w poprzednim eksperymencie.



**Rysunek 3-15: Wynik filtracji  $c_2$  po 300 iteracjach dla obrazu testowego nr 1 (zaszumienie  $\sigma_n = 400$ ).  
Wiersz drugi – detekcja krawędzi typu Sobel. Kolumna 1 – filtracja bez sprzężenia,  
kolumna 2 – filtracja sprzężona  $h_f=0.3$ , kolumna 3 – filtracja sprzężona  $h_f=0.9$ .**

Filtracja przeprowadzona przy współczynniku  $h_f=0.3$  nie powoduje znacznego poprawienia otrzymanych wyników. Proces odsumiania został spowolniony w porównaniu do filtracji bez sprzężenia. Jest to zgodne ze wzorem (3 – 26) określającym tego rodzaju sprzężenie. Zależność ta zakłada, że proces ma być celowo spowolniony w obszarach, gdzie prawdopodobnie występują krawędzie. Ze względu na zastosowanie małego współczynnika sprzężenia całkowita dyfuzja również została spowolniona efektem czego jest wolniejsze odsumianie obszarów, gdzie krawędzie nie są odnalezione. Pozytywnym działaniem

wynikającym z zastosowania sprzężenia jest nieznaczne, lecz zauważalne wyostrenie krawędzi obiektu. Otrzymane krawędzie są także bardziej regularne niż w przypadku filtracji bez użycia sprzężenia.

Użycie większej wartości współczynnika daje zauważalnie lepsze efekty. Obszary jednolite są szybciej odszumiane niż jest to w przypadku algorytmu bez sprzężenia. Otrzymane krawędzie obiektu są natomiast znacznie wyraźniejsze i regularne niż w obu poprzednich przypadkach. Kolejne iteracje powodują ciągłą poprawę obrazu bez utraty ostrości krawędzi, która jest regularyzowana przez stałe sprzężenie z przestrzenią Hougha.

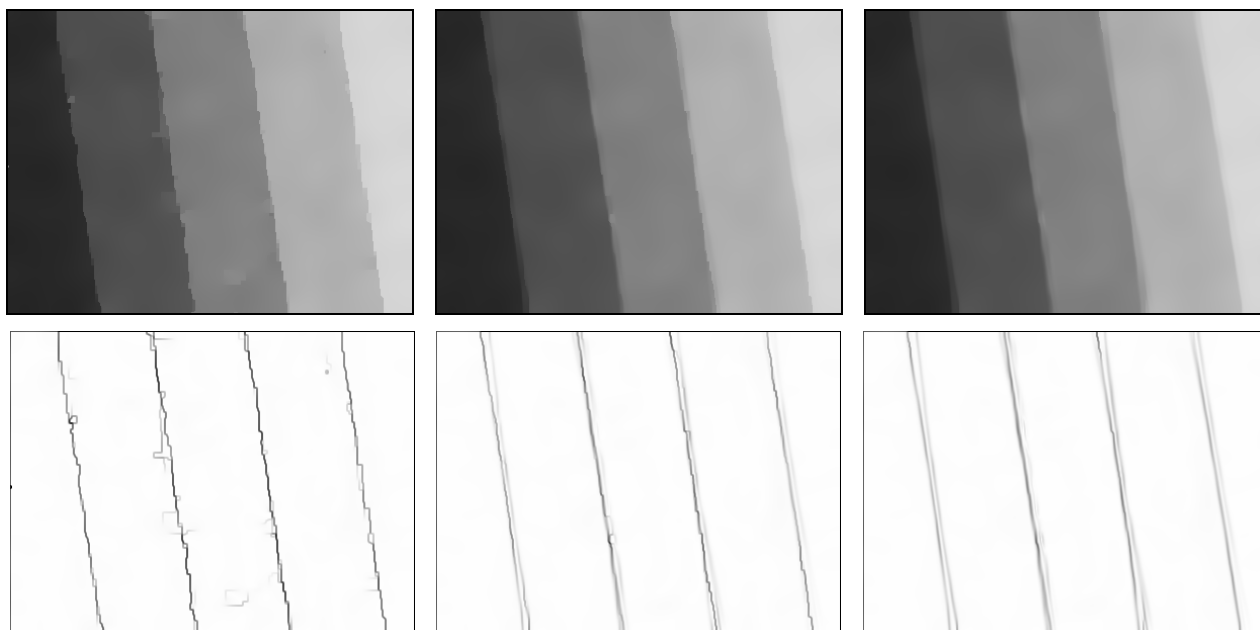
Należy zauważyć, że przy użyciu współczynnika sprzężenia zarówno o małej, jak i dużej wartości udało się prawie całkowicie zniwelować wpływ sprzężenia w obszarach, gdzie wyznaczona struktura nie pokrywa się z krawędziami obiektu na obrazie. Wpływ ten był bardzo widoczny w przypadku sprzężenia addytywnego. Tutaj jedynie niewielkie fragmenty takiego sprzężenia widoczne są przy zastosowaniu współczynnika  $h_f$  o dużej wartości. Z tego punktu widzenia nie istnieje więc konieczność używania w tym przypadku adaptacyjnej wersji algorytmu wyznaczającej współczynnik  $h_f$  w zależności od zaszumienia występującego na obrazie. Jednak użycie adaptacyjnego algorytmu może być podyktowane innymi względami – np. znacznym dużym zaszumieniem obrazu początkowego, w przypadku którego możliwa jest błędna detekcja kształtów. Poza tym zastosowanie wersji adaptacyjnej może być konieczne w przypadku modyfikacji parametru skalującego  $h_s$  w celu przyspieszenia procesu filtracji.

Podsumowując należy stwierdzić, że alternatywna postać sprzężenia  $c_2$  nie daje tak znaczącej poprawy jakości filtracji, jaka występowała w przypadku sprzężenia addytywnego. Jednakże pozwala na uzyskanie dobrej jakości rezultatów filtracji w przypadku, gdy wynik transformaty odwrotnej nie pokrywa się z krawędziami na obrazie. Sytuacja taka może mieć często miejsce także w przypadku bardzo dużego zaszumienia obrazu oraz dla obrazów rzeczywistych. W sytuacjach tego typu detekcja struktur występujących na obrazie może być często błędna, zwłaszcza dla początkowych iteracji (patrz rozdział 3.4.2). Sprzężenie multiplikatywno-addytywne pozwala na przeprowadzenie procesu filtracji bez wprowadzania błędnych informacji prowadzących do niepotrzebnego zahamowania dyfuzji w pewnych obszarach. Proces dyfuzji jest w tym przypadku prowadzony wolniej, lecz jest mniej podatny na rozbieżność informacji wyznaczonych na podstawie oszacowania krawędzi oraz oszacowania struktury.

### **3.4.3 Charakterystyka filtracji sprzężonej przy zastosowaniu wartości progowej dla przestrzeni transformaty odwrotnej**

Wyniki przedstawione w poprzednich podrozdziałach dotyczyły algorytmu, w którym do generacji odwrotnej transformaty brana była pod uwagę określona przez parametr  $\eta_H$  liczba maksimumów z przestrzeni Hougha. Wersja ta jest użyteczna w przypadku, gdy możliwe jest określenie z góry liczności struktur na obrazie, które powinny być uwydatnione. Jeżeli jednak wiedza dodatkowa ograniczona jest do znajomości typu struktury, alternatywne podejście jest wygodniejsze. W takim przypadku wygodniej jest zdefiniować względny próg czułości  $\psi_H$  w przestrzeni parametrów, powyżej którego struktura będzie odwzorowywana w procesie transformacji odwrotnej (patrz – konstrukcja transformaty odwrotnej).

Na rysunku 3-16 przedstawiono wynik filtracji dla 400 iteracji bez sprzężenia (lewa kolumna) oraz ze sprzężeniem o parametrach  $h_f=0.3$  (środkowa kolumna) oraz  $h_f=0.5$  (prawa kolumna). Użyte zostały następujące pozostałe parametry: sprzężenia addytywne  $c_1$ , funkcja stopująca,  $g_2$ ,  $T_H = 2$ ,  $\psi_H = 0.3$ .



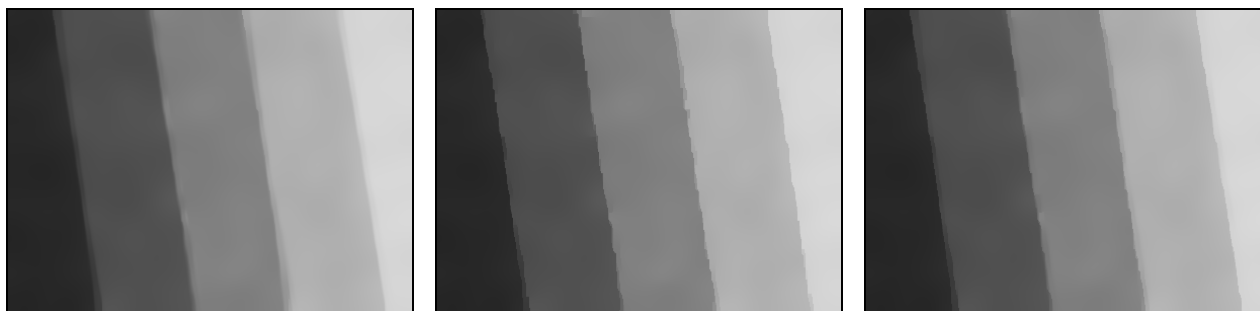
**Rysunek 3-16: Wynik filtracji po 400 iteracjach dla obrazu testowego nr 2 (zaszumienie  $\sigma_n = 100$ ).  
Wiersz drugi – detekcja krawędzi typu Sobel. Kolumna 1 – filtracja bez sprzężenia,  
kolumna 2 – filtracja sprzężona  $h_f=0.3$ , kolumna 3 – filtracja sprzężona  $h_f=0.5$ .**

Przeprowadzona filtracja daje bardzo dobre rezultaty:

- Detekcja istotnych struktur na obrazie została prawidłowo automatycznie przeprowadzona – wyostnione zostały tylko dominujące struktury tworzące linie proste.
- Dyfuzja w obszarach jednolitych (poza krawędziami) została przyspieszona w porównaniu do filtracji bez sprzężenia.
- Krawędzie znajdujące się na obrazie są znacznie bardziej regularne niż przy filtracji standardowej (co jest szczególnie dobrze widoczne w drugim rzędzie na rysunku 3-16).
- Proces przebiega stabilnie nawet dla bardzo wielu iteracji (dla 400 i więcej) – krawędzi pozostają stabilne ze względu na stopujące własności sprzężenia.

Należy zauważyć, że powyższe wyniki otrzymano przy stałym współczynniku sprzężenia  $h_f$ . Na rysunku 3-16 widoczne jest, że zwiększenie sprzężenia powoduje, że poszukiwane struktury (krawędzie pomiędzy obszarami o różnej intensywności) stają się bardziej regularne. Niepożądanym efektem wynikającym z charakteru transformacji Hougha jest wielokrotna reprezentacja pojedynczej linii przez kilka punktów w przestrzeni Hougha (nawet po zastosowaniu filtru *butterfly*). Przy tworzeniu przekształcenia odwrotnego brane są pod uwagę wszystkie punkty powyżej pewnej względnej wartości granicznej. Powoduje to, że dana linia ma kilka reprezentacji różniących się o pojedyncze wartości parametru. Są one bardziej uwydatnione przy wyższym współczynniku  $h_f$  ze względu na zmniejszoną dyfuzyjność w tych rejonach. Aby zminimalizować wpływ tego efektu możliwe jest:

- modyfikacja rozmiaru akumulatora używanego w transformacji Hougha tak, aby kwantyzacja przestrzeni była lepiej przystosowana do struktur geometrycznych występujących na obrazie – rozwiązanie to jest kłopotliwe w przypadku zwiększenia przestrzeni Hougha (może znacznie zwiększyć się złożoność obliczeniowa),
- przetwarzanie przestrzeni Hougha po wyznaczeniu transformaty np. przez odpowiednio przystosowany algorytm segmentacji pozwalający na wyodrębnienie obszarów reprezentujących maksima lokalne odnoszące się do pojedynczych struktur – w tym przypadku należy zwrócić uwagę, aby nie nastąpiła znaczna utrata informacji w sytuacji, gdy wiele podobnych struktur ma być rozróżnionych;
- uwzględnienie w procesie filtracji adaptacyjny sposób wyznaczania współczynnika sprzężenia – analogicznie jak to zostało pokazane w podrozdziale 3.4.1 proces taki pozwoliłby na dodatkowe wygładzenie błędnie wprowadzonych struktur w końcowej fazie filtracji, gdy większość zakłóceń została już usunięta dzięki sprzężeniu z przestrzenią Hougha.

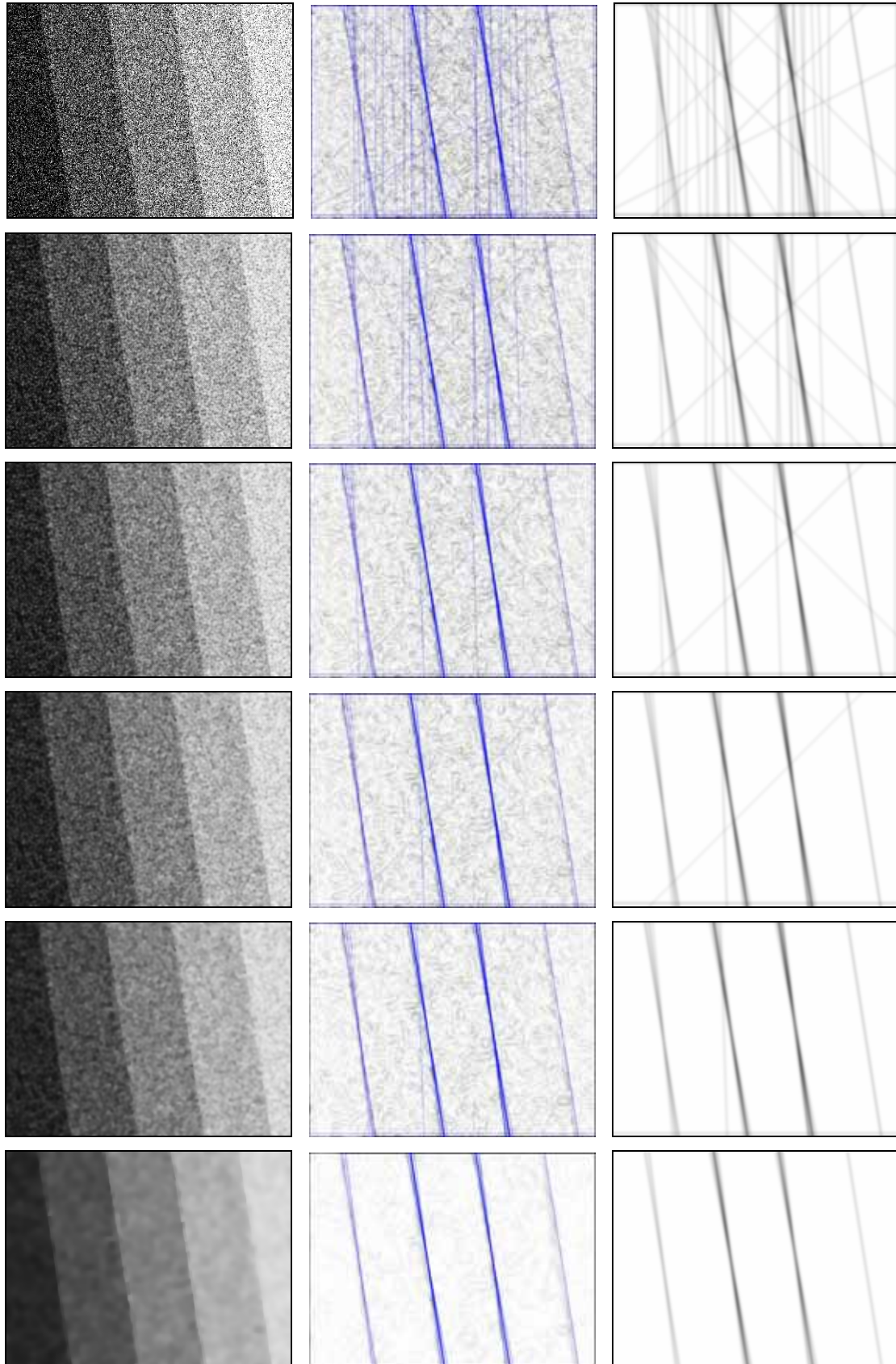


**Rysunek 3-17: Wynik filtracji po 400 iteracjach dla obrazu testowego nr 2 (zazumienie  $\sigma_n = 100$ ).  
 Obraz po lewej: stały współczynnik  $h_f = 0.5$ , obraz środkowy: współczynnik adaptacyjny  $h_\sigma = 15$ ,  
 obraz po prawej: współczynnik adaptacyjny  $h_\sigma = 25$ .**

Na rysunku 3-17 porównano wyniki filtracji uzyskane przy użyciu filtracji ze sprzężeniem przy stałym współczynniku  $h_f = 0.5$  oraz współczynnikami adaptacyjnym o parametrach odpowiednio  $h_\mu = 8$ ,  $h_\sigma = 15$  (obraz środkowy) oraz  $h_\sigma = 25$  (obraz prawy). Pozostałe parametry filtracji pozostały bez zmian.

Po zastosowaniu adaptacyjnego określania współczynnika sprzężenia udało się zmniejszyć efekt spowodowany wielokrotną reprezentacją pojedynczej struktury. W obu przypadkach uzyskane linie mają nadal zwiększoną regularność (w porównaniu do filtracji bez sprzężenia) oraz są ostrzejsze w porównaniu do filtracji o stałym współczynniku sprzężenia. Zastosowanie większego współczynnika  $h_\sigma$  daje efekty zbliżone coraz bardziej do filtracji o stałym współczynniku. Mniejsza wartość  $h_\sigma$  powoduje, że obraz będzie w mniejszym stopniu przetworzony przez proces sprzężony, przez co rezultaty zbliżone są do filtracji bez sprzężenia. Wybór optymalnej wartości współczynnika zależeć będzie od konkretnego zastosowania.

Rysunek 3-18 przedstawia etapy odsumiania obrazu testowego dla tych samych parametrów filtracji, które zostały użyte w poprzednim eksperymencie. W kolejnych wierszach przedstawiono rezultaty uzyskane po określonej liczbie iteracji. Pierwszy wiersz pokazuje oryginalny zasumiony obraz testowy przed rozpoczęciem filtracji (iteracja nr 0). W środkowej kolumnie przedstawiono detekcję krawędzi typu Sobel (kolor czarny i biały – pokazane w negatywie) z nałożoną wizualizacją przestrzeni  $H^1$  (kolor



Rysunek 3-18: Wynik filtracji sprzężonej  $h_f=0.3$  (kolumna lewa), detekcja krawędzi z nałożoną wizualizacją  $HT^{-1}$  (kolumna środkowa), wizualizacja transformaty  $HT^{-1}$  użytej w sprzężeniu (kolumna prawa). Wiersze odnoszą się odpowiednio do 0, 2, 4, 6, 10, 50 iteracji. Obraz testowy nr 2 (zazumienie  $\sigma_n = 100$ ).

niebieski). Odwrotna transformata używana przetworzona przez splot z jądrem Gaussa  $\sigma = 2$  przedstawiona jest w kolumnie 3. Tak przygotowana transformata odwrotna używana jest następnie w procesie sprzężenia.

Znajdujący się na początkowym obrazie szum powoduje, że odpowiadająca mu transformata Hougha również jest bardzo nieregularna. Zastosowanie progowania jasności pikseli oraz filtru *butterfly* pozwala wyodrębnić najbardziej wyraźne struktury znajdujące się na obrazie. Po dwóch iteracjach liczba struktur powstałych na podstawie detekcji krawędzi na obrazie zaczyna się zmniejszać. Struktury powstałe z rozróżnienia obszarów pozostają dominujące – przede wszystkim zostały usunięte struktury powstałe z zakłóceń formujących się w pionowe linie.

Kolejne iteracje powodują dalszą redukcję zaszumienia, a co za tym idzie redukcją struktur dominujących w przestrzeni Hougha. Już po 50 iteracjach wybrany podzbiór przestrzeni Hougha ograniczony jest do struktur opisujących poszukiwane krawędzie na obrazie. Dzięki temu dyfuzja w obszarze tych krawędzi będzie dodatkowo zahamowana, co przyspieszy filtrację i spowoduje, że krawędzie będą bardziej regularne. Stąd można uogólnić własność 3.1 w następujący sposób.

### **Własność 3.3**

Filtracja algorytmem realizującym sprzężenie z przestrzenią Hougha nie powoduje powstawania nowych struktur geometrycznych na przetwarzanym obrazie. Kolejne iteracje filtracji dają w wyniku coraz bardziej uproszczoną wersję obrazu oryginalnego zarówno dla funkcji jasności  $u(x, y)$ , jak i dla przestrzeni Hougha  $H$ .

## **3.5 Dyskusja i podsumowanie metod filtracji sprzężonej**

W rozdziale 3 przedstawiono możliwości uwzględnienia w procesie filtracji dodatkowej wiedzy o obrazie. Konstrukcja zaproponowanego podejścia jest ściśle oparta na metodach realizujących przestrzenną filtrację obrazów dwuwymiarowych. Krótko zostaną przedstawione zagadnienia implementacyjne związane z konkretną realizacją przedstawionego algorytmu.

### **Złożoność obliczeniowa algorytmu**

Pojedyncza iteracja podstawowej wersji algorytmu filtracji nieliniowej opisanej dyskretyzacją (2 – 61) wymaga wykonania  $O(W(I)H(I))$  operacji wyznaczenia nowej wartości pikseli. W praktyce należy również uwzględnić czas niezbędny na wyznaczenie oszacowania wystąpienia krawędzi, które jest używane, jako zmienna sterująca procesem filtracji.

W oszacowaniu złożoności algorytmu sprzężonego należy wziąć pod uwagę czas potrzebny na wyznaczenie nowej wartości piksela oraz obliczenie transformacji Hougha. Wyznaczenie nowej wartości piksela jest wykonywane w czasie  $O(W(I)H(I))$  operacji – sprzężenie nie wpływa na tą składową algorytmu, gdyż zostaje jedynie zmieniona postać dla lokalnego współczynnika dyfuzji.

Obliczenie transformacji Hougha jest najbardziej czasochłonnym zadaniem wykonywanym w trakcie filtracji algorytmem sprzężonym. Złożoność obliczeniowa transformacji zależy przede wszystkim od



przyjętego rodzaju transformaty oraz zastosowanego sposobu parametryzacji krzywych. Zastosowanie standardowej wersji transformaty jest najbardziej kosztowne obliczeniowo. W przypadku transformacji wykrywającej linie proste dla przestrzeni parametrów o rozmiarze  $s_\alpha$  złożoność obliczeniowa jest ograniczona przez  $O(W(I)H(I)s_\alpha)$ . Detekcja krzywych opisanych przez większą liczbę parametrów pociąga za sobą dalszy wzrost złożoności obliczeniowej. Jednocześnie znacznie wzrasta zapotrzebowanie na zasoby pamięci wynikające z rozmiaru akumulatora.

W praktyce nakład obliczeniowy związany z wyznaczeniem przestrzeni parametrów jest ograniczony przez liczbę punktów, które brane są pod uwagę przy wypełnianiu akumulatora. W przypadku skomplikowanych obrazów liczba ta może być znacząca, co powoduje znaczny wzrost wymaganych obliczeń. Częściowym rozwiązaniem problemu dużej złożoności algorytmu jest zastosowanie zmodyfikowanej wersji algorytmu transformaty Hougha. Zastosowanie probabilistycznej lub hierarchicznej odmiany transformaty pozwoliłoby na zmniejszenie nakładów obliczeniowych przy zachowaniu wysokiej skuteczności detekcji struktur geometrycznych. Jednakże badanie wpływu rodzaju transformaty na skuteczność algorytmu sprzężonego wykracza poza zakres prezentowanej pracy.

Złożoność obliczeniowa algorytmu dotyczy pojedynczej iteracji filtracji obrazu. W trakcie eksperymentów zauważono, że nie ma konieczność wyznaczania transformaty Hougha dla każdej iteracji. Doświadczenia praktyczne wykazały, że najbardziej istotne jest zaktualizowanie przestrzeni parametrów na początku procesu filtracji. Szybka poprawa jakości obrazu w pierwszych iteracjach algorytmu pozwala na znaczne zwiększenie dokładności wyznaczenia transformaty Hougha, a co za tym idzie także transformaty odwrotnej (podobny problem jest rozważany w rozdziale 3.3.2). W kolejnych iteracjach algorytmu aktualizacja przestrzeni Hougha może być znacznie rzadsza, dzięki czemu możliwe jest znaczne przyspieszenie wykonywanych obliczeń.

### **Alternatywne określenie sąsiedztwa**

Konstrukcja algorytmu sprzężonego zaprezentowanego w rozdziale 3.3.3 opiera się na dyskretyzacji równania różniczkowego procesu dyfuzji opisanego wyrażeniem (2 – 46). W dyskretyzacji zastosowany został schemat różnicowy wzorowany na przekształceniach opisanych w pracy [79]. Przyjęty sposób konstruowania różnic skończonych powoduje, że algorytm realizujący filtrację dla obrazów dyskretnych oparty jest na czterosąsiedztwie zdefiniowanym w rozdziale 2.1.

Jednakże możliwa jest konstrukcja algorytmu, która uwzględni ośmiosąsiedztwo przy przetwarzaniu pojedynczego piksela. Podejście opisane wyrażeniem (2 – 66) zostało zaproponowane między innymi w pracy [30], jako jedna z metod regularyzacji procesu filtracji. W takim przypadku należy rozpatrzyć wartości wagowe przy uwzględnianiu pikseli sąsiadujących diagonalnie. W cytowanej pracy autorzy proponują przyjąć wartość  $\sqrt{2}$ , jako wagę dla pikseli diagonalnych, co wynika z dyskretnej miary odległości dla siatki pikseli obrazu.

Wyznaczanie nowych wartości pikseli przy użyciu ośmiosąsiedztwa dla punktu może pozytywnie wpływać na własności procesu. Jednakże w takim przypadku należy również uwzględnić warunek

monotoniczności procesu w odniesieniu do dyskretyzacji warstwy czasowej równania różniczkowego. W ogólności mniejsza wartość kroku czasowego pozwala dokładniej odwzorować przebieg procesu ciągłego przy użyciu dyskretnej aproksymacji. Można pokazać [30], że dla zwiększającej się liczby punktów uwzględnianych dla sąsiedztwa piksela centralnego, zmniejsza się maksymalna dopuszczalna wartość kroku czasowego, przy której dyskretyzacja procesu jest poprawna.

### **Kryterium stopu algorytmu**

W przedstawionych badaniach badano przebieg procesu filtracji po wykonaniu określonej liczby iteracji. W zastosowaniach praktycznych najczęściej istnieje konieczność określenia automatycznego kryterium stopu algorytmu. Dla zagadnienia filtracji obrazów trudno jest określić moment, kiedy filtracja powinna zostać zakończona. W przypadku filtrów liniowych można wyznaczyć, po jakim czasie zostaną usunięte zakłócenia występujące z określonym natężeniem. Ze względu na brak ogólnego rozwiązania dla nieliniowego równania różniczkowego, analiza taka nie jest możliwa dla algorytmów opartych na modelu Perona-Malik. Jest wiele sposobów zdefiniowania kryterium stopu dla aproksymacji numerycznej równania nieliniowego, które mogą być także bezpośrednio używane do zaproponowanego algorytmu sprzężonego. Wśród nich można wymienić następujące rozwiązania:

- Zmiana opisu procesu z równania różniczkowego w postaci (2 – 66) w model reakcyjno-dyfuzyjny przedstawiony w pracy [77]. Jedną z podstawowych zalet modelu opisanego w cytowanej pracy jest właśnie brak konieczności określania warunków stopu dla algorytmu. Jednakże zamiast ustalania chwili czasowej, w której algorytm kończy swoje działanie, w modelu tym występuje konieczność określenia nowego parametru, który decyduje o zatrzymaniu się algorytmu. W praktyce oba modele mogą być w dużej części rozpatrywane równoważnie, a ich wybór zależy od wygody określenia danego kryterium stopu dla konkretnego zastosowania.
- Wyznaczenie współczynnika globalnej zmiany szybkości dyfuzji na podstawie dwóch następujących po sobie pojedynczych przebiegów algorytmu. Konstrukcja takiego rodzaju kryterium jest analogiczna do sposobu określenia warunków stopu dla wielu problemów optymalizacyjnych. W tym przypadku mogą być zastosowane globalne miary odległości pomiędzy obrazami przedstawione w rozdziale 2.1.
- Wyznaczenie współczynnika globalnej zmiany szybkości dyfuzji na podstawie aktualnej (po filtracji) oraz oryginalnej zawartości obrazu. W pracy [74] zaproponowane zostało przykładowe kryterium wyznaczone na podstawie korelacji pomiędzy stanem po filtracji oraz stanem początkowym.

Wybór kryterium stopu do obrazów nie jest zadaniem łatwym ze względu na brak jednoznacznego określenia osiągnięcia pożądanej jakości obrazu. Na podstawie przeprowadzonych eksperymentów, można jednakże stwierdzić, że zastosowanie filtracji sprzężonej dodatkowo powinno ułatwić określenie kryterium stopu. W rozdziale 3.4 zostały przedstawione eksperymentalnie potwierdzone własności algorytmu, które zapewniają, że krawędzie obrazu dla wielu iteracji pozostają stabilne. Dzięki temu łatwiejsze jest zastosowanie kryterium opartym na porównywaniu zmiany odległości pomiędzy rezultatami dwóch

kolejnych iteracji filtracji. Zastosowanie takiego kryterium dla innego rodzaju algorytmów byłoby znacznie utrudnione np. dla oryginalnego modelu Perona-Malik, a praktycznie niemożliwe dla modelu liniowego. Jest to więc dodatkowa zaleta wynikająca z zastosowania zaproponowanego algorytmu sprzężonego.

## Podsumowanie

Filtracja realizująca nieliniowe równanie różniczkowe cząstkowe uwzględnia jedynie informację o lokalnie wyznaczonym gradiencie funkcji obrazu. Zaproponowana została metoda uwzględnienia w procesie filtracji wiedzy o strukturze geometrycznej obrazu. Metoda została opisana i przebadana, a otrzymane rezultaty można podsumować w następujących punktach:

- zaproponowano algorytm filtracji realizujący uwzględnienie wiedzy o strukturze geometrycznej obrazu przez sprzężenie z przestrzenią Hougha,
- zastosowanie transformacji Hougha powoduje przyspieszenie procesu odszumiania dzięki uwzględnieniu jej filtrujących własności,
- krawędzie na obrazach przetwarzanych przez algorytm sprzężony pozostawały przez dłuższy czas stabilne w porównaniu do algorytmu standardowego,
- filtracja algorytmem sprzężonych pozwoliła uzyskać bardziej regularne krawędzie,
- sprzężenie z przestrzenią Hougha przeznaczoną dla określonej grupy struktur pozwala na uwydatnienie geometrycznie określonych cech na obrazie,
- opracowano dwa modele sprzężenia, których zastosowanie zależy od użytego rodzaju transformacji Hougha,
- opracowano i przetestowano metodę adaptacyjnego sposobu określania wartości współczynnika sprzężenia  $h_f$ ,
- adaptacyjne wyznaczanie współczynnika  $h_f$  pozwala na zastosowanie sprzężenia addytywnego nawet w przypadku niecałkowitego odwzorowania struktur obrazu przez odwrotną transformację Hougha,
- przetestowano dwa sposoby tworzenia transformaty odwrotnej: uwzględniający ściśle określoną liczbę struktur oraz progowe uwzględnianie maksimum lokalnych przestrzeni  $H$ ,
- realizacja filtracji przy progowym uwzględnianiu struktur daje bardzo dobre wyniki bez konieczności określania liczności struktur,
- zaproponowana metoda nie narzuca rodzaju struktur, które mają być uwydatnione przy filtracji – konstrukcja algorytmu realizującego metodę pozwala na zastosowanie dowolnego rodzaju transformacji Hougha (także dotyczącej dowolnych kształtów, a nie tylko krzywych analitycznych),
- przedstawiony algorytm został przetestowany na obrazach testowych, przedstawiono otrzymane wyniki w postaci otrzymanych obrazów oraz porównanie skuteczności różnych rodzajów algorytmów z filtracją bez sprzężenia przy użyciu miary  $SNR$ ,

- prawidłowy dobór parametrów algorytmu pozwala na uzyskanie lepszych rezultatów w porównaniu do filtracji bez sprzężenia.

Zaproponowana metoda daje obiecujące wyniki. Główną jej zaletą jest połączenie własności filtracji opartej na nieliniowym procesie dyfuzji z odpornością transformacji Hougha na zakłócenia i nieregularności występujące na obrazie. Algorytm realizujący filtrację sprzężoną może być używany w przypadku konieczności uwydatnienia określonych struktur (krzywych analitycznych, kształtów). Nie jest on ograniczony do jednego typu transformacji. W przedstawionym rozdziale skoncentrowano się na pokazaniu, w jaki sposób sprzężenie pomiędzy różnymi estymatorami cech może zostać zrealizowane oraz jak skutecznie takie sprzężenie wpływa na proces filtracji.

## **Część II**

# **Przetwarzanie trajektorii w przestrzeni rotacji**

## Rozdział 4

### Modele opisu przestrzennej orientacji obiektów

Aby w pełni opisać położenie obiektu w przestrzeni należy podać zestaw parametrów zależny od przyjętej reprezentacji danych oraz od zdefiniowania układu odniesienia. W przestrzeni trójwymiarowej położenie punktu materialnego w pełni określone jest przez wektor translacji względem początku układu współrzędnych. W przypadku bryły sztywnej konieczne jest dodatkowo określenie orientacji. Opis komplikuje się jeszcze bardziej, gdy występuje konieczności przedstawienia zmiany wymiarów ciała (czynnik skalujący). W praktyce jednak skalowanie wymiarów rzadko jest używane. Jeżeli nie zaznaczono inaczej w prezentowanych zagadnieniach przyjmuje się, że jedynymi dopuszczalnymi transformacjami są translacje oraz rotacje. Dzięki temu zapewniony jest warunek, że zachowana zostaje odległość pomiędzy punktami należącymi do obiektu.

Prawidłowe określenie położenia i orientacji obiektu w przestrzeni  $SE(3)$  niezbędne jest w bardzo wielu zastosowaniach praktycznych. Podstawowe zastosowania obejmują między innymi następujące dziedziny:

- w robotyce odpowiedni zestaw parametrów określa położenie robota w przestrzeni roboczej oraz położenie oraz orientację jego efektorów względem korpusu – konieczne jest zaplanowanie trajektorii ruchu oraz kontrola umiejscowienia poszczególnych członów w przestrzeni [61],
- w lotnictwie niezbędne jest określenie pozycji i orientacji samolotu oraz dodatkowych parametrów potrzebnych do planowania sposobu wykonywania manewrów (np. lądowania); szczególnym przypadkiem jest opis obiektów umiejscowionych w przestrzeni kosmicznej (satelity, promy kosmiczne),
- w nawigacji do określenia położenia obiektu względem punktów odniesienia na mapie,
- w astronomii do określenia położenia ciał niebieskich,
- w animacji komputerowej do opisanie położenia oraz przetwarzania danych reprezentujących położenie oraz orientację obiektów w wirtualnym świecie generowanym przez komputer; w tym przypadku często informacje dotyczące świata wirtualnego łączone są z danymi pochodzącymi ze świata rzeczywistego (efekty specjalne w produkcjach filmowych, grach, itp.).

W wielu przypadkach położenie punktu w przestrzeni zmienia się w czasie kreśląc trajektorię. Trajektoria może być rejestrowana na podstawie aktualnego położenia lub generowana w celu określenia

położenia w zaplanowanym momencie czasu. W wielu przypadkach planowana trajektoria jest zależna od innych czynników i sił wpływających na położenie obiektu (np. w przypadku ruchu satelity). Informacje o zapisanej trajektorii mogą służyć do dalszej analizy. Odpowiednie układy sterujące (wykonawcze) mają za zadanie osiągnięcie zaplanowanego położenia obiektu przy zachowaniu zdefiniowanej trajektorii.

Sposoby opisu pozycji obiektu w przestrzeni są dobrze opracowane. W prosty sposób można uogólnić metody opisu dla przestrzeni dwuwymiarowej i zastosować je dla wyższych wymiarów – w praktyce dotyczy to przestrzeni trójwymiarowej, choć uogólnienie przenosi się bezpośrednio dla dowolnych wymiarów. Wszelkie miary odległości również w prosty sposób przenoszą się na euklidesowe przestrzenie wielowymiarowe. Przetwarzanie danych opisujących położenie w niższych wymiarach ma swoje bezpośrednie odpowiedniki dla wymiarów wyższych.

Bardziej skomplikowane jest zdefiniowanie orientacji obiektu w przestrzeni. Informacja taka może być opisana przy pomocy różnych parametryzacji, które będą bezpośrednio wpływać na właściwości zapisywanych danych. Przetwarzanie danych orientacji układu jest również znacznie mniej zbadanym zagadnieniem. Kolejne rozdziały skoncentrowane będą na zagadnieniu opisu i przetwarzania orientacji w przestrzeni

Szczególnym przypadkiem zastosowania określenia położenia obiektu w przestrzeni jest opis ruchu człowieka przy użyciu uproszczonej hierarchicznej struktury reprezentującej szkielet postaci. Opis taki jest bardzo istotny w systemach animacji wykorzystujących dane przedstawiające naturalny ruch ludzki. Źródło danych o ruchu może być różnego rodzaju: naturalne (systemy przechwytywania ruchu) lub sztuczne (syntetyczna generacja sekwencji). Niezależnie jednak od sposobu uzyskania danych, ruch ludzki może być reprezentowany jako zestaw trajektorii opisujących położenie w przestrzeni najistotniejszych części struktury reprezentującej szkielet postaci. Problematyka związana z opisem i przetwarzaniem danych ruchu ludzkiego jest bardzo istotna w wielu zagadnieniach praktycznych, przykładowo w następujących zastosowaniach:

- systemy animacji komputerowej – tworzenie realistycznych efektów specjalnych w produkcjach filmowych, grach, itp.,
- systemy diagnostyki medycznej – analiza danych reprezentujących ruch pacjentów z punktu widzenia poprawności budowy układu motorycznego, diagnostyki chorób układu ruchu, wyników terapii przeprowadzanej z pacjentem,
- systemy nadzorowanego treningu sportowego – analiza sposobu poruszania się zawodnika z punktu widzenia kinematyki i dynamiki ciała,
- systemy bezpieczeństwa – identyfikacja postaci na podstawie zapisanego ruchu, semantyczna analiza ruchu, wczesne wykrywanie i ostrzeganie przed działaniami terrorystycznymi itp.,
- systemy kontroli dostępu – określenie uprawnień w dostępie do wybranych pomieszczeń/obszarów na podstawie sposobu wykonywania ruchów jednoznacznie identyfikującego osobę,

- systemy wspomagające dla osób upośledzonych – np. analiza języka migowego na podstawie rozpoznawania wykonywanych ruchów, interakcja człowiek-komputer na podstawie gestów.

W kolejnych podrozdziałach zostaną zaprezentowane różne sposoby opisu orientacji w przestrzeni. W ostatnim podrozdziale krótko zostanie omówiona tematyka związana z reprezentacją położenia ciała ludzkiego (w ogólności hierarchicznej struktury szkieletowej) w przestrzeni. W rozdziale 5 zostaną przedstawione istniejące oraz zaproponowane nowe metody pozwalające na wydajne przetwarzanie danych opisujących orientację w przestrzeni zapisaną przy użyciu omawianych sposobów parametryzacji.

## 4.1 Parametryzacja przestrzeni translacji i orientacji

Na opis położenia w przestrzeni składa się podanie pozycji i orientacji ciała sztywnego. Pozycję punktu materialnego określa się w postaci wektora  $P$  reprezentującego translację względem wybranego układu współrzędnych  $A$  (oznaczenia układów współrzędnych oraz przekształcenia na podstawie [18]).

$${}^A P = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (4-1)$$

Opis orientacji obiektu wymaga związania z nim lokalnego układu współrzędnych  $B$ . Orientacja obiektu może być określona, jako położenie układu współrzędnych  $B$  względem układu odniesienia  $A$ . Z takiego podejścia wynika bezpośrednio sposób opisu orientacji polegający na zapisie wersorów osi głównych układu  $B$  w układzie  $A$ .

Wersorem wektora  $\vec{x}$  nazywa się wektor  $\hat{x}$  spełniający warunek

$$\hat{x} = \frac{\vec{x}}{\|\vec{x}\|} \quad (4-2)$$

Wersory o kierunkach i zwrotach zgodnych z osiami prostokątnego układu współrzędnych  $A$  nazywają się wersorami osi głównych układu i oznaczane będą przez  $\hat{x}_A, \hat{y}_A, \hat{z}_A$ . Zestawienie wersorów zapisanych względem układu  $A$  tworzy macierz obrotu  ${}^A R$  układu  $B$  względem układu odniesienia  $A$ :

$${}^A R = \begin{bmatrix} {}^A \hat{x}_B & {}^A \hat{y}_B & {}^A \hat{z}_B \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \hat{x}_B \cdot \hat{x}_A & \hat{y}_B \cdot \hat{x}_A & \hat{z}_B \cdot \hat{x}_A \\ \hat{x}_B \cdot \hat{y}_A & \hat{y}_B \cdot \hat{y}_A & \hat{z}_B \cdot \hat{y}_A \\ \hat{x}_B \cdot \hat{z}_A & \hat{y}_B \cdot \hat{z}_A & \hat{z}_B \cdot \hat{z}_A \end{bmatrix} \quad (4-3)$$

W macierzy obrotu składowe każdego wektora są rzutami tego wektora w kierunku osi jego układu odniesienia. W równaniu (4-3) zapisane to zostało jako iloczyn skalarny pary wersorów odpowiednich układów.

Transformacją ortogonalną nazywa się transformację  $F: V \rightarrow V$ , która dla dowolnych wektorów  $\vec{v}_1, \vec{v}_2$  zachowuje warunek:

$$\langle \vec{v}_1, \vec{v}_2 \rangle = \langle F\vec{v}_1, F\vec{v}_2 \rangle, \quad (4-4)$$

gdzie  $\langle \cdot, \cdot \rangle$  oznacza iloczyn skalarny. Transformacja ortogonalna zachowuje długości oraz kąty pomiędzy wektorami. Analogicznie macierzą ortogonalną  $A$  nazywa się macierz, która spełnia wyrażenie



$$AA^T = A^T A = I, \quad (4-5)$$

gdzie  $I$  oznacza macierz jednostkową. Wszystkie wiersze macierzy ortogonalnej tworzą ortonormalną bazę, to znaczy mają długość jeden oraz są liniowo niezależne (wzajemnie prostopadłe).

Przy realizacji opisu położenia obiektu należy rozróżnić orientację obiektu od rotacji. **Orientacja** jest określona przez położenie osi lokalnego układu współrzędnych  $B$  względem osi układu odniesienia  $A$ . Orientacja jest więc częścią składową opisu położenia obiektu. **Rotacja** jest natomiast ortogonalną transformacją, która przekształca pewien wektor (lub układ współrzędnych) w inny wektor przy spełnieniu własności (4-4). Macierz  $R$  może być użyta jako operator (ponieważ operacja dotyczy wektorów, nie są oznaczane indeksami układy współrzędnych odniesienia i przekształcany) do zmiany orientacji wektora przez wykonanie rotacji określonej jako:

$$\vec{v}_1 = R\vec{v}_0 \quad (4-6)$$

Punkt obrotu niech będzie zaczepiony w początku układu współrzędnych razem z wektorem  $\vec{v}_0$ .

Rotacja ma zachowywać odległości, czyli długość wektora pozostaje stała:

$$|\vec{v}_1| = |\vec{v}_0| \quad (4-7)$$

Stąd wynikają następujące zależności [12]:

$$|\vec{v}_1|^2 = |\vec{v}_0|^2 \Rightarrow \vec{v}_1^T \vec{v}_1 = \vec{v}_0^T \vec{v}_0 \Rightarrow \vec{v}_0^T R^T R \vec{v}_0 = \vec{v}_0^T \vec{v}_0 \Rightarrow R^T R = I \quad (4-8)$$

Oznacza to, że macierz obrotu  $R$  musi być ortogonalna, co wynika z definicji przekształcenia oraz warunku podanego w równaniu (4-5). Macierze rotacji dla przestrzeni trójwymiarową tworzą specjalną grupę ortogonalną  $SO(3)$ . Jest to grupa macierzy ortogonalnych o wymiarach  $3 \times 3$ , o wyznaczniku 1.

Własności specjalnej grupy ortogonalnej mogą być zdefiniowane w alternatywny sposób. Niech macierz obrotu  $R$  zapisana będzie w postaci trzech kolumnowych wektorów  $R = [noa]$ . Wektory składowe macierzy spełniają warunek jednostkowości, to znaczy  $\|\vec{n}\| = \|\vec{o}\| = \|\vec{a}\| = 1$ . Ponadto przyjmuje się, że w dalszej części pracy rozważane będą układy prawoskrętne, dla których zachodzi następująca własność:

$$\vec{n} \times \vec{o} = \vec{a}, \quad (4-9)$$

gdzie  $\times$  oznacza operację iloczynu wektorowego. Dla układów lewoskrętnych zachodzi analogiczny warunek określony jako  $\vec{n} \times \vec{o} = -\vec{a}$ .

Odwzorowanie układów może być w ogólnym przypadku realizowane przez macierz przekształcenia jednorodnego uwzględniającą jednocześnie rotację i translację. Niech wektor  $P$  określony będzie względem układu  $B$ . Aby wyznaczyć jego położenie względem układu  $A$  należy uwzględnić translację  ${}^A_B T$  oraz rotację  ${}^A_B R$  układu  $B$  względem układu  $A$ :

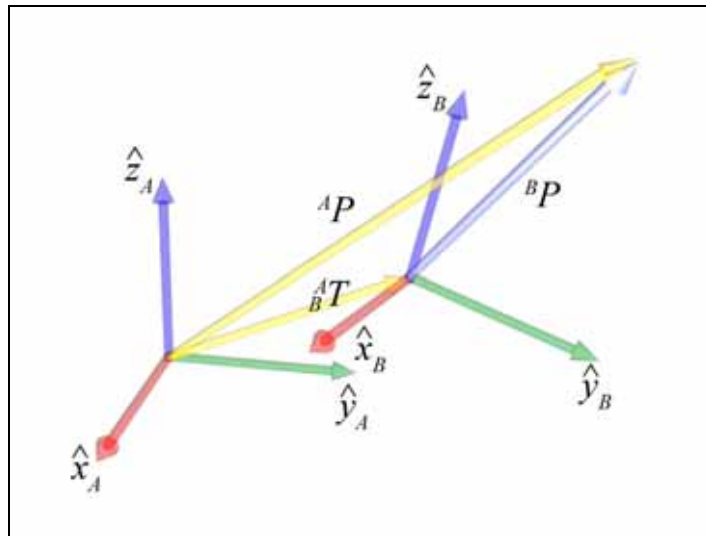
$${}^A P = {}^A_B R {}^B P + {}^A_B T, \quad (4-10)$$

co można zapisać w postaci przekształcenia jednorodnego:

$$\begin{bmatrix} {}^A P \\ \dots \\ 1 \end{bmatrix} = \begin{bmatrix} {}^A_B R & | & {}^A_B T \\ \dots & & \dots \\ 0 & 0 & 0 & | & 1 \end{bmatrix} \begin{bmatrix} {}^B P \\ \dots \\ 1 \end{bmatrix}. \quad (4-11)$$

Dodatkowy wiersz macierzy przekształcenia można modyfikować, aby opisać przekształcenie realizujące rzutowanie i skalowanie.

Na rysunku 4-1 przedstawiono przypadek przekształcenia układu współrzędnych  $B$  względem  $A$ . Wektor  $P$  jest zaczepiony w początku układu  $B$  (kolor jasnoniebieski). Współrzędne wektora wyznaczone równaniem (4 – 11) względem układu  $A$  są przedstawione kolorem żółtym.



Rysunek 4-1: Przekształcenie układu współrzędnych  $B$  względem  $A$  przez złożenie translacji i rotacji.

Dalsza część pracy skoncentrowana jest na przekształceniu orientacji realizowane przez wykonywanie rotacji. Jeżeli nie zaznaczono inaczej, część translacyjna przekształcenia nie będzie uwzględniana.

#### 4.1.1 Interpretacja macierzy rotacji – twierdzenie Eulera

Zależność (4 – 6) przedstawia operację przekształcenia wektora za pomocą rotacji opisanej macierzą  $R$ . Wiadomo, że macierz ta należy do specjalnej grupy ortogonalnej, z czego wynika, że  $\det(R) = 1$ . Wartości własne macierzy mogą być wyznaczone przez znalezienie miejsc zerowych wielomianu charakterystycznego (wyprowadzenie na podstawie [12]):

$$\det(R - \lambda I) = 0 \quad (4 - 12)$$

$$\det(R - \lambda I) = -\lambda^3 + \lambda^2 \operatorname{tr}(R) - \lambda(M_{11} + M_{22} + M_{33}) + \det(R) = 0, \quad (4 - 13)$$

gdzie  $\operatorname{tr}(R)$  jest śladem macierzy zdefiniowanym jako:

$$\operatorname{tr}(R) = \sum_{i=1}^3 r_{ii}, \quad (4 - 14)$$

gdzie  $r_{ii}$  oznacza element macierzy rotacji  $R$ . W przekształceniach  $M_{ii}$  oznacza minor główny macierzy  $R$ . Minor główny macierzy  $R$  wyznacza się, jako wyznacznik macierzy utworzonej na podstawie macierzy  $R$  przez wykreślenie  $i$ -tego wiersza oraz  $i$ -tej kolumny. Można pokazać, że dla macierzy w  $SO(3)$  zachodzi  $M_{ii} = r_{ii}$ , co prowadzi do zależności:

$$\det(R - \lambda^{(R)} I) = -(\lambda^{(R)} - 1) \left[ (\lambda^{(R)})^2 - \lambda^{(R)} (\operatorname{tr}(R) - 1) + 1 \right] = 0, \quad (4 - 15)$$

Przyjmując podstawienie

$$\cos \varphi = \frac{\text{tr}(R) - 1}{2} \quad (4-16)$$

otrzymuje się

$$\det(R - \lambda^{(R)} I) = -(\lambda^{(R)} - 1) \left( (\lambda^{(R)})^2 - 2\lambda^{(R)} \cos \varphi + 1 \right) = 0 \quad (4-17)$$

Skąd wartości własne macierzy  $R$  wynoszą odpowiednio

$$\lambda_1^{(R)} = 1 \quad (4-18)$$

$$\lambda_{2,3}^{(R)} = \cos \varphi \pm i \sin \varphi = e^{\pm i\varphi} \quad (4-19)$$

Niech  $\vec{e}_1^{(R)}$ ,  $\vec{e}_2^{(R)}$ ,  $\vec{e}_3^{(R)}$  będą wektorami własnymi macierzy  $R$  skojarzonymi odpowiednio z wyznaczonymi wartościami własnymi:

$$R \vec{e}_j^{(R)} = \lambda_j^{(R)} \vec{e}_j^{(R)} \quad (4-20)$$

Wektor własny  $\vec{e}_1^{(R)}$  jest osią obrotu realizowanego przez macierz  $R$ . Z definicji wynika, że operacja rotacji nie zmienia położenia tego wektora:

$$R \vec{e}_1^{(R)} = \lambda_1^{(R)} \vec{e}_1^{(R)} = \vec{e}_1^{(R)} \quad (4-21)$$

Przy użyciu pozostałych wektorów własnych konstruuje się dwa wektory rzeczywiste:

$$\vec{c}_2 = \frac{1}{2} (\vec{e}_2^{(R)} + \vec{e}_3^{(R)}) \quad (4-22)$$

$$\vec{c}_3 = \frac{i}{2} (\vec{e}_2^{(R)} - \vec{e}_3^{(R)}) \quad (4-23)$$

Wektory  $\vec{c}_2$  i  $\vec{c}_3$  są ortogonalne względem siebie oraz względem wektora  $\vec{e}_1^{(R)}$ . Dodatkowo wektory te rozpinają płaszczyznę, która jest ortogonalna względem wektora  $\vec{e}_1^{(R)}$ . Po przekształceniu tych wektorów przy użyciu macierzy  $R$  otrzymuje się:

$$\begin{aligned} R\vec{c}_2 &= \frac{1}{2} R(\vec{e}_2^{(R)} + \vec{e}_3^{(R)}) = \frac{1}{2} (\lambda_2^{(R)} \vec{e}_2^{(R)} + \lambda_3^{(R)} \vec{e}_3^{(R)}) = \frac{1}{2} (\cos \varphi + i \sin \varphi) \vec{e}_2^{(R)} + \frac{1}{2} (\cos \varphi - i \sin \varphi) \vec{e}_3^{(R)} \\ &= \vec{c}_2 \cos \varphi + \vec{c}_3 \sin \varphi \end{aligned} \quad (4-24)$$

$$\begin{aligned} R\vec{c}_3 &= \frac{i}{2} R(\vec{e}_2^{(R)} - \vec{e}_3^{(R)}) = \frac{i}{2} (\lambda_2^{(R)} \vec{e}_2^{(R)} - \lambda_3^{(R)} \vec{e}_3^{(R)}) = \frac{i}{2} (\cos \varphi + i \sin \varphi) \vec{e}_2^{(R)} - \frac{i}{2} (\cos \varphi - i \sin \varphi) \vec{e}_3^{(R)} \\ &= \vec{c}_3 \cos \varphi - \vec{c}_2 \sin \varphi \end{aligned} \quad (4-25)$$

Macierz  $R$  powoduje obrót wektorów  $\vec{c}_2$  i  $\vec{c}_3$  wokół osi obrotu reprezentowanej przez wektor własny  $\vec{e}_1^{(R)}$  o kąt  $\varphi$ . Otrzymane rezultaty można podsumować twierdzeniem Eulera.

### Twierdzenie Eulera

Każde przekształcenie zmieniające orientację wektora (lub układu współrzędnych) znajdującego się w przestrzeni  $R^3$  może być opisane za pomocą osi obrotu  $\hat{n}$  oraz kąta obrotu  $\varphi$ .

#### 4.1.2 Sposoby parametryzacji rotacji

Zostało pokazane, że macierz rotacji  $R \in SO(3)$  reprezentuje przekształcenie wektora lub układu współrzędnych. Taka parametryzacja jest wygodna z punktu widzenia konstruowania kolejnych

przekształceń ze względu na możliwość ich składania. Wykonanie dwóch rotacji może zostać zastąpione jedną opisaną iloczynem poszczególnych macierzy:

$$R_2 R_1 = R_{21} \quad (4-26)$$

Należy pamiętać, że kolejność mnożenia macierzy niekoniecznie jest przemienne. Stąd wynika, że kolejność wykonywanych rotacji niekoniecznie jest przemienne, co jest zgodne z przestrzenną intuicją. Kolejność wykonywania rotacji może zostać zmieniona tylko dla szczególnych przypadków (np. obie rotacje wykonywane są wokół tej samej osi lub gdy jedna z rotacji jest zerowa).

Rotacje zapisane w postaci macierzowej nie zawsze są wygodne do użycia. W szczególności nie jest możliwe łatwe określenie rotacji, która ma być wykonana przez wypełnienie macierzy  $R$ . Przy użyciu twierdzenia Cayleya [9] oraz zastosowaniu wzoru Rodrigueza [12] można pokazać, że orientację można opisać przy użyciu 3 zamiast 9 liczb (patrz dodatek). Na tej podstawie konstruuje się różnego rodzaju parametryzacje służące do opisu orientacji. Użycie wybranego sposobu opisu często zależy od specyfiki zastosowania.

### Parametryzacja oś-kąt

Sposób opisu orientacji przez zdefiniowanie osi oraz kąta obrotu wynika bezpośrednio z twierdzenia Eulera. Można pokazać, że dla osi obrotu zdefiniowanej jako  $\hat{n} = [n_x \ n_y \ n_z]^T$  oraz kąta obrotu  $\varphi$  otrzymuje się następującą równoważną macierz obrotu [18]:

$$R_n(\varphi) = \begin{bmatrix} n_x n_x v_\varphi + c_\varphi & n_x n_y v_\varphi - n_z s_\varphi & n_x n_z v_\varphi + n_y s_\varphi \\ n_x n_y v_\varphi + n_z s_\varphi & n_y n_y v_\varphi + c_\varphi & n_y n_z v_\varphi - n_x s_\varphi \\ n_x n_z v_\varphi - n_y s_\varphi & n_y n_z v_\varphi + n_x s_\varphi & n_z n_z v_\varphi + c_\varphi \end{bmatrix}, \quad (4-27)$$

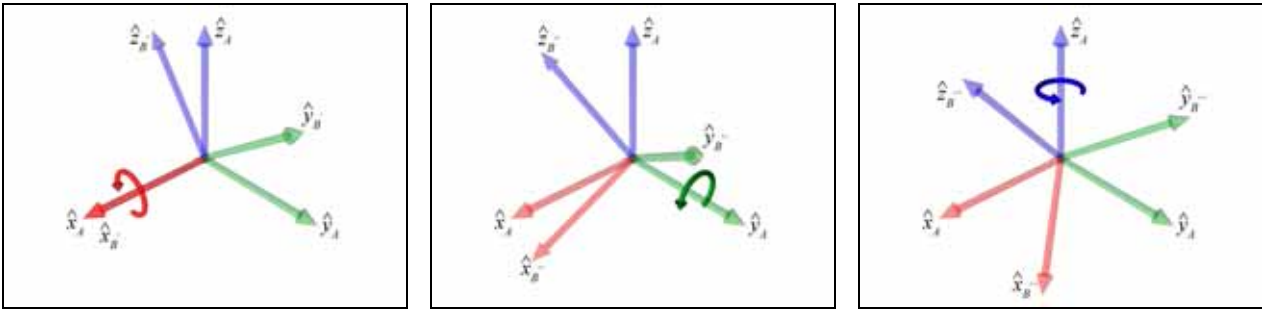
gdzie  $c_\varphi = \cos \varphi$ ,  $s_\varphi = \sin \varphi$ ,  $v_\varphi = 1 - \cos \varphi$ .

Analogicznie można zdefiniować zależność odwrotną, to znaczy na podstawie macierzy obrotu  $R$  określić oś oraz kąt wykonywanego wokół niej obrotu.

Zaletą parametryzacji typu oś-kąt jest możliwość intuicyjnego zdefiniowania pożądanej orientacji. Bezpośrednie określenie wartości macierzy obrotu jest znacznie trudniejsze w sytuacji konieczności podania orientacji dla obiektu. Jednakże parametryzacja ta nie pozwala na efektywne składanie przekształceń ani nie istnieją bezpośrednie metody interpolacji pomiędzy dwoma orientacjami.

### Parametryzacja orientacji przez obrót wokół osi ustalonego układu współrzędnych

Dowolną orientację można opisać przez złożenie 3 pojedynczych obrotów wokół osi ustalonego układu współrzędnych. Kolejność osi może być dowolna, lecz musi być stała przy wykonywaniu kolejnych obrotów. Parametryzacja ta jest wykorzystywana czasami w lotnictwie do określenia orientacji samolotu w przestrzeni. W takim przypadku kolejne obroty wiąże się odpowiednio z kątami przechylenia, pochylenia i skręcenia (ang. *Roll, Pitch, Yaw*).



Rysunek 4-2: Parametryzacja rotacji przez złożenie trzech rotacji wokół osi ustalonego układu współrzędnych.

Niech obroty będą wykonywane kolejno wokół osi  $\hat{x}_A$  o kąt  $\gamma$ ,  $\hat{y}_A$  o kąt  $\beta$ ,  $\hat{z}_A$  o kąt  $\alpha$ . Macierz obrotu będzie iloczynem macierzy obrotów wokół poszczególnych osi [18]:

$$\begin{aligned}
 {}^A_B R &= R_z(\alpha)R_y(\beta)R_x(\gamma) = \begin{bmatrix} c_\alpha & -s_\alpha & 0 \\ s_\alpha & c_\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_\beta & 0 & s_\beta \\ 0 & 1 & 0 \\ -s_\beta & 0 & c_\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\gamma & -s_\gamma \\ 0 & s_\gamma & c_\gamma \end{bmatrix} \\
 &= \begin{bmatrix} c_\alpha c_\beta & c_\alpha s_\beta s_\gamma - s_\alpha c_\gamma & c_\alpha s_\beta c_\gamma + s_\alpha s_\gamma \\ s_\alpha c_\beta & s_\alpha s_\beta s_\gamma + c_\alpha c_\gamma & s_\alpha s_\beta c_\gamma - c_\alpha s_\gamma \\ -s_\beta & c_\beta s_\gamma & c_\beta c_\gamma \end{bmatrix}
 \end{aligned} \tag{4-28}$$

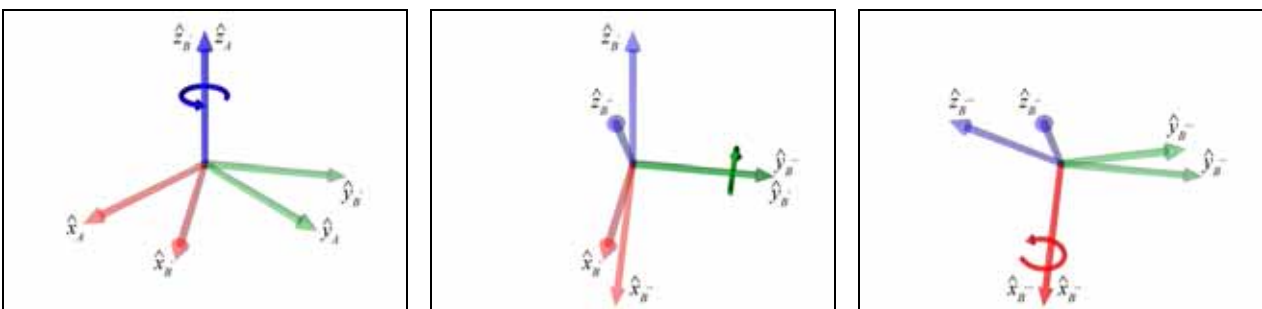
W równaniu przyjęto oznaczenia  $\cos(\alpha) = c_\alpha$ ,  $\sin(\alpha) = s_\alpha$ .

Należy zauważyć, że kolejność mnożenia jest w tym wypadku istotna (mnożenie macierzy nie jest przemienne) i definiuje przyjętą konwencję obrotów.

### Kąty Eulera

Parametryzacja rotacji przy pomocy kątów Eulera jest transformacją podobną, jak w poprzednim przypadku. Wykonywane są kolejno 3 obroty wokół wybranych osi. Różnica polega na tym, że osią obrotu jest w tym przypadku oś układu, który jest obracany.

Na rysunku 4-3 pokazano pojedyncze rotacje opisane kątami Eulera dla kolejności Z-Y-X.



Rysunek 4-3: Parametryzacja rotacji przez złożenie trzech rotacji opisanych kątami Eulera (wokół osi przekształcanego układu współrzędnych).

Również w tym przypadku macierz obrotu będzie iloczynem macierzy obrotów wykonywanych wokół poszczególnych osi układu, który jest obracany.

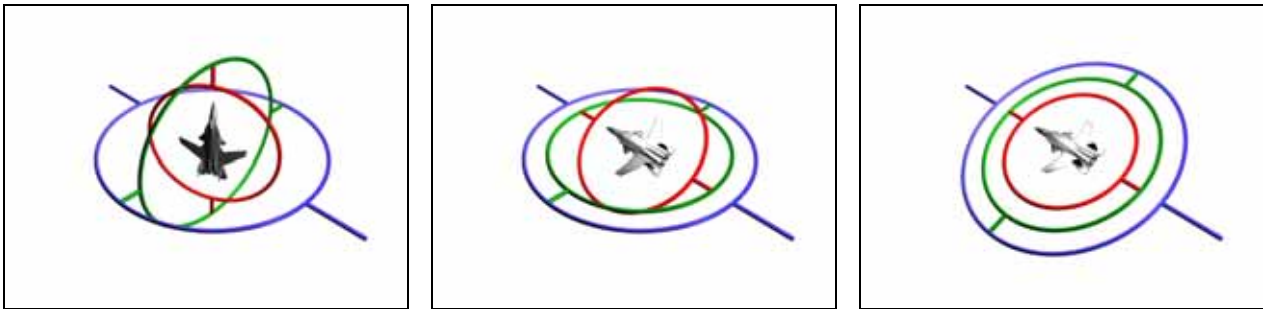
$${}^A_B R = {}^A_{B'} R(\alpha) {}^A_{B''} R(\beta) {}^A_{B'''} R(\gamma) \tag{4-29}$$

Wyznaczenie iloczynu pojedynczych macierzy obrotu prowadzi do uzyskania macierzy całkowitej, która ma identyczną postać, jak w równaniu (4 – 28). Należy jednak zauważyć, że tę samą pozycję można uzyskać wykonując obroty wokół osi ustalonego układu współrzędnych w odwrotnej kolejności niż jest to w przypadku kątów Eulera.

Istnieje wiele konwencji, które definiują kolejność wykonywania obrotów w przypadku obu parametryzacji. Macierze obrotów dla wszystkich 24 konwencji zebrane zostały i przedstawione w [18].

### 4.1.3 Osobliwości parametryzacji orientacji

Parametryzacja oparta na macierzach obrotu (co jest równoważne złożeniem obrotów w przestrzeni) jest podatna występowanie efektu utraty stopnia swobody zwanego *gimbal lock* (w wolnym tłumaczeniu oznacza to blokadę Cardana, czyli przegubu do przekazywania momentu obrotowego pomiędzy wałami). Jest to związane z występowaniem osobliwości parametryzacji orientacji, które powodują, że liczba stopni swobody układu zostaje zredukowana. Obrazowo efekt ten został przedstawiony na rysunku 4-4.



Rysunek 4-4: Występowanie efektu *gimbal lock*

Na rysunku przedstawiono obiekt, który może wykonywać obrót w każdym kierunku dzięki mechanizmowi trzech połączonych ze sobą pierścieni. Jest to praktyczna realizacja układu, którego orientacja opisywana jest przy użyciu kątów Eulera. Zakłada się, że obiekt umieszczony jest w płaszczyźnie wewnętrznego pierścienia (rysunek lewy). Po obrocie centralnego pierścienia o kąt  $\frac{\pi}{2}$  otrzymuje się konfigurację, w których utracony zostaje jeden stopień swobody. Niezależnie od tego, czy obrót dokonywany jest względem pierścienia środkowego (kolor czerwony, rysunek środkowy), czy względem pierścienia zewnętrznego (kolor niebieski, rysunek prawy) – uzyskuje się takie samo położenie obiektu.

Osobliwość występuje dla punktów 0 lub  $\pi/2$  w zależności od zdefiniowania układu początkowego.

Macierz obrotu w takim punkcie przyjmuje postać (dla  $\beta = \pi/2$ ):

$$R(\alpha, \beta, \gamma) = \begin{bmatrix} 0 & c_\alpha s_\gamma - s_\alpha c_\gamma & c_\alpha c_\gamma + s_\alpha s_\gamma \\ 0 & s_\alpha s_\gamma + c_\alpha c_\gamma & s_\alpha c_\gamma - c_\alpha s_\gamma \\ -1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & \sin(\gamma - \alpha) & \cos(\gamma - \alpha) \\ 0 & \cos(\gamma - \alpha) & \sin(\gamma - \alpha) \\ -1 & 0 & 0 \end{bmatrix} \quad (4 - 30)$$

W tym przypadku rotacja układu zależy jedynie od wielkości  $(\gamma - \alpha)$ , co oznacza, że posiada tylko jeden stopień swobody. Każdy rodzaj parametryzacji rotacji przez wielkości będące otwartym podzbiorem przestrzeni euklidesowej skazany jest na występowanie tego typu osobliwości [34]. Przykładem

parametryzacji, która jest zdefiniowana w przestrzeni nieeuklidesowej są kwaterniony. W przypadku takiej parametryzacji nie występują osobliwości typu *gimbal lock*.

#### 4.1.4 Wizualizacja danych w przestrzeni rotacji

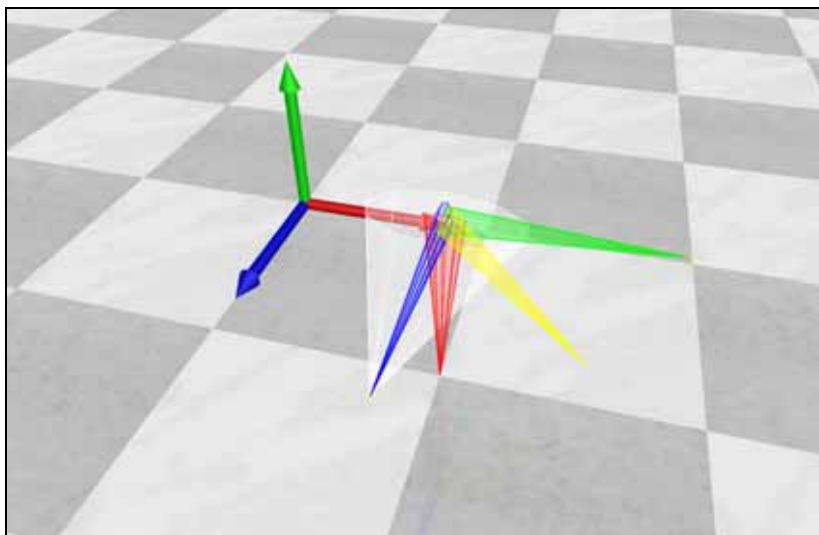
Niezależnie od przyjętego sposobu parametryzacji rotacji obiektu w przestrzeni (macierze rotacji, kąty Eulera, kwaterniony prezentowane w następnym rozdziale) w zastosowaniu praktycznym będzie występowała konieczność wizualizacji tak opisanych danych. Wizualizacja danych translacji jest za każdym razem intuicyjna, nawet jeżeli jest wykonywana dla przestrzeni trójwymiarowej. Przedstawienie trajektorii opisanych w przestrzeni rotacji trudne jest do zrealizowania ze względu na powiązania występujące pomiędzy poszczególnymi składowymi rotacji. Można przykładowo każdą składową rotacji reprezentować na oddzielnym wykresie dwuwymiarowym, jednak nie oddaje się w ten sposób całości charakteru rozpatrywanych danych.

W celu wizualizacji tego typu danych będzie stosowana parametryzacja rotacji realizowana przez złożenie 3 obrotów wokół osi ustalonego układu współrzędnych, co pozwoli na intuicyjne podawanie położenia kluczowych przy tworzeniu trajektorii. Przyjmuje się, że obroty będą podawane w konwencji X-Y-Z. Dodatkowo używana będzie orientacja osi współrzędnych pokazana na rysunku 4-5. Założenie takie zostało zrobione ze względu na zachowanie zgodności układu współrzędnych z konwencją przyjętą w używanej do implementacji biblioteki graficznej (*OpenGL*). Kolory osi są zgodne z założeniami przyjętymi w poprzednich podrozdziałach.

Na rysunku umieszczono również przykładowe rotacje reprezentowane przez położenie pojedynczego fragmentu łańcucha kinematycznego (patrz rozdział 4.5). W dalszych zastosowaniach opis taki będzie używany do określenia położenia członu manipulatora lub części szkieletu ludzkiego (w przypadku złożonych modeli hierarchicznych). Przyjmuje się, że początkowe położenie członu zgodne jest z kierunkiem i zwrotem osi Z lokalnego układu współrzędnych. Translacja względem globalnego układu współrzędnych nie jest istotna ze względu na to, że rozpatrywane są tylko rotacje. Na rysunku położenie początkowe pokazane jest kolorem białym, a wizualizacja członu jest celowo powiększona (dla lepszego odróżnienia tej pozycji od pozostałych). Na rysunku zaznaczono również 4 przykładowe położenia członu, które reprezentują odpowiednio następujące rotacje (podawane w stopniach przy konwencji X-Y-Z):

- $(90, 0, 0)$  – kolor niebieski
- $(0, 90, 0)$  – kolor zielony
- $(0, 0, 90)$  – kolor czerwony
- $(45, 90, 0)$  – kolor żółty

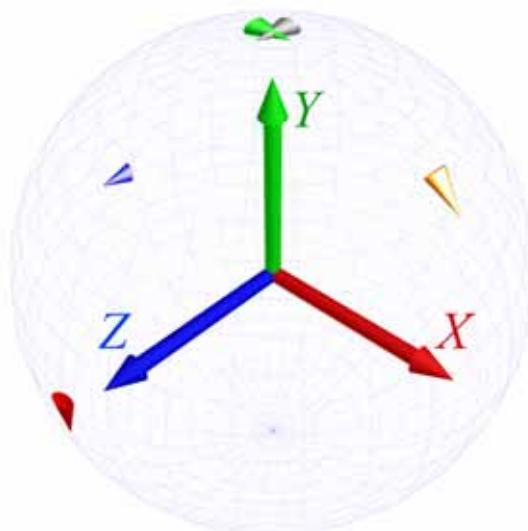
Pierwsze trzy położenia zaznaczone są kolorami odpowiadającymi kolorom osi, wokół których wykonywany jest dany obrót. Ostatnia rotacja jest tożsama z rotacją  $(0, 90, -45)$  ze względu na występowanie efektu utraty stopnia swobody dla obrotu 90 stopni wokół osi Y (patrz rozdział 4.1.3).



**Rysunek 4-5: Orientacja globalnego układu współrzędnych oraz przedstawienie podstawowych rotacji przez wizualizację położenia przykładowego członu manipulatora.**

Każdy człon został rozciągnięty 4 krotnie wzdłuż osi X lokalnego układu współrzędnych tak, aby możliwe było zaobserwowanie dowolnie wykonanego obrotu.

Zastosowana na rysunku 4-5 wizualizacja jest intuicyjna i pozwala szybko zorientować się nad charakterem określonej rotacji, a następnie nad przebiegiem trajektorii rotacji tworzącej wybraną sekwencję. Stąd tego typu przedstawienie położenia będzie stosowane w rozdziale 5.6 obrazującym własności filtracji w przestrzeni rotacji. Takie podejście będzie także stosowane przede wszystkim przy wizualizacji sekwencji rotacji dla złożonego modelu hierarchicznego (rozdział 6.5).



**Rysunek 4-6: Wizualizacja przykładowych rotacji przy pomocy sfery rotacji.**

W pracy [45] zaproponowana jest alternatywna metoda wizualizacji rotacji, która w pewnych przypadkach może pozwalać na czytelniejszą interpretację wyników. Rotacja w tym przypadku reprezentowana jest przez położenie obiektu o kształcie stożka na powierzchni trójwymiarowej sfery.



Rozpoczynając od położenia początkowego (zaznaczonego kolorem srebrnym na rysunku 4-6) wykonywane są kolejno 3 rotacje wokół ustalonego układu współrzędnych, którego początek pokrywa się ze środkiem sfery. Mając do dyspozycji 3 stopnie swobody, można w ten sposób przedstawić dowolną rotację. Zastosowanie sfery rotacji pozwala nie tylko czytelnie pokazać pojedynczą rotację, ale również całą trajektorię realizowaną w przestrzeni rotacji. Na rysunku 4-6 pokazano wizualizację przykładowych rotacji przedstawionych na rysunku 4-5. Zachowana została kolejności rotacji (X-Y-Z), konwencja osi oraz kolorystyczna.

Należy pamiętać, że przedstawiona sfera rotacji nie jest powiązana ze sferą jednostkową w przypadku kwaternionów, która będzie omawiana w następnym rozdziale. W przyjętej konwencji została zastosowana sfera w celu ułatwienia umiejscowienia w przestrzeni rotacji wybranych wartości. Dla zachowania czytelności rysunków, oznaczenia osi nie będą umieszczane, a ich rozróżnienie będzie wynikać z przyjętych kolorów. W pozostałych przypadkach kolory poszczególnych elementów nie będą więcej powiązane z kolorami przyporządkowanymi do osi. W rozdziale 6.1 pokazane jest, że sposób ten bardzo dobrze nadaje się również do przedstawienia trajektorii zdefiniowanych w przestrzeni rotacji.

## 4.2 Zastosowanie przestrzeni kwaternionów

Kwaterniony są czterowymiarowym uogólnieniem liczb zespolonych odkrytym przez Sir Williama Rowana Hamiltona w 1843. Hamilton po raz pierwszy opisał swoje odkrycie w pracy [91]. Celem Hamiltona było użycie przestrzeni liczb zespolonych do opisu rotacji wykonywanej w przestrzeni trójwymiarowej. Można pokazać, że trójwymiarowe uogólnienie liczb zespolonych nie tworzy zamkniętego zbioru względem mnożenia. Dopiero czterowymiarowe rozwinięcie liczb zespolonych tworzy nieprzemienią grupę rzędu 8 względem mnożenia.

Kwaternion można zdefiniować w różny sposób. Przez proste rozszerzenie zapisu liczb zespolonych otrzymuje się:

$$q = s + ix + jy + kz, \quad (4 - 31)$$

gdzie  $s, x, y, z \in R$ , a  $i, j, k$  są odpowiednikami jednostki urojonej w taki sposób, że są spełnione następujące zależności:

$$i^2 = j^2 = k^2 = ijk = -1, \quad (4 - 32)$$

$$ij = -ji = k \quad (4 - 33)$$

$$jk = -kj = i \quad (4 - 34)$$

$$ki = -ik = j \quad (4 - 35)$$

Zbiór kwaternionów oznacza się przez  $H$  od nazwiska jego odkrywcy. Kwaternion może być zapisany w alternatywnej postaci, gdzie część rzeczywista  $s$  jest określana jako skalar, a część urojona, jako wektor:

$$q = [s, \vec{v}] = [s, (x, y, z)], \quad (4 - 36)$$

gdzie  $s \in R$ ,  $\vec{v} \in R^3$ . Można dodatkowo określić, że liczba rzeczywista  $s \in R$  w przestrzeni kwaternionów jest określona, jako  $[s, 0]$  (zerowa część wektorowa), a wektory przestrzeni trójwymiarowej są podzbiorem kwaternionów o zerowej części skalarnej  $\{[0, \vec{v}] \mid \vec{v} \in R^3\}$ .

#### 4.2.1 Podstawowe własności kwaternionów

Wiele własności kwaternionów wynika bezpośrednio z własności liczb zespolonych. Krótko zostaną one zaprezentowane ze szczególnym uwzględnieniem różnic występujących w obu przestrzeniach ([38], [91]). We wszystkich własnościach przyjmuje się następujące oznaczenia:

$$q_1 = [s_1, \vec{v}_1] \in H, \quad (4-37)$$

$$q_2 = [s_2, \vec{v}_2] \in H \quad (4-38)$$

##### Dodawanie

$$q_1 + q_2 = [s_1, \vec{v}_1] + [s_2, \vec{v}_2] = [s_1 + s_2, \vec{v}_1 + \vec{v}_2], \quad (4-39)$$

Dodawanie kwaternionów jest zarówno łączne, jak i przemienne. Stąd wniosek, że zbiór kwaternionów tworzy Abelową grupę względem dodawania.

Odejmowanie może być analogicznie zrealizowane przez dodanie kwaternionu przeciwnego  $-q_1 = [-s_1, -\vec{v}_1]$ .

##### Mnożenie

$$q_1 q_2 = [s_1, \vec{v}_1][s_2, \vec{v}_2] = [s_1 s_2 - \vec{v}_1 \cdot \vec{v}_2, \vec{v}_1 \times \vec{v}_2 + s_1 \vec{v}_2 + s_2 \vec{v}_1], \quad (4-40)$$

gdzie „ $\cdot$ ” oznacza iloczyn wektorowy, a kropka iloczyn skalarny. Operacja mnożenia kwaternionów ma w ogólnym przypadku następujące własności:

$$q_1 q_2 \neq q_2 q_1 \quad (\text{mnożenie nie jest przemienne}) \quad (4-41)$$

$$(p q_1) q_2 = p (q_1 q_2) \quad (\text{mnożenie jest łączne}) \quad (4-42)$$

$$p (q_1 + q_2) = p q_1 + p q_2 \quad (\text{mnożenie jest rozdzielne względem dodawania}) \quad (4-43)$$

$$(q_1 + q_2) p = q_1 p + q_2 p \quad (4-44)$$

Mnożenie przez skalar odpowiada mnożeniu dwóch kwaternionów przy czym mnożnik ma część wektorową równą zero. Mnożenie przez skalar jest przemienne.

$$q_1 r = r q_1 = [r, 0][s_1, \vec{v}_1] = [r s_1, r \vec{v}_1] \quad (4-45)$$

##### Sprzężenie

$$q_1^* = [s_1, \vec{v}_1]^* = [s_1, -\vec{v}_1] \quad (4-46)$$

Sprzężenie kwaternionów ma następujące własności:

$$(q_1^*)^* = q_1 \quad (4-47)$$

$$(p q_1)^* = q_1^* p^* \quad (4-48)$$

$$(p + q_1)^* = p^* + q_1^* \quad (4-49)$$

$$q_1 q_1^* = q_1^* q_1 \quad (4-50)$$

## Norma

$$\|q_1\| = \sqrt{q_1 q_1^*} = \sqrt{s_1^2 + v_1 \cdot v_1} = \sqrt{s_1^2 + x_1^2 + y_1^2 + z_1^2} \quad (4-51)$$

Można pokazać, że norma kwaternionów spełnia zależności:

$$\|q_1^*\| = \|q_1\| \quad (4-52)$$

$$\|q_1 q_2\| = \|q_1\| \|q_2\| \quad (4-53)$$

## Odwrotność

Dla kwaternionu  $q_1$  należącego do podzbioru  $\overset{\circ}{H} = H \setminus \{[0, (0,0,0)]\}$  istnieje element odwrotny  $q_2 \in H$  określony, jako:

$$q_2 = q_1^{-1} = \frac{q_1^*}{\|q_1\|^2} \quad (4-54)$$

Element odwrotny spełnia zależność:

$$q_1 q_2 = q_1 q_1^{-1} = q_1^{-1} q_1 = I, \quad (4-55)$$

gdzie  $I$  jest elementem neutralnym zbioru  $\overset{\circ}{H}$  względem mnożenia,  $I = [1, 0]$ .

## Podsumowanie własności algebraicznych

Na podstawie powyższych własności można udowodnić następujące twierdzenia dotyczące własności algebraicznych grupy kwaternionów  $\overset{\circ}{H} = H \setminus \{[0, (0,0,0)]\}$  [19]:

- Zbiór  $\overset{\circ}{H}$  jest nie-Abelową grupą względem mnożenia kwaternionowego.
- Zbiór  $H$  jest Abelową grupą  $(H, +)$  względem dodawania kwaternionowego.
- Zbiór  $H$  jest nie-Abelowym pierścieniem  $(H, +, \cdot)$ , gdzie „+” jest dodawaniem kwaternionowym, a „ $\cdot$ ” jest mnożeniem kwaternionowym.

### 4.2.2 Parametryzacja przestrzeni rotacji przez kwaterniony

#### Kwaterniony jednostkowe

W rozważaniach rotacji szczególną rolę odgrywają kwaterniony jednostkowe – są to kwaterniony o normie 1,  $\|q\| = 1$ . Kwaterniony jednostkowe tworzą podgrupę grupy  $\overset{\circ}{H}$ , którą standardowo oznacza się przez  $H_1$ . Dla kwaternionów  $q_1, q_2$  jednostkowych zachodzi:

$$\|q_1 q_2\| = 1, \quad (4-56)$$

$$q_1^{-1} = q_1^* \quad (4-57)$$

Można pokazać, że każdy kwaternion  $q \in H_1, q = [s, \vec{v}]$  da się przedstawić w postaci

$$q = [\cos \varphi, \vec{n} \sin \varphi], \quad (4-58)$$

gdzie  $\vec{n} \in R^3$  oraz  $\varphi \in (-\pi, \pi]$ .

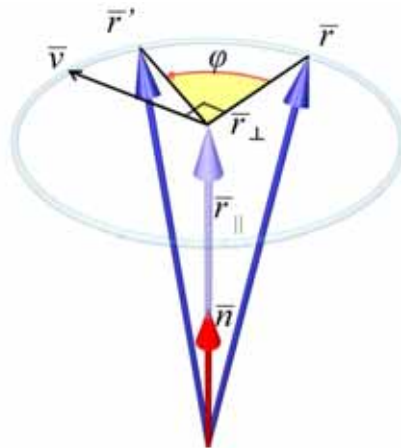
### Obrót w przestrzeni

Hamilton rozważał uogólnienie liczb zespolonych, aby opisać rotacje w przestrzeni trójwymiarowej analogicznie do obrotów na płaszczyźnie. Niech punkt  $\vec{r} = (x, y, z)$  w przestrzeni będzie przedstawiony w postaci kwaternionu  $p = [0, \vec{r}]$ . Niech kwaternion  $q$  będzie dowolnym niezerowym kwaternionem,  $q \in H_1$ ,  $q = [s, \vec{v}] = [\cos \varphi, n \sin \varphi]$ . Można pokazać, że następujące twierdzenia są prawdziwe [89]:

$$p' = [0, \vec{r}'] = qpq^{-1} = (mq)p(mq)^{-1} \text{ dla każdego } m \neq 0, m \in R, \quad (4-59)$$

$$\|\vec{r}\| = \|\vec{r}'\| \quad (4-60)$$

Przekształcenie (4-59) powoduje obrót wektora  $\vec{r}$  wokół osi  $\vec{n}$  o kąt  $2\varphi$ , w wyniku czego otrzymuje się wektor  $\vec{r}'$  (zapisany w postaci kwaternionu  $p'$ ). Jest to podstawowa właściwość, dzięki której kwaterniony mogą opisywać zmianę orientacji wektora lub układu. Aby ją udowodnić wystarczy pokazać równoważność przekształcenia dokonanego przez obrót w przestrzeni  $R^3$  z obrotem dokonanym przez iloczyn kwaternionów (dowód zaczerpnięty z pracy [19]).



Rysunek 4-7: Rotacja wektora  $\vec{r}$  wokół osi  $\vec{n}$  o kąt  $\varphi$ .

Wektor  $\vec{r}$  obracany jest względem osi  $\vec{n}$  o kąt  $\varphi$  do położenia  $\vec{r}'$ . Pokazuje to rysunek 4-7. Rozpatrywana będzie płaszczyzna, w której wektor  $\vec{r}$  będzie wykonywał obrót. Można wyznaczyć składową ortogonalną oraz równoległą do osi obrotu:

$$\vec{r}_{\parallel} = (\vec{r} \cdot \vec{n})\vec{n} \quad (4-61)$$

$$\vec{r}_{\perp} = \vec{r} - \vec{r}_{\parallel} = \vec{r} - (\vec{r} \cdot \vec{n})\vec{n} \quad (4-62)$$

Wektor  $\vec{v}$  jest jednocześnie ortogonalny do  $\vec{r}$  i  $\vec{n}$  (patrz rysunek 4-5):

$$\vec{v} = \vec{n} \times \vec{r}_{\perp} = \vec{n} \times (\vec{r} - (\vec{r} \cdot \vec{n})\vec{n}) = \vec{n} \times \vec{r} - \vec{n} \times (\vec{r} \cdot \vec{n})\vec{n} = \vec{n} \times \vec{r} \quad (4-63)$$

Z prostych zależności geometrycznych pokazanych na rysunku można wyznaczyć składową równoległą oraz składową ortogonalną wektora po obrocie wokół osi  $\vec{n}$ :

$$\vec{r}'_{\perp} = \vec{r}_{\perp} \cos \varphi + \vec{v} \sin \varphi \quad (4-64)$$

$$\vec{r}'_{\parallel} = \vec{r}_{\parallel} \quad (4-65)$$

Wektor  $\vec{r}'$  jest sumą tych składowych, co prowadzi do zależności:

$$\begin{aligned}\vec{r}' &= \vec{r}'_{\perp} + \vec{r}'_{\parallel} = \vec{r}_{\perp} + r_{\parallel} \cos \varphi + \vec{v} \sin \varphi = (\vec{r} \cdot \vec{n})\vec{n} + (r - (\vec{r} \cdot \vec{n}))\vec{n} \cos \varphi + \vec{v} \sin \varphi \\ &= \vec{r} \cos \varphi + (\vec{r} \cdot \vec{n})\vec{n} - (\vec{r} \cdot \vec{n})\vec{n} \cos \varphi + \vec{v} \sin \varphi = \vec{r} \cos \varphi + (1 - \cos \varphi)(\vec{r} \cdot \vec{n})\vec{n} + (\vec{n} \times \vec{r}) \sin \varphi\end{aligned}\quad (4-66)$$

Jednocześnie można wyznaczyć wektor  $\vec{r}'$ , jako wynik operacji kwaternionu  $q = [\cos \varphi, \vec{n} \sin \varphi]$

opisanej równaniem (4-59) (gdzie  $p = [0, \vec{r}]$ ):

$$\begin{aligned}\vec{p}' &= qpq^{-1} = qpq^* = [s, \vec{v}][0, \vec{r}][s, -\vec{v}] = [s, \vec{v}][\vec{v} \cdot \vec{r}, s\vec{r} - \vec{r} \times \vec{v}] \\ &= [s(\vec{v} \cdot \vec{r}) - \vec{v} \cdot (s\vec{r} - \vec{r} \times \vec{v}), s(s\vec{r} - \vec{r} \times \vec{v}) + (\vec{v} \cdot \vec{r})\vec{v} + \vec{v} \times (s\vec{r} - \vec{r} \times \vec{v})] \\ &= [\vec{v} \cdot (\vec{r} \times \vec{v}), s^2\vec{r} - s(\vec{r} \times \vec{v}) + (\vec{v} \cdot \vec{r})\vec{v} + s(\vec{v} \times \vec{r}) - \vec{v} \times (\vec{r} \times \vec{v})] \\ &= [0, s^2\vec{r} + (\vec{v} \cdot \vec{r})\vec{v} - 2s(\vec{r} \times \vec{v}) - \vec{v} \times (\vec{r} \times \vec{v})] = [0, s^2\vec{r} + (\vec{v} \cdot \vec{r})\vec{v} + 2s(\vec{v} \times \vec{r}) - (\vec{v} \cdot \vec{v})\vec{r} + (\vec{v} \cdot \vec{r})\vec{v}] \\ &= [0, (s^2 - \vec{v} \cdot \vec{v})\vec{r} + 2(\vec{v} \cdot \vec{r})\vec{v} + 2s(\vec{v} \times \vec{r})]\end{aligned}\quad (4-67)$$

W przekształceniu użyto następującej własności iloczynu wektorowego 3 składowych:

$$\vec{v}_1 \times (\vec{v}_2 \times \vec{v}_3) = (\vec{v}_1 \cdot \vec{v}_3)\vec{v}_2 - (\vec{v}_1 \cdot \vec{v}_2)\vec{v}_3\quad (4-68)$$

Kwaternion  $q$  został określony w ten sposób, że  $s = \cos \varphi$  oraz  $\vec{v} = \vec{n} \sin \varphi$ , przy czym jest to

kwaternion jednostkowy  $\|q\| = 1 \Rightarrow \|n\| = 1$ . Podstawiając do otrzymanej zależności otrzymuje się:

$$\begin{aligned}\vec{r}' &= (\cos^2 \varphi - \sin^2 \varphi)(\vec{n} \cdot \vec{n})\vec{r} + 2((\sin \varphi)\vec{n} \cdot \vec{r})(\sin \varphi)\vec{n} + 2 \cos \varphi((\sin \varphi)\vec{n} \times \vec{r}) \\ &= (\cos^2 \varphi - \sin^2 \varphi)\vec{r} + (2 \sin \varphi)(\vec{n} \cdot \vec{r})\vec{n} + 2 \cos \varphi \sin \varphi(\vec{n} \times \vec{r}) \\ &= \vec{r} \cos 2\varphi + (1 - \cos 2\varphi)(\vec{r} \cdot \vec{n})\vec{n} + (\vec{n} \times \vec{r}) \sin 2\varphi\end{aligned}\quad (4-69)$$

### Wniosek

Każdy obrót wektora  $\vec{r}$  w przestrzeni trójwymiarowej opisany za pomocą osi obrotu  $\vec{n}$  ( $\|\vec{n}\| = 1$ ) oraz kąta  $\varphi$  jest odpowiednikiem operacji obrotu  $q[0, \vec{r}]q^{-1}$  dokonywanego przy użyciu jednostkowego kwaternionu

$$q = \left[ \cos \frac{\varphi}{2}, \vec{n} \sin \frac{\varphi}{2} \right].$$

Mając informację o obrocie zapisaną w postaci kwaternionowej, można ją w prosty sposób przekształcić do parametryzacji oś-kąt przez następujące podstawienie:

$$\varphi = 2 \arccos(s)\quad (4-70)$$

$$n_x = \frac{x}{\sin(\varphi/2)}, n_y = \frac{y}{\sin(\varphi/2)}, n_z = \frac{z}{\sin(\varphi/2)}\quad (4-71)$$

W podobny sposób można skonstruować macierz obrotu  $R$  odpowiadającemu użyciu kwaternionu  $q$  na wektor  $\vec{r}$ . Do tego celu używa się następującej postaci macierzy [89]:

$$R = \begin{bmatrix} 1 - 2y^2 - 2z^2 & 2xy + 2sz & 2xz - 2sy \\ 2xy - 2sz & 1 - 2x^2 - 2z^2 & 2yz + 2sx \\ 2xz + 2sy & 2yz - 2sx & 1 - 2x^2 - 2y^2 \end{bmatrix}\quad (4-72)$$

## Hipersfera $S^3$

Kwaterniony jednostkowe mogą być rozpatrywane, jako wektory jednostkowe w przestrzeni  $R^4$ . Oznacza to, że wszystkie kwaterniony jednostkowe tworzą hipersferę w przestrzeni  $R^4$ , która jest oznaczana jako  $S^3$  (ponieważ każdy element przestrzeni posiada 3 stopnie swobody). Dla każdego kwaternionu  $q$  można określić jego antypodę  $-q$  (kwaternion przeciwny) leżącą na przeciwległym krańcu hipersfery  $S^3$  (nie należy mylić kwaternionu  $-q = [-s, -v]$  z kwaternionem sprzężonym  $q^* = [s, -v]$ ). Można pokazać, że kwaterniony antypodalne  $q, -q$  reprezentują tę samą rotację wektora przestrzeni  $R^3$ . Zjawisko to nazywa się symetrią antypodalną [48].

Występowanie dwóch kwaternionów dla tej samej rotacji powoduje *podwójne pokrycie* (ang. *double-covering, dual-valued representation*) przestrzeni  $SO(3)$  przez przestrzeń  $S^3$ . Można zdefiniować projekcję przestrzeni  $S^3$  na przestrzeń obrotów, które mogą być opisane przez specjalną grupę ortogonalną  $SO(3)$  (w rozdziale 4.1 pokazano, że grupa taka reprezentuje rotacje w przestrzeni  $R^3$ ). Projekcja taka będzie typu dwa do jednego, to znaczy każdy element (rotacja) z przestrzeni  $SO(3)$  odpowiada dwóm elementom przestrzeni  $S^3$ . Ma to konkretne praktyczne konsekwencje wymagające szczególnej uwagi, gdy rotacje porównywane są w przestrzeni kwaternionów jednostkowych. W praktyce problem podwójnego pokrycia rozwiązuje się przez ograniczenie dziedziny np. do obszary  $\varphi \in [0, \Pi]$ . Kwaterniony, które wykraczają poza tą dziedzinę są odpowiednio modyfikowane dzięki właściwości symetrii antypodycznej.

## Składanie obrotów

Jak to zostało pokazane, kwaterniony jednostkowe mogą reprezentować rotacje. Jedną z zalet reprezentacji kwaternionów jest możliwość prostego składania następujących po sobie rotacji. Przykładowo niech kwaternion  $q_1$  reprezentuje rotację, po której następuje rotacja opisana kwaternionem  $q_2$ . Aby otrzymać reprezentację rotacji wynikowej wystarczy wyznaczyć kwaternion będący iloczynem kolejnych kwaternionów:

$$q_3 = q_2 q_1 \quad (4 - 73)$$

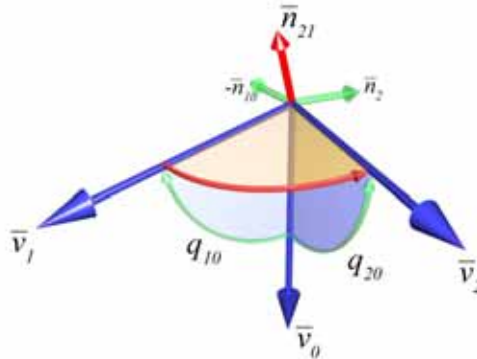
W prosty sposób można pokazać, że tak utworzony kwaternion wynikowy w prawidłowy sposób zrealizuje obrót dowolnego wektora według zależności (4 - 67) [19]:

$$\begin{aligned} q_2 (q_1 p q_1^{-1}) q_2^{-1} &= (q_2 q_1) p (q_1^{-1} q_2^{-1}) = (q_2 q_1) p (q_1^* q_2^*) = (q_2 q_1) p (q_2 q_1)^* \\ &= (q_2 q_1) p (q_2 q_1)^{-1} = q_3 p q_3^{-1} \end{aligned} \quad (4 - 74)$$

## Opis orientacji

Jak to zostało zasygnalizowane na początku tego rozdziału, należy wyraźnie rozróżnić rotację (przekształcenie, transformację wektora/przestrzeni) od orientacji (składowa opisu położenia osi lokalnego układu współrzędnych względem układu odniesienia). Do opisu orientacji można również zastosować kwaterniony. Na rysunku 4-8 pokazano przypadek zmiany orientacji układu. Dla uproszczenia rysunku zamiast wszystkich osi układu współrzędnych, przedstawiono przykładowy wektor o orientacji odniesienia

$\bar{v}_0$ . Zmiana orientacji w położeniu pierwszym opisana jest kwaternionem  $q_{10}$  (podwójny indeks  $q_{ij}$  oznacza, że rotacja jest wykonywana od położenia  $j$  do położenia  $i$ ). Analogicznie zmiana orientacji w położeniu drugim opisana jest kwaternionem  $q_{20}$ .



Rysunek 4-8: Zmiana orientacji lokalnego układu współrzędnych w opisie kwaternionowym.

Niech kwaternion  $q_{21}$  reprezentuje rotację, jaką należy wykonać z położenia pierwszego do drugiego. Na podstawie zależności (4 – 73) opisującej składanie obrotów można napisać:

$$q_{20} = q_{10}q_{21} \quad (4 – 75)$$

Stąd można wyznaczyć kwaternion  $q_{21}$ , który reprezentuje rotację, jaką trzeba wykonać, aby z orientacji  $q_{10}$  przejść do orientacji  $q_{20}$ :

$$q_{21} = q_{10}^{-1}q_{20} = q_{10}^* q_{20} \quad (4 – 76)$$

Zależność jest prawdziwa dla kwaternionów jednostkowych, dla których  $q^{-1} = q^*$ .

### 4.2.3 Przestrzeń styczna

Jednoznaczne przyporządkowanie przestrzeni  $R^3$  przestrzeni kwaternionów  $S^3$  nazywa się mapą eksponencjalną zdefiniowaną w następujący sposób [34]:

$$e^{\bar{v}} = [1, (0,0,0)] \quad \text{dla } \bar{v} = [0,0,0]$$

$$e^{\bar{v}} = \sum_{m=0}^{\infty} \left( \frac{1}{2} \tilde{v} \right)^m = \left[ \cos \frac{\varphi}{2}, \hat{v} \sin \frac{\varphi}{2} \right] \quad \text{dla } \bar{v} \neq [0,0,0], \quad (4 – 77)$$

gdzie  $\tilde{v} = [0, \bar{v}]$ ,  $\varphi = \|\bar{v}\|$ ,  $\hat{v} = \frac{\bar{v}}{\|\bar{v}\|}$ .

Mapy eksponencjalne są wykorzystywane w różnego typu przyporządkowaniach. Przykładowo mapą eksponencjalną jest przyporządkowanie przestrzeni macierzy skośnosymetrycznych do specjalnej grupy ortogonalnej  $SO(3)$  również realizowane przez nieskończony szereg. W praktyce tak określona mapa eksponencjalna wyznaczana jest za pomocą reguły Rodrigueza [18].

Istnieje odwrotne przyporządkowanie, które określa funkcję logarytmu dla kwaternionów:

$$\log q = \left[ 0, \frac{\varphi}{2} \vec{v} \right] \quad (4 - 78)$$

Funkcja logarytmu definiuje przestrzeń styczną  $T_1 S^3$  w punkcie  $[1, (0,0,0)]$ . Aby odwzorowanie było dobrze określone ogranicza się dziedzinę do  $\|\vec{v}\| < \pi$ .

Dzięki temu można sformułować definicję potęgowania kwaternionu, dla potęgi rzeczywistej  $p$ :

$$q^p = \exp(p \log q) \quad (4 - 79)$$

W przestrzeni kwaternionów odwzorowanie potęgowe ma podobne własności, jak w przestrzeni liczb rzeczywistych:

$$q^a q^b = q^{a+b} \quad (4 - 80)$$

$$(q^a)^b = q^{ab} \quad (4 - 81)$$

Odwzorowania te będą niezbędne do dalszych własności i przekształceń dokonywanych na kwaternionach.

#### 4.2.4 Interpolacja kwaternionowa

Istnieją różne metody interpolacji pomiędzy dwiema orientacjami wektora w przestrzeni. Jedną z ważniejszych zalet kwaternionów jest możliwość zastosowania interpolacji pozwalającej na uzyskanie stałej prędkości kątowej wzdłuż krzywej interpolacyjnej. Interpolacja w przestrzeni kwaternionów ma znacznie lepsze własności w porównaniu do metod interpolacji orientacji reprezentowanych przez macierze lub kąty Eulera. Istnieje bogata literatura badająca problem interpolacji dla kwaternionów. Shoemake zaproponował w [89] wykorzystanie sferyczno-liniowej interpolacji (ang. *spherical-linear interpolation = slerp*). Szczegółowe badania nad jej własnościami zostały zaprezentowane w [19]. Krótko zostaną przedstawione najistotniejsze rodzaje interpolacji, które będą używane w kolejnych rozdziałach pracy.

##### Liniowa interpolacja *lerp*

Liniowa interpolacja kwaternionów (ang. *lerp*) realizowana jest w ten sam sposób, jak odbywa się to w przestrzeni euklidesowej  $R^n$ . Niech dane będą kwaterniony opisujące dwie orientacje wektorów (lokalnych układów współrzędnych) w przestrzeni  $q_{10}, q_{20}$  w odniesieniu do orientacji referencyjnej  $\vec{v}_0$ . Zadanie interpolacji polega na znalezieniu kwaternionu  $q_p$ , który reprezentuje orientację wektora w określonym punkcie interpolacji  $p \in [0,1]$ . Dla  $p=0$  kwaternion  $q_p$  pokrywa się z  $q_{10}$ , dla  $p=1$  kwaternion  $q_p$  pokrywa się z  $q_{20}$ . Aby rozwiązać problem ekstrapolacji, należy dopuścić wartości  $p$  spoza zdefiniowanego zakresu.

Dla interpolacji liniowej kwaternion interpolowany przyjmuje postać:

$$q_p = \text{lerp}(q_1, q_2, p) = (1-p)q_1 + pq_2 \quad (4 - 82)$$



Z otrzymanej zależności wynika, że kwaternion  $q_p$  po interpolacji może nie być kwaternionem jednostkowym. Z zależności (4 – 59) wynika, że wszystkie kwaterniony postaci  $mq$ , gdzie  $m \in R, m \neq 0$  reprezentują tą samą rotację. Stąd otrzymany wynik należy znormalizować, aby kwaternion należał do sfery jednostkowej  $S^3$ . Powoduje to jednak, że interpolacja nie zachowuje stałej prędkości kątowej. Prędkość jest największa w połowie krzywej interpolacyjnej ze względu na zmniejszenie normy kwaternionu. Interpolacja liniowa jest czasami stosowana, jeżeli zachowanie prędkości kątowej nie jest wymagane. Jej zaletą jest duża szybkość działania w porównaniu do alternatywnych podejść.

### Sferyczno-liniowa interpolacja *slerp*

Podstawową wadą interpolacji *lerp* jest to, że krzywa interpolacyjna nie należy całkowicie do sfery jednostkowej  $S^3$ . Wadę tę eliminuje interpolacja sferyczno-liniowa, która jest najczęściej stosowana od jej spopularyzowania przez pracę [89].

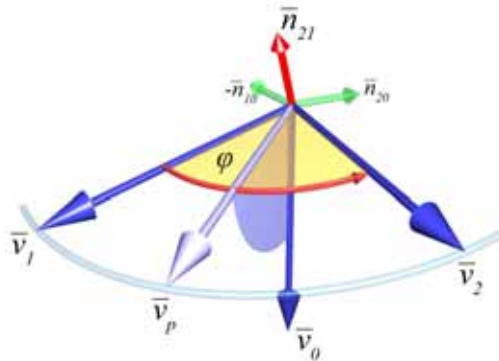
Dla oznaczeń stosowanych w poprzednim przypadku, interpolacja *slerp* określona jest następującą zależnością:

$$q_p = \text{slerp}(q_1, q_2, p) = q_1 \exp(p \log(q_1^{-1} q_2)) = q_1 (q_1^{-1} q_2)^p \quad (4 - 83)$$

Zależność (4 – 83) reprezentuje interpolację w postaci wygodnej do analizy jej własności i dalszych przekształceń. Można ją wyprowadzić na podstawie własności grupy kwaternionów. Konstruując schemat interpolacyjny na podstawie własności geometrycznych można otrzymać równoważne wyrażenie, częściej stosowane w praktycznych obliczeniach numerycznych.

$$\text{slerp}(q_1, q_2, p) = \frac{\sin((1-p)\varphi)}{\sin \varphi} q_1 + \frac{\sin(p\varphi)}{\sin \varphi} q_2 \quad (4 - 84)$$

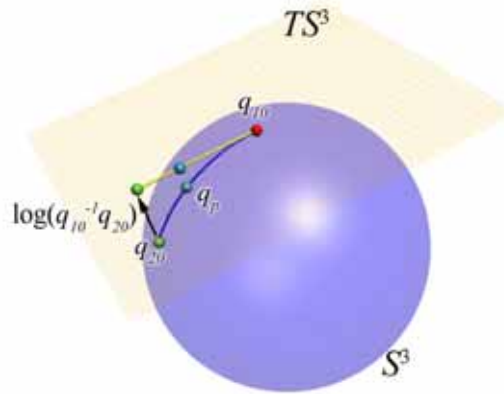
Rysunek 4-9 przedstawia przebieg interpolacji dla przykładowych początkowej i końcowej orientacji wektorów. Na rysunku zaznaczono położenie odpowiednich wektorów oraz osie obrotów opisujących rotacje pomiędzy wybranymi położeniami. Obroty reprezentowane są przez kwaterniony odpowiadające osiom obrotu  $\vec{n}$ . Wektor  $\vec{v}_p$  reprezentuje orientację układu interpolowaną opisaną kwaternionem  $q_p$ .



Rysunek 4-9: Interpolacja orientacji wektorów przy użyciu kwaternionów.

Interpretując równanie (4 – 84) na podstawie rysunku 4-9 widoczne jest, że interpolacja powoduje podział kąta  $\varphi$  w zależności od punktu interpolacji  $p$ . Kąt ten wyznacza rotację, jaką trzeba wykonać wokół osi  $\vec{n}_{21}$ , aby orientację wektora  $\vec{v}_1$  przekształcić w orientację  $\vec{v}_2$ .

Podobna interpretacja dotyczy zależności (4 – 83). W tym przypadku cała operacja odbywa się w przestrzeni kwaternionów  $S^3$  oraz w przestrzeni stycznej  $TS^3$ . Ilustruje to rysunek 4-10.



**Rysunek 4-10: Interpolacja orientacji dla kwaternionów w przestrzeniach  $S^3$ ,  $TS^3$  (ilustracja własna na podstawie [48]).**

Wyrażenie  $\log(q_{10}^{-1}q_{20})$  w równaniu (4 – 83) oznacza odwzorowanie kwaternionu  $q_{20}$  w przestrzeni stycznej w punkcie  $q_{10}$  do hipersfery  $S^3$ . Przestrzeń styczna do  $S^3$  jest trójwymiarową przestrzenią wektorową  $R^3$ . Interpolacja pomiędzy odwzorowaniem kwaternionów  $q_{10}$  oraz  $q_{20}$  jest interpolacją wzdłuż linii prostej wyznaczonej w przestrzeni ze stałą prędkością liniową. Stała prędkość liniowa w przestrzeni stycznej jest odwzorowywana przez mapę eksponencjalną przez stałą prędkość kątową w przestrzeni kwaternionów jednostkowych. Wynik interpolacji jest otrzymywany przez złożenie rotacji  $q_{10}(q_{10}^{-1}q_{20})^p$ , gdzie  $(q_{10}^{-1}q_{20})^p$  jest rotacją cząstkową pomiędzy orientacjami reprezentowanymi przez  $q_{10}$  oraz  $q_{20}$ .

Aby sprawdzić orientacje początkową i końcową do równania (4 – 83) podstawia się brzegowe wartości współczynnika  $p$ :

$$slerp(q_1, q_2, 0) = q_1 \exp(0 \log(q_1^{-1}q_2)) = q_1 \quad (4 - 85)$$

$$slerp(q_1, q_2, 1) = q_1 \exp(1 \log(q_1^{-1}q_2)) = q_1 q_1^{-1} q_2 = q_2 \quad (4 - 86)$$

Otrzymane orientacje zgodne są z założeniami interpolacji. Dla zerowego współczynnika  $p$  otrzymuje się kwaternion początkowy. Dla współczynnika  $p=1$  otrzymuje się kwaternion końcowy.

Jakkolwiek mnożenie kwaternionów nie jest przemienne, to tę samą interpolację można wyrazić zamieniając miejscami kwaternion początkowy i końcowy:

$$slerp(q_1, q_2, p) = slerp(q_2, q_1, 1 - p) \quad (4 - 87)$$

Podobna zależność występuje w przypadku standardowej interpolacji w przestrzeni  $R^n$ . Bazując na własnościach kwaternionów wyrażenie na interpolację można przedstawić w następujących równoważnych postaciach:

$$slerp(q_1, q_2, p) = q_1 (q_1^{-1} q_2)^p = (q_1 q_2^{-1})^{1-p} q_2 = (q_2 q_1^{-1})^p q_1 = q_2 (q_2^{-1} q_1)^{1-p} \quad (4-88)$$

Można pokazać, że interpolacja typu *slerp* posiada szereg, bardzo istotnych z punktu widzenia praktycznego, własności (dowód znajduje się w pracy [19]).

### Własności interpolacji *slerp*

- Wynik interpolacji należy do hipersfery jednostkowej  $S^3$ .
- Interpolacja przebiega ze stałą prędkością kątową wzdłuż całej krzywej interpolacyjnej.
- Interpolacja przebiega wzdłuż linii geodezyjnej wyznaczonej przez kwaternion początkowy i końcowy.
- Krzywa interpolacyjna tworzy najkrótszy łuk pomiędzy kwaternionami należący do linii geodezyjnej.

Ostatnia własność nie zapewnia jednak, że przeprowadzona interpolacja będzie optymalna z punktu widzenia zmiany orientacji układów. Jak to zostało pokazane wcześniej, każda orientacja jest reprezentowana przez dwa kwaterniony:  $q$  oraz  $-q$ . Interpolacja *slerp* wyznacza najkrótszą ścieżkę pomiędzy dwoma kwaternionami. Jednak interpolacja  $slerp(q_1, -q_2, p)$  może wynikowo dać krótszą ścieżkę niż interpolacja  $slerp(q_1, q_2, p)$ . Aby uzyskać optymalną krzywą interpolacyjną należy na początku sprawdzić, czy kwaterniony początkowy i końcowy znajdują się po tej samej stronie hipersfery.

### Złożone metody interpolacji

Przedstawione metody interpolacji biorą pod uwagę tylko dwa kwaterniony – początkowy i końcowy, między którymi dokonywana jest interpolacji. Podobnie, jak jest to realizowane w przestrzeni  $R^n$ , również w przypadku kwaternionów możliwe jest skonstruowanie odpowiedników złożonych krzywych interpolacyjnych. Shoemake w 1987 roku zaproponował schemat będący odpowiednikiem krzywej Béziera, który został nazwany *squad* (ang. *spherical and quadrangle interpolation*). Oryginalna praca, w której metoda jest prezentowana nie jest dostępna, jednak konstrukcja takiej krzywej oraz własności dotyczące jej różniczkowalności zostały zaprezentowane w [19].

Do interpolacji typu *squad* potrzebne są kwaternion początkowy i końcowy oraz dwa dodatkowe kwaterniony służące do parametryzacji krzywej. Wyrażenie służące do konstrukcji takiej interpolacji jest podobne do wyrażenia na krzywą Béziera [19]:

$$lin(x_0, x_1, p) = x_0(1-p) + x_1 p \quad (4-89)$$

$$Bézier(P_1, P_2, B_1, A_2, p) = lin(lin(P_1, P_2, p), lin(B_1, A_2, p), 2p(1-p)), \quad (4-90)$$

gdzie  $P_1, P_2$  – punkt początkowy i końcowy,  $B_1, A_2$  – dodatkowe punkty kontrolne.

Dla kwaternionów zamiast liniowej interpolacji stosuje się interpolację typu *slerp*, co prowadzi do następującego wyrażenie:

$$squad(q_i, q_{i+1}, s_i, s_{i+1}, p) = slerp(slerp(q_i, q_{i+1}, p), slerp(s_i, s_{i+1}, p), 2p(1-p)) \quad (4-91)$$

$$s_i = q_i \exp\left(-\frac{1}{4}(\log(q_i^{-1} q_{i+1}) + \log(q_i^{-1} q_{i-1}))\right) \quad (4-92)$$

W tej samej pracy proponowane są inne metody oparte na minimalizacji energii krzywej. W wielu pracach pojawiają się nowe metody pozwalające na uzyskanie kwaternionowych krzywych interpolacyjnych o zadanych własnościach ([7], [48], [53]). W pracy [88] pokazane jest, w jaki sposób można uzyskać całą rodzinę krzywych interpolacyjnych opierając się na geometrycznych własnościach przestrzeni  $S^3$ . Praca [54] proponuje schemat konstrukcji krzywych w przestrzeni kwaternionów, które zachowują własności algebraiczne krzywych sklejaných w przestrzeni  $R^n$ .

#### 4.2.5 Miary odległości w przestrzeni $S^3$

W twierdzeniu Eulera rotacja jest reprezentowana przez oś oraz kąt obrotu wokół niej. Intuicyjnie można więc porównywać dwie orientacji przez kąt obrotu, jaki należy wykonać, aby orientacje były ze sobą zgodne. Kąt jest miarą obrotów dla grupy rotacji niezależnie od przyjętej reprezentacji – w przypadku reprezentacji typu “oś-kąt”, parametr jest dostępny bez dalszych przekształceń. W innych przypadkach należy go odpowiednio wyznaczyć.

Bazując na tym podejściu możliwe jest określenie analogicznej miary odległości w przestrzeni kwaternionów. Odległością będzie w tym przypadku długość najkrótszej krzywej zawartej w hipersferze  $S^3$  łączącej dwa kwaterniony jednostkowe  $q_1, q_2$ . Kwaternion  $q_d$  reprezentujący rotację, jaką trzeba wykonać, by orientację układu  $q_1$  przekształcić w  $q_2$  wyraża się zależnością:

$$q_d = q_1^{-1} q_2 = q_1^* q_2 \quad (4 - 93)$$

Kwaternion reprezentuje dwie informacje: o odległości katowej rozumianej w ścisłym sensie miary odległości w przestrzeni orientacji oraz o kierunku wykonywania rotacji. Pierwsza informacja spełnia wszystkie warunki, jakie narzucone są na metrykę w przestrzeni. Druga takich warunków nie spełnia ze względu na nieprzemienność mnożenia kwaternionów.

Jak to zostało pokazane w rozdziale 4.2.3 funkcja logarytmiczna dla kwaternionów jednoznacznie odwzorowuje (ograniczoną) przestrzeń kwaternionów i przestrzeń styczną. Logarytm wyrażenia (4 – 93) określa punkt  $q_2$  w przestrzeni stycznej do  $S^3$  w punkcie  $q_1$ . Norma wektora w przestrzeni  $TS^3$  będzie więc odwzorowaniem odległości pomiędzy kwaternionami  $q_1$  i  $q_2$  położonych na hipersferze. Norma ta określa długość krzywej geodezyjnej, w której zawarte są oba kwaterniony (podobny problem rozpatrywany był w przypadku interpolacji sferyczno liniowej).

$$d(q_1, q_2) = 2 \|\log q_d\| = 2 \|\log(q_1^* q_2)\| \quad (4 - 94)$$

Na podstawie zależności geometrycznych można wyznaczyć wyrażenie równoważne:

$$d(q_1, q_2) = 2 \arccos(\operatorname{Re}(q_d)) = 2 \arccos(\operatorname{Re}(q_1^* q_2)) \quad (4 - 95)$$

Oba te wyrażenie określają tę samą wielkość. Niech kwaternion opisany będzie wyrażeniem

$$q_d = \left[ \cos \frac{\varphi}{2}, \vec{v}_d \sin \frac{\varphi}{2} \right]. \text{ Otrzymuje się następujące wyrażenia:}$$

$$\begin{aligned} 2\|\log q_d\| &= 2\left\|\log\left(\left[\cos\frac{\varphi}{2}, \vec{v}_d \sin\frac{\varphi}{2}\right]\right)\right\| = 2\left\|\left[0, \frac{\varphi}{2}\vec{v}_d\right]\right\| = 2\left\|\frac{\varphi}{2}(x_d, y_d, z_d)\right\| \\ &= 2\frac{\varphi}{2}\sqrt{x_d^2 + y_d^2 + z_d^2} = \varphi \end{aligned} \quad (4-96)$$

W wyprowadzeniu skorzystano z własności, że oś obrotu jest wektorem jednostkowym  $\|\vec{v}_d\| = 1$ .

Wyrażenie (4-95) przekształca się w następujący sposób:

$$2\arccos(\operatorname{Re}(q_d)) = 2\arccos\left(\cos\frac{\varphi}{2}\right) = \varphi \quad (4-97)$$

W obu przypadkach otrzymuje się ten sam wynik określający odległość pomiędzy kwaternionami, jako kąt obrotu, który trzeba wykonać, aby kwaterniony reprezentowały tę samą orientację układu.

#### 4.2.6 Odpowiedniki wielkości statystycznych w przestrzeni kwaternionów

W algorytmach analizy i przetwarzania trajektorii w przestrzeni rotacji, niezbędna jest możliwość wyznaczenia odpowiednika wybranych wielkości statystycznych. Ze względu na specyfikę przestrzeni kwaternionów, nie można bezpośrednio stosować zależności znanych z klasycznej statystyki. W literaturze istnieją propozycje podejścia uwzględniające własności kwaternionów jednostkowych. W niniejszym podrozdziale zostaną zaprezentowane wybrane odpowiedniki, które będą używane w kolejnych częściach pracy.

##### Odpowiednik estymatora wartości średniej

Standardowe sformułowanie średniej arytmetycznej dane jest wzorem:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (4-98)$$

Przestrzeń kwaternionów może być rozważana (pod pewnymi założeniami) jako przestrzeń wektorowa. Jednak bezpośrednio wyznaczenie wartości średniej wyrażonej wzorem (4-98) nie jest właściwe z punktu widzenia interpretacji rotacyjnej dla kwaternionów. Wynikają z niego następujące problemy:

- Wyznaczony kwaternion średni nie będzie w ogólności należał do hipersfery jednostkowej  $S^3$ .
- W ogólności symetria antypodalna nie jest zachowana.
- W szczególności średnia arytmetyczna z kwaternionów antypodalnych  $q$  oraz  $-q$  wynosi  $[0, (0,0,0)]$  - jest to wynik, którego nie można znormalizować i interpretować z punktu widzenia rotacji.

Bazując na mierze odległości zdefiniowanej dla przestrzeni  $S^3$  (rozdział 4.2.5) można wyprowadzić następującą zależność na wartość średnią kwaternionu:

$$\bar{\varphi}_n = \frac{1}{n} \sum_{i=1}^n \|\log q_i\| \quad (4-99)$$

W ten sposób można wyznaczyć średni kąt rotacji, jaki jest reprezentowany przez zbiór kwaternionów. Nie jest to jednak pełna informacja reprezentująca średnią rotację, gdyż brakuje składowej wyznaczającej jej kierunek.

Propozycja rozwiązania tego problemu zawarta jest w pracy [48]. Zamiast wyznaczania średniej arytmetycznej, autor proponuje znalezienie wartości modalnej (dominandy) w postaci kwaternionu należącego do badanej populacji, który minimalizuje kwadrat odległości do wszystkim pozostałych elementów.

$$\bar{q} = \arg \min_p \sum_{i=1}^n d_{|S^3|}(p, q_i)^2, \quad (4 - 100)$$

gdzie funkcja odległości postaci  $d_{|S^3|}$  uwzględnia symetrię antypodalną:

$$d_{|S^3|}(q_1, q_2) = 2 \min_{q=q_1, -q_1} dist(q, q_2). \quad (4 - 101)$$

Podejście to jest dobrym odpowiednikiem wartości średniej rozumianej jako dominanta w rozkładzie. Jednak zakres jego stosowalności jest ograniczony do względnie dużych populacji. Dla grup kwaternionów o małej liczności tak wyznaczona wartość średnia może być obciążona znacznym błędem, znacznie większym niż w przypadku innych metod wyznaczających średnią arytmetyczną.

Bezpośrednio przekształcając zależność (4 – 98) do postaci rekurencyjnej otrzymuje się:

$$\bar{x}_0 = 0 \quad (4 - 102)$$

$$\bar{x}_n = \frac{(n-1)\bar{x}_{n-1} + x_n}{n} = \left(1 - \frac{1}{n}\right)\bar{x}_{n-1} + \frac{1}{n}x_n \quad (4 - 103)$$

W każdym kroku wyznaczana jest wartość średnia z populacji pomniejszonej o jeden element. Dzięki temu wykonywane jest  $n$  kroków liniowej interpolacji pomiędzy wartością średnią wyznaczoną w kroku poprzednim, a kolejnym elementem z grupy. Jak to zostało pokazane w rozdziale 4.2.4 liniowa interpolacja w dziedzinie kwaternionów nie ma dobrych własności (wynik nie znajduje się na hipersferze jednostkowej, krzywa interpolacyjna nie ma stałej prędkości kątowej). Dlatego w wielu przypadkach można zamiast liniowej zastosować interpolację sferyczno-liniową, która zachowuje własności przestrzeni kwaternionów jednostkowych. Wyrażenie (4 – 103) przyjmuje wtedy postać:

$$\bar{q}_0 = [1, (0,0,0)] \quad (4 - 104)$$

$$\bar{q}_n = \bar{q}_{n-1} (\bar{q}_{n-1}^{-1} q_n)^{\frac{1}{n}} \quad (4 - 105)$$

Otrzymana zależność rekurencyjna nie jest jednak wartością średnią dla zbioru kwaternionów. Jak to zostało pokazane w pracy [11] nie jest możliwe dokładne wyznaczenie wartości średniej przez wyznaczenie serii złożenia pojedynczych uśrednień. Niemniej jednak zależność ta jest zgrubnym przybliżeniem kwaternionu średniego, które dla pewnych zastosowań może być przybliżeniem wystarczającym. Im większe jest zróżnicowanie rozpatrywanej grupy, tym większy błąd zostanie wprowadzony do obliczeń. Jednak jest on niewielki w przypadku grupy pojedynczych elementów, które nie różnią się znacząco wartością. Wyrażenie to posiada następujące własności:

- Otrzymany kwaternion reprezentuje przybliżoną rotację, która jest interpretowana jako obrót o średni kąt  $\bar{\varphi}_n$  wokół osi  $\bar{n}$  definiującej średni kierunek obrotu wykonywanej w grupie kwaternionów  $(q_1, \dots, q_n)$ .
- W każdym kroku uśredniania otrzymany wynik należy do hipersfery jednostkowej  $S^3$  – nie ma konieczności normalizacji.
- Zależność rekurencyjna powoduje, że nie ma konieczności przechowywania wszystkich danych w pamięci – mogą one być dodawane pojedynczo w kolejnych krokach
- Otrzymuje się zależność, która jest zgrubnym oszacowaniem średniej, jednak jest realizowana bardzo szybko - złożoność obliczeniowa takiego procesu jest liniowa  $O(n)$ .

Pomimo błędu wynikającego z odmiennych własności przestrzeni kwaternionów, wyrażenie (4 – 105) może być lepszym estymatorem wartości średniej od metody opartej na wyznaczeniu dominanty. Otrzymana wartość ta będzie tym lepiej estymowała średnią, im mniejsze oraz o mniejszym zróżnicowaniu zbiory są rozważane.

W cytowanej pracy [48] prezentowany jest algorytm częściowo podobny do proponowanej metody wyznaczenia średniej kwaternionowej. Algorytm nazwany *slime* (ang. *spherical linear interpolation of multiple examples*) pozwala na wyznaczenie wartości interpolowanej na podstawie wielu elementów grupy. Algorytm wymaga jednak wyznaczenia kwaternionu referencyjnego, który będzie reprezentował punkt styczności z przestrzenią styczną. Operacja ważonego łączenia kwaternionów dokonywana jest we wspólnej przestrzeni stycznej. Może to być interpretowane jako średnia ważona elementów. Jednak wynik takiej operacji będzie mocno zależał od wyboru kwaternionu referencyjnego. W prezentowanym powyżej algorytmie wyznaczenie średniej realizowane jest przez ciąg sferyczno-liniowych interpolacji, dzięki czemu wynik nie zależy do wybranego punktu początkowego, ani nie wymaga wyznaczenia referencyjnego punktu do przestrzeni stycznej.

W pracy [13] został przedstawiony optymalizacyjny algorytm do wyznaczenia odpowiednika średniej dla kwaternionów (w ogólności średniej sferycznej). Jest to jeden z najdokładniejszych algorytmów istniejących obecnie w literaturze. Algorytm wymaga wyznaczenia dowolną metodą początkowej wartości średniej  $\bar{q}_0$ . Następnie wykonywany jest iteracyjny krok poprawy określony jako:

$$\bar{q}_n = \exp\left(\bar{q}_{n-1} \sum_{i=1}^n \left(\frac{1}{n} \log q_i\right)\right) \quad (4 - 106)$$

W każdym kolejnym kroku wykonywane jest uśrednianie wszystkich punktów w przestrzeni stycznej do sfery jednostkowej w punkcie  $\bar{q}_{n-1}$  będącym estymatorem wartości średniej wyznaczonym w poprzedniej iteracji. Algorytm powtarza się aż do momentu, w którym spełniony jest warunek stopu określony, jako

$$\left\| \frac{1}{n} \log q_i \right\| < \varepsilon, \quad (4 - 107)$$

gdzie  $\varepsilon$  jest określoną małą wartością.

Algorytm ten znajduje wartość średnią w procedurze iteracyjnej, w której estymacja średniej jest za każdym razem poprawiana. W praktyce miał on szybką zbieżność i również będzie używany w dalszej części pracy, gdy wystąpi konieczność wyznaczenia wartości średniej z dużej liczbie grupy kwaternionów.

### Miara zróżnicowania rozkładu – średnie odchylenie bezwzględne

Wartość średnia jest przykładem statystycznej miary położenia rozkładu. Często występuje konieczność określenia miary zróżnicowania rozkładu, które może być wyznaczona np. jako wariancja. Bezpośrednie zastosowanie podejścia przedstawionego przy wyznaczaniu kwaternionowej wartości średniej nie jest możliwe w przypadku wyliczania wariancji ze względu sposób określenia tej miary. W wyrażeniu na wariancję występuje kwadrat odległości parametrów od wartości średniej, czego nie da się zinterpretować w odniesieniu do kwaternionów analogicznie, jakby to dotyczyło liczb rzeczywistych. Można natomiast estymować średnie odchylenie bezwzględne MD (ang. *mean deviation*). Obie miary poprawnie określają miarę zróżnicowania rozkładu na podstawie populacji. Historycznie wariancja została przyjęta, jako podstawowy sposób określania takiej miary, co spowodowało, że cała teoria prawdopodobieństwa bazuje przede wszystkim na niej. Jednak średnie odchylenie bezwzględne ma wiele zalet, które z reguły nie są wykorzystywane. Miara MD lepiej opisuje zróżnicowanie układu w przypadku rzeczywistym, gdy występować mogą drobne, jak i grube błędy pomiaru. Wariancja natomiast w lepszy sposób opisuje przypadek idealny, gdy błędy grube nie są brane pod uwagę. Postać MD jest również prostsza pojęciowo, jednak trudniejsza do dalszych przekształceń algebraicznych. Szczegółowe porównanie obu miar znajduje się w [33].

Standardowo miara MD zdefiniowana dla populacji  $n$ -elementowej jest jako:

$$x = \frac{1}{n} \sum_{i=1}^n |x_i - \bar{x}_n|, \quad (4 - 108)$$

gdzie  $\bar{x}_n$  oznacza średnią arytmetyczną populacji.

W przypadku kwaternionów proponuje się następującą metodę wyznaczania odpowiednika miary MD. Niech kwaternion  $\bar{q}$  oznacza kwaternionową wartość średnią populacji wyznaczoną przy pomocy iteracyjnego algorytmu poprawy określonego zależnością (4 – 106). Definiuje się kwaternion oznaczający odchylenie elementu  $q_i$  od wartości średniej (na podstawie zależności (4 – 76)) jako:

$$\tilde{q}_i = \bar{q}^{-1} q_i \quad (4 - 109)$$

Średnie odchylenie bezwzględne  $q_n^{MD}$  dla populacji  $n$  kwaternionów może być wyznaczone przy użyciu analogicznej zależności iteracyjnej:

$$\bar{q}_n^{MD} = \exp\left(\bar{q}_{n-1}^{MD} \sum_{i=1}^n \left(\frac{1}{n} \log \tilde{q}_i\right)\right) \quad (4 - 110)$$

Wykorzystywany jest algorytm poprawy wyznaczonej wartości w odniesieniu do zbioru kwaternionów reprezentujących kwaternionowe odchylenie elementu od wartości średniej. Zarówno warunek zakończenia procesu, jak i jego przebieg są takie same, jak w przypadku wyznaczania odpowiednika wartości średniej dla kwaternionów. Ten sam algorytm jest więc wykorzystywany dwukrotnie



dla dwóch zbiorów danych. Otrzymana miara może być interpretowana, jako rotacja, która jest najbardziej zbliżona do wszystkich rotacji wykonywanych z orientacji opisanej kwaternionem  $\bar{q}$  do poszczególnych orientacji reprezentowanych przez elementy grupy kwaternionów  $q_i$ . Tak określona miara daje informacje zarówno o kącie, jak i kierunku zróżnicowania rozkładu.

### Miara zróżnicowania rozkładu – wariancja

Jak to zostało pokazane w poprzednim punkcie, nie jest możliwe stworzenie miary odpowiadającej wariancji rozkładu, która będzie miała interpretację w przestrzeni orientacji (kierunek i kąt obrotu). Jednakże można taką informację wyznaczyć biorąc pod uwagę miarę odległości zdefiniowaną dla przestrzeni  $S^3$ , która jest wielkością skalarną. Klasyczne sformułowanie wariancji jest określone przez zależności (w zależności od używanej wersji jest to estymatora obciążony  $\sigma_n^2$  lub nieobciążony  $\sigma_{n-1}^2$ ):

$$\sigma_n^2 = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (4-111)$$

$$\sigma_{n-1}^2 = \frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2 \quad (4-112)$$

Niech kwaternion  $\bar{q}$  będzie estymatorem średniej kwaternionowej. Można zdefiniować następującą skalarną miarę zróżnicowania rozkładu biorąc pod uwagę funkcję odległości w przestrzeni kwaternionów:

$$\sigma_{qn}^2 = \frac{1}{n} \sum_{i=1}^n \left\| \log(\bar{q}^{-1} q_i) \right\|^2 \quad (4-113)$$

W tym przypadku nie ma konieczności używania zależności rekurencyjnej. Wariancja jest określona jako liczba rzeczywista. Otrzymaną miarę można interpretować, jako średnie kątowe odchylenie standardowe od wartości średniej. Wartość średnia jest reprezentowana w postaci kwaternionowej obliczona za pomocą wcześniej wyprowadzonych zależności. Wyznaczenie wartości średniej w przestrzeni stycznej pozwala na otrzymanie skalarnej wielkości, która może być dalej w dowolny sposób przetwarzania. W ten sam sposób możliwe jest otrzymanie nieobciążonego kwaternionowego estymatora wariancji.

Wybór wariancji lub miary MD do ilościowego określenia miary rozbieżności rozkładu będzie zależało od konkretnego zastosowania. Do wyznaczenia skalarnej rozbieżności w kątach rotacji wystarczająca jest wariancja, której wyznaczenie jest mniej skomplikowane obliczeniowo. Kwaternionowy odpowiednik średniego odchylenia bezwzględnego daje informacje zarówno o kącie, jak i kierunku, w jakim odchylenie następuje.

W istniejącej literaturze nie znaleziono podejścia tożsamego z przedstawioną propozycją odpowiedników wielkości statystycznych. Najbardziej zbliżona jest praca [48], która prezentuje wiele aspektów związanych ze statystyką w przestrzeni  $S^3$ . Zaprezentowane są między innymi rozkłady prawdopodobieństwa realizowane dla kwaternionów.

#### 4.2.7 Generacja wartości pseudolosowych w przestrzeni kwaternionów

Problem generacji wartości o określonym rozkładzie statystycznym będzie w dalszej części pracy rozpatrywany w związku z wprowadzaniem do trajektorii kwaternionowej losowego sygnału zakłócającego. Generacja losowego położenia na hipersferze  $S^3$  jest rozpatrywana w kilku pozycjach literaturowych. Dla przypadku uproszonego, w którym rozpatrywane jest losowe próbkowanie dla trójkątów położone na sferze trójwymiarowej, został zaproponowany algorytm w pracy [3]. W pracy [90] przedstawionych zostało kilka metod pozwalających na generację rotacji o rozkładzie jednostajnym reprezentowanych przez kwaterniony jednostkowe. W głównej mierze przedstawione metody opierają się na właściwościach grupowych przestrzeni rotacji, które pozwalają na niezależną losową generację poszczególnych składowych obrotu reprezentowanego przez kwaternion. Podobny schemat postępowania dla macierzy rotacji prezentowany jest w pracy [2]. Obie prace są szczególnym zastosowaniem (związanym także z adaptacją do przestrzeni kwaternionów) ogólnego podejścia generacji losowych macierzy przedstawionego w pracy [22].

W dalszej części pracy generacja pseudolosowych kwaternionów jest przede wszystkim używana do wprowadzenia sygnału zakłócającego do badanych trajektorii (rozdział 6). Zakładając, że kwaternion  $q_i$  reprezentuje punkt, do którego wprowadzone będzie zakłócenie o określonym rozkładzie, stosowany będzie następujący algorytm zaadaptowany z pracy [48]. W cytowanej pracy algorytm generacji jest bardziej rozbudowany, gdyż uwzględnia macierz korelacji pomiędzy poszczególnymi składowymi rotacji. W dalszym zastosowaniu celowe jest użycie metody, w której poszczególne składowe losowej rotacji są niezależnymi zmiennymi losowymi o tym samym rozkładzie. Stąd stosuje się pojedyncze parametry określające rozkład w całej procedurze wprowadzania losowego zakłócenia.

---

#### Algorytm 3: Wprowadzenie losowego zakłócenia dla kwaternionów

Parametry:  $q_i$  – kwaternion wejściowy  
 $\sigma$  - odchylenie standardowe zakłócenia o rozkładzie normalnym  
 $[-k, k]$  - przedział dla zakłócenia o rozkładzie jednostajnym

**krok 1:** Wyznacz  $q_t = \log q_i$

**krok 2:** Wyznacz wektor niezależnych zmiennych losowych  $\vec{r}$ , gdzie  $r_i \leftarrow N(0, \sigma)$  lub

$r_i \leftarrow U[-k, k]$  (w zależności od wybranego rozkładu)

**krok 3:** Podstaw  $q_i' = \exp(q_t + \vec{r})$

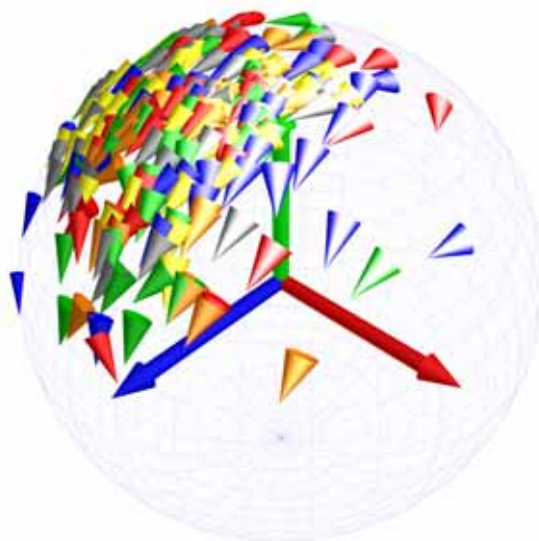
---

Przedstawiona metoda realizuje wprowadzenie losowego sygnału zakłócającego przez jego generację w przestrzeni stycznej do hipersfery jednostkowej w punkcie  $q_i$ . Dla każdego kierunku w przestrzeni stycznej generowana jest próbka pochodząca z identycznego, niezależnego rozkładu. Przyjmuje się, że wartość średnia używanego rozkładu wynosi za każdym razem 0. Otrzymana próbka modyfikuje kwaternion początkowy  $q_i$  przedstawiony w przestrzeni stycznej jako  $q_t$ . Ze względu na to, że rozpatrywane są kwaterniony jednostkowe, mają one postać trójwymiarowych wektorów w przestrzeni stycznej. Stąd wektor

zmiennych losowych  $r$  ma trzy składowe. Po dokonaniu operacji modyfikacji punktu w przestrzeni stycznej wyznacza się jego odpowiednik w przestrzeni kwaternionów przez zastosowanie mapy eksponencjalnej (krok 3, rozdział 4.2.3).

W dalszej części pracy przedstawiony algorytm będzie używany do generacji sygnału zakłócającego dla określonej trajektorii testowej. Zakłada się, że badana trajektoria jest wolna od zakłóceń, a jedynym sygnałem zakłócającym będzie dodana powyższa składowa losowa. Rozpatrywane będą zakłócenia o rozkładzie jednostajnym lub normalnym. W przypadku rozkładu jednostajnego poziom szumu określany jest przez zakres, na którym określona jest funkcja rozkładu prawdopodobieństwa. W przypadku rozkładu normalnego poziom szumu określa odchylenie standardowe. W obu przypadkach zakłada się, że rozkład ma wartość średnią wynoszącą zero (aby nie wyróżniać żadnego z kierunków).

Na rysunku 4-11 przedstawiono za pomocą sfery rotacji przykładowy wynik generacji 200 losowych wartości wokół punktu  $(30, 0, 0)$ . Do generacji użyto algorytmu 3, rozkładu normalnego o odchyleniu standardowym  $0,15$ . Jak widać na rysunku niewielkie odchylenie standardowe wprowadza duże zmiany w położeniu punktu w przestrzeni rotacji. Wynika to z niewielkich wartości bezwzględnych współrzędnych kwaternionów jednostkowych mieszczących się w zakresie  $[-1, 1]$ .



**Rysunek 4-11: Wizualizacja przykładowego wyniku generacji 200 losowych wartości w przestrzeni kwaternionów (rozkład normalny,  $\sigma = 0,15$ ).**

Rozkład otrzymywany przez algorytm 3 dla generacji  $N(0, \sigma)$  ma cechy rozkładu normalnego. Jednakże w analizie zastosowanego podejścia należy uwzględnić periodyczność występującą dla przestrzeni rotacji. W przypadku wprowadzenia w kroku 2 bardzo dużego zaburzenia otrzymuje się w rezultacie nowe położenie, którego odległość od położenia początkowego zmienia się periodycznie zgodnie z charakterem przestrzeni rotacji. W takim przypadku otrzymany rozkład traci cechy rozkładu normalnego.

Wpływ opisanej własności na próbkę losową będzie zależeć od przyjętej wartości odchylenia standardowego. Dla względnie niewielkich wartości  $\sigma$  efekt periodyczności prowadzący do utraty cech

rozkładu normalnego będzie zachodził z bardzo niewielkim, pomijalnie małym prawdopodobieństwem. W eksperymentach praktycznych (rozdział 6) zaobserwowano, że odchylenie standardowe o wartościach rzędu 0,1 już bardzo znacząco zniekształca sygnał testowy. Skuteczność algorytmów badana będzie przede wszystkim dla zakłócenia zdefiniowanego w zakresie użytecznym  $\sigma \leq 0,05$ . Przy takich wartościach odchylenia standardowego, wprowadzane zakłócenie będzie posiadało wystarczające cechy rozkładu normalnego. Częstość występowania deformacji rozkładu ze względu na periodyczność dziedziny będzie w takim przypadku pomijalnie mała. Dla rozkładu jedostajnego problem taki nie będzie występował w przypadku przyjęcia odpowiednio niewielkiej wartości szerokości przedziału  $k$ .

### **4.3 Podsumowanie sposobów parametryzacji przestrzeni orientacji**

W rozdziałach 4.1, 4.1.4 przedstawiono różne sposoby parametryzacji orientacji lokalnego układu współrzędnych. Istnieją przekształcenia pomiędzy poszczególnymi parametryzacjami, więc wybór parametryzacji powinien zależeć od konkretnego zastosowania. Często spotyka się sytuację, gdy w danym systemie orientacja układu jest zapisywana i przetwarzana przy pomocy różnych sposobów parametryzacji. Wynika to z różnych własności, które wynikają ze specyfiki każdej z nich. W rozdziale zostanie zaprezentowane krótkie podsumowanie wad i zalet różnych sposobów zapisów orientacji układu.

#### **Specjalna grupa ortogonalna $SO(3)$ – macierze przekształceń**

##### **Zalety**

- Macierze obrotów mogą być rozszerzone do przekształcenia jednorodnego, które zapewnia spójny opis orientacji i translacji układu. W ten sposób mogą być również zapisywane inne transformacje, np. skalowanie.
- Zależności matematyczne dla macierzy są dobrze znane, istnieją opracowane gotowe procedury numeryczne.
- Macierze przekształceń są używane w wielu systemach używających przekształceń przestrzennych (zarówno w dziedzinie robotyki, wizualizacji przestrzennej, systemów animacji, bibliotekach programowych itp.)
- W prosty sposób realizowane jest składanie przekształceń (mnożenie macierzy).

##### **Wady**

- Wyznaczenie macierzy na podstawie złożenia obrotów (np. kąty Eulera) może prowadzić do powstania osobliwości i utraty stopni swobody układu (efekt *gimbal lock*).
- Wprowadzanie danych jest skomplikowane – opis macierzowy jest całkowicie nieintuicyjny.
- Operacje na macierzy wymagają procedur stabilnych numerycznie. Na macierze nałożone jest 6 ograniczeń, które muszą być spełnione po każdym przekształceniu. Metody normalizacji macierzy do postaci ortogonalnej są skomplikowane.

- Parametryzacja jest nadmiarowa (macierze 3x3 lub 4x4 w przypadku przekształcenia jednorodnego).
- Brak dobrych bezpośrednich metod interpolacji pomiędzy orientacjami opisanymi macierzą przekształcenia.
- Przekształcenie do parametryzacji opartych na złożeniu obrotów (np. kąty Eulera) jest niejednoznaczne. Przykładowa propozycja sposobu dekompozycji macierzy przekształceń podana jest w pracach [32], [98].

## Złożenie obrotów, kąty Eulera

### Zalety

- Minimalna parametryzacja orientacji układu (3 liczby).
- Często używane w systemach wizualizacji przestrzennej – najprostszy sposób wprowadzania informacji o rotacji.
- Możliwa łatwe przekształcenie zbioru kątów do macierzy rotacji.

### Wady

- Złożenie obrotów może prowadzić do występowania osobliwości – efekt utraty stopni swobody *gimbal lock*. Osobliwości przy tej parametryzacji są nie do uniknięcia.
- Brak dobrych metod interpolacji – nie jest możliwa bezpośrednia interpolacja pomiędzy dwoma wektorami reprezentującymi kąty obrotów.
- Jest to najprostszy sposób wprowadzenia danych o rotacji, jednak mało intuicyjny, gdyż po każdym przekształceniu orientacja układu się zmienia.
- Bardzo istotna jest kolejność wykonywania rotacji – brak ustalonej konwencji obrotów (24 możliwe kombinacje). Wszelkie przekształcenia muszą uwzględniać tę kolejność.
- Składanie rotacji wymaga przekształcenia parametryzacji do macierzy obrotów – proces jest złożony obliczeniowo.

## Kwaterniony jednostkowe

### Zalety

- Spójny opis rotacji – do opisu wymagane jest podanie tylko 4 liczb, na które nałożone jest tylko jedno ograniczenie.
- Składanie rotacji jest realizowane w prosty sposób przez mnożenie kwaternionów.
- W przestrzeni kwaternionów nie występuje efekt *gimbal lock*.
- Istnieje dobrze opracowane metody interpolacji *slerp* zachowujące stałą prędkość kątową. Opracowano również całą grupę bardziej złożonych metod generowania krzywych interpolacyjnych.
- Większa stabilność numeryczna w porównaniu do przekształceń macierzowych, mniejsza złożoność obliczeniowa procedur ze względu na bardziej spójną parametryzację.

### Wady

- Nie wszystkie operacje matematyczne mogą być zinterpretowane w przestrzeni kwaternionów.
- Matematyka przekształceń nie jest powszechnie znana, a interpretacja zapisu kwaternionowego nie jest intuicyjna.
- Wynik przetwarzania musi leżeć na hipersferze jednostkowej – normalizacja kwaternionów jest jednak łatwiejsza niż normalizacja macierzy ortogonalnych.
- Dla przestrzeni kwaternionów występuje podwójne pokrycie grupy  $SO(3)$ . Każda orientacja w przestrzeni ma podwójną reprezentację wśród kwaternionów  $(q, -q)$
- Kwaterniony reprezentują tylko rotacyjną część przekształcenia w przestrzeni. W pracy [70] zaproponowana jest metoda zapisu pełnej transformacji przy użyciu kwaternionów, jednak w praktyce zastosowanie tego podejścia nie jest wygodne.

### Podsumowanie

Każda z przedstawionych parametryzacji orientacji układu ma pewne ograniczenia. W praktyce wykorzystanie danej parametryzacji jest podyktowane konkretnymi wymaganiami systemu (np. prosty sposób znajdowania rozwiązania kinematyki odwrotnej, prosta interpolacja). Parametryzacja kwaternionowa ma wiele zalet, które nie występują ani w przypadku macierzy obrotów ani kątów Eulera. Do podstawowych zalet kwaternionów należy brak efektu *gimbal lock*, proste składanie obrotów i dobre metody interpolacji. Z punktu widzenia obliczeniowego parametryzacja ta wymaga szczególnej uwagi, aby wyniki należały do hipersfery jednostkowej. Jak to zostało pokazane w rozdziale 4.1.4 w wielu przypadkach możliwe jest stworzenie procedur numerycznych, które zachowują tę właściwość. W rozdziale 5 przedstawione zostaną jeszcze bardziej złożone metody przetwarzania orientacji układu zapisanej w postaci kwaternionowej. Metody te często nie mają swoich prostych odpowiedników dla alternatywnych sposobów parametryzacji orientacji układu. Z tego powodu dalsza część pracy będzie bazować na parametryzacji kwaternionowej, jako podstawowym sposobie opisu rotacji w przestrzeni. Ponieważ istnieją proste metody konwersji do pozostałych parametryzacji, wybór taki nie wprowadza żadnych ograniczeń dotyczących stosowalności zaproponowanych algorytmów.

## 4.4 Model danych opisujących orientację układów

W kolejnych rozdziałach pracy będą rozpatrywane trajektorie w przestrzeniach orientacji lokalnych układów współrzędnych. Rozpatrywany lokalny układ współrzędnych powiązany jest z ciałem sztywnym, dla którego definiuje się położenie w przestrzeni. Praca koncentruje się tylko na przetwarzaniu orientacji układu ze względu na to, że istnieje wiele dobrze zbadanych metod dotyczących przetwarzania translacji zapisanych w postaci standardowej przestrzeni wektorowej  $R^3$ .

#### 4.4.1 Model danych trajektorii

Analogicznie, jak to zostało zrobione dla obrazów, również dla trajektorii przestrzennej zostanie określony model danych reprezentujących orientacje układu, na którym bazować będą algorytmy przetwarzające. Przedstawiony model danych może w ogólności dotyczyć dowolnej parametryzacji orientacji układów. Z powodów przedstawionych w rozdziale 4.3 jako konkretna parametryzacja dla prezentowanego modelu będzie wybrana przestrzeń kwaternionów. Proponowany model danych posiada następujące własności:

- Prototyp orientacji definiuje się jako orientację referencyjną, względem której definiowane będą wszelkie rotacje  $q_0 = [1, (0,0,0)]$
- Określony jest zbiór indeksów dla siatki czasowej definiujących kolejne chwile czasowe  $IT = (1; n)$  Indeksy parametryzują trajektorię w całym odcinku czasowym, który jest rozpatrywany.
- Istnieje zdefiniowane przekształcenie prototypu orientacji w dowolny punkt należący do rozpatrywanej trajektorii.
- W  $IT$  definiuje się relację sąsiedztwa przy użyciu dyskretnej funkcji odległości  $\rho: E_i$  jest sąsiadem  $E_{i'}$ , wtedy i tylko wtedy, gdy  $\rho(i, i') = 1$ . W praktyce oznacza to relację dwusąsiedztwa: dla  $\rho(i, i-1) = 1$  jest to sąsiad “lewy” L,  $\rho(i, i+1) = 1$  jest to sąsiad “prawy” P.

Trajektoria orientacji w przestrzeni kwaternionów określona jest przez krzywą kwaternionową realizującą przyporządkowanie:

$$q: R \rightarrow S^3, \quad (4-114)$$

co oznacza, że funkcja  $q(t)$  jest jednoznacznym przyporządkowaniem wybranej chwili czasowej do przestrzeni orientacji układu względem orientacji referencyjnej w przestrzeni kwaternionowej. Zależność definiuje przyporządkowanie ciągłe. W przypadku rozpatrywania dyskretnej wersji trajektorii (co ma miejsce we wszystkich praktycznych zastosowaniach) przyporządkowanie dotyczy zbioru indeksów dyskretnych chwil czasowych:

$$q: IT \rightarrow S^3 \quad (4-115)$$

Otrzymuje się uporządkowany zbiór kwaternionów reprezentujący trajektorię w przestrzeni orientacji lokalnego układu współrzędnych

$$\{q_i \in S^3 : i \in N \cap [1, n]\}, \quad (4-116)$$

gdzie  $n$  jest liczbą elementów składających się na trajektorię.

W modelu danych reprezentujących trajektorię określona została miara odległości pomiędzy jej elementami. Miara ta realizuje funkcję odległości zdefiniowaną dla przestrzeni  $S^3$ :

$$d_{ij} = d(q_i, q_j) = 2 \|\log(q_i^* q_j)\| \quad (4-117)$$

#### 4.4.2 Miary odległości dla trajektorii

Podobnie jak w przypadku obrazów, również niezbędne jest porównywanie trajektorii – także z punktu widzenia poprawności i skuteczności przeprowadzonej filtracji. Ponieważ w przestrzeni kwaternionów zdefiniowane zostały podstawowe wielkości statystyczne, możliwe jest zastosowanie podobnych zależności, jakie były używane w przypadku obrazów. Pamiętać należy jednak o używanej interpretacji odległości, która bierze pod uwagę jedynie kąt obrotu pomiędzy dwoma kwaternionami.

Odpowiedniki standardowych wielkości służących do oszacowania odległości pomiędzy trajektoriami kwaternionów można sformułować w następujący sposób:

- Kwaternionowy błąd średniokwadratowy:

$$MSE_Q(p, q) = \frac{1}{n} \sum_{i=1}^n \left( 2 \left\| \log q_i^{-1} p_i \right\| \right)^2, \quad (4-118)$$

gdzie  $p, q$  oznacza odpowiednio trajektorię badaną i referencyjną (wzorcową).

- Znormalizowany kwaternionowy błąd średniokwadratowy:

$$NMSE_Q(p, q) = \frac{MSE_Q(q, p)}{\sigma_q^2} \quad (\text{znormalizowany błąd średniokwadratowy}), \quad (4-119)$$

gdzie  $\sigma_q^2$  - kwaternionowa wariancja wyznaczona dla trajektorii referencyjnej  $q$ .

- Kwaternionowy stosunek sygnału do szumu (ang. *signal to noise ratio*):

$$SNR_Q(p, q) = 10 \log_{10} \frac{\sigma_q^2}{MSE_Q(q, p)} = -10 \log_{10} NMSE_Q(q, p) [dB] \quad (4-120)$$

Zależność (4-118) pozwala wyznaczyć miarę odległości dla dwóch trajektorii kwaternionowych pod warunkiem, że mają one jednakową długość. Jeżeli na skutek przetwarzania sygnału liczba elementów trajektorii się zmieniła, to zależności tej nie można już wprost zastosować. Przy przetwarzaniu obrazów przypadek taki dotyczy głównie algorytmów skalujących obraz. Dla trajektorii przestrzennych sytuacja taka częściej występuje z punktu widzenia praktycznego, gdy na przykład przetwarzania polega na wybieraniu pewnej liczby elementów z trajektorii referencyjnej i generowaniu krzywej interpolacyjnej. Do porównania krzywych o różnej długości należy użyć innego podejścia.

#### 4.5 Adaptacja metody dynamicznego zniekształcania dziedziny czasu dla kwaternionów

Metodą, która pozwala porównywać szeregi czasowe o różnej długości jest dynamiczne zniekształcanie dziedziny czasu. Dodatkową zaletą takiego podejścia jest wyznaczenie miary podobieństwa szeregu czasowego i znalezienie takiego dopasowania elementów, które minimalizuje określoną miarę odległości. Początkowo algorytm dynamicznego zniekształcania dziedziny czasu DTW (ang. *Dynamic Time Warping*) był używany do porównywania i rozpoznawania sygnałów mowy [81]. Algorytm pozwala nie



tylko na wyznaczenie miary odległości pomiędzy szeregami czasowymi, ale także na dopasowanie jednego z nich do drugiego.

Użycie algorytmu do porównywania dwóch kwaternionowych szeregów czasowych o jednakowych długościach również może być przydatne w praktyce. Może się zdarzyć, że szeregi są do siebie wizualnie podobne, jednak przesunięte w czasie o jedną próbkę. W ogólności może to być dowolne zniekształcenie, które zachowuje ogólne podobieństwo, a zmienia przyporządkowanie charakterystycznych fragmentów. W takim przypadku wyznaczenie prostej miary odległości nie oddaje w pełni otrzymanych rezultatów. Mimo, że szeregi można rozpatrywać, jako podobne, to przykładowo miara SNR wskazuje, że znacznie się od siebie różnią. Algorytm DTW może być pomocny przy rozróżnianiu tego typu podobieństwa.

Ponieważ trajektorie kwaternionowe są szeregiem czasowym (kolejność elementów jest ustalona) o specyficznych danych, to algorytm DTW może zostać w tym przypadku zastosowany. Adaptacja podstawowej wersji algorytmu podana zostanie w rozdziale 4.5.1. Następnie algorytm ten będzie uogólniony, aby wykorzystać specyficzne własności przestrzeni kwaternionów jednostkowych.

#### 4.5.1 Podstawowy schemat algorytmu

Dane są dwie trajektorie kwaternionowe (szeregi czasowe) składające się odpowiednio z  $n$  oraz  $m$  elementów:

$$q = \{q_1, q_2, \dots, q_n\} \quad \text{- trajektoria wzorcowa (referencyjna)} \quad (4 - 121)$$

$$p = \{p_1, p_2, \dots, p_m\} \quad \text{- trajektoria testowa} \quad (4 - 122)$$

Tworzy się macierz  $M_D$  o wymiarach  $n \times m$ . Każdy element  $(i, j)$  macierzy  $M_D$  reprezentuje odległość pomiędzy elementami trajektorii  $p_i$  oraz  $q_j$ . W przypadku zastosowania podstawowej miary odległości (będącej odpowiednikiem miary euklidesowej używanej dla danych w przestrzeni rzeczywistych) używane jest wyrażenie (4 – 94):

$$M_D(i, j) = d(p_i, q_j) = 2 \|\log p_i^{-1} q_j\| \quad (4 - 123)$$

W ogólności funkcja odległości pomiędzy pojedynczymi elementami może być dowolnie zdefiniowana, co zostanie wykorzystane przy konstrukcji algorytmu wykorzystującego specyfikę przestrzeni kwaternionów.

Każdy element trajektorii  $p$  może być przypisany do jednego lub więcej elementów trajektorii  $q$ . Przypisanie dwóch elementów do siebie oznacza się jako  $(i, j)$ , gdzie  $i, j$  są indeksami elementów odpowiednio trajektorii  $p$  oraz  $q$ . Ścieżką dopasowania (ang. *warping path*)  $L$  nazywa się ciągły szereg elementów  $l_k$  określających przypisanie do siebie elementów z rozpatrywanych trajektorii:

$$L = l_1, l_2, \dots, l_k, \dots, l_K, \quad (4 - 124)$$

gdzie  $K$  jest całkowitą długością ścieżki dopasowania taką, że  $\max(m, n) \leq K \leq m + n - 1$ . Element  $k$ -ty ścieżki określa przypisanie dwóch elementów trajektorii do siebie  $l_k = (i, j)_k$ . Na ścieżkę dopasowania  $L$  nakłada się następujące warunki:

- *warunki brzegowe* – określające, że pierwsze i ostatnie elementy szeregów muszą być przypisane do siebie. Oznacza to, że  $l_1 = (1,1)$  oraz  $l_K = (n, m)$ ,
- *warunek ciągłości* – jeżeli  $l_k = (a, b)$ , to  $l_{k+1} = (a', b')$  takie, że  $a - a' \leq 1$  oraz  $b - b' \leq 1$ , co ogranicza dozwolone kroki w ścieżce dopasowania do sąsiednich pól macierzy  $D$  (również sąsiednich po przekątnej),
- *warunek monotoniczność* – jeżeli  $l_k = (a, b)$ , to  $l_{k+1} = (a', b')$  takie, że  $a - a' \geq 0$  oraz  $b - b' \geq 0$ , co oznacza, że punkty ścieżki dopasowania muszą być rozłożone monotonicznie w czasie (nie mogą powracać do raz przypisanych elementów).

Definiuje się funkcję celu dla ścieżki dopasowania  $L$  określającą całkowity koszt dopasowania trajektorii  $p$  do  $q$  jako:

$$\Lambda_L = \sum_{k=1}^K M_D(l_k), \quad (4 - 125)$$

gdzie dla  $l_k = (i, j)$  element macierzy  $M_D(l_k) = M_D(i, j) = d(p_i, q_j)$  (odległość elementów trajektorii).

Istnieje wiele ścieżek dopasowania spełniających przedstawione warunki. Z punktu widzenia wyznaczenia miary odległości dla trajektorii interesująca jest ta, która minimalizuje funkcję celu określoną równaniem (4 – 125). Będzie to miara odległości trajektorii rozumiana w sensie metody DTW:

$$DTW(p, q) = \min_L \Lambda_L = \min_L \left\{ \sum_{k=1}^K M_D(l_k) \right\} \quad (4 - 126)$$

Jest to problem optymalizacyjny, który (dla przypadku jednowymiarowego) można rozwiązać w czasie wielomianowym [81]. W tym celu tworzy się macierz o wymiarach  $n \times m$ . W macierzy, dla każdego elementu trajektorii zapisany jest minimalny koszt dopasowania ścieżki liczony od początku macierzy. Każdy element macierzy może być zdefiniowany rekurencyjnie, jako:

$$\delta(i, j) = M_D(i, j) + \min\{\delta(i-1, j-1), \delta(i, j-1), \delta(i-1, j)\}, \quad (4 - 127)$$

gdzie  $D$  jest macierzą odległości elementów.

Element macierzy określony jako  $\delta(m, n) = DTW(p, q)$  jest szukaną minimalną wartością kosztu dopasowania. Ścieżka, która daje taką wartość jest natomiast szukaną optymalną ścieżką dopasowania trajektorii  $p$  i  $q$ . Złożoność obliczeniowa tak działającego algorytmu jest rzędu  $O(mn)$ .

Opisany sposób programowania dynamicznego daje w wyniku macierz, której element  $\delta(m, n)$  określa koszt dopasowania do siebie dwóch trajektorii. Na podstawie macierzy można wyznaczyć optymalną ścieżkę dopasowania minimalizującą funkcję celu podaną równaniem (4 – 125). W ten sposób otrzymuje się trajektorie testową  $p$  dopasowaną do trajektorii wzorcowej  $q$ . Aby wyznaczyć ścieżkę dopasowania należy prześledzić jej przebieg. Zaczynając od końca, należy przechodzić przez kolejne sąsiadujące elementy macierzy zgodnie z regułą przechodzenia do elementu, który jest najmniejszy. Należy również zachować podane dodatkowe ograniczenia na ścieżkę dopasowania. Jeżeli macierz została skonstruowana prawidłowo, to ostatnim elementem ścieżki (czyli pierwszym elementem ścieżki wynikowej) będzie element  $(0,0)$  macierzy. Ostatnim etapem jest odwrócenie kolejności w wyznaczonej ścieżce.

Tworzenie trajektorii dopasowanej  $r$  polega na wykonaniu odpowiednich operacji na trajektorii testowej  $p$  według wyznaczonego dopasowania jej do trajektorii wzorcowej  $q$ . Trajektorja dopasowana ma długość równą trajektorii wzorcowej  $m$ , oraz indeksowana jest przez zmienną  $s$ :  $r = \{r_1, r_2, \dots, r_s, \dots, r_M\}$ .

Wyróżnia się trzy rodzaje operacji:

- *przepisanie* – polegające na wstawieniu do trajektorii dopasowanej punktu z trajektorii testowej. Odpowiada to jednoznaczному przyporządkowanie elementów trajektorii do siebie. W macierzy reprezentującej ścieżkę dopasowania przedstawione jest to diagonalnym przejściem pomiędzy elementami macierzy.
  - *Warunek*:  $\min\{\delta(i-1, j-1), \delta(i, j-1), \delta(i-1, j)\} = \delta(i-1, j-1)$
  - *Operacja*:  $r_s = p_i$
- *usunięcie* – polegające na usunięciu elementów z trajektorii testowej, czyli nie przepisywanie ich do sygnału dopasowanego. Praktycznie odbywa się to przez uśrednienie wartości zastąpionych elementów. Odpowiada to sytuacji, gdy wiele elementów trajektorii testowej przypisanych jest do jednego punktu trajektorii wzorcowej. W obrazie ścieżki dopasowania przedstawione jest to przez poziome przejście pomiędzy kolejnymi elementami macierzy.
  - *Warunek*:  $\min\{\delta(i-1, j-1), \delta(i, j-1), \delta(i-1, j)\} = \delta(i, j-1)$
  - *Operacja*:  $r_{f-1} = p_{i-1} \left( p_{i-1}^{-1} p_i \right)^{1/2}$  - jeżeli przypisanych jest więcej niż dwa elementy, należy przeprowadzić uśrednianie kwaternionowej na całej grupie elementów,
- *wstawienie* – polegające na wstawieniu dodatkowych elementów do trajektorii testowej. Wstawienie to polega na interpolacji elementów na podstawie elementów, które są sobie przypisane. Odpowiada to sytuacji, gdy wiele elementów trajektorii wzorcowej przypisanych jest do jednego elementu trajektorii testowej. W obrazie ścieżki dopasowania jest to pionowe przejście pomiędzy kolejnymi elementami macierzy.
  - *Warunek*:  $\min\{\delta(i-1, j-1), \lambda(i, j-1), \delta(i-1, j)\} = \delta(i-1, j)$
  - *Operacja*:  $r_f = p_i \left( p_i^{-1} p_{i+1} \right)^{1/2}$  - jeżeli przypisanych jest więcej niż dwa elementy, należy wyznaczyć odpowiednią ilość elementów przez kilkukrotną interpolację.

Przełgądając ścieżkę dopasowania od jej początku i wykonując w każdym kroku operację zgodnie z przyjętymi zasadami otrzymuje się sygnał dopasowany  $r$ .

#### 4.5.2 Uogólnienie pojęcia lokalnej miary podobieństwa elementów trajektorii

Jedną z najważniejszych zalet algorytmu DTW jest określenie miary podobieństwa, która uwzględnia charakter sygnału. Jest to szczególnie istotne w przypadku sygnałów, które zostają poddane subiektywnej ocenie ludzkiej. Przykładowo sygnał dźwięku (mowa, muzyka, itp.) jest rozpoznawany przez człowieka na podstawie podobieństwa do znanego wzorca. Możliwe jest więc pewne zniekształcenie tego sygnału, aby nadal był on prawidłowo rozpoznany. W przypadku automatycznych systemów klasyfikujących sygnały,

również takie właściwości powinny być brane pod uwagę. Algorytm DTW jest jednym z podejść, które z założenia bierze pod uwagę nie tylko odległość naturalnie odpowiadających sobie elementów sygnału, ale próbuje je dopasować do siebie tak, aby odległość ta została zminimalizowana (przy zadanych ograniczeniach). Szukana jest więc miara podobieństwa, w której brany jest pod uwagę cały sygnał lub pewne otoczenie rozpatrywanego elementu.

Własności porównawcze algorytmu DTW będą używane do oceny efektów filtracji trajektorii w przestrzeniach rotacji. Zaprezentowaną poprzednio adaptację algorytmu dla kwaternionów można uogólnić biorąc pod uwagę dodatkowe czynniki. Podobne do prezentowanego podejścia zostało przedstawione w pracy [50] (Keogh, Pazzani). W pracy dla lokalnej miary odległości zastosowana dyskretną pochodną sygnału w punkcie. Ten sam autor prowadził następnie badania dotyczące wykorzystania algorytmu DTW do porównywania ruchów w przestrzeni trójwymiarowej [51]. Jednakże praca ta nie dotyczy bezpośrednio analizy możliwości przetwarzania trajektorii rotacji, gdyż jedynie podstawowa odległość euklidesowa jest używana w algorytmie porównywania. Podejście oparte biorące pod uwagę zarówno wartość sygnału w punkcie oraz jego pierwszą pochodną zostało przedstawione w pracy [60].

Nowym podejściem związanym z wykorzystaniem algorytmu DTW do porównywania trajektorii w przestrzeni rotacji jest propozycja, aby lokalna miara podobieństwa złożona była z iloczynu trzech składowych:

$$M_D(i, j) = d_q(p_i, q_j) d_d(p_i, q_j) d_k(p_i, q_j) \quad (4 - 128)$$

Każda ze składowych miary jest ilościowym wskaźnikiem określającym podobieństwo rotacji o indeksach  $i$  oraz  $j$ . Pierwsza składowa  $d_q$  jest równa standardowej mierze odległości zdefiniowanej w przestrzeni kwaternionów:

$$d_q(p_i, q_j) = 2 \left\| \log p_i^{-1} q_j \right\| \quad (4 - 129)$$

Druga składowa dotyczy miary odległości pomiędzy kwaternionami będącymi dyskretnymi pochodnymi trajektorii w punktach. Miara ta wyrażona jest następująco:

$$d_d(p_i, q_j) = 2 \left\| \log(p_i^{-1} p_{i+1})^{-1} (q_j^{-1} q_{j+1}) \right\| \quad (4 - 130)$$

Dla każdej trajektorii wyznaczana jest wartość pochodnej rozumianej, jako kwaternionowa różnica skończona wyrażająca się miarą zmienności położenia w przestrzeni rotacji. Dla otrzymanych kwaternionów pochodnych bierze się następnie skalarną miarę odległości między nimi. W ten sposób otrzymuje się informację o podobieństwie sygnałów ze względu na podobieństwo lokalnie wyznaczonych pochodnych kwaternionowych w punkcie (chwilowej prędkości ruchu).

W praktycznym zastosowaniu przydatne okazało się także pośrednie rozpatrzenie pochodnej wyższego rzędu. Interesującą cechą charakterystyczną trajektorii z punktu widzenia obserwatora jest podobieństwo ze względu na lokalną krzywiznę. Dokładne rozpatrzenie problemu wyznaczania krzywizny dyskretnej trajektorii kwaternionów zostanie przedstawione w rozdziale 5.4. Prezentowany algorytm korzysta z tych rezultatów, co zostało umieszczone w rozdziale 4 dla zapewnienia pełnego opisu metody.

Trzecia składowa miary podobieństwa oparta jest na wyznaczeniu skalarnej wartości bezwzględnej z różnicy miar określających krzywiznę rozpatrywanych trajektorii w punkcie.

$$d_k(p_i, q_j) = |\kappa(p_i) - \kappa(q_j)| = \left\| \omega_A(p_i) - \omega_A(p_{i-1}) \right\| - \left\| \omega_A(q_j) - \omega_A(q_{j-1}) \right\|, \quad (4 - 131)$$

gdzie  $\omega_A(p_i)$  oznacza oś obrotu wyznaczoną na podstawie prędkości kątowej dla kwaternionu  $i$ -tego trajektorii  $p$ :

$$\omega_A(p_i) = \frac{\log(p_i^{-1} p_{i+1})}{\left\| \log(p_i^{-1} p_{i+1}) \right\|} \quad (4 - 132)$$

Dzięki zaprezentowanemu podejściu lokalna miara podobieństwa trajektorii uwzględnia informacje lokalne, jak i z pewnego otoczenia punktów. Brane są pod uwagę: różnica położenia w przestrzeni rotacji, różnica prędkości kątowej (w postaci kwaternionowej pochodnej) oraz różnica lokalnej krzywizny trajektorii (w postaci zmiany przyspieszenia stycznego). Dzięki temu możliwe jest często bardziej prawidłowe porównanie trajektorii z punktu widzenia oceny zbliżonej do subiektywnego podobieństwa określanego przez człowieka (którego ocena jest oparta w dużej mierze właśnie na krzywiznie).

Algorytm uwzględniający przedstawione powyżej specyficzne własności przestrzeni kwaternionów nazywany będzie QDTW (ang. *Quaternion Dynamic Time Warping*). Używana w nim wielkość wyznaczona (4 – 128) nie jest miarą odległości rozumianą, jako funkcja określona w przestrzeni metrycznej. Bierze ona pod uwagę nie tylko położenie punktu w przestrzeni, ale również własności trajektorii w jego otoczeniu. To może powodować, że nie muszą być spełnione aksjomaty metryki. Niemniej jednak spełnienie warunków na przestrzeń metryczną nie jest konieczne z punktu widzenia poprawności działania algorytmu. Będzie to oczywiście wpływało na metody selekcji pozwalające na prawidłowe rozróżnienie badanych obiektów. Doświadczenia numeryczne przeprowadzone przez autora pokazały, że tak skonstruowana miara ma dobre właściwości w zastosowaniach praktycznych. Może więc być użyta, jako metoda porównawcza dla trajektorii z zastrzeżeniem, że lokalne porównanie jest określone miarą podobieństwa (a nie odległości) pomiędzy wybranymi elementami.

#### 4.5.3 Konstrukcja miary zaszumienia jakości sygnału opartej na algorytmie QDTW

Algorytm QDTW działa według standardowego schematu używanego w metodzie dynamicznego zniekształcania dziedziny czasu. W wyniku jego działania otrzymuje się zarówno dopasowanie trajektorii testowej do trajektorii wzorcowej oraz miarę podobieństwa przebiegów do siebie. Miara ta jest pewną bezwzględną skalarną wartością, która następnie może być znormalizowana podobnie, jak to jest realizowane w przypadku standardowych wielkości będących oszacowaniem odległości pomiędzy trajektoriami kwaternionów – zależności (4 – 118) – (4 – 120). W tym celu proponuje się zdefiniowanie następujących wartości znormalizowanych opartych na mierze podobieństwa QDTW:

- Znormalizowana miara podobieństwa trajektorii oparta na QDTW:

$$QDTW_{norm}(p, q) = \frac{QDTW(q, p)}{n\sigma_q^2} \quad (\text{znormalizowany błąd średniokwadratowy}), \quad (4 - 133)$$

gdzie  $\sigma_q^2$  - kwaternionowa wariancja wyznaczona dla trajektorii referencyjnej  $p$ .

- Kwaternionowy stosunek sygnału do szumu (ang. *Quaternion Signal to Noise Ratio*):

$$QSNR(p, q) = 10 \log_{10} \frac{n \sigma_q^2}{QDTW(q, p)} = -10 \log_{10} QDTW_{norm}(q, p) [dB] \quad (4 - 134)$$

Przeprowadzona normalizacja dotyczy dwóch elementów: długości oraz wariancji rozpatrywanego sygnału. W przeciwieństwie do błędu średniokwadratowego używanego przy standardowej mierze jakości, wartość podobieństwa trajektorii otrzymywana przy użyciu algorytmu QDTW nie jest wartością średnią, a całkowitą rozbieżnością pomiędzy sygnałami. Normalizacja ze względu na wariancję sygnału bierze pod uwagę wartość wariancji kwaternionowej wyznaczonej wyrażeniem (4 – 113). Wyrażenie to jest wartością skalarną interpretowaną jako miara rozbieżności rozkładu elementów. Dzięki tym operacjom otrzymuje się znormalizowaną wartość miary podobieństwa sekwencji do siebie, która następnie może być przedstawiona w postaci współczynnika jakości sygnału do szumu za pomocą zależności (4 – 134).

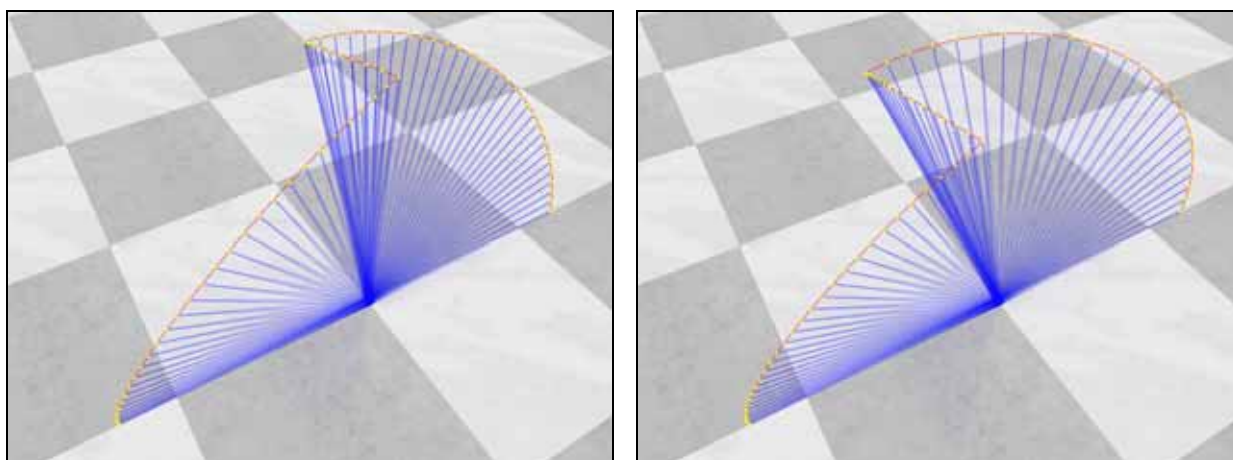
Ostatecznie otrzymuje się miarę podobieństwa trajektorii uwzględniającą następujące czynniki:

- specyfikę danych opisujących rotacje, jako trajektorie w przestrzeni kwaternionów,
- możliwość porównywania trajektorii o różnych długościach,
- złożoną lokalną miarę podobieństwa elementów biorącą pod uwagę różnice w lokalnej prędkości oraz lokalnej krzywiznie trajektorii,
- znormalizowaną wielkość ze względu na miarę rozbieżności rozkładu elementów,
- znormalizowany współczynnik jakości sygnału do szumu określony w decybelach.

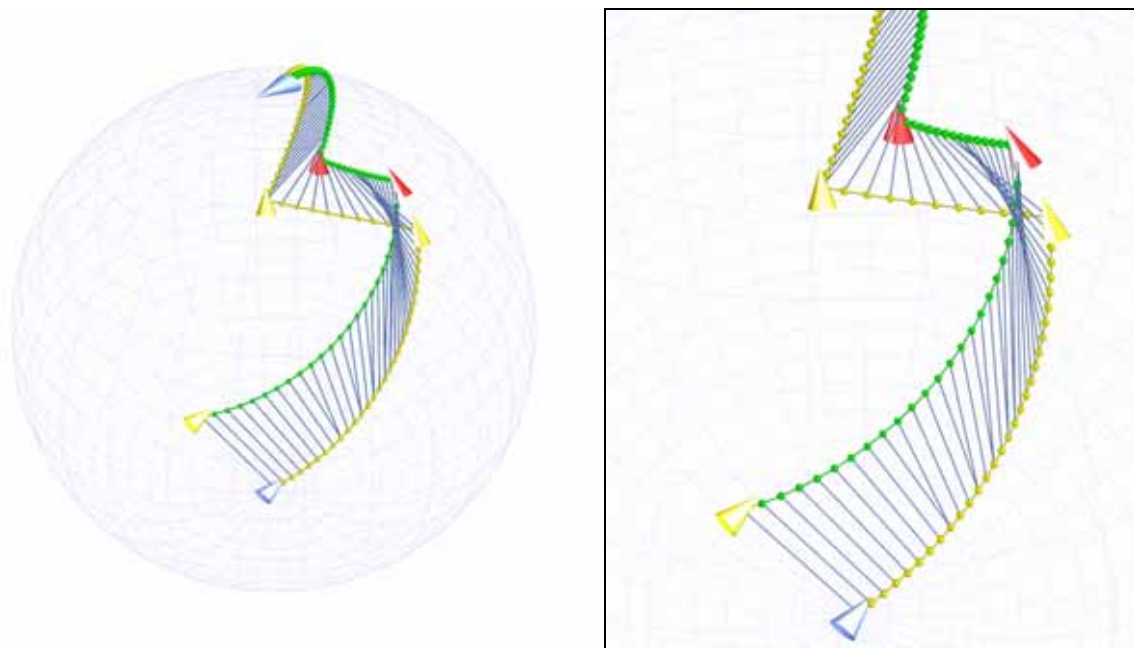
Zaproponowany współczynnik jakości sygnału będzie używany w dalszej części pracy do określenia skuteczności filtracji przeprowadzanej dla trajektorii.

#### 4.5.4 Przykład działania algorytmu QDTW

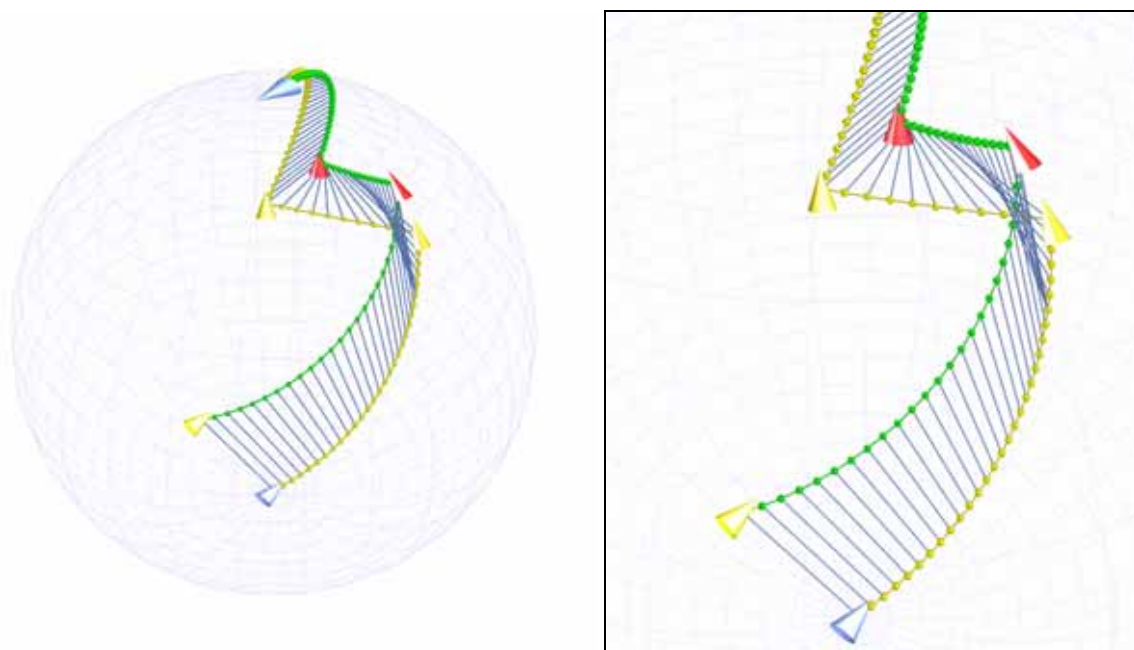
W celu zobrazowania sposobu działania uogólnionego algorytmu zostały wygenerowane dwie proste sekwencje testowe. Ich przebieg pokazany został na rysunku 4-12.



Rysunek 4-12: Przykładowe testowe trajektorie rotacji (wzorcowa po lewej, testowa po prawej) podobne ze względu na subiektywne porównanie.



Rysunek 4-13: Wynik działania algorytmu DTW dla lokalnej miary uwzględniającej jedynie kwaternionową odległość elementów trajektorii.



Rysunek 4-14: Wynik działania algorytmu QDTW dla uogólnionej lokalnej miary podobieństwa elementów.

Dla obserwatora obie sekwencje są do siebie podobne i składają się z 3 odcinków. Odcinek środkowy ruchu różni się zmianą zarówno położenia w przestrzeni rotacji oraz jego długością (ma 5 równo rozłożonych elementów więcej dla sekwencji testowej). Podobieństwo sekwencji jest analizowane przez obserwatora w dużej mierze na podstawie różnic występujących dla każdego z odcinka osobno. W związku z tym naturalnym jest wzajemne przyporządkowanie punktów biorąc pod uwagę także zmianę krzywizny (a nie tylko położenia), która w sposób dokładniejszy określa zmianę kierunku ruchu i przykładowo wyróżnia

poszczególne odcinki. Może to zostać zrealizowane przy użyciu uogólnionej wersji algorytmu dynamicznego zniekształcania dziedziny czasu QDTW.

Na rysunkach 4-13, 4-14 pokazane zostały trajektorie testowe przedstawione przy użyciu sfery rotacji (patrz rozdział 4.1.4). Trajektorja wzorcowa zaznaczona jest kolorem żółtym. Trajektorja testowa zaznaczona jest kolorem zielonym. Wzajemnie przyporządkowanie elementów każdej z trajektorii do siebie zaznaczono linią granatową. Rysunki pozwalają na porównanie działania standardowego algorytmu uwzględniającego jedynie położenie (i miarę odległości w funkcji położenia) oraz algorytmu uogólnionego QDTW. Prawa kolumna jest powiększeniem interesującego obszaru trajektorii.

Prawidłowe przyporządkowanie punktów powinno uwzględniać podobieństwo trajektorii dla fragmentów początkowego i końcowego. Odcinek środkowy różni się liczbą punktów, które muszą być przyporządkowane w relacji jeden do wielu. Algorytm oparty na mierze wykorzystującej tylko kwaternionową odległość pomiędzy elementami ma znaczne problemy z właściwym przyporządkowaniem punktów już w obrębie pierwszego odcinka ruchu (rysunek 4-13). Pomimo, że liczba punktów obu trajektorii dla tego odcinka jest jednakowa, to przyporządkowanie otrzymane przy użyciu algorytmu podstawowego generuje relacje jeden do wielu. Podobna sytuacja występuje dla ostatniego odcinka trajektorii. Zastosowanie algorytmu QDTW uwzględniającego cechy trajektorii wyższego rzędu, co pozwala na bardziej prawidłowe przyporządkowanie pokazane na rysunku 4-14. Przyporządkowanie punktów trajektorii w relacji jeden do wielu występuje dopiero w okolicach środkowego odcinka, który różni się długością dla obu przebiegów. Punkty należące do odcinka początkowego i końcowego są prawidłowo przyporządkowane w relacji jeden do jednego. Dzięki tym cechom miara podobieństwa całych trajektorii uzyskana algorytmem QDTW uwzględnia podobieństwo sygnałów w bardziej naturalny sposób.

#### 4.5.5 Podsumowanie kwaternionowej wersji algorytmu DTW

W celu adaptacji algorytmu na potrzeby porównywania trajektorii kwaternionowych zostały wprowadzone przez autora następujące modyfikacji w stosunku do wersji standardowej dla sygnału jednowymiarowego:

- uwzględnienie kwaternionowej miary odległości opisanej wyrażeniem (4 – 94),
- zamiana operacji uśredniania przy usuwaniu elementów na wyznaczanie średniej kwaternionowej opisanej wyrażeniem (4 – 105),
- zamiana liniowej interpolacji przy wstawianiu elementów na sferyczno-liniową interpolację *sterp* opisaną wyrażeniem (4 – 83).

W przypadku porównywania sekwencji jednowymiarowych sygnałów o wartościach rzeczywistych użycie euklidesowej odległość poszczególnych elementów jest często wystarczające i pozwala na właściwe wyznaczenie podobieństwa trajektorii. Jednakże porównywanie złożonych trajektorii, do jakich należą sekwencje rotacji reprezentujących ruch, jest realizowane przez obserwatora zewnętrznego przy uwzględnieniu dodatkowych cech sygnału. W szczególności istotnym elementem świadczącym o podobieństwie lub jego braku dwóch trajektorii jest zróżnicowanie szybkości ruchu oraz różnice w



zakrzywieniu sygnałów. Zaadaptowany algorytm został następnie uogólniony przez zastosowanie nowej miary lokalnej rozbieżności elementów trajektorii wykorzystującej dodatkowe informacje o sygnale. Brane są pod uwagę trzy elementy:

- odległość określona dla przestrzeni kwaternionów (4 – 130),
- różnica chwilowej prędkości kątowej wyznaczonej przy użyciu kwaternionowej pochodnej w punkcie (4 – 131),
- różnica w zakrzywieniu trajektorii określona przez różnicę pomiędzy składowymi stycznymi przyspieszenia kątowego (4 – 132).

Uwzględnienie wszystkich trzech elementów pozwala na bardziej naturalne przyporządkowanie punktów trajektorii, a co za tym idzie lepszą estymację miary podobieństwa pomiędzy badanymi sygnałami. Jest to ważne z punktu widzenia oceny jakości sygnału przetworzonego przy użyciu różnych algorytmów (patrz rozdział 6). Opracowanie kwaternionowej wersji algorytmu DTW pozwala więc nie tylko na porównywanie trajektorii kwaternionów o różnych długościach, ale także na uwzględnienie przy porównaniu dodatkowych czynników wpływających na charakter sygnału i ruchu reprezentowanego przez trajektorię.

## **4.6 Reprezentacja danych dla modeli hierarchicznych łańcuchów kinematycznych**

W rozdziale 4.4 przedstawiono model, który pozwala na zapis danych reprezentujących trajektorię w przestrzeni orientacji dla lokalnego układu współrzędnych. Model taki jest często wykorzystywany w praktyce do opisu ruchu układów składających się z kilku ruchomych elementów (z których każdy najczęściej jest ciałem sztywnym). Jedną z najistotniejszych dziedzin, w których występuje tego typu model danych jest problematyka opisu ruchu manipulatora. Ponieważ nie są w tym przypadku uwzględniane przyczyny tego ruchu, otrzymany opis będzie nazwany kinematycznym.

W praktyce pewna klasa manipulatorów może się składać z wielu połączonych ze sobą członów. Każdy człon ma określoną liczbę stopni swobody ruchu wykonywanego względem członu go poprzedzającego. Połączenie poszczególnych członów tworzy łańcuch kinematyczny. W rozdziale 4.6 zostaną przedstawione podstawowe sposoby opisu łańcuchów kinematycznych i zagadnień związanych z badaniem wykonywanych przez nich ruchów. Przedstawiony zostanie również przykładowy hierarchiczny układ złożonego łańcucha kinematycznego, jakim jest szkielet ludzki.

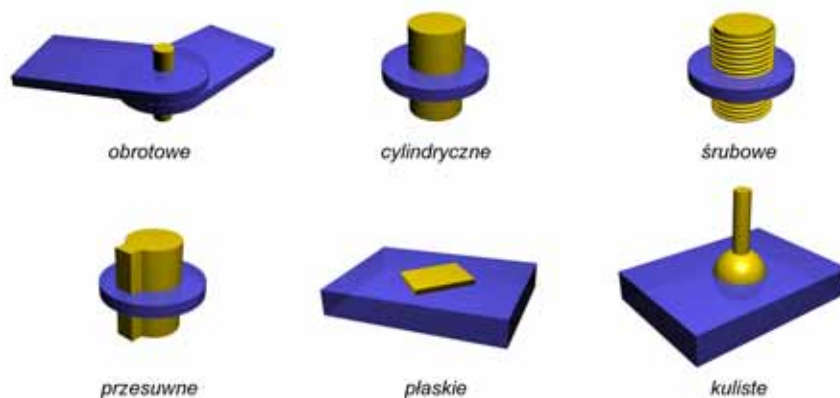
Problematyka hierarchicznych łańcuchów kinematycznych w postaci manipulatorów jest przykładem jednej z dziedzin, w jakiej są stosowane modele opisujące trajektorie w przestrzeni rotacji. W związku z tym również dla tych zagadnień będą miały zastosowanie metody, które zaproponowane zostaną w niniejszej rozprawie w rozdziale 5. Jak to zostanie pokazane, ruch pojedynczego elementu obrotowego może być rozpatrywany i przetwarzany, jako trajektoria w przestrzeni rotacji. Złożony model hierarchiczny, rozumiany jako wielokrotne połączenie łańcuchów kinematycznych jest natomiast realizowany przez szkielet postaci ludzkiej. Jest to więc kolejny przykład zastosowania problematyki manipulatorów w pokrewnej dziedzinie

związanej z analizą ruchu człowieka. W obu przypadkach zastosowanie mogą znaleźć metody przetwarzania i filtracji trajektorii w przestrzeni rotacji.

#### 4.6.1 Zadanie kinematyki prostej i odwrotnej dla łańcuchów kinematycznych

W zagadnieniach robotyki problemy kinematyczne rozpatrywane są między innymi dla manipulatorów składających się z ze zbioru ciał sztywnych (zwanych członami) połączonych ze sobą w otwarty łańcuch kinematyczny [18]. Dwa sąsiadujące ze sobą człony mogą mieć połączenie ruchowe lub stałe (w takim przypadku człony są rozpatrywane jako jeden człon). Połączenie ruchowe dwóch członów, w którym względny ruch opisywany jest przez dwie współpracujące ze sobą powierzchnie nazywa się parą niższego rzędu. Na rysunku 4-15 przedstawiono schematycznie 6 możliwych połączeń dla par niższego rzędu.

W praktyce najczęściej wykorzystywane są połączone obrotowe i przesuwne o jednym stopniu swobody. Układ kinematyczny o  $n$  stopniach swobody można zamodelować złożeniem  $n$  członów o pojedynczym stopniu swobody. Dlatego często w rozważaniach teoretycznych rozpatruje się manipulatory o jednym stopniu swobody. W następnym rozdziale zostanie zaprezentowany złożony model hierarchiczny, w którym poszczególne człony mają zróżnicowaną liczbę stopni swobody.



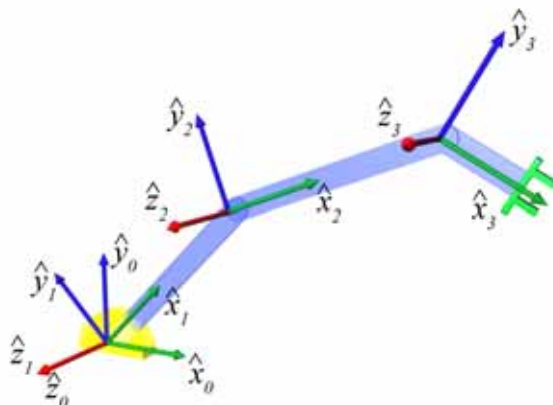
Rysunek 4-15: Rodzaje połączeń ruchowych dla par niższego rzędu [18].

Człony manipulatora rozważa się w uproszczonym przypadku, jako ciało sztywne o dwóch podstawowych parametrach: długość oraz kąt skręcenia członu. Parametry te opisują powiązanie pomiędzy dwoma członami tworzącymi parę obrotową przez opisanie względnego położenia osi rozpatrywanego członu.

Dla niektórych zagadnień rozpatrywane są układy o zamkniętym łańcuchu kinematycznym (to znaczy takie, w którym łańcuch kinematyczny może być opisany za pomocą zamkniętego grafu) [75]. Układy tego typu mają bardzo duże znaczenie praktyczne, jednak analiza ich kinematyki, jak i dynamiki jest bardziej skomplikowana od analizy układów otwartych i wykracza poza ramy tej pracy. W pracy rozpatrywane będą jedynie układy o otwartym łańcuchu kinematycznym.

**Zadanie kinematyki prostej** (ang. *forward kinematics*) polega na określeniu położenia (lub innych parametrów kinematycznych takich jak prędkość lub przyspieszenie) wybranego punktu manipulatora względem globalnie zdefiniowanego układu współrzędnych odniesienia. W ogólności zadanie to dotyczy dowolnego lub wszystkich punktów manipulatora (na przykład w przypadku realizowania problemu wizualizacji trajektorii członów manipulatora). Często spotykanym problemem jest określenie położenia narzędzia powiązanego z ostatnim (roboczym) członem manipulatora. Układ odniesienia, względem którego poszukiwane jest rozwiązanie również może zależeć od konkretnego zadania. Standardowymi układami odniesienia rozpatrywanymi w praktycznym problemach kinematyki manipulatorów są przykładowo [18]: układ odniesienia podstawy, stanowiska, kiści, narzędzia, układ docelowy.

W zadaniu kinematyki prostej informacja o cząstkowym położeniu każdego z członów manipulatora jest używana w celu wyznaczenia informacji o położeniu globalnym. Dla każdego członu określona jest transformacja opisująca jego położenie względem członu poprzedzającego go w łańcuchu kinematycznym (rysunek 4-16). Na przykładowym rysunku rozpatrywany jest manipulator składający się jedynie z obrotowych członów o jednym stopniu swobody. Jednak wyrażenie wyznaczające położenie końcówki manipulatora jest prawdziwie niezależne od użytych transformacji. Dalsze rozważania złożonego łańcucha kinematycznego również ograniczone będą do transformacji orientacji.



**Rysunek 4-16:** Zmiana orientacji lokalnych układów współrzędnych przyporządkowanych do członów.

Niech manipulator składa się z łańcucha kinematycznego o  $n$  członach. Oznaczenie  ${}^j_iT$  oznacza przekształcenie określające położenie lokalnego układu członu  $i$  względem układu odniesienia związanego z członem  $j$ . Przyjmuje się, że z członem 0 związany jest globalny układ odniesienia zazwyczaj przypisany do nieruchomej podstawy manipulatora. Przekształcenie  ${}^j_iT$  jest funkcją wszystkich parametrów członu  $i$ : zarówno parametrów wynikających z konstrukcji mechanicznej, jak i parametrów opisujących zmianę położenia względem członu  $j$ . Sposoby wyznaczania przekształcenia w zależności od parametryzacji członu przedstawiony jest w [18].

Wyznaczenie położenia końcówki manipulatora (ostatniego członu z łańcucha kinematycznego) jest złożeniem wszystkich przekształceń występujących w łańcuchu kinematycznym:

$${}^0_n T = {}^0_1 T {}^1_2 T {}^2_3 T \dots {}^{n-1}_n T \quad (4-135)$$

Złożenie to jest przekształceniem przestrzeni współrzędnych konfiguracyjnych układu (współrzędnych opisujących konfigurację pojedynczych elementów) w przestrzeń współrzędnych kartezjańskich. Wyróżnia się również przestrzeń współrzędnych napędowych, w której określone są pozycje napędów powiązanych z członami manipulatora. W niektórych zastosowaniach wygodniej jest operować na pozycjach napędów. Przestrzeń konfiguracyjna jest wtedy funkcją przestrzeni współrzędnych napędowych.

W przypadku, gdy rozważana będzie trajektoria członu manipulatora opisana krzywą kwaternionową, to wyznaczenie kartezjańskich współrzędnych krzywej będzie wymagać zastosowania podobnego przekształcenia, które opisane jest równaniem (4-135). Jak to zostało pokazane w rozdziale 4.2.2, złożenie orientacji reprezentowanych przez kwaternion jest realizowane przez mnożenie kwaternionów w ustalonej kolejności

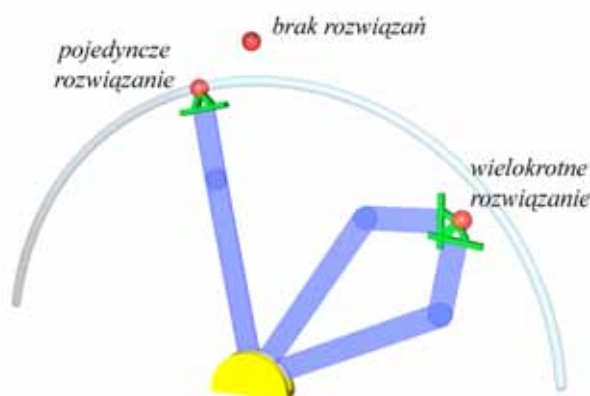
$${}^0_n T = q_n q_2 q_1 \dots q_0 \quad (4-136)$$

Gdy rozpatrywany będzie ruch rzeczywistego manipulatora o niezerowej długości członów, do przekształcenia należy również dodać parametry każdego z członów. Problematyka prezentowana w pracy dotyczy przede wszystkim badania przestrzennych trajektorii kwaternionowych, więc najistotniejszym elementem przekształceń jest opis orientacji układów. Przekształcenie dla całkowitej orientacji układu łańcucha kinematycznego opisane jest równaniem (4-136), natomiast ewentualne pomocnicze pozostałe przekształcenia zrealizowane będą za pomocą przekształcenia jednorodnego (4-11).

**Zadanie kinematyki odwrotnej** (ang. *IK – inverse kinematics*) polega na wyznaczeniu parametrów przestrzeni konfiguracyjnej (a w dalszym kroku współrzędnych napędowych) na podstawie znajomości parametrów kartezjańskim opisujących docelowe położenie końcówki roboczej manipulatora. Określa się przestrzeń roboczą manipulatora, jako podzbiór punktów przestrzeni, które końcówka robocza może osiągnąć. Manipulacyjna przestrzeń robocza jest podzbiorem przestrzeni roboczej zawężonym do tych punktów, które manipulator osiągnąć może z dowolną orientacją. Określa się osiągalną przestrzeń roboczą manipulatora składającą się z punktów przestrzeni, które mogą być osiągnięte przynajmniej przy jednej orientacji. Aby zadanie kinematyki odwrotnej istniało, punkt docelowy (parametry przestrzeni kartezjańskiej) muszą należeć do osiągalnej przestrzeni roboczej. Problem istnienia rozwiązania przedstawiony jest na rysunku 4-17. Na rysunku zaznaczono przykładowe punkty, dla których istnieje rozwiązanie wielokrotne, rozwiązanie pojedyncze (ale niekoniecznie jest to punkt należący do manipulacyjnej przestrzeni roboczej, ponieważ człon roboczy osiąga je tylko przy jednej orientacji) oraz nie istnieje rozwiązanie zadania kinematyki odwrotnej.

Ze względu na swoją nieliniowość zadanie to jest znacznie bardziej złożone niż w przypadku problemu kinematyki prostej. Złożoność problemu rośnie wraz ze wzrostem liczby stopni swobody manipulatora. Praktyczna realizacja zadania wymaga uwzględnienia dodatkowych ograniczeń związanych z możliwościami ruchu członów manipulatora, przeszkód występujących na trasie, itp. Istnieje wiele algorytmów wyznaczających rozwiązanie zadania kinematyki odwrotnej. Dla prostych układów możliwe jest wyznaczenie jawnego rozwiązania w postaci analitycznej, to znaczy takiego, które nie wymaga iteracyjnych

procedur numerycznych [18]. Zarówno prosta i odwrotna kinematyka układów z redundancją została przedstawiona w pracy [75]. Różnego rodzaju metody heurystyczne znajdowania rozwiązań kinematyki odwrotnej powstały w systemach związanych z grafiką i animacją komputerową [35], [69]. Praca [4] przedstawia przegląd różnych metod rozwiązywania zadań kinematyki odwrotnej. W pracy [48] zadanie kinematyki odwrotnej rozpatrywane jest dla danych parametryzowanych w przestrzeni kwaternionów. Metody realizujące to zadanie różnią się od siebie wydajnością czasową i jakością prezentowanych rozwiązań (np. z punktu widzenia płynności wyznaczonego ruchu). W każdym jednak przypadku jako wynik otrzymuje się zestaw parametrów konfiguracyjnych układu dla zadanego punktu docelowego.



**Rysunek 4-17: Liczba rozwiązań zadanie kinematyki odwrotnej w zależności od położenia punktu docelowego (punkt docelowy zaznaczono kolorem czerwonym).**

Problemem bardziej ogólnym jest zadanie planowania ruchu manipulatora przy uwzględnieniu różnego rodzaju ograniczeń nałożonych na ciało. Celem jest wyznaczenie trajektorii w przestrzeni stanów spełniającej zadane ograniczenia przy znanym położeniu początkowym i końcowym. Szczegółowo metody planowania ruchu robotów i manipulatorów zostały zaprezentowane w pracach [25], [26]. Problematyka ta jest również rozpatrywana w złożonych zagadnieniach wyznaczania optymalnej trajektorii i sterowania dla mobilnych czujników używanych do estymacji procesów o parametrach rozłożonych [94], [100]. Ograniczenia tego typu są nieholonomiczne i dodatkowo zależą od otrzymanych wyników. W ten sposób odczytane przez czujniki wartości pozwalają na poprawienie oszacowania parametrów modelowanego procesu, co z kolei pozwala na wyznaczenie poprawionej trajektorii dla czujników.

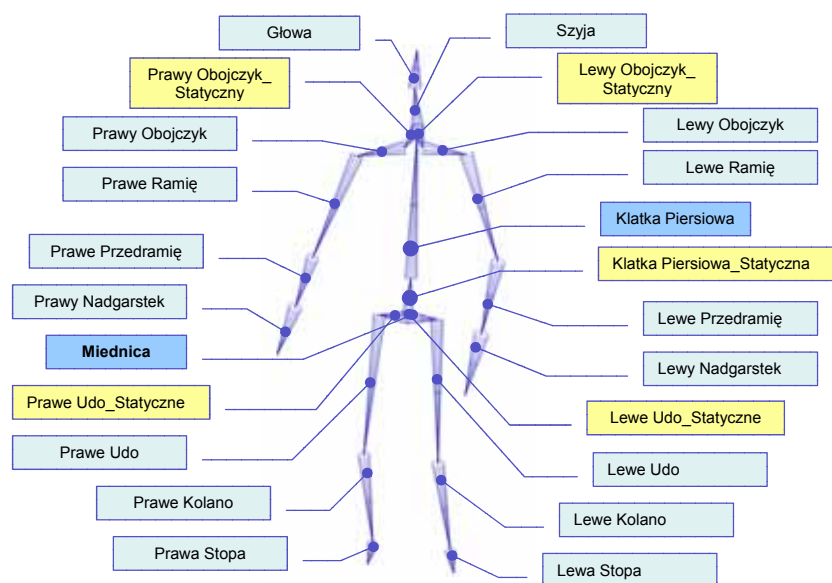
Zadania kinematyki odwrotnej oraz planowania ruchu nie będzie rozpatrywane w szczególności w prezentowanej pracy. Niemniej jednak algorytmy rozwiązujące te problemy mogą wspomagać metody zaproponowane w rozdziale 5.

#### 4.6.2 Przykład złożonego łańcucha kinematycznego

Szczególnym przypadkiem manipulatora o złożonym łańcuchu kinematycznym jest szkielet ludzki. Szkielet taki może zostać zamodelowany przy pomocy zbioru połączeń ruchowych niższego rzędu, przy czym każdy człon ruchomy reprezentuje wybrany fragment szkieletu (bardziej lub mniej uproszczony).

Poziom szczegółowości modelu zależy od konkretnego zastosowania. Szkielet ludzki składa się z 206 kości, z których tylko niektóre są połączone w sposób umożliwiający zmianę położenia. Do symulacji ruchu człowieka nie trzeba jednak dokładnie odwzorować pełnej struktury człowieka. Przykładowo ruch kości dłoni jest niezbędny do analizy języka migowego, jednak nie trzeba go uwzględniać przy analizie sposobu chodzenia postaci.

Połączenie szkieletowe kości realizuje złożony łańcuch kinematyczny o jednej kości bazowej, zazwyczaj reprezentowanej przez zespół kości miednicy. Każda para kości jest modelowana przez kuliste połączenie ruchowe o określonej liczbie stopni swobody. Przykładowa konstrukcja realizująca model szkieletu w postaci złożonego łańcucha kinematycznego przedstawiona jest na rysunku 4-18.



**Rysunek 4-18: Złożony łańcuch kinematyczny na podstawie hierarchicznego modelu szkieletu ludzkiego.**

W opisie kolorem ciemnoniebieskim zaznaczono kości, które są nadrzędne w stosunku do trzech innych kości. Kolor żółty oznacza kości, które nie biorą bezpośrednio udziału w ruchu – nie jest dla nich zapisywana trajektoria położenia. Mają jednak zdefiniowaną długość i statyczną orientację względem kości poprzedzającej, co wpływa na całkowite ukształtowanie szkieletu. Kości oznaczone kolorem jasnoniebieskim reprezentują główne elementy szkieletu. Wszystkie posiadają co najwyżej jeden człon następujący w hierarchii. Tego typu szkielet jest użyteczny do symulacji ruchu całej postaci bez uwzględnienia szczegółowego położenia np. kości dłoni lub stopy. Bardziej szczegółowe modele biorą przykładowo pod uwagę złożony model kręgosłupa człowieka [4]. Problem ten wykracza jednak poza ramy prezentowanej pracy, a do zastosowań zaprezentowanych w rozdziale 6 wystarczająca jest zaproponowana konstrukcja szkieletu.

Można wyróżnić zestaw podstawowych ograniczeń wynikających z własności fizycznych szkieletu, które powinny być uwzględnione przy modelowaniu ruchu postaci:

- każda kość (oprócz członu bazowego) jest powiązana w łańcuch kinematyczny z jedną kością poprzedzającą oraz może być powiązana z kośćmi następnymi w hierarchii szkieletu,

- dla każdej kości określona jest stała długość oraz statyczne położenie względem kości poprzedzającej w hierarchii szkieletu (zakłada się, że poza szczególnymi przypadkami, długości kości nie zmieniają się),
- dla każdego połączenia ruchowego określony jest zakresu możliwych do wykonania ruchów oraz stopni swobody (przykładowo kolano może zginać się tylko w jednej płaszczyźnie),
- dodatkowe ograniczenia pozwalające realizować dynamiczny model ruchu (np. masa kości, mięśni),
- model dynamiczny powinien również uwzględniać możliwości wygenerowania określonej siły przez układ mięśniowy związany z układem kostnym.

Ograniczenia wynikające z własności fizycznych powodują ograniczenie dziedziny dopuszczalnych ruchów. Znajomość oraz dobry sposób opisu ograniczeń czasoprzestrzennych pozwala na tworzenie realistycznych sekwencji ruchów postaci. Zmniejszenie przestrzeni rozwiązań jest także pomocne w zadaniu rozpoznawania ruchu trójwymiarowego na podstawie obrazu dwuwymiarowego [40].

Uwzględnienie wszystkich fizycznych uwarunkowań i oddziaływań prowadzi do skonstruowania w pełni analitycznego modelu szkieletu człowieka. Model taki trudny jest do praktycznej realizacji ze względu na bardzo wysoką złożoność problemu. Dlatego częściej stosuje się techniki przechwytywania ruchu, które powyższe ograniczenia mają automatycznie uwzględnione dzięki materiałowi źródłowemu, jakim jest wykonywanie ruchu przez żywego człowieka [20], [92]. Techniki przechwytywania ruchu (ang. *motion capture*) wspomagane są nie tylko przez analizę obrazów, ale przede wszystkim realizują ruch wybranych punktów na ciele człowieka. Przetwarzanie danych ruchu powinno uwzględniać przynajmniej częściowo powyższe ograniczenia. Uwzględnianie rozbudowanych ograniczeń może być zastosowane, jako rozwinięcie metod zaproponowanych w rozdziale 5. Pozwoliłoby to zwiększyć jakość oraz szybkość filtracji, analogicznie, jak to zostało wykorzystane w sprzężonej metodzie filtracji (rozdział 3).

Dla każdej części szkieletu określona jest funkcja czasu opisująca jego położenie. Ruch całej postaci modelowany przez hierarchiczny łańcuch kinematyczny jest więc wielowymiarową funkcją czasu zapisaną przy pomocy określonej reprezentacji. Do modelowania ruchu stosuje się różnego rodzaju reprezentacje danych, które wykorzystują różne parametryzacje przestrzeni położenia przedstawione w poprzednich rozdziałach. Szczegółowa analiza reprezentacji używanych w zadaniach symulacji ruchu ludzkiego przedstawiona jest w pracy [47], w której rozważane są między innymi następujące sposoby opisu danych:

- Reprezentacja oparta na **translacjach bezwzględnych AT** (ang. *Absolute Translations*). Informacja o położeniu kości zapisywana jest dla każdej kości jako bezwzględne przemieszczenie względem początku układu współrzędnych. Reprezentacja ta może być pomocna w niektórych przypadkach wizualizacji danych. Jednak użycie jej do przetwarzania danych wymaga dodatkowego sprawdzenia zachowania podstawowych ograniczeń czasoprzestrzennych (np. na stałą długość kości).
- Reprezentacja oparta na **translacjach względnych RT** (ang. *Relative Translations*). Informacje o położeniu kości jest zapisywana jako przesunięcie początku kości liczone względem początku

kości poprzedzającej w strukturze szkieletu. Reprezentacja ta może być pomocna do wykonywania niektórych obliczeń, jednak w praktyce rzadko kiedy jest stosowana. Przy jej używaniu również może wystąpić niespełnienie ograniczeń czasoprzestrzennych.

- Reprezentacja oparta na parametryzacji orientacji przez **macierze przekształceń lub kąty Eulera**. Zakłada się, że ruch poszczególnych kości zredukowany jest do rotacji wokół zdefiniowanych stawów. Rotacje te są zapisywane względem położenia końca kości poprzedzającej, więc nie występuje problem zachowania podstawowego ograniczenia czasoprzestrzennego na stałą długość kości. Reprezentacja ta dobrze się nadaje do wymiany danych pomiędzy różnymi formatami plików, a także do wizualizacji przy użyciu standardowych bibliotek graficznych.
- Reprezentacja oparta na rotacjach parametryzowanych **kwaternionami**. Podobnie, jak w przypadku reprezentacji opartej na macierzach przekształceń, również w tym przypadku położenie kości jest definiowane względem końca kości poprzedzającej za pomocą kwaternionu reprezentującego odpowiednią rotację. Reprezentacja ta jest powszechnie używana w różnych systemach graficznych i animacyjnych [80] ze względu na dobrze zdefiniowane własności interpolacyjne i brak występowania efektu utraty stopni swobody w przestrzeni kwaternionów.

Najwygodniejszy sposób opisu ruchu zakłada, że informacja o położeniu poszczególnych części łańcucha kinematycznego jest zapisywana względem członu poprzedzającego go w hierarchii. Sposób ten jest zaadaptowany z problemów kinematyki manipulatorów – w przypadku szkieletu ludzkiego kinematyka jest jednak bardziej skomplikowana. Zamiast szukania rozwiązań analitycznych stosuje się rozwiązania numeryczne i przetwarzanie już istniejących sekwencji. Przykładem takiego podejścia jest wspomniana technika *motion capture*.

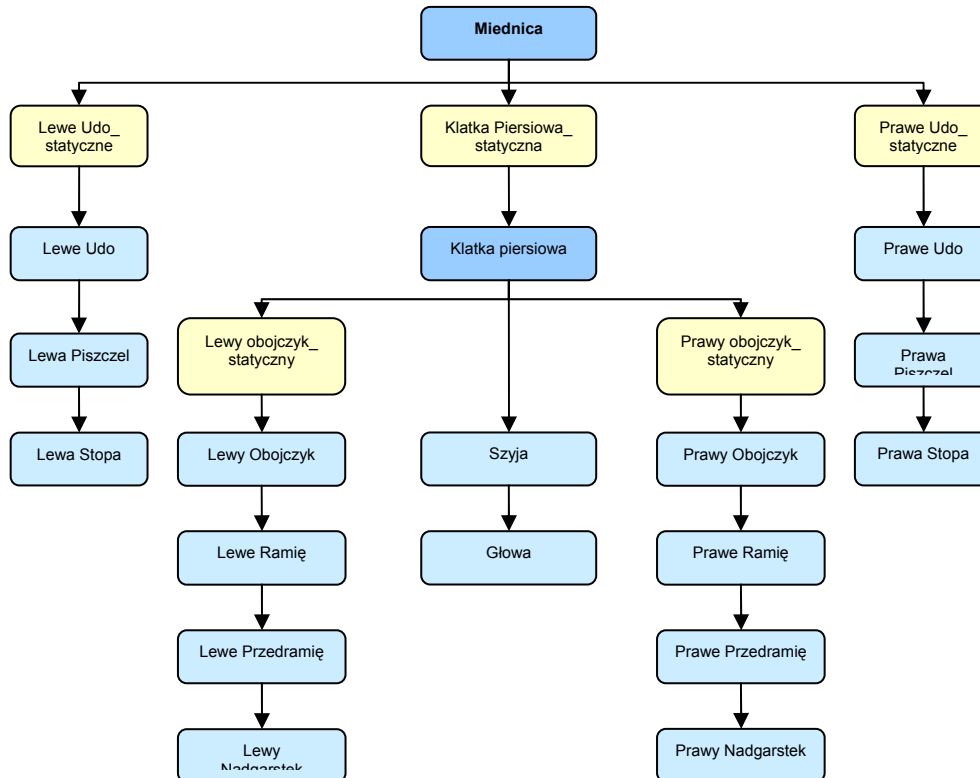
Zadanie kinematyki prostej dla hierarchicznego modelu szkieletu jest niezbędne do wszelkich operacji wizualizacji ruchu oraz wyznaczania bezwzględnych trajektorii wybranych kości. Zadanie kinematyki odwrotnej pozwala natomiast na wyznaczenie pozycji (rozumianej, jako zestaw parametrów konfiguracyjnych dla wszystkich kości), jaką ma przyjąć postać, aby zostały zachowane ograniczenia czasoprzestrzenne, a poszczególne części ciała osiągnęły zdefiniowane punkty w przestrzeni. Jest to jeden z przykładów znacznie bardziej skomplikowanych problemów ze względu na duży stopień złożoności opisu ciała w przestrzeni. Interesująca metoda zastosowania kinematyki odwrotnej do tworzenia sekwencji animacji postaci przedstawiona jest w [111] – autorzy proponują nałożenie dodatkowych ograniczeń czasoprzestrzennych przez zdefiniowanie punktów, które są kontrolowane, a dla pozostałych wyznaczane jest przybliżone rozwiązanie kinematyki odwrotnej szkieletu.

W dalszej części pracy przykładowym zastosowaniem proponowanych algorytmów będzie filtracja ruchu postaci przygotowanego za pomocą techniki *motion capture*. W tych zastosowaniach rozpatrywany będzie model szkieletu ludzkiego z następującymi założeniami:

- w szkielecie występują 23 kości połączone w niezmienną strukturę hierarchiczną przedstawioną na rysunku 4-19,



- każda kość ma określoną długość, która nie może się zmieniać,
- funkcja ruchu opisana jest za pomocą trajektorii w przestrzeni kwaternionów osobno dla każdej kości ruchomej,
- położenie opisane przy pomocy kwaternionu uwzględnia rotacyjną część transformacji względem kości poprzedzającej w szkielecie,
- jedyna informacja o translacji całego szkieletu względem początku globalnego układu współrzędnych jest zapisywana dla kości bazowej (*Miednica*),
- inne transformacje dla kości nie są dozwolone.



Rysunek 4-19: Hierarchia członów w modelu szkieletu ludzkiego.

Zaproponowany model szkieletu postaci jest tylko przykładem, w jaki sposób możliwe jest przedstawienie złożonego łańcucha kinematycznego w postaci hierarchii członów. W rozdziale pokazano, że szkielet taki może być rozpatrywany w sposób podobny do realizacji manipulatora o uproszczonej kinematyce. Do przetwarzania informacji o położeniu poszczególnych członów można więc używać tych samych metod, które używane są w zagadnieniach robotyki. Specyficzne zagadnienie modelowania ruchu ludzkiego będzie służyło jako jeden ze sposobów pokazania własności zaproponowanych metod filtracji trajektorii przestrzennych. Jest to możliwe dzięki zapisywaniu informacji o przekształceniu w szkielecie w postaci szeregu czasowego kwaternionów reprezentujących rotację względem elementu poprzedzającego w hierarchii. Tak zdefiniowane szeregi czasowe będą następnie rozpatrywane, jako dyskretna realizacja trajektorii przestrzennej zawartej w hipersferze jednostkowej  $S^3$ .

## Rozdział 5

### Propozycje przetwarzania trajektorii w przestrzeniach rotacji $S^3$

W rozdziale 4 przedstawiono różne sposoby zapisu orientacji oraz trajektorii reprezentujących rotacje lokalnego układu współrzędnych obiektu. W wielu zastosowaniach trajektorie takie są zapisywane w postaci cyfrowej na podstawie przekształcenia wykonywanego w świecie rzeczywistym. Problem ten występuje w aplikacjach związanych ze śledzeniem ruchu (ang. *motion tracking*), przechwytywaniem ruchu (ang. *motion capture*), analizą ruchu (ang. *motion analysis*) oraz w wielu innych. Należy podkreślić, że celem nie jest dwuwymiarowa reprezentacja ruchu, jako szeregu obrazów następujących po sobie (jak to jest robione w klasycznej kinematografii). Wyznaczenie reprezentacji trójwymiarowej pozwala na dalsze przekształcanie danych biorąc pod uwagę wszystkie uwarunkowania występujące w świecie rzeczywistym. Dzięki takiemu podejściu możliwe jest między innymi:

- analiza trajektorii wykonywanej przez manipulator z ewentualnym sprzężeniem zwrotnym – jest to niezbędne w sytuacji kontroli poprawności lokalnych czujników przemieszczenia członów manipulatora, lub gdy czujniki takie nie mogą być zastosowane,
- analiza ruchu z punktu widzenia poprawności jego wykonania – zastosowanie to może być użyte w systemie diagnostycznym sprawdzającym poprawność budowy układu ruchowego człowieka,
- analiza ruchu w celu określenia jego zawartości znaczeniowej (rozpoznawanie czynności, emocji, identyfikacja biometryczna, itp.),
- przetwarzanie ruchu w celu generacji nowych sekwencji opartych na sekwencji oryginalnej,
- odtwarzanie ruchu przez wirtualne postacie wygenerowane i narysowane syntetycznie – podejście takie jest stosowane wielokrotnie w produkcjach filmowych.

Zapisanie ruchu w postaci cyfrowej tak, aby był on reprezentowany jako przekształcenie w przestrzeni trójwymiarowej jest realizowane w różny sposób. Można wyróżnić dwie główne grupy metod pozwalających na rekonstrukcję w przestrzeni wirtualnej rzeczywistego trójwymiarowego obiektu:

- Tradycyjne podejście oparte na zapisywaniu położenia wybranych punktów zlokalizowanych na powierzchni obiektu. Podejście to było intensywnie stosowane i rozwijane w ramach techniki przechwytywania ruchu *motion capture*. W praktyce stosuje się dwa rodzaje systemów: optyczne [41] – wykorzystujące kamery do śledzenia markerów odbłaskowych oraz magnetyczne [39] –

wyznaczające zmianę położenia odpowiednich markerów w zaburzonym przez nie polu magnetycznym. W przypadku śledzenia ruchu człowieka markery przyłączone są zazwyczaj do najważniejszych stawów wyznaczających osie wykonywanego obrotu. Spotyka się również propozycje użycia innego rodzaju obliczania transformacji dokonanych przez wybrane części ciała (np. mechaniczne czujniki przemieszczenia). Interesującym rozwiązaniem jest odblaskowy kostium pozwalający na rekonstrukcję położenia obiektu (postaci) z dużą rozdzielczością [96]. Wyznaczenie pozycji markerów prowadzi do stworzenia hierarchii obiektu przedstawiającej złożony łańcuch kinematyczny. Aby wyznaczyć położenie w przestrzeni trójwymiarowej należy zebrać informację z kilku niezależnych źródeł danych o znanych położeniach względnych (np. kilku kamer śledzących markery odblaskowe). W komercyjnych systemach tego typu często używa się nadmiarowej liczby kamer (nawet do 24), aby umożliwić śledzenie markerów w sytuacji, gdy są one częściowo zasłonięte przez różne części obiektu.

- Metody oparte na analizie kształtów wyznaczonych na podstawie obrazu dwuwymiarowego pozwalają na wyznaczenie położenia obiektu bez używania dodatkowych markerów [10], [83]. W wielu przypadkach metody te wykorzystują różne algorytmy wyznaczające położenie poruszającego się obiektu na nieruchomym tle. Dalsza analiza położenia oparta jest na analizie kształtu w odniesieniu do określonej hierarchii (na przykład szkieletowej), która w sposób uproszczony reprezentuje obiekt. Najczęściej podejście to wymaga dodatkowego określenia początkowej pozycji obiektu na obrazie. Użycie pojedynczej kamery może być wystarczające dla specyficznego przemieszczenia obserwowanego z kierunku prostopadłym do kierunku ruchu. Często przeprowadzana jest skorelowana analiza dla kilku kamer obserwujących śledzony obiekt pod różnymi kątami.

Każda metoda śledzenia ruchu i zapisywania go w postaci trójwymiarowej jest obciążona niedokładnościami oraz szumem wprowadzanym przez różne elementy systemu śledzącego. Praktyczne próby zastosowania takich systemów wskazują na dwa rodzaje zakłóceń. Szum statyczny występuje nawet dla nieporuszających się obiektów, co jest spowodowane zakłóceniami wprowadzanymi przez urządzenia pomiarowe (lub niedokładnościami wprowadzanymi przez algorytmy przetwarzające dane) [28]. Szum dynamiczny związany jest z ruchem obiektu przykładowo przez opóźnienia w torze przetwarzającym systemu.

W prezentowanej pracy nie są uwzględniane metody wczesnego przetwarzania ruchu, które stanowią odrębną bardzo rozbudowaną dziedzinę. Wśród nich można przykładowo znaleźć metody, których celem jest wyznaczenie optymalnych parametrów kamery w problemie śledzenia obiektów [21].

Ze względu na rosnącą popularność systemów przetwarzających ruch do postaci trójwymiarowej, istnieje coraz większa konieczność opracowania skutecznych algorytmów pozwalających na zmniejszenie zaszumienia sygnału reprezentującego ruch. Jak to zostało pokazane w rozdziale 4, sygnał opisujący trajektorię obiektu składa się z zapisu translacji oraz orientacji. Problem przetwarzania trajektorii trójwymiarowych reprezentujących translację jest dobrze zbadany i opracowany. W pracy główny nacisk

położony zostanie na metodach realizujących odszumianie trajektorii orientacji lokalnego układu współrzędnych obiektu. Tematyka filtracji trajektorii orientacji nie jest jeszcze dokładnie przebadana. Jakkolwiek pojawiające się prace świadczą o dużym zainteresowaniu i zapotrzebowaniu na tego typu metody. Zostanie zaprezentowany zbiór najważniejszych prac oraz kierunków badań prowadzonych w dziedzinie odszumiania trajektorii orientacji układu. W kolejnych rozdziałach zostanie przedstawiona propozycja metod filtracji będących uogólnieniem algorytmów opartych na rozwiązywaniu równań różniczkowych cząstkowych. Zostanie pokazana zależność pomiędzy wybranymi istniejącymi metodami filtracji ruchów, a ogólnym modelem opartym na równaniu przewodnictwa ciepła. Model ten pozwoli następnie na skonstruowanie nowych nieliniowych filtrów dla trajektorii orientacji układu reprezentowanej w przestrzeni kwaternionów jednostkowych  $S^3$ .

## 5.1 Przegląd istniejących metod przetwarzania trajektorii w przestrzeniach orientacji

Istniejące metody filtracji orientacji układu można podzielić na dwie główne grupy: oparte na konstrukcji liniowych filtrów dla szeregu czasowego będących odpowiednikiem dyskretnego splotu w przestrzeni rzeczywistej oraz oparte na bezpośredniej minimalizacji energii sygnału najczęściej przez rodzaj optymalizacji gradientowej. Zostaną również krótko zaprezentowane pozostałe metody, które wykorzystują specyficzne indywidualne podejście. Metody zaproponowane w rozdziale 5.2 tworzą nową, trzecią grupę opartą na uogólnieniu rozwiązania równania różniczkowego procesu dyfuzji. Zostanie pokazane, że realizowana jest w ten sposób minimalizacja energii sygnału oraz że filtry z rozdziału 5.1.1 są specyficzną realizacją filtrów proponowanych w rozdziale 5.2.

### 5.1.1 Metody oparte na liniowym przetwarzaniu szeregu czasowego w przestrzeni stycznej $R^3$

Filtracja trajektorii w tej grupie metod jest realizowana przez odpowiednią adaptację klasycznego liniowego filtra stacjonarnego (LTI – ang. *Linear Time Invariant*). Niech  $p_i \in R^3$  będzie  $i$ -tym elementem sygnału  $P$  opisującego trajektorię w rzeczywistej przestrzeni  $R^3$ . Definiuje się wektor współczynników reprezentujący maskę filtra  $(a_{-k}, \dots, a_0, \dots, a_k)$ , co prowadzi do odpowiednika dyskretnego splotu sygnału prezentowanego w rozdziale 2.2 dla obrazów. Przekształcenie (filtr)  $F$  dla elementu  $p_i$  określa się jako:

$$F(p_i) = a_{-k}p_{i-k} + \dots + a_0p_i + \dots + a_kp_{i+k} \quad (5 - 1)$$

### Symetryczna lokalna parametryzacja przestrzeni $S^3$ (Lee, 2002)

Konstrukcja filtrów czasowych ([64], [65], [66]) dla trajektorii w przestrzeni orientacji  $S^3$  oparta jest na odpowiednim przekształceniu szeregu czasowego kwaternionów do przestrzeni  $R^3$ . Niech  $Q$  reprezentuje dyskretny szereg czasowy składający się z kwaternionów jednostkowych  $q_i$ . Każdy element tego szeregu

można przedstawić, jako złożenie przekształceń od położenia opisanego przez element początkowy. Złożenie przekształceń realizowane jest w przestrzeni kwaternionów przez odpowiedni iloczyn dwóch elementów:

$$q_i = (q_0 q_0^{-1})(q_1 q_1^{-1}) \cdots (q_{i-1} q_{i-1}^{-1}) q_i = q_0 (q_0^{-1} q_1)(q_1^{-1} q_2) \cdots (q_{i-1}^{-1} q_i) \quad (5-2)$$

Prędkość kątowa ruchu wykonywanego wzdłuż linii geodezyjnej łączącej dwa kwaterniony może być wyrażona, jako:

$$\omega_j = \log(q_j^{-1} q_{j+1}), \quad (5-3)$$

gdzie  $q_j$  oznacza kwaternion początkowy, a  $q_{j+1}$  kwaternion końcowy. Wyrażenie (5-2) przyjmuje postać:

$$q_i = q_0 \prod_{j=0}^{i-1} \exp(\omega_j) \quad (5-4)$$

Celem przekształcenia jest znalezienie wektorowego odpowiednika  $P$  sygnału  $Q$ , dla którego liniowe przemieszczenie  $p_{i+1} - p_i$  będzie odpowiadać przemieszczeniu kątowemu  $\log(q_i^{-1} q_{i+1})$ . Szukane przekształcenie jest realizowane przez następujące wzajemnie odwrotne wyrażenia:

$$p_i = p_0 + \sum_{j=0}^{i-1} \log(q_j^{-1} q_{j+1}) \quad (5-5)$$

$$q_i = q_0 \prod_{j=0}^{i-1} \exp(p_{j+1} - p_j) \quad (5-6)$$

Filtracja zaproponowana w cytowanej pracy [66] realizowana jest przez następujące złożenie operacji:

$$\hat{F}(q_i) = q_i \exp(F(p_i) - p_i), \quad (5-7)$$

gdzie  $F$  jest filtracją typu  $LTI$  opisaną równaniem (5-1) działającą na rzeczywistym szeregu czasowym  $p_i$  otrzymanym za pomocą równania (5-5). Właściwości filtru  $\hat{F}$  będą bezpośrednio zależeć od przyjętych współczynników maski  $a_m$  w filtrze  $F$ . Dalsze przekształcenia zależności prowadzą do następującej postaci:

$$\hat{F}(q_i) = q_i \exp\left(\left(\sum_{m=-k}^{m=k} a_m p_{i+m}\right) - p_i\right) = q_i \exp\left(\sum_{m=-k}^{m=k} a_m (p_{i+m} - p_i)\right) = q_i \exp\left(\sum_{m=-k}^{m=k} a_m p_m^i\right), \quad (5-8)$$

gdzie  $p_m^i = p_{i+m} - p_i$  (oznaczenia zgodne z cytowaną pracą) jest odpowiednikiem kąowego przemieszczenia parametryzowanego w przestrzeni rzeczywistej. Wartość ta może być zapisana jako:

$$p_m^i = \begin{cases} \sum_{j=1}^m \log(q_{i+j-1}^{-1} q_{i+j}) & \text{dla } m \geq 1 \\ 0 & \text{dla } m = 0 \\ \sum_{j=m}^{-1} -\log(q_{i+j}^{-1} q_{i+j+1}) & \text{dla } m \leq -1 \end{cases} \quad (5-9)$$

Równanie (5-8) można przedstawić w postaci uwzględniającej prędkość kątową określoną w dziedzinie szeregu czasowego kwaternionów, jako  $\omega_i = \log(q_i^{-1} q_{i+1})$  (dokładne przekształcenie znajduje się w [66]):

$$\hat{F}(q_i) = q_i \exp\left(\sum_{m=-k}^{m=k-1} b_m \omega_{i+m}\right), \quad (5-10)$$

gdzie

$$b_m = \begin{cases} \sum_{j=m+1}^k a_j & \text{dla } 0 \leq m \leq k-1 \\ \sum_{j=-k}^{k-1} -a_j & \text{dla } -k \leq m < 0 \end{cases} \quad (5-11)$$

Filtracja realizowana jest w rzeczywistej przestrzeni stycznej  $R^3$  do hipersfery jednostkowej w punkcie  $q_i$ . Parametryzacja przestrzeni jest przeprowadzana lokalnie (patrz rozdział 4.2.3), dzięki czemu unika się problemów z określeniem odpowiedzi filtru dla punktów osobliwych w przestrzeni. Wyrażenie (5-10) jest prawidłowe przy założeniu, że następujące po sobie przemieszczenia są względnie małe, to znaczy, że ich odległość kątowa nie przekracza  $\pi$  (co w praktyce jest zawsze spełnione). Przetwarzanie przy użyciu maski dotyczy nie pojedynczych punktów  $q_i$ , ale przemieszczenia opisanego prędkością kątową  $\omega_i$ . Prędkość kątowa jest wyznaczana jako iloczyn kwaternionów (zamiast różnicy położenia), co zapewnia, że ostateczny wynik operacji należy do hipersfery jednostkowej w przestrzeni kwaternionów.

Na podstawie schematu konstrukcji filtrów opisanego równaniem (5-10) w pracy [66] wprowadzone są różnego rodzaju filtry przy zastosowaniu standardowych masek. W ten sposób otrzymuje się następujące wyrażenia na realizację poszczególnych filtrów:

- Filtr wygładzający (ang. *smoothing*)

$$\hat{F}_S(q_i) = q_i \exp\left(\frac{\lambda}{24}(\omega_{i-2} - 3\omega_{i-1} + 3\omega_i - \omega_{i+1})\right) \quad (5-12)$$

- Filtr rozmywający – odpowiednik jądra gaussowskiego (ang. *blurring*)

$$\hat{F}_B(q_i) = q_i \exp\left(\frac{1}{16}(-\omega_{i-2} - 5\omega_{i-1} + 5\omega_i + \omega_{i+1})\right) \quad (5-13)$$

- Filtr wyostrzający – podbija wysokie częstotliwości (ang. *sharpening*)

$$\hat{F}_U(q_i) = q_i \exp\left(\frac{\lambda}{16}(\omega_{i-2} + 3\omega_{i-1} - 3\omega_i - \omega_{i+1})\right) \quad (5-14)$$

W przedstawionych wyrażeniach współczynnik  $\lambda$  określa szybkość przeprowadzanego procesu filtracji.

Praktyczne zastosowanie cytowanego schematu filtracji jest przedstawione w pracach [64], [65] do wielorozdzielczej analizy ruchu (realizującej częściowo koncepcję odpowiednika przestrzeni skal w postaci hierarchicznej konstrukcji ciągu filtrów) oraz w pracy [57] do znakowania ruchu (ang. *motion watermarking*).

### Asymetryczna lokalna parametryzacja przestrzeni $S^3$ (Fang et al, 1998)

Podstawową różnicą w procesie filtracji zaproponowanym w pracy [28] jest sposób przetwarzania pojedynczej próbki danych. Autorzy pracy proponują, aby element trajektorii wynikowej był wyznaczany, jako złożenie przemieszczeń opisanych prędkościami kątowymi, od położenia początkowego:

$$\hat{F}(q_i) = q_0 \exp\left(\prod_{m=1}^{i-1} \tilde{\omega}_m \Delta t\right), \quad (5-15)$$

gdzie  $\tilde{\omega}_m$  jest elementem przetworzonej trajektorii w dziedzinie prędkości kątowej:

$$\tilde{\omega}_m = A[L(\omega_m)] \quad (5-16)$$

Autorzy proponują w cytowanej pracy zastosowanie dwóch rodzajów filtrów: filtru dolnoprzepustowego  $L$  oraz adaptacyjnego  $A$ . Filtracja dolnoprzepustowa może być zrealizowana przez dyskretny spłot elementów szeregu czasowego z wybraną maską (analogicznie, jak w poprzednim przypadku) lub przez iloczyn współczynników w dziedzinie częstotliwości (po zastosowaniu transformaty Fouriera do sygnału prędkości kątowej). Filtr adaptacyjny ma na celu zminimalizowanie zakłóceń wynikających z opóźnienia wprowadzanego przez tor przetwarzający sygnał ruchu (bezwładność czujników, szybkość przetwarzania toru wizyjnego w przypadku systemu optycznego, itp.). W szczególności autorzy proponują zastosowanie algorytmu opartego na pracy Burga [13] w celu estymacji współczynników predykcji. Ponieważ przedstawiona metoda nie jest bezpośrednio związana ze specyficzną dziedziną kwaternionów, nie będzie ona tutaj w całości cytowana.

W porównaniu do poprzednio prezentowanej metody, filtracja sygnału również odbywa się w przestrzeni stycznej do hipersfery jednostkowej  $S^3$ . Jednak w tym przypadku punktem styczności jest początek trajektorii. Jak podane zostało w pracy [66], podejście takie może być obarczone błędem przetwarzania kumulującym się wzdłuż trajektorii. Wyznaczenie przestrzeni stycznej do punktu  $q_i$  (zamiast do punktu  $q_0$ ) powoduje, że stosowany filtr jest symetryczny. Dzięki temu błędy pojawiające się w danych źródłowych nie są propagowane wzdłuż całej trajektorii. Wymaga to oczywiście wyznaczenia innych współczynników dla operacji dyskretnego spłotu, co zostało przedstawione w poprzednim podpunkcie.

### 5.1.2 Iteracyjne metody minimalizacji energii trajektorii

Podstawą konstrukcji metod iteracyjnych jest określenie funkcji celu, dla której wykonywana jest następnie optymalizacja. Proces jest zatrzymywany po osiągnięciu minimum lokalnego funkcji. Najczęściej w metodach iteracyjnych przyjmuje się, jako funkcję celu, miarę określającą gładkość krzywej  $p$  zdefiniowanej dla krzywych w przestrzeni rzeczywistej, jako:

$$E(p) = \int_p \left\| \frac{d^2 p}{dt^2} \right\|^2 dt \quad (5-17)$$

Określenie to oznacza całkowitą energię sygnału rozumianą, jako przyspieszenie, w przypadku, gdy trajektoria sparametryzowana jest zmienną czasową. W pracy [63] zostało pokazane, że w przestrzeni kwaternionów bezpośrednie zastosowanie miary (5-17) nie daje dobrych rezultatów. Jako przykład można przytoczyć trajektorię wyznaczoną wzdłuż linii geodezyjnej, wzdłuż której punkt porusza się ze stałą prędkością kątową (odpowiednik poruszania się wzdłuż linii prostej w przestrzeni rzeczywistej). Miara energii, jako całki z kwadratu przyspieszenia powinna dać wartość 0, a zamiast tego otrzymuje się wartość 1.

W wielu pracach ([42], [55], [63]) proponowane jest podejście alternatywne, w którym brana pod uwagę jest prędkość określona w rzeczywistej przestrzeni stycznej do hipersfery jednostkowej. Otrzymuje się następującą funkcję określającą energię sygnału kwaternionów:

$$E(q) = \int_q \left\| \frac{d\omega(t)}{dt} \right\|^2 dt, \quad (5-18)$$

co pozwala uzyskać miarę o wartości 0 dla przypadku ruchu o stałej prędkości kątowej. W przypadku trajektorii opisanej szeregiem czasowym, na podstawie zależności (4-93) lokalna prędkość kątowa oraz jej pierwsza pochodna są równe odpowiednio:

$$\omega_i = \frac{\log(q_i^{-1}q_{i+1})}{h} \quad (5-19)$$

$$\frac{d\omega_i}{dt} = \frac{\log(q_i^{-1}q_{i+1}) - \log(q_{i-1}^{-1}q_i)}{h^2}, \quad (5-20)$$

gdzie  $h$  jest krokiem dyskretyzacji przy użyciu jawnego schematu różnicowego. Wyrażenie na energię sygnału (gładkość trajektorii)  $q$  przyjmuje postać:

$$E(Q) = \sum_{i=1}^{n-1} \left\| \frac{d\omega_i}{dt} \right\|^2 = \sum_{i=1}^{n-1} \left\| \frac{\log(q_i^{-1}q_{i+1}) - \log(q_{i-1}^{-1}q_i)}{h^2} \right\|^2 \quad (5-21)$$

W przypadku rozpatrywania całkowitej energii ruchu składającej się ze składowej translacyjnej  $p \in R^3$  oraz rotacyjnej  $q \in S^3$  otrzymuje się wyrażenie:

$$E(p) + E(q) = \sum_{i=1}^{n-1} \left\| \frac{d^2 p_i}{dt} \right\|^2 + \sum_{i=1}^{n-1} \left\| \frac{d\omega_i}{dt} \right\|^2 = \sum_{i=1}^{n-1} \left\| \frac{p_{i+1} - 2p_i + p_{i-1}}{h^2} \right\|^2 + \sum_{i=1}^{n-1} \left\| \frac{\log(q_i^{-1}q_{i+1}) - \log(q_{i-1}^{-1}q_i)}{h^2} \right\|^2 \quad (5-22)$$

Ponieważ prezentowana praca dotyczy przede wszystkim trajektorii w przestrzeniach orientacji, składowa translacyjna nie będzie w dalszej części rozpatrywana. Ponadto techniki przetwarzania trajektorii w przestrzeni rzeczywistej są znane i dobrze opracowane.

### Operatory wygładzające (Lee, 1996)

Aby zminimalizować energię opisaną równaniem (5-22) w pracy [63] proponuje się użycie operatorów wygładzających  $F_m$  rzędu  $m$ , dla których działanie na trajektorii  $R = (p, q)$  wyrażone jako

$$\tilde{R} = F_m(R) \quad (5-23)$$

spełnia warunek

$$E_m(\tilde{R}) < E_m(R), \quad (5-24)$$

gdzie  $E_m$  oznacza funkcję energii trajektorii rzędu  $m$ . Równanie (5-22) definiuje funkcję energii dla trajektorii  $R$  rzędu 2. Jest to więc operator, który minimalizuje energię będącą miarą gładkości krzywej w określonej przestrzeni położenia. Na podstawie energii położenia można wyznaczyć zależność, dla której energia przyjmuje wartość minimalną równą 0. Wartość ta jest osiągnięta dla punktu, w którym  $d\omega_i/dt = 0$ .

Po odpowiednich przekształceniach otrzymuje się następującą zależność na iteracyjną zależność minimalizującą energię trajektorii w przestrzeni  $S^3$ :



$$\tilde{q} = F_m(q), \quad (5-25)$$

gdzie dla każdego elementu  $q_i$  trajektorii  $Q$  określa się następującą operację:

$$\tilde{q}_i = \begin{cases} q_1 q_2^{-1} q_1 & \text{dla } i = 0 \\ q_{i+1} q_i^{-1} q_{i-1} & \text{dla } 0 < i < n-1 \\ q_{n-2} q_{n-3}^{-1} q_{n-2} & \text{dla } i = n-1 \end{cases} \quad (5-26)$$

Aby zapewnić stabilność procesu wprowadza się dodatkowe czynniki tłumiące w postaci interpolacji typu *slerp* o współczynniku  $\lambda \in (0,1]$  pomiędzy nowo wyznaczoną trajektorią, a trajektorią oryginalną (na podstawie równania (4-83)):

$$\hat{q} = \text{slerp}(q, F_m(q), \lambda) \quad (5-27)$$

Proces jest zbieżny dla małych wartości prędkości kątowych  $\omega_i$ . Dla dużych przemieszczeń może się zdarzyć, że zostaną osiągnięte punkty osobliwe przestrzeni.

W sposób analogiczny, autorzy pracy [63] wprowadzają operatory wygładzające wyższych rzędów. W każdym przypadku jest to proces iteracyjny, dla którego pojedynczy krok został wyznaczony na podstawie kryterium minimalizacji energii reprezentującej gładkość trajektorii określonego rzędu. Proces można zakończyć po zdefiniowanej z góry liczbie iteracji lub po osiągnięciu jednego z kryteriów zatrzymania, np.

$$\max_i (d(q_i^{(0)}, q_i^{(k)})) < \varepsilon, \quad (5-28)$$

gdzie  $d(\cdot, \cdot)$  jest kwaternionową miarą odległości,  $q_i^{(k)}$  oznacza  $k$ -tą iterację dla  $i$ -tego elementu trajektorii.

### Optymalizacja gradientowa (Kim et al, 1998)

W porównaniu do filtracji przedstawionej w pierwszym podpunkcie, w pracy [55] brana jest pod uwagę funkcja celu złożona z dwóch składowych, co prowadzi do następującego zadania nieliniowej optymalizacji:

$$\min G(\tilde{q}) = d(q, \tilde{q}) + \beta E(\tilde{q}), \quad (5-29)$$

gdzie  $\tilde{q}$  oznacza sygnał po filtracji,  $E(\tilde{q})$  jest zdefiniowana równaniem (5-22), a  $d(q, \tilde{q})$  jest odpowiednikiem odległości pomiędzy trajektorią oryginalną, a przefiltrowaną:

$$d(q, \tilde{q}) = \sum_{i=0}^{n-1} \left\| \log(q_i^{-1} \tilde{q}_i) \right\|^2 \quad (5-30)$$

Współczynnik  $\beta$  określa wpływ normy energetycznej nad normą odległościową w mierze całkowitej. Dla małych wartości współczynnika rozwiązaniem problemu optymalizacji byłaby trajektoria zbliżona do oryginalnej. Dla większych wartości współczynnika, składowa energetyczna staje się dominująca, co powoduje, że optymalizacja zmierza do linii prostej w przestrzeni kwaternionów jednostkowych. Zmiana współczynnika  $\beta$  pozwala na określenie, w jakim stopniu należy przeprowadzić wygładzanie, a w jakim należy uwzględnić trajektorię oryginalną.

Algorytm proponowany w cytowanej pracy opiera się na optymalizacji gradientowej realizowanej w trzech krokach:

$$\text{Inicjalizacja} \quad q^{(0)} \leftarrow q \quad (5 - 31)$$

$$\text{Iteracja} \quad q^{(k+1)} = q^{(k)} \exp(-\lambda \nabla G(q^{(k)})) \quad (5 - 32)$$

$$\text{Kryterium stopu} \quad \|\nabla G(q^{(k)})\| < \varepsilon \quad (5 - 33)$$

Minimalizacja energii jest więc przeprowadzana analogicznie, jak w klasycznym podejściu realizowanym w przestrzeni rzeczywistej. W tym przypadku jednak zamiast operacji zmniejszającej wartość funkcji wzdłuż kierunku gradientu, wykonuje się operacje zmiany orientacji kwaternionu realizowaną przez iloczyn z wynikiem uzyskanym w przestrzeni stycznej (mapa eksponencjalna – równanie (4 – 79) i następne). Zastosowana procedura zapewnia, że otrzymany wynik należy do hipersfery jednostkowej  $S^3$ . Kryterium stopu jest realizowane również na podstawie normy gradientu wyznaczonego w każdym kroku iteracyjnym.

Podobny schemat optymalizacyjny został użyty w pracy [101], w której postać funkcji celu jest bardziej skomplikowana. W pracy minimalizacja dotyczy czterech kryteriów: translacji, orientacji, prędkości liniowej oraz prędkości kątowej. Dla wielkości opisanych w przestrzeni rzeczywistej krok iteracyjny jest realizowany w standardowy sposób. W przypadku opisu orientacji w przestrzeni kwaternionów, minimalizacja również odbywa się w przestrzeni stycznej. W cytowanej pracy filtracja jest używana do śledzenia obiektów i wyznaczania ich położenia na podstawie obrazów dwuwymiarowych.

### 5.1.3 Pozostałe metody filtracji w przestrzeni kwaternionów

#### Algorytmy genetyczne (Hsieh , 2003)

Rozwinięcie optymalizacji gradientowej przedstawionej w rozdziale 5.1.2 jest metoda hybrydowa proponowana w pracy [43]. Proponowane podejście zakłada dwustopniowy schemat filtracji: optymalizację globalną przy użyciu algorytmów genetycznych, a następnie w wyróżnionych obszarach przestrzeni rozwiązań nieliniową optymalizację lokalną. Zarówno użyta funkcja kryterialna, jak i schemat optymalizacji lokalnej są użyte w tej samej formie, jak w pracy [55]. Nowością w pracy jest zastosowanie algorytmów genetycznych. Autorzy przedstawiają przykładowe wyniki działania algorytmu, z których wynika, że zastosowanie wstępnej globalnej optymalizacji poprawia jakość filtracji o około 15% (dla kryterium minimalizacji całkowitej energii trajektorii).

#### Kodowanie

W rozważanym problemie optymalizacji opisanym równaniem (5 – 29), nowe położenie w przestrzeni orientacji jest otrzymywane przez wprowadzenie przemieszczenia zapisanego w przestrzeni stycznej do hipersfery jednostkowej:

$$\Theta_i = \log(q_i^{-1} \tilde{q}_i) = [\theta_{3i} \theta_{3i+1} \theta_{3i+2}]^T, \quad (5 - 34)$$

Gdzie  $\theta_{3i+j} \in R$ ,  $i \in [0, n-1]$ ,  $j \in [0, 2]$ . Dla każdej składowej przemieszczenia  $\theta_{3i+j}$  tworzy się reprezentację binarną o określonej długości  $d$  (co wymaga określenia dziedziny dla parametrów  $\theta_{3i+j}$ ). Dla  $n$  elementów w szeregu kwaternionów otrzymuje się bitowy chromosom o długości  $3nd$ . Cytując za pracą [43] przyjmuje się oznaczenie  $Q_i^{(k)}$  na  $i$ -te rozwiązanie dopuszczalne otrzymane w  $k$ -tej iteracji.

### Funkcja przystosowania

Określenie funkcji przystosowania oparte jest na nieliniowym określeniu energii sygnału opisanej równaniem (5 – 29). Przyjmując, że maksymalna wartość energii występująca w trajektorii wynosi:

$$G_{\max} = \max_i \{G(q_i^{(k)})\}, \quad (5 - 35)$$

otrzymuje się następującą postać funkcji przystosowania

$$f_F(q_i^{(k)}) = G_{\max} - G(q_i^{(k)}) \quad (5 - 36)$$

Aby zapobiec sytuacji, gdy populacja zostanie zdominowana przez osobniki o bardzo dużej wartości funkcji przystosowania, wprowadza się skalowanie, które pozwoli na kontrolę licznosci najlepszych osobników w populacji:

$$\hat{f}_F(Q_i^{(k)}) = \alpha f_F(Q_i^{(k)}) + \beta \quad (5 - 37)$$

Na wartości współczynników liniowego skalowania  $\alpha, \beta$  wprowadzone są następujące ograniczenia:

$$\sum_{i=1}^m \hat{f}_F(q_i^{(k)}) = \sum_{i=1}^m f_F(q_i^{(k)}) \quad (5 - 38)$$

$$\max_i \hat{f}_F(q_i^{(k)}) = \frac{\nu \sum_{i=1}^m \hat{f}_F(q_i^{(k)})}{m}, \quad (5 - 39)$$

gdzie  $\nu$  oznacza liczbę oczekiwanych osobników w najlepszym chromosomie, a  $m$  jest rozmiarem populacji.

Aby dodatkowo zwiększyć obszar przeszukiwania w przestrzeni rozwiązań dopuszczalnych, zmniejsza się wartość funkcji przystosowania dla osobników znajdujących się w małej odległości. Funkcja przystosowania przyjmuje postać:

$$\tilde{f}_F(q_i^{(k)}) = \frac{\hat{f}_F(q_i^{(k)})}{\sum_{i \neq j} S(q_i^{(k)}, q_j^{(k)})}, \quad (5 - 40)$$

gdzie  $S$  jest funkcją określającą bliskość dwóch osobników. Funkcja w przestrzeni kwaternionów może być zrealizowana jako:

$$S(q_i^{(k)}, q_j^{(k)}) = 1 - \frac{\left\| \log \left( (q_i^{(k)})^{-1} q_j^{(k)} \right) \right\|}{S_{\max}}, \quad (5 - 41)$$

$$S_{\max} = \max_{i,j} \left\| \log \left( q_i^{(k)} \right)^{-1} q_j^{(k)} \right\| \quad (5 - 42)$$

Równania (5 – 37) – (5 – 42) nie są podane w pracy [43] dla elementów z przestrzeni kwaternionów, lecz zostały wyprowadzone na podstawie zależności umieszczonych w pracy dla elementów trajektorii w przestrzeni rzeczywistej.

### Metody selekcji i operatory genetyczne

W cytowanej pracy proponowana jest selekcja oparta na prawdopodobieństwie wyboru chromosomu proporcjonalnym do względnej wartości funkcji przystosowania. Użyte są dwa operatory genetyczne: krzyżowanie (także w wersji wielopunktowej) oraz mutacja.

Praca została zacytowana, aby przedstawić pełny zakres podejść stosowanych do tej pory w dziedzinie filtrów trajektorii rotacji. Metody hybrydowe, wykorzystujące przykładowo algorytmy

genetyczne, mogą być rozwinięciem proponowanego w niniejszej rozprawie podejść. Jednakże ich dokładna analiza i badanie wykracza poza zakres prezentowanej pracy i może być punktem wyjścia do dalszych badań.

### Przykłady rozwiązań alternatywnych

W literaturze można znaleźć również zupełnie odmienne podejścia oparte na zastosowaniu klasycznych technik filtracji do danych reprezentujących orientacje w przestrzeni. Przykładem takich realizacji są prace [62], [72], w których użyty został rozszerzony filtr Kalmana do orientacji opisanych kwaternionami. Estymacja położenia obiektów na podstawie obrazów dwuwymiarowych jest realizowana przy użyciu podobnej klasy filtrów Kalmana dla kwaternionów dualnych w pracy [31]. Filtry Kalmana są w tym obszarze często stosowane w praktycznych zastosowaniach do systemów śledzących obiekty na obrazie. W podobny sposób zastosowanie filtrów realizujących układ dynamiczny zostało przedstawione w pracy [110] do generacji ruchów wykonywanych przy zmiennych ograniczeniach.

## 5.2 Wykazanie związku filtrów przetwarzających orientacje układów z rozwiązaniem liniowego równania procesu dyfuzji

W kolejnych podrozdziałach zostanie zaprezentowane uogólnienie wybranych schematów filtracji przedstawionych w poprzednim podrozdziale. Proponowana metoda filtracji oparta jest na numerycznym wyznaczeniu rozwiązania równania różniczkowego dla warunku początkowego opisanego trajektorią oryginalną. Prezentowane uogólnienie jest realizowane w sposób analogiczny do metod filtracji obrazów przedstawionych w rozdziale 2.4, jednak przy uwzględnieniu specyfiki trajektorii orientacji układów zdefiniowanej w przestrzeni kwaternionów. Podejście oparte na procesie dyfuzji jest następnie podstawą do konstrukcji bardziej zaawansowanych filtrów w przestrzeni  $S^3$  uwzględniających krzywiznę przetwarzanej trajektorii. Wstępne wyniki proponowanych rozwiązań zostały zaprezentowane przez autora w pracy [47].

### 5.2.1 Sformułowanie procesu dyfuzji

Dany jest proces dyfuzji dla przypadku jednowymiarowego opisany zagadnieniem początkowym:

$$\begin{cases} \frac{\partial u(x, t)}{\partial t} = c \frac{\partial^2 u(x, t)}{dx^2}, \\ u(x, t) = u^{(0)}(x) \end{cases}, \quad (5 - 43)$$

gdzie  $c$  jest stałym współczynnikiem dyfuzji. Niech krzywa w przestrzeni  $S^3$  będzie reprezentowana przez funkcję  $q^{(0)}(\tau)$ , gdzie  $\tau \in R$  jest zmienną parametryzującą krzywą. Ponieważ kwaterniony parametryzujące rotacje nie muszą należeć do hipersfery jednostkowej, bez utraty ogólności można założyć, że krzywa  $q^{(0)}(\tau) \in H$ . W kolejnych podpunktach zostanie pokazane, w jaki sposób można zrealizować opisywany proces, aby warunek zachowania krzywej na hipersferze jednostkowej był zachowany. Otrzymuje się następujące sformułowanie procesu:

$$\begin{cases} \frac{\partial q(\tau, t)}{\partial t} = c \frac{\partial^2 q(\tau, t)}{d\tau^2}, \\ q(\tau, 0) = q^{(0)}(\tau) \end{cases}, \quad (5 - 44)$$

gdzie  $t, \tau \in R$ ,  $q(\tau, t) \in H$ . Dziedziną zmiennej czasowej jest  $t \in [0, \infty)$ , natomiast zmienna parametryzująca krzywą jest ograniczona do  $\tau \in [0, T]$ .

W dalszej części pracy stosowane będą następujące oznaczenia:  $q(\tau, t)$  - krzywa kwaternionowa parametryzowana w przestrzeni  $H$  zmienną  $\tau$  oraz w przestrzeni skal (w wyniku operacji filtracji) zmienną  $t$ ;  $q^{(t)}(\tau)$  - realizacja krzywej kwaternionowej dla filtracji przeprowadzanej po czasie  $t$ ;  $q_i^{(t)}$  -  $i$ -ty element dyskretyzacji krzywej kwaternionów dla dyskretnego momentu filtracji (numeru iteracji)  $t$ .

Pojęcie pochodnej dla przestrzeni  $H$  może być rozpatrywane analogicznie, jak dla przestrzeni rzeczywistej [7] (zakładając, że nie ograniczamy się do podprzestrzeni kwaternionów jednostkowych). Dodatkowo w przedstawionym sformułowaniu procesu dyfuzji nie występuje mnożenie kwaternionowe, którego wynik zależałby od kolejności argumentów. Z formalnego punktu widzenia równanie (5-44) jest określone prawidłowo. Jednakże analityczne rozwiązanie równania różniczkowego określonego w przestrzeni kwaternionów może sprawiać znaczne trudności obliczeniowe. Dlatego dalsza konstrukcja algorytmów filtracji będzie oparta na uogólnieniu do przestrzeni kwaternionów dyskretnej postaci równania różniczkowego dyfuzji oryginalnie określonego w przestrzeni rzeczywistej.

Ponieważ rozpatrywana krzywa ma ograniczoną dziedzinę, należy określić warunki brzegowe dla zagadnienia. Można również w tym przypadku posłużyć się analogią do zagadnienia dwuwymiarowego przedstawionego w rozdziale 2. Uwzględniając specyfikę wyrażenia należy rozpatrzeć następujące przypadki.

#### Stała wartość funkcji na granicy obszaru (warunek A)

$$\begin{cases} q(0, t) = q^{(0)}(0) \\ q(T, t) = q^{(0)}(T) \end{cases} \quad (5 - 45)$$

Przypadek ten jest odpowiednikiem warunku brzegowego Dirchleta oznaczającym utrzymywanie stałej temperatury na brzegu dwuwymiarowego obszaru. W tym przypadku utrzymywana jest stała wartość funkcji w punktach granicznych trajektorii. Warunek ten również może być przydatny w przypadku filtracji, co zostanie pokazane w kolejnych punktach.

#### Układ odizolowany (warunek B)

Przypadek ten jest odpowiednikiem szczególnego przypadku warunku brzegowego Neumanna oznaczającym brak wymiany ciepła z otoczeniem na granicach obszaru. W praktyce przy dyskretyzacji układu oznacza to, że dyfuzja dla elementów na granicy obszarów nie będzie występować.

#### Przyjęcie założenia o periodyczności sygnału (warunek C)

$$q(\tau, t) = q(\tau + kT, t) \text{ dla } k \in Z \quad (5 - 46)$$

W takim przypadku funkcja została rozszerzona na dziedzinę nieskończoną. Dzięki temu nie ma konieczności definiowania dodatkowego warunku brzegowego. Sytuacja taka jest często wykorzystywana w praktyce, ponieważ dużo trajektorii reprezentujących ruch wykonywany w przestrzeni rzeczywistej jest okresowych lub wykazuje własności okresowości.

### Rozszerzenie funkcji na dziedzinę nieskończoną przez odbicie (warunek D)

$$q(\tau, t) = \begin{cases} q(-\tau, t) & \text{dla } \tau < 0 \\ q(\tau, t) & \text{dla } 0 \leq \tau \leq T \\ q(2T - \tau, t) & \text{dla } \tau > T \end{cases} \quad (5 - 47)$$

Podejście takie pozwala przyjąć założenie o nieskończonej dziedzinie trajektorii bez nakładania ograniczenia na jego periodyczność [66]. W tym przypadku również nie ma konieczności definiowania warunku brzegowego. W praktyce rozszerzenie nie musi dotyczyć całej dziedziny – w większości schematów dyskretyzacji jedynie kilka początkowych punktów trajektorii jest brane pod uwagę przy numerycznym rozwiązywaniu zagadnienia.

Z przedstawionych warunków brzegowych dwa pierwsze mają bezpośredni odpowiednik w przestrzeni rzeczywistej w postaci interpretacji matematycznej. Warunku brzegowe C i D zostały zdefiniowane ze względu na specyfikę ich zastosowania w algorytmach przetwarzania trajektorii przestrzennych. Dzięki temu można sformalizować podejście często stosowane w przypadkach praktycznych. Warunek D jest także używany w przypadku algorytmów filtracji obrazów, jak to zostało pokazane w rozdziale 2.

## 5.2.2 Dyskretyzacja modelu liniowego

Konstrukcja algorytmu filtracji dla kwaternionów oparta będzie na dyskretyzacji przeprowadzanej analogicznie, jak to jest realizowane przy równaniu różniczkowym cząstkowym w przestrzeni rzeczywistej. Dyskretyzacja zagadnienia początkowego (5 – 44) będzie zrealizowana przy użyciu jawnego schematu różnicowego, który jest opisany dla przypadku rzeczywistego w rozdziale 2.3. Dla przypadku ciągłego postać pochodnej funkcji w dziedzinie kwaternionów jest inaczej zdefiniowana, co zostało przedstawione w pracy [56]. Jednak w przypadku dyskretnym można przyjąć schemat zaproponowany w pracy [7] realizujący odpowiedniki równań dyskretyzacji (2 – 23), (2 – 26) dla kwaternionów:

$$\left. \frac{\partial q}{\partial \tau} \right|_{\tau=\tau_i} \approx \frac{q_{i+1}^{(t)} - q_i^{(t)}}{h} \quad (5 - 48)$$

$$\left. \frac{\partial^2 q}{\partial \tau^2} \right|_{\tau=\tau_i} \approx \frac{q_{i-1}^{(t)} - 2q_i^{(t)} + q_{i+1}^{(t)}}{h^2} \quad (5 - 49)$$

$$\left. \frac{\partial q}{\partial t} \right|_{t=t_i} \approx \frac{q_i^{(t+1)} - q_i^{(t)}}{h} \quad (5 - 50)$$

W praktyce zakładane będzie, że dyskretyzacja przestrzeni parametrów krzywej  $\tau$  (częstotliwość próbkowania sygnału) oraz dyskretyzacja przestrzeni czasu  $t$  jest jednostkowa  $h = 1$ .

Jawny schemat różnicowy dla zagadnienia (5 – 44) jest opisany jako:

$$q_i^{(t+1)} - q_i^{(t)} = c(q_{i-1}^{(t)} - 2q_i^{(t)} + q_{i+1}^{(t)}) \quad (5 - 51)$$

Dla zwiększenia czytelności przyjmuje się następujące oznaczenia  $\tilde{q}_i = q_i^{(t+1)}$ ,  $q_i = q_i^{(t)}$ . Po przekształceniu równania otrzymuje się:

$$\tilde{q}_i = q_i + c(q_{i-1} - 2q_i + q_{i+1}) \quad (5 - 52)$$

$$\tilde{q}_i = (1 - 2c)q_i + c(q_{i-1} + q_{i+1}) = (1 - 2c)q_i + 2c \frac{q_{i-1} + q_{i+1}}{2} = (1 - \hat{c})q_i + \hat{c} \frac{q_{i-1} + q_{i+1}}{2}, \quad (5 - 53)$$

gdzie  $\hat{c} = 2c$  jest nowym współczynnikiem dyfuzji. Dzięki postaci otrzymanej w równaniu (5 – 53) możliwe jest wyznaczenie wartości  $\tilde{q}_i$  przy użyciu złożenia dwóch interpolacji, które dla przypadku liniowego reprezentowane są przez wyrażenie (4 – 82):

$$q_p = \text{lerp}(q_1, q_2, p) = (1 - p)q_1 + pq_2 \quad (5 - 54)$$

Przepisując równanie (5 – 53) przy użyciu tych oznaczeń otrzymuje się:

$$\tilde{q}_i = \text{lerp}(q_i, \text{lerp}(q_{i-1}, q_{i+1}, 2), \hat{c}) \quad (5 - 55)$$

Na początku definiowania zagadnienia dyfuzji dla trajektorii kwaternionów przyjęto uproszczone założenie, że elementy trajektorii nie muszą leżeć na powierzchni hipersfery jednostkowej  $S^3$ . W większości przypadków wynik operacji opisanej równaniem (5 – 55) nie będzie kwaternionem jednostkowym. Kolejnym krokiem procedury powinna być więc normalizacja wyników do przestrzeni  $S^3$ . Jednakże rozwiązanie takie nie zachowywałoby stałej prędkości kątowej przy interpolacji. Możliwe jest alternatywne podejście przez modyfikację otrzymanej zależności przy uwzględnieniu specyfiki przestrzeni kwaternionów jednostkowych. Bazując na metodach interpolacji przedstawionych w rozdziale 4.2.4, odpowiednikiem interpolacji liniowej jest interpolacja sferyczno-liniowa *slerp* (4 – 83) realizowana w przestrzeni stycznej:

$$q_p = \text{slerp}(q_1, q_2, p) = q_1 \exp(p \log(q_1^{-1} q_2)) = q_1 (q_1^{-1} q_2)^p \quad (5 - 56)$$

Podobne podejście oparte na zamianie liniowej interpolacji na sferyczno-liniową została użyte w algorytmach konstrukcji krzywych w przestrzeni kwaternionów w pracy [88]. Zastosowanie tego typu interpolacji do schematu (5 – 53) pozwala na uzyskanie wyniku, który nie narusza pierwotnego ograniczenia na jednostkowość kwaternionu:

$$\tilde{q}_i = \text{slerp}(q_i, \text{slerp}(q_{i-1}, q_{i+1}, 2), \hat{c}) = q_i \left( q_i^{-1} \left( q_{i-1} (q_{i-1}^{-1} q_{i+1}) \right)^{1/2} \right)^{\hat{c}} \quad (5 - 57)$$

Użycie sferyczno liniowej interpolacji generuje rozwiązanie, które należy do hipersfery jednostkowej. Również dzięki wprowadzonemu zabiegowi zachowana jest stała prędkość kątowa w przestrzeni rozwiązań generowanych przez zastosowany jawny schemat różnicowy.

Dla zwiększenie czytelności zapisu oraz zgodności oznaczeń z pozostałymi algorytmami, w dalszej części pracy współczynnik dyfuzji  $\hat{c}$  w równaniu (5 – 57) będzie oznaczany przez  $c$ .

### 5.2.3 Konstrukcja algorytmu filtracji i aspekty implementacyjne

W algorytmie filtracji opartej na rozwiązaniu równania różniczkowego cząstkowego wykorzystywana jest sferyczno-liniowa interpolacja kwaternionowa. Należy jednak pamiętać, że odwzorowanie kwaternionów na przestrzeń rotacji jest przekształceniem 2 na 1 – każda rotacja ma podwójną parametryzację w przestrzeni kwaternionów  $q$  oraz  $-q$ . Z punktu widzenia parametryzacji rotacji nie ma znaczenia, który kwaternion został użyty. Jednak w kontekście procesu interpolacji, ma znaczenie, jaka parametryzacja przestrzeni rotacji została użyta. Aby zapobiec nieprawidłowym wynikom interpolacji proponuje się przeprowadzić wstępne przygotowanie trajektorii kwaternionów w oparciu o propozycję zamieszczoną w pracy [66].

---

#### Algorytm 4: Przygotowanie trajektorii kwaternionów

Parametry:  $q = \{q_0, \dots, q_i, \dots, q_n\}$  – trajektoria kwaternionów o długości  $n$

**krok 1:** Dla elementu trajektorii o indeksie  $i \leq n$  wykonaj

**krok 2:** jeżeli  $2\|\log q_i^{-1}q_{i-1}\| > \frac{\pi}{2}$  podstaw  $q_i = -q_i$

**krok 3:** Podstaw  $i \leftarrow i + 1$

---

Zakłada się, że w trajektorii nie występują duże przemieszczenia. Jeżeli rozważana jest filtracja sygnałów, które odwzorowują ruch wykonywany w rzeczywistości, to w praktyce warunek ten jest zawsze spełniony. Algorytm 4 będzie wykonywany automatycznie przed rozpoczęciem każdego procesu filtracji. Dzięki temu rezultaty przeprowadzanej filtracji nie będą zależeć od podwójnej reprezentacji będącej własnością charakterystyczną przestrzeni kwaternionów.

Po przygotowaniu trajektorii kwaternionów można uruchomić właściwy algorytm filtracji. Filtracja *LinQuat* realizowana jest przy pomocy algorytmu 5.

---

#### Algorytm 5: Filtracja trajektorii kwaternionów typu *LinQuat*

Parametry:  $q = \{q_0, \dots, q_i, \dots, q_n\}$  – dyskretna trajektoria kwaternionów o długości  $n$

$t_{\max}$  - liczba iteracji algorytmu;  $c \in (0,1]$  - współczynnik dyfuzji

**krok 1:** Podstaw  $t = 0$

**krok 2:** Dla  $t < T_{\max}$  wykonuj

**krok 3:** Dla każdego elementu trajektorii o indeksie  $0 < i < n$  wykonaj

**krok 4:** Wyznacz  $\hat{q} = q_{i-1}(q_{i-1}^{-1}q_{i+1})^{1/2}$

**krok 5:** Podstaw  $q_i^{(t+1)} = q_i(q_i^{-1}\hat{q})^c$

**krok 6:** Podstaw  $t \leftarrow t + 1$

---

Podobnie, jak w przypadku algorytmów filtracji obrazów trudno jest określić kryterium stopu na podstawie przetwarzanych danych. W praktyce najlepszym rozwiązaniem jest określenie maksymalnej

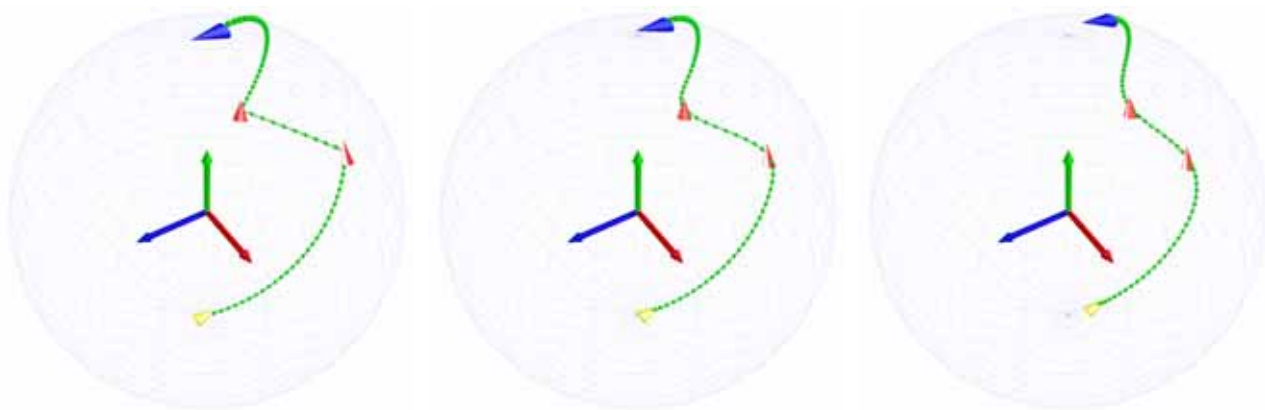


liczby wykonywanych iteracji. W kolejnych rozdziałach zostaną pokazane przykłady badań eksperymentalnych ilustrujące sposób działania algorytmu w zależności od liczby przeprowadzonych iteracji.

Współczynnik  $c$  decyduje o szybkości przeprowadzanego procesu. Jest to odpowiednik współczynnika dyfuzji w równaniu różniczkowym cząstkowym opisującym przewodzenie ciepła. Z własności interpolacji oraz charakteru dyskretyzacji wynika, że musi on być ograniczony do przedziału  $(0,1]$  w celu zachowania stabilnego schematu numerycznego. Przyjęcie wartości 0 byłoby równoznaczne z całkowitym zahamowaniem dyfuzji. Przyjęcie wartości większych od 1 jest prawidłowe z matematycznego punktu widzenia. Jednak w takim przypadku proces stanie się niestabilny numerycznie (patrz analiza numeryczna algorytmów – rozdział 6.2). Użyty jawny schemat różnicowy zakłada, że w kolejnych iteracjach realizowane będą małe kroki czasowe. Dlatego przyjęcie małych wartości współczynnika  $c$  spowoduje, że proces będzie bardziej zbliżony do odpowiednika ciągłego.

#### 5.2.4 Podstawowe własności procesu

Zaproponowany algorytm jest odpowiednikiem realizacji procesu dyfuzji przebiegającej dla krzywej należącej do hipersfery jednostkowej. Na rysunku 5-1 przedstawiono przykładowy proces filtracji przebiegający dla trajektorii testowej po 10 i 50 iteracjach. Widoczny jest charakterystyczne rozmywanie trajektorii, które w tym przypadku charakteryzuje się minimalizacją sumy odległości pomiędzy wszystkimi punktami. W rozdziale 6.2 zostaną szczegółowo przedstawione rezultaty filtracji z analizą numerycznych własności tego algorytmu.



**Rysunek 5-1: Przykładowy przebieg procesu filtracji *LinQuat* – kolejne fragmenty odpowiednio przedstawiają: trajektoria oryginalna, po 10, po 50 iteracjach ( $c=0,5$ ).**

Z punktu widzenia matematycznego otrzymana zależność nie jest dyskretyzacją rozwiązania równania różniczkowego w przestrzeni kwaternionów, ze względu na inne własności tej przestrzeni. Dyskretyzacja równania jest przeprowadzana dla równania w przestrzeni rzeczywistej, jakkolwiek kolejne kroki mają swoje odpowiedniki dla kwaternionów. Dalsza konstrukcja wynika z uogólnienia dyskretnej formy równania w celu uwzględnienia własności przestrzeni  $S^3$ .

Ze względu na przebieg filtracji, otrzymuje się algorytm posiadający podobne własności oraz pozwalający na osiągnięcie założonych celów analogicznych do realizacji odpowiadającego mu procesowi

dyfuzji w przestrzeni rzeczywistej. Analiza działania filtracji nie będzie więc oparta na matematycznych własnościach równań różniczkowych, lecz na efektach, które są otrzymywane w wyniku działania algorytmu. Podobne założenie dotyczyć również będzie pozostałych filtrów skonstruowanych w ramach prezentowanej pracy.

Zastosowanie interpolacji sferyczno-liniowej pozwoliło uwzględnić specyficzne własności przestrzeni  $S^3$  przy jednoczesnym zachowaniu dyfuzyjnego charakteru procesu. Należy zaznaczyć, że równanie różniczkowe (5 – 44) nie jest liniowe, jeżeli przyjmie się, że dziedzina funkcji ograniczona jest do kwaternionów jednostkowych. W prezentowanej pracy liniowość filtracji jest określana ze względu na charakter użytego współczynnika dyfuzji. Ze względu na stałość współczynnika dyfuzji (brak zależności od wartości funkcji) przedstawiony proces jest homogeniczny. Określenie prezentowanego algorytmu *LinQuat* odnosi się więc do liniowego (i stałego) współczynnika dyfuzji używanego w tym procesie.

Dla stałego współczynnika dyfuzji  $c$  rozwiązanie rozpatrywanego zagadnienia początkowego w dziedzinie rzeczywistej jest realizowane przez splot z jądrem Gaussa (rozdział 2.2). Rozwiązanie numeryczne jest więc iteracyjną wersją dyskretnego splotu otrzymywanego przy użyciu odpowiednio zdefiniowanej maski (rozdział 2.4). W przypadku proponowanej tego samego procesu algorytmem *LinQuat* otrzymuje się iteracyjną wersję będącą odpowiednikiem dyskretnego splotu trajektorii z maską reprezentującą jądro gaussowskie. Otrzymana procedura jest więc zbieżna z podejściem przedstawionym w rozdziale 5.1.1 realizującym liniowe przetwarzanie trajektorii kwaternionów przez zastosowanie filtru klasy LTI – równanie (5 – 13):

$$\hat{F}_B(q_i) = q_i \exp\left(\frac{1}{16}(-\omega_{i-2} - 5\omega_{i-1} + 5\omega_i + \omega_{i+1})\right) \quad (5 - 58)$$

Podejście zaproponowane w pracy [66] jest więc szczególnym przypadkiem realizacji splotu w dziedzinie kwaternionów (a ściślej w przestrzeni stycznej do hipersfery  $S^3$ ) wynikającym z analitycznego rozwiązania liniowego równania różniczkowego cząstkowego dyfuzji ciepła. Podobnie metody oparte na dyskretnym splocie obrazu z maską reprezentującą jądro Gaussa wynikają bezpośrednio z rozwiązania równania dyfuzji dla przypadku dwuwymiarowego. Stąd wynika asymptotyczna zbieżność filtracji typu *LinQuat* z filtracją przy użyciu liniowych filtrów LTI dla kwaternionów.

Konstrukcja proponowanej metody jest również podobna do schematu przedstawionego w [63] opisanego równaniem (5 – 25) – rozdział 5.1.2:

$$\tilde{q}_i = q_{i+1}q_i^{-1}q_{i-1} \quad (5 - 59)$$

Cytowane podejście jest złożeniem dwóch rotacji wykonywanych w przestrzeni kwaternionów (patrz rozdział 4.2.2) – pierwsza jest wykonywana od kwaternionu  $q_i$  do  $q_{i-1}$ , w drugim kroku wykonywane jest złożenie rotacji reprezentowanej przez kwaternion  $q_{i+1}$  oraz wyniku rotacji wykonanej w kroku pierwszym. Działanie algorytmu *LinQuat* opiera się na podobnej zasadzie. Zamiast jednak składać rotacje, wykonywana jest interpolacja kwaternionowa o określonym współczynniku interpolacji. Krok 5 proponowanego algorytmu jest tożsamy z równaniem (5 – 27) w procesie filtracji przy użyciu operatorów wygładzających.

Cytowany algorytm został skonstruowany na podstawie kryterium energetycznego, natomiast proponowane rozwiązanie wynika z rozwiązania równania różniczkowego cząstkowego.

Krótką analizą zaproponowanej metody filtracji wskazuje na podobieństwa z dwoma niezależnie wyprowadzonymi procesami:

- ze względu na jawną postać rozwiązania równania różniczkowego dyfuzji, otrzymuje się proces zbieżny do filtracji typu LTI opartej na splocie z dyskretną maską (rozdział 5.1.1),
- ze względu na kryterium energetyczne otrzymuje się podobnie działający operator wygładzający (rozdział 5.1.2).

Jakkolwiek efekty działania algorytmu *LinQuat* oraz algorytmów wcześniejszych są podobne, istnieje podstawowa różnica w sposobie konstrukcji tak zaproponowanej filtracji. Konstrukcja wynikająca z bezpośredniego rozwiązywania równania dyfuzji pozwala na uogólnienie procesu na nieliniowe, niehomogeniczne równania różniczkowe opisane w rozdziale 2.5. Propozycje uogólnienia oraz uwzględnienie dodatkowego czynnika wynikającego z własności kwaternionów zostaną przedstawione w kolejnych rozdziałach pracy.

### 5.3 Uogólnienie nieliniowego procesu dyfuzji dla przetwarzania orientacji układu w przestrzeni $S^3$

Opracowanie metody filtracji opartej na dyskretniej postaci procesu dyfuzji pozwala na rozszerzenie podejścia na procesy oparte na równaniu różniczkowym o nieliniowym współczynniku zależnym od wartości funkcji. Analogicznie, jak to jest realizowane w nieliniowym podejściu dla filtracji obrazów, proponuje się rozpatrzenie następującego zagadnienia początkowego:

$$\begin{cases} \frac{\partial q}{\partial t} = \text{div}(c(\tau, q, t)\nabla q) & \text{w } (0, T) \times (0, n) \\ q(\tau, 0) = q^{(0)}(\tau) & \text{na } (0, n), \text{ dla } \tau \in R \end{cases} \quad (5 - 60)$$

Podobnie, jak w przypadku wersji *LinQuat* celem jest, aby proces był realizowany dla  $q \in S^3$ . Jednak w ogólności można przyjąć, że proces jest poprawnie określony dla każdego kwaternionu  $q \in H$  (niekoniecznie jednostkowego). Zastosowanie odpowiedniej dyskretyzacji zagadnienia (opartej na dyskretyzacji zagadnienia rzeczywistego) pozwoli znaleźć rozwiązanie dla wybranej chwili czasowej przy zachowaniu warunku, aby wynik należał do hipersfery jednostkowej.

Wprowadzenie modelu nieliniowego ma na celu modyfikację własności procesu filtracji tak, aby wraz z usuwaniem zakłóceń, nie był zniekształczany przebieg trajektorii. Analogicznie w przypadku filtracji obrazów celem było usunięcie zakłóceń, przy jednoczesnym zachowaniu krawędzi reprezentujących właściwą informację na obrazie.

#### Odpowiednik krawędzi dla trajektorii orientacji

Aby zachować przebieg trajektorii zawartej w  $S^3$  proponuje się uogólnienie podejścia stosowanego do filtracji obrazów. W tym celu należy zdefiniować odpowiednik estymatora krawędzi w przypadku

trajektorii. Odwołując się do modelu danych opisujących orientacje układów (rozdział 4.4) należy zaznaczyć, że odpowiednik krawędzi będzie opierał się na pochodnej trajektorii rozumianej, jako wektor prędkości kątowej (w odróżnieniu do obrazów, dla których estymator krawędzi definiowany jest przy użyciu gradientu). Estymator krawędzi dla trajektorii powinien więc spełniać następującą zależność:

- $E(\tau, t) = 0$  - dla fragmentów trajektorii jednolitych
- $E(\tau, t) = \|K\omega(\tau, t)\|$  - w punkcie należącym do krawędzi, gdzie  $\omega$  jest wektorem prędkości kątowej w chwili  $t$ , a  $K$  jest zmienną reprezentującą kontrast w punkcie rozumiany jako różnica prędkości pomiędzy krawędzią, a fragmentem otaczającym

Również w tym przypadku definicja odpowiednika krawędzi jest realizowana przez określenie estymatora. Możliwe są alternatywne definicje, jednak ta jest wystarczająca z punktu widzenia definicji nieliniowego procesu dyfuzji.

### Definicja

Współczynnik dyfuzji zależy w proponowanym modelu od estymatora krawędzi w taki sposób, aby we fragmentach jednolitych trajektorii przyjmował wartość 1, a w obszarach zbliżonych do krawędzi (znaczej skalarnej zmianie prędkości kątowej) malał monotonicznie do 0:

$$c(\tau, q, t) = g(\|E\|) = g\left(\left\|\frac{\partial q(\tau, t)}{\partial \tau}\right\|\right) = g(\|\omega(\tau, t)\|) \quad (5 - 61)$$

Prędkość kątowa w równaniu (5 - 61) musi być wyznaczona przy zachowaniu własności przestrzeni kwaternionów. Uwzględnianie modułu wektora prędkości pozwala na prostą interpretację względem funkcji stopującej  $g$  oraz użycie takiej parametryzacji w definicji współczynnika  $c$ . Numeryczny sposób obliczania współczynnika dyfuzji zostanie przedstawiony wraz z dyskretyzacją zagadnienia początkowego. W dalszej części pracy zostanie pokazane, w jaki sposób należy skonstruować proces filtracji, aby brana była pod uwagę również informacja o kierunku wyznaczanym przez wektor prędkości (zmianie krzywizny).

W prezentowanej pracy będą używane dwie postacie funkcji stopujących – te same, które zostały użyte w metodach filtracji obrazów w równaniach (2 - 51), (2 - 52):

$$g_1(s) = \exp\left(-\left(s/K\right)^2\right) \quad (5 - 62)$$

$$g_2(s) = \left(1 + \left(\frac{s}{K}\right)^2\right)^{-1} \quad (5 - 63)$$

Współczynnik  $K$  jest drugim parametrem procesu filtracji, który będzie decydował o jego właściwościach. Dla dużych wartości współczynnika dyfuzja będzie szybciej przeprowadzana, ponieważ dopiero większe przemieszczenia (większa prędkość kątowa) będą uznawane przez funkcję, jako krawędź. Dla małych wartości współczynnika dyfuzja jest wolniej przeprowadzana – niewielkie przemieszczenia będą rozpatrywane jako krawędź, więc dyfuzja jest hamowana. Podobnie, jak w przypadku algorytmów filtracji obrazów, również w tym przypadku współczynnik  $K$  może być wyznaczany adaptacyjnie na podstawie

danych występujących w trajektorii. W tym celu używana będzie zależność (2 – 57) zaadaptowana do przestrzeni kwaternionowej.

Otrzymany model ma następujące podstawowe własności:

- jest nieliniowy ze względu dziedzinę, w jakiej jest opisana trajektoria (kwaterniony jednostkowe),
- jest nieliniowy oraz niehomogeniczny (zależny od wartości funkcji) ze względu na współczynnik dyfuzji  $c$ .

### 5.3.1 Dyskretyzacja modelu nieliniowego

Realizacja dyskretyzacji zagadnienia początkowego (5 – 60) oparta będzie na schemacie różnicowym (2 – 30) używanym w przypadku obrazów w równaniu (2 – 61). Dla trajektorii kwaternionów równanie dyskretyzacji przyjmuje postać:

$$q_i^{(t+1)} - q_i^{(t)} = \tilde{\lambda} [c_P \cdot \nabla_P q + c_N \cdot \nabla_N q]_i^{(t)}, \quad (5 - 64)$$

gdzie indeksy  $P$ ,  $N$  oznaczają poprzedniego oraz następnego sąsiada elementu  $i$  według relacji sąsiedztwa zdefiniowanej w modelu danych trajektorii w rozdziale 4.4. Dla stabilności procesu musi być zachowane  $\tilde{\lambda} \leq 0.5$ . Lokalne dyskretyzacje pochodnych są realizowane przez różnice skończone w zmodyfikowanej postaci równania (5 – 48):

$$\nabla_P q_i = q_{i-1} - q_i \quad (5 - 65)$$

$$\nabla_N q_i = q_{i+1} - q_i \quad (5 - 66)$$

Podobnie, jak w przypadku dyskretyzacji modelu liniowego początkowo przyjmuje się, że dziedziną funkcji jest cała przestrzeń kwaternionów. Warunku na jednostkowość elementów nie jest na tym etapie rozpatrywany. Przekształcając równanie (5 – 64) kolejno otrzymuje się:

$$\begin{aligned} \tilde{q}_i &= q_i + \tilde{\lambda} (c_P (q_{i-1} - q_i) + c_N (q_{i+1} - q_i)) = \frac{q_i}{2} + \tilde{\lambda} c_P (q_{i-1} - q_i) + \frac{q_i}{2} + \tilde{\lambda} c_N (q_{i+1} - q_i) \\ &= \frac{q_i (1 - 2\tilde{\lambda} c_P) + 2\tilde{\lambda} c_P q_{i-1}}{2} + \frac{q_i (1 - 2\tilde{\lambda} c_N) + 2\tilde{\lambda} c_N q_{i+1}}{2} \\ &= \frac{(q_i (1 - \tilde{c}_P) + q_{i-1} \tilde{c}_P) + (q_i (1 - \tilde{c}_N) + q_{i+1} \tilde{c}_N)}{2} \end{aligned} \quad (5 - 67)$$

Ponieważ współczynnik  $\tilde{\lambda}$  jest stały, więc iloczyny  $2\tilde{\lambda} c_P$  oraz  $2\tilde{\lambda} c_N$  mogą być zastąpione nowymi liniowo przeskalowanymi zmiennymi oznaczonymi odpowiednio  $\tilde{c}_P$  oraz  $\tilde{c}_N$ . Również w tym przypadku rozwiązanie w postaci (5 – 67) nie jest realizowane w przestrzeni kwaternionów jednostkowych. Aby uniknąć konieczności normalizacji i zapewnić, że wynik nadal będzie należał do sfery jednostkowej proponuje się zmianę interpolacji liniową na sferyczno-liniową:

$$\tilde{q}_i = q_i (q_i^{-1} q_{i-1})^{\tilde{c}_P} \left[ \left( q_i (q_i^{-1} q_{i-1})^{\tilde{c}_P} \right)^{-1} \left( q_i (q_i^{-1} q_{i+1})^{\tilde{c}_N} \right) \right]^{1/2} \quad (5 - 68)$$

Dla proponowanego algorytmu nieliniowego współczynniki dyfuzji są nieliniowe i zależne od odpowiednika estymatora krawędzi dla trajektorii. Proponuje się, aby współczynniki te były wyznaczane na podstawie miary odległości zdefiniowanej w  $S^3$  dla sąsiadujących elementów trajektorii. Uwzględniając wyrażenie (4 – 94) otrzymuje się odpowiednio:

$$\tilde{c}_P = g(\|E_P\|) = \lambda g\left(\|2\log(q_i^{-1}q_{i-1})\|\right) \quad (5 - 69)$$

$$\tilde{c}_N = g(\|E_N\|) = \lambda g\left(\|2\log(q_i^{-1}q_{i+1})\|\right) \quad (5 - 70)$$

W równaniach (5 – 62), (5 – 63) został uwzględniony w przeskalowanej postaci parametr  $\lambda = 2\tilde{\lambda}$ . Ze względu na ograniczenie na współczynnik  $\tilde{\lambda}$ , występuje analogiczne ograniczenie na parametr określający szybkość lokalnej dyfuzji  $\lambda \in (0,1]$ . Jeżeli w równaniach zastosowana zostanie odpowiednia funkcja stopująca, to wartość współczynników dyfuzji  $c$  w przedstawionych równaniach zawsze ograniczona jest do przedziału  $[0,1]$ . Nie są więc naruszone warunki stabilności procesu.

Przy dyskretyzacji zagadnienia początkowego z postaci (5 – 64) do postaci (5 – 67) współczynnik  $\tilde{\lambda}$  został włączony do schematu interpolacji, dzięki czemu możliwe było przedstawienie całego równania w postaci złożenia sferyczno-liniowej interpolacji w przestrzeni kwaternionów. Współczynnik ten odgrywa rolę tłumiącą oraz regulującą w jawnym schemacie różnicowym. Zmniejszenie kroku dyskretyzacji pozwala na zwiększenie stabilności schematu. Można również zaproponować dodatkowy współczynnik tłumiący  $c$ , który będzie używany w ostatnim kroku algorytmu filtracji. Współczynnik ten będzie realizował rolę analogiczną, jaką pełni czynnik tłumiący w schematach (5 – 27), (5 – 57). Aby zachować jednorodne oznaczenia parametrów, współczynnik ten będzie nazywany  $c$ :

$$q_i^{(t+1)} = q_i(q_i^{-1}\tilde{q})^c \quad (5 - 71)$$

Aby parametr  $c$  pełnił rolę tłumiącą, jego wartość musi być zawarta w przedziale  $(0,1]$ , co wynika z ograniczeń narzuconych także na parametr  $\lambda$ . W praktyce oba współczynniki pełnią podobną rolę tłumiąco-regularyzacyjną procesu. Gdyby przedstawiony schemat był rozpatrywany w przestrzeni liniowej, to rola obu wielkości całkowicie by się pokrywała. Analizując wyniki obliczeń numerycznych, które zostaną zaprezentowane w rozdziale 7, okazało się, że w praktyce występowanie jednego współczynnika tłumiącego w procesie jest wystarczające do zachowania zarówno stabilności procesu, jak i kontrolowania charakteru przebiegu filtracji. Stąd w wielu testach przyjęto, że domyślną wartość współczynnika  $c = 1$ .

### 5.3.2 Konstrukcja algorytmu filtracji nieliniowej

Proponowana metoda filtracji jest uogólnieniem na trajektorię w przestrzeni kwaternionów filtracji nieliniowej zaprezentowanej przez Perona, Malik w [78], [79]. Stąd algorytm realizujący proponowaną metodę nazwany został *NonLinQuat*. Parametry filtracji nieliniowej są analogiczne do parametrów stosowanych w algorytmie przetwarzania obrazów. Istotnym czynnikiem jest wybór rodzaju funkcji stopującej  $g$ , która może być dowolną, monotonicznie malejącą funkcją spełniającą warunek  $g(0) = 1$  oraz

$\lim_{s \rightarrow \infty} g(s) = 0$ . Algorytm realizuje dyskretyzację przedstawioną w podrozdziale 5.3.1 przy uwzględnieniu warunków brzegowych opisanych szczegółowo w podrozdziale 5.2.1. Warunki brzegowe są nazwane symbolicznie  $A, B, C, D$ .

---

**Algorytm 6: Nieliniowa filtracja trajektorii kwaternionów *NonLinQuat***

Parametry:  $q = \{q_0, \dots, q_i, \dots, q_n\}$  – trajektoria kwaternionów o długości  $n$

$t_{\max}$  - liczba iteracji algorytmu

$c \in (0,1]$  (domyślnie  $c=1$ )

$\lambda \in (0,1]$  - współczynniki szybkości filtracji (współczynniki tłumiące)

$K$  – współczynnik wrażliwości dla funkcji stopującej  $g$

$A, B, C, D$  – rodzaj warunku brzegowego (rozdział 5.2.1)

**krok 1:** Podstaw  $t = 0$

**krok 2:** Dla  $t < t_{\max}$  wykonuj

**krok 3:** Dla każdego elementu trajektorii o indeksie  $0 \leq i \leq n$  wykonaj

**krok 4:** Jeżeli warunek **A** oraz  $i = 0 \vee i = n$  wróć do kroku 3

**krok 5:** Jeżeli warunek **B** oraz  $i = 0$

**krok 6:** Wyznacz  $\tilde{c}_P = 0$ ,  $\tilde{c}_N = \lambda g(\|2 \log(q_0^{-1} q_1)\|)$

**krok 7:** Przypisz  $\tilde{q}_P = q_0$ ,  $\tilde{q}_N = q_0(q_0^{-1} q_1)^{\tilde{c}_N}$

**krok 8:** Jeżeli warunek **B** oraz  $i = n$

**krok 9:** Wyznacz  $\tilde{c}_P = \lambda g(\|2 \log(q_n^{-1} q_{n-1})\|)$ ,  $\tilde{c}_N = 0$

**krok 10:** Przypisz  $\tilde{q}_P = q_n(q_n^{-1} q_{n-1})^{\tilde{c}_P}$ ,  $\tilde{q}_N = q_n$

**krok 11:** Jeżeli warunek **C** oraz  $i = 0$  przypisz  $q_P = q_n$ ,  $q_N = q_1$

**krok 12:** Jeżeli warunek **C** oraz  $i = n$  przypisz  $q_P = q_{n-1}$ ,  $q_N = q_0$

**krok 13:** Jeżeli warunek **D** oraz  $i = 0$  przypisz  $q_P = q_1$ ,  $q_N = q_1$

**krok 14:** Jeżeli warunek **D** oraz  $i = n$  przypisz  $q_P = q_{n-1}$ ,  $q_N = q_{n-1}$

**krok 15:** Jeżeli  $i \in [1, n-1]$  przypisz  $q_P = q_{i-1}$ ,  $q_N = q_{i+1}$

**krok 16:** Jeżeli  $i \in [1, n-1]$  lub warunek **C** lub warunek **D**

**krok 17:** Wyznacz  $\tilde{c}_P = \lambda g(\|2 \log(q_i^{-1} q_P)\|)$

**krok 18:** Wyznacz  $\tilde{c}_N = \lambda g(\|2 \log(q_i^{-1} q_N)\|)$

**krok 19:** Wyznacz  $\tilde{q}_P = q_i(q_i^{-1} q_P)^{\tilde{c}_P}$

**krok 20:** Wyznacz  $\tilde{q}_N = q_i(q_i^{-1} q_N)^{\tilde{c}_N}$

**krok 21:** Wyznacz  $\tilde{q}_i = \tilde{q}_P(\tilde{q}_P^{-1} \tilde{q}_N)^{1/2}$

**krok 22:** Podstaw  $q_i^{(t+1)} = q_i(q_i^{-1}\tilde{q}_i)^c$

**krok 23:** Podstaw  $t = t \leftarrow 1$

Wartość współczynnika  $K$  może być przyjęta, jako stała lub też może być wyznaczana adaptacyjnie na podstawie analizy zmienności sygnału w danej iteracji. Dla odpowiednika równania (2 – 57) używanego do adaptacyjnego określenia współczynnika  $K$  w filtracji obrazów, proponuje się następujące uogólnienie w przypadku trajektorii w  $S^3$ :

$$K(t) = K_{\max} \frac{1}{n} \sum_{j=1}^{n-1} |\nabla q_i^{(t)}| = K_{\max} \frac{1}{n} \sum_{j=1}^{n-1} (\|2 \log(q_i^{-1} q_{i-1})\|) \quad (5 - 72)$$

Otrzymane wyrażenie na współczynnik  $K$  jest estymatorem sygnału zakłóceń w przestrzeni kwaternionów przy użyciu skalarnej miary odległości. Parametr  $K_{\max}$  pełni rolę skalującą i ograniczającą wyznaczoną wartość całkowitej zmienności sygnału.

Zaprezentowany algorytm jest uogólnieniem nieliniowej filtracji dla trajektorii opisującej orientację obiektu przy użyciu parametryzacji kwaternionowej. Algorytm powoduje, że trajektoria nie zostanie wygładzona w sposób liniowy, lecz w zależności od położenia opisanego przez elementy trajektorii oraz ich najbliższego otoczenia. Dla danych, które reprezentują dużą zmianę położenia w przestrzeni kwaternionów (określoną przy użyciu kwaternionowej miary odległości), dyfuzja zostanie zahamowana. W ten sposób możliwe będzie zachowanie częściowego kształtu trajektorii przy jednoczesnym usunięciu zakłóceń.

Należy zwrócić uwagę, że definicja miary odległości podana równaniem (4 – 94) bezpośrednio wpływa na własności przedstawionego procesu filtracji. W dalszej części pracy zostanie zaprezentowany model uwzględniający informację o gładkości trajektorii także z punktu widzenia zmiany kierunku orientacji, a nie tylko części skalarnej kwaternionu.

## 5.4 Propozycja anizotropowego procesu dyfuzji dla trajektorii w przestrzeni orientacji

W kolejnych podrozdziałach zostanie przedstawione dalsze rozwinięcie zaproponowanej metody uwzględniające zmienność kierunku w przebiegu trajektorii należącej do hipersfery jednostkowej. Algorytm realizujący dyskretyzację procesu dyfuzji będzie uwzględniał nie tylko informacje o położeniu elementu trajektorii (algorytm *LinQuat*) oraz informacji o chwilowej prędkości w punkcie (algorytm *NonLinQuat*), ale także informację o zmianie składowej stycznej prędkości chwilowej (przyspieszenia stycznego), która decyduje o kierunku ruchu oraz kształcie trajektorii. Sformułowanie algorytmu bazuje na własnościach wyprowadzonych w poprzednich punktach, lecz dodatkowo wprowadza uwzględnienie dodatkowych własności trajektorii związanych z kierunkiem ruchu, przez co nazywany będzie anizotropowym.



### 5.4.1 Cel wprowadzenia współczynnika uwzględniającego kierunek

W rozdziale 5.3 przedstawiona została metoda realizująca nieliniową filtrację trajektorii orientacji zapisanych w przestrzeni kwaternionów. Nieliniowość pozwoliła uwzględnić zmienną szybkość ruchu wykonywanego wzdłuż trajektorii. Zakłada się, że możliwe jest odróżnienie drobnych zakłóceń wynikających z szumów pomiarowych od ogólnej charakterystyki trajektorii. Znaczna zmiana prędkości nie powinna być więc wygładzana, aby trajektoria zachowała swój pierwotny charakter.

Właściwe dla przestrzeni kwaternionów przekształcenie dyskretyzacji równania różniczkowego pozwoliło otrzymać postać rozwiązania będącą odpowiednikiem dyskretnego procesu dyfuzji realizowanego w przestrzeni rzeczywistej. W ten sposób możliwe było wprowadzenie nieliniowego współczynnika analogicznie, jak to jest realizowane w modelu *Perona-Malik*. Jednakże nieliniowy współczynnik dyfuzji uwzględnia jedynie skalarną miarę odległości pomiędzy kolejnymi położeniami wzdłuż trajektorii. W dwuwymiarowej przestrzeni obrazów podejście takie dało dużą poprawę uzyskiwanych efektów filtracji i było naturalne z punktu widzenia dziedziny przetwarzanych danych. Dla trajektorii w przestrzeni kwaternionów podejście takie może być pomocne jedynie w wyjątkowych sytuacjach, gdy w ruchu występują nagłe zmiany prędkości. Jednakże możliwe jest dalsze rozwinięcie zaproponowanego poprzednio modelu nieliniowego.

Uogólnieniem podejścia nieliniowego jest propozycja, która zostanie zaprezentowana w niniejszym rozdziale. Proces filtracji zostanie uzależniony nie tylko od skalarnej zmiany położenia ciała, ale także od zmiany jego kierunku. Dlatego algorytm realizujący taki proces został nazwany anizotropowym. Anizotropia odnosi się więc do zmiennego współczynnika w zależności od zmiany kierunku trajektorii. Jego specyfika polega na uwzględnieniu zarówno skalarnej zmiany w trajektorii (odpowiadającej za wartość prędkości kątowej), jak i kierunku tej zmiany przez analizę odpowiednika stycznego przyspieszenia kątowego dla kwaternionów.

Należy podkreślić, że anizotropia jest w tym przypadku inaczej interpretowana niż analogiczny algorytm filtracji dla obrazów. W przypadku trajektorii parametryzacja dziedziny funkcji jest jednowymiarowa (chwile czasowe lub elementy łuku krzywej). Dla obrazów dziedzina funkcji jest dwuwymiarowa, co wprowadza pojęcie anizotropii rozumianej jako zmienność własności procesu filtracji w zależności od kierunku działania operatora (patrz rozdział 2.5.4). W przypadku trajektorii anizotropia dotyczy uwzględnienia zmiany zarówno składowej skalarnej, jak i wektorowej kwaternionu. Dzięki zmodyfikowanemu procesowi filtracji zostaje w znaczny sposób poprawiona zdolność do usuwania zakłóceń z trajektorii przy jednoczesnym zachowaniu kształtu krzywej w przestrzeni orientacji. Szczegółowe praktyczne różnice pomiędzy poszczególnymi algorytmami zostaną pokazane w rozdziale 6 przy eksperymentalnym badaniu ich właściwości.

### 5.4.2 Sformułowanie procesu filtracji

Wprowadzenie jawnego współczynnika tłumiącego  $c$  w równaniu (5 – 71) będzie pomocne w sformułowaniu procesu anizotropowego. Należy pamiętać, że współczynnik ten występuje jawnie w ciągłej

wersji zagadnienia początkowego opisanego równaniami różniczkowymi. Faza dyskretyzacji prowadząca do postaci interpolacyjnej równania powoduje, że współczynnik ten zostaje włączony do przeskalowanej wartości współczynnika dyfuzji. Aby jednak występował on w postaci jawnej, proponuje się przeprowadzenie dodatkowej interpolacji wynikającej z początkowej dyskretyzacji, co wpływa na poprawę stabilności procesu.

Współczynnik tłumiący może odgrywać również rolę modyfikacji własności procesu. Można przyjąć interpretację, że  $c$  wpływa na zmianę kształtu przetwarzanej trajektorii. Jeżeli  $c \rightarrow 0$ , to zmiana przebiegu trajektorii będzie minimalna. Dla  $c \rightarrow 1$ , zmiana kształtu jest najbardziej widoczna. Dla wartości  $c > 1$  proces przestaje być stabilny i zachowywać ważne własności modelu (szczegółowo opisane w rozdziale 5.6).

Celem opracowywanej filtracji jest usunięcie zakłóceń przy jak najmniejszym zniekształceniu danych zawartych w oryginalnym sygnale (co może być określone jako maksymalizacja stosunku sygnału do szumu). Jak to zostało pokazane, jedna z istotniejszych informacji o trajektorii jest zawarta w jej kształcie. Aby zmniejszyć zniekształcenie kształtu trajektorii proponuje się zastosować podejście analogiczne, jakie jest używane w nieliniowym, niehomogenicznym procesie wygładzania. Niech szybkość modyfikacji kształtu (kontrolowana przy użyciu współczynnika  $c$ ) będzie zależna nieliniowo od krzywizny trajektorii w punkcie (lokalnej miary zmiany kształtu). Przyjmując  $\kappa(t)$ , jako miarę krzywizny w punkcie  $t$  sformułowanie przyjmuje postać:

$$\tilde{c}_A(q, t) = \lambda_A g(\kappa(t)) \quad (5 - 73)$$

Funkcja użyta do tej parametryzacji powinna realizować zadanie funkcji stopującej użytej w poprzednich rozwiązaniach. Stąd zostało użyte oznaczenie  $g(\cdot)$ , które może reprezentować przykładowo funkcje określone wyrażeniami (5 - 62), (5 - 63). Współczynnik  $c$  jest uzależniony od czasu będącego parametrem trajektorii, a także od wartości funkcji  $q$  reprezentującej trajektorię, ponieważ jest ona niezbędna do wyznaczenia krzywizny trajektorii. Parametr  $\lambda_A$  jest opcjonalny dla zachowania jednolitości oznaczeń i konstrukcji algorytmu. W praktyce najczęściej może on być pominięty i przyjmuje stałą wartość  $\lambda_A = 1$ .

Jak to zostało wcześniej przedstawione (podrozdział 5.1.2) często ([42], [55], [63]) stosuje się następującą miarę określającą całkowitą krzywiznę trajektorii:

$$E(q) = \int_q \|\kappa(t)\|^2 dt = \int_q \left\| \frac{d\omega(t)}{dt} \right\|^2 dt, \quad (5 - 74)$$

gdzie prędkość kątowna reprezentowana jest jako

$$\omega_i = \frac{\log(q_i^{-1} q_{i+1})}{h}, \quad (5 - 75)$$

co prowadzi do wyrażenia na pochodną prędkości

$$\frac{d\omega_i}{dt} = \frac{\log(q_i^{-1} q_{i+1}) - \log(q_{i-1}^{-1} q_i)}{h^2}, \quad (5 - 76)$$

W ten sposób krzywizna w punkcie  $i$  trajektorii może być wyrażone przez pochodną prędkości kątownej reprezentującą przyspieszenie bez bezpośredniego wyznaczania drugiej pochodnej w punkcie dla przestrzeni kwaternionów (operacja logarytmu jest równoznaczna z wyznaczeniem wektora w przestrzeni

stycznej). Jest to dyskretna postać przyspieszenia interpretowana, jako zmiana prędkości kątowych. Jednakże jedynie składowa styczna prędkości ma wpływ na zmianę kierunku, która będzie rozpatrywana w prezentowanej metodzie. W związku z tym interesującą wielkością przy wyznaczaniu przyspieszenia kątowego jest znormalizowany jednostkowy wektor prędkości kątowej określający oś obrotu. Uwzględniając w wyrażeniu na krzywiznę tak wyznaczoną składową prędkości oraz przyjmując za krok dyskretyzacji  $h=1$ , ostatecznie otrzymuje się lokalną wartość krzywizny dla  $i$ -tego elementu:

$$\kappa_i = \left\| \frac{\omega_i}{\|\omega_i\|} - \frac{\omega_{i-1}}{\|\omega_{i-1}\|} \right\|^2 = \left\| \frac{\log(q_i^{-1}q_{i+1})}{\|\log(q_i^{-1}q_{i+1})\|} - \frac{\log(q_{i-1}^{-1}q_i)}{\|\log(q_{i-1}^{-1}q_i)\|} \right\|^2 \quad (5 - 77)$$

Anizotropowy współczynnik tłumienia będzie dążył do zera, gdy krzywizna będzie dążyć do nieskończoności. Dzięki temu dyfuzja w tym fragmencie zostanie odpowiednio szybko zahamowana, co pozwala na zachowanie kształtu trajektorii. Jeżeli natomiast krzywizna dąży do zera, to dyfuzja będzie realizowana szybciej – jest to fragment trajektorii, który jest wystarczająco płaski (z punktu widzenia krzywej w  $S^3$ ), że wygładzanie przez dyfuzję nie zniekształci jego charakteru.

Pojęcie krawędzi uogólnione w rozdziale 5.3 może być analogicznie użyte do opisu kształtu trajektorii z punktu widzenia jej krzywizny. Krawędzią można nazwać nie tylko nagłą zmianę szybkości ruchu (jak to było w poprzednim przypadku), ale również znaczną zmianę krzywizny trajektorii. Pojęcia te można w dalszym ciągu rozszerzać na kolejne pochodne sygnału, które jednak z praktycznego punktu widzenia nie mają tak dużego znaczenia. Zaproponowane podejście realizuje założenie, aby znaczna zmiana krzywizny trajektorii nie była wygładzana – dyfuzja w tym fragmencie jest zahamowana przez użycie zmiennego współczynnika tłumienia z funkcją stopującą.

Należy podkreślić, że we wszystkich punktach dotyczących przyspieszenia, brane jest pod uwagę przyspieszenie styczne do kierunku ruchu. Tylko ta składowa jest odpowiedzialna za zmianę kierunku w przypadku ruchu w przestrzeni zakrzywionej. Dzięki odpowiedniej dyskretyzacji krzywizny trajektorii uwzględniona zostanie dowolna zmiana kierunku niezależnie od przyjętej osi obrotu przy realizacji rotacji (dalsze rozważania dotyczące niezmienności od współrzędnych przedstawione są w rozdziale 5.6).

Przedstawiony model filtracji można opisać następującymi podstawowymi własnościami:

- ze względu na nieliniowy charakter przestrzeni, zagadnienie początkowe opisujące proces ciągły jest nieliniowe,
- ze względu na współczynnik dyfuzji  $\tilde{c}$  proces jest nieliniowy oraz niehomogeniczny (współczynnik jest nieliniową funkcją trajektorii  $q$ ),
- ze względu na zmienny współczynnik tłumienia  $\tilde{c}_A$  otrzymuje się nieliniowy proces anizotropowy (współczynnik zależy od zmiany kierunku trajektorii modelowanego przez drugą pochodną przestrzenną określającą krzywiznę).

Dzięki wymienionym własnościom filtracja realizowana zaproponowaną metodą pozwala na uzyskanie następujących efektów:

- energia sygnału zostaje zminimalizowana dzięki zastosowaniu procesu dyfuzyjnego,

- filtracja zachowuje geometrię trajektorii charakteryzowaną przez prędkość kątową dzięki nieliniowemu charakterowi współczynnika dyfuzji,
- filtracja zachowuje geometrię trajektorii charakteryzowaną przez lokalną krzywiznę dzięki anizotropowemu charakterowi współczynnika tłumienia.

Pozostałe właściwości zaproponowanego modelu zostaną zebrane i zaprezentowane w rozdziale 5.6.

### 5.4.3 Algorytm filtracji anizotropowej

Algorytm realizujący zaproponowaną filtrację nazwany został *AnisoQuat* ze względu na anizotropowe przeprowadzenie filtracji przez zmienny współczynnik tłumienia. Podobnie, jak w przypadku algorytmów *LinQuat* oraz *NonLinQuat* przed rozpoczęciem działania należy przeprowadzić wstępne przygotowanie trajektorii kwaternionów przy użyciu algorytmu 3.

Istotnym elementem filtracji jest prawidłowe wyznaczenie krzywizny trajektorii, która będzie odpowiadać za właściwe zachowanie kształtu trajektorii. Należy pamiętać, że algorytm będzie stosowany dla sygnałów zawierających sygnał zakłócający. W praktyce sygnał ten może wpłynąć niekorzystnie na określenie krzywizny trajektorii. Może to prowadzić do uzyskania błędnych rezultatów, w których zakłócenia zamiast zostać usunięte, będą uwydatnione. Analogiczny problem występował przy realizacji filtracji obrazów dwuwymiarowych, dla której współczynnik dyfuzji zależał od funkcji jasności pikseli. Aby zminimalizować wpływ zaszumienia na stabilność procesu proponuje się przeprowadzić regularyzację.

#### Regularyzacja procesu

Jedną z metod regularyzacji nieliniowego modelu dla obrazów było wstępne wygładzenie warunku początkowego (funkcji jasności pikseli) przy użyciu liniowej filtracji realizowanej przez splot z jądrem Gaussa (2 – 64). W celu wyznaczenia lokalnej krzywizny trajektorii proponuje się podobne podejście realizowane przez wstępną filtrację niezależną od wartości funkcji  $q$ . W tym celu można przykładowo użyć wyrażenia (5 – 13) będącego odpowiednikiem filtru o jądrze Gaussa. W praktyce jednak dowolna metoda wygładzania trajektorii dobrze realizuje zmniejszenie podatności metody na zakłócenia występujące w sygnale. Proponuje się użycie następującego uśrednienia 3 elementów opartego na wyrażeniu na średnie położenie kwaternionowe (4 – 105). Estymator średniego położenia odpowiednio dla elementu poprzedzającego i następującego po  $q_i$  wyrażony zostanie jako:

$$\tilde{q}_{MP} = q_{i-3} \left( q_{i-3}^{-1} \left( q_{i-2} \left( q_{i-2}^{-1} q_{i-1} \right)^{1/2} \right) \right)^{2/3} \quad (5 - 78)$$

$$\tilde{q}_{MN} = q_{i+3} \left( q_{i+3}^{-1} \left( q_{i+2} \left( q_{i+2}^{-1} q_{i+1} \right)^{1/2} \right) \right)^{2/3} \quad (5 - 79)$$

Należy pamiętać, że wyrażenia te nie są wartością średnią dla grupy kwaternionów [14]. Istnieje kilka sposobów wyznaczenia średniej wartości kwaternionowej, które zostały szczegółowo omówione w rozdziale 4.2.6. Jednakże algorytm podany w pracy [48] przydatny jest przede wszystkim dla odpowiednio dużej statystycznie próby losowej. Natomiast podejście zaproponowane w pracy [11] oparte jest na algorytmie minimalizacji w kierunku, która może wymagać wielu iteracji dla osiągnięcia rozwiązania przybliżonego. Wyrażenie (5 – 80) jest przybliżone z punktu widzenia własności statystycznych w

zależności od rozkładu elementów – im większa odległość pomiędzy analizowanymi kwaternionami, tym mniej dokładny otrzymany wynik. Z praktycznego punktu widzenia występują dwie przesłanki uzasadniające użycie takiego podejście. Rozpatrywane elementy są zazwyczaj położone blisko siebie (ponieważ są następującymi po sobie elementami trajektorii), więc takie uśrednianie nie jest obciążone dużym błędem. Dodatkowo dokładna wartość średniej nie jest niezbędna dla prawidłowego działania algorytmu. Podejście to ma jedynie za zadanie regularyzować proces w sytuacji występowania znacznego zaszumienia procesu i do takiego celu zaproponowane wyrażenia przybliżone są wystarczające.

### Konstrukcja algorytmu

Regularyzacja procesu powoduje, że anizotropowy współczynnik tłumiący zależy od zmiany prędkości pomiędzy kwaternionem  $q_i$  oraz uśrednionymi kwaternionami poprzedzającym go  $\tilde{q}_{MP}$  i następującym  $\tilde{q}_{MN}$  (indeks  $M$  wskazuje na wielkość uśrednioną). Jednym z parametrów dyfuzji anizotropowej jest współczynnik wrażliwości  $K_A$  dla funkcji stopującej  $g_A$ . Zarówno rodzaj wybranej funkcji, jak i wartość jej współczynnika mogą być określone niezależnie od ogólnej funkcji stopującej dla uwzględniania zmienności położenia. Dodatkowo, aby móc kontrolować szybkość dyfuzji anizotropowej wprowadza się analogiczny współczynnik tłumiący. Dla czytelności i zgodności oznaczeń w dalszej części pracy przyjmuje się, że anizotropowy współczynnik tłumienia dyfuzji kierunkowej oznaczany będzie  $\lambda_A$ , natomiast współczynnik dyfuzji kierunkowej wyrażony wzorem (5 – 73) oznaczony będzie, jako:

$$\tilde{c}_A = \lambda_A g(\kappa_i) = \lambda_A g\left(\left\|\frac{\log(q_i^{-1}q_{i+1})}{\|\log(q_i^{-1}q_{i+1})\|} - \frac{\log(q_{i-1}^{-1}q_i)}{\|\log(q_{i-1}^{-1}q_i)\|}\right\|^2\right) \quad (5 - 80)$$

Zakładając, że warunki brzegowe zostały uwzględnione w postaci przedstawionej w algorytmie 5 (opis warunków jest identyczny, jak w przypadku algorytmu *NonLinQuat*) proponuje się filtrację anizotropową przy użyciu następującego algorytmu *AnisoQuat*.

---

#### Algorytm 7: Nieliniowa anizotropowa filtracja trajektorii kwaternionów *AnisoQuat*

Parametry:  $q = \{q_0, \dots, q_i, \dots, q_n\}$  – trajektoria kwaternionów o długości  $n$

$t_{\max}$  - liczba iteracji algorytmu

$\lambda \in (0, 1]$  - współczynnik szybkości filtracji (tłumiący)

$\lambda_A \in (0, 1]$  - opcjonalny anizotropowy współczynnik szybkości filtracji (tłumiący), domyślnie  $\lambda_A = 1$

$K$  – współczynnik wrażliwości dla funkcji stopującej  $g$

$K_A$  – współczynnik wrażliwości dla anizotropowej (kierunkowej) funkcji stopującej  $g_A$

$A, B, C, D$  – rodzaj warunku brzegowego (rozdział 5.2.1)

**krok 1:** Podstaw  $t = 0$

**krok 2:** Dla  $t < T_{\max}$  wykonuj

**krok 3:** Dla każdego elementu trajektorii o indeksie  $0 < i < n$  wykonaj

**krok 4:** Wyznacz  $\tilde{c}_p = \lambda g(\|2 \log(q_i^{-1}q_{i-1})\|)$

---

- krok 5:** Wyznacz  $\tilde{c}_N = \lambda g(\|2 \log(q_i^{-1} q_{i+1})\|)$
- krok 6:** Wyznacz  $\tilde{q}_P = q_i (q_i^{-1} q_{i-1})^{\tilde{c}_P}$
- krok 7:** Wyznacz  $\tilde{q}_N = q_i (q_i^{-1} q_{i+1})^{\tilde{c}_N}$
- krok 8:** Wyznacz  $\tilde{q}_i = \tilde{q}_P (\tilde{q}_P^{-1} \tilde{q}_N)^{1/2}$
- krok 9:** Wyznacz  $\tilde{q}_{MP}, \tilde{q}_{MN}$  - równania (5 - 78), (5 - 79)
- krok 10:** Wyznacz  $\kappa_i = \left\| \frac{\log(q_i^{-1} \tilde{q}_{MN})}{\|\log(q_i^{-1} \tilde{q}_{MN})\|} - \frac{\log(q_i^{-1} \tilde{q}_{MN})}{\|\log(q_i^{-1} \tilde{q}_{MN})\|} \right\|^2$
- krok 11:** Wyznacz  $\tilde{c}_A = \lambda_A g_A(\kappa_i)$  przy parametrze  $K_A$
- krok 12:** Podstaw  $q_i^{(t+1)} = q_i (q_i^{-1} \tilde{q})^{\tilde{c}_A}$
- krok 13:** Podstaw  $t = t \leftarrow 1$

### Uwzględnienie warunków brzegowych

Dla zachowania czytelności w powyższym algorytmie nie uwzględniono wszystkich szczególnych przypadków występujących dla warunków brzegowych określonych w rozdziale 5.2.1. Warunki te oraz ich realizacja przy wyznaczaniu krzywizny zostaną zebrane w punktach.

#### Warunek A – stała wartość na granicach

Modyfikacja położenia elementów o indeksach  $i = 0$  lub  $i = n$  nie jest dokonywana. Dla kolejnych elementów  $i \leq 2$  należy zmodyfikować sposób wyznaczenia krzywizny przez zmianę równania (5 - 78):

$$\text{dla } i=1 \quad \tilde{q}_{MP} = q_{i-1} \quad (5 - 81)$$

$$\text{dla } i=2 \quad \tilde{q}_{MP} = q_{i-2} (q_{i-2}^{-1} q_{i-1})^{1/2} \quad (5 - 82)$$

Analogicznie należy zmodyfikować wyznaczanie kwaternionu  $\tilde{q}_{MN}$  dla elementów  $i \geq n - 2$ .

#### Warunek B – układ odizolowany

W tym przypadku zakłada się, że współczynnik dyfuzji na granicach obszaru jest równy zero. Proponuje się przyjęcie następujących wartości dla elementów brzegowych.

dla  $i=0$ :

$$\text{dyfuzja położenia: } \tilde{c}_P = 0, \tilde{c}_N = g(\|2 \log(q_0^{-1} q_1)\|), \tilde{q}_P = q_0, \tilde{q}_N = q_0 (q_0^{-1} q_1)^{\tilde{c}_N}$$

$$\text{dyfuzja kierunkowa: } \tilde{q}_{MP} = q_0 \Rightarrow \kappa_i = 1$$

dla  $i=n$ :

$$\text{dyfuzja położenia: } \tilde{c}_P = g(\|2 \log(q_n^{-1} q_{n-1})\|), \tilde{c}_N = 0, \tilde{q}_P = q_n (q_n^{-1} q_{n-1})^{\tilde{c}_P}, \tilde{q}_N = q_n$$

dyfuzja kierunkowa:  $\tilde{q}_{MN} = q_n \Rightarrow \kappa_i = 1$

Wyznaczenie wartości  $\tilde{q}_{MP}$ ,  $\tilde{q}_{MN}$  odbywa się przy analogicznych ograniczeniach, jak przy warunku A – równania (5 – 81), (5 – 82).

### Warunek C – sygnał periodyczny

Warunek brzegowy tego rodzaju jest realizowany przez założenie, że sygnał jest periodyczny, czyli elementy końcowe sąsiadują ze sobą.

dla  $i=0$ :

dyfuzja położenia:  $\tilde{c}_P = g\left(\left\|2 \log(q_0^{-1} q_n)\right\|\right)$ ,  $\tilde{c}_N = g\left(\left\|2 \log(q_0^{-1} q_1)\right\|\right)$ ,  $\tilde{q}_P = q_0(q_0^{-1} q_n)^{\tilde{c}_P}$ ,  $\tilde{q}_N = q_0(q_0^{-1} q_1)^{\tilde{c}_N}$

dyfuzja kierunkowa:  $\tilde{q}_{MP} = q_{n-2} \left( q_{n-2}^{-1} \left( q_{n-1}^{-1} (q_{n-1} q_n)^{1/2} \right) \right)^{2/3}$

dla  $i=n$ :

dyfuzja położenia:  $\tilde{c}_P = g\left(\left\|2 \log(q_n^{-1} q_{n-1})\right\|\right)$ ,  $\tilde{c}_N = g\left(\left\|2 \log(q_n^{-1} q_0)\right\|\right)$ ,  $\tilde{q}_P = q_n(q_n^{-1} q_{n-1})^{\tilde{c}_P}$ ,  $\tilde{q}_N = q_n(q_n^{-1} q_0)^{\tilde{c}_N}$

dyfuzja kierunkowa:  $\tilde{q}_{MN} = q_2 \left( q_2^{-1} \left( q_1^{-1} (q_1 q_0)^{1/2} \right) \right)^{2/3}$

### Warunek D – odbicie sygnału

Warunek brzegowy tego rodzaju jest realizowany przez założenie, że dziedzina funkcji jest rozszerzona na dziedzinę nieskończoną przez odbicie sygnału na brzegach dziedziny oryginalnej.

dla  $i=0$ :

dyfuzja położenia:  $\tilde{c}_P = g\left(\left\|2 \log(q_0^{-1} q_1)\right\|\right)$ ,  $\tilde{c}_N = g\left(\left\|2 \log(q_0^{-1} q_1)\right\|\right)$ ,  $\tilde{q}_P = q_0(q_0^{-1} q_1)^{\tilde{c}_P}$ ,  $\tilde{q}_N = q_0(q_0^{-1} q_1)^{\tilde{c}_N}$

dyfuzja kierunkowa:  $\tilde{q}_{MP} = q_3 \left( q_3^{-1} \left( q_2^{-1} (q_2 q_1)^{1/2} \right) \right)^{2/3}$

dla  $i=n$ :

dyfuzja położenia:  $\tilde{c}_P = g\left(\left\|2 \log(q_n^{-1} q_{n-1})\right\|\right)$ ,  $\tilde{c}_N = g\left(\left\|2 \log(q_n^{-1} q_{n-1})\right\|\right)$ ,  $\tilde{q}_P = q_n(q_n^{-1} q_{n-1})^{\tilde{c}_P}$ ,  $\tilde{q}_N = q_n(q_n^{-1} q_{n-1})^{\tilde{c}_N}$

dyfuzja kierunkowa:  $\tilde{q}_{MN} = q_{n-3} \left( q_{n-3}^{-1} \left( q_{n-2}^{-1} (q_{n-2} q_{n-1})^{1/2} \right) \right)^{2/3}$

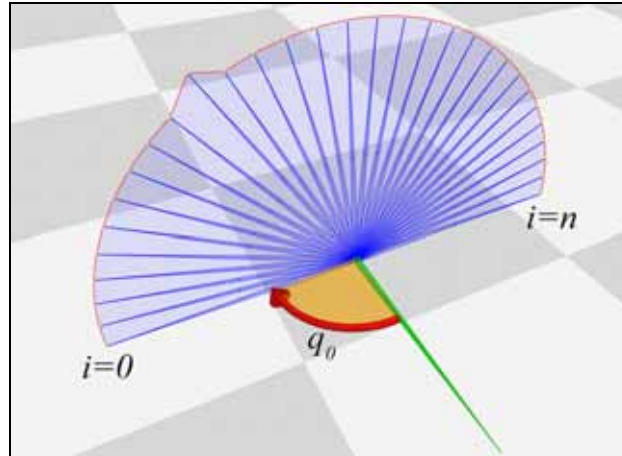
## 5.5 Geometryczna interpretacja metod opartych na równaniach dyfuzji

Aby lepiej zobrazować kolejne kroki zaproponowanych algorytmów filtracji trajektorii orientacji w przestrzeni zaprezentowana zostanie ich geometryczna reprezentacja. Ilustracją będą syntetycznie wygenerowane proste sekwencje przedstawione w postaci kolejnych położenia przykładowego elementu pełniącego funkcję wektora w lokalnym układzie współrzędnych. Aby zachować czytelność rysunków, rozpatrywane będą tylko dwie składowe rotacji przedstawionej w postaci trajektorii zakreślonej przez obiekt rotujący obiekt w przestrzeni trójwymiarowej. Należy jednak pamiętać, że w proponowanej metodzie rotacje wokół wszystkich trzech osi są rozpatrywane w jednakowy sposób. Wynika to z zalet przyjętej

parametryzacji kwaternionowej, która nie wyróżnia żadnej z orientacji lokalnego układu współrzędnych (patrz własności procesu zebrane w rozdziale 5.6.

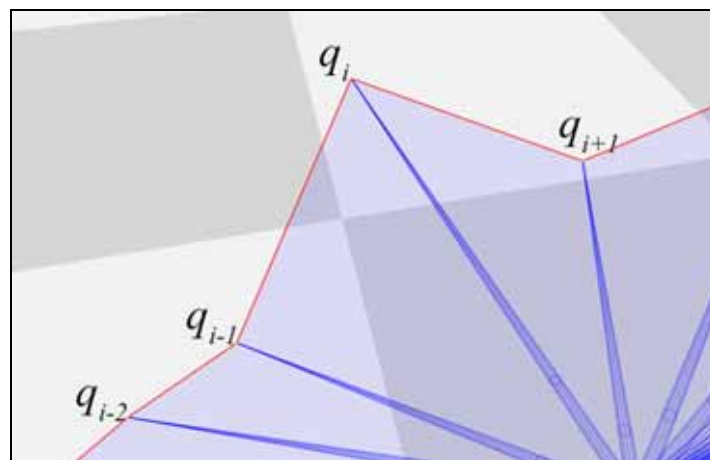
### 5.5.1 Algorytm filtracji liniowej *LinQuat*

Filtracja będąca odpowiednikiem liniowego procesu dyfuzji (rozdział 5.2) zostanie zaprezentowany na podstawie podstawowej sekwencji przedstawionej na rysunku 5-2.



**Rysunek 5-2:** Sekwencja użyta do geometrycznej interpretacji filtracji liniowej. Kolorem zielonym zaznaczono położenie referencyjne. Kolorem niebieskim zaznaczono orientacje układu w kolejnych chwilach czasowych.

Prezentowana sekwencja składa się z trajektorii w przestrzeni orientacji, która jest zapisana w postaci  $n$  dyskretnych transformacji lokalnego układu współrzędnych (trajektoria na rysunku składa się z 30 elementów). Na rysunku kolorem zielonym zaznaczono przykładowy wektor referencyjny, względem którego dokonywana jest transformacja w każdej chwili czasowej. Kolejne położenia otrzymane przez realizację odpowiedniej rotacji zaznaczone są kolorem niebieskim. Dla łatwiejszego śledzenia położeń w kolejnych chwilach czasowych połączone one zostały czerwoną trajekcją, a kąt przestrzenny pomiędzy nimi zaznaczono kolorem jasnoniebieskim. Dodatkowo zaznaczono przykładową transformację orientacji lokalnego układu współrzędnych opisaną kwaternionem  $q_0$ .



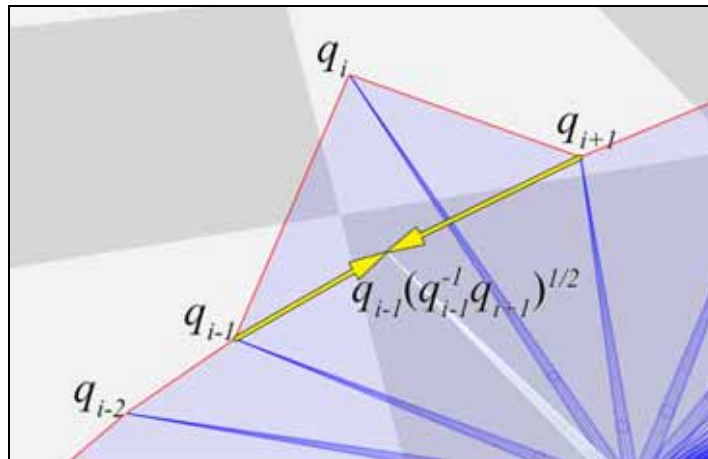
**Rysunek 5-3:** Wybrany fragment trajektorii, na którym wyróżniony została zakłócona orientacja uzyskana przez rotację opisaną kwaternionem  $q_i$ .



Na rysunku widoczne jest, że orientacja układu w przypadku pojedynczego położenia została zakłócona (wybrany fragment trajektorii pokazano na rysunku 5-3). Na przykładzie tej chwili czasowej zostaną rozpatrzone kolejne kroki algorytmu filtracji, który ma za zadanie usunąć zakłócenie z podanej sekwencji.

W kroku 4 algorytmu *LinQuat* (rozdział 5.2.3) wykonywana jest następująca interpolacja, której przykładową geometryczną interpretacją przedstawiono na rysunku 5-4:

$$\hat{q} = q_{i-1} (q_{i-1}^{-1} q_{i+1})^{1/2} \quad (5 - 83)$$

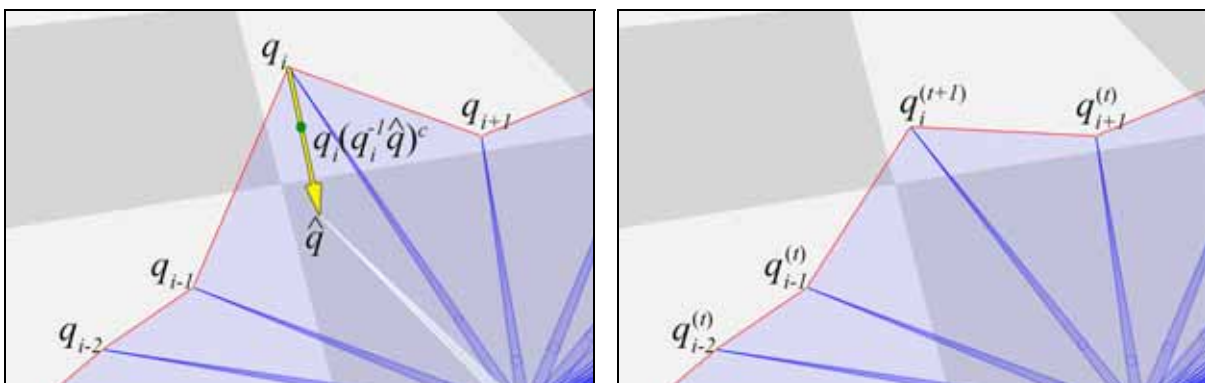


Rysunek 5-4: Krok 4 algorytmu *LinQuat*: interpolacja położenia sąsiadujących elementów – równanie (5 – 83).

W wyniku tej operacji otrzymuje się kwaternion wynikowy  $\hat{q}$ , który jest następnie operandem w interpolacji przeprowadzonej w kroku 5:

$$q_i^{(t+1)} = q_i (q_i^{-1} \hat{q})^c \quad (5 - 84)$$

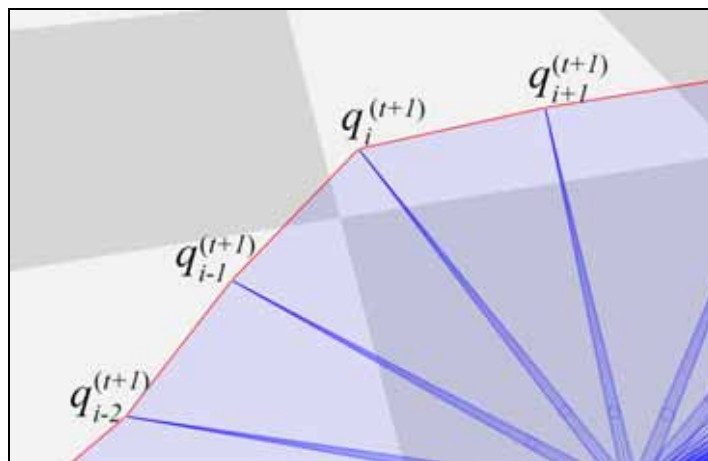
Reprezentację położenia wynikowego zaznaczono kolorem jasnoniebieskim. Położenie to nie należy do sekwencji wynikowej, lecz jest niezbędne do wyznaczenia końcowego wyniku przetwarzania elementu. Rysunek 5-5 przedstawia wizualizację kierunku wzdłuż którego interpolacja jest dokonywana oraz rezultat operacji.



Rysunek 5-5: Wizualizacja kroku 5 algorytmu *LinQuat*: interpolacja położenia elementu przekształcanego w zależności od współczynnika dyfuzji – równanie (5 – 84).

Zielony punkt na rysunku 5-5 reprezentuje wartość współczynnika  $c$  odpowiedzialnego za szybkość dyfuzji. Mała wartość współczynnika powoduje, że proces filtracji będzie zachodził wolniej. Duża wartość współczynnika może powodować niestabilność zastosowanego schematu dyskretyzacji.

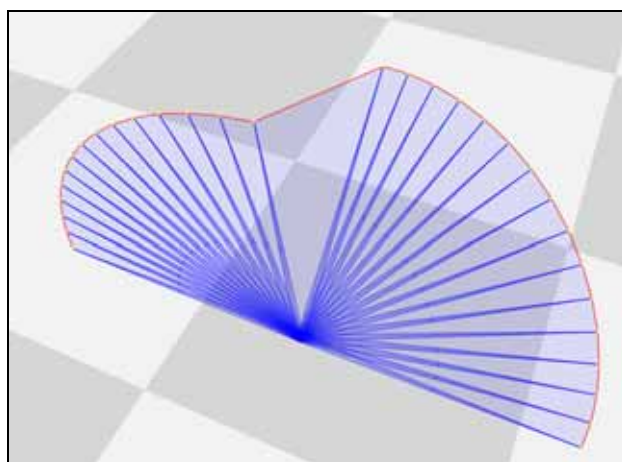
Cały algorytm oparty jest na przetwarzaniu kolejnych elementów dyskretnej trajektorii. Rysunek 5-6 przedstawia wizualizację przykładowej sekwencji po wykonaniu jednej pełnej iteracji algorytmu. Widoczne jest, że położenie zarówno elementu  $q_i$ , jak i elementów sąsiadujących zostało zmodyfikowane. Jest to zgodne z intuicją, że zakłócenie wprowadzone do układu będzie wpływało na wartości funkcji w innych chwilach czasowych.



Rysunek 5-6: Przykładowa sekwencja orientacji układu po wykonaniu jednej pełnej iteracji algorytmu *LinQuat*.

### 5.5.2 Algorytm filtracji liniowej *NonLinQuat*

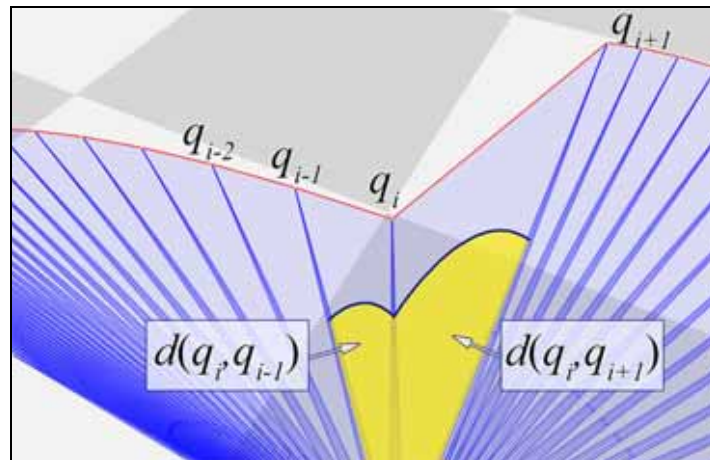
Zobrazowanie sposobu działania filtracji nieliniowej (rozdział 5.3) zostanie pokazane na przykładzie sekwencji trajektorii, w której w pewnym momencie prędkość zmienia znacząco swoją wartość. Jakkolwiek metoda dotyczy zmiany skalarnej odległości (bez uwzględniania części wektorowej), to w celu zwiększenia czytelności, w przykładowej sekwencji ruch uwzględnia również zmianę kierunku. W ten sposób w wygodniejszy sposób możliwa jest wizualizacja kolejnych kroków algorytmu. Cała przykładowa sekwencja pokazana jest na rysunku 5-7.



Rysunek 5-7: Sekwencja użyta do geometrycznej interpretacji filtracji nieliniowej.

Algorytm *NonLinQuat* realizuje filtrację w zależności od wartości prędkości kątowej, z jaką jest wykonywany ruch wzdłuż trajektorii. Na rysunku 5-8 przedstawiono fragment, w którym występuje nagła

zmiana prędkości wraz z wizualizacją miary odległości położenia pomiędzy klatką poprzedzającą i następującą po klatce przekształcanej  $q_i$ .



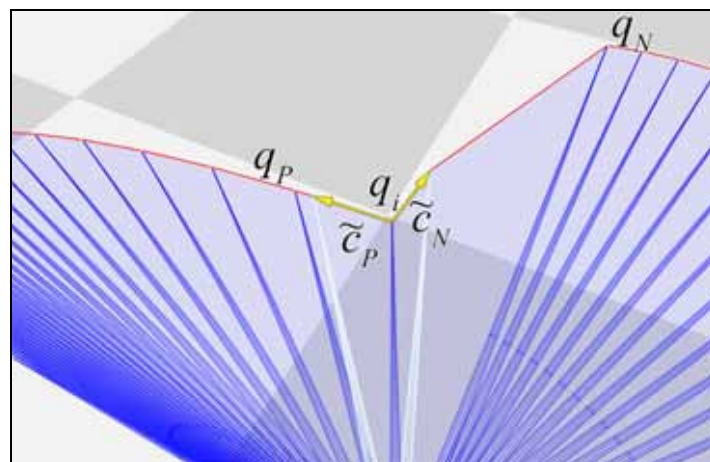
**Rysunek 5-8: Wybrany fragment trajektorii, w którym występuje nagła zmiana prędkości kątowej ruchu.**

Otrzymane odległości są wykorzystywane w kroku 17 i 18 algorytmu w celu obliczenia lokalnych nieliniowych współczynników interpolacji wykonywanej w kierunku położenia poprzedniego i następnego:

$$\tilde{c}_p = \lambda g(\|2 \log(q_i^{-1} q_p)\|) \quad (5-85)$$

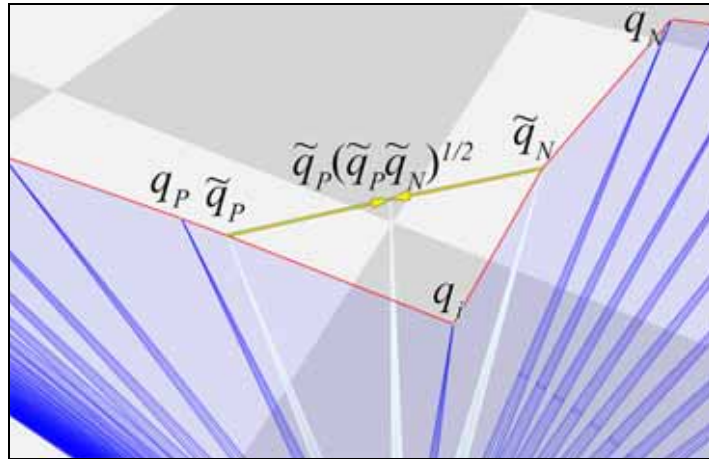
$$\tilde{c}_N = \lambda g(\|2 \log(q_i^{-1} q_N)\|) \quad (5-86)$$

Im większa jest kwaternionowa miara odległości przedstawiona wzorami (5-85), (5-86) tym mniejszy będzie współczynnik lokalnej dyfuzji z danym kierunkiem. Wyznaczenie tymczasowych kwaternionów  $\tilde{q}_p$  oraz  $\tilde{q}_N$  pokazane jest na rysunku 5-9, co jest realizowane przez krok 19 i 20 algorytmu *NonLinQuat*.

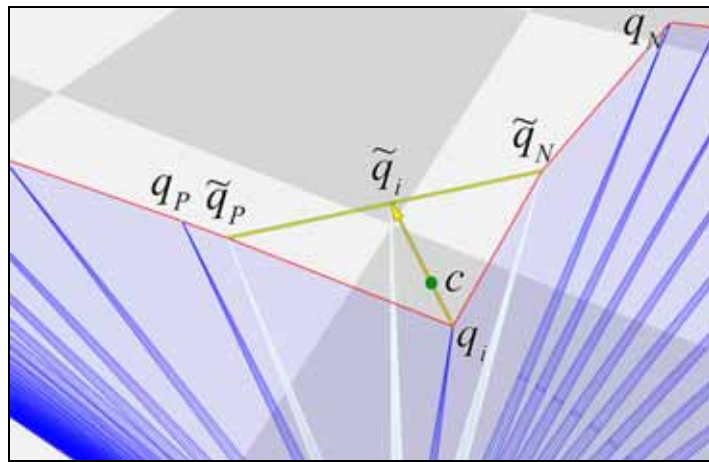


**Rysunek 5-9: Wybrany fragment trajektorii – interpolacja w kierunku położenia poprzedniego i następnego (krok 19 i 20 algorytmu).**

Otrzymany rezultat w postaci dwóch kwaternionów pokazany jest w zbliżeniu na rysunku 5-10.



Rysunek 5-10: Wybrany fragment trajektorii – interpolacja do pośredniego kwaternionu wynikowego.



Rysunek 5-11: Wybrany fragment trajektorii – uwzględnienie czynnika tłumiącego.

Kwaterniony tymczasowe są potrzebne do wyznaczenia w kroku 21 średniego położenia opisanego kwaternionem  $\tilde{q}_i$  - równanie (5 – 87). Gdyby nie został wprowadzony dodatkowy czynnik tłumiący, to kwaternion ten byłby wynikiem przetwarzania elementu  $q_i$  :

$$\tilde{q}_i = \tilde{q}_P (\tilde{q}_P^{-1} \tilde{q}_N)^{1/2} \quad (5 - 87)$$

Wprowadzenie czynnika tłumiącego pokazane jest na rysunku 5-11, co jest realizacją kroku 22 algorytmu opisanego równaniem:

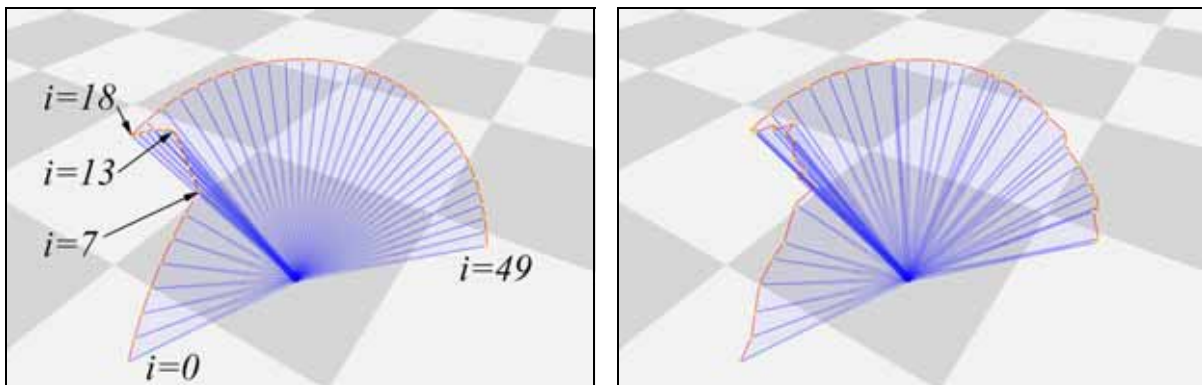
$$q_i^{(t+1)} = q_i (q_i^{-1} \tilde{q})^c \quad (5 - 88)$$

Kierunek interpolacji został zaznaczony strzałką. Zielony punkt reprezentuje miejsce, do którego interpolacja zostanie wykonana. Wprowadzenie końcowej interpolacji pomaga więc stabilizować proces oraz regulować szybkość jego przebiegu.

Podobnie, jak w przypadku algorytmu liniowego, modyfikacja położenia odnosi się do wszystkich elementów dyskretnej trajektorii. Podstawową różnicą pomiędzy tym algorytmów dotyczy mniejszego wygładzania fragmentów, w których występują duże zmiany prędkości. Podejście to nie uwzględnia wektora prędkości, lecz jej wartość skalarną, której wizualizację przedstawiono. Dalsze rozwinięcie tej metody znajduje swoją realizację w anizotropowej filtracji orientacji układów.

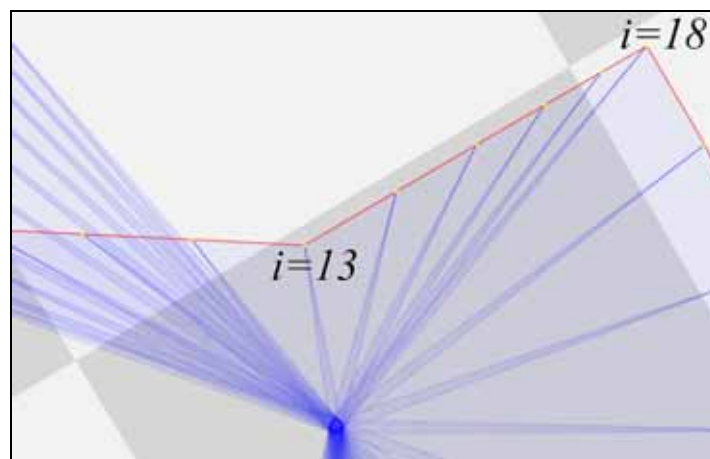
### 5.5.3 Algorytm filtracji anizotropowej *AnisoQuat*

Algorytm anizotropowy przedstawiony w rozdziale 5.4 jest rozwinięciem algorytmu filtracji nieliniowej. Z tego powodu zostaną przedstawione najważniejsze różnice, które wynikają z proponowanego podejścia uogólnionego. W szczególny sposób jest w tym algorytmie uwzględniana zmiana kierunku wykonywanego ruchu. Aby własności proponowanego podejścia były czytelnie przedstawione, zostanie użyta trajektoria orientacji, w której występują zmiany kierunku ruchu o różnym charakterze. Na rysunku 5-12 przedstawiono trajektorię testową użytą w dalszej części rozdziału. Prawa kolumna rysunku przedstawia trajektorię testową wraz z wprowadzonym sygnałem zakłócającym – przebieg zostanie użyty przy analizie sposobów wyznaczania krzywizny trajektorii.



**Rysunek 5-12: Sekwencja użyta do geometrycznej interpretacji filtracji anizotropowej. Na rysunku zaznaczono chwile czasowe, w których dokonywana jest zmiana kierunku ruchu. Kolumna prawa – przebieg testowy z wprowadzonym sygnałem zakłócającym.**

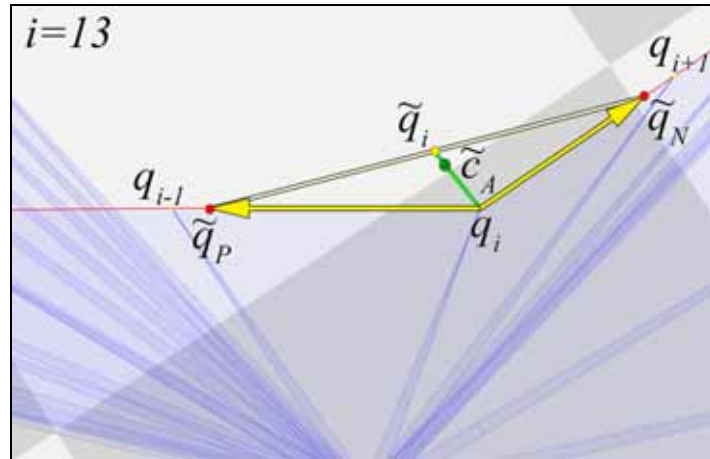
Na rysunku zostały zaznaczone chwile czasowe, które definiują klatki kluczowe przebiegu. W chwilach tych zmienia się kierunek ruchu utworzony przez zastosowanie interpolacji sferyczno-liniowej pomiędzy klatkami kluczowymi. Rysunek 5-13 szczegółowo przedstawia dwie chwile czasowe prezentowanej trajektorii testowej (niezakłóconej), dla których będą analizowane kolejne kroki algorytmu filtracji.



**Rysunek 5-13: Wybrane chwile czasowe trajektorii, dla których będzie przeprowadzona analiza filtracji anizotropowej. Należy zwrócić uwagę, że w wybranych momentach kierunek ruchu zmienia się w różny sposób.**

Kroki algorytmu od 4 do 8 są wykonywane w taki sam sposób, jak w przypadku algorytmu nieliniowego. Na rysunku 5-14 zaznaczono wyniki częściowe i końcowe poszczególnych kroków filtracji

anizotropowej. Nieliniowe wyznaczenie położenia opisanego przez kwaterniony  $\tilde{q}_P$ ,  $\tilde{q}_N$  zaznaczone jest czerwonym punktem (krok 6, 7). W kroku 8 wyznaczany jest pośredni wynik filtracji opisany kwaternionem  $\tilde{q}_i$  (żółty punkt). Ostateczne położenie otrzymuje się przez interpolację wzdłuż kierunku zaznaczonego kolorem jasnozielonym (krok 12). Zielony punkt oznacza końcowy wynik filtracji określony współczynnikiem współczynnik anizotropowej składowej dyfuzji  $\tilde{c}_A$  (krok 11).

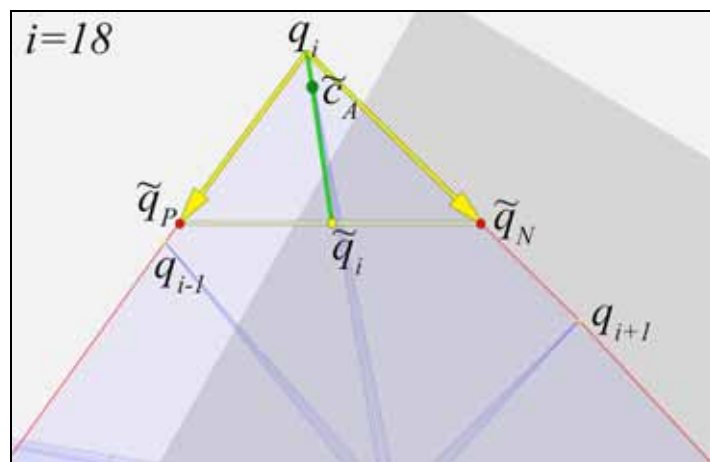


Rysunek 5-14: Realizacja iteracji filtracji anizotropowej dla chwili czasowej  $i=13$  przebiegu testowego.

Najważniejsza różnica w procesach filtracji w proponowanych algorytmach dotyczy kroków 10 – 12. W kroku 10 wyznaczana jest lokalna krzywizna trajektorii w punkcie  $q_i$ :

$$\kappa_i = \left\| \frac{\log(q_i^{-1}q_{i+1})}{\|\log(q_i^{-1}q_{i+1})\|} - \frac{\log(q_{i-1}^{-1}q_i)}{\|\log(q_{i-1}^{-1}q_i)\|} \right\|^2 \quad (5 - 89)$$

Na podstawie tej wartości jest określany współczynnik anizotropowej składowej dyfuzji  $\tilde{c}_A$  tak, aby dyfuzja była zmniejszana dla zwiększającej się wartości lokalnego oszacowania krzywizny. Na rysunku 5-15 zaznaczono te rezultaty tych samych kroków algorytmu, jednak dla innej chwili czasowej trajektorii  $i=18$ .



Rysunek 5-15: Realizacja iteracji filtracji anizotropowej dla chwili czasowej  $i=13$  przebiegu testowego.

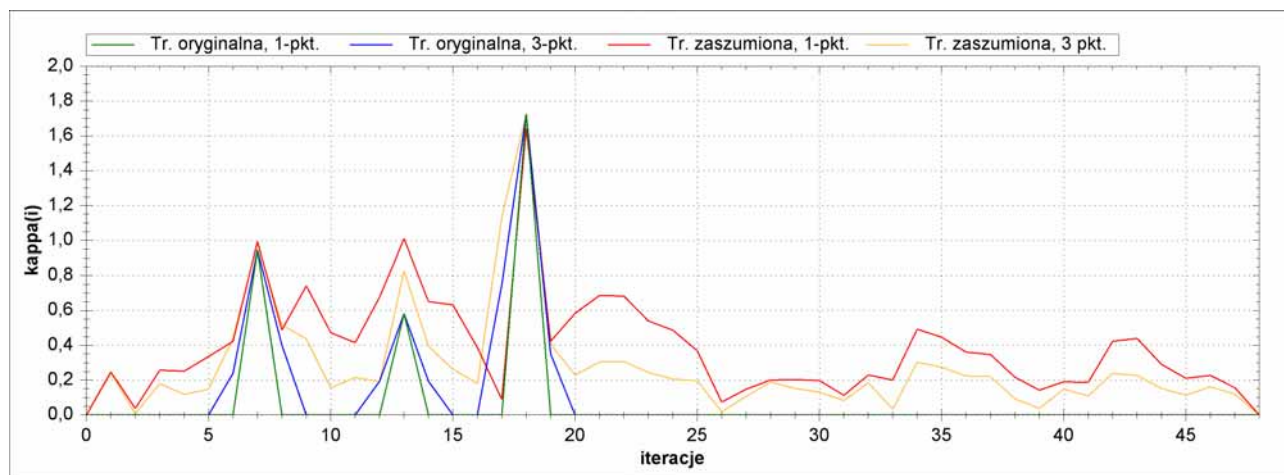
Ze względu na znaczną zmianę kierunku ruchu w tej chwili czasowej, zmniejszona została wartość współczynnika dyfuzji  $\tilde{c}_A$ . Stąd dyfuzja zostaje w danym fragmencie znacznie zahamowana. W pozostałych

obszarach trajektoria jest w dalszym ciągu wygładzana ze względu na mniejszą krzywiznę, która prowadzi do większej wartości  $\tilde{c}_A$ . Ostatecznie można scharakteryzować zaproponowany proces filtracji przez następującą własność.

### Własność 5.1

Algorytm filtracji realizujący anizotropowy proces dyfuzji trajektorii orientacji *AnisoQuat* wygładza przebieg czasowy zmniejszając zaszumienie sygnału przy jednoczesnym zachowywaniu kształtu zdefiniowanego przez oszacowanie lokalnej krzywizny. We fragmentach, dla których występuje znacząca zmiana krzywizny, wartość anizotropowego współczynnika dyfuzji (będącego funkcją krzywizny) jest zmniejszana.

Wykres na rysunku 5-16 przedstawia oszacowanie krzywizny wyznaczone przy użyciu wzoru (5 – 89).



**Rysunek 5-16: Oszacowanie lokalnej krzywizny trajektorii dla sygnału testowego (szczegółowy opis w tekście).**

Kolorami zielonym i niebieskim zaznaczono wartości  $\kappa_i$  dla oryginalnej trajektorii testowej (odpowiednio przy użyciu 1-punktowej i 3-punktowej wersji estymatora). Widoczne jest, że dla chwil czasowych  $i = 7, 13, 18$  zauważalny jest znaczny wzrost wyznaczonego współczynnika. Jest to zgodne z używanym przebiegiem testowym, dla którego w tych chwilach czasowych następuje zmiana kierunku ruchu. Najmniejsza zmiana występuje dla  $i = 13$ , co jest zgodne z otrzymanym oszacowaniem. Należy zwrócić uwagę, że zastosowana postać estymatora określa wartość bezwzględna zmiany bez rozróżniania jej kierunku. Jest to właściwość pożądana, gdyż szybkość dyfuzji nie powinna zależeć od tego, którego kierunku dotyczy zmiana krzywizny. Widać również, że krzywizna wyznaczana metodą 3-punktową (kolor zielony) bierze pod uwagę nie tylko pojedynczy punkt, ale pewne najbliższe sąsiedztwo. Jest to znak, że sposób wyznaczania w tym przypadku jest regularyzowany według zależności (5 – 78), (5 – 79).

Czerwony przebieg na przedstawionym wykresie jest wartością 1-punktowego estymatora krzywizny dla testowego sygnału zaszumionego (rysunek 5-12). Wprowadzenie niewielkiej składowej zakłócającej powoduje, że estymacja zmiany kierunku ruchu nie jest dobrze realizowana. Na podstawie wykresu trudno

jest rozróżnić, czy w danym fragmencie dyfuzja anizotropowa powinna zostać zahamowana (tj. czy oszacowanie krzywizny jest względnie duże). Stąd wynika konieczność zastosowania schematu alternatywnego. Możliwe jest użycie różnego rodzaju podejścia, między innymi pomocnicze przeprowadzenie lokalnej wstępnej liniowej filtracji dla badanego fragmentu trajektorii. Można także użyć trzypunktowej metody wyznaczania estymatora lokalnej krzywizny. Wyznaczone w ten sposób oszacowanie zaznaczone jest kolorem pomarańczowym na wykresie.

Na podstawie wykresu widoczne jest, że trzypunktowa metoda wyznaczania oszacowania krzywizny jest mniej wrażliwa na zakłócenia sygnału. W przypadku tej samej zaszumionej trajektorii możliwe jest wyznaczenie fragmentów, w których wyraźnie zauważalna jest zmiana kierunku wykonywania ruchu. W kolejnych iteracjach procesu filtracji wyznaczenie tych fragmentów będzie realizowane z jeszcze większą dokładnością ze względu na zmniejszający się poziom zakłóceń. Z tego powodu warto rozważyć adaptacyjny sposób wprowadzania składnika anizotropowego do procesu, który będzie zwiększał udział tego typu dyfuzji wraz ze zmniejszającą się zawartością zakłóceń na obrazie. Podejście takie byłoby podobne do adaptacyjnego sposobu wyznaczania współczynnika  $K$  dla procesu filtracji obrazów (rozdział 2.5.1) oraz zmiennego współczynnika sprzężenia wielokanałowego procesu dyfuzji (rozdział 3.3.2).

Trzypunktowa metoda wyznaczania oszacowania krzywizny jest jednym ze sposobów regularyzacji procesu filtracji. Warunki narzucone na parametry procesu (rozdział 5.4.3) zapewniają, że otrzymany wynik nie będzie naruszał statycznych ograniczeń związanych z własnościami przestrzeni kwaternionów. Modyfikacja sposobu wyznaczania oszacowania krzywizny jest odpowiednikiem regularyzacji procesu filtracji obrazów (rozdział 2.5.3) i pozwala na zmniejszenie wpływu zakłóceń na uzyskanie poprawnego, odfiltrowanego rezultatu. Bez wprowadzenia tego typu podejścia istniałoby większe prawdopodobieństwo otrzymania wyniku nieprawidłowego, w przypadku którego proces szybko osiąga minimum lokalne (w odniesieniu do minimalizacji energii), lecz pozostaje część zakłóceń, które nie mogą zostać usunięte.

Należy pamiętać, że przedstawiona interpretacja geometryczna została zobrazowana położeniami odpowiedników wektorów w przestrzeni trójwymiarowej (przy uwzględnieniu dwóch stopni swobody). Zaproponowany właściwy proces filtracji jest przeprowadzany bezpośrednio w przestrzeni kwaternionów. Dzięki temu uwzględniane są wszystkie stopnie swobody związane z orientacją lokalnego układu współrzędnych (obroty we wszystkich dostępnych osiach). Własność ta wynika bezpośrednio z właściwości operacji użytych do dyskretyzacji równania opisującego procesu dyfuzji.

## **5.6 Własności filtracji dla trajektorii w $S^3$ opartych na równaniu różniczkowym dyfuzji**

Zaproponowane algorytmy filtracji trajektorii orientacji układu zostaną przeanalizowane ze względu na podstawowe własności, jakie charakteryzują sposób działania filtrów cyfrowych. W kilku przypadkach prezentowanych własności nie jest możliwe ich wykazanie przy użyciu standardowej interpretacji filtrów opartych na sygnałach rzeczywistych. Zostanie zaproponowana interpretacja własności oparta na geometrycznym opisie przestrzeni kwaternionowej i parametryzacji przestrzeni orientacji układu. Dowody



wybranych własności oparte są częściowo na schemacie prezentowanym w pracy [66], w której jednak rozważane są innego rodzaju przekształcenia (patrz rozdział 5.1).

W niniejszym rozdziale wprowadzone zostają następujące oznaczenia, aby możliwe było czytelne i spójne sformułowanie własności i dowodów.

- $F_L$  - przekształcenie (filtracja) typu *LinQuat*

$$F_L(q_i) = q_i \left( q_i^{-1} q_{i-1} (q_{i-1}^{-1} q_{i+1})^{1/2} \right)^c = q_i \exp \left( c \log \left( q_i^{-1} q_{i-1} \exp \left( 0.5 \log \left( q_{i-1}^{-1} q_{i+1} \right) \right) \right) \right) \quad (5-90)$$

- $F_N$  - przekształcenie (filtracja) typu *NonLinQuat*

$$F_N(q_i) = q_i \left( q_i^{-1} \tilde{q}_P (\tilde{q}_P^{-1} \tilde{q}_N)^{1/2} \right)^c = q_i \exp \left( c \log \left( q_i^{-1} \tilde{q}_P \exp \left( 0.5 \log \left( \tilde{q}_P^{-1} \tilde{q}_N \right) \right) \right) \right) \quad (5-91)$$

- $F_A$  - przekształcenie (filtracja) typu *AnisoQuat*

$$F_A(q_i) = q_i \exp \left( \lambda_A g \left( \left\| \log \left( q_i^{-1} q_{i+1} \right) - \log \left( q_{i-1}^{-1} q_i \right) \right\| \right) \log \left( q_i^{-1} \tilde{q}_P \exp \left( 0.5 \log \left( \tilde{q}_P^{-1} \tilde{q}_N \right) \right) \right) \right) \quad (5-92)$$

Oznaczenia pomocnicze:

$$\tilde{q}_P = q_i \exp \left( \tilde{c}_P \log \left( q_i^{-1} q_{i-1} \right) \right) = q_i \exp \left( \lambda g \left( \left\| 2 \log \left( q_i^{-1} q_{i-1} \right) \right\| \right) \log \left( q_i^{-1} q_{i-1} \right) \right) \quad (5-93)$$

$$\tilde{q}_N = q_i \exp \left( \tilde{c}_N \log \left( q_i^{-1} q_{i+1} \right) \right) = q_i \exp \left( \lambda g \left( \left\| 2 \log \left( q_i^{-1} q_{i+1} \right) \right\| \right) \log \left( q_i^{-1} q_{i+1} \right) \right) \quad (5-94)$$

Przydatną własnością do przeprowadzenia dowodów jest pokazanie, że przekształcenie  $F_L$  jest szczególnym przypadkiem przekształcenia  $F_N$ . Otrzymuje się to przez przyjęcie stałej wartości współczynników  $c$  w równaniach (5-93), (5-94):

$$\tilde{c}_P = 1 \Rightarrow \tilde{q}_P = q_{i-1} \quad (5-95)$$

$$\tilde{c}_N = 1 \Rightarrow \tilde{q}_N = q_{i+1} \quad (5-96)$$

Analogicznie można pokazać, że przekształcenie  $F_N$  jest szczególnym przypadkiem przekształcenia  $F_A$  przy założeniu, że anizotropowy współczynnik dyfuzji  $\tilde{c}_A = 1$ .

Dzięki takim przekształceniom wystarczy przeprowadzić dowód własności dla przypadku przekształcenia  $F_A$ , aby taka sama własność dotyczyła przekształceń  $F_L$  oraz  $F_N$ . Jednocześnie wystarczy pokazać, że przekształcenie  $F_L$  nie spełnia własności, z czego wynika, że również przekształcenia  $F_N$  oraz  $F_A$  jej nie spełniają.

### 5.6.1 Liniowość przekształcenia

Do określenia liniowości zaproponowanych przekształceń wykorzystuje się następującą definicję:

#### Definicja 5.1 – liniowość przekształcenia

Przekształcenie  $F$  jest liniowe wtedy i tylko wtedy, gdy dla każdego  $\alpha, \beta \in \mathbb{R}$ ,  $p_i, r_i \in \mathbb{R}^n$  zachodzi:

$$F(\alpha p_i + \beta r_i) = \alpha F(p_i) + \beta F(r_i) \quad (5-97)$$

Zostanie sformułowana następująca własność zaproponowanych przez autora rozprawy algorytmów filtracji trajektorii w przestrzeni orientacji.

### Własność 5.2

Algorytmy filtracji trajektorii *LinQuat*, *NonLinQuat*, *AnisoQuat* realizowane na hipersferze jednostkowej  $S^3$  nie są liniowe w rozumieniu definicji 5.1.

### Dowód

Schematy filtracji nieliniowej oraz anizotropowej są skonstruowane na podstawie schematu filtracji liniowej. Aby udowodnić własność 5.2 wystarczy pokazać, że przekształcenie  $F_L$  nie jest liniowe.

Niech  $a_i, b_i \in H$  będą elementami trajektorii orientacji układu opisane w przestrzeni kwaternionów.

Otrzymuje się następujące składowe twierdzenia 5.1:

$$\alpha F_L(a_i) = \alpha a_i \exp(c \log(a_i^{-1} a_{i-1} \exp(0.5 \log(a_{i-1}^{-1} a_{i+1})))) \quad (5 - 98)$$

$$\beta F_L(b_i) = \beta b_i \exp(c \log(b_i^{-1} b_{i-1} \exp(0.5 \log(b_{i-1}^{-1} b_{i+1})))) \quad (5 - 99)$$

$$\begin{aligned} & F_L(\alpha a_i + \beta b_i) \\ &= (\alpha a_i + \beta b_i) \exp(c \log((\alpha a_i + \beta b_i)^{-1} (\alpha a_{i-1} + \beta b_{i-1}) \exp(0.5 \log((\alpha a_{i-1} + \beta b_{i-1})^{-1} (\alpha a_{i+1} + \beta b_{i+1})))))) \end{aligned} \quad (5 - 100)$$

Twierdzenie musi zachodzić dla wszystkich dopuszczalnych parametrów  $c$ . Bez zmniejszenia ogólności można przyjąć, że twierdzenie musi zachodzić dla  $c = 1$ . Upraszcza to równania do postaci:

$$F_L(\alpha a_i) = \alpha a_{i-1} \exp(0.5 \log(a_{i-1}^{-1} a_{i+1})) \quad (5 - 101)$$

$$F_L(\beta b_i) = \beta b_{i-1} \exp(0.5 \log(b_{i-1}^{-1} b_{i+1})) \quad (5 - 102)$$

$$\begin{aligned} F_L(\alpha a_i + \beta b_i) &= (\alpha a_{i-1} + \beta b_{i-1}) \exp(0.5 \log((\alpha a_{i-1} + \beta b_{i-1})^{-1} (\alpha a_{i+1} + \beta b_{i+1}))) \\ &= \alpha a_{i-1} \exp(0.5 \log((\alpha a_{i-1} + \beta b_{i-1})^{-1} (\alpha a_{i+1} + \beta b_{i+1}))) \\ &\quad + \beta b_{i-1} \exp(0.5 \log((\alpha a_{i-1} + \beta b_{i-1})^{-1} (\alpha a_{i+1} + \beta b_{i+1}))) \end{aligned} \quad (5 - 103)$$

$$\alpha F_L(a_i) + \beta F_L(b_i) = \alpha a_{i-1} \exp(0.5 \log(a_{i-1}^{-1} a_{i+1})) + \beta b_{i-1} \exp(0.5 \log(b_{i-1}^{-1} b_{i+1})) \quad (5 - 104)$$

Porównując wyrażenia według wyrazów stojących przy składowych  $\alpha a_{i-1}$  oraz  $\beta b_{i-1}$  równań (5 - 103), (5 - 104) otrzymuje się następujące zależności:

$$\alpha a_{i-1} \exp(0.5 \log(a_{i-1}^{-1} a_{i+1})) = \alpha a_{i-1} \exp(0.5 \log((\alpha a_{i-1} + \beta b_{i-1})^{-1} (\alpha a_{i+1} + \beta b_{i+1}))) \quad (5 - 105)$$

$$\beta b_{i-1} \exp(0.5 \log(b_{i-1}^{-1} b_{i+1})) = \beta b_{i-1} \exp(0.5 \log((\alpha a_{i-1} + \beta b_{i-1})^{-1} (\alpha a_{i+1} + \beta b_{i+1}))) \quad (5 - 106)$$

Wyrażenia będą sobie równe wtedy i tylko wtedy, gdy składowe funkcji eksponencjalne będą sobie równe:

$$a_{i-1}^{-1} a_{i+1} = (\alpha a_{i-1} + \beta b_{i-1})^{-1} (\alpha a_{i+1} + \beta b_{i+1}) \quad (5 - 107)$$

$$b_{i-1}^{-1} b_{i+1} = (\alpha a_{i-1} + \beta b_{i-1})^{-1} (\alpha a_{i+1} + \beta b_{i+1}) \quad (5 - 108)$$

Prowadzi to do następującego warunku:

$$a_{i-1}^{-1} a_{i+1} = b_{i-1}^{-1} b_{i+1}, \quad (5 - 109)$$

co jest spełnione tylko w szczególnym przypadku. Dla dowolnie wybranych trajektorii  $a_i$ ,  $b_i$  zależność (5 – 109) nie jest prawdziwa. Stąd przekształcenia  $F_L$ ,  $F_N$ ,  $F_A$  nie są przekształceniami liniowymi.

□

Nieliniowość zaproponowanych algorytmów wynika bezpośrednio z realizacji filtracji w przestrzeni kwaternionów jednostkowych – otrzymywany wynik przekształcenia zawsze należy do hipersfery jednostkowej. Dzięki temu zachowana jest stała wartość prędkości kątowej przy wyznaczaniu wyniku przez interpolację pomiędzy położeniami. Gdyby jednak założyć, że wynik filtracji nie musi leżeć na hipersferze jednostkowej (co powoduje, że prędkość kątowa nie jest zachowana), a przekształcenie realizowane jest przez interpolację liniową, to algorytm *LinQuat* byłby przekształceniem liniowym. Algorytmy *NonLinQuat* oraz *AnisoQuat* są nieliniowe niezależnie od przyjętego ograniczenia na podprzestrzeń dopuszczalną wyniku.

Algorytm *LinQuat* jest nieliniowy w rozumieniu definicji 5.1. Jednakże ze względu na stałą wartość użytego w nim współczynnika dyfuzji, jest on czasami nazywany algorytmem liniowym. Ponadto jego konstrukcja oparta jest na uogólnieniu dyskretyzacji liniowego równania różniczkowego. Pamiętać jednak należy, że przyjęte ograniczenie narzucające operacje na sferze jednostkowej sprawia, że algorytm ten nie może być rozpatrywany pod żadnym kątem, jako liniowy.

### 5.6.2 Stacjonarność przekształcenia (ang. *time-invariance*)

Niech operator przesunięcia czasowego będzie zdefiniowany jako:

$$S^k(p_i) = p_{i-k} \quad (5 - 110)$$

#### Definicja 5.2 – stacjonarność przekształcenia

Przekształcenie  $F$  jest stacjonarne wtedy i tylko wtedy, gdy dla dyskretnego sygnału  $p_i$  zachodzi:

$$S^k(F(p_i)) = F(S^k(p_i)) \quad (5 - 111)$$

Zostanie udowodniona następująca własność dotycząca proponowanej rodziny filtrów.

#### Własność 5.3

Algorytmy filtracji trajektorii *LinQuat*, *NonLinQuat*, *AnisoQuat* realizowane przez przekształcenia  $F_L$ ,  $F_N$ ,  $F_A$  są stacjonarne.

#### Dowód

Proponowana własność będzie przeprowadzona na podstawie przekształcenia  $F_A$ . Spełnienie jej dla tego przekształcenia oznacza, że jest ona spełniona dla pozostałych proponowanych algorytmów. Na podstawie równania (5 – 92) otrzymuje się:

$$\begin{aligned}
 S^k(F_A(q_i)) &= S^k \left[ q_i \exp \left( \lambda_A g \left( \left\| \log(q_i^{-1} q_{i+1}) - \log(q_{i-1}^{-1} q_i) \right\| \right) \log(q_i^{-1} \tilde{q}_{S(P)} \exp(0.5 \log(\tilde{q}_{S(P)}^{-1} \tilde{q}_{S(N)}))) \right) \right] \\
 &= q_{i-k} \exp \left( \lambda_A g \left( \left\| \log(q_{i-k}^{-1} q_{(i+1)-k}) - \log(q_{(i-1)-k}^{-1} q_{i-k}) \right\| \right) \log(q_{i-k}^{-1} \tilde{q}_{S(P)} \exp(0.5 \log(\tilde{q}_{S(P)}^{-1} \tilde{q}_{S(N)}))) \right) \quad (5-112)
 \end{aligned}$$

Oznaczenia pomocnicze również są przekształcone do postaci:

$$\tilde{q}_{S(P)} = q_{i-k} \exp(\tilde{c}_P \log(q_{i-k}^{-1} q_{(i-1)-k})) = q_i \exp(g(\|2 \log(q_{i-k}^{-1} q_{(i-1)-k})\|) \log(q_{i-k}^{-1} q_{(i-1)-k})) \quad (5-113)$$

$$\tilde{q}_{S(N)} = q_{i-k} \exp(\tilde{c}_N \log(q_{i-k}^{-1} q_{(i+1)-k})) = q_i \exp(g(\|2 \log(q_{i-k}^{-1} q_{(i+1)-k})\|) \log(q_{i-k}^{-1} q_{(i+1)-k})) \quad (5-114)$$

Po przegrupowaniu indeksów otrzymuje się

$$\begin{aligned}
 S^k(F_A(q_i)) &= q_{(i-k)} \exp \left( \lambda_A g \left( \left\| \log(q_{(i-k)}^{-1} q_{(i-k)+1}) - \log(q_{(i-k)-1}^{-1} q_{(i-k)}) \right\| \right) \log(q_{(i-k)}^{-1} \tilde{q}_{S(P)} \exp(0.5 \log(\tilde{q}_{S(P)}^{-1} \tilde{q}_{S(N)}))) \right) \quad (5-115) \\
 &= F_A(q_{i-k}) = F_A(S^k(q_i))
 \end{aligned}$$

□

Ponieważ algorytmy *LinQuat* oraz *NonLinQuat* są szczególnym przypadkiem bardziej ogólnego algorytmu *AnisoQuat* pokazano w ten sposób, że własność filtracja jest spełniona dla wszystkich proponowanych algorytmów.

### 5.6.3 Niezmienniczość względem transformacji (ang. *coordinate-invariance*)

Niezmienniczość względem transformacji współrzędnych (ang. *coordinate-invariance*) oznacza, że przekształcenie ma takie same własności w całej dopuszczalnej rozważanej przestrzeni. W związku z tym wykonanie dodatkowej rotacji opisanej przez kwaterniony o stałej wartości da taki sam efekt gdy zostanie wykonane dla trajektorii oryginalnej, jak i dla trajektorii przekształconej. Określa to następująca definicja.

#### Definicja

Kwaternionowe przekształcenie  $F$  jest niezmiennicze względem transformacji wtedy i tylko wtedy, gdy dla dowolnych kwaternionów  $a, b \in S^3$  zachodzi  $aF(q_i)b = F(aq_ib)$ .

#### Własność 5.4

Algorytmy filtracji trajektorii *LinQuat*, *NonLinQuat*, *AnisoQuat* realizowane przez przekształcenia  $F_L$ ,  $F_N$ ,  $F_A$  są niezmiennicze względem transformacji.

#### Dowód

Zostanie zaprezentowany dowód zaadaptowany na podstawie dowodu przedstawionego w pracy [66] dotyczącego schematu filtracji opisanej w rozdziale 5.1.1. Autorzy cytowanej pracy udowadniają następującą własność potrzebną w dalszej części dowodu:

$$\begin{aligned} \exp(a^{-1}va) &= \left( \cos\|v\|, \frac{\sin\|v\|}{\|v\|} a^{-1}va \right) = (\cos\|v\|, 0) + \left( 0, \frac{\sin\|v\|}{\|v\|} a^{-1}va \right) \\ &= a^{-1}(\cos\|v\|, 0)a + a^{-1} \left( 0, \frac{\sin\|v\|}{\|v\|} v \right) a^{-1} = a^{-1} \left( \cos\|v\|, \frac{\sin\|v\|}{\|v\|} v \right) a = a^{-1} \exp(v)a \end{aligned} \quad (5-116)$$

gdzie  $v \in R^3$ ,  $a \in S^3$ . Analogicznie można pokazać, że:

$$\log(a^{-1}qa) = a^{-1} \log(q)a, \quad (5-117)$$

dla  $q \in S^3$ . Oba wyprowadzenia opierają się na zależności:

$$\|v\| = \|b^{-1}vb\|, \quad (5-118)$$

Rozważając przekształcenie  $F_A$ , jako najbardziej ogólne, otrzymuje się:

$$F_A(aq_i b) = aq_i b \exp(\tilde{c}_A \log((aq_i b)^{-1} \tilde{q}_P \exp(0.5 \log(\tilde{q}_P^{-1} \tilde{q}_N)))) \quad (5-119)$$

gdzie

$$\tilde{q}_P = aq_i b \exp(\tilde{c}_P \log((aq_i b)^{-1} aq_{i-1} b)) \quad (5-120)$$

$$\tilde{q}_N = aq_i b \exp(\tilde{c}_N \log((aq_i b)^{-1} aq_{i+1} b)) \quad (5-121)$$

Współczynniki dyfuzji nie zmieniają swojej wartości, co można pokazać przy użyciu własności (5-116)(5-118):

$$\begin{aligned} \tilde{c}_P &= g(\|2 \log((aq_i b)^{-1} aq_P b)\|) = g(\|2 \log(b^{-1} q_i^{-1} a^{-1} aq_P b)\|) = g(\|2b^{-1} \log(q_i^{-1} q_P) b\|) \\ &= g(\|2 \log(q_i^{-1} q_P)\|) \end{aligned} \quad (5-122)$$

$$\tilde{c}_N = g(\|2 \log((aq_i b)^{-1} aq_N b)\|) = g(\|2 \log(q_i^{-1} q_N)\|) \quad (5-123)$$

$$\tilde{c}_A = \lambda_A g(\|\log((aq_i b)^{-1} aq_{i+1} b) - \log((aq_{i-1} b)^{-1} aq_i b)\|) = \lambda_A g(\|\log(q_i^{-1} q_{i+1}) - \log(q_{i-1}^{-1} q_i)\|) \quad (5-124)$$

Przekształcając równania (5-120), (5-121) otrzymuje się odpowiednio:

$$\begin{aligned} \tilde{q}_P &= aq_i b \exp(\tilde{c}_P \log(b^{-1} q_i^{-1} a^{-1} aq_{i-1} b)) = aq_i b \exp(\tilde{c}_P b^{-1} \log(q_i^{-1} q_{i-1}) b) \\ &= aq_i b b^{-1} \exp(\tilde{c}_P \log(q_i^{-1} q_{i-1})) b = a\tilde{q}_P b \end{aligned} \quad (5-125)$$

$$\begin{aligned} \tilde{q}_N &= aq_i b \exp(\tilde{c}_N \log(b^{-1} q_i^{-1} a^{-1} aq_{i+1} b)) = aq_i b \exp(\tilde{c}_N b^{-1} \log(q_i^{-1} q_{i+1}) b) \\ &= aq_i b b^{-1} \exp(\tilde{c}_N \log(q_i^{-1} q_{i+1})) b = a\tilde{q}_N b \end{aligned} \quad (5-126)$$

Podstawiając zależności do równania (5-119) otrzymuje się:

$$\begin{aligned} F_A(aq_i b) &= aq_i b \exp(\tilde{c}_A \log(b^{-1} q_i^{-1} a^{-1} a\tilde{q}_P b \exp(0.5 \log((a\tilde{q}_P b)^{-1} a\tilde{q}_N b)))) \\ &= aq_i b \exp(\tilde{c}_A \log(b^{-1} q_i^{-1} \tilde{q}_P b \exp(b^{-1} 0.5 \log(\tilde{q}_P^{-1} \tilde{q}_N) b))) \\ &= aq_i b \exp(b^{-1} \tilde{c}_A \log(q_i^{-1} \tilde{q}_P b b^{-1} \exp(0.5 \log(\tilde{q}_P^{-1} \tilde{q}_N))) b) \\ &= aq_i \exp(\tilde{c}_A \log(q_i^{-1} \tilde{q}_P \exp(b^{-1} 0.5 \log(\tilde{q}_P^{-1} \tilde{q}_N)))) b = aF_A(q_i) b \end{aligned} \quad (5-127)$$

□

## Wniosek

Jeżeli współczynniki dyfuzji  $(\tilde{c}_A, \tilde{c}_P, \tilde{c}_N)$  są ustalone, to przekształcenie  $F_A$  redukuje się odpowiednio do przekształceń  $F_N$  lub  $F_L$ . Stałość współczynników dyfuzji nie zmienia własności przekształcenia (5 – 127). Stąd na podstawie powyższego dowodu otrzymuje się, że przekształcenia  $F_N$  oraz  $F_L$  są niezmiennicze względem transformacji.

W wielu przypadkach dla algorytmów filtracji obrazów pokazywane jest, że posiadają one własność niezmienniczości względem translacji. Należy zauważyć, że przedstawiona własność niezmienniczości względem transformacji jest pojęciem szerszym, gdyż uwzględnia również rotacje wykonywane w przestrzeni kwaternionów. Zmiana współrzędnych trajektorii, zarówno globalnych, jak i lokalnych jest dokonywana przez iloczyn z kwaternionem przekształcającym. Taką transformację nie należy mylić ze skalowaniem wykonywanym dla wielkości skalarnych. Transformacja przez iloczyn kwaternionów realizuje zmianę współrzędnych przez rotację orientacji układu. Nie można natomiast zdefiniować odpowiednika własności niezmienniczości skalowania.

### 5.6.4 Ograniczenie przestrzennego rozrzutu

W przypadku filtracji obrazów jest formułowana następująca zasada maksimum (ang. *maximum principle*) ([79], [107]):

$$(I_m)_{i,j}^t \leq I_{i,j}^{t+1} \leq (I_M)_{i,j}^t, \quad (5 - 128)$$

gdzie wartości ekstremalne z otoczenia piksela  $(i, j)$  są zdefiniowane jako (szczegółowy opis oznaczeń dyskretyzacji procesu zawarty jest w rozdziale 2.5.2):

$$(I_m)_{i,j}^t = \min \left\{ (I, I_N, I_S, I_E, I_W)_{i,j}^t \right\} \quad (5 - 129)$$

$$(I_M)_{i,j}^t = \max \left\{ (I, I_N, I_S, I_E, I_W)_{i,j}^t \right\} \quad (5 - 130)$$

W praktyce własność ta oznacza, że proces filtracji nie wprowadzi żadnych nowych wartości, które byłyby odpowiednio większe lub mniejsze od już istniejących ekstremów. Spełnienie tego warunku jest szczególnie istotne przy analizie przestrzeni skal generowanych przez proces filtracji. Wystąpienie ekstremum funkcji dla danej chwili czasowej filtracji  $t$  oznacza, że we wcześniejszych chwilach istniało ekstremum o niemniejszej wartości. Dzięki temu możliwe jest przeprowadzenie śledzenia zmienności położenia ekstremum na całym obszarze dla kolejnych skal generowanych przez przekształcenie. Ponadto zachowanie zasady maksimum ma podstawowe znaczenie w przypadku określenia stabilności algorytmu – jeżeli algorytm jest stabilny, to nie zostaną wygenerowane wartości większe od maksymalnej wartości występującej w zbiorze początkowym.

W przypadku rozważania trajektorii w przestrzeni rotacji pożyteczne jest rozpatrzenie podobnej pojęciowo, jednakże inaczej zdefiniowanej własności. Interesującą własnością może być ograniczenie przestrzennego rozrzutu przekształcenia w sensie odległościowym zdefiniowane w następujący sposób.

## Definicja

Dany jest zbiór elementów  $q_i$  będących punktami trajektorii  $q$ . Niech  $\chi$  reprezentuje powłokę wypukłą wyznaczoną przed przekształceniem w dziedzinie kwaternionów jednostkowych (w ogólności może to być dowolne skończone otoczenie, np. stożkowe, na sferze jednostkowej zawierające wszystkie punkty trajektorii). Przekształcenie  $F$  działające na elementach trajektorii  $q$  będzie posiadało własność ograniczenia przestrzennego rozrzutu, jeżeli spełniony jest warunek

$$\forall q_i. F(q_i) \in \chi \quad (5 - 131)$$

Warunek (5 – 131) oznacza, że przetworzony element trajektorii  $q_i$  nie może się oddalić od całego zgrupowania wszystkich elementów. Gdyby taka sytuacja zaszła, to byłaby możliwość, że kolejne przekształcenia spowodują, że elementy stają coraz bardziej oddalone od siebie. Oznacza to, że proces nie dąży do stanu ustalonego, przez co trajektoria jest coraz bardziej rozrzucana. Na podstawie tej definicji proponuje się udowodnić następującą własność:

## Własność 5.5

Przekształcenia realizowane przez algorytmy *LinQuat*, *NonLinQuat*, *AnisoQuat* posiadają własność ograniczonego przestrzennego rozrzutu.

## Dowód

Rozważane będzie przekształcenie  $F_A$ , ponieważ jest ono najbardziej uniwersalne z zaproponowanej rodziny algorytmów filtracji. W pojedynczym przekształceniu uwzględniany jest kwaternion  $q_i$ , którego nowe położenie wyznaczone zostaje na podstawie elementów sąsiednich w trajektorii:  $q_{i-1}$ ,  $q_{i+1}$ . Należy rozpatryć następujące przypadki położenia kwaternionów z podzbioru  $\hat{Q} = \{q_{i-1}, q_i, q_{i+1}\}$  względem powłoki wypukłej  $\chi$ :

- elementy  $q_{i-1}$ ,  $q_{i+1}$  zawierają się we wnętrzu  $\chi$ ,
- elementy  $q_{i-1}$ ,  $q_{i+1}$  należą do brzegu  $\chi$ ,
- jeden z kwaternionów  $q_{i-1}$ ,  $q_{i+1}$  należy do brzegu  $\chi$ , a drugi zwarty jest we wnętrzu  $\chi$ .

Przekształcenie  $F_A$  opisane jest następującym wyrażeniem:

$$F_A(q_i) = q_i \exp\left(\lambda_A g\left(\left\|\log(q_i^{-1}q_{i+1}) - \log(q_{i-1}^{-1}q_i)\right\|\right)\log(q_i^{-1}\tilde{q}_P \exp(0.5 \log(\tilde{q}_P^{-1}\tilde{q}_N))\right)\right), \quad (5 - 132)$$

przy oznaczeniach pomocniczych zdefiniowanych w wyrażeniach (5 – 93), (5 – 94).

Zakres zmienności wyrażenia realizującego anizotropowy współczynnik dyfuzji  $c_A$  jest ograniczony ze względu na wartości funkcji  $g$  do przedziału  $[0,1]$ . Ponieważ krok filtracji realizowany jest przez sferyczno-liniową interpolację, to odległość zdefiniowana zależnością (5 – 132) zmienia się liniowo wzdłuż linii

geodezyjnej wyznaczonej przez kwaterniony początkowy  $q_i$  i końcowy. Stąd wystarczy sprawdzić przypadki krańcowe (generujące odpowiednio odległości  $d_0, d_1$ ):

$$c_{A0} = \lambda_A g \left( \left\| \log(q_i^{-1} q_{i+1}) - \log(q_{i-1}^{-1} q_i) \right\| \right) = 0 \quad (5 - 133)$$

$$c_{A1} = \lambda_A g \left( \left\| \log(q_i^{-1} q_{i+1}) - \log(q_{i-1}^{-1} q_i) \right\| \right) = 1 \quad (5 - 134)$$

Dla  $c_A = 0$  wyrażenie znacznie się upraszcza do postaci:

$$F_A(q_i) = q_i \exp(c_{A0} \log(q_i^{-1} \tilde{q}_P \exp(0.5 \log(\tilde{q}_P^{-1} \tilde{q}_N)))) = q_i \quad (5 - 135)$$

Oznacza to, że nie wystąpiła transformacja elementu  $q_i$ . Dla  $c_A = 1$  otrzymuje się:

$$F_A(q_i) = q_i \exp(c_{A1} \log(q_i^{-1} \tilde{q}_P \exp(0.5 \log(\tilde{q}_P^{-1} \tilde{q}_N)))) = \tilde{q}_P \exp(0.5 \log(\tilde{q}_P^{-1} \tilde{q}_N)) \quad (5 - 136)$$

Położenia opisane przez kwaterniony  $\tilde{q}_P, \tilde{q}_N$  są otrzymywane przez liniową interpolację opisaną równaniami (5 - 93), (5 - 94). Współczynniki interpolacji  $\tilde{c}_P, \tilde{c}_N$  są ograniczone do zakresu  $[0,1]$ .

Przyjmując wartości współczynników równe 0 otrzymuje się przypadek poprzedni opisany równaniem (5 - 135). Przyjmując wartości współczynników  $\tilde{c}_P = \tilde{c}_N = 1$  otrzymuje się:

$$F_A(q_i) = q_{i-1} \exp(0.5 \log(q_{i-1}^{-1} q_{i+1})) \quad (5 - 137)$$

ponieważ

$$\tilde{q}_P = q_i \exp(\tilde{c}_P \log(q_i^{-1} q_{i-1})) \quad (5 - 138)$$

$$\tilde{q}_N = q_i \exp(\tilde{c}_N \log(q_i^{-1} q_{i+1})) \quad (5 - 139)$$

W wyniku przekształcenia otrzymuje się więc średnie położenie elementów  $q_{i-1}, q_{i+1}$ . Na tej podstawie widoczne jest, że dla przypadku granicznego rezultat przekształcenia nie zależy od położenia kwaternionu  $q_i$ . W odniesieniu do wcześniej wyszczególnionych przypadków na podstawie własności otoczki wypukłej otrzymuje się:

- Jeżeli elementy  $q_{i-1}$  oraz  $q_{i+1}$  zawarte są we wnętrzu powłoki wypukłej  $\chi$ , to także każde położenie uzyskiwane, jako interpolacja pomiędzy nimi zawarta jest we wnętrzu  $\chi$ . W szczególności graniczna interpolacja opisana wyrażeniem (5 - 137) należy do wnętrza  $\chi$ .
- Jeżeli elementy  $q_{i-1}$  oraz  $q_{i+1}$  należą do brzegu  $\chi$ , to także każde położenie uzyskiwane, jako interpolacja pomiędzy nimi zawarta jest we wnętrzu  $\chi$  lub należy do brzegu  $\chi$ . W granicznym przypadku położenie otrzymane po przekształceniu należy do jednej z tych dwóch dziedzin.
- Jeżeli jeden z elementów  $q_{i-1}$  lub  $q_{i+1}$  należy do brzegu  $\chi$ , a drugi zawarty jest w  $\chi$  otrzymuje się przypadek łączący własności dwóch poprzednich. Ostatecznie jednak interpolacja opisana wyrażeniem (5 - 137) należy do wnętrza  $\chi$ .

Oznacza to, że rezultat przekształcenia  $F_A$  musi być zawarty we wnętrzu lub należeć do brzegu powłoki wypukłej wyznaczonej przed przekształceniem. Ze względu na ograniczony zakres dopuszczalnych



wartości współczynników szybkości dyfuzji  $c_A$ ,  $\tilde{c}_P$ ,  $\tilde{c}_N$  nie ma możliwości, aby położenie kwaternionu wynikowego nie należało do tego podobszaru w przestrzeni.

□

Ponieważ przekształcenia  $F_L$ ,  $F_N$  są szczególnym przypadkiem przekształcenia  $F_A$ , to własność ograniczenia przestrzennego rozrzutu jest dla nich również zachowana. W dowodzie pokazano, że można przyjąć dla zależności (5 – 132) stałe, ekstremalne wartości współczynników dyfuzji procesu, co odpowiednio prowadzi do procesu nieliniowego lub liniowego.

W dowodzie przyjęto założenie, że  $\lambda = \lambda_A = 1$ , co nie zmniejsza ogólności dowodu. Jednakże bardzo istotne jest, aby zarówno ten parametr, jak i współczynniki szybkości dyfuzji  $c_A$ ,  $\tilde{c}_P$ ,  $\tilde{c}_N$  nie przekraczały wartości 1. Jeżeli warunek ten nie jest zachowany, to powyższy dowód nie obowiązuje, a przekształcenie nie posiada własności ograniczenia przestrzennego rozrzutu. W rezultacie filtracja przeprowadzana jest niestabilnie, co prowadzi do wprowadzenia niekontrolowanego zniekształcenia. Przypadek taki zostanie zaprezentowany w praktyce w rozdziale 6.2.1.

Należy zauważyć, że w odróżnieniu od analogicznej zasady sformułowanej dla obrazów, w tym przypadku nie występuje kryterium minimum. Porównywane jest jedynie kryterium maksimum ze względu na przyjęcie warunku opartego na zachowaniu własności geometrycznych grupy kwaternionów tworzących trajektorię.

## 5.7 Podsumowanie propozycji filtrów przetwarzających trajektorie w przestrzeni $S^3$

W rozdziale 5 przedstawiono metody realizujące filtrację trajektorii orientacji układów. Tematyka przetwarzania danych opisujących orientacje (oraz rotacje) nie jest jeszcze dobrze zbadana w porównaniu do przetwarzania wartości skalarnych. Główny nacisk położony został na metody uwzględniające trajektorie opisane w przestrzeni kwaternionów. Można wyróżnić kilka podstawowych propozycji, które zostały wcześniej sformułowane w ramach metod filtracji trajektorii orientacji. Metody oparte na liniowym przetwarzaniu szeregu czasowego opierają się na rzutowaniu przestrzeni kwaternionów do płaskiej przestrzeni stycznej, w której wykonywana jest filtracja przy użyciu standardowych metod przetwarzania [28], [66]. Praktyczna realizacja filtracji w tej grupie metod jest więc wykonywana w przestrzeni stycznej  $R^3$  do hipersfery jednostkowej. W innych propozycjach realizowany jest schemat optymalizacyjny przy określonym kryterium energetycznego dla trajektorii kwaternionów. Przykłady tego typu realizacji filtracji można znaleźć w pracach [55], [63], [101]. Różne metody optymalizacji są stosowane do lokalnej minimalizacji energii. Metoda hybrydowa, która we wstępnej fazie używa algorytmu genetycznego przedstawiona jest w pracy [43].

W przedstawionym rozdziale zostało zaproponowane nowe podejście do realizacji filtracji trajektorii orientacji układu reprezentowanych przez kwaterniony jednostkowe. Otrzymane rezultaty można podsumować w następujących punktach:

- zaproponowano algorytm filtracji trajektorii kwaternionów oparty na realizacji liniowego równania dyfuzji (algorytm *LinQuat*) – algorytm jest nazywany liniowym ze względu na użyty stały współczynnik dyfuzji; w ogólności nie jest to przekształcenie liniowe,
- zaproponowany różne rodzaje warunków brzegowych, jakie mogą być uwzględnione przy realizacji filtracji przez poszukiwanie rozwiązania zagadnienia początkowego dyfuzji,
- pokazano powiązanie pomiędzy wcześniej cytowanymi metodami, a algorytmem *LinQuat* zrealizowanym na podstawie równań różniczkowych cząstkowych,
- wyprowadzenie algorytmu na podstawie równań różniczkowych pozwoliło na zaproponowanie uogólnienia nieliniowego procesu dyfuzji (modelu *Perona-Malik* [78], [79] stosowanego w filtracji obrazów) do algorytmu filtracji trajektorii kwaternionowych (algorytm *NonLinQuat*),
- nieliniowy algorytm dla kwaternionów został rozwinięty do postaci anizotropowej, w której dyfuzja zależy nie tylko od położenia w danej chwili czasowej, ale także od lokalnej krzywizny trajektorii – dzięki zaproponowanemu podejściu *AnisoQuat* możliwa jest filtracja zakłóceń występujących w trajektorii przy jednoczesnym zachowaniu kształtu i charakteru przetwarzanej trajektorii,
- zaprezentowano geometryczną interpretację algorytmów pokazującą w sposób obrazowy poszczególne kroki filtracji,
- przeprowadzono analizę wraz z dowodami podstawowych własności zaproponowanych algorytmów pod względem liniowości, stacjonarności, niezmienniczości względem transformacji oraz przestrzennej zbieżności.

Realizacja algorytmów na podstawie rozwiązywania równań różniczkowych możliwa była dzięki wnikliwej analizie podobnych metod stosowanych do filtracji obrazów (rozdział 2). Nie jest jednak możliwe bezpośrednie zastosowanie metod używanych w przestrzeniach  $R^n$  do przestrzeni kwaternionów jednostkowych ze względu na nieliniowy charakter dziedziny  $S^3$ . Proponowane metody pozwalają uwzględnić ograniczenia przestrzeni kwaternionów jednocześnie zachowując podobne własności procesu dyfuzji.

Rezultaty osiągnięte w tym rozdziale można rozpatrywać jako powiązanie i uogólnienie dwóch (do tej pory niezależnych) nurtów związanych z filtracją danych. Z jednej strony jest to uogólnienie (ale też dalsze rozwinięcie) metod filtracji opartych na równaniach różniczkowych o pochodnych cząstkowych używanych do obrazów dwuwymiarowych [78]. Z drugiej strony najprostszy z zaproponowanych algorytmów *LinQuat* jest w granicy realizacją optymalizacji przy określonym kryterium energetycznym dla trajektorii ([55], [63]). Jednocześnie podstawowa konstrukcja algorytmu jest podobna do metod opartych na liniowym przetwarzaniu szeregu czasowego ([28], [66]). Jednak dzięki wyprowadzeniu schematu algorytmu opartego na podstawie równań różniczkowych uzyskuje się większą uniwersalność metody, co pozwala na dalsze jej uogólnienie do postaci nieliniowej i anizotropowej. Eksperymenty numeryczne porównujące działanie algorytmów zostaną przedstawione w rozdziale 6.

## Rozdział 6

### Wyniki badań eksperymentalnych

W rozdziale 5 przedstawiono propozycje metod filtracji trajektorii w przestrzeniach rotacji parametryzowanych przez kwaterniony. Zaprezentowane metody były punktem wyjścia do konstrukcji algorytmów. Praktyczna realizacja algorytmów pozwoliła na uzyskanie eksperymentalnego potwierdzenia wcześniej przedstawionych własności metody filtracji. W kolejnych punktach rozdziału zostanie przedstawiona metoda badawcza, trajektorie testowe (proste i złożone) oraz wyniki przeprowadzonych eksperymentów. Otrzymane rezultaty pozwolą przedstawić zalety proponowanej anizotropowej nieliniowej filtracji trajektorii w porównaniu do algorytmów filtracji liniowej prezentowanych w cytowanych pracach [55], [63], [66].

#### 6.1 Stosowana procedura badawcza

W ramach eksperymentów praktycznych porównywana będzie skuteczność algorytmów implementujących trzy metody filtracji. W dalszej części pracy będą stosowane następujące oznaczenia w prezentacji wyników:

- LQ – algorytm *LinQuat* (rozdział 5.2)
- NLQ – algorytm *NonLinQuat* (rozdział 5.3)
- AQ – algorytm *AnisoQuat* (rozdział 5.4)

W rozdziale 5.2.4 pokazano, że algorytm *LinQuat* jest zbieżny z algorytmami filtracji wykorzystującymi liniowe filtry stacjonarne LTI [66]. Wyniki eksperymentów dla tego algorytmu są więc zbieżne z wynikami, które otrzymuje się za pomocą cytowanych metod. Dzięki temu te same badania pozwolą na analizę porównawczą wyników pomiędzy nową metodą filtracji anizotropowej i podejściem zastosowanych w cytowanych pracach.

Ogólna procedura badawcza stosowana w prezentowanej pracy jest analogiczna do procedur stosowanych przy badaniu własności sygnałów cyfrowych (w szczególności przy badaniu obrazów, jak to zostało zaprezentowane w rozdziale 3). Eksperymenty będą się składać z następujących kroków:

- Wybór sekwencji testowej (syntetycznej – rozdział 7.2, złożonej – rozdział 7.3, 7.4).
- Wprowadzenie sygnału zakłócającego o znanym charakterze.

- Filtracja przy użyciu prezentowanych metod filtracji trajektorii realizowana dla określonej liczby iteracji.
- Wyznaczenie miary jakości filtracji w odniesieniu do sekwencji oryginalnej dla każdej przeprowadzonej iteracji.
- Wizualizacja graficzna rezultatów pośrednich i końcowych.
- Ewentualna powtórka całej procedury dla zmienionych parametrów użytych w metodzie lub parametrów środowiska eksperymentu (zmiana poziomu zaszumienia, zmiana sekwencji testowej, zmiana warunków uwzględniania dodatkowych informacji, itp.).

Ponieważ w eksperymentach wprowadza się losowy czynnik zakłócający, cała procedura zostanie powtórzona określoną liczbę razy, a otrzymane wyniki określające jakość filtracji będą uśrednione, aby został zmniejszony błąd wyznaczania współczynnika jakości filtracji. W praktyce wyznaczenie wartości dla pojedynczej iteracji przy określonych parametrach wykonywano od 3 do 19 razy. Mniejszą liczbę powtórzeń stosowano dla długich sekwencji, które wymagały bardzo długiego czasu obliczeń. Nie zmniejsza to jednak ogólności otrzymanych obliczeń, gdyż w trakcie przeprowadzania eksperymentów zauważono, że już po 3 losowych powtórzeniach sekwencji testowej, otrzymywano wyniki z dokładnością nie gorszą niż 95 % (w porównaniu do wyników otrzymanych dla 19 powtórzeń). Wyniki numeryczne reprezentują dane uśrednione z serii eksperymentów przeprowadzanych w takich samych warunkach. Oznacza to, że do trajektorii za każdym razem był dodawany sygnał zakłócający o wartościach losowych, lecz zawsze takich samych dla danej trajektorii i konkretnego powtórzenia filtracji.

Prezentacja wyników w postaci graficznej wizualizacji trajektorii dotyczyć natomiast będzie pojedynczego, losowo wybranego rezultatu przeprowadzonej filtracji.

### **6.1.1 Zastosowane trajektorie testowe**

Należy podkreślić, że w przeciwieństwie do dziedziny filtracji obrazów dwuwymiarowych, nie istnieją ogólnie przyjęte wzorce sekwencji testowych używanych do badania algorytmów filtracji w przestrzeni trajektorii. Jednym z powodów takiej sytuacji jest to, że pierwsze tego typu algorytmy powstały stosunkowo niedawno, a liczba publikacji rozważających problem filtracji w przestrzeni rotacji jest nadal niewielka. Zazwyczaj każdy autor stosuje apriorycznie wybrane sekwencje testowe, często bez dokładniejszego opisanie sposobu generacji takich sekwencji. W prezentowanej pracy zostaną zaproponowane dwie grupy sekwencji testowych z dokładnym opisem sposobu ich generacji oraz argumentacją, dlaczego takie podejście jest stosowane. Ocena wydajności algorytmów jest dokonywana w sposób porównawczy, co pozwala uzyskać bardziej obiektywne wyniki niż odwoływanie się do rezultatów otrzymanych w innych pracach, w których metoda testowa nie jest ustandaryzowana.

### **Syntetyczne trajektorie testowe**

Pierwsza grupa składa się z trajektorii w przestrzeni orientacji wygenerowanych syntetycznie. Oznacza to, że tworzenie tych przebiegów nie było związane z zapisywaniem ruchów wykonywanych w

przestrzeni rzeczywistej. Trajektoria jest przygotowywana przez odcinkowo ciągłą sferyczno-liniową interpolację pomiędzy ustalonymi punktami węzłowymi w przestrzeni rotacji. Nie nakłada się jednak żadnych dodatkowych ograniczeń na otrzymaną trajektorię (np. związanych z różniczkowalnością krzywej, fizycznej możliwości realizacji trajektorii itp.). Celem tego eksperymentu jest sprawdzenie własności prezentowanych metod w przypadkach sygnału o określonych własnościach. Syntetyczne wygenerowanie trajektorii pozwala na uzyskanie pożądanego przebiegu, którego własności są z góry określone. Podobne podejście stosuje się przy badaniu algorytmów przetwarzania obrazów (rozdział 3).

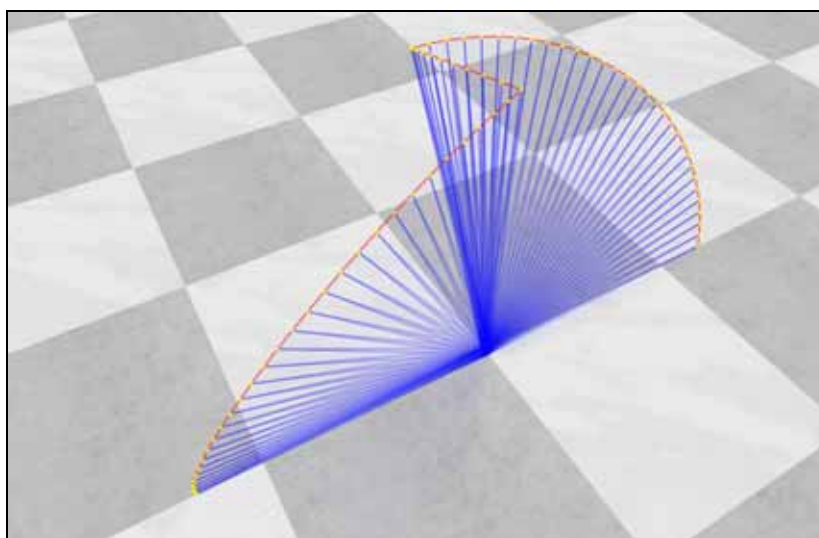
W prezentowanej pracy wyróżnia się 3 główne rodzaje syntetycznych trajektorii testowych. Jeżeli nie zaznaczono inaczej, to w każdym z przypadków przyjmuje się, że liczba punktów interpolowanych pomiędzy punktami węzłowymi wynosi  $n = 30$  (włączając początkowy punkt węzłowy).

#### *Trajektoria testowa nr 1: QuatEdge*

Trajektoria ma na celu modelować odpowiednik krawędzi rozważanych dla algorytmów filtracji obrazów. Jednym z celów opracowania anizotropowych algorytmów filtracji trajektorii jest zmniejszenie wpływu deformacji trajektorii z punktu widzenia jej kształtu. Wizualnie deformacja kształtu jest związana z wprowadzeniem znacznej zmiany w kierunku wykonywanego ruchu. Przyjmuje się ogólne założenie, że odpowiednikiem krawędzi dla trajektorii w przestrzeni rotacji, nazywać się będzie taki przebieg, w którym występuje nagle zmiana położenia lub kształtu trajektorii.

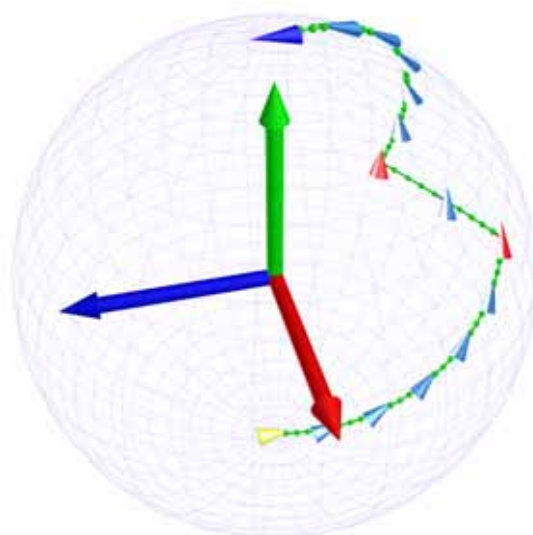
Pierwsza trajektoria testowa reprezentuje ciągły ruch, w którym w dwóch punktach występuje nagle zmiana kierunku. Podejście takie ma symulować odpowiednik krawędzi dla trajektorii w przestrzeni rotacji (*QuatEdge*). Realizowane jest to przez generację trajektorii przy użyciu następujących punktów węzłowych:

- $(0, 0, 0)$  – punkt początkowy,  $n$  elementów
- $(-90, 45, 0)$  –  $n/3$  elementów
- $(-90, 0, 30)$  –  $n$  elementów
- $(-180, 0, 0)$  – punkt końcowy



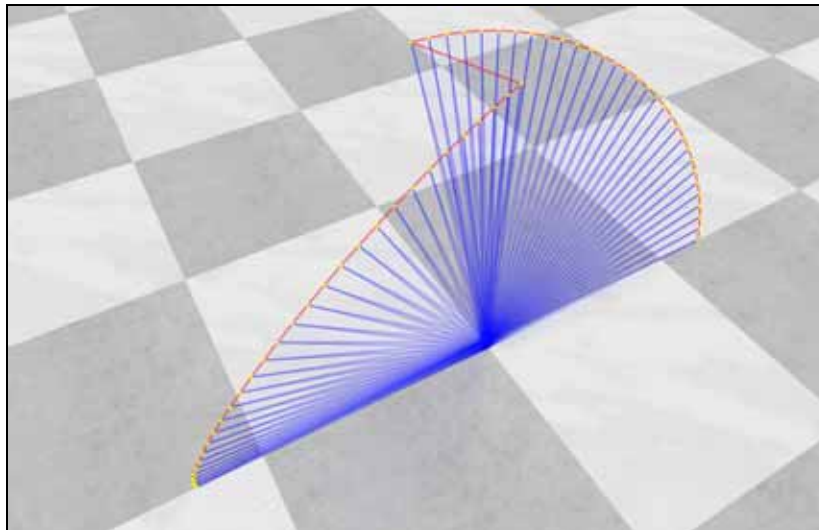
Rysunek 6-1: Trójwymiarowa wizualizacja trajektorii testowej nr 1 (*QuatEdge*).

Na rysunku 6-1 pokazana jest trójwymiarowa wizualizacja trajektorii testowej numer 1. Widoczne są dwa punkty, w których występuje zmiana charakteru trajektorii (zmiana kierunku ruchu). Aby lepiej uwypuklić właściwości tego sygnału, na rysunku 6-2 pokazany jest on przy użyciu sfery rotacji. Kolorem niebieskim zaznaczono punkt początkowy. Kolorem czerwonym zaznaczone są punkty węzłowe (z góry określone). Kolor żółty oznacza punkt końcowy trajektorii. Wartości interpolowane zaznaczone są zielonymi punktami i połączone odcinkami. Celowo zrezygnowano z zaznaczania wszystkich punktów przy użyciu stożka pokazującego trzecią składową rotacji. Wystarczające jest pokazanie w ten sposób tylko kilku wybranych punktów w całej trajektorii. W rozpatrywanym przykładzie co 5 punktów interpolowanych rysowany jest stożek reprezentujący pełną rotację (kolor jasnoniebieski). W pozostałych przypadkach wartość rotacji zaznaczona jest punktem zielonym. Sfera została obrócona o około -30 stopni wokół osi Y w porównaniu do poprzednio prezentowanej orientacji w celu czytelniejszej wizualizacji prezentowanej trajektorii.



**Rysunek 6-2: Wizualizacja trajektorii testowej nr 1 (*QuatEdge*) przy użyciu sfery rotacji.**

Użycie zaprezentowanej trajektorii testowej ma na celu sprawdzenie działania algorytmów w przypadku sygnału, w którym występuje istotna zmiana kształtu (przez zmianę kierunku) trajektorii. Prosta forma trajektorii (tylko 2 punkty węzłowe) pozwoli na pokazanie własności i różnic otrzymywanych przy przetwarzaniu sygnału przez różnego rodzaju algorytmy. Podobny ruch w przestrzeni jest często realizowany w różnych przypadkach praktycznych. Rzadziej występuje przypadek, w którym trajektoria opisuje ruch o znacznej zmianie zarówno kierunku, jak i chwilowej prędkości kątowej. Trajektorja testowa realizująca taką sytuację pokazana jest na rysunku 6-3. Jest ona modyfikacją poprzedniego przypadku, która będzie pomocna w szczególności do testowania algorytmu filtracji nieliniowej (*NonLinQuat*). W trajektorii tej nie występują dodatkowej położenia interpolowane pomiędzy drugim i trzecim punktem węzłowym. Pozostałe własności trajektorii pozostają bez zmian.



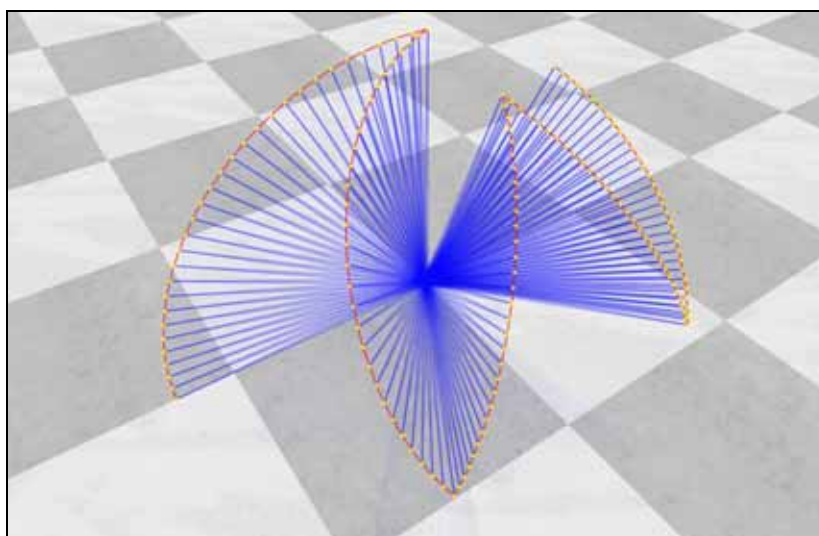
Rysunek 6-3: Trójwymiarowa wizualizacja zmodyfikowanej trajektorii testowej nr 1a.

*Trajektoria testowa nr 2: QuatJumps*

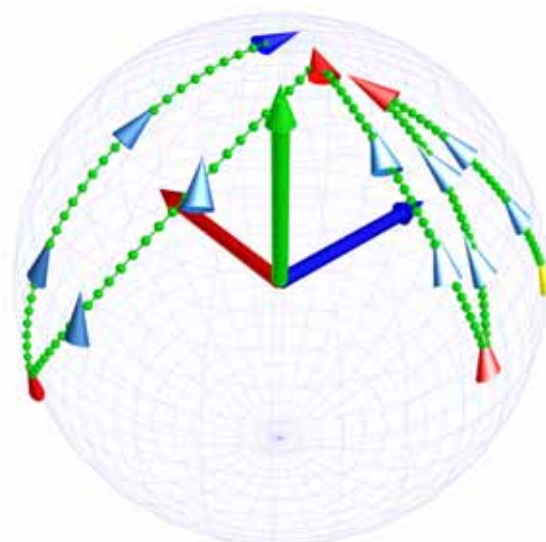
Druga trajektoria testowa pokazana została na rysunkach 6-4, 6-5 (w tym przypadku stożek reprezentujący rotację jest rysowany co 10 elementów, zastosowano obrót sfery rotacji o 180 stopni).

Trajektoria jest opisana przy użyciu następujących punktów węzłowych:

- $(0, 0, 0)$  – punkt początkowy,  $n$  elementów
- $(-90, -10, 0)$  –  $n$  elementów
- $(0, 60, 10)$  –  $n$  elementów
- $(-70, 90, 0)$  –  $n$  elementów
- $(-10, 110, 10)$  –  $n$  elementów
- $(-60, 120, 10)$  – punkt końcowy



Rysunek 6-4: Trójwymiarowa wizualizacja trajektorii testowej nr 2 (*QuatJumps*).



**Rysunek 6-5: Wizualizacja trajektorii testowej nr 2 (*QuatJumps*) przy użyciu sfery rotacji.**

Ruch wykonywany wzdłuż prezentowanej trajektorii testowej zmienia swój kierunek czterokrotnie. Prędkość kątowna w każdym ze stałych odcinków jest w przybliżeniu jednakowa. Trajektoria ta pozwoli przetestować działanie algorytmów dla bardziej skomplikowanego ruchu, w którym kierunek wielokrotnie się zmienia. Przy ocenie wydajności sprawdzane będzie przede wszystkim możliwość zachowania charakteru sygnału z punktu widzenia stałej prędkości kątowej przy jednoczesnej dużej zmienności kierunków. Ponieważ przebieg sygnału w przestrzeni rotacji charakteryzują szybkozmienne regularne skoki, trajektoria nazywana będzie *QuatJumps*.

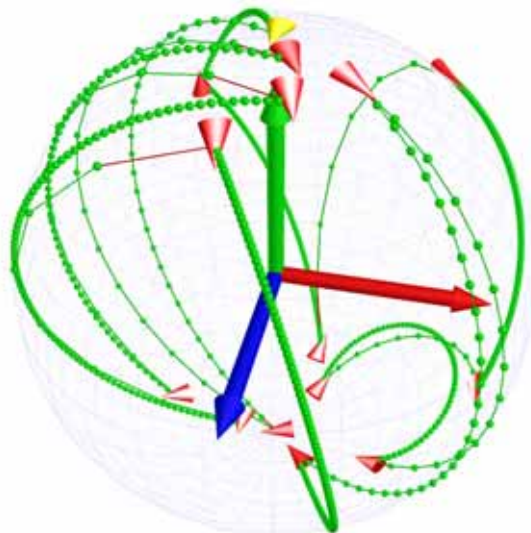
#### *Trajektoria testowa nr 3: QuatMulti*

Ostatnia syntetyczna trajektoria testowa jest najbardziej skomplikowana z punktu widzenia liczby punktów węzłowych oraz zmian występujących w sygnale. Trajektoria pokazana na rysunku 6-6 tworzona jest przez odcinkowo ciągłą sferyczno-liniową interpolację zdefiniowaną przez następujące punkty węzłowe:

- $(0, 0, 0)$  – punkt początkowy,  $n$  elementów
- $(0, 70, 180)$  –  $n/2$  elementów
- $(10, 30, 0)$  –  $2*n$  elementów
- $(0, 50, 150)$  –  $n/3$  elementów
- $(20, 20, 0)$  –  $3*n$  elementów
- $(0, 0, 170)$  –  $n/4$  elementów
- $(30, 10, 10)$  –  $4*n$  elementów
- $(170, 45, 0)$  –  $n$  elementów
- $(20, 60, 0)$  –  $n/2$  elementów
- $(160, 60, 10)$  –  $2*n$  elementów
- $(-160, -30, 0)$  –  $n/3$  elementów



- $(0, 0, -120)$  –  $3*n$  elementów
- $(-30, 45, -60)$  –  $n/4$  elementów
- $(-150, -15, 0)$  –  $4*n$  elementów
- $(-60, 30, 10)$  –  $n$  elementów
- $(0, 0, 0)$  – punkt końcowy



Rysunek 6-6: Wizualizacja trajektorii testowej nr 3 (*QuatMulti*) przy użyciu sfery rotacji.

Stopień skomplikowania tak zdefiniowanej trajektorii jest zbyt wysoki, aby możliwe było przedstawienie jej przy użyciu wizualizacji ruchu w przestrzeni trójwymiarowej. Stąd pokazano jedynie postać trajektorii przy użyciu sfery rotacji. Z tego względu nie zaznaczono również pozycji pośrednich pomiędzy punktami węzłowymi przy użyciu stożka rotacji. Sfera rotacji została obrócona o około 30 stopni względem osi Y.

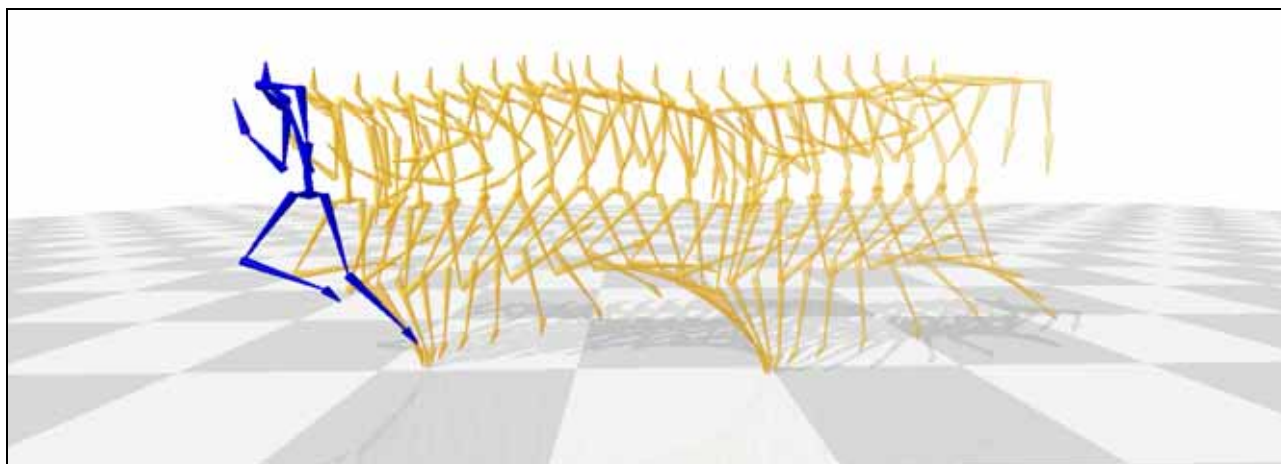
Tak określony sygnał charakteryzuje się zarówno dużymi zmianami kierunku, jak i prędkości kątowej (należy zwrócić uwagę na liczbę elementów interpolowanych wzdłuż odcinków sfery). Pozwoli to na sprawdzenie własności algorytmów z punktu widzenia możliwości zachowania charakteru skomplikowanego, nieregularnie zmiennego sygnału. Ze względu na dużą zmienność sygnału trajektoria testowa nr 3 nazywana będzie *QuatMulti*.

### 6.1.2 Złożone sekwencje testowe

Syntetycznie przygotowane trajektorie pozwalają na szczegółowe sprawdzenie własności badanych algorytmów. Jednak w praktyce algorytmy te będą najczęściej stosowane do przetwarzania złożonych sekwencji ruchów, których trajektorie zostały zarejestrowane w przestrzeni rzeczywistej. Idealnym sposobem na sprawdzenie działania algorytmów w takim przypadku są sekwencje ruchów wykonywanych przez człowieka, zarejestrowane przy użyciu techniki *motion capture*. W rozdziale 4.5.2 na przykładzie szkieletu ludzkiego przedstawiono model złożonego łańcucha kinematycznego. Przy użyciu tego modelu

zostaną przetestowane algorytmy filtracji rotacji zastosowane do trajektorii otrzymanych w wyniku rejestracji ruchu żywych postaci. Wszystkie prezentowane sekwencje ruchów postaci pochodzą z bibliotek *LifeForms 3.0*, *LifeForms 4.0*, *MeGa MoCap 2.0* firmy *Credo Interactive*.

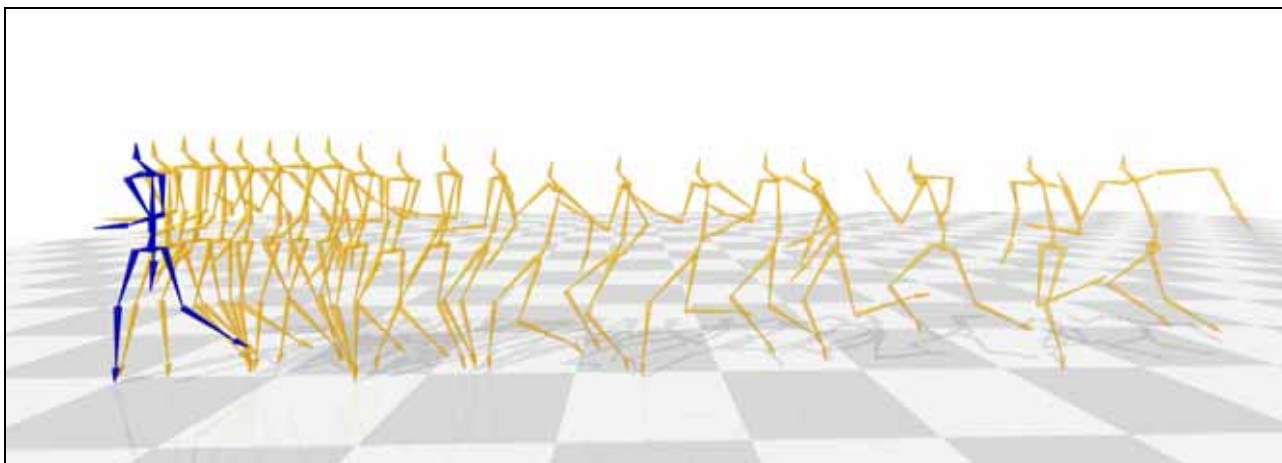
Pierwsza sekwencja testowa reprezentuje prosty ruch postaci typu bieg przedstawiony na rysunku 6-7 (oryginalna nazwa sekwencji używana w bibliotece *LifeForms*, to *Run loopable*). Wykonywany ruch składa się z dwóch kroków, które są opisane przy pomocy modelu złożonego z hierarchicznie powiązanych trajektorii w przestrzeni rotacji. Dla każdej klatki definiuje się dodatkowo globalne przemieszczenie (translacje) postaci względem początku układu współrzędnych sceny. Ze względu na specyfikę omawianych algorytmów, globalne przemieszczenie postaci nie będzie przetwarzane ani brane pod uwagę w ocenie skuteczności przeprowadzonej filtracji.



**Rysunek 6-7: Wizualizacja testowej sekwencji ruchu nr 1 – nazwa oryginalna *Run loopable* (długość 17 klatek).**

Drugą sekwencją testową reprezentuje bardziej złożony ruch nazywany w bibliotece *Sprint to walk*. Ruch pokazany na rysunku 6-8 przedstawia postać, która początkowo biegnie, a następnie przyhamowuje, aby zmienić charakter ruchu na chód. Sekwencja ta jest o wiele bardziej złożona od poprzedniej, gdyż występują zarówno zmiana położenia, jak i zmiana prędkości kątowej dla kolejnych chwil czasowych. Początkowo prędkość kątowa jest znaczna, gdyż cała postać porusza się szybko. Na rysunku widoczne jest, że końcówka ruchu jest znacznie spowolniona, więc również prędkości kątowe występujące dla części szkieletu są mniejsze. Zastosowanie tego typu trajektorii pozwoli na sprawdzenie przebiegu procesu filtracji dla trajektorii o zróżnicowanej charakterystyce.

Należy zwrócić uwagę, że w przypadku obu trajektorii duży stopień informacji o rodzaju wykonywanego ruchu jest zawarty w kształcie trajektorii. Przykładowo ruch kończyn dolnych może być opisany przez czasową odległość punktów, dla których następuje zmiana kierunku ruchu (co jest realizacją ruchu wahadłowego). W związku z tym można będzie sprawdzić, czy testowane algorytmy posiadają zdolność do zachowywania kształtu przetwarzanej trajektorii. Teoretycznie założenie tego typu dotyczy przede wszystkim filtracji typu *AnisoQuat*. Przeprowadzone eksperymenty pozwolą na praktyczne potwierdzenie wcześniej sformułowanych teoretycznie własności algorytmów.



Rysunek 6-8: Wizualizacja testowej sekwencji ruchu nr 2 – nazwa oryginalna *Sprint to walk* (długość 74 klatki).

### 6.1.3 Metody oceny wyników przetwarzania

Aby wyniki filtracji mogły zostać właściwie ocenione stosowane będą następujące metody wizualizacji i porównywania:

- wizualizacja kolejnych położeń wybranego członu (lub pełnego) łańcucha kinematycznego w przestrzeni trójwymiarowej na podstawie modelu przedstawionego w rozdziale 4.4.1,
- wizualizacja rotacji realizowanej na trójwymiarowej sferze jednostkowej przy dodatkowym uwzględnieniu lokalnego stopnia swobody [45] opisana w rozdziale 4.1.4,
- wykres zależności kwaternionowej miary jakości sygnału QSNR w odniesieniu do sekwencji wzorcowej opisana w rozdziale 4.5.3,
- wyznaczenie maksymalnej uzyskanej miary jakości QSNR po wykonaniu określonej liczby iteracji ruchu w zależności od zmiany parametrów.

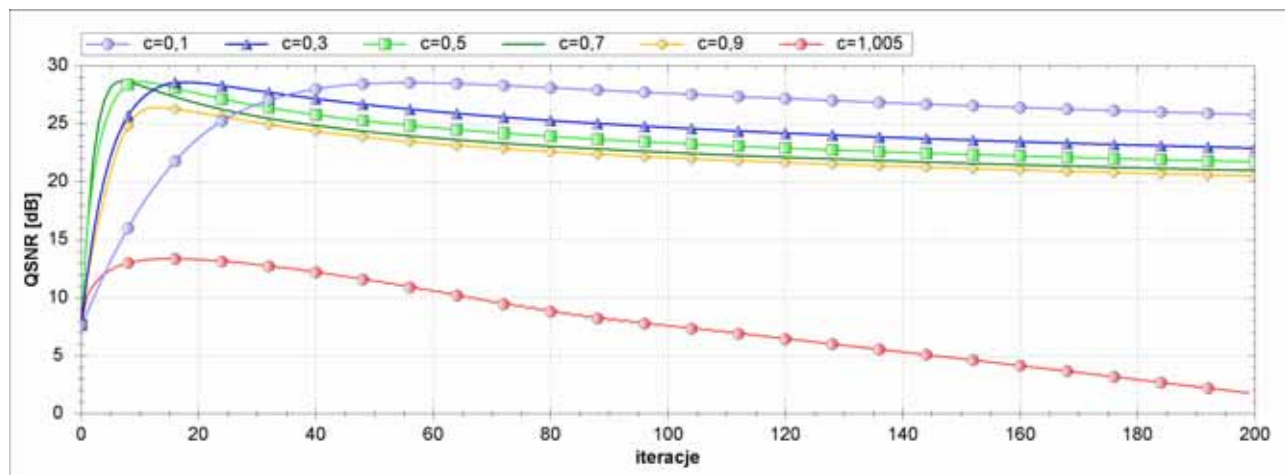
Wykres miary QSNR będzie najbardziej skuteczną metodą analizy charakterystyki algorytmu i przebiegu procesu filtracji dla poszczególnych iteracji. Zamieszczenie wykresów na jednym rysunku pozwoli na podkreślenie podobieństw i różnic pomiędzy realizacjami filtracji przy różnych parametrach lub procesów otrzymanych przez różne algorytmy. Na podstawie przeprowadzonych wykresów zostaną sformułowane wnioski jakościowe i ilościowe dotyczące proponowanych algorytmów.

## 6.2 Skuteczność algorytmu *LinQuat* dla syntetycznych trajektorii testowych

Algorytm filtracji *LQ* jest najprostszym z opisywanych algorytmów. Jak to zostało pokazane w rozdziale 5.2 jest on odpowiednikiem metod zaproponowanych w pracach [63], [66]. Algorytm różni się od podejścia umieszczonego w cytowanych pracach z punktu widzenia źródła schematu postępowania - oparty jest na dyskretnym schemacie różnicowym dla równania dyfuzji zamiast na optymalizacji energetycznej. Natomiast ostateczny kształt przekształcenia oraz sposób działania są zbieżne z algorytmami cytowanymi. Stąd będzie on służył, jako podejście referencyjne przy porównaniu skuteczności filtracji różnymi metodami.

### 6.2.1 Skuteczność filtracji LQ w zależności od współczynnika dyfuzji $c$

Według omówionej w rozdziale 6.1 procedury badawczej dokonano eksperymentów, w którym filtrowano wybrane sekwencje testowe przy różnych wartościach parametru  $c$ . Na rysunku 6-9 pokazany jest wykres zmiany współczynnika jakości sygnału do szumu QSNR w funkcji iteracji procesu dla sekwencji testowej nr 1. Zastosowany został warunek brzegowy typu D. Do sygnału testowego została dodana jest składowa losowa o rozkładzie normalnym i odchyleniu standardowym  $\sigma=0.05$ .



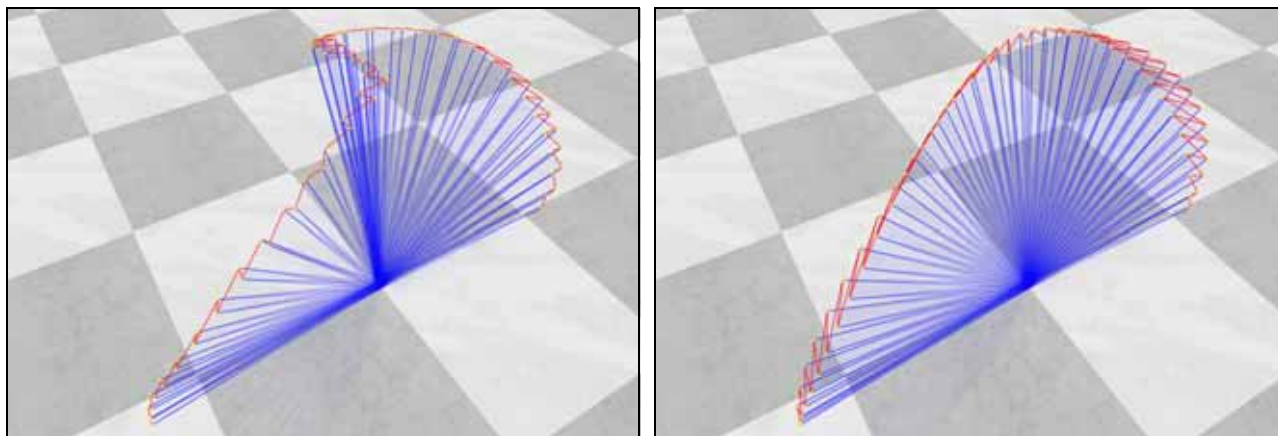
Rysunek 6-9: Miara jakości QSNR dla rezultatów filtracji LQ w zależności od parametru  $c$  – syntetyczna trajektoria testowa nr 1.

Skuteczność filtracji zmienia się w różny sposób w zależności od przyjętego współczynnika dyfuzji. Początkowo niezależnie od przyjętego współczynnika następuje względnie szybki przyrost jakości sygnału. Po osiągnięciu maksimum jakość sygnału w sposób stały obniża się dążąc do jednakowej wartości niezależnie od przyjętego współczynnika dyfuzji  $c$ . Jest to związane ze sposobem działania filtru, które zmierza do ujednoczenia wszystkich wartości występujących w sygnale. W przypadku kwaternionów filtracja ta zmierza do momentu, w którym każdy element trajektorii znajduje się w tym samym miejscu przestrzeni rotacji.

Należy zauważyć, że dla dużych wartości współczynnika  $c$  współczynnik jakości filtracji pogasza się. Dla  $c=0,9$  szybkość poprawy jakości sygnału jest mniejsza od większości pozostałych przypadków. Ponadto maksymalna osiągnięta wartość współczynnika QSNR również jest zmniejszona. Dla wartości bardzo małych następuje natomiast powolna poprawa jakości sygnału. Jednakże również wolniej w takim przypadku następuje pogorszenie się sygnału po osiągnięciu maksymalnej wartości współczynnika QSNR.

W rozdziale 5 przy przekształceniu równania dyskretyzacji procesu został postawiony warunek  $c \leq 1$ . Na rysunku 6-9 umieszczono dodatkowo wykres reprezentujący zachowanie się procesu w sytuacji nieznacznego (o 0,5 %) warunku na parametr dyfuzji. Wykres wskazuje na znaczną niestabilność takiego procesu. Początkowo niewielka poprawa jakości sygnału szybko się kończy i zaczyna się nieregularne, ale stało jego pogorszenie. Stąd wniosek, że dla zachowania stabilności procesu, ze względu na możliwość kumulowania się błędów numerycznych, parametr  $c$  powinien spełniać warunek  $c < 1$ . Na rysunku 6-10 pokazano wynik filtracji sygnału testowego nr 1 przy użyciu współczynnika dyfuzji  $c=1,005$ . Już po 10

iteracjach widoczne jest efekt niestabilności objawiający się przez oscylacje położenia punktów. Niestabilność procesu szybko rośnie z czasem, co jest dobrze widoczne po 200 iteracjach.

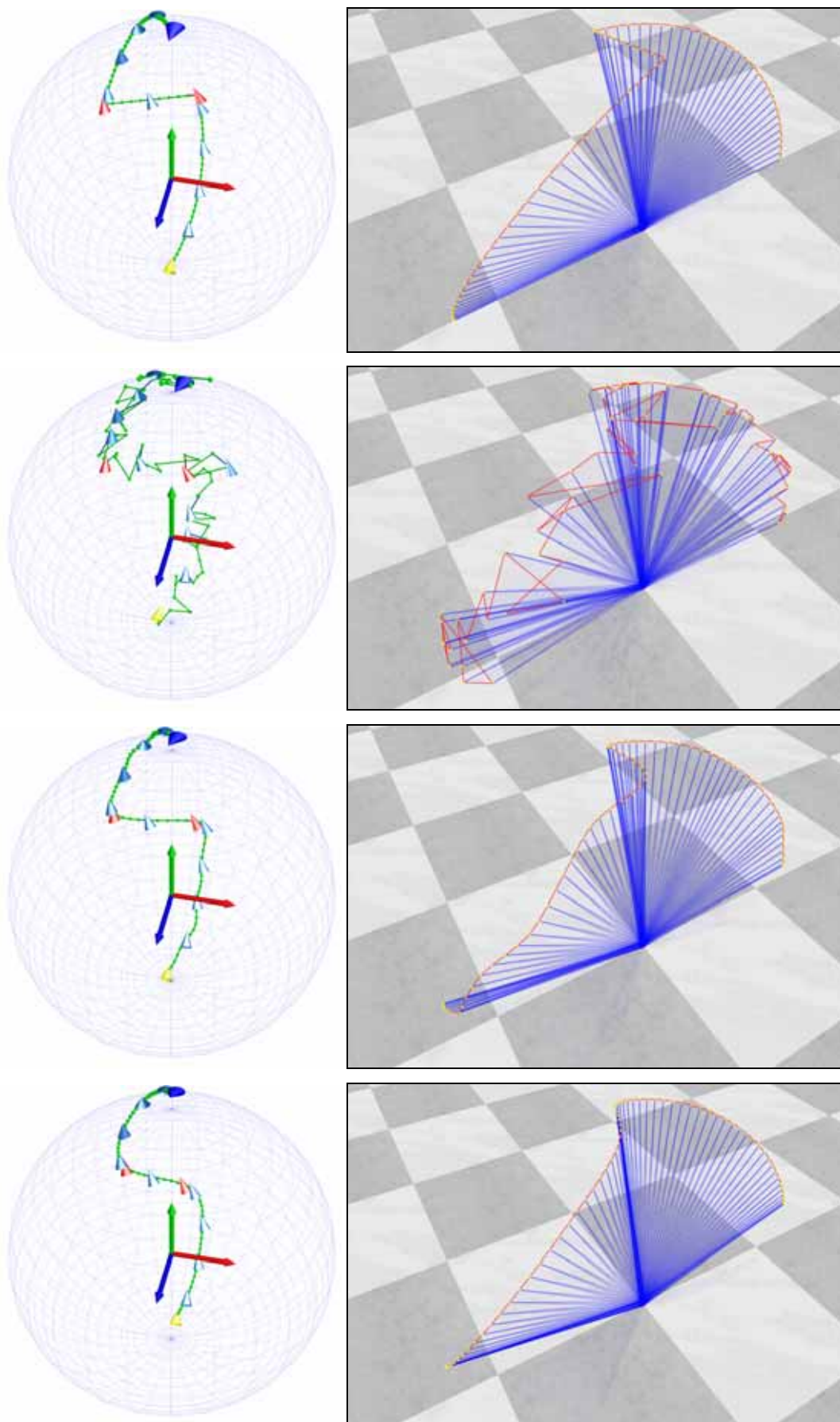


**Rysunek 6-10: Wizualizacja rezultatów filtracji *LinQuat* – przykład niestabilnego procesu dla parametru  $c > 1$  ( $c = 1,005$ ). Wynik filtracji odpowiednio po 10 i 200 iteracjach.**

Charakter zależności przebiegu procesu filtracji od współczynnika dyfuzji jest podobny do przypadku filtracji obrazów dwuwymiarowych przy użyciu filtrów opartych na liniowym równaniu różniczkowym dyfuzji (rozdział 2.6.2, rysunek 2-10). W tamtym przypadku również filtracja nie mogła poprawić jakości sygnału powyżej pewnej wartości maksymalnej. Natomiast pogorszenie się sygnału po osiągnięciu maksymalnej wartości współczynnika w obu sytuacjach wynika z jego rozmyciem, czyli dalszą utratą informacji następującej w wyniku dyfuzji już po usunięciu sygnału zakłócającego. W przypadku obrazów było to rozmycie krawędzi obiektów, w przypadku trajektorii kwaternionów jest to zakłócenie położenia na skutek interpolacji.

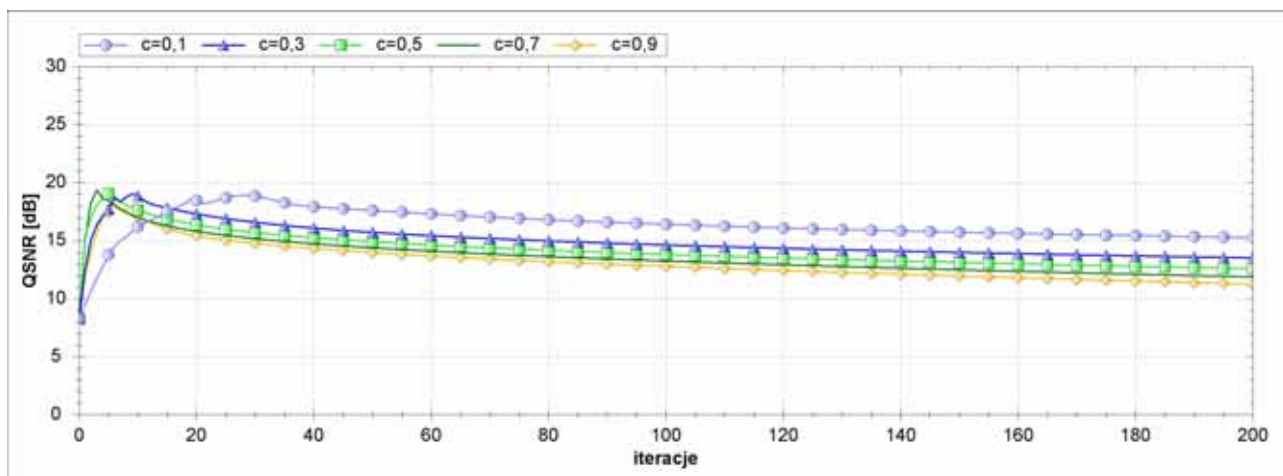
Na rysunku 6-11 przedstawiono wybrane iteracje procesu filtracji sygnału testowego nr 1 (parametry zgodne z przykładem rozpatrywanym poprzednio). W pierwszym wierszu dla łatwiejszego porównania przedstawiono sygnał oryginalny. Drugi wiersz przedstawia sygnał zakłócony przed rozpoczęciem procesu filtracji. W trzecim wierszu pokazany jest wynik filtracji otrzymany po 10 iteracjach (maksymalna wartości miary QSNR). Wizualizacja na sferze rotacji dobrze pokazuje, że sygnał zachowuje względnie podobny przebieg do jego wersji oryginalnej. Widoczny jest jednak znaczący wpływ wcześniejszego zaszumienia, które spowodowało zniekształcenie trajektorii we wszystkich 3 osiach rotacji.

Ostatni wiersz przedstawia wynik filtracji po 50 iteracjach. Zniekształcenie sygnału występuje w tej przede wszystkim w związku z nadmierną filtracją, która powoduje, że położenia elementów trajektorii (odpowiednik wartości jasności punktów dla procesu operującego na obrazach) zmniejszają swoją odległość i dążą do jednego punktu. Należy zwrócić również uwagę na efekt wywołany użyciem warunku brzegowego (traktowanie sygnału, jako nieskończonego przez lustrzane odbicie trajektorii na osi czasu  $\tau$  (której nie należy mylić z czasem przebiegu filtracji  $t$ )). Powoduje to, że elementy krańcowe trajektorii również zaczynają się oddalać od swojego położenia początkowego. W kolejnym podpunkcie zostaną przedstawione różnice wyników filtracji w zależności od zastosowanego warunku brzegowego.

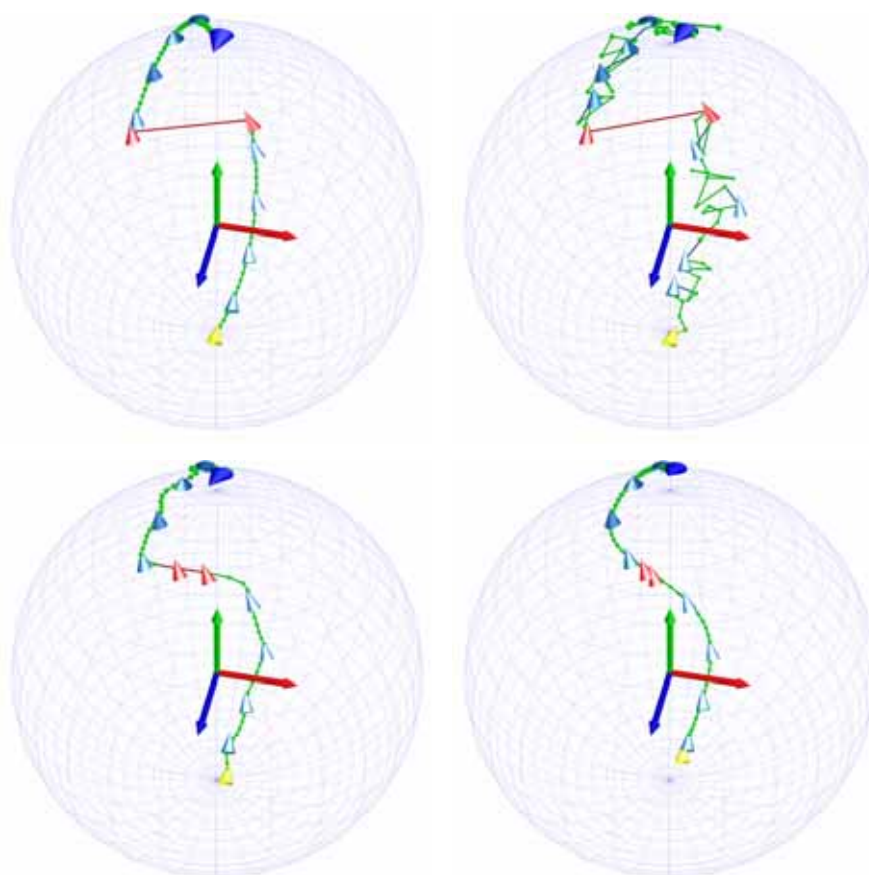


Rysunek 6-11: Wizualizacja rezultatów filtracji *LinQuat* – syntetyczna trajektoria testowa nr 1 (w kolejnych wierszach: sygnał oryginalny, zaszumiony, iteracje 10 i 50).

O wiele gorszą skutecznością cechuje się filtracja przeprowadzona dla sygnału testowego nr 1a. Wykres współczynnika jakości QSNR w funkcji iteracji dla tego przypadku pokazany jest na rysunku 6-12. Parametry filtracji pozostają niezmienione (warunek brzegowy typu D). Wizualizacja wyników dla wybranych iteracji przedstawiona jest na rysunku 6-13 (współczynnik dyfuzji  $c=0,5$ ).



Rysunek 6-12: Miara jakości QSNR dla rezultatów filtracji LQ w zależności od parametru  $c$  – syntetyczna trajektoria testowa nr 1a.



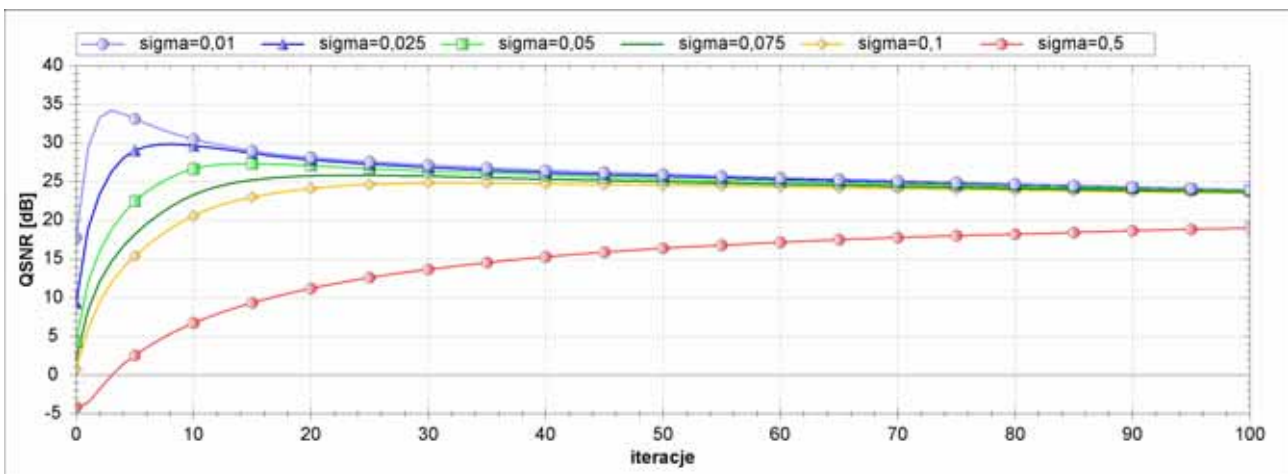
Rysunek 6-13: Wizualizacja rezultatów filtracji *LinQuat* – syntetyczna trajektoria testowa nr 1a (sygnał oryginalny, zaszumiony, iteracje 5 i 50).

Sygnal ten jest dość specyficzny ze względu na nagłą zmianę wartości rotacji, która występuje w klatce 30. Został on przygotowany specjalnie w celu przetestowania własności filtracji dla trajektorii, w której występuje nagła zmiana prędkości kątowej. Wykres na rysunku 6-12 pokazuje, że filtr *LinQuat* bardzo słabo sobie radzi z taką sytuacją. Na rysunku 6-13 dobrze widać, że już po 5 iteracjach (dla których występuje maksimum wartości współczynnika QSNR) fragment trajektorii reprezentujący nagłą zmianę prędkości kątowej jest znacznie zniekształcony (zaznaczony kolorem czerwonym). Dalsze zniekształcenie prowadzi do sytuacji, gdy sygnał staje się coraz bardziej zbliżony do sygnału testowego nr 1, w którym skok taki nie występował. Potwierdza to wniosek, że filtracji typu *LinQuat* dąży do ujednoczenia prędkości kątowej oraz zminimalizowania krzywizny trajektorii. W przypadku zastosowanego warunku brzegowego typu D oznacza to, że w granicy chwilowa prędkość kątowa będzie minimalizowana do zera.

Dla pozostałych trajektorii testowych zależność pomiędzy skutecznością filtracji *LQ* oraz wartością współczynnika dyfuzji  $c$  była podobna. Analiza działania filtracji dla tych trajektorii będzie przeprowadzona w zależności od innych czynników.

## 6.2.2 Skuteczność filtracji *LQ* w zależności od stopnia zaszumienia trajektorii

Dla stałej wartości współczynnika dyfuzji  $c=0,5$  przeprowadzono 5 prób filtracji trajektorii testowej nr 2 przy różnym stopniu wprowadzenia sygnału zakłócającego. Miarą stopnia zakłóceń dodanych do sygnału oryginalnego jest odchylenie standardowe  $\sigma$  stosowane w rozkładzie normalnym użytym do generacji składowej losowej (patrz rozdział 4.2.7). Otrzymane wyniki w postaci wykresu miary jakości QSNR przedstawione są na rysunku 6-14.



Rysunek 6-14: Miara jakości QSNR dla rezultatów filtracji *LQ* w zależności od parametru stopnia zaszumienia sygnału — syntetyczna trajektoria testowa nr 2,  $c=0,5$ .

Efekty filtracji silnie zależą od początkowego stopnia zaszumienia w pierwszej fazie przeprowadzanego procesu. Widoczne jest, że dla niewielkiego zaszumienia  $\sigma=0,01$  jakość sygnału szybko zostaje poprawiona, a następnie obniża się na skutek wcześniej omawianego efektu będącego odpowiednikiem rozmycia krawędzi. Wprowadzona dyfuzja nie jest w stanie bardziej poprawić jakości trajektorii, a staje się w tym przypadku powodem do jego pogorszenia. Pogorszenie jakości sygnału występuje w sensie zmniejszenia jego podobieństwa do sygnału oryginalnego według miary QSNR. Proces



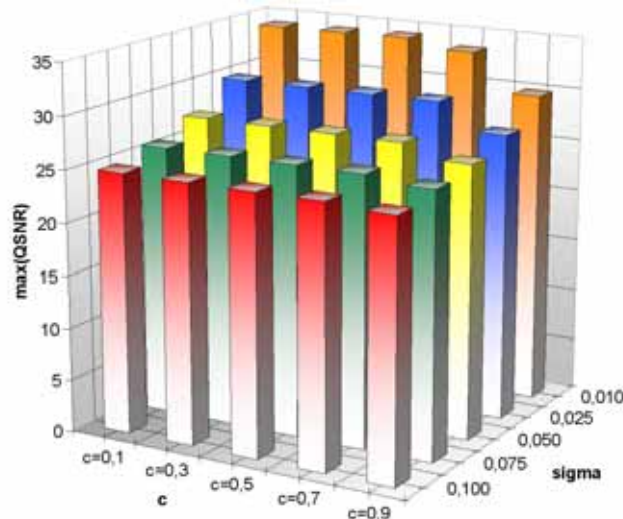
minimalizacji energii sygnału (według własności przedstawionych w rozdziale 5) jest stały i dąży do skupienia trajektorii w jednym punkcie.

Efekt ten jest dobrze widocznego dla wyników filtracji otrzymanych po dużej liczbie iteracji. Widoczne jest, że niezależnie od początkowego zaszumienia sygnału, miara QSNR przyjmuje podobną wartość już po około 80 iteracjach. Od tego momentu rozpoczyna się jakość sygnału zaczyna konsekwentnie się zmniejszać. Taki rezultat można łatwo wytłumaczyć jeszcze raz analizując wizualizację wyników filtracji pokazaną na rysunku 6-11. Już po 50 filtracjach prawie wszystkie składowe o wyższych częstotliwościach wprowadzone przez początkowe zaszumienie, zostały usunięte. Kolejne iteracje przynoszą dalsze pogorszenie sygnału, ponieważ trajektoria filtrowana stale oddala się od trajektorii początkowej – zarówno ze względu na położenie poszczególnych punktów, jak i ze względu na podobieństwo kształtu, który również staje się coraz bardziej zniekształcone.

Dokonano również eksperymentu, w którym wyznaczono maksymalną wartość współczynnika QSNR uzyskaną w ciągu 500 iteracji dla różnego stopnia zaszumienia początkowego w zależności od wartości współczynnika dyfuzji. Wyniki zostały zebrane w tabeli 3 oraz zobrazowane na wykresie (rysunek 6-15).

sigma	c=0,1	c=0,3	c=0,5	c=0,7	c=0,9
0,010	34,08	34,13	34,23	33,45	29,85
0,025	29,80	29,84	29,88	29,93	27,45
0,050	27,33	27,34	27,35	27,36	26,11
0,075	25,82	25,84	25,85	25,87	25,33
0,100	24,86	24,87	24,88	24,90	24,61

Tabela 3: Maksymalna wartość miary QSNR [dB] uzyskana w ciągu 500 iteracji dla filtracji LQ. Trajektoria testowa nr 2, sygnał zakłócający o rozkładzie normalnym.



Rysunek 6-15: Maksymalna wartość miary jakości QSNR dla filtracji LQ w zależności od parametru  $c$  i stopnia zaszumienia sygnału.

Otrzymane wyniki potwierdzają wcześniej zaobserwowaną prawidłowość dotyczącą spadku jakości otrzymywanego sygnału dla dużych wartości współczynnika  $c$ . Jest to szczególnie dobrze widocznego dla małej wartości zaszumienia, kiedy usunięcie zakłóceń może być szybko zrealizowane, a duża wartość współczynnika dyfuzji powoduje szybkie zniekształcenie sygnału. Widać również, że dla dużego

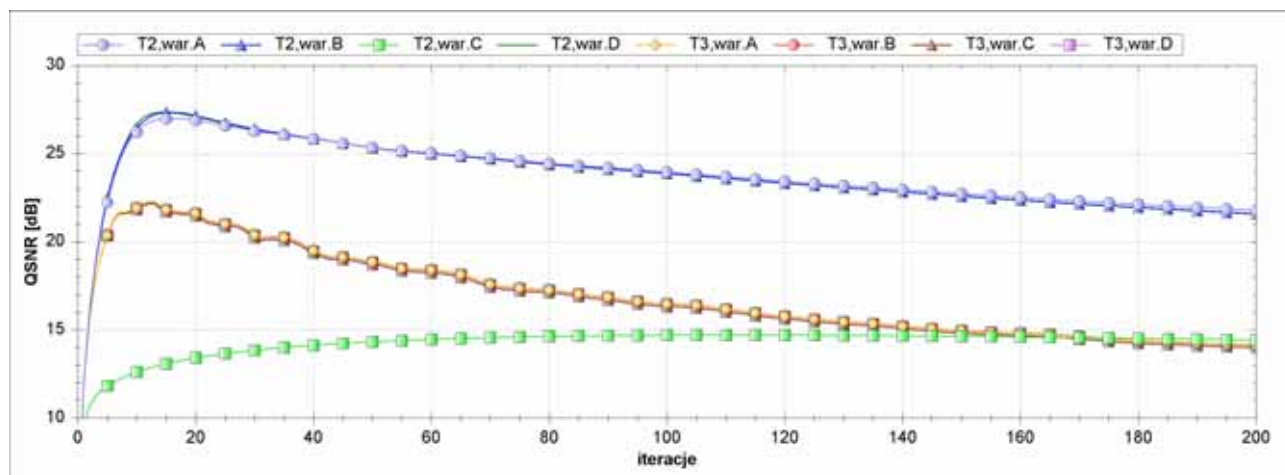
zasmucenia ( $\sigma = 0,1$ ) niezależnie od przyjętego współczynnika dyfuzji nie da się uzyskać trajektorii przefiltrowanej o jakości lepszej niż około 24,9 dB. Wynika to z opisywanego efektu, który w tym przypadku jest uwypuklony – usunięcie składowej zakłócającej jest ściśle związane z wprowadzeniem rozmycia wynikającego z dyfuzji.

Na podstawie grupy przeprowadzonych eksperymentów można stwierdzić, że optymalnym współczynnikiem dyfuzji jest wartość około 0,3 – 0,7. Otrzymuje się proces, który charakteryzuje się możliwością osiągnięcia maksymalnej wartości miary jakości QSNR przy jednoczesnym zachowaniu kształtu sygnału przez dużą liczbę iteracji. W przypadku, gdy najistotniejsze jest wprowadzenie małej liczby zniekształceń wynikających z dyfuzji należy stosować współczynnik  $c$  o niewielkich wartościach. W dalszych eksperymentach częstko będzie przyjmowana wartość  $c=0,5$ , jako filtracji referencyjna typu LQ.

Otrzymane zależności od stopnia zasmucenia oraz współczynnika dyfuzji są charakterystyczne jedynie dla homogenicznej filtracji *LinQuat*. W kolejnych rozdziałach zostanie pokazane, że zależności takie nie występują, gdy proces jest uzależniony od danych występujących w trajektorii.

### 6.2.3 Skuteczność filtracji LQ w zależności od warunku brzegowego

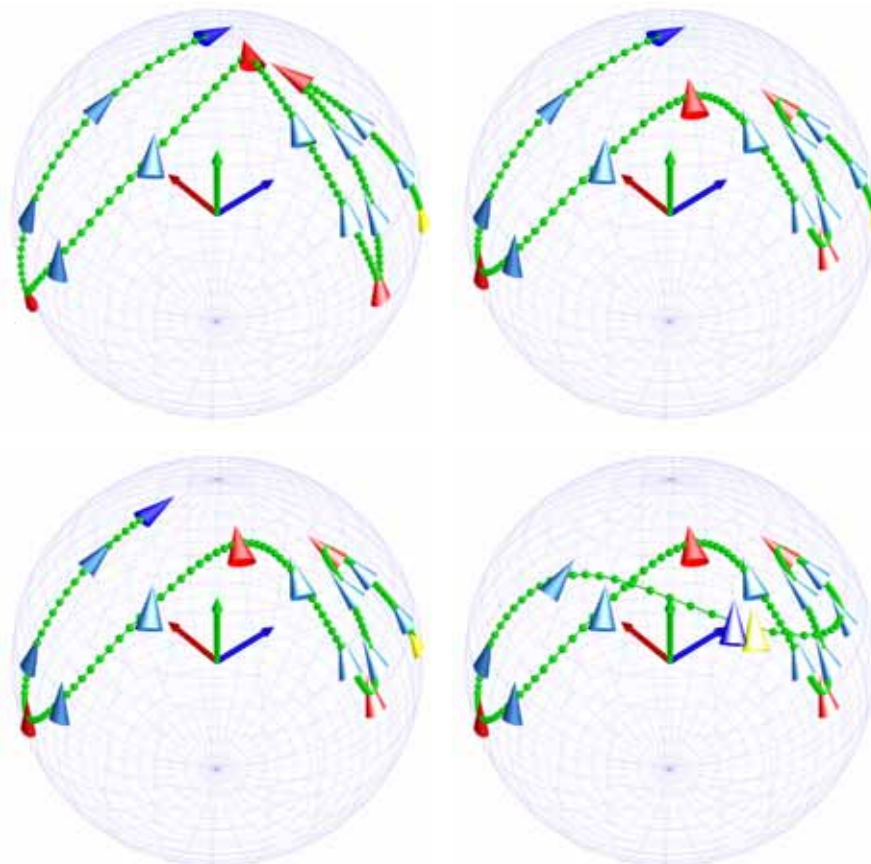
Wynik filtracji może zależeć od przyjętego warunku brzegowego. Wykonano filtrację dla trajektorii testowych nr 2 i 3 przy identycznej wartości  $c=0,5$  i zasmuceniu o rozkładzie normalnym  $\sigma=0,05$ . Rezultaty w postaci wykresu współczynnika QSNR przedstawione są na rysunku 6-16.



**Rysunek 6-16: Miara jakości QSNR dla rezultatów filtracji LQ w zależności od przyjętego warunku brzegowego – syntetyczna trajektoria testowa nr 2 (T2) oraz nr 3 (T3).**

Dla trajektorii testowej nr 2, wyniki filtracji przy użyciu warunków brzegowych A, B, D w praktyce się pokrywają. Identyfikacja rezultatów dla warunku B i D jest charakterystyczna dla filtracji typu *LinQuat*. W przypadku, gdy brana będzie pod uwagę wartość pochodnej dla elementów brzegowych (jak to jest dla filtrów NLQ, AQ) rezultat ten będzie inny. Podobne rezultaty otrzymane dla warunku A nie wynikają bezpośrednio z identyczności otrzymywanych rezultatów, ale także z charakteru stosowanej miary podobieństwa sekwencji. Przy ocenie jakości sygnału brana jest także składowa opisująca zmianę krzywizny trajektorii, która powyżej 50 iteracji jest taka sama dla omawianych warunków brzegowych. Dla wielu iteracji różnica miary dla filtracji z warunkami A i D może okazać się znacząca.

Największa różnica w otrzymanych wynikach dotyczy warunku typu C. Niska jakość sygnału po filtracji wynika z przyjętego błędnego (dla trajektorii nr 2) założenia o periodyczności sygnału. Przez to proces filtracji zakłada, że elementy pierwszy i ostatni trajektorii tworzą ze sobą relację sąsiedztwa. Stąd interpolacja w procesie dyskretyzacji równania różniczkowego powoduje znaczne zniekształcenie trajektorii przez zbliżenie krańcowych elementów do siebie. Efekty te zostały zobrazowane na rysunku 6-17.



**Rysunek 6-17: Wizualizacja rezultatów filtracji *LinQuat* w zależności od warunku brzegowego dla trajektorii testowej nr 2. Pierwszy wiersz – tr. oryginalna, warunek A. Drugi wiersz – warunek B/D, warunek C.**

Dla zwiększenia czytelności rysunek sposób wyniki filtracji dokonanej na niezaszumionej trajektorii testowej nr 2. Dla porównania, jako pierwsza została pokazana niezmodyfikowana trajektoria oryginalna. W prawym górnym rogu przedstawiony jest wynik filtracji dla warunku brzegowego typu A. Widoczne jest, że elementy pierwszy i ostatni są stałe, a pozostałe elementy trajektorii zostały odpowiednio przekształcone. Dobrze widoczny w tym przypadku jest efekt nadmiernego wygładzania trajektorii w punktach o dużej zmianie krzywizny ruchu. Ponadto w takim przypadku filtracja nie dąży do pojedynczego punktu w przestrzeni rotacji. W tym przypadku stanem ustalonym jest trajektoria, dla której elementy są równorozłożone (stała prędkość kątowna w każdym punkcie) wzdłuż linii geodezyjnej wyznaczonej przez element początkowy i końcowy.

W lewym dolnym rogu pokazany jest efekt filtracji dla warunku brzegowego typu B. Ponieważ dla filtracji LQ jest on identyczny, jak w przypadku warunku typu D, zrezygnowano z dodatkowej wizualizacji dla obu tych warunków. W tym przypadku widoczne jest, że w stanie ustalonym trajektoria jest zredukowana do pojedynczego punktu w przestrzeni rotacji.

W prawym dolnym rogu dobrze widoczny jest efekt periodyzacji sygnału, który oryginalnie nie jest okresowy. Elementy brzegowe zostały do siebie znacznie zbliżone, co powoduje, że podobieństwo takiego sygnału jest dalekie od sygnału oryginalnego. W efekcie zastosowanie tego warunku generuje wysoce zniekształconą trajektorię wynikową.

Podobne zależności związane z warunkiem brzegowym dotyczą trajektorii testowej nr 3. Jednakże trajektoria ta z założenia jest okresowa. Stąd również dla warunku brzegowego typu C otrzymuje się podobne rezultaty, jak w przypadku warunków typu A, B, D. W efekcie wszystkie 4 krzywe na rysunku 6-16 pokrywają się.

Podobieństwo otrzymanych wyników ze względu na uzyskaną jakość QSNR nie jest jedynym wyznacznikiem możliwości zastosowania danego warunku. Warunek typu A powinien być stosowany w sytuacji, gdy wiadomo, że początkowe i końcowe położenie elementów trajektorii jest niezakłócone lub zakłócone w niewielkim stopniu (sytuacja taka może wystąpić w niektórych systemach przechwytywania i analizy ruchu). Warunki B/D wydają się być najbardziej naturalne pozwalające na wprowadzenie najmniejszej ilości dodatkowych zakłóceń (dla niewielkiej liczby iteracji). Warunek C powinien być stosowany tylko w przypadku sekwencji periodycznych (sytuacja często spotykana przy rejestracji ruchów periodycznych takich jak bieg, itp.).

#### 6.2.4 Podsumowanie badań dla filtracji *LinQuat*

Na podstawie przeprowadzonych badań dotyczących algorytmu realizującego filtrację LQ można zebrać następujące wnioski:

- filtracja w sposób skuteczny usuwa wprowadzony sygnał zakłócający, co prowadzi do trajektorii, która jest wygładzona w znacznym stopniu,
- usuwanie zakłóceń może wprowadzać dodatkowe zniekształcenia polegające na nadmiernym wygładzaniu zmian kierunku (krzywizny) trajektorii; wynika to z własności przeprowadzanego procesu dyfuzji, który dąży do minimalizacji odległości pomiędzy wszystkimi punktami,
- efekty filtracji można porównać do algorytmów rozmywania gaussowskiego (ang. *Gaussian blur*), które realizują liniowe równanie różniczkowe dla obrazów; w przypadku trajektorii kwaternionów rozmycie wynika z utraty cech charakterystycznych dla sygnału reprezentowanych przez zmianę prędkości kątowej oraz krzywiznę trajektorii (analogia do utraty cech obrazów reprezentowanych przez krawędzie).

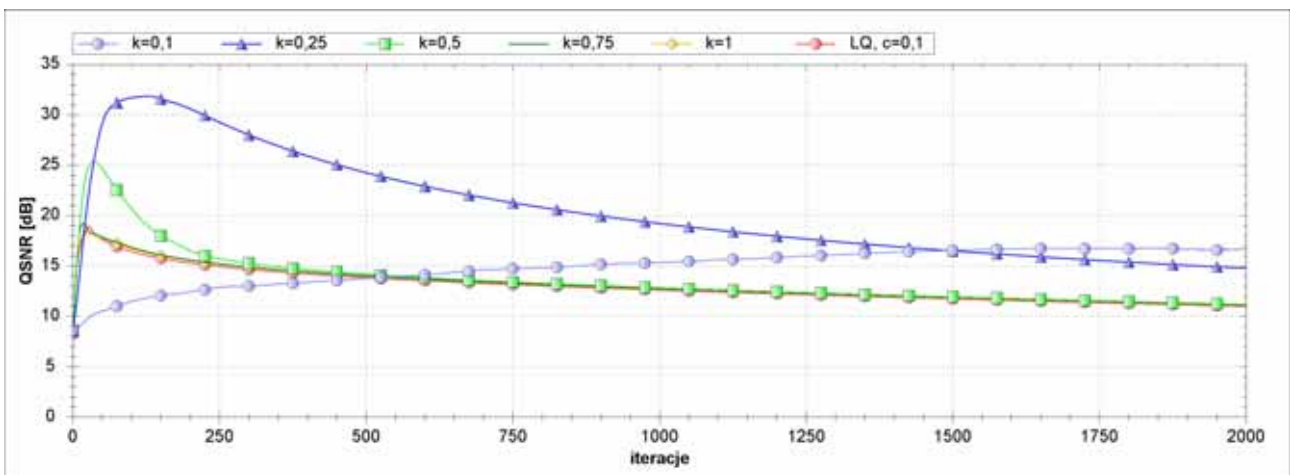
### 6.3 Skuteczność algorytmu *NonLinQuat* dla syntetycznych trajektorii testowych

W rozdziale 5.3 został zaproponowany nieliniowy algorytm filtracji trajektorii rotacji nazwany *NonLinQuat*. Jego konstrukcja została oparta na dyskretyzacji równania różniczkowego realizującego model *Perona-Malik* dla obrazów. Istotną różnicą, która już wcześniej była podkreślana, jest inny charakter danych opisujących obraz dwuwymiarowy i trajektorię w przestrzeni rotacji. W przypadku obrazów elementem

pozwalającym na rozróżnianie znaczeniowe jego fragmentów jest zmiana jasności pikseli. Dla trajektorii reprezentujących ruch następuje ciągła zmiana położenia będącego odpowiednikiem wartości punktu na obrazie. Zmiana prędkości ruchu jest więc tylko jednym z kryteriów, które będą decydowały o przekazie znaczeniowym danej trajektorii. W praktyce rzadko występuje przypadek, w którym zmiana prędkości ruchu jest jedynym takim kryterium. Algorytm NLQ, będący uogólnieniem modelu PM dla kwaternionów, uwzględnia jedynie czynnik prędkości ruchu. Jest on podstawą do realizacji bardziej złożonego podejścia do filtracji przedstawioną w pełni w postaci algorytmu *AnisoQuat*. Stąd testowanie algorytmu zostanie ograniczone jedynie do sekwencji testowej numer 1a, która pozwoli na pokazanie jego własności w przypadku, gdy w trajektorii w przestrzeni rotacji występuje nagła zmiana prędkości wykonywanego ruchu. Dla pozostałych trajektorii typu 1 lub 2 algorytmu NLQ nie będzie wprowadzał żadnej poprawy działania ze względu na stałą chwilową prędkość kątową ruchu. W takich przypadkach algorytm NLQ może co najwyżej dawać tak samo dobre rezultaty filtracji, jak w przypadku podstawowego algorytmu LQ.

### 6.3.1 Skuteczność filtracji NLQ w zależności od współczynnika wrażliwości $K$

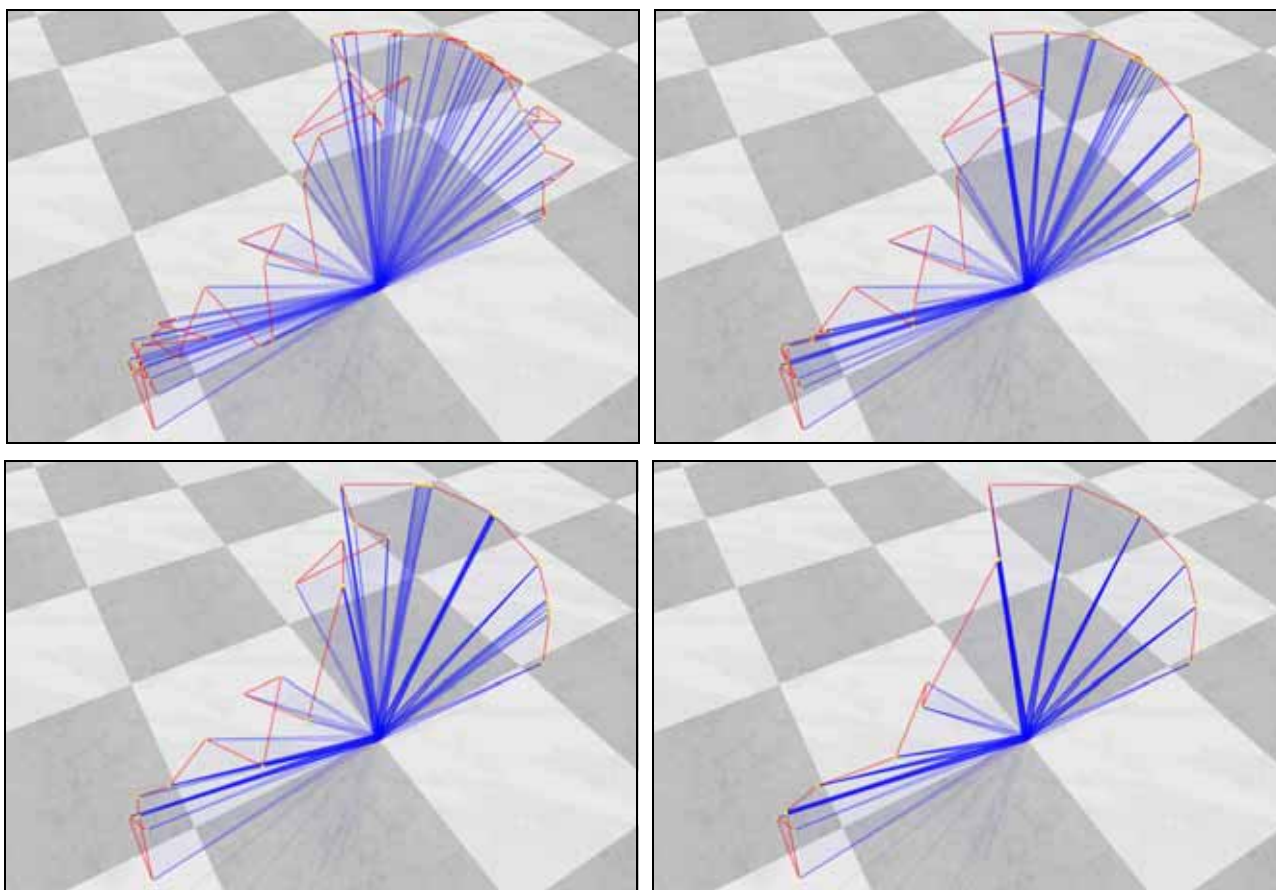
Parametr  $K$  jest używany w modelu *Perona-Malik* do określania kontrastu (lokalnej różnicy jasności punktów), który jest wymagany do rozróżnienia krawędzi obszarów. Odpowiadający mu parametr w algorytmie NLQ decyduje o zmianie skalarnej wartości prędkości kątowej wymaganej do zróżnicowania odcinków ruchu reprezentowanego przez trajektorię. Dla trajektorii testowej nr 1a wykonano serię eksperymentów numerycznych dla różnej wartości parametru  $K$  (o stałej wartości). Pozostałe parametry filtracji w każdym z przebiegów są odpowiednio równe:  $\lambda=0,1$ ;  $c=1,0$ ; funkcja  $g_I$ ; warunek brzegowy typu D. Do trajektorii wprowadzano sygnał zakłócający o rozkładzie normalnym,  $\sigma=0,5$ . Otrzymane rezultaty zebrane zostały na rysunku 6-18. Na wykresie zaznaczono dodatkowo krzywą reprezentującą wynik filtracji typu LQ przy współczynniku dyfuzji  $c=0,1$ .



Rysunek 6-18: Miara jakości QSNR dla rezultatów filtracji NLQ w zależności od współczynnika wrażliwości  $k$  – syntetyczna trajektoria testowa nr 1a.

Na podstawie otrzymanych rezultatów można wyróżnić trzy grupy wyników zależne od przyjętego parametru  $K$ . Dla bardzo małych wartości ( $K=0,1$  – krzywa niebieska) otrzymuje się sytuację, w której dyfuzja jest zbyt mocno zahamowana przez czynnik nieliniowy w równaniu. Jeżeli sygnał nie byłby

zakłócony, to dzięki temu nie byłby on w znacznym stopniu deformowany przez za wysoki poziom dyfuzji, co obserwowano dla filtracji LQ. Jednak wprowadzony czynnik zakłócający powoduje, że lokalnie wyznaczone zmiany chwilowej prędkości kątowej są traktowane, jako znaczące, co powoduje zmniejszenie szybkości dyfuzji dla tych elementów. W praktyce w takim przypadku w trakcie procesu filtracji zaczynają występować oscylacje elementów trajektorii, które prowadzą do chwilowych stanów ustalonych. Niedokładności wynikające z błędów numerycznych mogą być przyczyną wytrącenia trajektorii z tego stanu, który prowadzi do kolejnej postaci ustalonej. Na rysunku 6-19 przedstawiono wybrane iteracje przykładowej realizacji tego typu procesu. Widoczne jest, że dla kolejnych iteracji elementy trajektorii tworzą skupiska, które pozostają stabilne przez pewien określony odcinek czasu. Analizując iteracje 100 (prawa góra rysunku) i 300 (lewy dół rysunku) można zauważyć grupy elementów, które były zgrupowane, a następnie ich wzajemna odległość się ponownie zwiększyła. Po 1000 iteracjach otrzymuje się stan, w którym elementy zostają zgrupowane w lokalne skupiska. Ich wzajemne położenie wynika z przyjętej wartości współczynnika  $K$ , który dla takiego przypadku hamuje lokalną dyfuzję.



**Rysunek 6-19: Wizualizacja rezultatów filtracji *NonLinQuat* – syntetyczna trajektoria testowa nr 1a. Trajektoria po odpowiednio 0, 100, 300, 1000 iteracjach.**

Należy zwrócić uwagę, że w tym przypadku również dyfuzja na krańcach trajektorii została zahamowana. Jest to wynikiem wystąpienia opisanego efektu hamowania dyfuzji, która dla elementów brzegowych wynika z rozszerzenia sygnału na dziedzinę nieskończoną (zastosowano warunek brzegowy typu D).

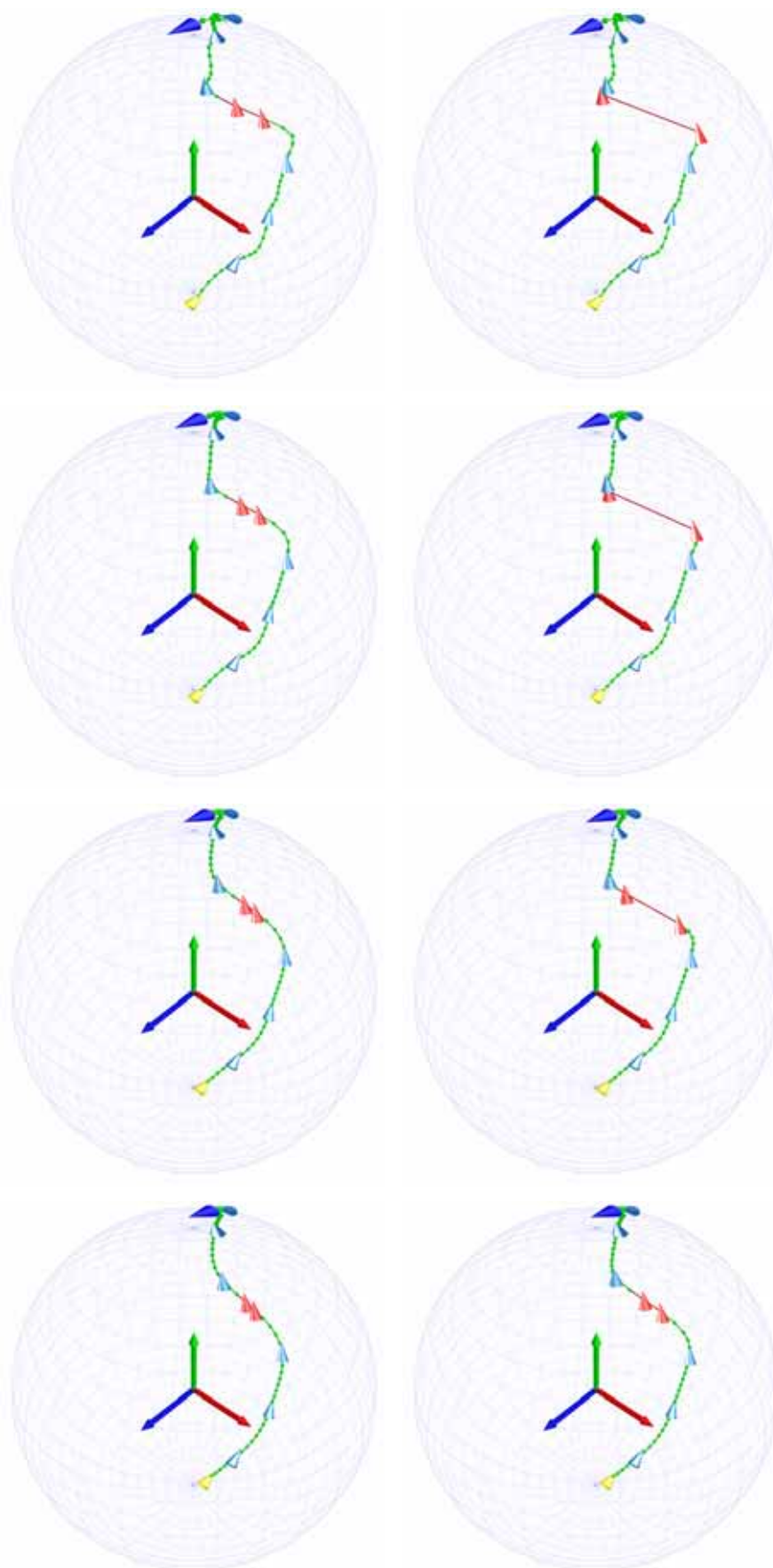
Druga grupa wyników występuje w rozpatrywanych badaniach dla filtracji przy parametrach  $K=0,25$  oraz  $K=0,5$ . Wzrost wartości parametru  $K$  pozwolił na przeprowadzenie dyfuzji dla tych fragmentów trajektorii, w których była ona zbyt mocno zahamowana w poprzednim przypadku. Jednocześnie parametr ten jest jeszcze na tyle niewielki, że dla znaczącej zmiany prędkości kątowej (która występuje w połowie trajektorii) dyfuzja jest hamowana. Dzięki temu otrzymuje się niezahamowany proces dyfuzji dla fragmentu jednolitego, a zostaje ona zmniejszona dla fragmentu o dużym gradiencie prędkości. Jest to analogia do nieliniowego procesu dyfuzji typu *Perona-Malik* dla obrazów. Wizualizację wyników filtracji dla przykładowej realizacji procesu przy parametrze  $k=0,25$  pokazano na rysunku 6-20 (prawa kolumna). Dla porównania procesów, w lewej kolumnie umieszczono wyniki filtracji uzyskane przy użyciu procesu *LinQuat* o odpowiadających parametrach ( $c=0,1$ ). Użyto sygnału zakłócającego o rozkładzie normalnym,  $\sigma=0,05$ .

Dla pierwszych 25 i 50 iteracji widać największą różnicę dla obu procesów. Filtracja LQ jest realizowana ze stałą szybkością niezależnie od jakichkolwiek parametrów trajektorii. Dla filtracji NLQ przyjęty parametr  $K$  jest wystarczająco mały, aby zahamować dyfuzję przy dużej zmianie prędkości kątowej, a jednocześnie wystarczająco duży, aby filtracja dla pozostałych fragmentów zachodziła z szybkością podobną, jak w procesie LQ. Funkcja stopująca  $g$  pełni więc rolę ograniczenia uzależnionego od wartości elementów trajektorii.

Po około 100 iteracjach dyfuzja zahamowana dla fragmentu środkowego trajektorii zaczyna powoli być realizowana. Zmniejszenie odległości w przestrzeni rotacji dla wyróżnionych elementów powoduje, że dyfuzja będzie w coraz mniejszym stopniu hamowana. Po 150 iteracjach (ostatni wiersz na rysunku) filtracja przebiega z podobną prędkością dla całej trajektorii, co powoduje, że sygnał staje się coraz bardziej zniekształcony.

Efekty te tłumaczą początkowy znaczny wzrost jakości sygnału pokazany na rysunku 6-18. Gdy dyfuzja przestaje być hamowana, to rozpoczyna się nadmierne wygładzanie trajektorii objawiające się w zmniejszeniu podobieństwa do trajektorii oryginalnej. Jest to kolejna analogia do procesu typu *Perona-Malik*, który w nieskończoności jest równoważny procesowi dyfuzji liniowej.

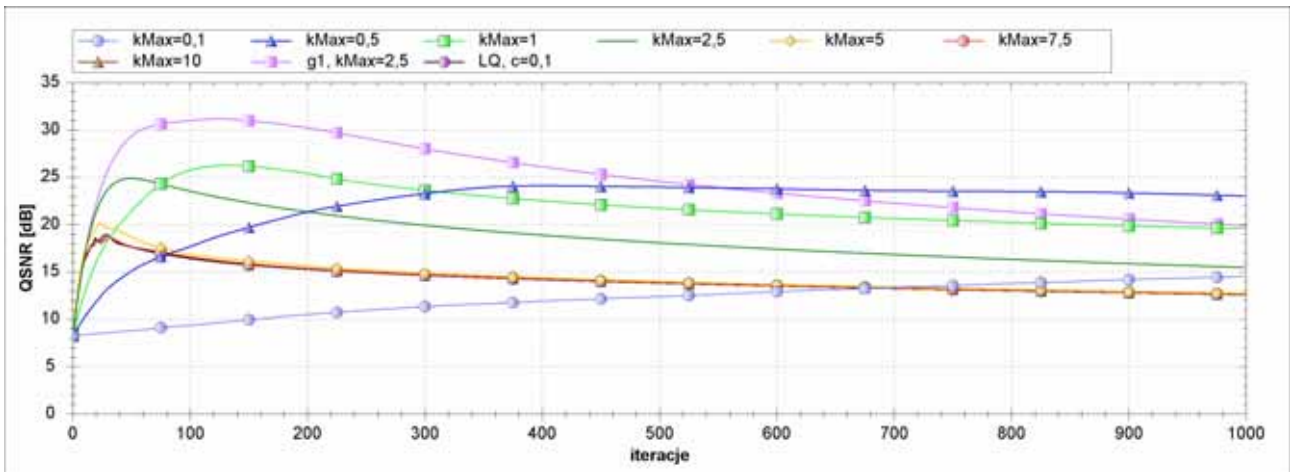
Rezultaty filtracji dla parametru  $K>0,5$  w praktyce dokładnie pokrywają się z wynikami otrzymanymi przy filtracji LQ. Oznacza to, że wartość parametru różnicującego  $K$  jest na tyle duża, że funkcja  $g$  przyjmuje wartości zbliżone do 1. Ograniczenie związane z występowaniem czynnika nieliniowego nie jest w tym przypadku znacząco aktywne, przez co filtracja przebiega identycznie, jak to jest realizowane w procesie typu *LinQuat*.



Rysunek 6-20: Porównanie rezultatów filtracji LQ (lewa kolumna) i NLQ (prawa kolumna) dla sekwencji testowej nr 1a. W kolejnych wierszach iteracje: 25, 50, 100, 150.



Podobne własności zostały zaobserwowane zarówno dla filtracji wykorzystującej funkcję stopującą drugiego rodzaju  $g_2$ , jak i dla przypadku, gdy współczynnik  $k$  jest wyznaczany adaptacyjnie. Na rysunku 6-21 zebrano wyniki tak przeprowadzonych eksperymentów dla trajektorii testowej nr 1a przy parametrach  $c=1,0$ ;  $\lambda=0,1$ . Dodatkowo zamieszczono wyniki filtracji tej samej trajektorii otrzymanej przy użyciu funkcji  $g_1$  oraz adaptacyjnym wyznaczaniu współczynnika  $K$  o parametrze  $K_{max}=2,5$  (przebieg, dla którego uzyskano maksymalną miarę jakości filtracji QSNR). Pokazano również rezultat filtracji typu LQ z parametrem  $c=0,1$  (pokrywa się z przebiegami dla  $K_{max}>7,5$ ).



**Rysunek 6-21: Miara jakości QSNR dla rezultatów filtracji NLQ w zależności od adaptacyjnego współczynnika wrażliwości  $k$ , funkcja  $g_2$  – syntetyczna trajektoria testowa nr 1a.**

Analogiczną charakterystykę wyników otrzymuje się przy użyciu funkcji  $g_1$ . Przedstawiony na wykresie przebieg dla tej funkcji obrazuje nieznacznie odmienny sposób przebiegu procesu przy tych parametrach. Funkcja ta pozwala na otrzymanie dużego zysku jakości sygnału. Natomiast przy zastosowaniu funkcji stopującej  $g_2$  i parametru  $K_{max}=1$  otrzymuje się wynik filtracji, który dla większej liczby iteracji pozostaje stabilny. Po 3000 iteracji wynik miary jakości sygnału dla filtracji z funkcjami  $g_1$  i  $g_2$  wynosił odpowiednio 13,68 oraz 16,0 dB.

Parametr  $K_{max}$  jest w przypadku adaptacyjnej wersji algorytmu pomocniczy. Jednakże podobnie, jak dla funkcji  $g_1$  również w tym przypadku dla dużych jego wartości otrzymuje się proces nieograniczony przez funkcję stopującą, co prowadzi do filtracji analogicznej, jak przy użyciu algorytmu LQ. Widoczne to jest dla wyników filtracji przy  $K_{max}>0,5$ , które pokrywają się z odpowiadającym przebiegiem LQ.

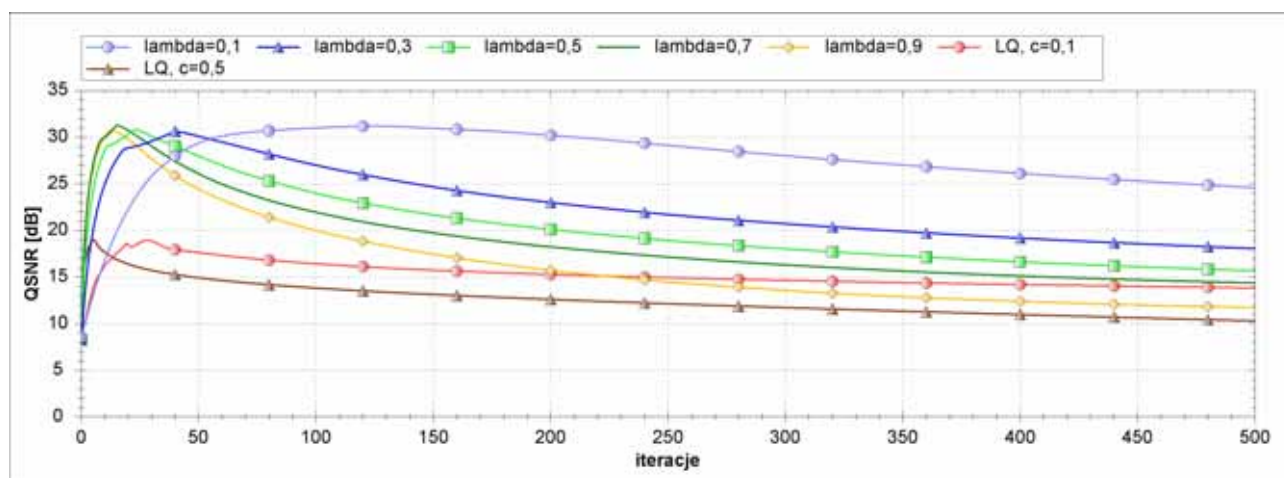
Filtracja przy użyciu funkcji  $g_2$  dla małych parametrów  $K$  przebiega łagodniej od jej odpowiedników dla funkcji  $g_1$ . Również w tym przypadku występuje efekt oscylacyjny prowadzący do grupowania się elementów, jednakże odbywa się to wolniej i bardziej regularnie.

We wszystkich tych przypadkach należy podkreślić jedną prawidłowość. Za każdym razem, nawet gdy występuje lokalna oscylacja elementów trajektorii, spełniona jest własność przestrzennej zbieżności procesu przedstawiona w rozdziale 5.6.3. Jakkolwiek elementy zmieniają swoje położenie, jednak nawet przypadek grupowania nie łamie tej zasady, gdyż wszelka zmiana wynikać może jedynie z położenia pozostałych punktów trajektorii. Przy zachowaniu odpowiednich warunków nałożonych na parametry

algorytmu, interpolacja wykonywana w każdej iteracji zapewnia, że nowe położenie punktu trajektorii nie spowoduje wystąpienie odległości od pozostałych punktów większej od poprzednio istniejącej.

### 6.3.2 Skuteczność filtracji NLQ w zależności od współczynnika $\lambda$

Jak to zostało przy konstrukcji algorytmu filtracji NLQ, współczynnik  $\lambda$  jest równoważny ze współczynnikiem  $c$ . W przedstawionych eksperymentach przyjmuje się wartość  $c=1,0$  dla algorytmu NLQ. O szybkości filtracji decydować więc będzie współczynnik  $\lambda$ . Na rysunku 7-19 zebrano wykresy miary jakości QSNR dla wyników filtracji w zależności od przyjętej wartości parametru  $\lambda$ . Dla wszystkich przebiegów użyto funkcji stopującej  $g_I$ , warunku brzegowego typu D, współczynnika  $K_{max}=2,5$  (wersja adaptacyjna). Dla porównania zaznaczono również wybrane rezultaty uzyskane w trakcie filtracji LQ.



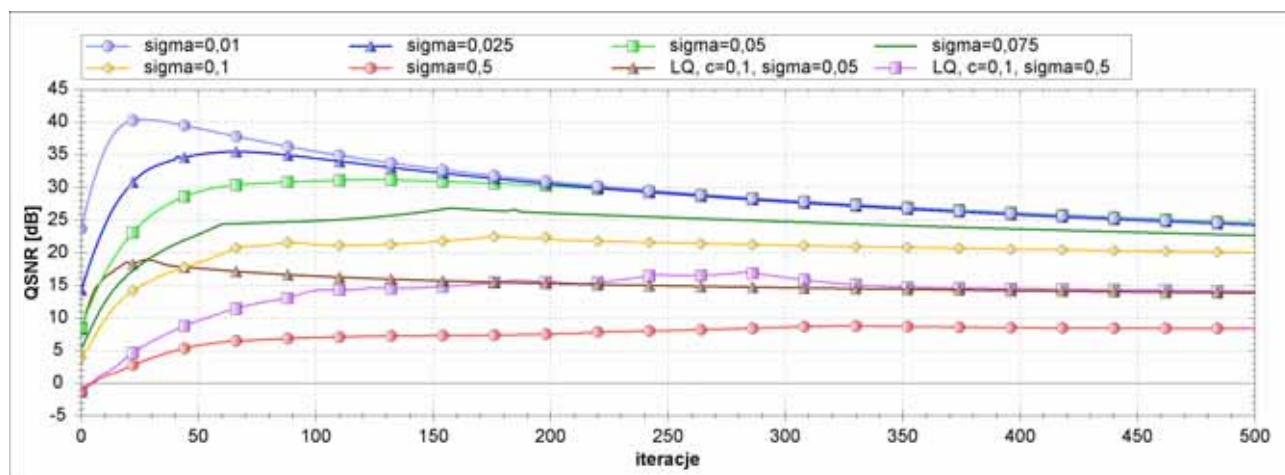
Rysunek 6-22: Miara jakości QSNR dla rezultatów filtracji NLQ w zależności od parametru  $\lambda$  – syntetyczna trajektoria testowa nr 1a.

Otrzymany wykres potwierdza, że parametr  $\lambda$  pełni rolę czynnika sterującego szybkością procesu. Dla małych wartości parametru proces pozwala na powolne osiągnięcie sygnału o wysokiej jakości. Spadek jakości również w tym przypadku następuje powoli – pogarsza się jakości sygnału filtrowanego porównywalnie do filtracji algorytmem LQ. Duże wartości parametru pozwalają natomiast na szybkie osiągnięcie dobrych rezultatów. Jednakże w trakcie kolejnych iteracji równie szybko następuje deformacja trajektorii i spadek wartości miary QSNR. Proces ten powoduje, że trajektoria staje się w granicy tak samo zniekształcona, jak w przypadku odpowiadającemu mu procesowi LQ. Jest to potwierdzenie wcześniej zaobserwowanego efektu (6-18) polegającego na tym, że proces, w którym przestaje występować działanie funkcji stopującej w swoim przebiegu zbliża się do filtracji bez ograniczeń typu LQ.

Otrzymane rezultaty wskazują, że dla filtracji realizowanej algorytmem NLQ w praktyce lepiej jest używać niewielkich wartości współczynnika  $\lambda$ . Również dla takiego współczynnika otrzymuje się jeden z najwyższych uzyskanych wyników filtracji (QSNR=31,22 dB). Reguła ta jest inna niż uzyskana dla filtracji LQ, w której maksymalną poprawę trajektorii uzyskiwało się dla średnich wartości współczynnika dyfuzji. Pokazuje to znaczący wpływ składnika nieliniowego, modyfikującego charakterystykę procesu zależnie od występujących punktów w trajektorii.

### 6.3.3 Podatność filtracji NLQ na poziom zaszumienia sygnału

W kolejnym eksperymencie sprawdzono możliwość odszumiania trajektorii w zależności od początkowego stopnia zaszumienia sygnału. W tym celu rozpatrywana była trajektoria testowa nr 1a z sygnałem zakłócającym o rozkładzie normalnym. Algorytm odszumiający używał funkcji stopującej nr 1, adaptacyjnego wyznaczania współczynnika wrażliwości z parametrem  $K_{max}=2,5$  i szybkością dyfuzji  $\lambda=0.1$ . Na rysunku 6-23 przedstawiono uzyskane rezultaty w funkcji kolejnych iteracji procesu. Dla porównania zaznaczono również wyniki dla dwóch przypadków filtracji algorytmem LQ.



Rysunek 6-23: Miara jakości QSNR dla rezultatów filtracji NLQ w zależności od stopnia zaszumienia sygnału (rozkład normalny) – syntetyczna trajektoria testowa nr 1a.

Podobnie, jak było w przypadku filtracji LQ, otrzymane rezultaty w postaci miary QSNR zależą od początkowego poziomu zaszumienia trajektorii testowej. Im niższy poziom zaszumienia (w postaci mniejszego odchylenia standardowego generowanego rozkładu), tym lepszy rezultat filtracji udaje się osiągnąć. Jednak po około 100-150 iteracjach wyniki filtracji dla zaszumienia o poziomie  $\sigma \leq 0,05$  stają się identyczne. Dla dalszych iteracji taka prawidłowość dotyczy także bardziej znaczącego zaszumienia ( $\sigma = 0,075$ ). Jest to efekt podobny do uzyskiwanego dla filtracji LQ (patrz rysunek 6-14 oraz dwie ostatnie krzywe na wykresie).

Istnieje istotna różnica w otrzymanych wynikach dla obu typów filtracji. Należy zwrócić szczególną uwagę na krzywą zaznaczoną kolorem czerwonym (zaszumienie  $\sigma = 0,5$ ). Reprezentuje ona filtrację dla przypadku bardzo silnie zaszumionego. Widoczne jest, że otrzymany rezultat tylko nieznacznie poprawił początkową jakość trajektorii i utrzymuje się na stałym bardzo niskim poziomie. W przeciwieństwie do pozostałych wyników, ten znajduje się znacząco poniżej rezultatów otrzymywanych dla filtracji LQ. Istotne jest również to, że dla takiego samego poziomu zaszumienia, filtracja LQ uzyskuje po około 300 iteracjach tak samo dobry wynik, jak dla zaszumienia mniejszego.

Na podstawie przeprowadzonych badań można wyciągnąć wniosek, że algorytm NLQ jest bardziej podatny na zaszumienie od algorytmu LQ. Oznacza to, że istnieje pewien poziom początkowego zaszumienia trajektorii, który powoduje, że algorytm nie jest w stanie osiągnąć znaczącej poprawy filtracji sygnału. W takim przypadku zakłócenia w trajektorii powodują, że dyfuzja jest zatrzymana, a występujące

ewentualne oscylacje punktów (analizowane poprzednio) nie prowadzą do poprawy jakości sygnału. Algorytm NLQ jest więc przydatny jedynie do przypadków o zaszumieniu nie przekraczającym określonej granicy. Wartość tej granicy zależy jednak będzie od rozpatrywanego sygnału i jego charakterystyki.

#### 6.3.4 Podsumowanie badań dla filtracji *NonLinQuat*

Eksperymentalne badanie algorytmu NLQ pozwoliło na sformułowanie następujących wniosków:

- Algorytm NLQ jest przydatny tylko w przypadku, gdy w trajektorii należy rozróżnić zmianę chwilowej prędkości kątowej. W pozostałych przypadkach algorytm pozwala najwyżej na uzyskanie efektów nie gorszych niż w przypadku LQ.
- Proces wymaga dobrego określenia parametru wrażliwości  $K$ , od którego zależy rozróżnienie, jaka zmiana prędkości chwilowej jest jeszcze zakłóceniem lub już zmianą pożądaną.
- Zastosowanie funkcji stopującej  $g_1$  pozwala na szybsze osiągnięcie lepszych efektów. Funkcja  $g_2$  daje w rezultacie trajektorię o niższej jakości, jednak bardziej stabilną dla dużej liczby iteracji (jakość nie pogarsza się dla kolejnych iteracji).
- Dla rozważanej trajektorii testowej nr 1a dobry dobór parametrów procesu pozwolił na ponad 60% wzrost jakości sygnału w porównaniu do filtracji LQ. Poprawa jakości sygnału zależy jednak od poziomu zaszumienia trajektorii. Dla bardzo dużego zaszumienia algorytm traci własności filtrujące i nie można uzyskać efektów lepszych niż algorytmem LQ.
- Algorytm NLQ można rozważać, jako kwaternionowy odpowiednik procesu *Perona-Malik* dla obrazów. Proces PM bierze pod uwagę estymację krawędzi, a algorytm NLQ estymację chwilowej prędkości kątowej. Jednakże w przypadku obrazów krawędź jest bardzo charakterystyczną cechą o dużym poziomie znaczeniowości. Dla trajektorii w przestrzeni rotacji odpowiadająca cecha w postaci zmiany prędkości kątowej nie niesie znacznej informacji o przebiegu ruchu. Może być pomocna jedynie w wybranych przypadkach.

#### 6.4 Skuteczność algorytmu *AnisoQuat* dla syntetycznych trajektorii testowych

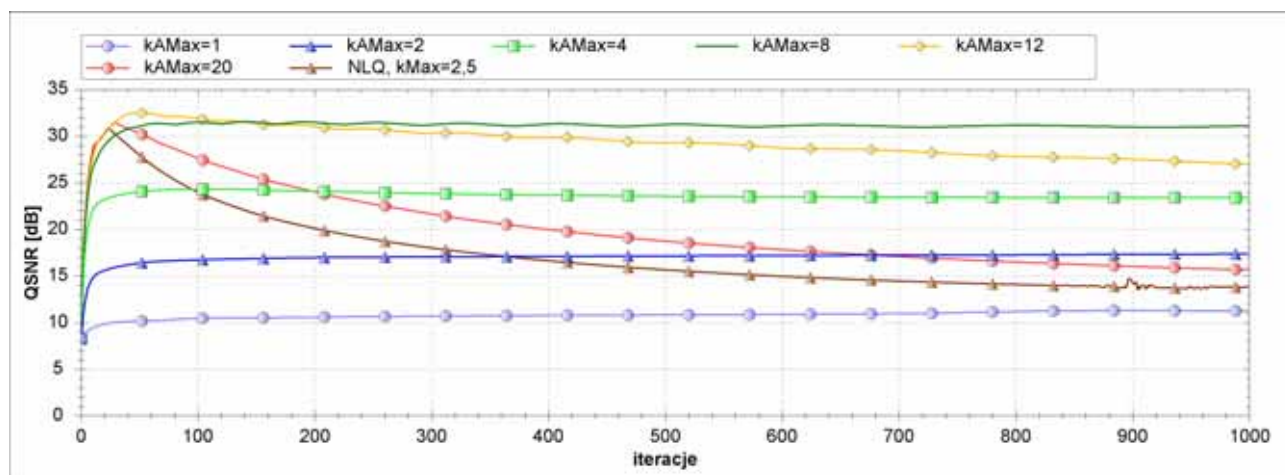
Algorytm AQ realizuje najbardziej zaawansowany sposób z przedstawionych metod przetwarzania trajektorii w przestrzeni rotacji. Zostaną zaprezentowane wyniki badań wykonanych na wybranych sekwencjach testowych, które pozwolą na ocenę własności algorytmu ze szczególnym uwzględnieniem zachowywania kształtu trajektorii. Trajektorja testowa nr 1a zostanie użyta w celu porównania działania algorytmów NLQ oraz AQ. Najintensywniej badanym sygnałem będzie trajektorja testowa nr 2, która zawiera kilkukrotną zmianę kształtu. Trajektorja testowa nr 3 zostanie przebadana w celu sprawdzenia działania algorytmu dla przypadku skomplikowanego i znacznie zróżnicowanego ruchu.

### 6.4.1 Skuteczność filtracji AQ w zależności od współczynników wrażliwości

Na podstawie wstępnych badań przedstawionych w poprzednim rozdziale, można wyciągnąć wniosek, że przebieg filtracji w największym stopniu zależy od przyjętego współczynnika wrażliwości dla funkcji stopującej. Współczynnik ten jest odpowiedzialny za zahamowanie lub przyspieszenie dyfuzji w zależności od położenia punktów występujących w przetwarzanej trajektorii.

#### Porównanie skuteczność filtracji AQ z wynikami działania algorytmu NLQ

Pierwsza grupa eksperymentów polega na porównaniu własności filtracji AQ z najlepszym otrzymanym wynikiem filtracji typu *NonLinQuat*. Filtracja będzie przeprowadzona dla trajektorii testowej nr 1a, ponieważ ona została przede wszystkim przebadana w przypadku algorytmu NLQ. Trajektorja została zaszumiona sygnałem o rozkładzie normalnym i odchyleniu standardowym  $\sigma=0,05$ . Dla obu algorytmów użyto warunku brzegowego typu D, parametru  $\lambda=0,5$  oraz funkcji stopującej  $g_I$  z adaptacyjnym sposobem wyznaczania współczynnika wrażliwości. Dla sekwencji referencyjnej NLQ przyjęto współczynnik  $K_{max}=2,5$ , dla którego filtracja przebiegała z najlepszą skutecznością. Taką samą wartość współczynnika przyjęto dla filtracji AQ w pierwszym z przeprowadzonych testów. Na rysunku 6-24 przedstawiono wykresy otrzymane dla procesu filtracji przy różnych wartościach anizotropowego współczynnika jakości  $K_{Amax}$ .

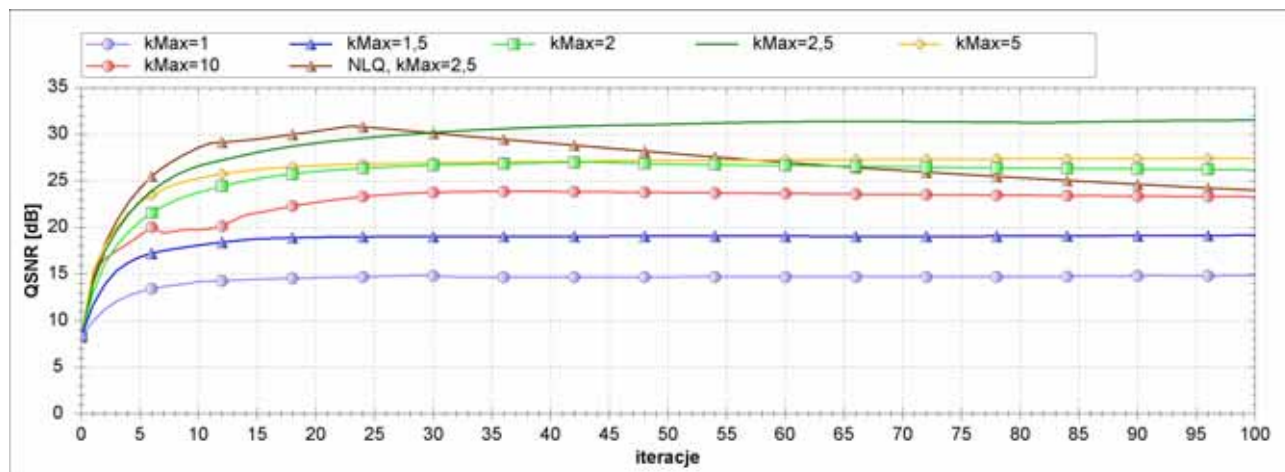


Rysunek 6-24: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od anizotropowego współczynnika wrażliwości  $K_{Amax}$  (funkcja  $g_I$ ) – syntetyczna trajektorja testowa nr 1a.

Podobnie, jak było w przypadku filtracji typu NLQ, również w tym przypadku zastosowanie współczynnika o zbyt małej wartości powoduje, że algorytm traci własności filtracyjne. Współczynnik, który wpływa pośrednio na szybkość filtracji jest zbyt mały, aby usunąć szum, który jest w takim przypadku traktowany, jako charakterystyczna cecha sygnału. Dopiero wzrost współczynnika do wartości 4, a następnie 8 powoduje, że otrzymuje się proces, który prowadzi do usunięcia szumów przy jednoczesnym zachowaniu własności sygnału, co prowadzi do uzyskania dużej wartości miary jakości. Dalszy wzrost tego współczynnika (wartości 12 i 20) powoduje, że przestaje on pełnić rolę ograniczenia i przebieg procesu zbliża się do sytuacji, w którym nie ma ograniczenia anizotropowego. Ze względu na równość współczynników wrażliwości  $K_{max}$  z porównywanym procesem NLQ, w przypadku zmniejszenia roli

współczynnika anizotropowego, przebieg obu procesów zaczyna być podobny i zbieżny. Świadczy o tym podobieństwo wykresów dla filtracji AQ (kolor czerwony) oraz NLQ (kolor brązowy).

Przeprowadzając filtrację dla ustalonej wartości anizotropowego współczynnika  $K_{Amax}=8$  w zależności od współczynnika  $K_{max}$  otrzymuje się zestaw wykresów pokazanych na rysunku 6-25. W tym przypadku rozważane jest pierwszych 100 iteracji, gdyż w tym przedziale występuje największa różnica w przebiegu filtracji NLQ oraz AQ. Pozostałe parametry filtracji pozostają bez zmian.



Rysunek 6-25: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od współczynnika wrażliwości  $K_{max}$  (funkcja  $g_l$ ) – syntetyczna trajektoria testowa nr 1a.

Podobnie, jak dla współczynnika anizotropowego oraz dla przypadku filtracji NLQ, zbyt mała wartość  $K_{max}$  powoduje, że filtracja nie może być zrealizowana z jakością lepszą od określonej wartości. Aż do wartości  $K_{max}=2$  jest to jakość znacznie zaniżona. Dopiero dla wartości  $K_{max}=2,5$  otrzymuje się proces nie gorszy od odpowiadającemu mu procesowi NLQ. Dalszy wzrost wartości tego współczynnika powoduje zmniejszenie jakości sygnału, gdyż filtracja zaczyna przejmować cechy procesu LQ. Powoduje to zbyt intensywną dyfuzję elementów, która prowadzi do zniekształcenia trajektorii.

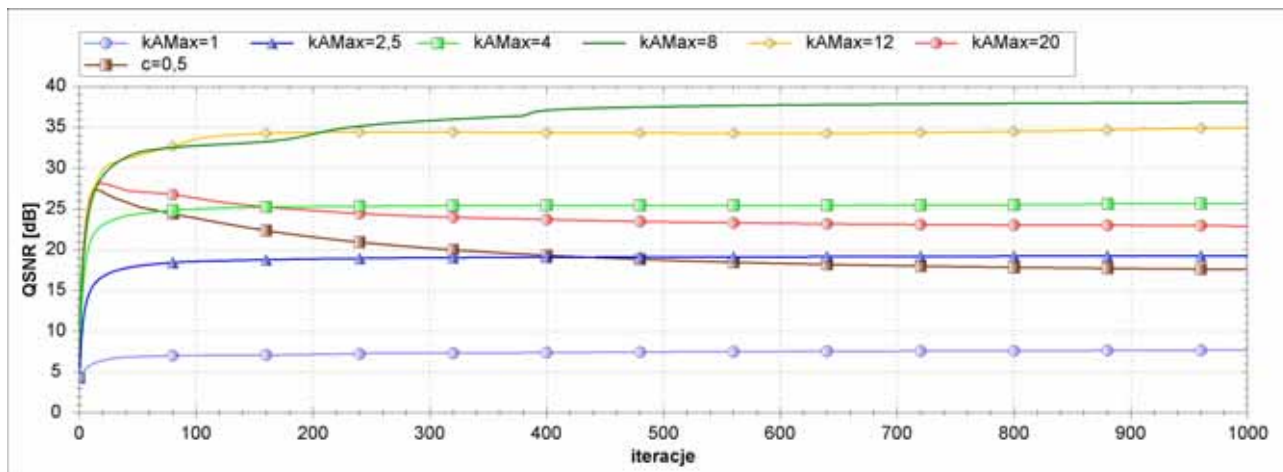
Najistotniejszym wnioskiem, który można wyciągnąć na podstawie przedstawionych wykresów jest to, że filtracja AQ oraz NLQ pozwalają na uzyskanie podobnej maksymalnej wartości miary jakości QSNR. Jednakże dla filtracji NLQ po uzyskaniu tej wartości po około 20 iteracjach, mimo wszystko rozpoczyna się proces nadmiernej dyfuzji, która prowadzi do zmniejszenia jakości sygnału. Algorytm AQ po osiągnięciu maksymalnej jakości sygnału, zachowuje jego własności dla kolejnych iteracji. Zachowanie takie wynika ze zmniejszenia dyfuzyjności anizotropowej ze względu na znaczną wartość krzywizny w charakterystycznym punkcie trajektorii.

Przedstawiony eksperyment pozwala rozpatrywać filtrację AQ, jako proces, który posiada cechy filtracji NLQ z dodatkowymi własnościami, które mają odtwarzać i zachowywać lokalną krzywiznę trajektorii.

### Badanie własności filtracji przy adaptacyjnym wyznaczaniu współczynników wrażliwości

Przeprowadzono badania filtracji dla bardziej złożonej trajektorii testowej, jaką jest trajektoria nr 2 (*QuatJumps*). Rozważany jest algorytm używający funkcję stopującą  $g_l$ , adaptacyjne wyznaczanie

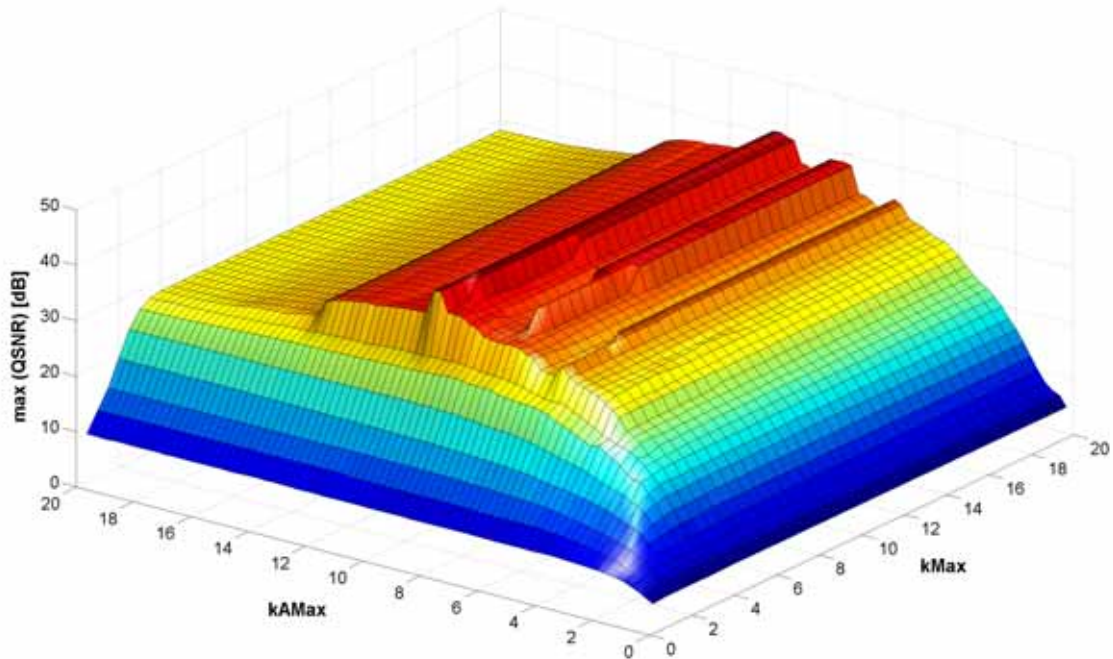
współczynników wrażliwości, parametry dyfuzji  $\lambda=0,5$  oraz  $\lambda_A=1,0$ . W pierwszym badaniu dla stałej wartości parametru  $K_{max}$  sprawdzono skuteczność filtracji w zależności od wartości anizotropowego współczynnika wrażliwości  $K_{Amax}$ . Otrzymane rezultaty pokazane zostały na rysunku 6-26. Dla porównania uwzględniono wykres miary jakości otrzymany dla algorytmu LQ przy takim samym współczynniku dyfuzji  $c=0,5$  (odpowiednik  $\lambda$ ).



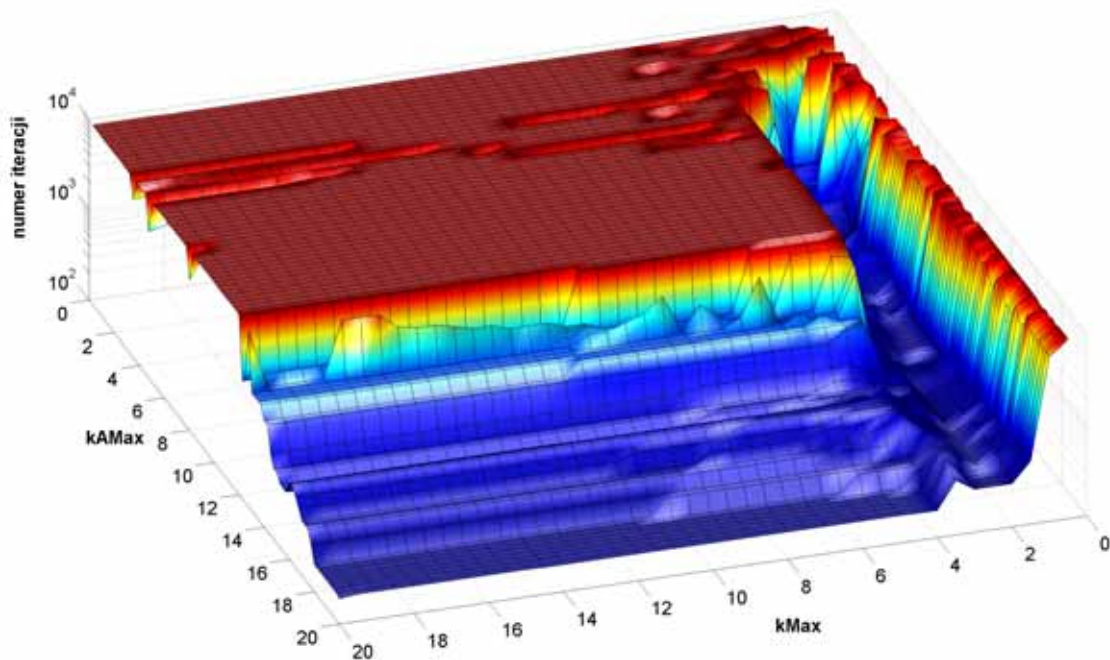
**Rysunek 6-26: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od współczynnika wrażliwości  $K_{Amax}$  (funkcja  $g_I$ ) – syntetyczna trajektoria testowa nr 2.**

Otrzymane rezultaty są podobne do uzyskanych w trakcie badania trajektorii testowej nr 1. Również w tym przypadku nadmierne zmniejszenie wartości anizotropowego współczynnika dyfuzji powoduje pozbawienia procesu własności filtracyjnych. Także zbyt duża wartość współczynnika powoduje, że proces przyjmuje przebieg podobny do przebiegu uzyskanego algorytmem LQ. Najlepsze rezultaty uzyskano dla współczynnika  $K_{Amax}=8$ . Filtracja w takim przypadku pozwoliła na uzyskanie sygnału o jakości wyższej o ponad 40% w porównaniu do filtracji typu LQ. Drugim równie ważnym rezultatem jest stabilność otrzymanych rezultatów. Wysoka wartość miary jakości utrzymuje się stabilnie przez bardzo długi czas (przetestowano maksymalnie 10 000 iteracji). Sygnał nie jest poddawany nadmiarowej dyfuzji, która w sposób niepożądany deformuje trajektorię w przypadku procesu LQ.

Wykonano wielokryterialny eksperyment polegający na znalezieniu maksymalnej wartości miary jakości sygnału uzyskanej dla filtracji trajektorii testowej nr 2 algorytmem AQ przy zmiennych parametrach  $K_{max}$ ,  $K_{Amax}$ . Pozostałe parametry procesu pozostają bez zmian. Dla każdego z parametrów wyznaczano maksymalną wartość miary QSNR dla zakresu 10 000 iteracji. Zebrane wyniki przedstawiono na zbiorczym wykresie na rysunku 6-27. Komplementarny wykres został pokazany na rysunku 6-28. Dla każdej pary parametrów filtracji AQ określono numer iteracji, przy której po raz pierwszy została osiągnięta maksymalna wartość miary QSNR (zaznaczonej na rysunku 6-27). Ze względu na dużą rozpiętość wartości w tym przypadku, na wykresie zastosowano skalę logarymiczną dla osi numerów iteracji. Na obu rysunkach wykresy zostały pokolorowane według automatycznie dobranej skali kolorów, gdzie kolor niebieski oznacza wartość minimalną, a kolor ciemnoczerowony wartość maksymalną na wykresie.



Rysunek 6-27: Maksymalna wartość miary jakości QSNR uzyskana w czasie 10 000 iteracji dla rezultatów filtracji AQ w zależności od współczynników  $K_{max}$ ,  $K_{Amax}$  (funkcja  $g_1$ ).



Rysunek 6-28: Numer iteracji, przy której osiągnięto maksymalną wartość miary jakości QSNR dla filtracji AQ w zależności od współczynników  $K_{max}$ ,  $K_{Amax}$  (funkcja  $g_1$ ).



Na podstawie rezultatów przeprowadzonego eksperymentu zbiorczego można wyciągnąć kilka bardzo wniosków, istotnych z punktu widzenia określenia istotności roli parametrów algorytmu. Przede wszystkim dla małych wartości obu parametrów występuje charakterystyczny spadek maksymalnej wartości uzyskiwanej miary QSNR. Dla tego zakresu parametrów proces jest nadmiernie hamowany, czego mechanizm został wyjaśniony w związku z rysunkiem 6-26. Jednakże z rysunku 6-27 można odczytać, że hamowanie procesu dla tego zakresu nie zależy od tego, który parametr przyjmuje małą wartość. Oznacza to, że oba parametry muszą być prawidłowo dobrane, aby żadna ze składowych filtracji (realizowana przez złożenie interpolacji) nie została zahamowana.

Dla wartości współczynnika  $K_{max}$  większych od określonej wartości granicznej (wynoszącej około 4,5) otrzymywane rezultaty przestają od niego zależeć. W celu interpretacji tego efektu należy przeanalizować przebieg rozpatrywanej trajektorii testowej nr 2. Rozważany sygnał ma w przybliżeniu stałą wartość modułu prędkości kątowej w każdym momencie czasowym. Oznacza to, że prędkość kątowa nie jest wyznacznikiem określającym charakter tego sygnału. Wzrost miary QSNR dla większych wartości  $K_{max}$  wynika z ograniczenia działania składowej nieliniowej biorącej pod uwagę prędkość kątową. Jeżeli prędkość ta jest stała, to zminimalizowanie tego ograniczenia będzie dawało lepsze efekty niż próba minimalizacji przy uwzględnieniu czynników nie mających wpływu na sygnał. Jest to efekt odblokowania szybkości filtracji przez przyjęcie dużej wartości współczynnika  $K_{max}$ . Otrzymany rezultat dotyczy specyfiki trajektorii testowej nr 2. Należy pamiętać, że analogiczne badanie dla trajektorii nr 1a (uwzględniającej zmianę prędkości kątowej) dało inne efekty. W praktyce jednak często zdarza się, że wartość prędkości kątowej jest stała w trakcie wykonywania ruchu.

Dla  $K_{max} > 4,5$  otrzymuje się natomiast istotną zależność otrzymywanych rezultatów od współczynnika  $K_{Amax}$ . Jest to inny sposób prezentacji danych, których wycinek był pokazany na rysunku 6-26. Bardzo dobrze w tym przypadku widoczne jest, że najwyższą wartość miary QSNR uzyskuje się dla współczynnika  $K_{Amax} = 8$ . Optymalna wartość współczynnika będzie jednym ze specyficznych parametrów dla określonego sygnału. Wynika ona z wartości krzywizny sygnału, dla której powinna zostać zahamowana dyfuzja anizotropowa. Przyjęcie większej wartości współczynnika powoduje, że proces zaczyna przyjmować charakter filtracji LQ, dla której nie występują ograniczenia i sterowanie jego przebiegiem wynikające z położenia punktów trajektorii (żółty obszar dla  $K_{Amax} > 12$ ).

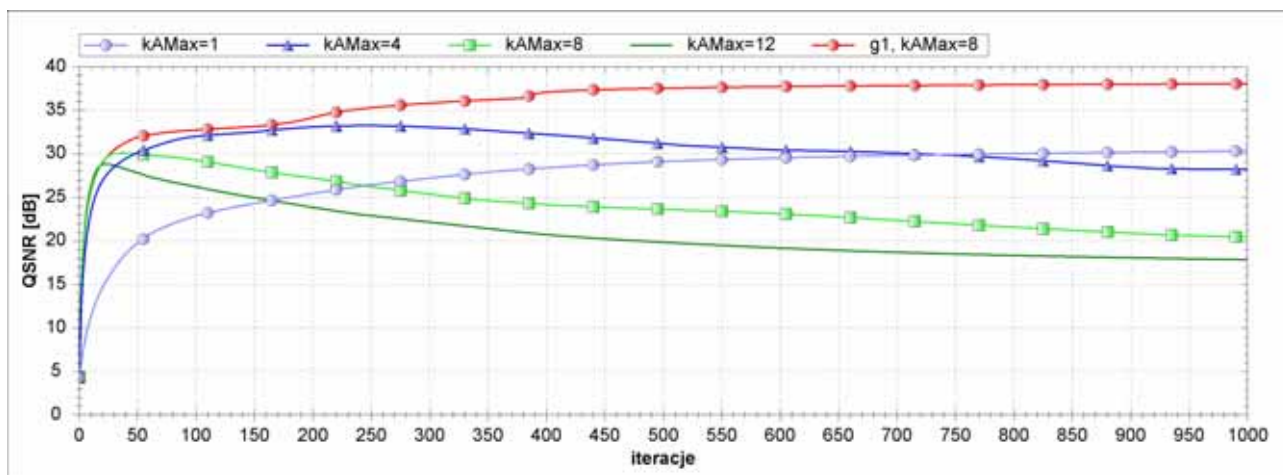
Interesujące może być zestawienie wyników przedstawionych na obu prezentowanych wykresach. Dzięki temu można wyciągnąć następujące dodatkowe wnioski o charakterze filtracji. Dla dużych wartości  $K_{Amax}$  maksymalna wartość miary QSNR jest uzyskiwana na początku filtracji (około 50-100 iteracji – niebieski kolor na wykresie). Oznacza to, że proces ten szybko osiąga swoje maksimum, a następnie pogarsza się jakość trajektorii. Jest to zbieżne z analizą wykresu 6-27 dla tego obszaru parametrów, dla którego określono, że proces w takim przypadku zaczyna nabierać cech filtracji LQ. Dla tego typu filtracji również szybko zostaje osiągnięta wartość maksymalna, a następnie następuje znaczny spadek jakości. Wynika to z nadmiernej dyfuzji, która po usunięciu zakłócenia, dodatkowo deformuje przetwarzany sygnał.

Dla małych wartości parametrów  $K_{Amax} < 10$  maksimum zostaje osiągnięte dopiero dla ostatnich rozpatrywanych iteracji (brane było pod uwagę 10 000 iteracji). Biorąc pod uwagę przebiegi z rysunku 6-26 można wyciągnąć wniosek, że miara QSNR ma tendencję niemalejącą po przekroczeniu pewnej wartości granicznej. Jakkolwiek krótkookresowa analiza wykazała, że miara ta otrzymuje się na stałym poziomie, to rozszerzone badania wskazują na możliwość minimalnego wzrostu po osiągnięciu tego poziomu. Teoretycznie również tego rodzaju algorytm powinien prowadzić w granicy do położenia, dla którego została zminimalizowana do 0 energia sygnału. Jednakże może to nastąpić dopiero po znacznie większej liczbie iteracji.

Z punktu widzenia praktycznego, duża stabilność procesu w funkcji iteracji jest bardzo istotna. Zastosowanie algorytmu w bardziej złożonym systemie przetwarzającym wymaga określenia warunku stopu dla przeprowadzanej filtracji. Dla filtracji LQ warunek taki musiałby uwzględniać dokładnie maksimum globalne miary QSNR osiągane po kilkudziesięciu pierwszych iteracjach. Filtracji otrzymywana przy pomocy algorytmu AQ jest pod tym względem znacznie bardziej bezpieczna. Można z góry założyć, że wykonywane jest około 500 iteracji, po których nie występuje znaczący wzrost wartości miary QSNR.

### Porównanie działania filtracji dla alternatywnej funkcji stopującej $g_2$

Zastosowanie alternatywnej funkcji stopującej  $g_2$  daje inną charakterystykę procesu filtracji. Przy użyciu tej funkcji stopującej przeprowadzono eksperymenty, w których użyto tych samych wartości parametrów, co w poprzednim przypadku:  $\lambda=0,5$ ,  $\lambda_A=1,0$ , poziom zaszumienia  $\sigma=0,05$ . Parametr  $K_{max}=20$ , co oznacza minimalizację tej składowej filtracji. Na wykresie dodatkowo zaznaczono krzywą reprezentującą rezultaty filtracji trajektorii dla wcześniej omawianej funkcji  $g_1$ . Wyniki przedstawione są na rysunku 6-29.



Rysunek 6-29: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od współczynnika wrażliwości  $K_{Amax}$  (funkcja  $g_2$ ) – syntetyczna trajektoria testowa nr 2.

Użycie funkcji  $g_2$  daje w rezultacie gorsze rezultaty z punktu widzenia uzyskiwanej miary jakości QSNR. Dla początkowych iteracji (około 150) otrzymuje się zbliżoną wartość QSNR, jak dla przypadku zastosowania funkcji  $g_1$ . Jednak w kolejnych fazach procesu, filtracja stosująca funkcję  $g_1$  dodatkowo poprawia jakość sygnału o około 5 dB, natomiast dla filtracji używającej funkcję  $g_2$  jakość zaczyna systematycznie się pogarszać. Również dla przebiegu długoterminowego nie zauważono, żeby występował

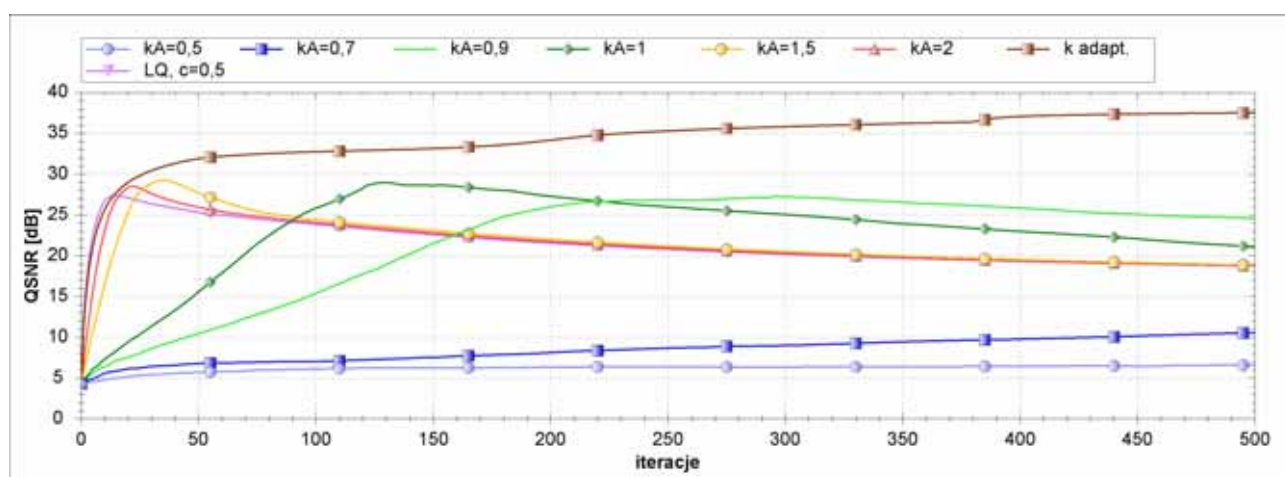
długotrwały efekt stabilizacji miary jakości, jaki występuje dla wielu parametrów procesu używającego funkcji  $g_1$ .

W trakcie eksperymentów zauważono, że funkcja  $g_2$  ma tendencję do nadmiernego wygładzania trajektorii. Jest to jednym z powodów spadku jakości trajektorii po przekroczeniu 200 iteracji. Analogiczne rezultaty można było zaobserwować dla procesu filtracji typu *Perona-Malik* przy użyciu tej samej funkcji stopującej. Również w tym przypadku występował podobny efekt – krawędzie na obrazie były bardziej wygładzane w trakcie procesu, jednak dzięki temu filtracja miała mniejszą tendencję do pozostawiania nieprawidłowości w postaci artefaktów.

Ostatecznie nie udało się uzyskać równie dobrych rezultatów filtracji przy użyciu alternatywnej funkcji stopującej. Jej użycie powoduje, że proces posiada charakter przejściowy pomiędzy filtracją AQ oraz LQ. Zauważono dodatkowo, że filtracja z funkcją  $g_2$  jest mniej wrażliwa na efekt całkowitego zatrzymania dyfuzji. Efekt taki występuje dla funkcji  $g_1$  przy za niskim współczynniku wrażliwości. Cecha ta może być pożyteczna dla filtracji przy znacznym zaszumieniu, dla którego trudno jest określić jednoznacznie stopnia zakłóceń oraz optymalnych wartości współczynnika wrażliwości  $K$ .

### Skuteczność filtracji przy stałej wartości współczynnika wrażliwości $K$

W poprzednich punktach w badanym procesie filtracji stosowany był adaptacyjny sposób wyznaczania współczynnika wrażliwości (przy określonym parametrze skalującym  $K_{max}$ ). Podstawowy schemat algorytmu bierze pod uwagę stałą wartość współczynnika. Zostanie sprawdzona skuteczność filtracji przy stałych wartościach  $K$ ,  $K_A$ . Nadal badana jest sekwencja testowa nr 2 przy poziomie zaszumienia  $\sigma=0,05$  oraz parametrach  $\lambda=0,5$ ,  $\lambda_A=1,0$ . W pierwszym badaniu ustalono wartość współczynnika  $K=1$ . Na rysunku 6-30 przedstawiono rezultaty filtracji przy zmiennej wartości  $K_A$ . Dodatkowo umieszczono wykresy otrzymane dla filtracji z adaptacyjnym sposobem wyznaczania współczynnika ( $K_{max}=20$ ,  $K_{Amax}=8$ ) oraz filtracji typu LQ z parametrem  $c=0,5$ .



Rysunek 6-30: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od współczynnika wrażliwości  $K_{Amax}$  (funkcja  $g_2$ ) – syntetyczna trajektoria testowa nr 2.

Filtracja przy stałym współczynniku  $K$  nie pozwala na osiągnięcie lepszych rezultatów niż w przypadku wersji adaptacyjnej. Najlepszy z otrzymanych rezultatów dla stałego współczynnika jest gorszy o

około 25% od najlepszego przypadku filtracji ze współczynnikiem adaptacyjnym. Dodatkowo dla współczynnika stałego występują dwa niepożądane efekty: szybkość poprawy jakości w pierwszej fazie filtracji jest niewielka, a po osiągnięciu maksimum wartości QSNR, jakość sygnału systematycznie się pogarsza.

Filtracja ze współczynnikiem adaptacyjnym jest szybsza w początkowej fazie ponieważ współczynnik ten jest w tym momencie względnie duży. Oznacza to, że dyfuzja przebiega z małymi ograniczeniami tłumiącymi jej szybkość. Stały współczynnik dla tego fragmentu procesu od początku w sposób jednostajny (ze względu na zawartość szumu w całej trajektorii) hamuje proces filtracji. Stąd gorszy przyrost jakości miary QSNR dla współczynnika stałego.

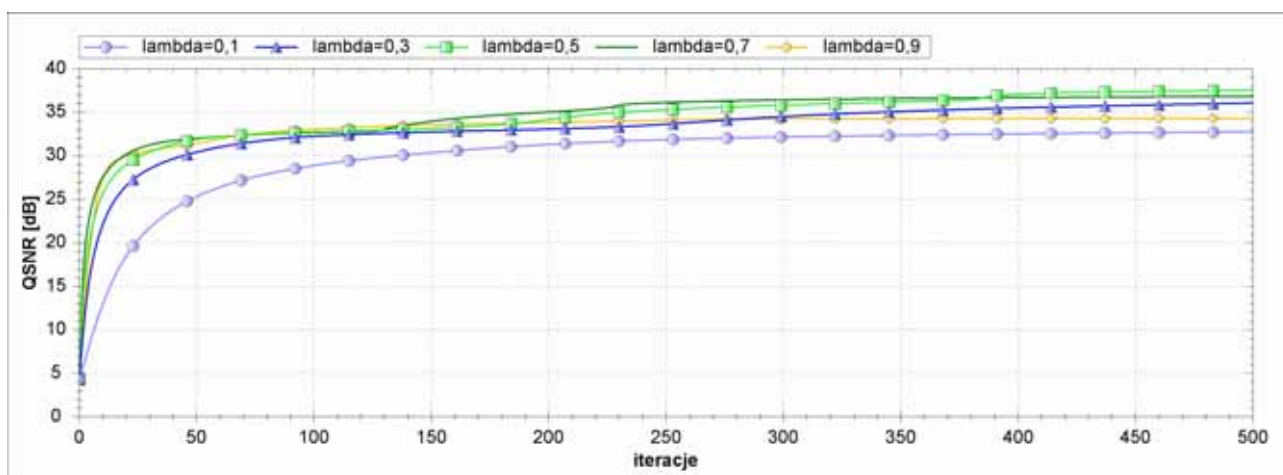
Po usunięciu znacznej części zakłóceń w początkowej fazie filtracji, wartość współczynnika  $K$  zostaje znacznie zmniejszona. Oznacza to, że dyfuzja w dalszej części procesu zachodzi bardzo powoli. Natomiast dla procesu ze stałym współczynnikiem dyfuzja nie jest dodatkowo hamowana przez globalne ograniczenie nałożone na prędkość dyfuzji. Nawet przy stałym współczynniku dyfuzji jest ona zależna od zaszumienia trajektorii, jednak tylko lokalnie – według modelu opisanego w rozdziale 5.4.3. Stąd ostatecznie filtracja ze stałym współczynnikiem daje lepsze rezultaty od algorytmu typu LQ, w przypadku którego prędkość dyfuzji nie jest w żaden sposób zależna od położenia punktów w trajektorii.

Kolejnym efektem, na który warto zwrócić uwagę jest podobieństwo do przebiegu filtracji typu LQ dla zwiększającej się wartości współczynnika  $K_A$ . Wynika to ze stopniowego zmniejszania udziału nieliniowej składowej ograniczającej prędkość dyfuzji, co ostatecznie prowadzi do całkowitego zaniku tej składowej sterowania procesem. Ze względu na dużą wartość współczynnika  $K$ , która ogranicza sterowanie dyfuzją zależną od prędkości, zminimalizowanie sterowania zależnego od krzywizny sprowadza proces do filtracji bez ograniczeń – podobną do przebiegu LQ. Warto jednak zwrócić uwagę na możliwość płynnej regulacji wpływu czynnika sterującego prędkością dyfuzji w zależności od przyjętego parametru  $K_A$ .

Zależność procesu od parametru  $K$  jest podobna, jak w przypadku liniowym. Ze względu na stałą prędkość kątową w badanej trajektorii przyjęta wartość  $K=1$  minimalizuje hamowanie dyfuzji przez tą składową. Przyjęcie mniejszych wartości, także dla funkcji  $g_2$  powoduje zmniejszenie skuteczności przeprowadzanej filtracji.

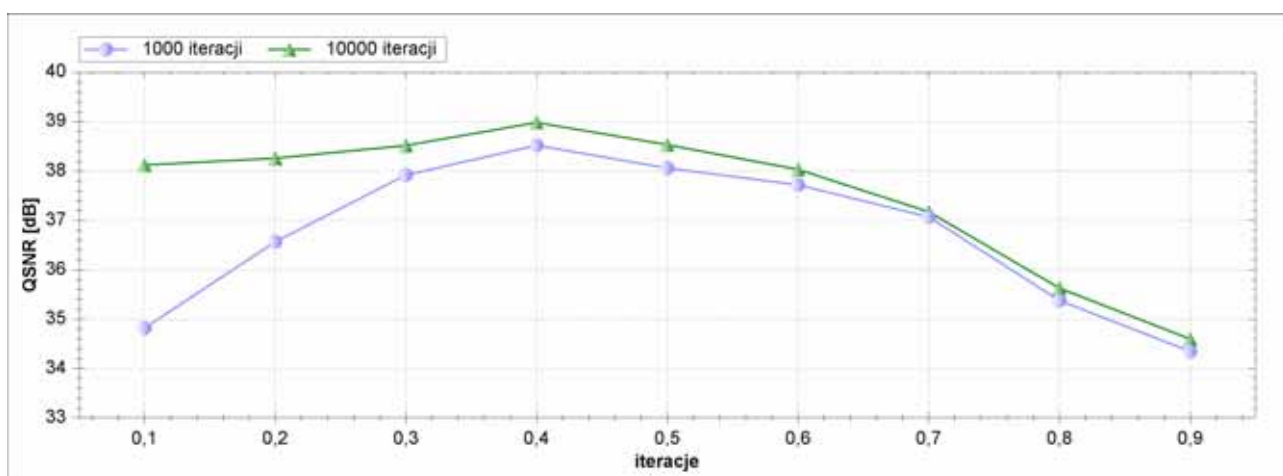
#### 6.4.2 Regulacja szybkości dyfuzji filtracji AQ

Badanie własności procesu w zależności od szybkości dyfuzji (określonej przy pomocy parametru  $\lambda$ ) przeprowadzono dla procesu dyfuzji o parametrach:  $\lambda_A=1,0$ ; funkcja stopująca  $g_1$ ; adaptacyjne wyznaczenie wrażliwości dla  $K_{max}=20$ ,  $K_{Amax}=8$ ; warunek brzegowy typu D; zaszumienie trajektorii sygnałem o rozkładzie normalnym  $\sigma=0,05$ . Na rysunku 6-31 przedstawiono otrzymane rezultaty filtracji trajektorii testowej nr 2 dla różnych wartości parametru  $\lambda$ .



Rysunek 6-31: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od współczynnika  $\lambda$  – syntetyczna trajektoria testowa nr 2.

Analogiczne badanie przeprowadzone dla procesu LQ wykazało znaczny wpływ właściwego doboru współczynnika dyfuzji. Analizując rysunek 6-31 można stwierdzić, że w tym przypadku wpływ tego parametru nie jest tak znaczący. Nawet dla skrajnych wartości parametrów otrzymuje się przebieg procesu o bardzo dobrych własnościach (za każdym razem znacznie lepszych od procesu typu LQ). Jednakże analiza długookresowa przebiegu procesu wykazuje, że jakość miary QSNR stabilizuje się na pewnym poziomie w zależności od przyjętego współczynnika  $\lambda$ . Na rysunku 6-32 przedstawiono maksymalną wartość QSNR, jaką uzyskano po odpowiednio 1000 i 10000 iteracjach w zależności od współczynnika  $\lambda$ .



Rysunek 6-32: Maksymalna wartość miary jakości QSNR uzyskana dla filtracji AQ przy wybranym parametrze  $\lambda$ .

Początkowy przebieg filtracji wydawał się pozwalać na uzyskanie podobnych rezultatów w znacznym stopniu niezależnych od przyjętego współczynnika  $\lambda$ . Dopiero na rysunku 6-32 widoczne jest, że nawet po 10 000 iteracjach występuje znaczna różnica pomiędzy jakością filtracji dla  $\lambda=0,4$  oraz filtracji dla  $\lambda=0,9$ . Optymalna filtracja jest więc realizowana dla współczynnika o wartościach około 0,4-0,5. Wynik ten jest podobny do przypadku algorytmu LQ.

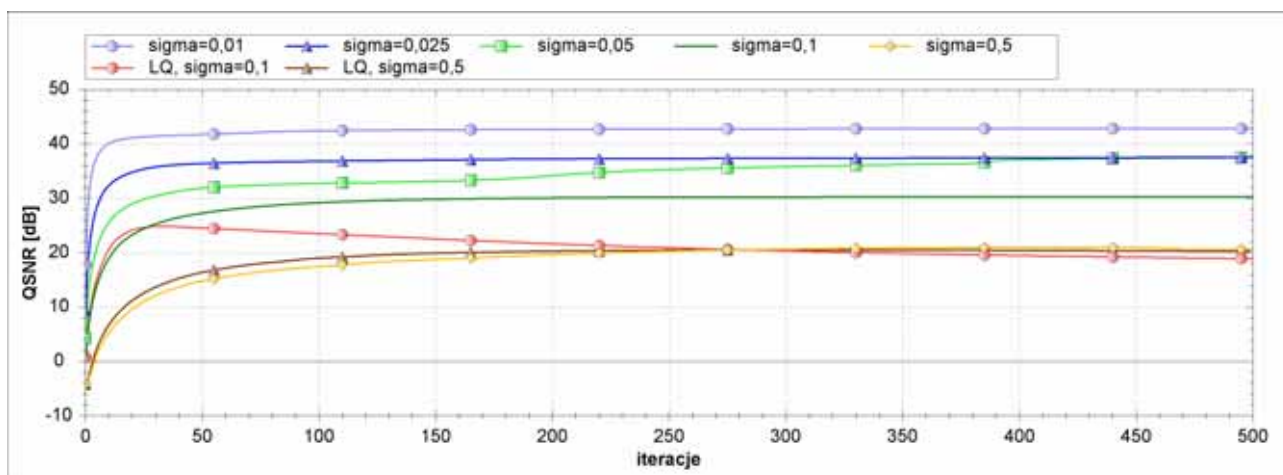
Można również zauważyć dwie nowe tendencje. Maksymalna jakość otrzymywana dla dużych współczynników  $\lambda$  jest osiągana względnie szybko. Występuje niewielka poprawa jakości dla 10 000 iteracji

w porównaniu do wartości QSNR otrzymanej już po 1 000 iteracji. W przypadku małych wartości  $\lambda$  rezultat jest inny. Jakość uzyskiwana na początku procesu jest znacznie niższa od tej, którą można uzyskać po wielu iteracjach. Stąd wniosek, że wolniej przeprowadzana filtracja daje długookresowo znacznie lepsze efekty. Po wielu iteracjach rezultaty otrzymywane dla  $\lambda \leq 0,5$  niewiele się od siebie różnią, a są znacznie lepsze od rezultatów otrzymanych dla dużej wartości współczynnika.

Podobnie, jak w przypadku algorytmu LQ, również w tym przypadku stosowanie parametru  $\lambda > 1$  powoduje niestabilność procesu. Ze względu na możliwość wystąpienia błędów numerycznych nie zaleca się także stosowania parametru  $\lambda = 1$ .

### 6.4.3 Rezultaty filtracji AQ w zależności od poziomu zaszumienia sygnału

Dla poprzednio stosowanych parametrów procesu AQ przeprowadzono eksperyment, w którym sprawdzono skuteczność filtracji w zależności od stopnia zaszumienia sygnału. Do trajektorii został dodany sygnał zakłócający o rozkładzie normalnym z różną wartością odchylenia standardowego  $\sigma$ . Otrzymane rezultaty zebrano i przedstawiono na rysunku 6-33. Dodatkowo dla porównania zamieszczono wykresy dla filtracji typu LQ przy dwóch poziomach znacznego zaszumienia ( $\sigma=0,1$ ,  $\sigma=0,5$ ).



**Rysunek 6-33: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od stopnia zaszumienia trajektorii – syntetyczna trajektoria testowa nr 2.**

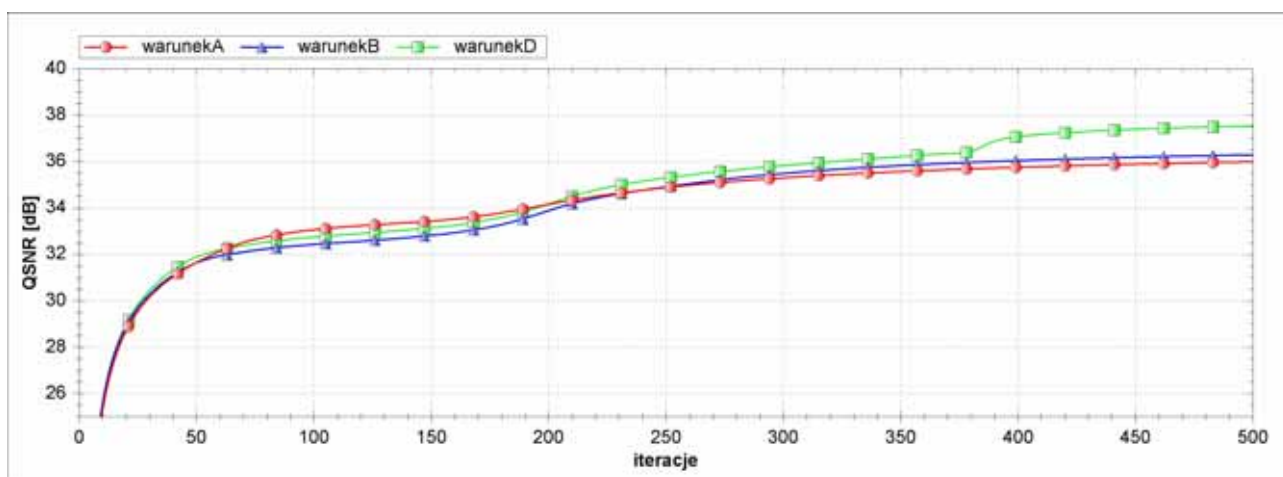
Analizując otrzymane rezultaty należy stwierdzić, że algorytm AQ zachowuje własności filtracyjne nawet dla trajektorii o bardzo dużym stopniu zawartości zakłóceń. Takiej własności nie posiadał wcześniej omawiany algorytm NLQ, który nie był w stanie w żadnym stopniu poprawić jakości sygnału, gdy poziom zakłóceń wzrósł ponad pewną wartość. Należy zwrócić uwagę na przebieg wykresu dla  $\sigma=0,5$  filtracji AQ, który jest zbliżony przez wiele iteracji do przebiegu dla filtracji LQ. Zbieżność wynika z przyjęcia adaptacyjnego sposobu wyznaczania współczynnika wrażliwości  $K$ . Dla rozważanego przypadku poziom zaszumienia sygnału jest na tyle duży, że wyznaczona wartość współczynnika  $K$  jest również wystarczająco wysoka, aby proces miał charakter zbliżony do typu LQ. Stąd wniosek, że kolejną zaletą adaptacyjnego sposobu określania  $K$  jest regularyzacja procesu w sytuacji bardzo znaczącego zniekształcenia sygnału.

Warto zauważyć, że dla względnie dużego stopnia zakłóceń  $\sigma=0,1$  proces AQ początkowo również przebiega współbieżnie z procesem LQ. Jednak po około 50 iteracjach następuje wyraźne rozróżnienie

przebiegu dla obu metod filtracji. Filtracja LQ powoduje zniekształcenie sygnału, natomiast adaptacyjnie prowadzony proces AQ stabilizuje się przez zmniejszenie wartości współczynnika wrażliwości  $K$ . Istotną cechą procesu jest to, że niezależnie od poziomu wprowadzonych zakłóceń, za każdym razem udaje się uzyskać stan ustalony, dla którego nie występuje zniekształcenie trajektorii przez przeprowadzenie nadmiernej dyfuzji. Jak to już została podkreślone wcześniej, jest stabilizacja miary QSNR jest ważna w zastosowaniu procesu we wszelkiego rodzaju systemach automatycznego odszumiania sygnałów. Pozwala to na określenie dobrego i skutecznego warunku zatrzymania filtracji także dla dużego stopnia zaszumienia sygnału.

#### 6.4.4 Wpływ przyjętego warunku brzegowego na przebieg filtracji

Ze względu na dodatkowe współczynniki wpływające na przebieg procesu AQ, inny niż dla algorytmu LQ, będzie również przebieg filtracji w zależności od przyjętego warunku brzegowego. Dla standardowych parametrów testowych ( $\lambda=0,5$ ,  $\lambda_A=1,0$ ,  $K_{max}=20$ ,  $K_{Amax}=8$ ,  $g_l$ ,  $\sigma=0,05$ , trajektoria testowa nr 2) przeprowadzono eksperyment dla różnego rodzaju warunków brzegowych. Interesujący fragment wykresu dla otrzymanych rezultatów został przedstawiony na rysunku 6-34. Na rysunku pominięto wykres dla warunku brzegowego typu C, który również w tym przypadku generuje znacznie mniejsze wartości miary QSNR ze względu na nieperiodyczność stosowanego sygnału testowego.



**Rysunek 6-34: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od wybranego warunku brzegowego – syntetyczna trajektoria testowa nr 2.**

W pierwszej fazie filtracji przedstawionej na rysunku (do około 200 iteracji) filtracja stosująca warunek brzegowy A uzyskuje nieznacznie lepsze rezultaty od pozostałych. Wynika to z faktu, że elementy krańcowe trajektorii pozostają nieruchome, przez co lokalna krzywizna sygnału staje się szybciej podobna do krzywizny sygnału oryginalnego. Najgorzej w tym zakresie wypada warunek brzegowy B. Jest to odmienny rezultat do odpowiadającemu mu wynikowi dla filtracji LQ (patrz rozdział 6.2.3).

W drugiej fazie filtracji (powyżej 200 iteracji) sytuacja się odменя. Po około 400 iteracjach jeszcze lepiej widoczne jest, że zastosowanie warunku brzegowego typu D daje znacząco lepsze rezultaty. Warunek brzegowy typu A powoduje, że to, co początkowo było zaletą (stałe umiejscowienie elementów brzegowych) zaczyna być składową wprowadzającą zniekształcenie do wyniku filtracji. Jeżeli elementy te były

początkowo zakłócone, to ich położenie różni się od położenia oryginalnego. Dodatkowo nie istnieje szansa, że położenie to zostanie skorygowane. Jediną składową pozytywnie wpływającą na wartość miary jakości dla obszarów skrajnych jest krzywizna trajektorii.

Dla warunku D sytuacja jest najbardziej korzystna. Wszystkie elementy trajektorii mogą zmieniać położenie. Dzięki temu położenie elementów brzegowych jest korygowane przez proces filtracji w pierwszej fazie procesu (w której występują znaczące wpływy procesu LQ). Następnie po zmniejszeniu stopnia zaszumienia sygnału, zmniejsza się także szybkość dyfuzji także dla elementów brzegowych. Należy pamiętać, że warunek D oznacza rozszerzenie sygnału na dziedzinę nieskończoną. Oznacza to, że dla elementów brzegowych występuje bardzo znacząca krzywizna, która zgodnie z zastosowanym modelem powoduje zmniejszenie dyfuzyjności w tym obszarze. Dzięki temu początkowo skorygowane położenie elementów, nie jest następnie znacząco zmieniane. W efekcie otrzymuje się proces, który również dla elementów brzegowych dąży do stanu ustalonego, w którym prędkość dyfuzji jest zminimalizowana. Pozwala to na osiągnięcie stabilnej wartości miary QSNR, która utrzymuje się dla bardzo wielu iteracji (co zostało pokazane w poprzednim paragrafie – do 10 000 kroków algorytmu).

Interesująca jest charakterystyka otrzymana dla warunku brzegowego typu B. Warunek ten oznacza układ izolowany. W związku z tym otrzymuje się sytuację, w której elementy mogą być cały czas modyfikowane przez dyfuzję, co zmniejsza jakość sygnału w drugiej fazie przedstawionej filtracji.

Podsumowując należy stwierdzić, że warunek brzegowy typu D jest najbardziej uniwersalnym rozwiązaniem dla filtracji nieperiodycznych trajektorii. W powiązaniu z własnościami algorytmu AQ pozwala on na przeprowadzenie filtracji także dla elementów brzegowych przy jednoczesnym zahamowaniu dyfuzji po zmniejszeniu stopnia zaszumienia trajektorii.

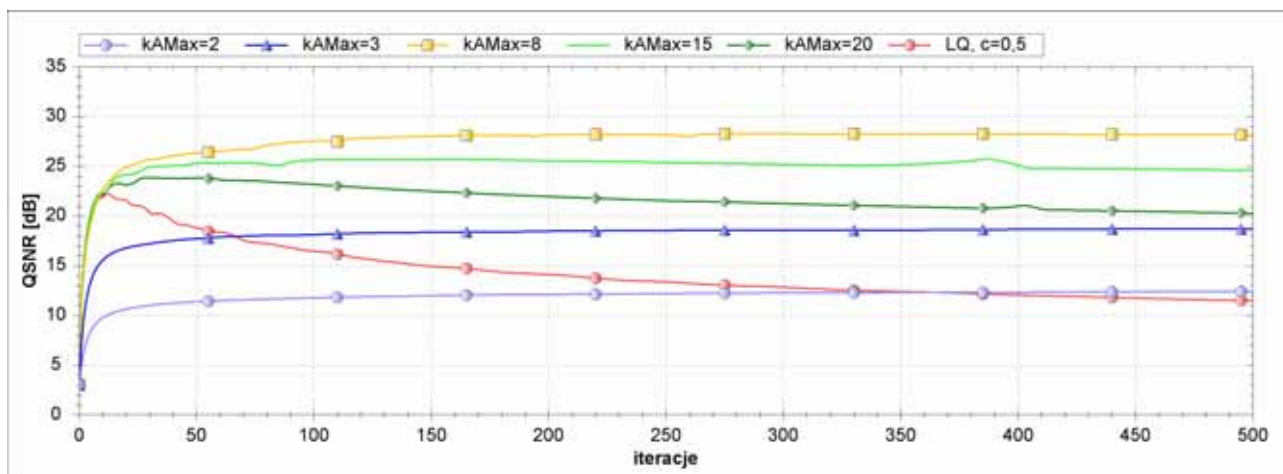
#### **6.4.5 Filtracja AQ dla skomplikowanej trajektorii testowej typu *QuatMulti***

Trajektoria testowa nr 3 przedstawia złożony sygnał, w którym występuje zróżnicowanie w każdym obszarze charakteryzującym przebieg: położenia punktów, chwilowej prędkości kątowej oraz lokalnej krzywizny trajektorii. Na rysunku 6-35 przedstawiono rezultaty filtracji tej trajektorii dla: funkcji stopującej  $g_I$ ,  $K_{max}=20$ ,  $\lambda=0,5$ ,  $\lambda_A=1,0$ ,  $\sigma=0,05$ .

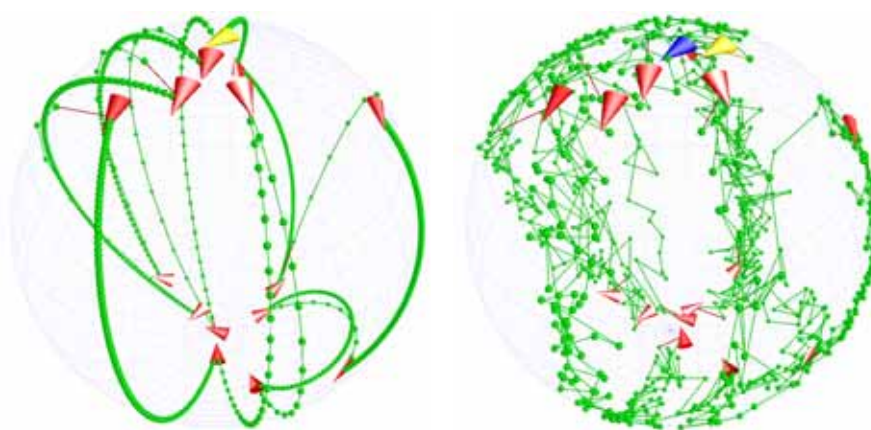
Otrzymane rezultaty pokazują, że własności filtracji nie zmieniły się znacząco dla wzrostu złożoności sygnału. Względna poprawa możliwej do uzyskania jakości sygnału jest tym przypadku mniejsza niż dla trajektorii nr 2. Niemniej jednak uzyskane rezultaty również wykazują tendencję do stabilizowania wartości miary QSNR przy odpowiednio dobranym parametrze  $K$ .

Przeprowadzany proces jest bardziej wrażliwy na dobór współczynnika wrażliwości. Stosowanie współczynnika o stałej wartości byłoby jeszcze trudniejsze w praktyce, niż to się okazało dla trajektorii nr 2. Należy zwrócić uwagę na znaczne zróżnicowanie sygnału testowego, który na każdym odcinku wykazuje inny charakter przebiegu. Oryginalny sygnał testowy oraz przykładowa zaszumiona postać zostały pokazane na rysunku 6-36. Zastosowano sygnał losowy o rozkładzie normalnym  $\sigma=0,05$ . Na rysunku wyraźnie widoczny jest bardzo znaczny stopień zniekształcenia trajektorii oryginalnej.





Rysunek 6-35: Miara jakości QSNR dla rezultatów filtracji AQ w zależności od współczynnika wrażliwości  $K_{Amax}$  – syntetyczna trajektoria testowa nr 3.



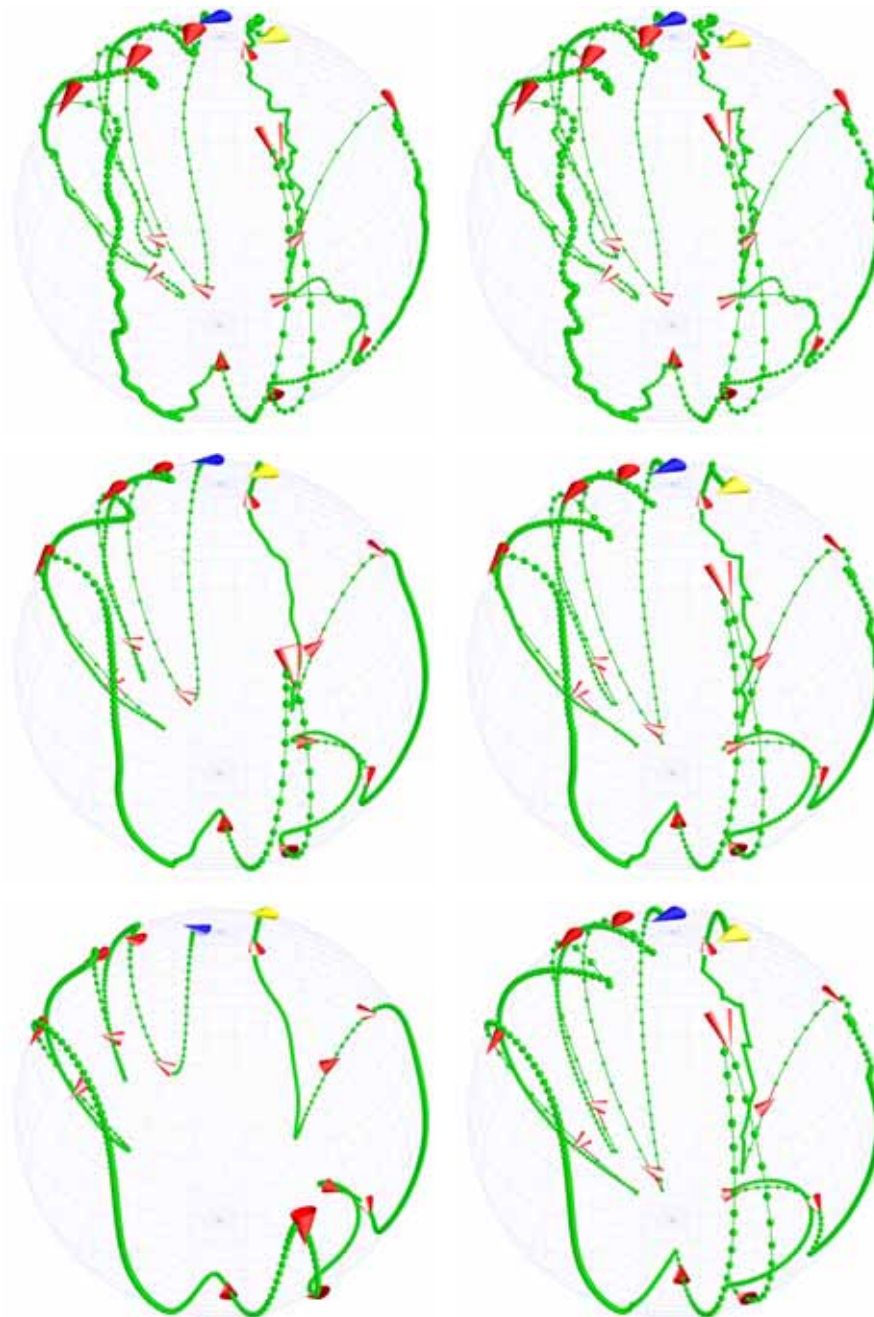
Rysunek 6-36: Oryginalna oraz zaszumiona trajektoria testowa nr 3. Szum o rozkładzie normalnym,  $\sigma=0,05$ .

Rysunek 6-37 pokazuje wybrane iteracje procesów filtracji LQ oraz AQ. Analizując rysunek można porównać procesy, które zachodzą w kolejnych chwilach iteracji. W pierwszej fazie obie metody generują podobne rozwiązanie (pierwszy wiersz rysunku). Następnie zaczyna występować widoczne zróżnicowanie w działaniu.

Na podstawie rezultatów dla 50 i 200 iteracji widoczne jest, że proces LQ powoduje usuwanie zniekształceń przez wygładzanie trajektorii. Szczególnie dla 200 iteracji widać, że proces dąży do ujednoczenia położenia i prędkości elementów. Powoduje to jednak zniekształcenie niektórych fragmentów sygnału przez nadmierną dyfuzję, która jest realizowana.

Odpowiedni proces AQ jest realizowany inaczej. Dla większości punktów, w którym zmieniany jest kształt trajektorii udało się zachować wyraźną różnicę tej zmiany. Jednakże zastosowanie wartości współczynnika wrażliwości stałej dla całego przebiegu powoduje, że w niektórych jego fragmentach otrzymuje się właściwie przeprowadzone filtrowanie, a na innych elementy trajektorii są dodatkowo zniekształcone. Dobrze widoczne to jest dla 200 iteracji procesu AQ na ostatnim odcinku trajektorii (przed klatką kluczową zaznaczoną kolorem żółtym). Szum, który występował dla tego odcinka został w znacznym

stopniu usunięty przez algorytm LQ. Niestety zastosowanie nieliniowego sterowania filtracji spowodowało, że dla tego fragmentu szum ten został wyróżniony w postaci schodkowego przebiegu trajektorii wynikowej. Jest to odpowiednik artefaktów, które pozostawały na obrazie dla procesu filtracji typu *Perona-Malik*.



**Rysunek 6-37: Wizualizacja kolejnych kroków filtracji LQ (lewa kolumna) oraz AQ (prawa kolumna). W wierszach pokazana odpowiednio: 10, 50, 200 iterację.**

Przedstawiony problem dotyczyć będzie głównie skomplikowanych, zróżnicowanych sekwencji. W takim przypadku trudno jest dobrać właściwe parametry  $K$ , które pozwoliłyby na jednoczesne usunięcie składowej zaszumiającej i prawidłowe zachowanie kształtu przetwarzanej trajektorii.

Ostatecznie nawet uwzględniając artefakty wprowadzone przez proces AQ, otrzymane rezultaty wykazują około prawie 30% poprawę w odniesieniu do filtracji LQ. Także w tym przypadku występuje efekt stabilizacji wartości normy QSNR dla bardzo wielu iteracji. Kolejne przebiegi filtracji AQ zostaną szczegółowo opisane i zobrazowane w następnym rozdziale.

#### 6.4.6 Porównanie własności filtracji algorytmami LinQuat oraz AnisoQuat

Ze względu na swoją zbieżność z innymi metodami przetwarzania trajektorii w przestrzeniach rotacji algorytm LQ jest filtracją referencyjną. Z punktu widzenia własności wynikających z równań różniczkowych cząstkowych, algorytm ten ma wysoką zdolność usuwania zakłóceń o rozkładzie normalnym. Częściowo jego własności były już porównywane z algorytmami filtracji proponowanymi w tej pracy. W tym rozdziale wyniki porównania zostaną zebrane i podsumowane.

W tabeli 4 zebrano najlepsze wartości miary QSNR uzyskane odpowiednio dla algorytmów AQ oraz LQ. W trzeciej kolumnie umieszczono procentową poprawę, jaką udało się uzyskać stosując algorytm AQ w odniesieniu do najlepszego wyniku filtracji LQ. Badania przeprowadzono dla wszystkich czterech syntetycznych sekwencji testowych.

	LQ	AQ	poprawa
<b>Test01a</b>	19,07	31,55	165,4%
<b>Test01</b>	28,63	41,16	143,8%
<b>Test02</b>	27,35	38,50	140,8%
<b>Test03</b>	22,25	28,28	127,1%

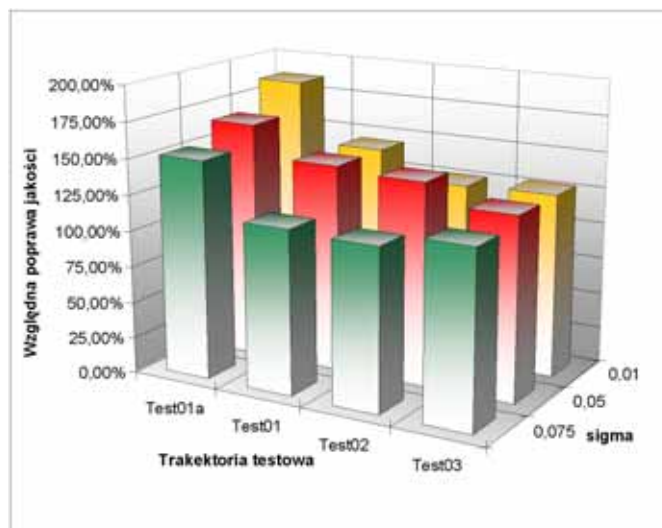
**Tabela 4: Maksymalna wartość miary QSNR [dB] uzyskana w ciągu 500 iteracji dla filtracji LQ oraz AQ. Sygnał zakłócający o rozkładzie normalnym,  $\sigma=0,05$ .**

Wyniki filtracji LQ dla trajektorii testowej nr 1a są znacznie niższe od pozostałych ze względu na specyficzną konstrukcję tego sygnału testowego. W rozdziale 6.3.1 pokazano, że sygnał jest trudny dla algorytmu LQ ze względu na znaczący skok położenia występujący w połowie sekwencji. Sygnał ten jest syntetyczny i został skonstruowany celowo w ten sposób, aby podkreślić i wyróżnić własności badanych filtrów.

Dla pozostałych filtrów otrzymywane wartości miary jakości sygnału są podobne. Zastosowanie algorytmu AQ pozwala uzyskać wyniki od 27 do 65 % lepsze od rezultatów filtracji LQ. Własności filtracyjne proponowanego podejścia są słabsze dla bardziej skomplikowanych sekwencji (nr 2 oraz nr 3). Tendencja ta jest jeszcze lepiej widoczna przy analizie tabeli 5. W tabeli zostały zebrane wyniki poprawy filtracji (rozumianej, jako największego możliwego zysku na jakości sygnału) dla sekwencji testowych w zależności od stosowanego stopnia zaszumienia. Wyniki zostały zobrazowane na wykresie pokazanym na rysunku 6-38.

	$\sigma=0,01$	$\sigma=0,05$	$\sigma=0,075$
<b>Test01a</b>	185,7%	165,4%	151,7%
<b>Test01</b>	143,5%	143,8%	113,8%
<b>Test02</b>	125,4%	140,8%	112,0%
<b>Test03</b>	127,4%	127,1%	121,0%

**Tabela 5: Poprawa jakości sygnału dla maksymalnej wartości QSNR algorytmu AQ względem maksymalnej wartości uzyskanej filtracją LQ. Sygnał zakłócający o rozkładzie normalnym.**



**Rysunek 6-38: Poprawa jakości sygnału dla maksymalnej wartości QSNR algorytmu AQ względem maksymalnej wartości uzyskanej filtracją LQ.**

Otrzymane rezultaty w większości pokrywają się z wcześniej otrzymaną zależnością dla różnego stopnia zaszumienia przetwarzanego sygnału. Największą poprawę uzyskuje się dla najmniej skomplikowanych trajektorii. W przypadku małego zaszumienia ( $\sigma=0,01$ ) uzyskiwana poprawa osiągała prawie 90 % względem filtracji LQ. W takich przypadkach filtracja LQ musi być przeprowadzana z małą prędkością dyfuzji, aby nie zniekształcić sygnału, w którym zawartość zakłóceń jest niewielka.

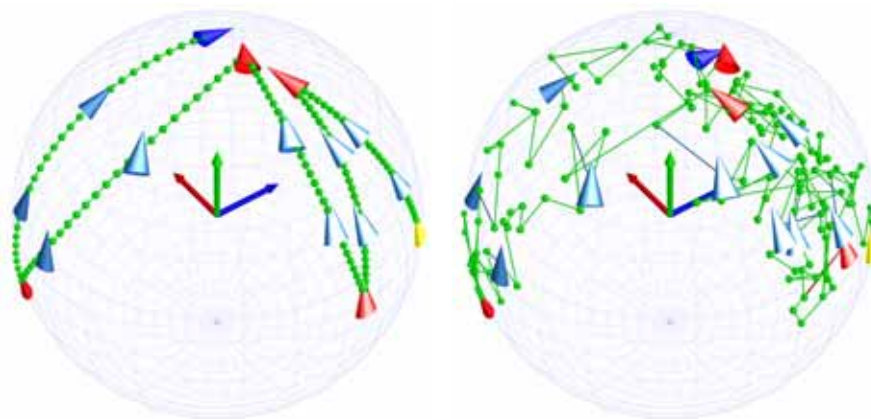
Interesujący rezultat otrzymuje się dla trajektorii testowej nr 3. W tym przypadku niezależnie od stopnia zaszumienia sygnału otrzymuje się podobną procentową poprawę możliwej do uzyskania jakości po zastosowaniu filtracji AQ. Dla małego poziomu zaszumienia obie metody filtracji szybko są w stanie usunąć zakłócenia w pierwszej fazie procesu. Otrzymuje się wysoką wartość QSNR dla filtracji LQ, natomiast artefakty wprowadzone przez filtrację AQ nie są znaczące dla tego poziomu zakłóceń.

Dla dużego zaszumienia proces typu LQ jest w stanie usunąć w znacznym stopniu zakłócenia przez wygładzanie trajektorii. Jednakże w momencie, w którym zakłócenia zostają usunięte, trajektoria jest jednocześnie znacznie zniekształcona przez przeprowadzoną dyfuzję. W przypadku procesu AQ duży stopień zakłóceń powoduje pojawianie się większej liczby artefaktów. Jednakże z otrzymanych rezultatów wynika, że występowanie tych artefaktów nie deformuje filtrowanego sygnału bardziej niż zniekształcenie wynikające z nadmiernej dyfuzji typu LQ. Ostatecznie w granicach zaszumienia  $\sigma \leq 0,075$  proces AQ pozwala na uzyskanie 20% poprawy w odniesieniu do analogicznej filtracji LQ. Dla wyższych poziomów zakłóceń wartość ta się znacznie zmniejsza, co zostało pokazane na wykresie 6-33 dla trajektorii testowej nr 2 (*QuatJumps*).

Wyniki przedstawione w tabeli 5 obrazują maksymalną możliwą poprawę, jaką można uzyskać stosując porównywane procesy filtracji. Jednak należy pamiętać, że proces AQ dąży do stabilizacji trajektorii w przeciwieństwie do modelu LQ, który po osiągnięciu maksimum wartości QSNR znacznie pogarsza jakość sygnału. Dla iteracji po osiągnięciu tego maksimu, chwilowa poprawa jakości dla obu procesów wyrażałaby się jeszcze większą wartością.

## Wizualizacja przebiegu procesu filtracji

W celu czytelnego przedstawienia własności procesu zostaną przedstawione wybrane iteracje dla filtracji AQ ze standardowymi parametrami ( $\lambda=0,5$ ,  $\lambda_A=1,0$ ,  $K_{max}=20$ ,  $K_{Amax}=8$ ,  $g_l$ ,  $\sigma=0,05$ , trajektoria testowa nr 2). Na rysunku 6-39 przedstawiono oryginalną trajektorię testową nr 2 oraz trajektorię zaszumioną sygnałem o rozkładzie normalnym,  $\sigma=0,05$ . Zastosowanie wizualizacji przy pomocy sfery rotacji pozwala zobaczyć, że jest to znaczny poziom zaszumienia, który dla obserwatora bardzo intensywnie deformuje trajektorię.

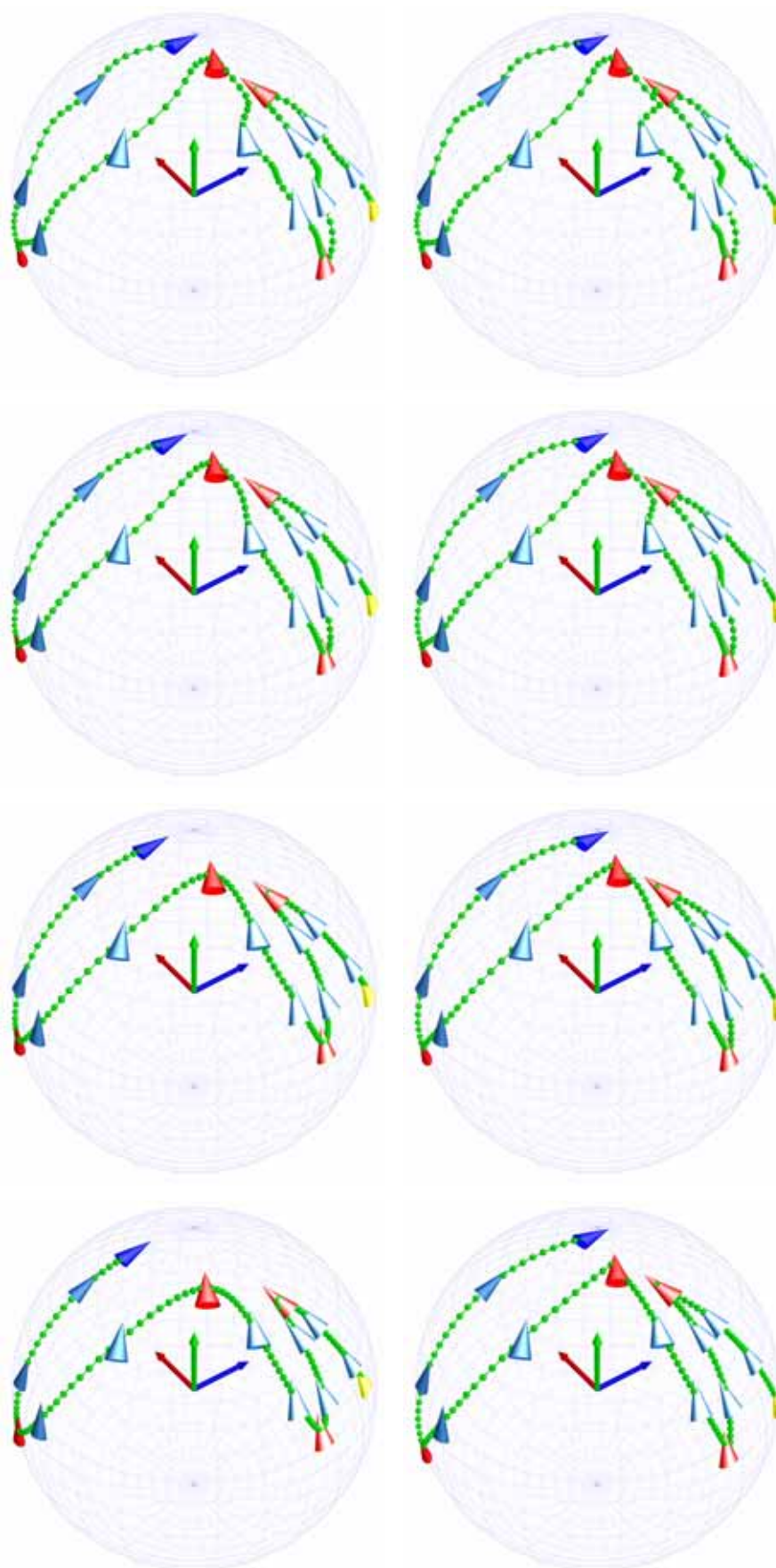


**Rysunek 6-39: Oryginalna oraz zaszumiona trajektoria testowa nr 2.**  
Trajektoria zaszumiona – rozkład normalny,  $\sigma=0,05$ .

W kolejnych wierszach rysunki 6-40 pokazane są wybrane iteracje (10, 20, 50, 100) dla porównywanych procesów. Widoczne jest, że w początkowej fazie filtracji proces LQ szybciej usuwa zakłócenia przez znaczne wygładzanie trajektorii (patrz iteracja nr 10, 20). Jest to zgodne z otrzymanymi poprzednio rezultatami, w których miara jakości QSNR była początkowo wyższa dla filtracji LQ.

Dla iteracji 20 zaczyna być dobrze widoczny, a w iteracji 50 zostaje podkreślony efekt wygładzania sygnału uzyskiwany w procesie LQ. Odznacza się to zmianą krzywizny trajektorii w charakterystycznych punktach trajektorii – krzywizna jest w tym przypadku minimalizowana. Po 100 iteracjach wygładzanie staje się intensywne, co objawia się zarówno w wygładzaniu punktów charakterystycznych trajektorii, jak i w nadmiernym przemieszczeniu elementów brzegowych. Wynika to z zastosowanego warunku brzegowego typu D, który nie wnosi żadnego ograniczenia do procesu typu LQ.

Dla procesu AQ dopiero po 50 iteracjach widoczny jest efekt usunięcia sygnału zakłócającego. Jednak w przeciwieństwie do filtracji LQ w tym przypadku nie następuje dalszy proces dyfuzji we fragmentach, w których występują charakterystyczne punkty o znacznej zmianie krzywizny. Dzięki temu sygnał nie jest zniekształcany, co prowadzi do stanu pokazanego dla iteracji nr 100. Stan ten jest następnie utrzymywany przez wiele iteracji, co zostało pokazane w poprzednich eksperymentach (aż do iteracji 10 000).

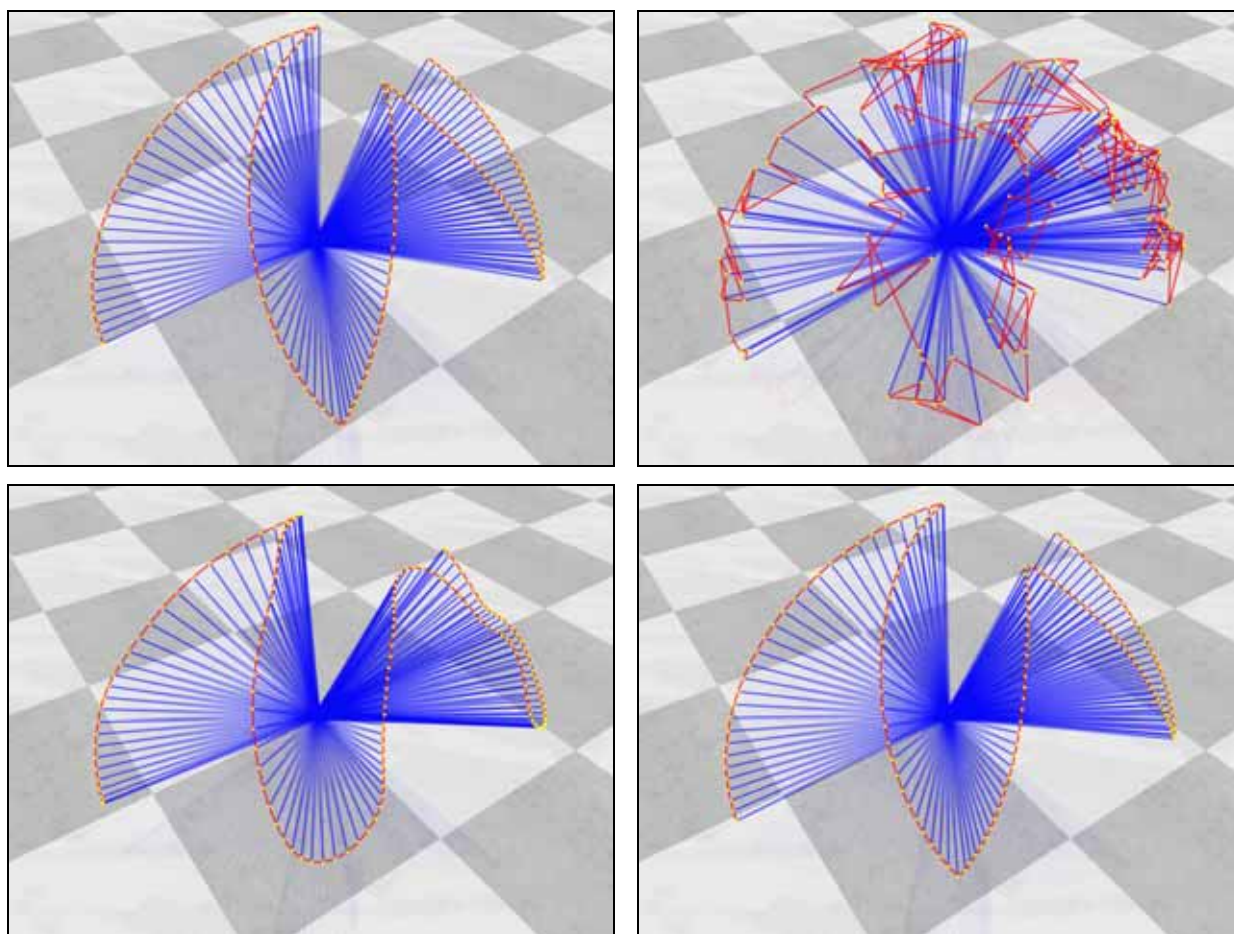


Rysunek 6-40: Wizualizacja kolejnych kroków filtracji LQ (lewa kolumna) oraz AQ (prawa kolumna). W wierszach pokazana odpowiednio iteracje nr: 10, 20, 50, 100.

Dodatkowym pozytywnym efektem jest ustabilizowanie pozycji elementów brzegowych trajektorii. Zastosowanie warunku brzegowego typu D pozwala w przypadku algorytmu AQ zminimalizować dyfuzję dla elementów początkowego i końcowego po zmniejszeniu stopnia zaszumienia sygnału. Jest to efekt, który pozwolił na zwiększenie jakości sygnału w porównaniu z innymi warunkami brzegowymi (patrz rysunek 6-34).

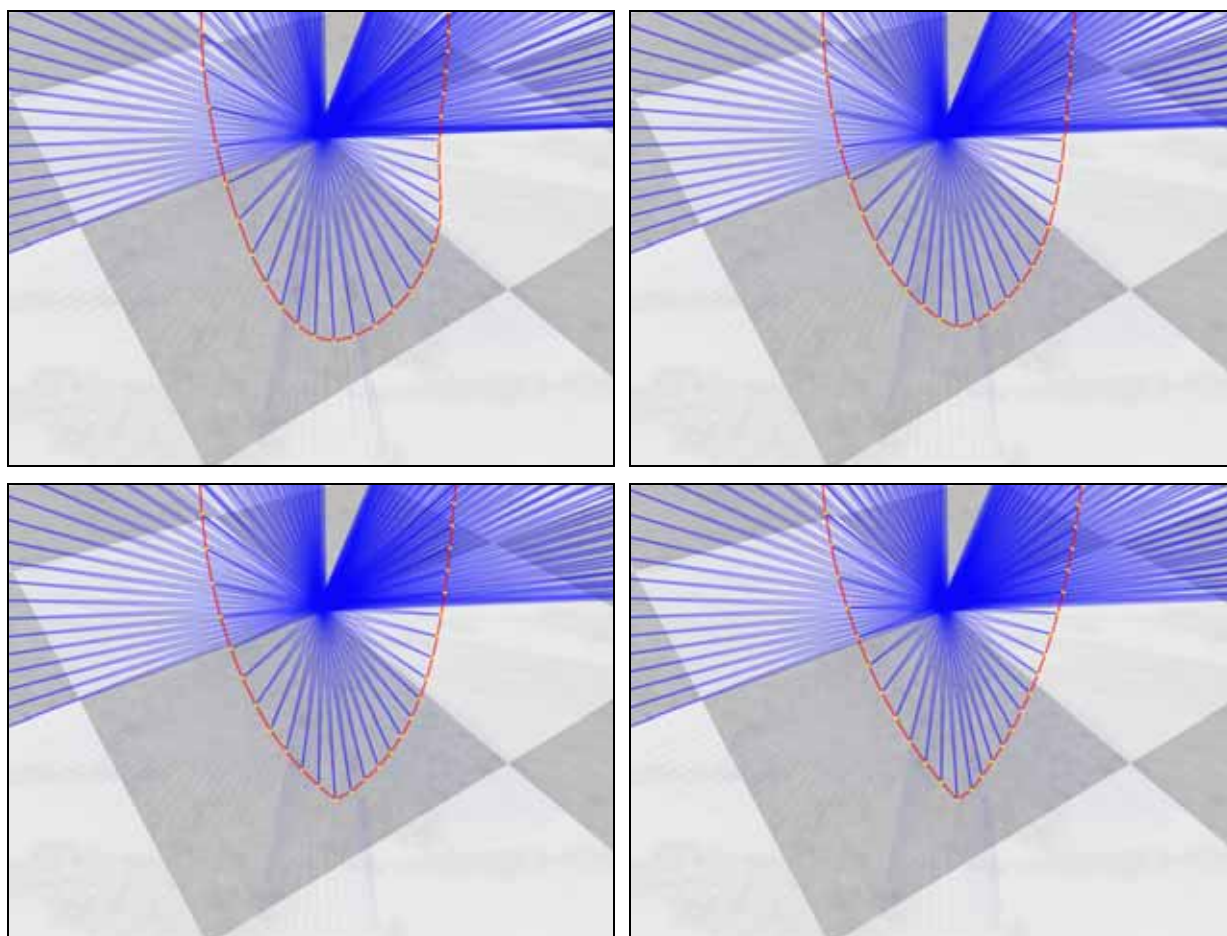
Najlepsze rezultaty dla filtracji LQ uzyskuje się po około 20 iteracjach. Stabilny stan trajektorii w przypadku trajektorii AQ otrzymany został po około 200 iteracjach. Na rysunku 6-41 pokazano wizualizację w przestrzeni 3D trajektorii dla iteracji, w których otrzymano najlepsze rezultaty filtracji. Dobrze widoczny jest efekt wygładzenia trajektorii przy użyciu filtracji LQ, który występuje już na początku procesu. Filtracja ta nie posiada możliwości elastycznej regulacji przebiegu procesu, więc nie ma możliwości usunięcia składowej zakłócającej bez przeprowadzenia dyfuzji, która jednocześnie nadmiernie wygładza trajektorię. Jest to widoczna analogia do efektów filtracji liniowej dla obrazów. W przypadku obrazów również otrzymywało się usunięcie zniekształceń przy jednoczesnym rozmyciu charakterystycznych cech, jakimi są krawędzie.

Dodatkowo widać przemieszczenie elementów brzegowych, jakie powstało w wyniku filtracji typu LQ. Odległość pomiędzy punktami w obszarach brzegowych zaczyna się zmniejszać ze względu na własności filtracji, która dąży do minimalizacji całkowitej odległości dla wszystkich punktów.



Rysunek 6-41: Wizualizacja trajektorii testowej nr 2. Lewa góra: tr. oryginalna. Prawa góra: tr. zaszumiona ( $\sigma=0,05$ ). Lewy dół: filtracja LQ, 20 iteracji. Prawy dół: filtracja AQ, 200 iteracji.

Efekt nadmiernej deformacji sygnału przez proces filtracji jest mniej widoczny dla przypadku modelu AQ. Rezultat otrzymany po 200 iteracjach pokazuje (rysunek 6-41, prawy dół) wykazuje znaczne podobieństwo do trajektorii oryginalnej (lewa góra). Przemieszczenie elementów brzegowych jest znacznie mniejsze niż w przypadku filtracji LQ. Dodatkowo charakterystyczne punkty zmiany krzywizny również zostały zachowane bez nadmiernego rozmycia sygnału. Widoczne jest, że położenie tych punktów jest nieznacznie inne niż w sygnale oryginalnym. Wynika to z wpływu sygnału zakłócającego, który zdeformował ogólny przebieg trajektorii. Jednakże ostatecznie kształt trajektorii jest zachowany zwłaszcza dla charakterystycznych punktów zmiany krzywizny. Właściwość ta została dokładniej przeanalizowana dla wybranego fragmentu trajektorii testowej na rysunku 6-42.



**Rysunek 6-42: Fragment trajektorii testowej nr 2a.  
Rezultaty filtracji AQ odpowiednio dla iteracji nr 25, 50, 75, 150.**

Na rysunku 6-42 pokazano przebieg filtracji dla fragmentu trajektorii testowej, dla której dobrze widoczny jest efekt pozwalający na zachowanie krzywizny sygnału. Po pierwszych 25 iteracjach sygnał zakłócający został usunięty z odfiltrowanej trajektorii. Jednakże spowodowało to także zniekształcenie fragmentu, dla którego oryginalnie występowała charakterystyczna zmiana kształtu sygnału. Po 50 iteracjach trajektoria jest jeszcze bardziej regularna. Po 75 iteracjach bardzo wyraźnie zaczyna być widoczne, że trajektoria przyjmuje kształt, który występował oryginalnie w tym fragmencie. Po 150 iteracjach otrzymuje się przebieg o kształcie w wysokim stopniu zbliżonym do kształtu oryginalnego.



Przedstawiony przebieg filtracji prezentuje własności analogiczne do filtracji typu *Perona-Malik* dla obrazów (patrz rozdział 2.6). W przypadku modelu PM również otrzymywało się efekt wyostżenia obrazu, które było realizowane przez dyfuzję przeprowadzaną w obszarach jednolitych. Zasada maksimum nie pozwalała na wprowadzenie nowych wartości do procesu, które nie występowałyby w obrazie oryginalnym. Mimo tego efekt wyostżania można było uzyskać dzięki zastosowaniu nieliniowego współczynnika dyfuzji zależnego od wartości funkcji jasności obrazu.

W przypadku trajektorii kwaternionów również występuje odpowiednik efektu wyostżania, który potwierdza teoretycznie zaprezentowane własności procesu (patrz rozdział 5.6). Analizując kolejne iteracje pokazane na rysunku 6-42 widać, że wyostżenie krzywizny nie wynika z oddalenia punktu, w którym występuje zakrzywienie. Wynika natomiast z szybszej dyfuzji przeprowadzanej w jednolitych fragmentach trajektorii. W wyniku tego otrzymuje się sygnał oczyszczony z zakłóceń przy jednoczesnym zachowaniu charakterystycznego kształtu decydującego o zawartości znaczeniowej.

#### 6.4.7 Podsumowanie badań dla filtracji AnisoQuat

Otrzymane rezultaty badań eksperymentalnych pozwoliły na zebranie następujących wniosków i obserwacji:

- Proces filtracji AQ posiada własności wyostżania trajektorii z punktu widzenia zachowania krzywizny trajektorii.
- Filtracja syntetycznych trajektorii testowych algorytmem AQ pozwala uzyskać od 20 do 80 procentową poprawę jakości sygnału w odniesieniu do odpowiadającego mu procesowi LQ.
- Otrzymywana poprawa jakości filtracji zmniejsza się dla wyższego stopnia zaszumienia sygnału oraz dla bardziej skomplikowanych trajektorii. Jednakże za każdym razem filtracja AQ wykazuje własności nie gorsze od procesu LQ.
- Zastosowanie adaptacyjnego sposobu wyznaczania współczynnika wrażliwości znacznie poprawia otrzymywane rezultaty.
- Filtracja ze współczynnikiem adaptacyjnym początkowo ma charakter zbliżony do filtracji LQ, lecz długookresowo wykazuje właściwości stabilizacji położenia wszystkich punktów trajektorii. Ostatecznie nawet po 10 000 iteracji sygnał był nadal stabilny o bardzo wysokiej wartości współczynnika QSNR.
- Zastosowanie warunku brzegowego typu D daje rezultaty znacząco lepsze od pozostałych. Pozwala on na korekcję położenia elementów brzegowych przy jednoczesnym zahamowaniu dyfuzji na brzegach trajektorii po usunięciu znacznej części zakłóceń.
- Algorytm AQ posiada własności algorytmu NLQ z możliwością dodatkowego sterowania przebiegiem filtracji ze względu na wartość lokalnej krzywizny trajektorii.

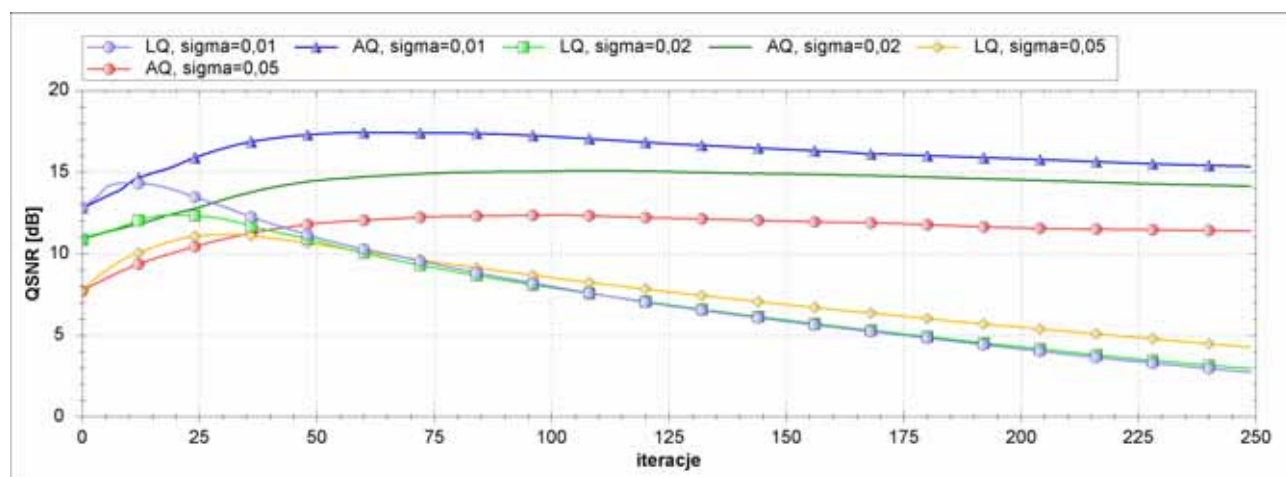
Należy zaznaczyć, że proponowane podejście różni się znacząco od algorytmów realizowanych przez wzory (5 – 12) - (5 – 14) [66]. W tamtym przypadku autorzy pracy proponują filtry, które mają jednoznacznie określone działanie – wygładzające lub wyostżające. Celem proponowanego algorytmu

*AnisoQuat* jest natomiast inteligentne wygładzenie trajektorii przeprowadzone w ten sposób, aby usunąć składową zakłócającą przy jednoczesnym zachowaniu znaczeniowych własności sygnału (np. zmiany krzywizny).

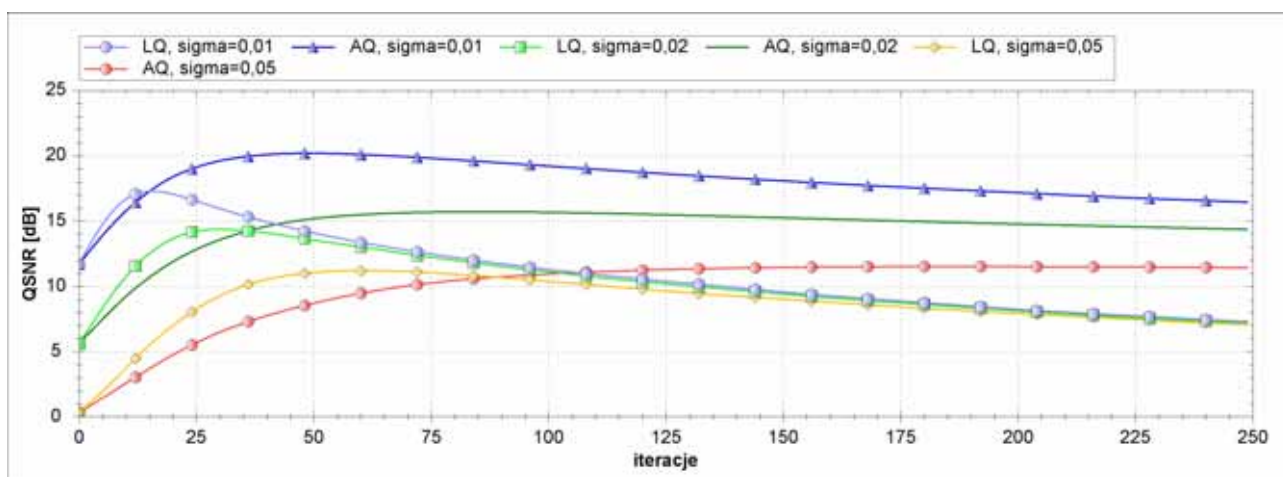
Podsumowując należy stwierdzić, że proponowany algorytm filtracji *AnisoQuat* pozwala na uzyskanie rezultatów o bardzo wysokiej jakości. Jest to osiągnięte przez złożenie własności filtracji wygładzającej z własnościami wyostrzającymi sygnał. W wyniku takiego połączenia otrzymuje się uniwersalny algorytm, który wykazuje dobre własności nawet dla trajektorii o bardzo wysokim stopniu zaszumienia.

## 6.5 Skuteczność filtracji dla sekwencji ruchu realizujących złożony model hierarchiczny

Zaproponowany w pracy algorytm *AnisoQuat* dobrze sprawdził się w eksperymentach numerycznych dla syntetycznych trajektorii testowych. Jego działanie zostało sprawdzone także dla sekwencji realizujących złożony hierarchiczny model kinematyczny zaprezentowany w rozdziale 4.6. Rozpatrywany jest ruch postaci ludzkiej zarejestrowany przy użyciu systemu *motion capture*. Zbadano charakterystykę filtracji AQ dla dwóch sekwencji testowych w zależności od stopnia zaszumienia sygnału. Uwzględniono najlepszy z otrzymanych przebiegów filtracji. Dla sekwencji *Run loopable* stosowano parametry  $K_{max}=4$ ,  $K_{Amax}=4$ , a dla sekwencji *Sprint to walk*  $K_{max}=4$ ,  $K_{Amax}=3$  (oprócz stopnia zaszumienia  $\sigma=0,05$ , dla którego użyto parametrów  $K_{max}=2$ ,  $K_{Amax}=4$ ). W obu przypadkach używana była adaptacyjna wersja funkcji stopującej  $g_t$ ,  $\lambda=0,1$ ,  $\lambda_A=0,5$ . Otrzymane rezultaty porównano z wynikami filtracji algorytmem LQ przy parametrze  $c = 0,05$ . Sygnał zakłócający wprowadzony do sekwencji miał rozkład normalny. Wyniki przeprowadzonych badań zostały przedstawione na rysunkach 6-43, 6-44.



Rysunek 6-43: Miara jakości QSNR dla rezultatów filtracji AQ oraz LQ w zależności od stopnia zaszumienia sygnału – rzeczywista, złożona trajektoria testowa *Run loopable*.



Rysunek 6-44: Miara jakości QSNR dla rezultatów filtracji AQ oraz LQ w zależności od stopnia zaszumienia sygnału – rzeczywista, złożona trajektoria testowa *Sprint to walk*.

Otrzymane rezultaty zbiorcze dotyczą wartości miary QSNR dla całego modelu hierarchicznego. Podobnie, jak w przypadku trajektorii testowych, również w tym przypadku filtracji AQ pozwala na uzyskanie lepszego rezultatu dla każdego z badanego poziomu zaszumienia sygnału. Otrzymana charakterystyka również wykazuje cechy stabilizacji trajektorii dla dużej liczby iteracji. Niemniej jednak w przypadku złożonych sekwencji stabilizacja taka nie występuje w wyraźnej postaci. Objawia się raczej w postaci wolniejszego spadku jakości sygnału po osiągnięciu maksymalnej wartości miary QSNR.

Dla obu testowanych sekwencji widoczna jest właściwość, która występowała (lecz w mniej wyraźny sposób) dla trajektorii syntetycznych. Przebieg filtracji LQ charakteryzuje się szybszym wzrostem jakości sygnału QSNR w porównaniu do odpowiadającemu mu procesowi AQ. Proces ten jest niehamowany z punktu widzenia zmiennych sterujących filtracją, więc dyfuzja zachodzi szybciej niż przy nawet minimalnym ograniczeniu przez funkcję stopującą. Jednakże charakterystyka długookresowa pokazuje, że proces AQ uzyskuje wyższą wartość miary jakości QSNR. W tabeli 6 zebrano maksymalny uzyskany przyrost jakości uzyskany algorytmem AQ w porównaniu do filtracji LQ w zależności od stopnia zaszumienia sekwencji.

	$\sigma=0,01$	$\sigma=0,015$	$\sigma=0,02$	$\sigma=0,05$
<b>Run loopable</b>	120,0%	118,8%	119,1%	108,7%
<b>Sprint to walk</b>	114,1%	111,3%	107,3%	101,1%

Tabela 6: Poprawa jakości sygnału dla maksymalnej wartości QSNR algorytmu AQ względem maksymalnej wartości uzyskanej filtracją LQ. Dwie sekwencje testowe dla złożonego modelu hierarchicznego.

Maksymalna poprawa uzyskana dzięki zastosowaniu algorytmu AQ jest w każdym z przypadków niższa niż rezultaty dla syntetycznych trajektorii testowych. Należy jednak pamiętać, że rozważana jest w tym przypadku sekwencja o wiele bardziej skomplikowana od trajektorii syntetycznych. Ruch postaci zawiera wiele subtelnych szczegółów, które nie zostały zamodelowane przez poprzednie trajektorie testowe. Jednakże nawet przy tak wysokim stopniu złożoności sekwencji, użycie algorytmu AQ pozwala na uzyskanie do 20% poprawy jakości filtracji w porównaniu do referencyjnego algorytmu LQ.

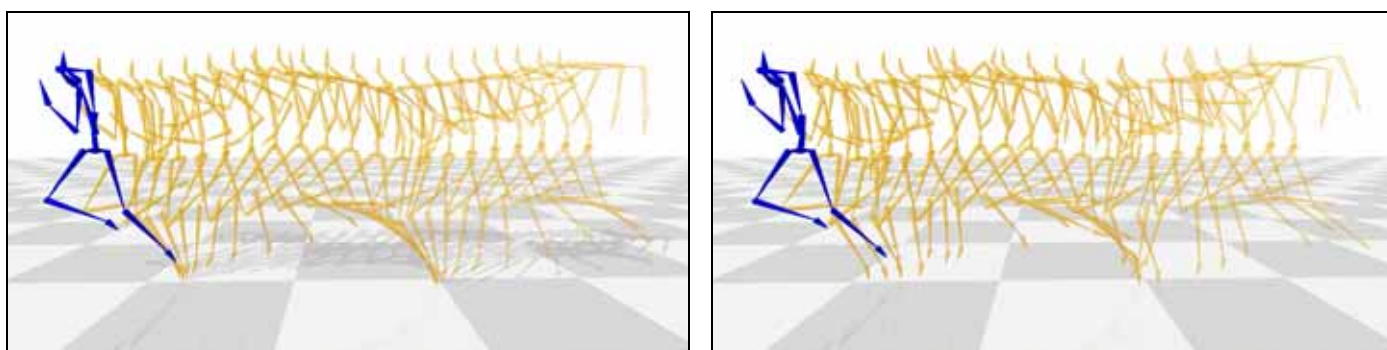
Biorąc pod uwagę tabelę 6 oraz wykresy na rysunkach 6-43, 6-44 widoczna jest właściwość, że sekwencja rzeczywista jest trudniejsza do filtracji dla każdego z algorytmów. Dotyczy to także wrażliwości

na poziom zaszumienia trajektorii. Już dla algorytmu LQ duży poziom zaszumienia  $\sigma=0,05$  nie pozwala na uzyskanie znacznej poprawy jakości filtracji (w porównaniu do sekwencji zaszumionej), a chwilowa poprawa związana jest z równie szybkim spadkiem miary QSNR. Dla porównania należy przypomnieć, że eksperymenty dla sekwencji syntetycznych wykazywały satysfakcjonujące własności algorytmów nawet dla 10-krotnie większego poziomu zaszumienia  $\sigma=0,5$ .

Jednocześnie można stwierdzić, że zaszumienie sekwencji ruchu na poziomie  $\sigma=0,02$  powoduje znaczne zniekształcenie sekwencji ruchu. Prawdopodobnie w praktyce nie będą występować zakłócenia na tak wysokim poziomie.

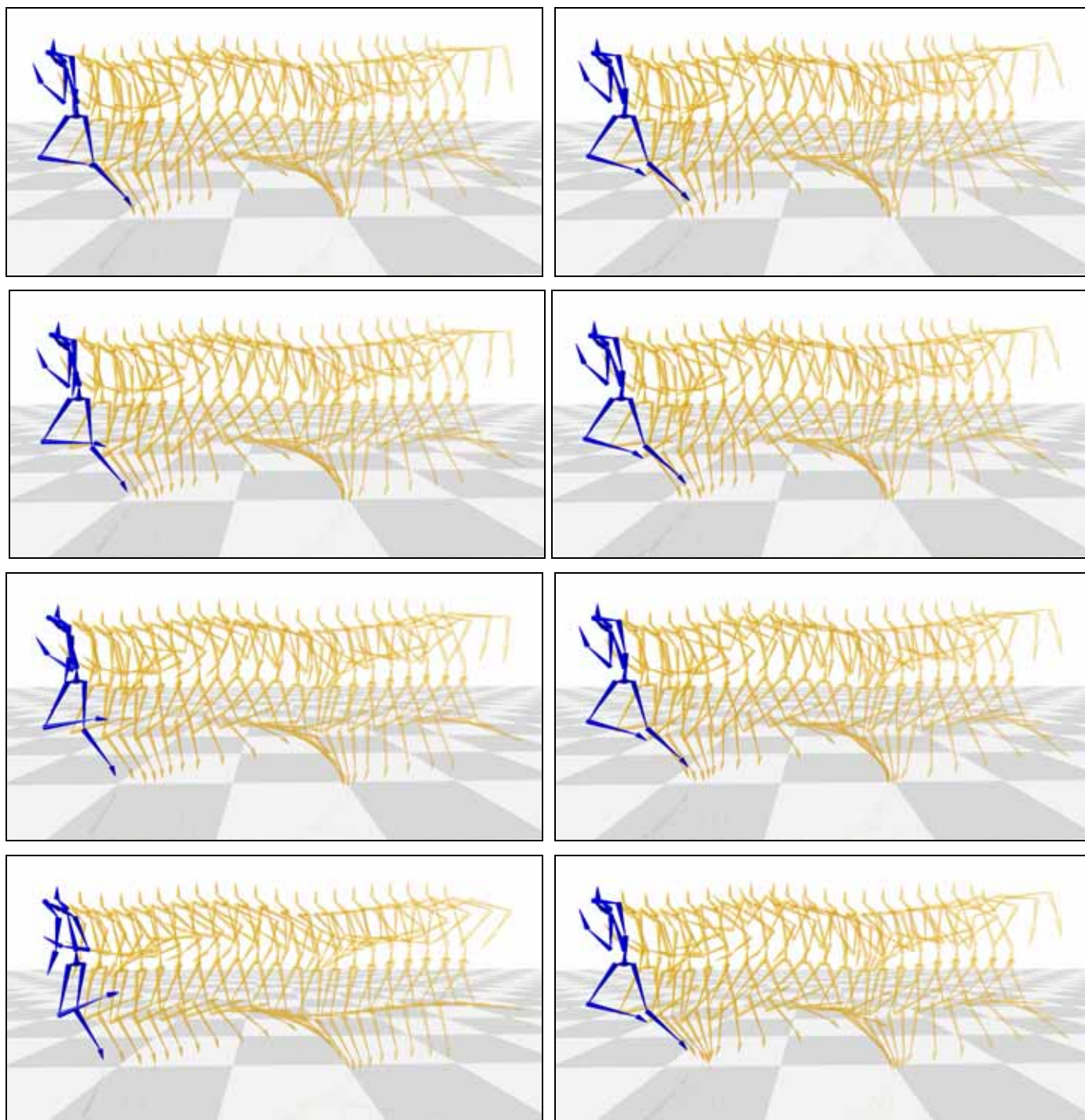
Wyniki zebrane w tabeli 6 wskazują także, że filtracja sygnału *Sprint to walk* jest trudniejsza w realizacji dla algorytmu AQ. Poprawa jakości filtracji w tym przypadku nie przekracza 15%, a dla większego zaszumienia spada do 1 %. Wynika to z wyższego stopnia zróżnicowania tej sekwencji testowej. Składa się ona z dwóch faz – biegu i chodu. Z tego względu chwilowa prędkość kątowa różni się znacząco w każdym z tych fragmentów. Dlatego trudniejszy jest odpowiedni dobór parametrów filtracji AQ tak, aby zachowane zostały charakterystyczne cechy ruchu dla obu części przetwarzanej sekwencji. Podobne zjawisko występowało dla syntetycznej sekwencji testowej nr 3.

Należy jednak zwrócić uwagę także na charakter przebiegu filtracji AQ uzyskiwanej dla obu sekwencji przy znacznym poziomie zaszumienia. O ile poprawa jakości uzyskiwanego sygnału jest niewielka (1 – 5 %), to sygnał ten nie jest podatny na zniekształcenia wynikające z rozmycia wynikającego z dyfuzji algorytmem LQ. Dlatego wartość miary QSNR dla przebiegów filtracji AQ pozostaje przez dłuższy czas stabilna na wyższym poziomie. Jest to bardzo ważna zaleta algorytmu AQ, nawet jeżeli względna poprawa jakości nie jest uwidocznioma przez wartość tego współczynnika. Aby dobrze zobrazować różnice w przebiegu filtracji zostaną one zaprezentowane na przykładzie sekwencji *Run loopable*.



**Rysunek 6-45: Oryginalna oraz zaszumiona testowa sekwencja ruchu *Run loopable*. Szum o rozkładzie normalnym,  $\sigma=0,02$ .**

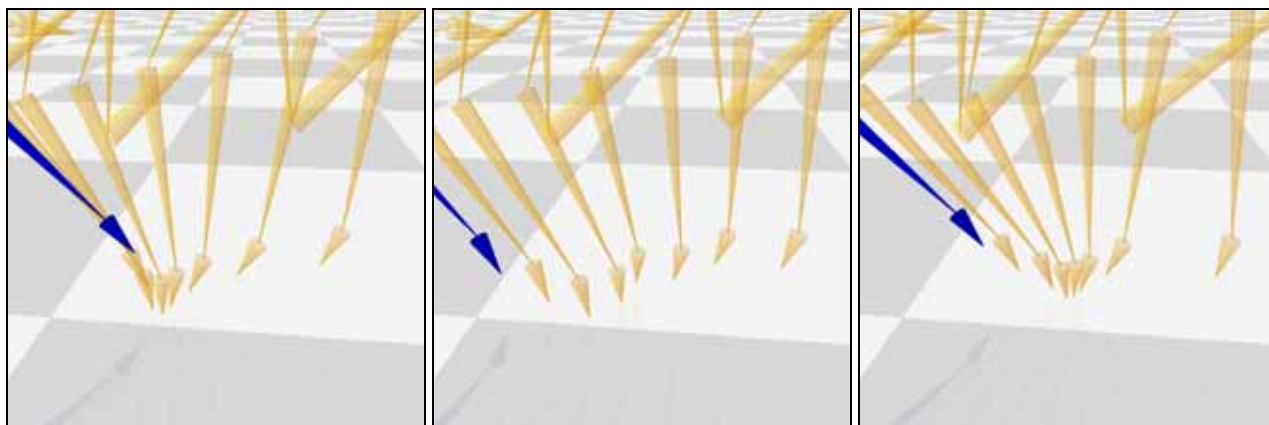
Na rysunku 6-45 przedstawiono oryginalną sekwencję ruchu *Run loopable* oraz sekwencję zaszumioną sygnałem o rozkładzie normalnym,  $\sigma=0,02$ . Rysunek 6-46 pozwoli na przeanalizowanie przebiegu filtracji wykonanej na tej sekwencji ruchu algorytmami LQ oraz AQ.



**Rysunek 6-46: Wizualizacja kolejnych kroków filtracji LQ (lewa kolumna) oraz AQ (prawa kolumna). W wierszach pokazana odpowiednio iteracje nr: 10, 20, 50, 200.**

Wizualizacja kolejnych iteracji procesów potwierdza wcześniej zaobserwowaną zależność – początkowy przebieg filtracji LQ pozwala na szybsze usunięcie wprowadzonych zakłóceń. Jednakże już po 20 iteracjach widoczny zaczyna być efekt rozmywania trajektorii przez nadmierny poziom niezahamowanej dyfuzji. Dla filtracji AQ dopiero wtedy rozpoczyna się właściwe odsumowanie algorytmu. Brak synchronizacji pomiędzy przemieszczeniem całego ciała (który nie jest zaszumiony), a wykonywanym ruchem kończyn jest szczególnie widoczny dla filtracji LQ po 50 i 200 iteracjach. Te same kroki algorytmu AQ pozwalają uzyskać sekwencję ruchu, który jest zbliżony do ruchu oryginalnego. Należy zwrócić uwagę

na położenie ciała postaci w ostatniej klatce ruchu. Proces LQ w bardzo znaczący sposób zmodyfikował położenie wszystkich kończyn postaci ze względu na minimalizację energii położenia. Proces AQ biorąc pod uwagę lokalną krzywiznę trajektorii pozwolił na uzyskanie przefiltrowanej sekwencji o charakterze znaczeniowym zbliżonym do sekwencji oryginalnej.



**Rysunek 6-47: Fragment sekwencji Run loopable. Wersja oryginalna, rezultat filtracji LQ (20 iteracji) rezultat filtracji AQ (200 iteracji).**

Na rysunku 6-47 pokazany został fragment sekwencji ruchu, w którym dobrze widoczny jest ruch nogi postaci. Filtracja LQ (środkowa część rysunku) już po 20 iteracjach (dla których uzyskiwana jest maksymalna wartość jakości QSNR) wprowadza znaczne zniekształcenie położenia dla tej kończyny. Na podstawie rysunku widoczne jest, że dla kolejnych iteracji zniekształcenie to staje się coraz bardziej znaczące. W prawej części rysunku przedstawiony jest rezultat filtracji AQ po 200 iteracjach. Pewne zniekształcenie sekwencji wynikające z zaszumienia pozostało. Jednakże charakter ruchu w postaci wyraźnie postawionego kroku jest w tym przypadku znacznie lepiej odwzorowany. Na podstawie otrzymanego rezultatu filtracji można następnie w sposób jednoznaczny określić na przykład chwile, w których wykonywane jest odbicie nogi postaci od podłoża. W dalszej analizie może to być pomocne do wyznaczenia globalnych parametrów ruchu takich, jak częstotliwość kroków itp.

## **Wnioski z eksperymentów dla sekwencji ruchu**

Otrzymane rezultaty wykazują dużo pozytywnych własności filtracji AQ także w odniesieniu do złożonych sekwencji rzeczywistych dla modeli hierarchicznych. Wśród nich należy następujące własności:

- Filtracja sekwencji ruchu jest trudniejsza do przeprowadzenia dla każdego z badanych algorytmów. Jednakże nawet dla tak trudnych warunków proces AQ uzyskiwał do 20% poprawy jakości sygnału.
- Proces AQ również w dla tego typu sekwencji jest bardziej stabilny. Oznaczy to, że nawet po bardzo wielu iteracjach nie deformuje znacząco sygnału przez wprowadzanie nadmiernej dyfuzji. Deformacja jest jednak słabiej zahamowana w porównaniu do filtracji syntetycznych trajektorii testowych ze względu na znaczne skomplikowanie sekwencji ruchów.
- Filtracja sekwencji ruchu wykazuje własności zachowywania znaczeniowego charakteru złożonego sygnału przez wyostrażanie trajektorii w punktach zmiany krzywizny ruchu. Jest to

bardzo istotna cecha ze względu na możliwość dalszego przetwarzania tak przefiltrowanych danych w algorytmach porównywania, klasteryzacji i rozpoznawania ruchów na podstawie wyodrębnionych właściwości globalnych.

Otrzymane rezultaty pokazały, że zaproponowany w pracy algorytm typu *AnisoQuat* ma bardzo wiele zalet w przypadku filtracji zarówno trajektorii syntetycznych, jak i rzeczywistych złożonych sekwencji ruchu. Jednak możliwe to było jedynie dzięki zastosowaniu nieliniowego, niejednostajnego sterowania procesem dyfuzji uzależnionego od wartości w przetwarzanej trajektorii kwaternionów. Przeprowadzone eksperymenty wykazały w praktyce zalety algorytmu oraz zweryfikowały właściwości procesu, które wcześniej zostały opracowane teoretycznie. Zróżnicowanie przygotowanych trajektorii testowych pozwoliło natomiast sprawdzić własności algorytmu dla różnych konfiguracji i przypadków, które mogą występować w problemach praktycznych.

## Rozdział 7

### Podsumowanie

W prezentowanej pracy zbadana została tematyka szeroko pojętej filtracji sygnałów na podstawie równań różniczkowych o pochodnych cząstkowych. Metody filtracji oparte na tego typu równaniach są powszechnie stosowane w dziedzinie przetwarzania obrazów. Od wielu lat stosowane są filtry liniowe wykorzystujące proces dyfuzji opisany liniowym równaniem różniczkowym. W latach 90 szczególnie znaczący wpływ na dalszy rozwój dziedziny miała praca *Perona-Malik* [79], w której sformalizowano zastosowanie nieliniowego równania różniczkowego do filtracji obrazów. Wprowadzenie czynnika nieliniowego pozwoliło na otrzymanie procesu, który usuwał składową zakłócającą przy jednoczesnym zachowywaniu ważnych elementów obrazu rozumianych, jako krawędzie. Filtry tego typu zostały następnie uogólnione do przypadku anizotropowego i przebadane między innymi w pracy [105] (Weickert).

W pracy przedstawiono podstawowe własności tego typu nieliniowego filtracji. W wyniku przeprowadzania wstępnych badań został zaproponowany sprzężony algorytm filtracji nieliniowej uwzględniający dodatkową informację o strukturze geometrycznej obrazu. W modelu *Perona-Malik* nieliniowy czynnik w postaci funkcji stopującej oparty jest na lokalnym estymatorze krawędzi realizowanym przez dyskretną postać gradientu jasności. W ogólności czynnik ten może być tensorem uwzględniającym anizotropię, jednak za każdym razem zależy on od lokalnie wyznaczonych wartości. Propozycja przedstawiona w rozdziale 3 pracy polega na uwzględnieniu informacji o strukturze geometrycznej przez zastosowanie globalnego deskryptora obrazu. Przykładowo zastosowany został deskryptor w postaci transformaty Hougha dla linii prostych. Proces filtracji wykorzystuje więc dwa źródła informacji do estymacji istotnych własności obrazu, jakimi są krawędzie: lokalnego gradientu jasności i globalnego deskryptora kształtu.

Przeprowadzone eksperymenty pokazały, że zastosowanie zaproponowanego procesu sprzężonego poprawia jakość filtracji na dwa sposoby: krawędzie obiektów stają się bardziej regularne, a fragmenty obszarów jednolitych zostają szybciej i dokładniej oczyszczone z zakłóceń. Zastosowanie algorytmu sprzężonego musi być powiązane z rodzajem informacji dodatkowej o przetwarzanym obrazie. Proponowany algorytm pozwala na dwójakie określenie wiedzy globalnej. Możliwe jest zdefiniowanie rodzaju struktur geometrycznych, które mają być wyróżnione na obrazie za pomocą wyboru odpowiedniej wersji transformacji Hougha (lub innego deskryptora globalnej geometrii obrazu). Można także określić



ilościowo stopień występowania wybranych struktur: dokładna liczba lub względny poziom struktur, które mają być brane pod uwagę. Druga metoda jest bardziej uniwersalna i pozwala na zastosowanie algorytmów sprzężonych w automatycznych systemach przetwarzania obrazów.

Szczególnym przypadkiem sygnału są trajektorie zapisane w przestrzeniach rotacji często wykorzystywane w wielu zastosowaniach praktycznych. W rozdziale 4 zebrano sposoby opisu rotacji ze szczególnym uwzględnieniem przestrzeni kwaternionów. Kwaterniony są często używane, jako alternatywa dla innych sposobów parametryzacji przestrzeni orientacji układów (np. macierzy obrotu lub kątów Eulera). W skrócie przedstawiono własności i możliwości zastosowania tego typu liczb do opisu trajektorii w przestrzeniach rotacji. Przedstawiono również przegląd literaturowy zagadnień związanych z tematyką kwaternionów. Najważniejsze zagadnienia, które są używane w dalszych częściach pracy zostały opisane dokładniej. Również na potrzebę kolejnych rozdziałów zostało zaprezentowane uogólnienie metody dynamicznego zniekształcania dziedziny czasu (ang. *Dynamic Time Warping*) dla trajektorii kwaternionów. Zaproponowano metodę QDTW, która wykorzystuje wielokryterialny wskaźnik podobieństwa elementów trajektorii. Na jej podstawie opracowano miarę podobieństwa trajektorii kwaternionów QSNR wykorzystywaną w dalszych badaniach praktycznych.

W pracy skoncentrowano się na trajektoriach kwaternionów reprezentujących ruch w przestrzeni rotacji. Istnieje wiele metod filtracji i przetwarzania sygnałów dla przestrzeni rzeczywistych. Jednakże przestrzeń rotacji charakteryzuje się swoją specyfiką, która nie pozwala na bezpośrednią adaptację tego rodzaju metod.

Dla trajektorii kwaternionów również istnieją pewne algorytmy przetwarzania. Krótki przegląd tego rodzaju metod został przedstawiony w rozdziale 6.1 Jednakże do tej pory nie są znane autorowi opracowania, w których zaproponowana została konstrukcja algorytmów filtracji oparta na dyskretnym rozwiązaniu równania różniczkowego dla przestrzeni kwaternionów. W rozdziale 6.2 zostało przedstawione opracowanie metody filtracji trajektorii kwaternionowych na podstawie liniowego równania różniczkowego dyfuzji ciepła. Metoda to jest zbieżna z kilkoma innymi istniejącymi podejściami cytowanymi wcześniej.

Uogólnienie modelu *Perona-Malik* pozwoliło na opracowanie metody filtracji trajektorii uwzględniającej lokalną zmianę chwilowej prędkości kątowej *NonLinQuat*. Zaproponowany algorytm został przedstawiony w rozdziale 5.3. Jednakże specyfika przestrzeni kwaternionów jest inna niż dwuwymiarowej przestrzeni obrazów. Algorytm *NonLinQuat* ma więc ograniczone zastosowanie do trajektorii, w których znaczącą informacją jest prędkość wykonywanego ruchu.

W praktyce istotniejszym czynnikiem pozwalającym na rozróżnienie trajektorii rotacji jest zmiana ich lokalnej krzywizny. Założenie to jest punktem wyjścia do uogólnienia dyskretyzacji modelu *Perona-Malik* do zaawansowanego algorytmu filtracji trajektorii kwaternionów *AnisoQuat*. Algorytm został zaprezentowany w rozdziale 5.4. W rozdziale 5.6 przedstawiono natomiast własności algorytmów, dla których wspólnym korzeniem jest równanie różniczkowe cząstkowe procesu dyfuzji.

Zaproponowane algorytmy wykazują się ciekawymi własnościami filtracji dla trajektorii kwaternionowych. Przeprowadzono serię eksperymentów przy użyciu zarówno syntetycznych, jak i

rzeczywistych trajektorii testowych. Wyniki badań zostały zebrane i opisane w rozdziale 6. Podsumowując uzyskane rezultaty można wskazać na następujące istotne cechy zaproponowanych algorytmów:

- Algorytm *AnisoQuat* wykazuje bardzo dobre własności filtracji trajektorii przy jednoczesnym zachowywaniu jej charakteru przez nieliniowy proces zależny od lokalnej krzywizny sygnału.
- Wykazano, że algorytm *AnisoQuat* posiada własności wyostrzające sygnał. Oznacza to, że dyfuzja zostaje zahamowana we fragmentach, które niosą informację znaczeniową (znaczna zmiana krzywizny ruchu).
- Długookresowa własności algorytmu *AnisoQuat* wskazują na uzyskiwanie stabilnych stanów trajektorii, dla których nadmiarowa dyfuzja zostaje zahamowana (w przeciwieństwie do filtracji *LinQuat*).
- Filtracja rzeczywistych sekwencji ruchu pozwala na prawidłowe odtworzenie charakteru ruchu. Użycie algorytmu *AnisoQuat* pozwala uzyskać przebieg sekwencji, w którym istotne właściwości zostają znacząco uwypuklone (na przykład chwila wykonywania kroku).
- Filtracja realizowana przez zaproponowane algorytmy zachowywały wcześniej teoretycznie opracowane własności: energia sygnału była minimalizowana (nierosnąca), nie zostały wprowadzone żadne nowe położenia punktów w trajektorii (przestrzenna zbieżność przekształcenia), algorytmy wykazywały stacjonarność i niezmienniczość względem transformacji.
- Zaproponowany współczynnik jakości QSNR dobrze spełniała rolę miary podobieństwa w procesie porównywania otrzymanych rezultatów z trajektoriami oryginalnymi.
- W porównaniu do referencyjnego algorytmu *LinQuat*, algorytm *AnisoQuat* wykazywał do 80 % poprawy jakości sygnału dla trajektorii syntetycznych oraz do 20 % dla trajektorii rzeczywistych.

Podsumowując należy stwierdzić, że w prezentowanej pracy udało się osiągnąć następujące nowe rezultaty dotyczące równań różniczkowych w problemach filtracji:

- Zaproponowano algorytm filtracji uwzględniający globalną (aprioryczną) wiedzę o strukturze geometrycznej obrazu.
- Przebadano i przeanalizowano wyniki badań sprzężonego algorytmu filtracji obrazów dla globalnego deskryptora struktury geometrycznej w postaci transformaty Hougha.
- Uogólniono algorytm *Dynamic Time Warping* dla trajektorii w przestrzeni kwaternionów.
- Zaproponowano algorytm *LinQuat* oparty na realizacji dyskretnej wersji liniowego równania różniczkowego dyfuzji ciepła.
- Uogólniono model *Perona-Malik* dla przypadku filtracji dla trajektorii kwaternionów.
- Zaproponowano konstrukcję zaawansowanego algorytmu filtracji trajektorii kwaternionów *AnisoQuat*, który zachowuje istotny charakter ruchu wzdłuż trajektorii przez uwzględnienie lokalnej krzywizny sygnału.

- Zrealizowano adaptacyjną wersję algorytmu *AnisoQuat*, której parametry lokalne są uzależnione od globalnych chwilowych własności sygnału.
- Przeprowadzono serię eksperymentów dla syntetycznych i rzeczywistych trajektorii w przestrzeni kwaternionów, które potwierdziły wysoką skuteczność zaproponowanych algorytmów.

Ponadto otrzymane zostały następujące wyniki teoretyczne:

- Opracowano miarę podobieństwa trajektorii kwaternionów QSNR. Miara ta oparta jest na opracowanym algorytmie QDTW oraz uwzględnia specyficzne własności trajektorii w przestrzeni orientacji.
- Wykazano, że żaden z prezentowanych algorytmów nie jest liniowy. W szczególności nie jest liniowy algorytm *LinQuat*, który pośrednio oparty jest na liniowym równaniu różniczkowym dyfuzji. Nieliniowość algorytmu wynika z nieliniowości sfery jednostkowej kwaternionów będącej dziedziną rozpatrywanej trajektorii.
- Dla wszystkich rozpatrywanych algorytmów wykazano stacjonarność przekształcenia oraz niezmienniczość względem transformacji w przestrzeni  $S^3$ .
- Wykazano, że każde z proponowanych przekształceń posiada własność przestrzennej zbieżności. Własność ta została następnie potwierdzona eksperymentalnie.

Zaproponowane algorytmy przetwarzania danych mogą być używane w wielu zastosowaniach praktycznych dotyczących przetwarzania trajektorii ruchu. Potencjalne obszary zastosowań obejmują między innymi: analizę trajektorii samolotu, przetwarzanie toru efektora manipulatora, rozpoznawanie ruchów człowieka. W ostatnim czasie coraz bardziej popularna staje się analiza semantyczna, której celem jest określenie znaczenia wykonywanego ruchu, a także dodatkowych informacji indywidualnie związanych z osobą wykonywującą ruch. Prezentowane algorytmy filtracji mogą być zastosowane do odszumiania sekwencji, które zostały zapisane w postaci trójwymiarowej. Oddzielnym możliwym zastosowaniem jest wykorzystanie możliwości uwypuklenia niektórych fragmentów trajektorii, które mogą być źródłem istotnych informacji o znaczeniu ruchu. Możliwe jest to dzięki własnościom algorytmów pozwalające na zachowywanie kształtu przetwarzanej trajektorii. Uwypuklone elementy mogą być następnie wykorzystane, jako cechy charakterystyczne w metodach porównywania i klasyfikacji danych.

Dalsze badania nad proponowanymi algorytmami mogą uwzględniać, analogicznie, jak w przypadku obrazów, dodatkowe informacje o wykonywanych ruchach. Informacje te mogą być wyrażony przykładowo w postaci modelu ograniczeń nałożonych na obserwowany obiekt. Dzięki temu może zostać jeszcze bardziej zwiększona szybkość oraz jakość otrzymywanej filtracji. Zaprezentowane metody mogą więc być punktem wyjścia zarówno do dalszego rozwoju dziedziny filtracji sygnałów w przestrzeniach rotacji, jak też zastosowania ich w złożonych, automatycznych systemach przetwarzających.

## Bibliografia

- [1] Aubert G., Kornprobst P., *Mathematical Problems in Image Processing: Partial differential Equations and the Calculus of Variations*, Springer Verlag, New York, 2002
- [2] Avro J., *Fast Random Rotation Matrices*, Graphics Gems III, pp. 117-120, Academic Press, 1992
- [3] Avro J., *Stratified sampling of spherical triangles*, Proc. of the 22nd annual conference on Computer graphics and interactive techniques, SIGGRAPH, pp. 437-438, 1995
- [4] Bagiński P., *Metody optymalnego łączenia sekwencji ruchu animowanych postaci ludzkich*, praca magisterska, Politechnika Wrocławska, Wrocław, 2003
- [5] Ballard D., *Generalized Hough transform to detect arbitrary patterns*, Pattern Recognition, Vol. 13, No. 2, pp. 111-122, 1981
- [6] Barenblatt G. I., Bertsh M., Dal Passo R., Ughi M., *A degenerate pseudoparabolic regularization of a nonlinear forward-backward heat equation arising in the theory of heat and mass exchange ion stably stratified turbulent shear flows*, SIAM Journal of Mathematic Analysis, Vol. 24, pp. 1414-1439, 1993
- [7] Barr A. H., Currin B., Gabriel S., Hughes J. F., *Smooth Interpolation of Orientations with Angular Velocity Constraints using Quaternions*, Computer Graphics, Vol. 26, No. 2, 1992
- [8] Black M., Sapiro G., Marimont D., Heeger D., *Robust Anisotropic Diffusion*, IEEE Trans. On Image Processing, Vol. 7, No. 3, 1998
- [9] Bottema O., Roth B., *Theoretical Kinematics*, Dover Publications, New York, 1990
- [10] Bregler C., Malik J., *Tracking People with Twists and Exponential Maps*, Proc. Of IEEE Computer Vision and Pattern Recognition, 1998
- [11] Brown J. L., Worsey A. J., *Problems with defining barycentric coordinates for the sphere*, Mathematical Modelling and Numerical Analysis, Vol. 26, pp. 37-49, 1992
- [12] Burdick J.W., *Introduction to Kinematics and Robotics – Course Handouts*, California Institute of Technology, 2006 (praca niepublikowana)
- [13] Burg J., *Maximum entropy spectral analysis*, PhD thesis, Standford University, 1975
- [14] Buss S. R., Fillmore J. P., *Spherical Averages and Applications to Spherical Splines and Interpolation*, ACM Trans. on Graphics, Vol. 20, No. 2, pp. 95-126, 2001
- [15] Catté F., Lions P.L., Morel J. M., Coll T., *Image selective smoothing and edge detection by nonlinear diffusion*, SIAM, Journal of Numerical Analysis, Vol. 29, pp. 182-193, 1992
- [16] Cottet G. H., *Neural networks: Continuous approach and applications to image processing*, Journal of Biological Systems, Vol. 3, pp. 1131-1139, 1995.
- [17] Cottet G. H., El Ayyadi M., *Nonlinear PDE operators with memory terms for image processing*, Proc. IEEE International Conference on Image Processing, Vol. 1, pp. 481-483, 1996
- [18] Craig J., *Wprowadzenie do robotyki. Mechanika i sterowanie*, WNT, Warszawa 1995
- [19] Dam E. B., Koch M., Lillholm M., *Quaternions, Interpolation and Animation*, Technical Report DIKU-TR-98/5, University of Copenhagen, 1998
- [20] Delaney B., *On the Trail of the ShadowWoman: The Mystery of Motion Capture*, Computer Graphics and Applications, Vol. 18, No. 5, pp. 14-10, 1998

- [21] Denzler J., Zobel M., Niemann H., *On Optimal Camera Parameter Selection in Kalman Filter Based Object Tracking*, Lecture Notes In Computer Science: Pattern Recognition, Vol. 2449, Springer Berlin, pp. 17-25, 2002
- [22] Devroye L., *Non-Uniform Random Variate Generation*, Springer-Verlag, New York, 1986
- [23] Didas S., Weickert J., Burgeth B., *Stability and Local Feature Enhancement of Higher Order Nonlinear Diffusion Filtering*, w DAGM 2005, LNCS, Vol. 3663, pp. 451-458, Springer-Verlag, Berlin, 2005
- [24] Duda R., Hart P., *Use of the Hough transformation to detect lines and curves in the pictures*, Communications of the ACM, Vol. 15, No. 1, pp. 11-15, 1972
- [25] Dulęba I., Khefifi W., *Layer Method of Motion Planning for Driftless Nonholonomic Systems*, w: Cybernetics of Robotnic Systems, red. Tchoń K., pp. 7-25, Wydawnictwa Komunikacji i Łączności, Warszawa, 2005
- [26] Dulęba I., *Metody i algorytmy planowania ruchu robotów mobilnych i manipulacyjnych*, Exit, 2005
- [27] Eberly D. *Quaternion Algebra and Calculus*, Magic Software Inc., 2002
- [28] Fang Y. C., Hsieh C. C., Kim M. J., Chang J. J., Woo T. C., *Real time motion fairing with unit quaternions*, Computer-Aided Design, Vol. 30, No. 3, pp. 191-198, Elsevier, 1998
- [29] Fisher B., Perkins S., Walker A., Wolfart E., *Hypermedia Image Processing Reference*, University of Edinburgh, 1995
- [30] Gerig G., Kikinis R., Kübler O., Jolesz F.A., *Nonlinear Anisotropic Filtering of MRI Data*, IEEE Trans. on Medical Imaging, Vol. 11, No. 2, 1992
- [31] Goddard J., *Pose and motion estimation from vision using dual quaternion-based extended Kalman filtering*, PhD Thesis, The University of Tennessee, 1997
- [32] Goldman R. N., *Recovering the data form the transformation matrix*, Graphics Gems II, pp. 324-331, Academic Press, 1991
- [33] Gorard S., *Revisiting a 90-year-old debate: the advantages of the mean deviation*, Proc. of British Educational Research Association Annual Conference, Manchester, 2004
- [34] Grassia F. S., *Practical Parameterization of Rotations Using the Exponential Map*, The Journal of Graphics Tools, Vol. 3, No. 3, 1998
- [35] Grassia F. S., *Believable Automatically Synthesized Motion by Knowledge-Enhanced Motion Transformation*, PhD thesis, Carnegie Mellon University, 2000
- [36] Guil N., Villalba J., Zapata E., *A Fast Hough Transform for Segment Detection*, IEEE Trans. on Image Processing, Vol. 4, No. 11, 1995
- [37] Guil N., Cózar J.R., Zapata E.L., *Planar 3D Object Detection by Using the Generalized Hough Transform*, Proc. of 10<sup>th</sup> International Conference on Image Analysis and Processing, Venice, Italy, 1999
- [38] Hamilton W. R., *On a new Species of Imaginary Quantities connected with a theory of Quaternions*, Proc. of the Royal Irish Academy, vol. 2, 1843
- [39] Hashi S., Tokunaga Y., Yabukami S., Toyoda M., Ishiyama K., Okazaki Y., Arai K.I., *Development of real-time and highly accurate wireless motion capture system utilizing soft magnetic core*, IEEE Trans. on Magnetics, Vol. 41, No. 10, pp. 4191 – 4193, 2005
- [40] Herda L., Fua P., Plankers R. D., Boulic R., Thalmann D., *Skeleton-Based Motion Capture for Robust Reconstruction of Human Motion*, Proc. of IEEE Conference on Computer Animation, Philadelphia, 2000
- [41] Hornung A. Sar-Dessai S., Kobbelt L., *Self-calibrating optical motion tracking for articulated bodies*, Proc. of IEEE Conf. on Virtual Reality, pp. 72-85, 2005

- [42] Hsieh C. C., Fang Y. C., Wang M. E., Wang C. K., Kim M. J., Shin S. Y., Woo T. C., *Noise smoothing for VR equipment in quaternions*, IIE Transactions, Vol. 30, No. 7, pp. 581- 587, Springer, 1998
- [43] Hsieh C. C., Chang T. Y., *Motion fairing using genetic algorithms*, Computer-Aided Design, Vol. 35, No. 8., pp. 739-749, 2003
- [44] Illingworth J., Kittler J., *The Adaptive Hough Transform*, IEEE Trans. on Pattern Recognition and Machine Intelligence, Vol. 9, No. 5, pp. 690-698, 1987
- [45] Isner M., *Solving rotation problems in XSI 2.0 or greater*, praca niepublikowana, 2001
- [46] Jabłoński B., *Metody porównywania generatorów ruchu animowanych postaci ludzkich*, praca magisterska, Politechnika Wrocławska, Wrocław 2003
- [47] Jabłoński B., Klempous R., Kulbacki M., *PDE-based filtering of motion sequences*, Journal of Computational and Applied Mathematics, Vol. 189, No. 1-2, pp. 660-675, Elsevier, 2006
- [48] Johnson M. P., *Exploiting Quaternions to Support Expressive Interactive Character Motion*, PhD thesis, Massachusetts Institute of Technology, 1995
- [49] Kącki E., *Równania różniczkowe cząstkowe w zagadnieniach fizyki i techniki*, WNT, Warszawa, 1992
- [50] Keogh E., Pazzani, M., *Dynamic Time Warping with Higher Order Features*. First SIAM International Conference on Data Mining (SDM'2001), Chicago, USA, 2001
- [51] Keogh E., Palpanas T., Zordan V., Gunopulos D., Cardle M., *Indexing Large Human-Motion Databases*. Proc. of the 30th International Conference on Very Large Data Bases, Toronto, Canada, 2004
- [52] Kesidis A. L., Papamarkos N., *On the Inverse Hough Transform*, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 21, No. 12, 1999
- [53] Kim M. J., Kim M.S., *A  $C^2$ -continuous B-spline Quaternion Curve Interpolating a Given Sequence of Solid Orientations*, Computer Animation, 1995
- [54] Kim M. J, Kim M.S., Shin S.Y., *A General Construction Scheme for Unit Quaternion Curves with Simple High Order Derivatives*, Proc. of SIGGRAPH'95, pp. 369-376, 1995
- [55] Kim M. J., Hsieh C. C., Wang M. E., Wang C. K., Fang Y. C., Woo T. C., *Noise smoothing for VR equipment in the quaternion space*, Proc. of Symposium on Virtual Reality in Manufacturing Research and Education, Chicago, 1996
- [56] Kim M. J., Kim M. S., *A Compact Differentia Formula for the First Derivative of a Unit Quaternion Curve*, Journal of Visualization and Computer Animation, Vol. 7, No. 1, pp. 43-57, 1996
- [57] Kim T., Lee J., Shin S. Y., *Robust Motion Watermarking based on Multiresolution Analysis*, Computer Graphics Forum, Vol. 19, No. 3, 2000
- [58] Kiryati N., Eldar Y., Bruckstein A., *A probabilistic Hough transform*, Pattern Recognition, Vol. 24, No. 4, pp. 303-316, Elsevier, 1991
- [59] Korbicz J., Zajda Z., Solnik W., Uciński D., *Wybrane metody numerycznego rozwiązywania równań różniczkowych cząstkowych*, Wydawnictwo Wyższej Szkoły Inżynierskiej, Zielona Góra 1991
- [60] Kulbacki M., *Podstawowe metody syntezy ruchu animowanych postaci ludzkich*, Praca magisterska, Politechnika Wrocławska, Wydział Elektroniki, Wrocław, 2001
- [61] La Valle S. M., *Planning Algorithms*, Cambridge University Press, 2006
- [62] LaViola J. J., *A comparison of Unscented and Extended Kalman Filtering for Estimating Quaternion Motion*, Proc. of the American Control Conference, IEEE Press, pp. 2435-2440, 2003
- [63] Lee J., Shin S. Y., *Motion Fairing*, Proc of Computer Animation, Geneva, Swiss, pp. 136-143, 1996

- [64] Lee J., Shin S. Y., *Multiresolution Motion Analysis with Applications*, Proc. of International workshop on Human Modeling and Animation, Seoul, pp. 131-143, June 2000
- [65] Lee J., Shin S. Y., *A Coordinate-Invariant Approach to Multiresolution Motion Analysis*, Graphical Models, Vol. 63, No. 2, pp. 87-105, 2001
- [66] Lee J., Shin S. Y., *General Construction of Time-Domain Filters for Orientation Data*, IEEE Trans. on Visualization and Computer Graphics, Vol. 8, No. 2, pp. 119-128, 2002
- [67] Lindeberg T., *Discrete Scale-Space Theory and the Scale-Space Primal Sketch*, PhD thesis, Department of Numerical Analysis and Computer Science, Royal Institute of Technology, Stockholm, 1991.
- [68] Lysaker M., Lundervold A., Tai X.-C., *Noise Removal Using Fourth-Order Partial Differential Equation With Application to Medical Magnetic Resonance Images in Space and Time*, IEEE Trans. on Image Processing, Vol. 12, No. 12, 2003
- [69] Maciejewski A., *Dealing with the Ill-Conditioned Equations of Motion for Articulated Figures*, IEEE Trans. on Computer Graphics and Applications, Vol. 10, No. 3, pp. 63-71, 1990
- [70] Maillot P.G., *Using quaternions for coding 3D transformations*, Graphics Gems I, pp. 498-515, Academic Press, 1990
- [71] Marczuk G. I., *Analiza numeryczna zagadnień fizyki matematycznej*, PWN, Warszawa, 1983
- [72] Marins J. L., Yun X., Bachmann E. R., McGhee R., Zyda M. J., *An Extended Kalman Filter for Quaternion-Based Orientation Estimation Using MARG Sensors*, Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 2003 – 2011, 2001
- [73] Matras J., Galambos C., Kittler J., *Progressive Probabilistic Hough Transform*, Proc. of British Machine Vision Conference, 1998
- [74] Mrazek P., *Selection of Optimal Stopping Time for Nonlinear Diffusion Filtering*, w *Space Theories in Computer Vision*, pp. 290 – 298, 2001
- [75] Nakamura Y., *Advanced Robotics – Redundancy and Optimization*, Addison-Wesley, Boston, 1991
- [76] Nitzberg M., Shiota T., *Nonlinear image filtering with edge and corner enhancement*, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 14, pp. 826-833, 1992
- [77] Nordström N., *Biased anisotropic diffusion – a unified regularization and diffusion approach to edge detection*, Proc. of the first European Conference on Computer vision., pp. 18 – 27 , 1990
- [78] Perona P., Malik J., *A Network for Edge Detection and Scale-Space*, Proc. of ISCAS '88, pp. 2565-2568, Helsinki, 1988
- [79] Perona P., Malik J., *Scale-Space and Edge Detection Using Anisotropic Diffusion*, IEEE Trans. On Pattern Analysis and Machine Intelligence, Vol. 12, No. 7, 1990
- [80] Pina A., Cerezo E., Serón F. J., *Computer animation: from avatars to unrestricted autonomous actors (A survey on replication and modelling mechanisms)*, Computer & Graphics, Vol. 24, pp. 297-311, 2000
- [81] Rabiner L., Rosenberg A., Levinson S., *Considerations in dynamic time warping algorithms for discrete word recognition*, IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. 26, No. 6, 1978
- [82] Rekeczky C., Roska T., Ushida A., *Cnn-Based Difference-Controlled Adaptive Nonlinear Image Filters*, International Journal of Circuit Theory and Applications, 1998
- [83] Rosenhahn B., Kersting U. G., Smith A. W., Gurney J. K., Brox T., Klette R., *A system for marker-less human motion estimation*, Pattern Recognition. Lecture Notes in Computer Science, Vol. 3663, pp. 230-237, Springer, Berlin, 2005
- [84] Samal A., Edwards J., *Generalized Hough transform for natural shapes*, Pattern Recognition Letters, Vol. 18, No. 5, pp. 473-480, Elsevier, 1997

- [85] Sánchez-Ortiz G. I., Rueckert D., Burger P., *Knowledge-Based Anisotropic Diffusion of Vector-Valued 4-Dimensional Cardiac MR Images*, Proc. of the British Machine Vision Conference, pp. 605-614, UK, 1996
- [86] Sánchez-Ortiz G. I., Rueckert D., Burger P., *Knowledge-based tensor anisotropic diffusion of cardiac magnetic resonance images*, Medical Image Analysis, Vol. 3, No. 1, Elsevier, 1999
- [87] Sapiro G., Ringach L., *Anisotropic Diffusion of Multivalued Images with Applications to Color Filtering*, IEEE Trans. on Image Processing, Vol. 5, No. 11, 1996
- [88] Schlag J., *Using Geometric constructions to interpolate orientation with quaternions*, Graphics Gems II, pp. 377-380, Academic Press, 1991
- [89] Shoemake K., *Animating Rotation with Quaternion Curves*, Computer Graphics, Vol. 19, No. 3, 1985
- [90] Shoemake K., *Uniform Random Rotations*, Graphics Gems III, pp. 124-132, Academic Press, 1992
- [91] Shoemake K., *Quaternions*, Praca niepublikowana, University of Philadelphia, 1994
- [92] Silva F. W. da, Velho L., Cavalcanti P. R., Gomen J., *An architecture for Motion Capture Based Animation*, Proc. of IEEE Conference on Computer Graphics and Image Processing, Brazylia, 1997
- [93] Skarbek W., *Metody reprezentacji obrazów cyfrowych*, Akademicka Oficyna Wydawnicza PLJ, Warszawa, 1993
- [94] Song Z., Chen Y., Liang J., Uciński D., *Optimal mobile sensor motion planning under nonholonomic constraints for parameter estimation of distributed systems*, IEEE Conference on Intelligent Robots and Systems, pp. 3163-3168, 2005
- [95] Strzodka R., Ihrke I., Magnor M., *A Graphics Hardware Implementation of the Generalized Hough Transform for fast Object Recognition, Scale, and 3D Pose Detection*, Proc. of the 12<sup>th</sup> International Conference on Image Analysis and Processing, IEEE, 2003
- [96] Tanie H., Yamane K., Nakamura, Y., *High Marker Density Motion Capture by Retroreflective Mesh Suit*, Proc. of IEEE Conf. on Robotics and Automation, pp. 2884 – 2889, 2005
- [97] Teboul S., Blanc-Féraud L., Aubert G., Barlaud M., *Variational Approach for Edge-Preserving Regularization Using Coupled PDE's*, IEEE Trans. on Image Processing, Vol. 7, No. 3, 1998
- [98] Thomas S. W., *Decomposing a matrix into simple transformations*, Graphics Gems II, pp. 320-323, Academic Press, 1991
- [99] Tian T. Y., Shah M., *Recovering 3D Motion of Multiple Objects Using Adaptive Hough Transform*, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19, No. 10, 1997
- [100] Uciński D., *Optimal sensor location for parameter estimation of distributed processes*, International Journal of Control, Vol. 73, No. 13, pp. 1235-1248, 2000
- [101] Ude A., *Filtering in a unit quaternion space for model-based object tracking*, Robotics and Autonomous Systems, Vol. 28, pp. 163-172, Elsevier, 1999
- [102] Ulrich M., Steger C., Baumgartner A., Ebner H., *Real-time Object Recognition Using a Modified Generalized Hough Transform*, Pattern Recognition, Vol. 26, No. 11, pp. 2557-2570, 2003
- [103] Vinci L., *Quaternions and Rotations in 3-Space: The Algebra and its Geometric Interpretation*, University of North Carolina at Chapel Hill, 2001
- [104] Weickert J., *Scale-Space Properties of Nonlinear Diffusion Filtering with a Diffusion Tensor*, Report, University of Kaiserslautern, 1994
- [105] Weickert J., *Theoretical Foundations of Anisotropic Diffusion in Image Processing*, Computing Supplement, Vol. 11, pp. 221-236, 1996
- [106] Weickert J., *A Review of Nonlinear Diffusion Filtering*, w: Scale-Space Theory in Computer Vision, LNCS, Springer-Verlag, Berlin, 1997



- [107] Weickert J., *Anisotropic Diffusion in Image Processing*, B.G. Teubner, Stuttgart, 1998
- [108] Woźnicki J., *Podstawowe techniki przetwarzania obrazu*, Wydawnictwa Komunikacji i Łączności, Warszawa 1996
- [109] Yacoub B., Jolion J.-M., *Hierarchical line extraction*, IEE Proceedings – Vision, Image and Signal Processing, Vol. 142, No. 1, pp. 7 – 14, 1995
- [110] Yamane K., Nakamura Y., *Dynamics Filter - Concept and Implementation of On-Line Motion Generator for Human Figures*, IEEE Trans. on Robotics and Automation, Vol. 19, No. 3, pp. 421-432, 2003
- [111] Yamane K., Nakamura Y., *Natural Motion Animation through Constraining and Deconstraining at Will*, IEEE Trans. On Visualization and Computer Graphics, Vol. 9, No. 3, 2003
- [112] You Y.L., Kaveh M., *Image Enhancement Using Fourth Order Partial Differential Equations*, IEEE Trans. on Image Processing, Vol. 9, No. 10, pp. 1723-1730, 2000
- [113] Żorski W., *Metody segmentacji obrazów oparte na transformacie Hougha*, Wojskowa Akademia Techniczna, Warszawa, 2000

## Publikacje własne

Najważniejsze rezultaty otrzymane w ramach rozprawy doktorskiej zostały przedstawione w następujących publikacjach:

- [1] Jabłoński B., *Anisotropic filtering of multidimensional rotational trajectories as a generalization of 2D diffusion process*, Multidimensional systems and signal processing, artykuł w recenzji, 2007
- [2] Jabłoński B., Klempous R., Kulbacki M., *PDE-based filtering of motion sequences*, Journal of Computational and Applied Mathematics, vol. 189, no. 1/2, s. 660-675, Elsevier, 2006
- [3] Jabłoński B., *Introduction extra knowledge into filtering process*, praca zaakceptowana i wygłoszona w ramach konferencji EURO 2006, artykuł w recenzji
- [4] Jabłoński B., Klempous R., Majchrzak D., *Feasibility analysis of human motion identification using motion capture*, Proc. of the International Conference on Modelling, Identification and Control, Acta Press, 2006
- [5] Jabłoński B., Klempous R., Majchrzak D., *Models and methods for biometric motion identification*, Annales UMCS, vol. 4, Lublin, 2006
- [6] Jabłoński B., *Porównywanie generatorów ruchu animowanych postaci ludzkich*, w *Diversitas Cybernetica*, red. R. Klempous, Wydawnictwa Komunikacji i Łączności, Warszawa, 2005
- [7] B. Jabłoński, R. Klempous, Majchrzak D., *Analiza możliwości identyfikacji ruchu postaci w oparciu o technikę Motion Capture*, Festiwal informatyki jutra. XXI Jesienne Spotkania PTI, 2005
- [8] Jabłoński B., Kulbacki M., *Grupowa miara podobieństwa, jako metoda porównywania charakteru ruchów animowanych postaci ludzkich*, w *Zastosowania i wdrożenia systemów informatycznych*, red. Grabara J., Nowak J., PTI, Katowice, 2005
- [9] Kulbacki M., Klempous R., Jabłoński B., *Badania nad wykorzystaniem technologii Motion Capture do tworzenia realistycznych animacji postaci*, w *Systemy sterowania* red. Greblicki W., Smutnicki C., Wydawnictwa Komunikacji i Łączności, Warszawa, 2005
- [10] Kulbacki M., Jabłoński B., Klempous R., Segen J., *Learning from Examples and Comparing Models of Human Motion*, Journal of Advanced Computational Intelligence, vol. 8, no. 5, 2004
- [11] Jabłoński B., Kulbacki M., Klempous R., Segen J., *Methods for comparison of animated motion generators*, Proc. of IEEE International Conference on Computational Cybernetics, Węgry, 2003
- [12] Kulbacki M., Jabłoński B., Klempous R., Segen J., *Multimodel approach to human motion designing.*, Proc. of IEEE International Conference on Computational Cybernetics, Węgry, 2003
- [13] Bagiński P., Jabłoński B., Klempous R., Kulbacki M., Rawicki J., Segen J., *Synthesizing realistic animation of human motion*, 9th International Workshop on Computer Aided Systems Theory. EuroCAST 2003, Extended abstracts, Spain, 2003

## Spis ilustracji

RYSUNEK 1-1: DIAGRAM PRZEDSTAWIAJĄCY SCHEMAT KONSTRUKCJI PREZENTOWANEJ ROZPRAWY.....	12
RYSUNEK 2-1: OBRAZ TESTOWY NR 1 (KWIAT).....	27
RYSUNEK 2-2: WYNIK DZIAŁANIA FILTRACJI OPARTEJ NA LINIOWYM RÓWNANIU DYFUZJI, $\sigma=0.48$ (ODPOWIEDNIO 50, 100, 150 ITERACJI).....	27
RYSUNEK 2-3: WYNIK DZIAŁANIA FILTRACJI UŚREDNIAJĄCEJ, ROZMIAR OKNA 3x3 (ODPOWIEDNIO 50, 100, 150 ITERACJI).....	27
RYSUNEK 2-4: FUNKCJE $G_1$ (LINIA NIEBIESKA), $G_2$ (LINIA ZIELONA) DLA PARAMETRU $K=10$ .....	30
RYSUNEK 2-5: OBRAZ TESTOWY NR 1 (ROZMIAR 400x300), DETEKcja KRAWĘDZI, PRZEBIEG FUNKCJI JEDNEJ LINII OBRAZU ( $y=105$ , LINIA ZAZNACZONA KOLOREM CZERWONYM).....	38
RYSUNEK 2-6: WYNIK DZIAŁANIA FILTRU LINIOWEGO DLA $\sigma=0.8$ ; ITERACJE 5, 10, 30 (DOLNY RZĄD – DETEKcja KRAWĘDZI TYPU NORMA GRADIENTU).....	38
RYSUNEK 2-7: WYNIK DZIAŁANIA FILTRU PM DLA $K=12$ , $G_1$ ; ITERACJE 5, 10, 30.....	39
RYSUNEK 2-8: PRZYKŁADOWY PRZEBIEG FUNKCJI $U(x,y)$ DLA JEDNEJ LINII OBRAZU $y=105$ (NIEBIESKA – OBRAZ ORYGINALNY; ZIELONA – FILTRACJA LINIOWA, $IT=30$ ; CZERWONA – FILTRACJA PM, $IT=30$ ).....	40
RYSUNEK 2-9: SYNTETYCZNY OBRAZ TESTOWY: ORYGINALNY, ZASZUMIONY $\sigma_N=40$ , ZASZUMIONY $\sigma_N=400$ .....	41
RYSUNEK 2-10: MIARA SNR DLA FILTRACJI LINIOWEJ, ZASZUMIENIE OBRAZU TESTOWEGO $\sigma_N=40$ .....	41
RYSUNEK 2-11: MIARA SNR DLA FILTRACJI LINIOWEJ, ZASZUMIENIE OBRAZU TESTOWEGO $\sigma_N=400$ .....	42
RYSUNEK 2-12: WYBRANE ITERACJE (5, 20, 40) FILTRACJI LINIOWEJ O PARAMETRZE $\sigma=0.5$ DLA OBRAZÓW O ZASZUMIENIU $\sigma_N=40, 400$ .....	42
RYSUNEK 2-13: MIARA SNR DLA FILTRACJI MODELU PM, FUNKCJA $G_1$ , ZASZUMIENIE OBRAZU TESTOWEGO $\sigma_N=40$ .....	43
RYSUNEK 2-14: WYNIKI FILTRACJI DLA 150 ITERACJI Z PARAMETREM $K=20, 50$ ORAZ DLA WERSJI ADAPTACYJNEJ, FUNKCJA $G_1$ . POCZĄTKOWE ZASZUMIENIE OBRAZU $\sigma_N=40$ .....	44
RYSUNEK 2-15: WYNIKI FILTRACJI DLA WYBRANYCH ITERACJI (10, 30, 150) Z PARAMETREM $K=100$ , FUNKCJA $G_1$ . POCZĄTKOWE ZASZUMIENIE OBRAZU $\sigma_N=40$ .....	44
RYSUNEK 2-16: MIARA SNR DLA FILTRACJI MODELU PM, FUNKCJA $G_1$ , ZASZUMIENIE OBRAZU TESTOWEGO $\sigma_N=400$ .....	45
RYSUNEK 2-17: WYNIKI FILTRACJI DLA WYBRANYCH ITERACJI (10, 30, 150) FUNKCJA $G_1$ ; GÓRNY RZĄD - $K=100$ , DOLNY RZĄD – $K$ ADAPTACYJNE. POCZĄTKOWE ZASZUMIENIE OBRAZU $\sigma_N=400$ .....	45
RYSUNEK 2-18: WYNIKI FILTRACJI PO 800 ITERACJACH, $K$ ADAPTACYJNE, FUNKCJA $G_1$ ; DETEKcja KRAWĘDZI; FUNKCJA OBRAZU DLA LINII $y=150$ . POCZĄTKOWE ZASZUMIENIE OBRAZU $\sigma_N=400$ .....	46
RYSUNEK 2-19: MIARA SNR DLA FILTRACJI MODELU PM, FUNKCJA $G_2$ , ZASZUMIENIE OBRAZU TESTOWEGO $\sigma_N=40$ .....	47
RYSUNEK 2-20: WYNIKI FILTRACJI DLA 30 ITERACJI Z PARAMETREM $K=20, 50$ ORAZ DLA WERSJI ADAPTACYJNEJ, FUNKCJA $G_2$ . POCZĄTKOWE ZASZUMIENIE OBRAZU $\sigma_N=40$ .....	47
RYSUNEK 2-21: MIARA SNR DLA FILTRACJI MODELU PM, FUNKCJA $G_2$ , ZASZUMIENIE OBRAZU TESTOWEGO $\sigma_N=400$ .....	47
RYSUNEK 2-22: WYNIKI FILTRACJI DLA 150 ITERACJI, $K$ ADAPTACYJNE, FUNKCJA $G_2$ ; DETEKcja KRAWĘDZI; FUNKCJA OBRAZU DLA LINII $y=150$ . POCZĄTKOWE ZASZUMIENIE OBRAZU $\sigma_N=400$ .....	48
RYSUNEK 2-23: MIARA SNR DLA FILTRACJI MODELU PM, FUNKCJE $G_2, G_1$ , DLA RÓŻNYCH WARTOŚCI PARAMETRU $L=\lambda/4$ . ZASZUMIENIE OBRAZU TESTOWEGO $\sigma_N=40$ .....	49
RYSUNEK 2-24: ZASZUMIONY OBRAZ TESTOWY NR 1, DETEKcja KRAWĘDZI, PRZEBIEG FUNKCJI WYBRANEJ LINII OBRAZU ( $y=105$ ). ZASZUMIENIE OBRAZU $\sigma_N=16$ .....	50
RYSUNEK 2-25: MIARA SNR DLA RÓŻNEGO TYPU FILTRACJI – RZECZYWISTY OBRAZ TESTOWY O ZASZUMIENIU $\sigma_N=16$ .....	50
RYSUNEK 2-26: WYNIK DZIAŁANIA FILTRÓW: LINIOWEGO $\sigma=0.8$ (ITERACJA 180), PM $G_1$ ADAPTACYJNY (ITERACJA 5), PM $G_2$ ADAPTACYJNY (ITERACJA 5); DOLNY RZĄD – DETEKcja KRAWĘDZI.....	52
RYSUNEK 2-27: WYNIK DZIAŁANIA FILTRU NIELINIOWEGO $G_1$ : $K=10$ , $IT=53$ , $SNR=13,39$ dB; $K=20$ , $IT=11$ , $SNR=18,16$ dB.....	52
RYSUNEK 2-28: WYNIK DZIAŁANIA FILTRU NIELINIOWEGO $G_2$ : $K=5$ , $IT=36$ , $SNR=18,45$ dB; $K=10$ , $IT=13$ , $SNR=19,88$ dB.....	52
RYSUNEK 3-1: TRANSFORMACJA HOUGHGA DLA LINII PROSTYCH – PO LEWEJ PRZYKŁADOWY OBRAZ, PO PRAWIEJ PRZESTRZEŃ HOUGHGA Z ZAZNACZONYMI MAKSYMAMI.....	62
RYSUNEK 3-2: PRZYKŁADOWY OBRAZ, ORYGINALNA PRZESTRZEŃ HOUGHGA ORAZ WYNIK DZIAŁANIA FILTRU TYPU BUTTERFLY.....	63
RYSUNEK 3-3: PRZYKŁADOWY OBRAZ ORAZ ZE ZNIEKSZTAŁCONĄ LINIĄ, ZASZUMIONY SZUMEM $\sigma_N=20$ ; PRZESTRZEŃ HOUGHGA ORYGINALNA ORAZ PO FILTRACJI TYPU BUTTERFLY.....	63
RYSUNEK 3-4: OGÓLNY SCHEMAT SPRZĘŻENIA PROCESU DYFUZJI Z PRZESTRZENIĄ HOUGHGA.....	65
RYSUNEK 3-5: OSZACOWANIE ZASZUMIENIA OBRAZU $N(t)$ DLA KOLEJNYCH KROKÓW FILTRACJI PRZY UŻYCIU MODELU PERONA-MALIK (OBRAZ TESTOWY NR 1, POCZĄTKOWE ZASZUMIENIE $\sigma_N=400$ ).....	70
RYSUNEK 3-6: SYNTETYCZNY OBRAZ TESTOWY NR 2 (PASKI); DETEKcja KRAWĘDZI; ZASZUMIONY OBRAZ TESTOWY $\sigma_N=$	

100.....	73
RYSUNEK 3-7: WARTOŚĆ WSPÓŁCZYNNIKA $H_F$ W ZALEŻNOŚCI OD PARAMETRÓW $H_\mu, H_\sigma$ (OZNACZONYCH ODPOWIEDNIO <i>MEAN, SIGMA</i> ). OBRAZ TESTOWY NR 1, POCZĄTKOWE ZASZUMIENIE $\sigma_N = 400$ .	73
RYSUNEK 3-8: WYNIK FILTRACJI PM PO 300 ITERACJACH DLA OBRAZU TESTOWEGO NR 1 (ZASZUMIENIE $\sigma_N = 400$ ). KOLUMNA 1 – FUNKCJA $G_1$ , KOLUMNA 2 – FUNKCJA $G_2$ .	75
RYSUNEK 3-9: WYNIK FILTRACJI SPRĘŻONEJ PO 300 ITERACJACH DLA OBRAZU TESTOWEGO NR 1 (ZASZUMIENIE $\sigma_N = 400$ ). KOLUMNA 1 – FUNKCJA $G_1$ , KOLUMNA 2 – FUNKCJA $G_2$ . WIERSZE ODPOWIEDNIO: $H_\sigma = 10$ ; $H_\sigma = 20$ .	75
RYSUNEK 3-10: JAKOŚĆ SYGNAŁU PO FILTRACJI WEDŁUG MIARY SNR DLA RÓŻNYCH WARTOŚCI PARAMETRU $H_\sigma$ ADAPTACYJNEGO SPRĘŻENIA ADDYTYWNEGO.	76
RYSUNEK 3-11: WARTOŚĆ WSPÓŁCZYNNIKA $H_F$ DLA ANALIZOWANEGO PROCESU FILTRACJI SPRĘŻONEJ. OBRAZ TESTOWY NR 1, POCZĄTKOWE ZASZUMIENIE $\sigma_N = 400$ .	78
RYSUNEK 3-12: WYNIK NIELINIOWEJ FILTRACJI STANDARDOWEJ (KOLUMNA LEWA) ORAZ SPRĘŻONEJ (KOLUMNA PRAWA) PO ODPOWIEDNIO 20, 40, 100, 200 ITERACJACH DLA OBRAZU TESTOWEGO NR 1 (ZASZUMIENIE $\sigma_N = 400$ ).	79
RYSUNEK 3-13: DETEKCCJA KRAWĘDZI TYPU SOBEL DLA REZULTATÓW NIELINIOWEJ FILTRACJI STANDARDOWEJ (KOLUMNA LEWA) ORAZ SPRĘŻONEJ (KOLUMNA PRAWA) PO ODPOWIEDNIO 20, 40, 100, 200 ITERACJACH. OBRAZ TESTOWY NR 1 (ZASZUMIENIE $\sigma_N = 400$ ).	80
RYSUNEK 3-14: WYNIK NIELINIOWEJ FILTRACJI PO 300 ITERACJACH, $G_1$ . PIERWSZY WIERSZ: FILTRACJA BEZ SPRĘŻENIA, SPRĘŻENIE $H_F = 0.1$ , SPRĘŻENIE $H_F = 0.3$ . DRUGI WIERSZ: SPRĘŻENIE $H_F = 0.5$ , SPRĘŻENIE $H_F = 0.7$ , SPRĘŻENIE $H_F = 0.9$ .	82
RYSUNEK 3-15: WYNIK FILTRACJI $C_2$ PO 300 ITERACJACH DLA OBRAZU TESTOWEGO NR 1 (ZASZUMIENIE $\sigma_N = 400$ ). WIERSZ DRUGI – DETEKCCJA KRAWĘDZI TYPU SOBEL. KOLUMNA 1 – FILTRACJA BEZ SPRĘŻENIA, KOLUMNA 2 – FILTRACJA SPRĘŻONA $H_F = 0.3$ , KOLUMNA 3 – FILTRACJA SPRĘŻONA $H_F = 0.9$ .	83
RYSUNEK 3-16: WYNIK FILTRACJI PO 400 ITERACJACH DLA OBRAZU TESTOWEGO NR 2 (ZASZUMIENIE $\sigma_N = 100$ ). WIERSZ DRUGI – DETEKCCJA KRAWĘDZI TYPU SOBEL. KOLUMNA 1 – FILTRACJA BEZ SPRĘŻENIA, KOLUMNA 2 – FILTRACJA SPRĘŻONA $H_F = 0.3$ , KOLUMNA 3 – FILTRACJA SPRĘŻONA $H_F = 0.5$ .	85
RYSUNEK 3-17: WYNIK FILTRACJI PO 400 ITERACJACH DLA OBRAZU TESTOWEGO NR 2 (ZASZUMIENIE $\sigma_N = 100$ ). OBRAZ PO LEWEJ: STAŁY WSPÓŁCZYNNIK $H_F = 0.5$ , OBRAZ ŚRODKOWY: WSPÓŁCZYNNIK ADAPTACYJNY $H_\sigma = 15$ , OBRAZ PO PRAWY: WSPÓŁCZYNNIK ADAPTACYJNY $H_\sigma = 25$ .	86
RYSUNEK 3-18: WYNIK FILTRACJI SPRĘŻONEJ $H_F = 0.3$ (KOLUMNA LEWA), DETEKCCJA KRAWĘDZI Z NAŁOŻONĄ WIZUALIZACJĄ $HT^{-1}$ (KOLUMNA ŚRODKOWA), WIZUALIZACJA TRANSFORMATY $HT^{-1}$ UŻYTEJ W SPRĘŻENIU (KOLUMNA PRAWA). WIERSZE ODNOŚĄ SIĘ ODPOWIEDNIO DO 0, 2, 4, 6, 10, 50 ITERACJI. OBRAZ TESTOWY NR 2 (ZASZUMIENIE $\sigma_N = 100$ ).	87
RYSUNEK 4-1: PRZEKSZTAŁCENIE UKŁADU WSPÓLRZĘDNYCH $B$ WZGLĘDEM $A$ PRZEZ ZŁOŻENIE TRANSLACJI I ROTACJI.	98
RYSUNEK 4-2: PARAMETRYZACJA ROTACJI PRZEZ ZŁOŻENIE TRZECH ROTACJI WOKÓŁ OSI USTALONEGO UKŁADU WSPÓLRZĘDNYCH.	101
RYSUNEK 4-3: PARAMETRYZACJA ROTACJI PRZEZ ZŁOŻENIE TRZECH ROTACJI OPISANYCH KĄTAMI EULERA (WOKÓŁ OSI PRZEKSZTAŁCANEGO UKŁADU WSPÓLRZĘDNYCH).	101
RYSUNEK 4-4: WYSTĘPOWANIE EFEKTU <i>GIMBAL LOCK</i> .	102
RYSUNEK 4-5: ORIENTACJA GLOBALNEGO UKŁADU WSPÓLRZĘDNYCH ORAZ PRZEDSTAWIENIE PODSTAWOWYCH ROTACJI PRZEZ WIZUALIZACJĘ POŁOŻENIA PRZYKŁADOWEGO CZŁONU MANIPULATORA.	104
RYSUNEK 4-6: WIZUALIZACJA PRZYKŁADOWYCH ROTACJI PRZY POMOCY SFERY ROTACJI.	104
RYSUNEK 4-7: ROTACJA WEKTORA $\vec{r}$ WOKÓŁ OSI $\vec{n}$ O KĄT $\varphi$ .	108
RYSUNEK 4-8: ZMIANA ORIENTACJI LOKALNEGO UKŁADU WSPÓLRZĘDNYCH W OPISIE KWATERNIONOWYM.	111
RYSUNEK 4-9: INTERPOLACJA ORIENTACJI WEKTORÓW PRZY UŻYCIU KWATERNIONÓW.	113
RYSUNEK 4-10: INTERPOLACJA ORIENTACJI DLA KWATERNIONÓW W PRZESTRZENIACH $S^3, TS^3$ (ILUSTRACJA WŁASNA NA PODSTAWIE [48]).	114
RYSUNEK 4-11: WIZUALIZACJA PRZYKŁADOWEGO WYNIKU GENERACJI 200 LOSOWYCH WARTOŚCI W PRZESTRZENI KWATERNIONÓW (ROZKŁAD NORMALNY, $\sigma = 0,15$ ).	123
RYSUNEK 4-12: PRZYKŁADOWE TESTOWE TRAJektorie ROTACJI (WZORCOWA PO LEWEJ, TESTOWA PO PRAWY) PODOBNE ZE WZGLĘDU NA SUBIEKTYWNE PORÓWNANIE.	134
RYSUNEK 4-13: WYNIK DZIAŁANIA ALGORYTMU DTW DLA LOKALNEJ MIARY UWZGLĘDNIĄCEJ JEDYNIIE KWATERNIONOWĄ ODLEGŁOŚĆ ELEMENTÓW TRAJektorii.	135
RYSUNEK 4-14: WYNIK DZIAŁANIA ALGORYTMU QDTW DLA UOGÓLNIONEJ LOKALNEJ MIARY PODOBIENSTWA ELEMENTÓW.	135
RYSUNEK 4-15: RODZAJE POŁĄCZEŃ RUCHOWYM DLA PAR NIŻSZEGO RZĘDU [18].	138
RYSUNEK 4-16: ZMIANA ORIENTACJI LOKALNYCH UKŁADÓW WSPÓLRZĘDNYCH PRZYPORZĄDKOWANYCH DO CZŁONÓW.	139
RYSUNEK 4-17: LICZBA ROZWIĄZAŃ ZADANIE KINEMATYKI ODWROTNEJ W ZALEŻNOŚCI OD POŁOŻENIA PUNKTU DOCELOWEGO (PUNKT DOCELOWY ZAZNACZONO KOLOREM CZERWONYM).	141
RYSUNEK 4-18: ZŁOŻONY ŁAŃCUCH KINEMATYCZNY NA PODSTAWIE HIERARCHICZNEGO MODELU SZKIELETU	

.....	142
RYSUNEK 4-19: HIERARCHIA CZŁONÓW W MODELU SZKIELETU LUDZKIEGO.....	145
RYSUNEK 5-1: PRZYKŁADOWY PRZEBIEG PROCESU FILTRACJI <i>LINQUAT</i> – KOLEJNE FRAGMENTY ODPOWIEDNIO PRZEDSTAWIAJĄ: TRAJEKTORIA ORYGINALNA, PO 10, PO 50 ITERACJACH ( $c=0,5$ ). .....	161
RYSUNEK 5-2: SEKWENCJA UŻYTA DO GEOMETRYCZNEJ INTERPRETACJI FILTRACJI LINIOWEJ. KOLOREM ZIEŁONYM ZAZNACZONO POŁOŻENIE REFERENCYJNE. KOLOREM NIEBIESKIM ZAZNACZONO ORIENTACJE UKŁADU W KOLEJNYCH CHWILACH CZASOWYCH. ....	176
RYSUNEK 5-3: WYBRANY FRAGMENT TRAJEKTORII, NA KTÓRYM WYRÓŻNIONY ZOSTAŁA ZAKŁÓCONA ORIENTACJA UZYSKANA PRZEZ ROTACJĘ OPISANĄ KWATERNIONEM $Q_t$ . ....	176
RYSUNEK 5-4: KROK 4 ALGORYTMU <i>LINQUAT</i> : INTERPOLACJA POŁOŻENIA SĄSIADUJĄCYCH ELEMENTÓW – RÓWNANIE (5 – 83). .....	177
RYSUNEK 5-5: WIZUALIZACJA KROKU 5 ALGORYTMU <i>LINQUAT</i> : INTERPOLACJA POŁOŻENIA ELEMENTU PRZEKSZTAŁCANEGO W ZALEŻNOŚCI OD WSPÓŁCZYNNIKA DYFUZJI – RÓWNANIE (5 – 84). .....	177
RYSUNEK 5-6: PRZYKŁADOWA SEKWENCJA ORIENTACJI UKŁADU PO WYKONANIU JEDNEJ PEŁNEJ ITERACJI ALGORYTMU <i>LINQUAT</i> . .....	178
RYSUNEK 5-7: SEKWENCJA UŻYTA DO GEOMETRYCZNEJ INTERPRETACJI FILTRACJI NIELINIOWEJ.....	178
RYSUNEK 5-8: WYBRANY FRAGMENT TRAJEKTORII, W KTÓRYM WYSTĘPUJE NAGŁA ZMIANA PRĘDKOŚCI KĄTOWEJ RUCHU. ....	179
RYSUNEK 5-9: WYBRANY FRAGMENT TRAJEKTORII – INTERPOLACJA W KIERUNKU POŁOŻENIA POPRZEDNIEGO I NASTĘPNEGO (KROK 19 I 20 ALGORYTMU). .....	179
RYSUNEK 5-10: WYBRANY FRAGMENT TRAJEKTORII – INTERPOLACJA DO POŚREDNIEGO KWATERNIONU WYNIKOWEGO. ....	180
RYSUNEK 5-11: WYBRANY FRAGMENT TRAJEKTORII – UWZGLĘDNIENIE CZYNNIKA TLUMIĄCEGO. ....	180
RYSUNEK 5-12: SEKWENCJA UŻYTA DO GEOMETRYCZNEJ INTERPRETACJI FILTRACJI ANIZOTROPOWEJ. NA RYSUNKU ZAZNACZONO CHWILE CZASOWE, W KTÓRYCH DOKONYWANA JEST ZMIANA KIERUNKU RUCHU. KOLUMNĄ PRAWĄ – PRZEBIEG TESTOWY Z WPROWADZONYM SYGNAŁEM ZAKŁÓCAJĄCYM. ....	181
RYSUNEK 5-13: WYBRANE CHWILE CZASOWE TRAJEKTORII, DLA KTÓRYCH BĘDZIE PRZEPROWADZONA ANALIZA FILTRACJI ANIZOTROPOWEJ. NALEŻY ZWRÓCIĆ UWAGĘ, ŻE W WYBRANYCH MOMENTACH KIERUNEK RUCHU ZMIENIA SIĘ W RÓŻNY SPOSOB. ....	181
RYSUNEK 5-14: REALIZACJA ITERACJI FILTRACJI ANIZOTROPOWEJ DLA CHWILI CZASOWEJ $t=13$ PRZEBIEGU TESTOWEGO. ....	182
RYSUNEK 5-15: REALIZACJA ITERACJI FILTRACJI ANIZOTROPOWEJ DLA CHWILI CZASOWEJ $t=13$ PRZEBIEGU TESTOWEGO. ....	182
RYSUNEK 5-16: OSZACOWANIE LOKALNEJ KRZYWIZNY TRAJEKTORII DLA SYGNAŁU TESTOWEGO (SZCZEGÓŁOWY OPIS W TEKŚCIE). ....	183
RYSUNEK 6-1: TRÓJWYMIAROWA WIZUALIZACJA TRAJEKTORII TESTOWEJ NR 1 ( <i>QUATEDGE</i> ). .....	197
RYSUNEK 6-2: WIZUALIZACJA TRAJEKTORII TESTOWEJ NR 1 ( <i>QUATEDGE</i> ) PRZY UŻYCIU SFERY ROTACJI. ....	198
RYSUNEK 6-3: TRÓJWYMIAROWA WIZUALIZACJA ZMODYFIKOWANEJ TRAJEKTORII TESTOWEJ NR 1A. ....	199
RYSUNEK 6-4: TRÓJWYMIAROWA WIZUALIZACJA TRAJEKTORII TESTOWEJ NR 2 ( <i>QUATJUMPS</i> ). .....	199
RYSUNEK 6-5: WIZUALIZACJA TRAJEKTORII TESTOWEJ NR 2 ( <i>QUATJUMPS</i> ) PRZY UŻYCIU SFERY ROTACJI.....	200
RYSUNEK 6-6: WIZUALIZACJA TRAJEKTORII TESTOWEJ NR 3 ( <i>QUATMULTI</i> ) PRZY UŻYCIU SFERY ROTACJI.....	201
RYSUNEK 6-7: WIZUALIZACJA TESTOWEJ SEKWENCJI RUCHU NR 1 – NAZWA ORYGINALNA <i>RUN LOOPABLE</i> (DŁUGOŚĆ 17 KLATEK). ....	202
RYSUNEK 6-8: WIZUALIZACJA TESTOWEJ SEKWENCJI RUCHU NR 2 – NAZWA ORYGINALNA <i>SPRINT TO WALK</i> (DŁUGOŚĆ 74 KLATKI). ....	203
RYSUNEK 6-9: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI LQ W ZALEŻNOŚCI OD PARAMETRU $c$ – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1. ....	204
RYSUNEK 6-10: WIZUALIZACJA REZULTATÓW FILTRACJI <i>LINQUAT</i> – PRZYKŁAD NIESTABILNEGO PROCESU DLA PARAMETRU $c > 1$ ( $c = 1,005$ ). WYNIK FILTRACJI ODPOWIEDNIO PO 10 I 200 ITERACJACH. ....	205
RYSUNEK 6-11: WIZUALIZACJA REZULTATÓW FILTRACJI <i>LINQUAT</i> – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1 (W KOLEJNYCH WIERSZACH: SYGNAŁ ORYGINALNY, ZASUMIONY, ITERACJE 10 I 50). .....	206
RYSUNEK 6-12: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI LQ W ZALEŻNOŚCI OD PARAMETRU $c$ – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A. ....	207
RYSUNEK 6-13: WIZUALIZACJA REZULTATÓW FILTRACJI <i>LINQUAT</i> – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A (SYGNAŁ ORYGINALNY, ZASUMIONY, ITERACJE 5 I 50). ....	207
RYSUNEK 6-14: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI LQ W ZALEŻNOŚCI OD PARAMETRU STOPNIA ZASUMIENIA SYGNAŁU — SYNTETYCZNA TRAJEKTORIA TESTOWA NR 2, $c=0,5$ . ....	208
RYSUNEK 6-15: MAKSYMALNA WARTOŚĆ MIARY JAKOŚCI QSNR DLA FILTRACJI LQ W ZALEŻNOŚCI OD PARAMETRU $c$ I STOPNIA ZASUMIENIA SYGNAŁU. ....	209
RYSUNEK 6-16: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI LQ W ZALEŻNOŚCI OD PRZYJĘTEGO WARUNKU BRZEGOWEGO – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 2 (T2) ORAZ NR 3 (T3). ....	210
RYSUNEK 6-17: WIZUALIZACJA REZULTATÓW FILTRACJI <i>LINQUAT</i> W ZALEŻNOŚCI OD WARUNKU BRZEGOWEGO DLA	

TRAJEKTORII TESTOWEJ NR 2. PIERWSZY WIERSZ – TR. ORYGINALNA, WARUNEK A. DRUGI WIERSZ – WARUNEK B/D, WARUNEK C .....	211
RYSUNEK 6-18: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI NLQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKA WRAŻLIWOŚCI $K$ – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A. ....	213
RYSUNEK 6-19: WIZUALIZACJA REZULTATÓW FILTRACJI <i>NonLinQuat</i> – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A. TRAJEKTORIA PO ODPOWIEDNIO 0, 100, 300, 1000 ITERACJACH. ....	214
RYSUNEK 6-20: PORÓWNANIE REZULTATÓW FILTRACJI LQ (LEWA KOLUMNA) I NLQ (PRAWA KOLUMNA) DLA SEKWENCJI TESTOWEJ NR 1A. W KOLEJNYCH WIERSZACH ITERACJE: 25, 50, 100, 150. ....	216
RYSUNEK 6-21: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI NLQ W ZALEŻNOŚCI OD ADAPTACYJNEGO WSPÓŁCZYNNIKA WRAŻLIWOŚCI $K$ , FUNKCJA $G_2$ – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A. ....	217
RYSUNEK 6-22: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI NLQ W ZALEŻNOŚCI OD PARAMETRU $\lambda$ – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A. ....	218
RYSUNEK 6-23: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI NLQ W ZALEŻNOŚCI OD STOPNIA ZASZUMIENIA SYGNAŁU (ROZKŁAD NORMALNY) – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A. ....	219
RYSUNEK 6-24: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD ANIZOTROPOWEGO WSPÓŁCZYNNIKA WRAŻLIWOŚCI $K_{MAX}$ (FUNKCJA $G_1$ ) – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A. ....	221
RYSUNEK 6-25: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKA WRAŻLIWOŚCI $K_{MAX}$ (FUNKCJA $G_1$ ) – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 1A. ....	222
RYSUNEK 6-26: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKA WRAŻLIWOŚCI $K_{MAX}$ (FUNKCJA $G_1$ ) – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 2. ....	223
RYSUNEK 6-27: MAKSYMALNA WARTOŚĆ MIARY JAKOŚCI QSNR UZYSKANA W CZASIE 10 000 ITERACJI DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKÓW $K_{MAX}$ , $K_{AMAX}$ (FUNKCJA $G_1$ ). ....	224
RYSUNEK 6-28: NUMER ITERACJI, PRZY KTÓREJ OSIĄGNIĘTO MAKSYMALNĄ WARTOŚĆ MIARY JAKOŚCI QSNR DLA FILTRACJI AQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKÓW $K_{MAX}$ , $K_{AMAX}$ (FUNKCJA $G_1$ ). ....	224
RYSUNEK 6-29: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKA WRAŻLIWOŚCI $K_{AMAX}$ (FUNKCJA $G_2$ ) – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 2. ....	226
RYSUNEK 6-30: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKA WRAŻLIWOŚCI $K_{AMAX}$ (FUNKCJA $G_2$ ) – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 2. ....	227
RYSUNEK 6-31: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKA $\lambda$ – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 2. ....	229
RYSUNEK 6-32: MAKSYMALNA WARTOŚĆ MIARY JAKOŚCI QSNR UZYSKANA DLA FILTRACJI AQ PRZY WYBRANYM PARAMETRZE $\lambda$ . ....	229
RYSUNEK 6-33: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD STOPNIA ZASZUMIENIA TRAJEKTORII – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 2. ....	230
RYSUNEK 6-34: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD WYBRANEGO WARUNKU BRZEGOWEGO – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 2. ....	231
RYSUNEK 6-35: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ W ZALEŻNOŚCI OD WSPÓŁCZYNNIKA WRAŻLIWOŚCI $K_{AMAX}$ – SYNTETYCZNA TRAJEKTORIA TESTOWA NR 3. ....	233
RYSUNEK 6-36: ORYGINALNA ORAZ ZASZUMIONA TRAJEKTORIA TESTOWA NR 3. SZUM O ROZKŁADZIE NORMALNYM, $\sigma=0,05$ . ....	233
RYSUNEK 6-37: WIZUALIZACJA KOLEJNYCH KROKÓW FILTRACJI LQ (LEWA KOLUMNA) ORAZ AQ (PRAWA KOLUMNA). W WIERSZACH POKAZANA ODPOWIEDNIO: 10, 50, 200 ITERACJE. ....	234
RYSUNEK 6-38: POPRAWA JAKOŚCI SYGNAŁU DLA MAKSYMALNEJ WARTOŚCI QSNR ALGORYTMU AQ WZGLĘDEM MAKSYMALNEJ WARTOŚCI UZYSKANEJ FILTRACJĄ LQ. ....	236
RYSUNEK 6-39: ORYGINALNA ORAZ ZASZUMIONA TRAJEKTORIA TESTOWA NR 2. TRAJEKTORIA ZASZUMIONA – ROZKŁAD NORMALNY, $\sigma=0,05$ . ....	237
RYSUNEK 6-40: WIZUALIZACJA KOLEJNYCH KROKÓW FILTRACJI LQ (LEWA KOLUMNA) ORAZ AQ (PRAWA KOLUMNA). W WIERSZACH POKAZANA ODPOWIEDNIO ITERACJE NR: 10, 20, 50, 100. ....	238
RYSUNEK 6-41: WIZUALIZACJA TRAJEKTORII TESTOWEJ NR 2. LEWA GÓRA: TR. ORYGINALNA. PRAWA GÓRA: TR. ZASZUMIONA ( $\sigma=0,05$ ). LEWY DÓŁ: FILTRACJA LQ, 20 ITERACJI. PRAWY DÓŁ: FILTRACJA AQ, 200 ITERACJI. ....	239
RYSUNEK 6-42: FRAGMENT TRAJEKTORII TESTOWEJ NR 2A. REZULTATY FILTRACJI AQ ODPOWIEDNIO DLA ITERACJI NR 25, 50, 75, 150. ....	240
RYSUNEK 6-43: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ ORAZ LQ W ZALEŻNOŚCI OD STOPNIA ZASZUMIENIA SYGNAŁU – RZECZYWISTA, ZŁOŻONA TRAJEKTORIA TESTOWA <i>RUN LOOPABLE</i> . ....	242
RYSUNEK 6-44: MIARA JAKOŚCI QSNR DLA REZULTATÓW FILTRACJI AQ ORAZ LQ W ZALEŻNOŚCI OD STOPNIA ZASZUMIENIA SYGNAŁU – RZECZYWISTA, ZŁOŻONA TRAJEKTORIA TESTOWA <i>SPRINT TO WALK</i> . ....	243
RYSUNEK 6-45: ORYGINALNA ORAZ ZASZUMIONA TESTOWA SEKWENCJA RUCHU <i>RUN LOOPABLE</i> . SZUM O ROZKŁADZIE NORMALNYM, $\sigma=0,02$ . ....	244
RYSUNEK 6-46: WIZUALIZACJA KOLEJNYCH KROKÓW FILTRACJI LQ (LEWA KOLUMNA) ORAZ AQ (PRAWA KOLUMNA). W WIERSZACH POKAZANA ODPOWIEDNIO ITERACJE NR: 10, 20, 50, 200. ....	245
RYSUNEK 6-47: FRAGMENT SEKWENCJI <i>RUN LOOPABLE</i> . WERSJA ORYGINALNA, REZULTAT FILTRACJI LQ (20 ITERACJI) REZULTAT FILTRACJI AQ (200 ITERACJI). ....	246

## **Wykaz algorytmów**

ALGORYTM 1: TRANSFORMATA HOUGH A DLA LINII PROSTYCH NA OBRAZACH BINARNYCH.....	60
ALGORYTM 2: NIELINIOWY PROCES DYFUZJI SPRZĘŻONY Z PRZESTRZENIĄ HOUGH A .....	71
ALGORYTM 3: WPROWADZENIE LOSOWEGO ZAKŁÓCENIA DLA KWATERNIONÓW .....	122
ALGORYTM 4: PRZYGOTOWANIE TRAJEKTORII KWATERNIONÓW .....	160
ALGORYTM 5: FILTRACJA TRAJEKTORII KWATERNIONÓW TYPU LINQUAT .....	160
ALGORYTM 6: NIELINIOWA FILTRACJA TRAJEKTORII KWATERNIONÓW NONLINQUAT .....	167
ALGORYTM 7: NIELINIOWA ANIZOTROPOWA FILTRACJA TRAJEKTORII KWATERNIONÓW ANISOQUAT .....	173

## Wykaz oznaczeń

$\langle f, g \rangle$	Iloczyn skalarny wielkości $f, g$
$A$	Tablica akumulatora przestrzeni Hougha
$\alpha$	Parametr kątowy w transformacji Hougha dla linii prostych
$c(x, y, t)$	Funkcja skalarnego współczynnika dyfuzji
$c_1, c_2$	Współczynniki dyfuzji dla układu równań różniczkowych
$c_A$	Anizotropowy współczynnik dyfuzji
$c_c$	Współczynnik dyfuzji dla modelu sprzężonego
$C_1, C_2$	Wektory jasności pikseli dla obrazów 3-spektralnych
$C_w$	Ciepło właściwe środowiska
$\chi$	Powłoka wypukła
$d_E$	Odległość euklidesowa
$d_M$	Odległość Manhattan
$d_{\max}$	Odległość maksimum
$d(C_1, C_2)$	Odległość pomiędzy wektorami jasności pikseli $c_1$ oraz $c_2$
$d(p, q)$	Odległość pomiędzy trajektorią $p$ oraz $q$
$\operatorname{div}(\vec{F}) = \nabla \cdot \vec{F}$	Operator dywergencji pola wektorowego $\vec{F}$
$D$	Współczynnik przewodzenia ciepła (skalarny lub tensorowy)
$D^{(G)}$	Diagonalna macierz współczynników dyfuzji $D^{(G)}$
$D^{(W)}$	Macierz wag uwzględniająca strukturę geometryczną obrazu
$\delta$	Gęstość środowiska
$\delta(i, j)$	Element macierzy minimalnego kosztu dopasowania ścieżki
$\vec{e}_1^{(D)}, \vec{e}_2^{(D)}$	Wektory własne tensora/macierzy $D$
$E(x, y, t)$	Funkcja estymatora krawędzi
$E(q)$	Energia trajektorii $q$
$\eta_H$	Liczność struktur w odwzorowaniu odwrotnej transformaty Hougha
$f_I$	Funkcja intensywności pikseli definiująca obraz $I$ w punkcie $(i, j)$
$\overline{f_I}$	Wartość średnia jasności pikseli na obrazie $I$
$f_F(q)$	Funkcja przystosowania dla trajektorii $q$
$F(q_i)$	Przekształcenie $F$ operujące na $i$ -tym elemencie trajektorii $q$
$F_A(q_i)$	Przekształcenie typu <i>AnisoQuat</i> na elemencie trajektorii $q_i$
$F_L(q_i)$	Przekształcenie typu <i>LinQuat</i> na elemencie trajektorii $q_i$
$F_N(q_i)$	Przekształcenie typu <i>NonLinQuat</i> na elemencie trajektorii $q_i$
$F[f(x)](\omega)$	Transformata Fouriera funkcji $f(x)$
$\Phi(s)$	Funkcja strumienia dla pola temperatury $T$
$g(s)$	Funkcja stopująca, kontrolująca szybkość przebiegu dyfuzji
$\operatorname{grad}(f) = \nabla f$	Operator gradientu funkcji skalarnej $f$



$G(q)$	Funkcja celu określona dla trajektorii $q$
$\gamma(P, t)$	Funkcja wydajności przestrzennych źródeł ciepła
$h_r$	Krok dyskretyzacji dla wymiaru o indeksie $r$
$h_{(x_i, y_i)}(a_1, \dots, a_N)$	Funkcja przynależności punktu $(x_i, y_i)$ do krzywej o parametrach $(a_1, \dots, a_N)$
$h_f(t)$	Funkcja określająca stopień sprzężenia z transformatą Hougha
$h_{\max}$	Maksymalny dopuszczalny poziom sprzężenia z transformatą Hougha
$h_s$	Współczynnik określający charakter sprzężenia z transformatą Hougha
$h_\mu, h_\sigma$	Parametry kontrolujące poziom sprzężenia w zależności od zaszumienia
$H(I)$	Wysokość obrazu $I$ w punktach
$H$	Grupa kwaternionów
$\overset{\circ}{H}$	Podgrupa kwaternionów $H$ bez elementu $[0, (0, 0, 0)]$
$H_1$	Podgrupa kwaternionów jednostkowych (o normie 1)
$HT$	Transformacja Hougha
$HT_\psi^{-1}$	Odwrotna transformata Hougha przy parametrze $\psi$
$i$	Licznik, numer iteracji, krok algorytmu
$IS$	Zbiór indeksów siatki prostokątnej
$J(u)$	Tensor struktury wyznaczony dla funkcji $u$
$k_\alpha$	Krok kwantyzacji parametru $\alpha$ przestrzeni Hougha
$k_r$	Krok kwantyzacji parametru $r$ przestrzeni Hougha
$K$	Współczynnik kontrastu dla funkcji stopującej $g$
$K(t)$	Funkcja współczynnika kontrastu zależna od chwili czasowej $t$
$K_\sigma$	Jądro rozkładu Gaussa o odchyleniu standardowym $\sigma$ , średniej $\mu = 0$
$\kappa_s$	Zbiór pikseli znajdujących się w przestrzennym sąsiedztwie piksela $s$
$\kappa(q_i)$	Lokalna krzywizna trajektorii $q$ dla elementu $i$
$\xi$	Wartość progowa jasności pikseli dla transformacji Hougha
$l(i, j)$	Operator przypisania elementów $(i, j)$ do siebie
$L$	Ścieżka dopasowania dla algorytmu $DTW$
$\lambda$	Parametr określający szybkość dyfuzji
$\lambda_A$	Parametr określający szybkość dyfuzji dla składowej anizotropowej
$\lambda_1^{(D)}, \lambda_2^{(D)}$	Wartości własne tensora $D$
$\Lambda_0$	Zbiór punktów należących do określonej krzywej $f$
$\Lambda_T$	Zbiór parametrów definiujących określoną krzywą $f$
$\nabla^2 f = \Delta f$	Laplasjan funkcji skalarnej $f$
$m$	Parametr określający rozmiar dyskretnej maski
$M_B$	Maska dla filtru typu <i>butterfly</i>
$M_D(i, j)$	Element macierz odległości pomiędzy elementami trajektorii
$MSE$	Błąd średniokwadratowy (ang. <i>Mean Square Error</i> )
$MSE_Q$	Błąd średniokwadratowy (ang. <i>Mean Square Error</i> ) dla kwaternionów
$\mu$	Wartość średnia rozkładu
$\vec{n}_\perp$	Wektor normalny do powierzchni
$\vec{n}$	Wektor reprezentujący oś obrotu
$N(t)$	Estymator poziomu całkowitego zaszumienia obrazu dla chwili czasowej $t$

$N(\mu, \sigma)$	Rozkład normalny o wartości oczekiwanej $\mu$ , odchyleniu standardowym $\sigma$
$NMSE$	Znormalizowany błąd średniokwadratowy
$NMSE_Q$	Znormalizowany błąd średniokwadratowy dla kwaternionów
$\Omega$	Obszar rozpatrywania równania różniczkowego
$\partial\Omega$	Powierzchnia (brzeg) obszaru $\Omega$
$P(x, y, z)$	Punkt w obszarze $\Omega$
${}^A P$	Wektor określający pozycję punktu względem układu współrzędnych $A$
$q$	Kwaternion, trajektoria kwaternionów
$\bar{q}$	Odpowiednik estymatora wartości średniej dla kwaternionów
$q_i$	Element $i$ trajektorii kwaternionów
$q_i^{(j)}$	Element $i$ trajektorii kwaternionów w iteracji $j$ algorytmu
$\bar{q}_n^{MD}$	Średnie odchylenie bezwzględne dla populacji $n$ kwaternionów
$\bar{Q}$	Wektor strumienia ciepła
$r$	Parametr odległości w transformacji Hougha dla linii prostych
${}^A_B R$	Macierz obrotu układu $B$ względem układu odniesienia $A$
$R_x(\alpha)$	Macierz obrotu wokół osi $x$ o kąt $\alpha$
$RT$	Transformacja Radona
$\rho$	Dyskretna funkcja odległości
$s$	Część rzeczywista (skalarna) kwaternionu
$s_\alpha$	Liczba kroków kwantyzacji parametru $\alpha$ przestrzeni Hougha
$s_r$	Liczba kroków kwantyzacji parametru $r$ przestrzeni Hougha
$S^3$	Hipersfera jednostkowa w przestrzeni czterowymiarowej
$S(p, q)$	Miara bliskości dwóch osobników dla algorytmu genetycznego
$S^k(p_i)$	Operator przesunięcia czasowego o $k$ działający na elemencie $p_i$
$SO(3)$	Specjalna grupa ortogonalna
$SNR$	Stosunek sygnału do szumu (ang. <i>Signal to Noise Ratio</i> )
$QSNR$	Kwaternionowy stosunek sygnału do szumu
$\sigma$	Odchylenie standardowe rozkładu
$\sigma_n$	Odchylenie standardowe sygnału zakłócającego o rozkładzie normalnym
$\sigma_I^2$	Wariancja jasności pikseli na obrazie $I$
$\sigma_{qn}^2$	Odpowiednik wariancji dla $n$ kwaternionów
$t$	Chwila czasowa (przypadek ciągły), numer iteracji (przypadek dyskretny)
$t_{\max}$	Maksymalna liczba iteracji
$tr(R)$	Ślad macierzy $R$
$T$	Graniczna chwila czasowa w rozpatrywanym okresie $[0, T]$
$\hat{T}(P, t)$	Funkcja pola temperatury w punkcie $P \in \Omega$ w chwili czasowej $t$
$T_H$	Okres wyznaczania transformaty Hougha i rekonstrukcji obrazu
${}^A_B T$	Wektor translacji układu $B$ względem układu $A$
$TS^3$	Przestrzeń styczna do hipersfery jednostkowej $S^3$
$\tau$	Zmienna parametryzująca trajektorię w przestrzeni (elementów trajektorii)
$\mathcal{G}$	Współczynnik skalujący dla funkcji kontrastu $K(t)$
$u(x)$	Rozwiązanie równania różniczkowego
$u_0(x)$	Warunek początkowy

$u_i$	Element $i$ -ty dyskretnej jednowymiarowej funkcji $u$
$u_{i,j}$	Element o współrzędnych $(i, j)$ dyskretnej dwuwymiarowej funkcji $u$
$u_s^t$	Wartość elementu o współrzędnych $s = (i, j)$ w chwili czasowej $t$
$u_\sigma$	Funkcja $u$ poddana regularyzacji przez splot z jądrem Gaussa $K_\sigma$
$\bar{u}$	Rozszerzenie funkcji $u$ na dziedzinę nieskończoną + inne z falką
$\vec{v}$	Część urojona (wektorowa) kwaternionu
$w_i$	Waga elementu $i$ -tego
$w(i, j)$	Element maski o indeksie $(i, j)$ dyskretnej maski pełniący rolę wagi
$W(I)$	Szerokość obrazu $I$ w pikselach
$\omega(q_i)$	Prędkość kątowna wyznaczona dla elementu $i$ trajektorii $q$
$\omega_A(q_i)$	Oś obrotu elementu $i$ trajektorii $q$ wyznaczona na podstawie prędkości kątowej
$\hat{x}$	Wersor wektora $\vec{x}$
$\hat{x}_A, \hat{y}_A, \hat{z}_A$	Wersory osi głównych układu współrzędnych $A$
$\psi_H$	Wartość progowa przy konstrukcji odwrotnej transformaty Hougha