

- 1.1.7. — metody stochastyczne
- 1.1.9. — modelowanie i symulacja
- 4.2. — robotyka

PRACA
DOKTORSKA

Adam Kuriański

DETEKCJA I ŚLEDZENIE RUCHU
PRZY UŻYCIU
CZASOWO-PRZESTRZENNEGO
MODELOWANIA OBRAZÓW
ZA POMOCĄ PÓL LOSOWYCH

37/1994

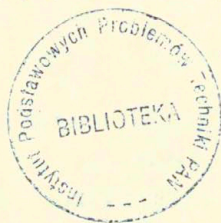
P. 269



WARSZAWA 1994

ISSN 0208-5658

Praca wpłynęła do Redakcji dnia 7 grudnia 1994 r.



56612



Praca doktorska

Recenzent — Prof. dr hab. inż. Wojciech Zamojski
Prof. dr hab. Witold Kosiński

Promotor — Doc. dr hab. Mariusz Nieniewski

Instytut Podstawowych Problemów Techniki PAN
Nakład 100 egz. Ark. wyd. 10,0 Ark. druk. 6,25
Oddano do drukarni w grudniu 1994 r.

Wydawnictwo Spółdzielcze sp. z o.o.
Warszawa, ul. Jasna 1

<http://rcin.org.pl>

Adam Kuriański
Samodzielna Pracownia
Optyczno-Komputerowych
Metod Mechaniki

DETEKCJA I ŚLEDZENIE RUCHU PRZY UŻYCIU CZASOWO-PRZESTRZENNEGO MODELOWANIA OBRAZÓW. ZA POMOCĄ PÓL LOSOWYCH

Podziękowania

Autor chciałby serdecznie podziękować promotorowi prof. dr hab. Mariuszowi Nienewskiemu za opiekę naukową w latach 1990–1994, tzn. w czasie stacjonarnych studiów doktoranckich odbytych przez autora w Instytucie Podstawowych Problemów Techniki Polskiej Akademii Nauk w Warszawie, między innymi za wprowadzenie świeżo upieczonego magistra inżyniera elektronika w świat nauki, za pomoc przy pisaniu pierwszych opracowań naukowych, za jego czas poświęcony w okresie bezpośredniej pracy nad rozprawą doktorską (lata 1992–1994), jak również za okazywaną wyrozumiałość, niebywałą cierpliwość oraz za wiele inspirujących uwag zgłaszanych w czasie długich i licznych konsultacji.

Podziękowania należą się również pozostałym członkom Samodzielnej Pracowni Optyczno-Komputerowych Metod Mechaniki IPPT PAN za wiedzę, którą starali się podzielić w trakcie pracownianych seminariów, co miało wpływ na rozumienie przez autora wielu problemów współczesnej nauki, jak również za życzliwość okazywaną najmłodszemu członkowi zespołu.

Spis treści

1	O rozprawie	9
1.1	Tematyka, zakres, teza oraz cele rozprawy	10
1.2	Streszczenie rozprawy	13
2	Wprowadzenie	15
2.1	Komputerowa analiza ruchu	16
2.1.1	Przedstawienie problemu	16
2.1.2	Metody niskiego poziomu	18
2.1.3	Metody wysokiego poziomu	27
2.2	Pola Markowa w komputerowej analizie obrazów	31
2.2.1	Pola Markowa a procesy Markowa	31
2.2.2	Pola Markowa – rys historyczny	33
2.2.3	Wprowadzenie do teorii pól Markowa	35
2.2.4	Nieprzyczynowe pola Markowa	38
2.2.5	Hierarchiczne pola Markowa	52
2.2.6	Ukryte pola Markowa	53
3	Ogólna metoda budowy modeli statystycznych opierających się na polach Markowa	57
3.1	Metoda budowy modeli opartych na polach Markowa	58
3.2	Zastosowania modeli opartych na polach Markowa	60
4	Detekcja zmian w obrazach – badania porównawcze	63
4.1	Wybór metod detekcji zmian w obrazie	64
4.2	Testowe sekwencje obrazów	65
4.3	Wyniki przeprowadzonych eksperymentów	65
4.4	Wnioski	78
5	Zastosowanie pól Markowa w detekcji i śledzeniu ruchu	81
5.1	Pola Markowa jako modele matematyczne wykorzystywane do detekcji masek obiektów ruchomych	82
5.2	Model Lalande'a-Bouthemy'ego (LB)	85
5.3	Nowy model z dwuwymiarowym wektorem obserwacji	89
5.4	Algorytmy estymacji mody pola Markowa-Gibbsa przeznaczonego do detekcji masek obiektów ruchomych	92
5.4.1	Relaksacja deterministyczna	92

5.4.2	Modyfikacja przyspieszająca algorytm relaksacji deterministycznej	93
5.4.3	Symulowane wyżarzanie	94
6	Kryterium oceny dokładności masek obiektów ruchomych	97
6.1	Ocena dokładności masek obiektów ruchomych	98
6.2	Testowe sekwencje obrazów	98
6.3	Estymacja parametrów modelu LB oraz nowego modelu z dwuwymiarowym wektorem obserwacji dla sekwencji testowych	99
7	Badania eksperymentalne modelu LB	105
7.1	Rola poszczególnych składników energii modelu LB	106
7.2	Badania eksperymentalne – sekwencja K	106
7.3	Badania eksperymentalne – sekwencje L,M,N	117
8	Badania eksperymentalne nowego modelu z dwuwymiarowym wektorem obserwacji	123
8.1	Badania eksperymentalne – sekwencje K,L,M,N	124
8.2	Porównanie nowego modelu z dwuwymiarowym wektorem obserwacji z modelem LB	135
9	Przykładowe wykorzystanie pól Markowa w detekcji i śledzeniu ruchu	137
9.1	Detekcja i śledzenie ruchu w przypadku sekwencji rzeczywistych	138
9.2	Filtracja masek	138
9.3	Dobór parametrów pola Markowa	142
9.4	Wyniki eksperymentów	144
9.5	Sumacyjna metoda detekcji maski obiektu ruchomego	148
10	Ukryte pole Markowa z trójwymiarowym wektorem obserwacji, przeznaczone do detekcji masek obiektów ruchomych	153
10.1	Analiza ruchu obiektów o lokalnie stałej funkcji jasności	154
10.2	Ukryte pole Markowa z trójwymiarowym wektorem obserwacji	159
10.3	Przykłady zastosowania pola Markowa z trójwymiarowym wektorem obserwacji do detekcji masek obiektów ruchomych	161
11	Podsumowanie	165
11.1	Wnioski	166
11.2	Kierunki dalszych badań	168
	Bibliografia	169
	Dodatek	176
A	Moduł różnicy jako źródło obserwacji dla nowego modelu z dwuwymiarowym wektorem obserwacji oraz modelu LB	177
A.1	Badania modelu LB	178
A.2	Badania nowego modelu z dwuwymiarowym wektorem obserwacji	178

B	Opis pakietu programów do detekcji ruchu	187
B.1	Koncepcja pakietu	188
B.2	Pliki tekstowe używane do wymiany informacji pomiędzy programami	188
B.3	Opis zasadniczych programów wchodzących w skład pakietu	191
B.3.1	Program CHANGE	191
B.3.2	Program MEDIAN	192
B.3.3	Program BINFILTR	192
B.3.4	Program ICM	192
B.3.5	Program ICM2	193
B.3.6	Program ICM4	193
B.4	Opis pomocniczych programów wchodzących w skład pakietu	193
B.4.1	Program GREY2MSK	193
B.4.2	Program SHOW_CON	193
B.4.3	Program GAUSS2D	193

Rozdział 1

O rozprawie

1.1 Tematyka, zakres, teza oraz cele rozprawy

Prezentowana rozprawa zajmuje się zagadnieniami detekcji i śledzenia ruchu za pomocą komputerowego systemu wizyjnego, przy wykorzystaniu modeli statystycznych zwanych polami Markowa. Z formalnego punktu widzenia powyższa tematyka może być zaliczona do *komputerowej analizy obrazów*, zwanej również czasami *wizją komputerową*.

Szybki postęp technologiczny w dziedzinie sprzętu komputerowego jaki się dokonał w ciągu ostatnich lat, z czym wiąże się znaczne zwiększenie możliwości przetwarzania informacji, spowodował, że czas potrzebny do prowadzenia komputerowej analizy ruchu przestaje być barierą nie do pokonania. Liczba potencjalnych zastosowań jest bardzo duża, od nadzorowania ruchu ulicznego [DA92], poprzez analizę ruchu mikroorganizmów [Kor93], aż do konstrukcji pojazdów bądź obiektów poruszających się bez nadzoru człowieka [Pom93, Tsu93]. Z każdym z wymienionych zastosowań wiąże się ogromna liczba szczegółowych problemów, które muszą być rozwiązane. To właśnie jest przyczyną rozległości szeroko rozumianej *komputerowej analizy ruchu*. Detekcja i śledzenie są tylko jednymi z wielu szczegółowych zagadnień.

Metody wizji komputerowej, zgodnie z propozycją przedstawioną w [Pav87], można podzielić na dwie zasadnicze grupy, tzn. na *przetwarzanie* i *rozpoznawanie* obrazów. Wyniki otrzymywane poprzez użycie metod z pierwszej grupy stanowią informacje wejściową dla algorytmów z grupy drugiej. Większość praktycznych zastosowań wykorzystuje metody z obydwu grup. *Rozpoznawanie obrazów*, tzn. zamiana informacji jaką jest obraz na jego opis, jest zazwyczaj zasadniczym zadaniem systemu wizyjnego. Znalaziona w ten sposób informacja może stanowić podstawę np. do wypracowania odpowiednich decyzji przez system sterujący. Liczba różnych znanych metod rozpoznawania obrazów jest bardzo duża, por. [Pra91, Ś191, TF91, Hor89]. Przetwarzanie obrazów służy często dokonaniu redukcji informacji obrazowej, aby ułatwić przeprowadzenie rozpoznawania. W dużej liczbie przypadków efektem przetwarzania są obrazy binarne.

Zakres niniejszej rozprawy ograniczono do zagadnień, które można zaliczyć do komputerowego przetwarzania obrazów. Zadanie *detekcji ruchu* rozumiane jest jako poszukiwanie dla obrazu z gradacją szarości odpowiedniego obrazu binarnego, w którym odróżnione są piksele poruszającego się obiektu od pozostałych, nazywanych również pikselami tła. Tak postawione zagadnienie można również nazwać *poszukiwaniem maski* poruszającego obiektu, gdzie maską nazywamy zbiór pikseli obiektu ruchomego. *Śledzenie ruchu* jest utożsamiane ze znalezieniem sekwencji masek obiektu ruchomego odpowiadającej sekwencji obrazów wejściowych. Przy takim sformułowaniu zagadnienia śledzenie jest bardzo ściśle związane z zagadnieniem detekcji ruchu i polega na przeprowadzeniu tej detekcji dla całej wejściowej sekwencji obrazów z gradacją szarości. Z tego powodu obydwaj zagadnienia w całej rozprawie są traktowane w sposób łączny.

Jako podstawę matematyczną dla tworzenia odpowiedniej metody wybrano statystyczne modele zwane polami Markowa. Zostało to zainspirowane chęcią potwierdzenia użyteczności pól Markowa do przeprowadzenia detekcji i śledzenia ruchu. Wybór tych modeli został spowodowany również przez stale rosnące zainteresowanie jakim cieszą się one wśród twórców algorytmów wizji komputerowej, ze względu na ich następujące cechy:

- statystyczny charakter, dzięki czemu tworzone na ich bazie metody są bardziej od-

porne na szum, gdyż uwzględniają jego istnienie już na etapie modelowania [Pra91, TQD86],

- właściwość lokalnego opisu złożonych problemów z uwzględnieniem wpływu pikseli sąsiednich.

Z jednej strony przegląd literatury światowej informuje o dość dużej liczbie nowych metod opartych na polach Markowa, z drugiej strony zastosowanie tych modeli do detekcji ruchu nie występuje zbyt często. W opisywanych aplikacjach najczęściej stosuje się metody potoku optycznego, natomiast wykrywanie masek obiektu ruchomego występuje rzadziej. Z tego powodu sformułowano następującą tezę rozprawy:

W komputerowej analizie ruchu obiektów znajdujących się w polu widzenia kamery, przedstawionych za pomocą sekwencji obrazów z gradacją szarości, możliwe jest u efektywne detekcji i śledzenia ruchu przeprowadzanych przy użyciu czasowo-przestrzennego modelowania obrazów, poprzez zmianę wyrażenia energii pola Markowa.

Aby wykazać słuszność powyższej tezy sformułowano następujące cele rozprawy:

1. Opracowanie nowego modelu statystycznego, opartego na czasowo-przestrzennych ukrytych polach Markowa z dwuwymiarowym wektorem obserwacji, przeznaczonych do detekcji masek obiektów ruchomych, uwzględniającego wszystkie możliwości etykietowania piksela, co nie miało miejsca w modelu dotychczasowym.
2. Opracowanie modelu statystycznego, opartego na czasowo-przestrzennych ukrytych polach Markowa z trójwymiarowym wektorem obserwacji, przeznaczonych do detekcji masek obiektów ruchomych, jako udoskonalenie spotykanych dotychczas modeli z obserwacją dwuwymiarową.
3. Opracowanie inżynierskiej metody budowy modeli statystycznych opartych na polach Markowa i przeznaczonych do zastosowań w algorytmach komputerowej analizy obrazów.
4. Opracowanie kryterium oceny dokładności masek obiektów ruchomych, znajdowanych za pomocą modeli statystycznych.
5. Ocena użyteczności modelu Lalande'a-Bouthemy'ego do znajdowania masek obiektów ruchomych, a w szczególności:
 - (a) określenie kryteriów doboru wartości parametrów tego modelu.
 - (b) określenie wrażliwości tego modelu na wartości jego parametrów.
 - (c) przebadanie dokładności metody.

6. Porównanie efektów otrzymywanych za pomocą nowego modelu z obserwacją dwuwymiarową z tymi, które można otrzymać za pomocą modelu Lalande'a-Boutliemy'ego, a w szczególności przeprowadzenie badań pozwalających na dokonanie oceny użyteczności nowego modelu według kryteriów jak w punkcie 5.
7. Opracowanie oraz przebadanie kompleksowej metody detekcji i śledzenia ruchu obiektów przedstawionych za pomocą sekwencji obrazów z gradacją szarości, a w szczególności określenie sposobu doboru wartości parametrów używanego modelu.
8. Opracowanie metody detekcji masek obiektów ruchomych za pomocą sumowania masek znajdujących przy różnych wartościach parametrów pola Markowa odpowiadających kilku typowym przypadkom.
9. Skrócenie czasu obliczeń potrzebnego do znalezienia sekwencji masek obiektów ruchomych poprzez opracowanie modyfikacji przyspieszającej działanie algorytmu relaksacji deterministycznej używanego do poszukiwania najbardziej prawdopodobnej realizacji pola Markowa.
10. Dokonanie wyboru optymalnego detektora zmian czasowych zachodzących między kolejnymi obrazami sekwencji z punktu widzenia przydatności do otrzymywania masek obiektów ruchomych.
11. Stworzenie pakietu oprogramowania, umożliwiającego przeprowadzanie detekcji i śledzenia ruchu metodami korzystającymi z czasowo-przestrzennych pól Markowa na komputerach o architekturze szeregowej.

1.2 Streszczenie rozprawy

Rozprawa przedstawia metody detekcji i śledzenia ruchu niewymagające wyznaczenia potoku optycznego, a korzystające z czasowo-przestrzennych pól Markowa, jak również przykładowe wyniki, które można otrzymać dzięki wykorzystaniu tych metod dla typowych sekwencji obrazów. Praca zawiera jedenaście rozdziałów, bibliografię oraz dodatek. W rozdziale pierwszym (niniejszym) przedstawiono w skrócie całą rozprawę. We wprowadzającym w tematykę rozdziale drugim opisano obecny stan wiedzy obejmującej zarówno komputerową analizę ruchu jak i pola Markowa. Omówienia zagadnień komputerowej analizy ruchu dokonano opierając się na przeprowadzonych studiach literaturo- wych. Zaprezentowana teoria pól Markowa dotyczy przede wszystkim ich wykorzystania w komputerowej analizie obrazów. W rozdziale trzecim przedstawiono metodę budowy modeli statystycznych korzystających z pól Markowa, będącą oryginalnym opracowaniem autora. Metoda powyższa jest też podsumowaniem aktualnego stanu wiedzy dotyczącego omawianych modeli. W rozdziale czwartym zamieszczono wyniki przeprowadzonych przez autora badań różnych znanych z literatury metod detekcji zmian czasowych zachodzących pomiędzy kolejnymi obrazami sekwencji. W rozdziale piątym zajęto się zagadnieniem budowy modeli statystycznych przeznaczonych do detekcji i śledzenia ruchu. W szczególności przedstawiono znany z literatury model Lalande'a-Bouthemy'iego, jak również zaproponowano model ulepszony, w którym podobnie jak w modelu Lalande'a-Bouthemy'iego wykorzystywany jest dwuwymiarowy wektor obserwacji. W rozdziale tym zamieszczono również oryginalną propozycję modyfikacji algorytmu relaksacji deterministycznej estymacji mody pola Markowa, tzn. poszukiwania najbardziej prawdopodobnej realizacji tego pola. W rozdziale szóstym przedstawiono kryterium oceny masek obiektów ruchomych znajdujących przy użyciu modeli statystycznych. W rozdziale siódmym zamieszczono wyniki badań eksperymentalnych metody detekcji masek obiektów ruchomych opartej na modelu znanym z literatury. Zawartość rozdziału ósmego jest analogiczna do zawartości rozdziału siódmego, z tym, że zaprezentowane wyniki eksperymentów dotyczą nowego modelu z dwuwymiarowym wektorem obserwacji. W rozdziale dziewiątym przedstawiono przykładowe wyniki detekcji i śledzenia ruchu dla typowych sekwencji obrazów rzeczywistych. Ponadto przedyskutowano zagadnienie detekcji ruchu dla tych sekwencji, ze szczególnym uwzględnieniem problemu estymacji parametrów stosowanego modelu oraz zagadnienia filtracji masek zmian czasowych i masek obiektów ruchomych. Zastosowanie metod filtracji jest niezbędne w przypadku niektórych sekwencji rzeczywistych. W rozdziale tym omówiono również sumacyjną metodę detekcji masek obiektów ruchomych, będącą oryginalnym opracowaniem autora. W rozdziale dziesiątym wprowadzono nowy model statystyczny z trójwymiarowym wektorem obserwacji, korzystający z dwuwymiarowego rozkładu normalnego jako sposobu uwzględnienia wpływu obserwacji na rozkład ukrytego pola losowego. Zaproponowany model charakteryzuje się właściwością uzupełniania braków informacji wykorzystywanej maski zmian czasowych. W rozdziale jedenastym przeprowadzono podsumowanie całej rozprawy oraz zasugerowano kierunki dalszych ba-

dań. W Dodatku A zamieszczono wyniki badań modelu Lalande'a-Bouthemy'iego oraz nowego modelu z dwuwymiarowym wektorem obserwacji analogiczne do wyników z rozdziałów siódmego i ósmego, z tym, że jako obserwację przyjęto moduł różnicy jasności przyporządkowanych pojedynczemu pikselowi w kolejnych obrazach, a nie różnicę tych jasności. W Dodatku B opisano pakiet programów komputerowych opracowanych w celu przeprowadzenia badań dla potrzeb niniejszej rozprawy.

Rozdział 2

Wprowadzenie

Celem niniejszego rozdziału jest przedstawienie aktualnego stanu wiedzy w dwóch dziedzinach, których dotyczą zasadnicze części rozprawy, tzn. komputerowej analizie ruchu, oraz modelowaniu obrazów za pomocą pól Markowa, wybranych jako przykład pól losowych. Teoria pól Markowa jest częścią statystyki matematycznej. W zaprezentowanym przeglądzie zagadnień dotyczących tych pól zwrócono uwagę szczególnie na ich wykorzystanie w komputerowej analizie obrazów. Doboru przedstawionych zagadnień analizy ruchu dokonano na podstawie przeprowadzonych studiów literaturowych.

2.1 Komputerowa analiza ruchu

2.1.1 Przedstawienie problemu

W obecnym czasie jedną z możliwości prowadzenia analizy ruchu jest wykorzystanie do tego celu *komputerowego systemu wizyjnego*. System taki składa się zazwyczaj z części optycznej, której zadaniem jest dostarczanie informacji wizyjnej o otaczającej rzeczywistości, oraz części komputerowej przetwarzającej zebraną przez system optyczny informację. Obydwie części systemu imitują w pewnym sensie układ oko-mózg człowieka. Podobnie jak w przypadku istoty ludzkiej, przetwarzanie przez system komputerowy informacji dostarczonej przez „imitację oka” może służyć bądź tylko rejestrowaniu obserwowanej rzeczywistości, bądź wypracowaniu reakcji na zachodzące wydarzenia. Przykładem może być zastosowanie komputerowego systemu wizyjnego do sterowania robota samojezdnego. Informacja jaką dostarcza system komputerowy jest wykorzystywana do wypracowania sygnałów sterujących umożliwiających odpowiednią nawigację robota w jego otoczeniu. W takim przypadku następuje zamknięcie pętli sprzężenia zwrotnego przez komputerowy system wizyjny.

W przypadku analizy ruchu bardzo ważną rolę odgrywa wzajemna relacja pomiędzy obserwatorem, czyli kamerą, a analizowanym zjawiskiem, którym najczęściej jest zachowanie się pewnego obiektu. W zasadzie istnieją trzy możliwości:

- statyczna kamera, ruchomy obiekt,
- ruchoma kamera, statyczny obiekt,
- ruchoma kamera, ruchomy obiekt.

Najtrudniejszym do analizy wydaje się przypadek trzeci, w którym dużo zależy od przyjętych założeń wstępnych.

Systemy wizyjne można podzielić na:

- systemy jednokamerowe,
- systemy wielokamerowe.

W zastosowaniach opartych na jednej kamerze system optyczny dostarcza sekwencji cyfrowych obrazów z gradacją szarości. Mogą to być np. zdigitalizowane ramki sygnału telewizyjnego, które następnie poddawane są analizie za pomocą systemu komputerowego. Wraz ze wzrostem liczby kamer ulega zwiększeniu ilość informacji o otoczeniu, dzięki czemu możliwe jest precyzyjniejsze wypracowanie sygnałów sterujących. Niestety, dla wielu systemów komputerowych zastosowanie więcej niż jednej kamery prowadzi do takiego wzrostu ilości informacji wejściowej, że niemożliwe staje się jej przetworzenie w zadowalającym czasie. W konsekwencji systemy wielokamerowe są znacznie droższe od systemów jednokamerowych, ze względu zarówno na cenę sprzętu optycznego jak i konieczności korzystania z systemów komputerowych o odpowiednio większej mocy obliczeniowej.

Zadania stawiane przed systemami komputerowymi mogą mieć bardzo różny charakter. W najprostszym przypadku może być to tylko udzielenie odpowiedzi na pytanie, czy

w obserwowanej scenie zaszły zmiany spowodowane np. ruchem jakiegoś obiektu. Niekiedy samo stwierdzenie faktu zajścia zmian jest niewystarczające. Koniecznym staje się zlokalizowanie ruchomego obiektu.

Bardziej złożonymi zadaniami stawianymi przed systemami komputerowymi mogą być: interpretacja ruchu, czyli odróżnienie obszarów o różnym typie ruchu, jak translacja lub obrót, rozpoznawanie obiektów ruchomych, estymacja ruchu, tzn. znajdowanie wartości i kierunku prędkości, śledzenie poruszającego się obiektu, czyli określenie kolejnych jego położeń.

Dla potrzeb niniejszej rozprawy metody komputerowej analizy ruchu zostaną podzielone na algorytmy niskiego i wysokiego poziomu. Do grupy metod niskiego poziomu zaliczane są te, które pozwalają na rozwiązanie zadań podstawowych, tzn. takie, dla których danymi wejściowymi jest analizowana sekwencja z gradacją szarości, a znalezione rezultaty są podstawą do wykorzystania w innych algorytmach. Na zasadzie kontrastu algorytmami wysokiego poziomu określa się te, które korzystają z wyników znalezionych przez algorytmy niskiego poziomu. W komputerowej analizie ruchu do metod niskiego poziomu można zaliczyć algorytmy:

1. detekcji zmian w obrazie [AKM93, IT93, NP91, NP90, SJ89, HNR84, Nag82],
2. oparte na cechach obrazów, tzn.
 - (a) detekcji narożników ruchomych [Kur92, RC92, SJ84, KR82],
 - (b) detekcji krawędzi ruchomych [Bou89, HJ83],
3. znajdowania potoku optycznego, tzn. metody oparte na czasowo-przestrzennym gradiencie [KD90, Hor89, VP92],
4. transformacji obrazów [Mah91a, Mah91b, Jai82].

Korzystając z wymienionych technik opracowano metody wysokiego poziomu, jak np.

1. detekcję ruchu opartą na:
 - ruchomych krawędziach [DA92],
 - maskach zmian w obrazie [BL90, LB90],
 - potoku optycznym [Bla90, MB87],
2. śledzenie ruchu korzystające z:
 - ruchomych narożników [SS90, SJ87],
 - ruchomych krawędzi [DA92],
 - potoku optycznego [AT90].
3. estymację ruchu wykorzystującą:
 - potok optyczny [HB90a, HB90b],
 - metody transformacji obrazów [Mah91a, Mah91b],

- ruchome narożniki [AN88, Nag83, DN82],
- ruchome krawędzie [WHA92],

4. interpretację ruchu opartą na:

- potoku optycznym [FB90a, FB90b],

Nie wszystkie znane w literaturze metody można w sposób oczywisty zaliczyć do którejś z wymienionych grup. Przykładem może być rozwiązanie zaproponowane w [HB90a], gdzie przedstawiono kompleksowe podejście do analizy ruchu. W [MC93] przedstawiono metodę, którą bezdyskusyjnie należy zaliczyć do grupy algorytmów wysokiego poziomu, gdyż dotyczy rozpoznawania poruszających się obiektów, a nie korzysta ona z żadnej z wymienionych metod niskiego poziomu. Autorzy zastosowali segmentację opartą jedynie na funkcji jasności, a wykorzystującą własności obrazu cienia poruszającego się obiektu.

W rozdziałach 2.1.2 oraz 2.1.3 przedstawiono przegląd metod komputerowej analizy ruchu.

2.1.2 Metody niskiego poziomu

DETEKCJA ZMIAN W OBRAZIE

Detekcja zmian w obrazie jest jednym z najbardziej podstawowych zadań komputerowej analizy ruchu. W najprostszych zastosowaniach może ona jedynie polegać na udzieleniu odpowiedzi na pytanie, czy w obserwowanej scenie zaszły istotne zmiany, np. spowodowane ruchem jakiegoś dużego obiektu, a nie tylko szumem elektronicznym [IT93]. W większości zastosowań wykrywanie zmian dla sekwencji obrazów z gradacją szarości rozumiane jest jednak inaczej. Każde dwa sąsiednie obrazy porównywane są ze sobą. Wynikiem porównania jest, poza stwierdzeniem faktu zajścia zmian, również ich lokalizacja. W języku komputerowej analizy obrazów oznacza to, że:

1. detekcję zmian przeprowadza się dla każdych dwóch kolejnych obrazów sekwencji wejściowej,
2. w wyniku otrzymywany jest obraz binarny, tzw. *maska zmian*, w którym wyróżnione są piksele należące do obszarów, w których zaszły zmiany.

Z przedstawionego opisu wynika, że jeżeli analizowana sekwencja składa się z N obrazów, to efektem poszukiwania zmian będzie sekwencja $N - 1$ obrazów binarnych. Poza pierwszym i ostatnim obrazem analizowanej sekwencji każdy pozostały bierze udział w tworzeniu dwóch masek zmian.

Poniżej przedstawiono skrócony przegląd metod detekcji zmian znanych z literatury.

Detekcja zmian na podstawie analizy histogramów sekwencji obrazów. Zaproponowane w [IT93] rozwiązanie ma jedynie na celu stwierdzenie faktu zajścia zmian. Uproszczony charakter metody powoduje, że może ona znaleźć zastosowanie głównie w

przypadkach typowych dla obserwacji scen statycznych, takich jak np. ochrona skarbcza bankowego.

Dla każdego obrazu sekwencji najpierw znajdujemy histogram, a następnie obliczane są wartości trzech statystyk, tzn. dystrybuanty empirycznej, średniej próby oraz wariancji. W następnym kroku porównywane są ze sobą odpowiednie statystyki dla dwóch sąsiednich obrazów sekwencji. Porównywanie dokonuje się przy użyciu trzech testów statystycznych: Kołmogorowa, χ^2 Pearsona oraz testu średnich. Jeżeli przy założonych wcześniej poziomach istotności w przypadku każdego z testów stwierdzono zajście zmian, to taką samą odpowiedź otrzymuje się dla całej metody.

Różnica jasności piksla w kolejnych chwilach czasowych. Z matematycznego punktu widzenia podstawą stwierdzenia lokalnej zmiany funkcji jasności w pojedynczym pikslu jest wartość pochodnej cząstkowej funkcji jasności względem czasu $\frac{\partial f(x,y,t)}{\partial t}$, przy czym $f(x,y,t)$ oznacza jasność przyporządkowaną pikselowi o współrzędnych (x,y) w chwili czasu t [LB90]. Ponieważ w komputerowej analizie obrazów mamy do czynienia z przypadkiem dyskretnym, zarówno jeśli chodzi o dziedzinę jak i zbiór wartości funkcji, więc obliczaniu pochodnej czasowej odpowiada zwykle odejmowanie np. postaci $f(x,y,k) - f(x,y,k-1)$, przy czym k - indeks przyporządkowany kolejnym obrazom sekwencji. Oznacza to, że odejmowanie jasności piksla z kolejnych chwil czasowych odpowiada różniczkowaniu numerycznemu. Skądinąd wiadomo, że wszystkie metody różniczkowania numerycznego są bardzo wrażliwe na szum. Stąd wynika ograniczona stosowalność różnicy jasności do detekcji zmian. Niewątpliwą jednak zaletą powyższej metody jest jej prostota, a tym samym krótki czas obliczeń.

Testy statystyczne oparte na lokalnym modelowaniu funkcji jasności. W artykule [HNR84] przedstawiono grupę metod detekcji zmian pozbawionych w pewnym sensie wad metody opartej na różnicy jasności. Ze względu na obecność szumu w obrazach przyjęto, że:

- w celu wykrycia zmian korzysta się nie tylko z wartości jasności w pojedynczym pikslu, ale i w jego otoczeniu.
- szum uwzględniony jest już na etapie modelowania.

Po pierwsze założono, że w niewielkim otoczeniu piksla funkcję jasności można aproksymować pewnym wielomianem. W zależności od rzędu wielomianu stworzono trzy metody, a mianowicie o stałym, liniowym i kwadratowym modelu funkcji jasności. Po drugie przyjęto, że niezgodność modelu z rzeczywistymi wartościami funkcji jasności opisana jest gaussowskim szumem o zerowej wartości średniej i stałej wariancji. Modelowanie takie przeprowadzono dla odpowiednich okien dwóch kolejnych obrazów sekwencji. Opierając się na przyjętych założeniach opracowano trzy testy statystyczne, po jednym dla każdego modelu funkcji jasności. Jako podstawę konstrukcji testów przyjęto stosunek wiarygodności [Zie90]. Przedstawione testy pozwalają na udzielenie odpowiedzi, czy niezgodność jasności w kolejnych chwilach czasowych jest spowodowana tylko szumem, czy zmianami treści przedstawianej sceny.

Test statystyczny oparty na transformacie Walsh'a funkcji jasności. Propozycję tej metody przedstawiono w [NP91, NP90]. Podobnie jak w metodzie poprzedniej podstawą jest modelowanie statystyczne pewnego otoczenia piksla oraz test oparty na stosunku wiarygodności dwóch hipotez. Podstawowymi różnicami są przyjęty stały rozmiar otoczenia piksla, tzn. 4×4 , oraz inny model funkcji jasności. Funkcje jasności w otoczeniu piksla rozłożono na szesnaście funkcji Walsh'a. A priori założono, że dziewięć pierwszych funkcji Walsh'a odpowiada treści obrazu, a siedem pozostałych związanych jest z szumem. Na tej podstawie skonstruowano test statystyczny porównując ze sobą odpowiednie okna dwóch kolejnych obrazów sekwencji.

Test oparty na modelu cieni. W [SJ89] przedstawiono rozważania, z których wynika, że wszystkie omówione dotychczas metody są bardzo wrażliwe na zmiany oświetlenia. Nawet niewielka zmiana warunków oświetlenia statycznej sceny powoduje wykrycie zmian, które utrudniają analizę ruchu. W pracy [SJ89] zaprezentowano metodę o zmniejszonej wrażliwości na efekty świetlne. Oparta jest ona na jednym z modeli zacienienia, używanym w grafice komputerowej. Na podstawie dłuższych rozważań stwierdzono, że podstawą konstrukcji testu musi być stosunek jasności przyporządkowanych temu samemu pikselowi w dwóch kolejnych obrazach sekwencji. Przyjmując pewne ustalone otoczenie piksla najpierw oblicza się stosunki jasności w dwóch kolejnych obrazach sekwencji $\frac{f(x,y,k+1)}{f(x,y,k)}$ dla wszystkich pikseli należących do tego otoczenia włącznie z pikselem aktualnym. Następnie liczy się średnią ze znalezionych wcześniej wartości, a potem wariancję. Obliczona wariancja jest miarą zmian. Przy odpowiednio dużej wariancji przyjmuje się, że została ona spowodowana zmianami treści sceny.

KRAWĘDZIE RUCHOME

Krawędzie są jedną z najczęściej poszukiwanych cech obrazów komputerowych [AS92, LHS87, BHM86, Can86, KL86, NB86]. Spróbujmy zdefiniować pojęcie krawędzi.

Dany piksel jest zaliczany do krawędzi, jeśli zmiany funkcji jasności w tym pikselu są odpowiednio duże.

Przedstawiona definicja ma oczywiście charakter nieformalny. Kryterium zaliczania danego piksla do krawędzi jest niejednoznaczne. W zależności od treści obrazu oraz dynamiki funkcji jasności rozumianej jako różnica pomiędzy jasnościami minimalną i maksymalną, piksel może zostać zaliczony do krawędzi lub nie. Najczęściej detekcja krawędzi dokonywana jest za pomocą jakiegoś lokalnego operatora działającego na funkcję jasności. Wartość takiego operatora jest porównywana z wcześniej określonym progiem. Gdy przekroczona zostanie wartość takiego progu, to dany piksel jest zaliczany do krawędzi.

Poza metodami znajdowania krawędzi możliwe jest również poprawianie wcześniej znalezionych obrazów krawędziowych, np. jak to zostało przedstawione przez autora w [Kur93, Kur92].

W detekcji krawędzi ruchomych, poza samą detekcją krawędzi, konieczne jest odseparowanie krawędzi należących do statycznego tła od tych, które należą do poruszającego się obiektu. Jak w przypadku większości metod wizji komputerowej, w literaturze znane

są różne algorytmy znajdowania ruchomych krawędzi [DA92, Bou89, HJ83]. W niniejszej pracy zostaną omówione dwie metody. Pierwsza została zaprezentowana w [HJ83] i oparta jest ona na maskach zmian w obrazie oraz dowolnym z operatorów umożliwiających znajdowanie krawędzi statycznych. Drugą z metod, korzystającą ze statystycznego czasowo-przestrzennego modelu sekwencji obrazów z gradacją szarości, przedstawiono w [Bou89].

Najprostszy sposób znajdowania krawędzi ruchomych [HJ83] związany jest z przeprowadzeniem dwóch lokalnych operacji na funkcji jasności:

- znajdowanie krawędzi dla danego obrazu sekwencji,
- wykrywanie zmian w przedstawianej scenie zachodzących pomiędzy aktualnym obrazem i obrazami sąsiednimi.

Jeżeli dany piksel należy do krawędzi i jednocześnie należy od obszaru, w którym zaszły zmiany, to jest on pikselem ruchomej krawędzi. Jeżeli obydwie wymienione powyżej operacje dokonywane są za pomocą operatorów działających lokalnie na funkcję jasności lub różnicę jasności dwóch obrazów, to możliwe jest połączenie ich w jeden operator za pomocą zwykłego iloczynu [HJ83]. Przy takim rozwiązaniu w celu wykrycia krawędzi ruchomych należy określić tylko jedną wartość progową. Poza tym wykorzystanie iloczynu dwóch operatorów umożliwiła znajdowanie zarówno tych ruchomych krawędzi, dla których wartość miary określającej ich krawędziowość jest mała ale krawędzie te znajdują się w obszarach silnych zmian, jak również „silnych” krawędzi leżących w obszarach niewielkich lokalnych zmian funkcji jasności. Ze względu na swoją prostotę rozwiązanie proponowane w [HJ83] ma również pewną wadę. Jako kryterium rozstrzygające o zachodzących zmianach używana jest różnica jasności, która, jak wiadomo, nie uwzględnia wpływu otoczenia i z tego powodu jest bardzo wrażliwa na szum obecny w obrazach komputerowych.

Pewną modyfikacją powyższej metody jest rozwiązanie zaproponowane w [DA92], nazywane przez autorów operatorem MOD. Operator ten jest iloczynem dwóch operatorów Sobela, z których jeden zastosowany jest do aktualnego obrazu sekwencji, a drugi do różnicy tego obrazu i obrazu sąsiadującego.

Metoda zaprezentowana w [Bou89] jest znacznie bardziej skomplikowana. Wykorzystuje ona test hipotezy statystycznej opartej na stosunku wiarygodności. Niech π oznacza pewien podzbiór czasoprzestrzeni odpowiadającej sekwencji obrazów z gradacją szarości. Sformułujmy hipotezy:

- H_0 : ruchoma krawędź w π nie występuje. Wtedy przyjmuje się, że jasność wszystkich pikseli z π ma stałą wartość c_0 , do której dodany jest gaussowski szum $N(0, \sigma^2)$ o zerowej wartości średniej i wariancji σ^2 .
- H_1 : ruchoma krawędź w π występuje. W takim przypadku przyjmuje się, że pewna powierzchnia $S(\phi)$, zdefiniowana przez wektor parametrów ϕ , dzieli π na π_1 oraz π_2 , jasności pikseli należących do π_1 mają wartości c_1 , a pikseli z π_2 jasności c_2 ($c_1 \neq c_2$). Podobnie jak dla hipotezy poprzedniej zakłada się, że do jasności dodany jest szum o rozkładzie $N(0, \sigma^2)$.

Przyjęte założenia powodują, że w każdym pikselu należącym do π jasność jest opisywana rozkładem normalnym o wartości średniej c_0 , c_1 lub c_2 i wariancji σ^2 . Ponadto uczyniono

założenie, że rozkłady prawdopodobieństwa w poszczególnych pikslach są od siebie niezależne. Dzięki takiemu założeniu funkcja wiarygodności jest iloczynem gęstości prawdopodobieństwa z poszczególnych piksli. Kolejnym krokiem metody jest zastosowanie standardowego testu statystycznego opartego na stosunku wiarygodności, por. [Zie90]. Oczywiście, konstrukcja takiego testu zależy między innymi od przyjętego modelu powierzchni generowanej przez poruszającą się krawędź w czasoprzestrzeni. W [Bou89] przedstawiono odpowiednie wzory, przy założeniu, że powierzchnia $S(\phi)$ jest płaszczyzną.

NAROŻNIKI RUCHOME

Pojęcie narożnika można zdefiniować następująco:

Narożnikiem jest ten piksel obrazu komputerowego, w którym krawędź wystarczająco zmienia swój kierunek.

Definicja powyższa ma charakter nieformalny, gdyż jest oparta raczej na intuicyjnej interpretacji sygnałów wizyjnych niż na zależnościach matematycznych. Samo pojęcie krawędzi, wykorzystywane w powyższej definicji, ma również podobną naturę. W dodatku sformułowanie mówiące o „wystarczającej zmianie kierunku” oznacza jedynie, że cechę „narożnikowości”, tzn. że dany piksel jest narożnikiem, można właściwie przypisywać prawie dowolnie, o ile piksel ten należy do krawędzi. Dla przykładu rozważmy obraz ciemnego prostokąta na jasnym tle. Narożnikiem będzie ten piksel, w którym krawędź zmienia swój kierunek o $\pm 90^\circ$, czyli zgodnie z arytmetyką kierunków wektorów na płaszczyźnie Z^2 przedstawioną przez autora w [Kur92, Kur93], kierunek krawędzi zmienia się o $(\pm 2) \bmod 8$. Gdyby na obrazie komputerowym przedstawiony był sześciokąt, wówczas cechę narożnikowości należałoby przypisać już przy zmianie kierunku biegu krawędzi o $(\pm 1) \bmod 8$. Jeśli jednak mielibyśmy do czynienia z komputerowym obrazem koła, to żadna zmiana kierunku krawędzi nie powinna implikować tego, że piksel jest narożnikiem. Przedstawiony przykład świadczy o niejednoznaczności wyników otrzymywanych za pomocą różnych metod detekcji narożników, gdyż oczekiwany wynik zależy prawie zawsze od treści przedstawianej sceny. W przypadku scen rzeczywistych sytuacja bywa o wiele bardziej skomplikowana niż ta naszkicowana powyżej. Z tego powodu opieranie się jedynie na kryterium lokalnej zmiany kierunku jest najczęściej niewystarczające.

Metody detekcji narożników mają zazwyczaj postać operatorów progowych [Kur92]. W zależności od konkretnych zastosowań konieczne są korekty progu dokonywane przez użytkownika, aby efekt otrzymywany chociaż w przybliżeniu przypominał ten oczekiwany intuicyjnie.

W literaturze znanych jest kilka metod wykrywania narożników [SJ84, KR82], takich jak metody Zuniga-Haralicka, Kitchen-Rosenfelda czy Dreschler-Nagela. W [SJ84] przedstawiono analizę porównawczą znanych rozwiązań, w efekcie której stwierdzono, że najlepsze własności ma metoda Zuniga-Haralicka (ZH). Z tego powodu na jej przykładzie zostanie omówione zagadnienie wykrywania narożników w obrazach.

Metoda ZH pozwala na stwierdzenie w trzech krokach, czy dany piksel obrazu komputerowego jest narożnikiem. Oparta jest na modelu ściankowym [HW81]. W celu uwzględnienia sąsiedztwa, funkcję jasności w pewnym otoczeniu aktualnego piksela aproksymuje

się wielomianem trzeciego stopnia dwóch zmiennych o postaci:

$$g(x, y) = k_1 + k_2x + k_3y + k_4x^2 + k_5xy + k_6y^2 + k_7x^3 + k_8x^2y + k_9xy^2 + k_{10}y^3, \quad (2.1)$$

przy czym: (x, y) – współrzędne piksła, $k_1 \dots k_{10}$ – współczynniki wielomianu aproksymującego. Kroki metody ZH dla jednego piksła wyglądają następująco:

1. *Aproksymacja.*

Znajdź wartości współczynników $k_1 \dots k_{10}$, np. za pomocą metody najmniejszych kwadratów.

2. *Sprawdzanie czy aktualny piksel należy do krawędzi.*

Oblicz wartości pochodnych cząstkowych g_x oraz g_y tworzących gradient (g_x, g_y) aproksymacji $g(x, y)$. Oblicz wartości pierwszej i drugiej pochodnej aproksymacji $g(x, y)$ w kierunku gradientu. Jeśli wartość pierwszej pochodnej jest większa od wcześniej ustalonego progu, a wartość drugiej pochodnej równa się w przybliżeniu zero, to przejdź do następnego kroku, w przeciwnym przypadku zacznij sprawdzać następny piksel.

3. *Sprawdzanie czy aktualny piksel jest narożnikiem.*

Oblicz wartość pochodnej kąta gradientu Θ w kierunku ortogonalnym do gradientu, tzn. znajdź wartość wyrażenia:

$$\frac{-2(k_2^2k_6 - k_2k_3k_5 + k_3^2k_4)}{(k_2^2 + k_3^2)^{1.5}} \quad (2.2)$$

Jeśli obliczona wartość jest większa od wcześniej założonego progu, to przyjmij, że dany piksel jest narożnikiem.

Oryginalna metoda ZH charakteryzuje się pewną wadą. Ze względu na dużą liczbę obliczeń zmiennooprzecinkowych dla każdego piksła, jest ona czasochłonna. W [Kur92] autor przedstawił metodę poprawioną, dzięki której uzyskano dwudziestokrotne skrócenie czasu potrzebnego na przeprowadzenie obliczeń dla wszystkich piksli należących do obrazu. Główna idea zaproponowanej w [Kur92] modyfikacji polega na wydzieleniu etapu detekcji krawędzi i zastąpieniu go metodą szybszą. Aby jednak krawędzie znalezione dzięki zastosowaniu takiej metody, jak np. operator Sobela, były porównywalnej lub nawet lepszej jakości niż te znalezione przy użyciu modelu ściankowego, można jeszcze przed wykrywaniem narożników zastosować algorytmy poprawy obrazów krawędziowych przedstawione w [Kur92, Kur93].

Wykrycie narożników w obrazie jest tylko pierwszym etapem w analizie ruchu. Ze zbioru wszystkich narożników trzeba pozostawić tylko te, które należą do poruszających się obiektów. Dla każdego narożnika jest sprawdzane, czy leży on w obszarze zmian. Do tego celu wykorzystywana jest operacja iloczynu logicznego. Jeśli dany piksel jest narożnikiem oraz jeśli należy on do obszaru zmian, to jest on narożnikiem ruchomym. W [SJ84] zaproponowano zamiast koniunkcji logicznej, wykorzystanie zwykłego iloczynu wyrażenia (2.2) i różnicy funkcji jasności w aktualnym piksle dla sąsiednich obrazów. Takie sformułowanie problemu detekcji narożników ruchomych powoduje, że możliwe jest wykrycie

zarówno „słabych” narożników, które leżą w obszarze „silnych” zmian, jak i „silnych” narożników, gdy obserwowane zmiany są niewielkie.

Zagadnienie detekcji zmian zachodzących pomiędzy kolejnymi obrazami zostało już omówione poprzednio w tym rozdziale.

POTOK OPTYCZNY

W [Hor89] zamieszczono nieformalną definicję potoku optycznego. Spróbujmy ją jednak przedstawić w języku matematyki.

Potokiem optycznym nazywamy pole wektorowe przyporządkowujące każdemu pikselowi obrazu należącemu do sekwencji obrazów z gradacją szarości wektor odpowiadający zmianom funkcji jasności w tym pikselu zachodzącym pomiędzy sąsiednimi obrazami analizowanej sekwencji.

Do analizy ruchu potrzebne jest jednak pole prędkości. Definicja tego pola jest prawie taka sama jak definicja potoku optycznego z tą różnicą, że każdemu pikselowi przyporządkowuje się wektor prędkości. Gdy system wizyjny składa się z jednej kamery zarówno wektory potoku optycznego jak i wektory pola prędkości mają tylko dwie współrzędne, tzn. potok optyczny i pole prędkości przedstawiają zachodzące zmiany w rzucie na płaszczyznę obrazu. W ogólnym przypadku potok optyczny oraz pole prędkości różnią się między sobą [Hor89, VP92]. Typowym przykładem może być sekwencja obrazów przedstawiająca statyczny obiekt ale w zmiennych warunkach oświetlenia. W takim przypadku pole prędkości jest polem zerowym, czego nie można powiedzieć o potoku optycznym. W komputerowej analizie ruchu opartej na systemach wizyjnych możliwe jest jedynie znajdowanie potoku optycznego. Z tego powodu prawie zawsze zakłada się, że są spełnione odpowiednie warunki, jak np. stałość oświetlenia tak, aby potok optyczny był dobrym przybliżeniem pola prędkości.

Przejdźmy teraz do zależności matematycznych opisujących potok optyczny. Niech $f(x, y, t)$ oznacza wartość funkcji jasności piksela o współrzędnych (x, y) w obrazie z chwili t . Dokonując rozkładu $f(x, y, t)$ w szereg Taylora otrzymujemy:

$$f(x + dx, y + dy, t + dt) = f(x, y, t) + f_x(x, y, t) dx + f_y(x, y, t) dy + f_t(x, y, t) dt + O^2 \quad (2.3)$$

przy czym: f_x oznacza $\frac{\partial f}{\partial x}$, $z = x, y, t$, O^2 – reszta wyrazów rozwinięcia w szereg Taylora. Oznaczając przez $\vec{\omega}_s(u_s, v_s)$ wektor prędkości, przy czym $u_s = \frac{dx}{dt}(s)$, $v_s = \frac{dy}{dt}(s)$, $s = (x, y, t)$ oraz zakładając, że:

- wyrazy wyższego rzędu rozwinięcia (2.3) są pomijalne ($O^2 \rightarrow 0$), tzn. przyjmując, że czasowo-przestrzenna funkcja jasności (ciąg funkcji jasności kolejnych obrazów sekwencji) może być lokalnie aproksymowana liniową funkcją czasu i przestrzeni, oraz
- jasność odpowiadająca dowolnemu punktowi z przestrzeni nie zależy od czasu (stałość warunków oświetlenia),

równanie (2.3) sprowadzamy do postaci:

$$f_x(s) \cdot u_s + f_y(s) \cdot v_s = -f_t(s). \quad (2.4)$$

Równanie (2.4) można również zapisać w postaci:

$$\nabla \vec{f}(s) \cdot \vec{\omega}_s + f_t(s) = 0, \quad (2.5)$$

przy czym: $\nabla \vec{f} = (f_x, f_y)^T$ oznacza gradient przestrzenny. Równanie (2.5) znane jest pod nazwą *równania potoku optycznego*. Dalsze przekształcanie tego równania prowadzi do postaci, z której wynika bardzo interesujący wniosek. Okazuje się, że na podstawie lokalnych obliczeń możliwe jest określenie wartości tylko jednej składowej wektora prędkości $\vec{\omega}_s$, a mianowicie składowej równoległej do gradientu przestrzennego [HB90a, Hor89]. Obserwacja powyższa znana jest w literaturze jako *problem apertury* [Hor89]. W celu znalezienia obu współrzędnych wektora prędkości dodatkowo zakłada się, że punkty położone blisko siebie poruszają się ze zbliżonymi prędkościami. W praktyce metody oparte na optymalizacji lokalnej zakładają stałą wartość prędkości w sąsiedztwie [HB90a], w przeciwieństwie do metod korzystających z optymalizacji globalnej, które oparte są na założeniu gładkich zmian prędkości [HB90a, HB90b].

Założenia będące podstawą do wyprowadzenia równania (2.5) niestety prowadzą do pewnych niekorzystnych konsekwencji. Po pierwsze, stałość jasności odpowiadającej jakiemś punktowi z przestrzeni w całej sekwencji obrazów w rzeczywistości jest bardzo trudna do osiągnięcia. Po drugie, założenie jedynie liniowej postaci rozwinięcia w szereg Taylora traci sens w miejscach gwałtownych zmian funkcji jasności, co w konsekwencji powoduje, że np. w pobliżu krawędzi znajdowanie potoku jedynie na podstawie równania (2.5) nie ma żadnego sensu [Hor89].

Z postaci równania (2.5) można również zauważyć, że gdy tylko ta część funkcji jasności, która reprezentuje poruszający się obiekt, ma stałą wartość w pewnym obszarze ($\nabla \vec{f}(s) = 0$), to znalezienie współrzędnych wektora potoku optycznego jest niemożliwe, gdyż w takim przypadku może być to dowolny wektor.

Zadanie poszukiwania potoku optycznego na pierwszy rzut oka nie wydaje się być bardzo skomplikowane. Badania przeprowadzone przez niektórych badaczy jednak tego nie potwierdzają, por. [Bla90]. Znane algorytmy znajdowania pola optycznego są o wiele bardziej skomplikowane niż tylko zwyczajne zastosowanie równania (2.5), por. [Jar94]. Innym przykładem może być podejście do problemu przedstawione w pracach [HB90a, HB90b]. Zaprezentowaną tam metodę nazwano *wielomodalną estymacją nieciągłego potoku optycznego*. W postaci pewnego pola Markowa opisano zależności pomiędzy potokiem optycznym otrzymanym dzięki zastosowaniu równania (2.5) i ruchomymi krawędziami znalezionymi za pomocą metody zaprezentowanej w [Bou89], a następnie znajdowano potok optyczny, na podstawie obydwu informacji cząstkowych.

ANALIZA RUCHU OPARTA NA TECHNIKACH TRANSFORMACJI OBRAZÓW

Techniki transformacji obrazów wykorzystywane są w metodach wizji komputerowej już od wielu lat. Do najpopularniejszych należy transformata Fouriera (FT) [Ś191, Pav87], do obliczania której znane są również szybkie wersje algorytmów, tzw. FFT. Oprócz FT istnieje

wiele innych transformat, por. [Ś191]. Niektóre z metod transformacji sygnałów dwuwymiarowych znalazły również zastosowanie w komputerowej analizie ruchu, między innymi transformata Hartley'a (HT) [Mah91a, Mah91b] oraz transformacja EMP [Jai82], czyli *transformacja funkcji jasności do współrzędnych biegunowych względem ogniska ekspansji*. Poniżej w skrócie omówiono przykłady metod wykorzystujących wspomniane techniki.

Obliczanie wartości prędkości dużych obiektów przy użyciu transformaty Hartley'a. Transformata Hartley'a dokonuje przekształcenia dziedziny sekwencji obrazów z gradacją szarości w dziedzinę częstotliwości przestrzenno-czasowych [MAG88, Mah91a, Mah91b]. W pracy [Mah91b] wykazano, że maksimum transformaty Hartley'a sekwencji obrazów z gradacją szarości związane jest z prędkością obiektu ruchomego. Korzystając jedynie z transformaty Hartley'a niemożliwym jest jednak znalezienie kierunku wektora prędkości. Z tego powodu w [Mah91b] zaproponowano metodę estymacji prędkości dużych obiektów opartą jednocześnie na transformatach Hartley'a i Fouriera. Zaproponowany algorytm wygląda następująco:

1. Znajdź wartość transformaty Hartley'a dla analizowanej sekwencji;
 $H[k, f] = ?$
2. Wybierz przestrzenną częstotliwość k_s .
3. Znajdź maksimum $H[k_s, f]$ oraz odpowiadającą jemu częstotliwość przestrzenną f_p . Prędkość poruszającego się obiektu wynosi $V_h = f_p/k_s$.
4. Znajdź widmo Fouriera z $H[k, f]$ dla częstotliwości przestrzennej k_s , tzn. oblicz $F[k_s, f]$.
5. Znajdź maksimum $F[k_s, f]$ oraz odpowiadającą jej częstotliwość \hat{f}_p . Prędkość poruszającego się obiektu wynosi $V_f = \hat{f}_p/k_s$, ze znakiem przeciwnym niż \hat{f}_p .
6. Porównaj V_h oraz V_f . Jeśli są różne to zwiększ k_s i skocz do 3, w przeciwnym przypadku koniec.

W [Mah91b] napisano, że przedstawiona metoda jest szybsza niż inne metody analizy ruchu. Niestety nie zamieszczono przykładów zastosowań dla sekwencji rzeczywistych, a jedynie przedstawiono wyniki dla sztucznie wygenerowanej sekwencji sygnałów jednowymiarowych.

Transformacja EMP. Transformacja EMP sekwencji obrazów z gradacją szarości zaprezentowana w [Jai82] jest transformacją obrazów z prostokątnego układu kartezjańskiego OXY do układu współrzędnych biegunowych, przy czym transformacji nie dokonuje się względem środka układu OXY a względem punktu zwanego *ogniskiem ekspansji* (ang. Focus of Expansion).

Ogniskiem ekspansji (FOE) nazywa się punkt wspólny kierunku ruchu obserwatora określonego w przestrzeni R^3 z płaszczyzną rzutu (obrazu).

Transformacja EMP oparta jest na następującym równaniu:

$$e_k(r, \theta) = f_k(x, y) \quad (2.6)$$

przy czym: $f_k(x, y)$ – jasność przyporządkowana pikselowi (x, y) w k -tym obrazie sekwencji, $e_k(r, \theta)$ – jasność k -tego obrazu po transformacji w pikselu o współrzędnych biegunowych (r, θ) .

Najpierw należy znaleźć wartości r oraz θ dla każdego piksela (x, y) analizowanej sekwencji. Następnie należy przeanalizować zmiany r oraz θ w pikselu (x, y) w całej sekwencji. Jak zostało wykazane w [Jai82], jeżeli kamera porusza się ruchem jednostajnym, to θ w pikselu (x, y) należącym do stacjonarnego tła ma stałą wartość w całej sekwencji. Każda zmiana kąta θ w pikselu (x, y) oznacza, że piksel ten należy do obiektu ruchomego. Na tej podstawie możliwe jest rozróżnianie pikseli, w których zmiany jasności spowodowane są ruchem obiektów a nie kamery. Dzięki takiej własności metoda wydaje się być niezastąpioną w analizie ruchu obiektów za pomocą ruchomej kamery.

2.1.3 Metody wysokiego poziomu

Liczba różnych algorytmów wysokiego poziomu zaprezentowanych w literaturze jest bardzo duża. Większość z nich można zakwalifikować do jednej z następujących grup:

- detekcja ruchu,
- śledzenie ruchu,
- estymacja ruchu,
- interpretacja ruchu,
- rozpoznawanie poruszających się obiektów.

Ze względu na tematykę niniejszej rozprawy poniżej zostaną omówione dwa zagadnienia, a mianowicie detekcja oraz śledzenie ruchu.

DETEKCJA RUCHU

Zadanie detekcji ruchu polega na odróżnieniu poruszającego się obiektu od statycznego tła. Można tego dokonać

- korzystając z cech obrazów komputerowych takich jak krawędzie czy narożniki ruchome, bądź
- opierając się na potoku optycznym, albo
- jako informację wejściową potraktować wynik detekcji zmian w obrazie.

Detekcja ruchu wykorzystująca cechy obrazów komputerowych połączona jest najczęściej z zadaniem estymacji ruchu. Znając np. położenie narożników możliwe jest obliczenie prędkości poruszającego się obiektu. O wiele bardziej złożonym jest rozpoznanie poruszającego się obiektu. Detekcja ruchu korzystająca z cech obrazów umożliwia także w pewnym stopniu oddzielenie piksli obiektu od tła, poprzez wykorzystanie segmentacji konturowej. Szczególnie do tego celu nadają się ruchome krawędzie.

Detekcja ruchu poprzez segmentację obszarową może być dokonana zarówno przy użyciu potoku optycznego jak i masek zmian w obrazie. W literaturze znacznie częściej prezentowane jest podejście opierające się na potoku optycznym. Idea tego podejścia polega na wydzieleniu obszarów, w których wielkość wektora zmian potoku optycznego ma dostatecznie dużą wartość. W praktyce metody detekcji ruchu nie mogą być oparte jedynie na tak prostym założeniu, chociażby dlatego, że wartość potoku optycznego jest nieznaną, gdy poruszający się obiekt reprezentowany jest za pomocą stałej jasności.

Konkurencyjnym podejściem do detekcji ruchu jest potraktowanie jako informacji wejściowej maski zmian w obrazie znalezionej przy użyciu jednej z wielu znanych metod. Ponieważ w niniejszej rozprawie wybrano właśnie to podejście do detekcji ruchu, dlatego zostanie ono omówione bardziej szczegółowo.

Detekcja ruchu opierająca się na maskach zmian w obrazie. Na rys. 2.1 schematycznie przedstawiono detekcję ruchu korzystającą z masek zmian w obrazie.

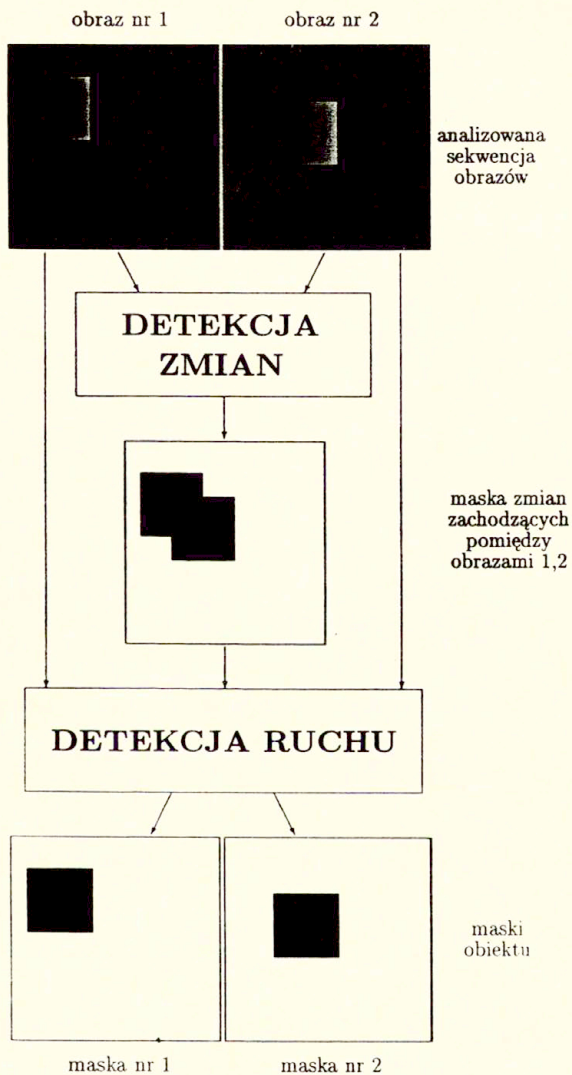
Pierwszym etapem rozważanego podejścia do detekcji ruchu jest oczywiście detekcja zmian zachodzących pomiędzy kolejnymi obrazami sekwencji. Jak wiadomo pojedynczą maskę zmian w obrazie znajduje się poprzez porównanie ze sobą dwóch kolejnych obrazów sekwencji. Założymy na chwilę, że dostępna jest pełna wiedza o dwóch analizowanych obrazach sekwencji, tzn. wiadomo, które piksele należą do statycznego tła, a które do obiektu ruchomego. Jeżeli piksel (i, j) we wcześniejszym obrazie należy do tła, a w późniejszym obrazie do poruszającego się obiektu, to mówimy, że w pikselu tym mamy do czynienia z sytuacją (tło, ruch). W sposób analogiczny można zdefiniować trzy pozostałe sytuacje: (tło, tło), (ruch, tło) oraz (ruch, ruch).

W idealnej masce zmian powinny być odróżnione piksele, w których występują sytuacje (ruch, ruch), (ruch, tło), (tło, ruch), od piksli, w których występuje sytuacja (tło, tło). W przypadku rzeczywistych sekwencji brak jest zazwyczaj pełnej wiedzy o wartości funkcji jasności obiektu. Znalezione maski zmian ma postać obrazu binarnego. Oznacza to, że gdy w jakimś pikselu wykryto zmiany, to nie wiadomo, która z sytuacji (ruch, ruch), (ruch, tło) czy (tło, ruch) je spowodowała.

Skądinąd wiadomo, że podczas detekcji zmian w obrazach najtrudniej wykrywalna jest sytuacja (ruch, ruch), gdyż najczęściej kłopotów sprawiają te części obiektu, które mają stałą, bądź zbliżoną do stałej funkcję jasności. Praktycznie żadnych trudności nie sprawiają *sytuacje przejściowe*, tzn. (tło, ruch) oraz (ruch, tło), chyba że jasność obiektu na jego brzegach nie różni się od tła. Jest to jednak przypadek zdegenerowany, gdyż w takiej sytuacji stwierdzenie ruchu byłoby niemożliwe nawet dla oka ludzkiego.

Zadanie detekcji ruchu, na podstawie maski zmian w obrazach, dla sekwencji złożonej z dwóch obrazów, można sformułować następująco:

Mając dostępne obydwa wejściowe obrazy z gradacją szarości oraz odpowiadającą im maskę zmian (obraz binarny), znaleźć dwa ob-



Rys. 2.1: Detekcja masek obiektu ruchomego opierająca się na masce zmian w obrazie.

razy binarne, tzw. *maski obiektu ruchomego*, odpowiadające obrazom wejściowym, w których odróżniony byłby poruszający się obiekt od statycznego tła.

W takim przypadku detekcja ruchu polega na zaetykietowaniu wszystkich pikseli sekwencji obrazów z gradacją szarości np. przy pomocy etykiet ze zbioru $\{a, b\}$, przy czym a symbolizuje przynależność piksla do obiektu ruchomego, b oznacza, że dany piksel należy do statycznego tła.

Gdyby detekcja ruchu została przeprowadzona bezbłędnie, to w przypadku maski obiektu ruchomego z chwili wcześniejszej, etykiety a powinny być przypisane wszystkim tym pikselom, w których zmiany funkcji jasności zostały spowodowane sytuacjami (ruch, tło), (ruch, ruch). W przypadku maski obiektu ruchomego z chwili późniejszej etykieta a musiałaby być przyporządkowana pikselom, w których wykryte zmiany zostały spowodowane sytuacjami (ruch, ruch) oraz (tło, ruch).

W tym miejscu należy wyraźnie zaznaczyć, że gdy w dalszej części rozprawy będzie mowa np. o sytuacji (ruch, ruch), to dotyczy to tylko rozważań teoretycznych, natomiast gdy wymieniona będzie realizacja (a, a) , to oznacza, że dany piksel w kolejnych dwóch obrazach sekwencji został zaliczony do poruszającego się obiektu.

W przypadku rzeczywistych sekwencji znalezienie dwóch masek obiektów, na podstawie tylko dwóch obrazów z gradacją szarości oraz odpowiadającej im jednej masce zmian w obrazie, jest niemożliwe. Wyjątek stanowią sekwencje, w których występuje wyraźna różnica pomiędzy jasnością tła i obiektu oraz dostępna jest pełna wiedza o wartościach tych jasności. W [Nie93] przedstawiono metodę znajdowania masek obiektu przy użyciu trzech obrazów, tj. na podstawie dwóch masek zmian. Przedstawiona metoda opiera się jednak na zbyt dużych uproszczeniach. Warunkiem wykrycia masek obiektu jest niewystępowanie sytuacji (ruch, ruch).

W [BL90, LB90] przedstawiono metodę dokonującą detekcji ruchu na podstawie masek zmian dla sekwencji dłuższych niż dwa obrazy. Powyższa metoda stała się punktem wyjścia dla opracowania nowych algorytmów, które zaprezentowano w dalszej części rozprawy.

ŚLEDZENIE RUCHU

Zadanie śledzenia ruchu polega na „podążaniu” za poruszającym się obiektem. Podążanie takie może być rozumiane w sensie dosłownym, tzn. jako zmiana położenia kamery, tak aby „obserwowany” obiekt znajdował się zawsze w środku sceny [AT90], lub jako przedstawienie zmian położenia obiektu w poszczególnych chwilach czasu, gdy kamera jest stacjonarna. Pierwszy sposób „obserwacji” obiektów można by nazwać *śledzeniem aktywnym* w przeciwieństwie do *śledzenia pasywnego*, gdy pozycja kamery nie ulega zmianie. Przejście z podejścia pasywnego do aktywnego wymaga dodatkowo określenia sposobu zmiany położenia kamery na podstawie wykrytego przemieszczenia obiektu.

W dalszej części pracy pod nazwą śledzenie będzie zawsze rozumiane śledzenie pasywne. Z formalnego punktu widzenia

zadanie śledzenia ruchu obiektu przy użyciu komputerowego systemu wizyjnego

polega na określeniu położenia tego obiektu w każdej chwili czasowej, czyli znalezieniu trajektorii jego ruchu.

W literaturze znanych jest kilka podejść do tak postawionego problemu. Często jako punkt wyjścia przyjmuje się ruchome narożniki [SJ87, SS90, DN82]. Niekiedy taki sposób „obserwacji” obiektu określa się mianem ustalania odpowiedności pomiędzy narożnikami z kolejnych obrazów sekwencji. Ze względu na możliwości pojawiania się i znikania pojedynczych narożników w kolejnych obrazach, śledzenie ruchu przy użyciu narożników ruchomych może być bardzo skomplikowane. W [SJ87] podano kryterium nazwane *spójnością ścieżki*, które zostało następnie zapisane w postaci funkcji zgodności. Określenie odpowiedności pomiędzy narożnikami z różnych obrazów polega na takim wyborze relacji pomiędzy narożnikami, aby zapewnić jednocześnie minimum funkcji zgodności.

Bardzo interesujące podejście do śledzenia ruchu przy użyciu narożników przedstawiono w [DN82]. Aby ustalić odpowiedność narożników dodatkowo założono, że poruszający się obiekt można aproksymować przy pomocy pewnej bryły przestrzennej, której wierzchołkami są znalezione narożniki.

W przypadku, gdy analiza ruchu polega na poszukiwaniu maski obiektu, zadanie śledzenia można rozumieć jako przedstawienie zmian położenia obiektu w kolejnych chwilach czasowych, tzn. na przedstawieniu sekwencji znalezionych masek obiektu ruchomego. Tak też jest ono rozumiane w dalszej części rozprawy.

2.2 Pola Markowa w komputerowej analizie obrazów

2.2.1 Pola Markowa a procesy Markowa

Pola Markowa są uogólnieniem modeli statystycznych, zwanych procesami Markowa, na sygnały określone w przestrzeniach wielowymiarowych. W algorytmach komputerowych opartych na statystyce matematycznej, ze względu na sposób reprezentacji przetwarzanej informacji, zazwyczaj wykorzystuje się modele określone na dyskretnej dziedzinie. Często przyjmowane jest też dodatkowe założenie o dyskretności zbioru możliwych stanów modelu losowego. Wymienione przyczyny składają się na fakt, że dla potrzeb komputerowej analizy obrazów pola Markowa są jedynie uogólnieniem na przestrzenie wielowymiarowe *łańcuchów Markowa* [por87, Bob86, Pac85]. Łańcuchy Markowa, ze względu na swój charakter, są wykorzystywane do modelowania sygnałów dyskretnych zależnych od jednej zmiennej dyskretnej. W przypadku pól losowych liczba dyskretnych zmiennych niezależnych jest większa niż jeden. W rozważaniach przedstawionych w niniejszym rozdziale ograniczono się do sygnałów o dyskretnym zbiorze wartości i o dwu- lub trójwymiarowej dziedzinie dyskretnej. Właśnie taki charakter ma informacja, jaką niosą ze sobą pojedyncze obrazy komputerowe lub ich sekwencje. Chociaż przedstawiony aparat matematyczny można bez problemu przenieść na sygnały określone na dziedzinach o dowolnym wymiarze, to jednak w praktyce, ze względu na rozmiar otrzymywanego zadania, jego wykorzystanie jest bardzo utrudnione.

W przypadku łańcucha Markowa istnieje dyskretna jednowymiarowa zmienna niezależna, przebiegającej pewien uporządkowany zbiór (np. podzbiór liczb naturalnych).



Rys. 2.2: Przykładowy ciąg Markowa.

Zmienna ta, zwyczajowo zwana czasem, determinuje fakt, że kolejne próbki modelowanego sygnału pojawiają się w sposób uporządkowany. Dzięki istnieniu takiej zmiennej, w teorii łańcuchów losowych pojawiają się takie pojęcia jak *przeszłość*, *teraźniejszość* czy *przyszłość*, por. rys. 2.2. Należy tu jednak podkreślić, że jednowymiarowość łańcucha losowego jest związana tylko ze zmienną niezależną i niekoniecznie musi dotyczyć informacji niesionej przez sam sygnał. Do modelowania tej informacji można użyć na przykład wielowymiarowej zmiennej losowej, ale kolejne próbki z takiego rozkładu muszą pojawiać się w pewnej sekwencji. Sposób uporządkowania łańcucha losowego określony jest poprzez wzajemne zależności sąsiednich próbek modelowanego sygnału. Zależności te określane są podczas identyfikacji modelu zwanego łańcuchem Markowa.

W przypadku pól losowych wielowymiarowość nie tylko może dotyczyć informacji niesionej przez sygnał, ale bezwzględnie wiąże się z dziedziną tego sygnału.

Jednym z podstawowych założeń w teorii łańcuchów Markowa jest to, że zależność informacji niesionej przez sygnał w danej chwili od próbek informacji z przeszłości jest ograniczona do pewnej liczby tych próbek. Liczba próbek sygnału z przeszłości, które mają wpływ na informację niesioną przez sygnał w chwili bieżącej nazywana jest *rzędem łańcucha*. Z pojęciem rzędu związane są pojęcia sąsiada i sąsiedztwa. *Sąsiadem* nazywamy tę próbkę sygnału z przeszłości, która ma wpływ na rozkład prawdopodobieństwa informacji w chwili aktualnej, natomiast *sąsiedztwem* zbiór wszystkich takich próbek.

Związki pomiędzy sąsiednimi próbkami informacji łańcucha Markowa określa się za pomocą warunkowych rozkładów prawdopodobieństwa informacji niesionej przez sygnał w chwili aktualnej pod warunkiem zaistnienia próbek sygnału z przeszłości. Oczywiście komplikacja modelu rośnie wraz ze wzrostem przyjętego rzędu łańcucha.

Znajomość próbek sygnału z poprzednich chwil daje możliwość predykcji informacji w chwili bieżącej. Wymieniona cecha decyduje często o użyteczności modeli statystycznych zwanych łańcuchami Markowa.

Pola Markowa, ze względu na wielowymiarowość przestrzeni, na której są określone, są pozbawione własności uporządkowania kolejnych próbek informacji. Wyjątek stanowią, znane z literatury, przyczynowe pola Markowa (ang. *Markov Mesh Models*) [Kan80, Lac87, JW91]. W przypadku nieprzyczynowych pól Markowa mamy do czynienia z pojęciem sąsiedztwa, natomiast nie istnieje pojęcie przeszłości, a to oznacza niemożność

przeprowadzenia predykcji informacji bieżącej. Istniejące algorytmy, na przykład znajdowania wartości modalnej (mody) pola Markowa, tzn. najbardziej prawdopodobnej realizacji rozkładu pola Markowa, często napotykają barierę czasu potrzebnego na znalezienie optymalnego rozwiązania.

Teoria łańcuchów Markowa jest dobrze znana. Pojawiło się wiele prac bezpośrednio zajmujących się tym problemem, por. [Fel68]. Istnieją również opracowania omawiające konkretne zastosowania tych modeli, np. w automatycznym rozpoznawaniu mowy [Wrz93]. Teoria pól Markowa rozwija się ostatnio dość szybko, jednak w zależności od konkretnych zastosowań przyjmowane są różne założenia upraszczające. W kolejnych podrozdziałach zajmiemy się polami Markowa w aspekcie ich zastosowań do rozwiązywania problemów wizji komputerowej.

2.2.2 Pola Markowa – rys historyczny

W literaturze ostatnich dwudziestu lat pojawiło się wiele prac poruszających problematykę wykorzystania pól Markowa w komputerowej analizie obrazów. Wydaje się jednak, że największe zasługi w rozwoju i zaadoptowaniu tej części statystyki matematycznej dla potrzeb wizji komputerowej ma Besag (dzięki publikacji pracy [Bes74]). W jego artykule zostały zamieszczone zasadnicze twierdzenia teorii pól Markowa. W szczególności, przytoczono twierdzenie Hammersley'a-Cliforda (HC), które przedstawiono również z nowym dowodem wykorzystującym mniej skomplikowany aparat matematyczny. Omówiono także wnioski wynikające z tego twierdzenia. Poza tym można tam znaleźć ciekawe przykłady wykorzystania omawianej teorii. W [Bes74] zostały zaproponowane również rodziny warunkowych rozkładów prawdopodobieństwa lokalnie opisujące zależności pola Markowa. Wspomniane rozkłady losowe zostały nazwane przez autora autorozkładami. W autorozkładach uwzględnione są ograniczenia wynikające z twierdzenia HC. Zastosowanie pojęcia kodinów (ang. codings), jako metody estymacji parametrów pola Markowa, miało również duże znaczenie, ze względu na późniejsze wykorzystanie tej metody przez innych autorów [CJ83]. Besag w kolejnych swoich pracach zaproponował inny sposób znajdowania parametrów pola Markowa, oparty na wprowadzonym przez siebie pojęciu pseudowiarygodności [Bes86].

Inną ciekawą pracą z tego zakresu jest artykuł Gemanów z roku 1984 [GG84]. W pracy tej przedstawiono twierdzenie o równoważności modeli pól Markowa z rozkładem Gibbsa, znanym od dawna w fizyce statystycznej. Co prawda, rozkład Gibbsa wprowadzili do analizy obrazów Hassner i Sklansky już w roku 1980 [HS80], ale wspomniane twierdzenie zostało po raz pierwszy przedstawione w sposób pełny dopiero w pracy [GG84].

Dzięki udowodnionej równoważności rozkładu pola Markowa z rozkładem Gibbsa, znaleziono sposób estymowania mody takiego rozkładu, zwany *symulowanym wyżarzaniem*. Pojęcie to wprowadzono do literatury w pracy [KGV83]. Oparte jest ono na analogii ze zjawiskiem wyżarzania, które w skrócie można przedstawić następująco: zaczynając od bardzo wysokiej temperatury układu wyżarzanego, a następnie obniżając ją stopniowo, można doprowadzić układ do stanu o minimalnej energii. W rozkładzie Gibbsa minimum energii odpowiada jego modzie. Na tej podstawie został zaproponowany algorytm stochastycznej relaksacji z wolnozmiennym parametrem temperatury. Algorytm ten znany jest pod nazwą *impulsatora Gibbsa* [GG84].

W pracy [GG84] zamieszczono odpowiednie twierdzenia i ich dowody, podając również funkcję zmiany temperatury, która zapewnia optymalność otrzymanego rozwiązania, niestety w czasie $t \rightarrow \infty$. W ten sposób został poruszony jeden z podstawowych problemów wiążących się z wykorzystaniem modeli pół Markowa, a mianowicie czas potrzebny do znalezienia rozwiązania. Każdy z autorów uporał się z tym zagadnieniem w zależności od swoich potrzeb, proponując jednak rozwiązania suboptymalne, por. [Cha88]. W dobie szybko rozwijającego się sprzętu komputerowego można mieć jednak nadzieję, że to co obecnie wymaga zbyt dużej mocy obliczeniowej w przyszłości będzie osiągalne nawet dla zwykłych komputerów osobistych, być może nawet o architekturze szeregowej.

Praca [GG84] miała również aspekt praktyczny. Przedstawiono tam przykład wykorzystania pół Markowa do restauracji obrazów, wraz ze zdjęciami pokazującymi efekt działania algorytmu symulowanego wyżarzania.

Inni autorzy pracowali bądź nad rozbudową aparatu matematycznego, bądź nad zastosowaniami praktycznymi. W pierwszym przypadku zajmowano się:

- algorytmami estymacji parametrów pół Markowa [Bes86, Vei91],
- metodami estymowania lokalnych maksimów pola Markowa [Bes86],
- dowodami zgodności metod maksymalnej entropii z metodami korzystającymi z rozkładu Gibbsa [Tru80],
- uogólnieniem metod wprowadzonych przez Gemanów na przypadek rozkładów ciągłych [JW91],
- estymowaniem mody (wartości modalnej) rozkładu Gibbsa metodami programowania dynamicznego [DE87],
- porównywaniem rezultatów otrzymywanych dzięki zastosowaniu symulowanego wyżarzania oraz deterministycznej relaksacji do estymacji mody pola Markowa [Jin90, KD90].

W zastosowaniach praktycznych zaproponowano i przebadano między innymi metody:

- estymacji i restauracji obrazów [Cha88, Lac87, QT91, EG92, EGK92],
- znajdowania lub poprawiania potoku optycznego [HB90a, HB90b, Bla90],
- znajdowania krawędzi [ZC90],
- detekcji masek obiektów ruchomych [BL90, LB90],
- segmentacji sceny [MB87],
- segmentacji obrazów głębi [GG91, JN90].

2.2.3 Wprowadzenie do teorii pól Markowa

PODSTAWOWE ZAŁOŻENIA I DEFINICJE

W dalszej części rozprawy przyjęto następującą konwencję oznaczeń:

- X_i - jednowymiarowa zmienna losowa z pozycji i w obrazie,
- \mathbf{X} - wielowymiarowa zmienna losowa modelująca cały obraz,
- x_i - realizacja zmiennej losowej z pozycji i w obrazie,
- \mathbf{x} - realizacja zmiennej losowej \mathbf{X} .

Ponieważ pole losowe jest uogólnieniem łańcucha losowego, więc na zasadzie analogii można zdefiniować następujące pojęcia:

Definicja 1 *Polem losowym określonym na obszarze $S \subset R^n$, przy czym R oznacza zbiór liczb rzeczywistych, nazywamy funkcję, która każdemu punktowi tego obszaru przyporządkowuje zmienną losową.*

Obszar S nazywamy *dziedziną pola losowego*. Zmienną losową przyporządkowaną dowolnemu punktowi należącemu do obszaru S nazywamy *wartością pola losowego*. Zbiór możliwych wartości przyjmowanych przez zmienne losowe przyporządkowane punktom z obszaru S nazywamy *zbiorem stanów pola losowego*.

W komputerowej analizie obrazów przyjmuje się dodatkowo następujące założenia:

- dziedzinę pola losowego stanowi skończony podzbiór przestrzeni Z^n , przy czym Z oznacza zbiór liczb całkowitych. Najczęściej zapisuje się go jako iloczyn kartezjański n czynników $[0, \dots, N-1] \times \dots \times [0, \dots, N-1]$, przy czym $[0, \dots, N-1]$ jest zbiorem kolejnych N liczb całkowitych od 0 do $N-1$. Dzięki takiemu założeniu, w komputerowych zastosowaniach rozważanej teorii, dziedzina pola losowego może być reprezentowana jako n -wymiarowa tablica. W praktyce $n = 2$ dla pojedynczego obrazu lub $n = 3$ dla sekwencji obrazów. Dla ustalenia uwagi, w dalszej części tego rozdziału, skoncentrujemy się na przypadku $n = 2$.
- punktami z definicji 1 są położenia najmniejszych elementów obrazu komputerowego. W niniejszym rozdziale, ze względu na przyporządkowywanie poszczególnym elementom dziedziny obrazu rozkładów losowych, pod nazwą *piksel* będzie rozumiane samo położenie najmniejszego elementu, określone przez współrzędne (i, j) , bez żadnej informacji temu położeniu przyporządkowanej,
- poszczególnym pikselom przyporządkowuje się jednowymiarowe dyskretne zmienne losowe. Niekiedy można się spotkać z odchodzeniem od założenia o dyskretności. Ma to miejsce, gdy modelowana informacja jest reprezentowana przez odpowiednio dużą liczbę dyskretnych wartości, na przykład 2^8 . W takiej sytuacji obliczenia są prowadzone tak samo jak dla ciągłych rozkładów prawdopodobieństwa, a dopiero w końcowym etapie dokonuje się dyskretyzacji otrzymanych wyników [Bes86].

Z definicji 1 oraz przyjętych założeń wynika, że w przypadku komputerowej analizy obrazów, pole losowe jest pewnym skończonym zbiorem jednowymiarowych zmiennych losowych. Pole takie może być również opisane wielowymiarowym rozkładem, którego rozkładami brzegowymi są rozkłady w poszczególnych pikselach obrazu. Gdyby te rozkłady

były między sobą niezależne, to łączny rozkład prawdopodobieństwa stałby się zwykłym iloczynem rozkładów brzegowych [por87]. Właśnie na takim założeniu jest oparta duża liczba metod komputerowej analizy obrazów korzystających z modeli statystycznych innych niż pola Markowa, np. por. [NP90, BHM86].

Oznaczmy przez $S = [0, \dots, N-1] \times [0, \dots, N-1]$ dziedzinę obrazu (dziedzinę pola losowego). Załóżmy, że wszystkie piksele (k, l) należące do S ustawimy w pewien wybrany sposób w ciąg, taki że:

$$\{X_1, \dots, X_i, \dots, X_M\} \text{ jest polem losowym, przy czym: } i \in S, M = N^2.$$

Definicja 2 *Przestrzeń obrazów określonych na dziedzinie S nazywamy skończony zbiór Ω możliwych odwzorowań $S \rightarrow L^M$, przy czym L -skończony zbiór wartości.*

Zbiór L jest zbiorem stanów pola losowego. Zgodnie z definicją 2 zbiór Ω nie musi być zbiorem wszystkich możliwych odwzorowań $S \rightarrow L^M$. W praktyce Ω określane jest w ten sposób, aby było zbiorem wszystkich realizacji \mathbf{x} , M -wymiarowej zmiennej losowej \mathbf{X} modelującej cały obraz, określonej na pewnej przestrzeni probabilistycznej o prawdopodobieństwie P .

Definicja 3 *Polem Markowa nazywamy pole losowe, opisane rozkładem wielowymiarowej zmiennej losowej \mathbf{X} określonej na pewnej przestrzeni probabilistycznej o prawdopodobieństwie P i o wartościach należących do pewnej przestrzeni obrazów Ω ; dla którego spełnione są następujące warunki:*

1) *warunek dodatniości:*

$$P(\mathbf{X} = \mathbf{x}) > 0 \text{ dla każdego } \mathbf{x} \in \Omega, \text{ przy czym } \mathbf{X} = \{X_1, \dots, X_M\},$$

2) *warunek Markowa:*

$$P(X_i = x_i \mid x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_M) = P(X_i = x_i \mid \text{sąsiedzi } i).$$

Dla konkretnego pola Markowa \mathbf{X} , o wartościach z pewnej ustalonej przestrzeni obrazów Ω , będącej zbiorem odwzorowań $S \rightarrow L^M$, zbiór L jest zbiorem stanów tego pola Markowa¹.

Z definicji 3 wynika również pojęcie sąsiada piksela i . Sąsiadami piksela i nazywane są wszystkie piksele j takie że: $P(X_i = x_i \mid x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_M)$ zależy od x_j .

Lemat 1 *Piksel i jest sąsiadem piksela j wtedy i tylko wtedy, gdy piksel j jest sąsiadem piksela i .*

W praktyce arbitralnie określa się, które piksele należące do dziedziny pola losowego są sąsiadami piksela i . Najczęściej dokonuje się tego na podstawie kryterium „odpowiednio małej” odległości euklidesowej w przestrzeni Z^n , przy czym n - wymiar przestrzeni, na której pole Markowa jest określone. Zazwyczaj ustala się pewną wartość progową odległości, taką, że wszystkie piksele oddalone więcej niż ta wartość od piksela aktualnego nie mają już wpływu na rozkład prawdopodobieństwa informacji w aktualnym pikselu.

Poniżej przedstawiono definicje jeszcze dwóch pojęć wiążących się z polami Markowa

¹W przypadku obrazów z gradacją szarości, L jest zazwyczaj przedziałem domkniętym ze zbioru liczb naturalnych $[0, L_{max} - 1]$, przy czym $L_{max} = 2^n$, natomiast a jest liczbą bitów przypadającą na jeden piksel.

Definicja 4 Sąsiedztwem \mathcal{S}_i piksla i nazywamy zbiór wszystkich piksli j , które są sąsiadami i .

Definicja 5 Systemem sąsiedztwa \mathcal{S} dziedziny obrazu S nazywamy zbiór sąsiedztw \mathcal{S}_i wszystkich piksli i należących do tej dziedziny.

RODZAJE PÓL MARKOWA

Pola Markowa można podzielić na dwie następujące klasy:

- przyczynowe;
- nieprzyczynowe.

Przyczynowe pola Markowa znane są w literaturze pod angielską nazwą *Markov Mesh Models*. W rzeczywistości, są one praktycznie bezpośrednim przeniesieniem pojęć z teorii łańcuchów Markowa na sygnały o więcej niż jednej zmiennej niezależnej. W modelach tych wprowadzono pojęcie przeszłości danego piksla. Oznacza to, że przynależność do sąsiedztwa jest zdeterminowana nie tylko bliskością geometryczną, ale także kierunkiem. Innymi słowy zakłada się, że nie wszystkie najbliższe położone piksle mają wpływ na rozkład prawdopodobieństwa informacji w pikslu rozważanym. Prace prowadzone przez niektórych autorów pokazały, że pomimo takich założeń, zastosowanie modeli przyczynowych w wizji komputerowej prowadzi niekiedy do całkiem dobrych rezultatów [JW91, Lac87, QT91]. W praktyce dość często stosuje się gaussowskie przyczynowe pola Markowa [JW91]. Formalnie, pole takie można opisać następującymi zależnościami:

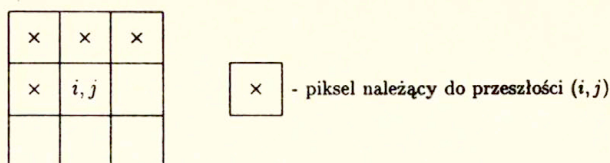
$$f(i, j) = \sum_{(k, l) \in R^+} c_{kl} \cdot f(i - k, j - l) + w(i, j), \quad (2.7)$$

przy czym: $R^+ = \{(k, l) : (-k, -l) \in R^+(0, 0)\}$ – przeszłość piksla, $f(i, j)$ – jasność w pikslu o współrzędnych (i, j) , $w(i, j)$ – biały szum, c_{kl} – współczynniki. Na rys. 2.3 przedstawiono przykład sąsiedztwa R^+ przyczynowego pola Markowa [JW91].

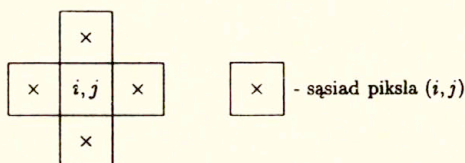
Zaletą przyczynowych pól Markowa jest to, że przeglądając obraz piksel po pikslu w pewnym określonym porządku można, znając rozkłady warunkowe przejścia z jednego stanu (np. jednego poziomu jasności) do drugiego, przeprowadzić symulację stochastyczną takiego pola. Tak więc, znając wartości funkcji jasności z przeszłości oraz macierz transycji, można dokonać predykcji informacji związanej z rozważanym pikselem.

Nieprzyczynowe pola Markowa. W przypadku nieprzyczynowych pól Markowa pojęcie przeszłości traci sens. Jako kryterium sąsiedztwa przyjmuje się jedynie możliwie małą wartość odległości euklidesowej. Rozkład prawdopodobieństwa informacji w danym pikslu zależy od jej próbek we wszystkich pikslach położonych odpowiednio blisko względem piksla rozważanego. Formalnie, w przypadku gaussowskich nieprzyczynowych pól Markowa, zależność taka daje się opisać za pomocą wzorów [JW91]:

$$f(i, j) = \sum_{(k, l) \in R} c_{kl} \cdot f(i - k, j - l) + w(i, j), \quad (2.8)$$



Rys. 2.3: Przykład sąsiedztwa przyczynowego pola Markowa.



Rys. 2.4: Przykładowe sąsiedztwo nieprzyczynowego pola Markowa.

przy czym: $f(i, j)$ – jasność w pikselu o współrzędnych (i, j) , c_{kl} – współczynniki, R – sąsiedztwo piksela, $w(i, j)$ – pole losowe Gaussa o kowariancji:

$$E[w(i, j)w(k, l)] = \begin{cases} \sigma_w^2 & (i, j) = (k, l) \\ -c_{i-k, j-l} \cdot \sigma_w^2 & (i-k, j-l) \in R \\ 0 & \text{w pozostałych przypadkach.} \end{cases} \quad (2.9)$$

Najprostszy model sąsiedztwa R został przedstawiony na rys. 2.4.

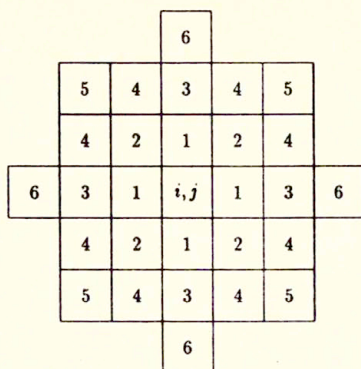
Nieprzyczynowe pola Markowa wydają się w pełni uwzględniać cechy charakterystyczne sygnałów, jakimi są obrazy, tzn. uwzględniają zależność informacji związanej z sąsiednimi pikselami. Ze względu na brak pojęcia przeszłości, predykcja informacji nie jest tak łatwa do przeprowadzenia jak w przypadku pól przyczynowych. Istnieją jednak metody, o czym będzie mowa w dalszej części tego rozdziału, pozwalające znajdować parametr położenia rozkładu pola Markowa, jakim jest jego moda.

2.2.4 Nieprzyczynowe pola Markowa

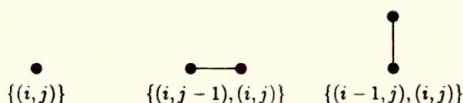
W dalszej części pracy zajmiemy się wyłącznie polami nieprzyczynowymi, nazywając je po prostu polami Markowa.

RZĄD POLA MARKOWA ORAZ POJĘCIE KLIKI

Pojęcie rzędu pola Markowa jest zbliżone do pojęcia rzędu łańcucha Markowa. Jak wiadomo, rząd łańcucha Markowa równa się liczbie próbek sygnału z przeszłości, które mają wpływ na rozkład prawdopodobieństwa informacji w chwili bieżącej. Podobnie dla pól



Rys. 2.5: Model sąsiedztwa pola Markowa w zależności od przyjętego rzędu.



Rys. 2.6: Rodzaje klik dla modelu sąsiedztwa rzędu 1.

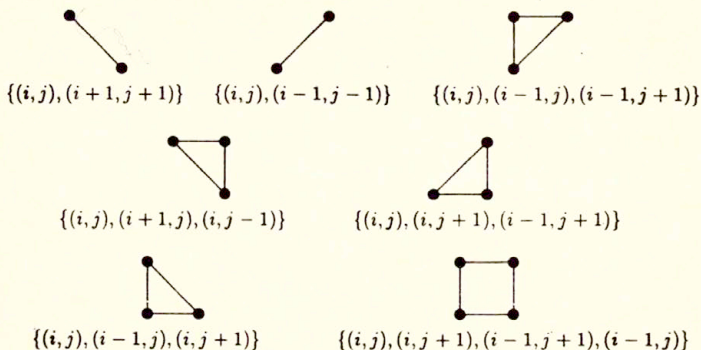
losowych. rząd określa, które piksele z otoczenia wpływają na rozkład prawdopodobieństwa w pikselu aktualnym, a więc determinuje przynależność do sąsiedztwa danego piksela. W przeciwieństwie do łańcuchów, zależność pomiędzy rzędem a liczbą sąsiadów piksela nie ma charakteru liniowego. Oznaczając sąsiedztwo piksela (i, j) przez \mathcal{S}_{ij} można wyrazić warunek przynależności piksela (k, l) do sąsiedztwa piksela (i, j) przez:

$$(k, l) \in \mathcal{S}_{ij} \Leftrightarrow 0 < (k - i)^2 + (l - j)^2 \leq c, \quad (2.10)$$

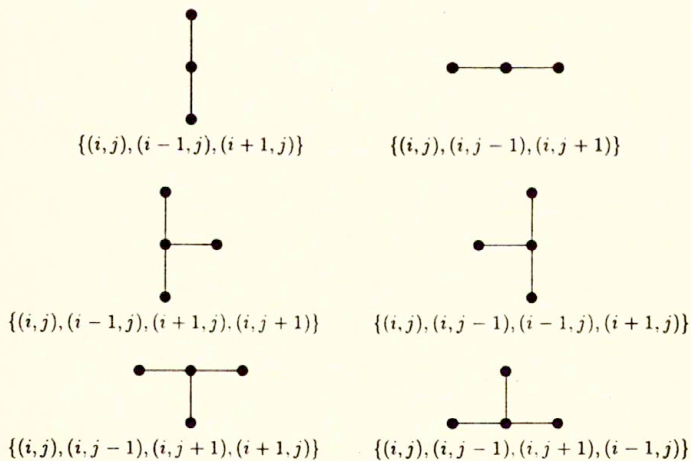
przy czym: c – pewna stała większa od zera.

Przyjmując $c = 1, 2, 4, 5, \dots$, otrzymujemy sąsiedztwa odpowiadające kolejnym rzędem 1, 2, 3, 4, ... pola Markowa. Na rys. 2.5 odpowiednimi liczbami zaznaczone są piksele, które należą do sąsiedztwa określonego rzędu, przy czym piksele o numerze mniejszym należą również do sąsiedztwa o numerze większym. Często w celu podkreślenia faktu, że wraz ze zmianą rzędu pola zmienia się kształt sąsiedztwa piksela, mówi się że rząd pola Markowa odpowiada za jego *model sąsiedztwa*. Pojęcia sąsiedztwa i modelu sąsiedztwa używane są wymiennie.

Rząd pola Markowa jest zazwyczaj parametrem wybieranym arbitralnie. Tylko niektórzy autorzy próbowali określić wpływ przyjętego rzędu na wynik działania proponowanych przez siebie algorytmów, na przykład w [CJ83], ale otrzymane wyniki wskazują raczej na



Rys. 2.7: Dodatkowe kliki dla modelu sąsiedztwa rzędu 2.



Rys. 2.8: Dodatkowe kliki dla modelu sąsiedztwa rzędu 3.

powierzchowne przebadanie zagadnienia.

Bezpośrednio z modelem sąsiedztwa, a więc i z rzędem pola Markowa, związane jest pojęcie klik. Poniżej zamieszczono jego definicję:

Definicja 6 *Każdy podzbiór dziedziiny pola losowego, taki że należy do niego pojedynczy piksel, bądź więcej niż jeden piksel, ale każde dwa należące do tego podzbioru piksele są sąsiadami, nazywa się kliką.*

Przy modelach sąsiedztwa pierwszego, drugiego i trzeciego rzędu mamy do czynienia z rodzajami klik przedstawionymi na rys. 2.6, 2.7 i 2.8. Oczywiście zbiór klik modelu wyższego rzędu zawiera również wszystkie klik modelu rzędu niższego. Zazwyczaj zbiór wszystkich klik obrazu oznacza się przez C , a przez C^i podzbiór klik i -pikselowych.

ZALEŻNOŚCI MATEMATYCZNE OPISUJĄCE POLA MARKOWA

Zgodnie z definicją 3 pole losowe może być formalnie opisane przy użyciu:

- łącznego, wielowymiarowego rozkładu prawdopodobieństwa [Bes74], oraz
- zbioru lokalnych, warunkowych rozkładów prawdopodobieństwa zwanych charakterystykami lokalnymi [GG84].

Gdyby model matematyczny miał wykorzystywać łączny rozkład prawdopodobieństwa, to w celu jego identyfikacji należałoby, bądź podać M -wymiarową funkcję tego rozkładu, przy czym M – liczba punktów modelowanego obszaru, lub funkcję gęstości w przypadku rozkładów ciągłych, bądź w jakiś sposób określić wszystkie rozkłady brzegowe aż do wymiaru $M - 1$. W praktyce, poza rozważaniami teoretycznymi, taki opis nie bardzo ma sens ze względu na wymiar łącznego rozkładu prawdopodobieństwa. Z tego powodu, model wykorzystujący lokalne charakterystyki wydaje się o wiele atrakcyjniejszy. Merytoryczna poprawność opisu pola losowego za pomocą warunkowych rozkładów prawdopodobieństwa została pokazana w [Bes74]. Oczywiście, jak to przedstawiono we wspomnianej pracy, istnieje ścisły związek pomiędzy obydwoma wymienionymi opisami.

Niech $\mathbf{x} = \{x_1, \dots, x_M\}$ i $\mathbf{y} = \{y_1, \dots, y_M\}$ będą dwiema realizacjami z przestrzeni obrazów Ω . Dzięki warunkowi dodatności z definicji 3, można pokazać słuszność następującego równania:

$$\frac{P(\mathbf{X} = \mathbf{x})}{P(\mathbf{X} = \mathbf{y})} = \prod_{i=1}^M \frac{P(X_i = x_i \mid x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_M)}{P(X_i = y_i \mid x_1, \dots, x_{i-1}, y_{i+1}, \dots, y_M)} \quad (2.11)$$

Dowód poprawności równania (2.11) zamieszczono w [Bes74]. Podstawiając $\mathbf{y} = \mathbf{x}_i$, przy czym $\mathbf{x}_i = \{x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_M\}$, tzn. obraz \mathbf{x}_i jest obrazem \mathbf{x} z wstawioną wartością 0 w pozycji i , oraz korzystając z pojęcia sąsiada piksela i , otrzymujemy:

$$\frac{P(\mathbf{X} = \mathbf{x})}{P(\mathbf{X} = \mathbf{x}_i)} = \frac{P(X_i = x_i \mid \text{sąsiedzi } i)}{P(X_i = 0 \mid \text{sąsiedzi } i)} \quad (2.12)$$

Istnieje ścisły związek pomiędzy rzędem pola Markowa, czyli liczebnością sąsiedztwa, a postacią funkcji opisujących lokalne rozkłady warunkowe. Zależność tę określa twierdzenie Hammersley'a-Cliforda (HC) przedstawione w [Bes74].

Twierdzenie 1 (Hammersley-Clifford) Niech będzie dany model sąsiedztwa pola Markowa oraz niech \mathbf{o} oznacza taki obraz z przestrzeni obrazów Ω , dla którego funkcja jasności jest równa zero w każdym pikselu. Przedstawmy funkcję wyrażającą stosunek $P(\mathbf{X} = \mathbf{x})$ i $P(\mathbf{X} = \mathbf{o})$, przy czym \mathbf{x} jest pewnym obrazem z przestrzeni Ω , w postaci:

$$\frac{P(\mathbf{X} = \mathbf{x})}{P(\mathbf{X} = \mathbf{o})} = \exp \left[\sum_{1 \leq i \leq M} x_i G_i(x_i) + \sum_{1 \leq i < j \leq M} x_i x_j G_{i,j}(x_i, x_j) + \dots + x_1 x_2 \dots x_M G_{1,2,\dots,M}(x_1, \dots, x_M) \right]. \quad (2.13)$$

Przy powyższych założeniach, funkcje $G_{i,j,\dots,s}(\dots)$ mogą być różne od zera wtedy i tylko wtedy, gdy $\{i, j, \dots, s\}$ stanowią klikę.

Bezpośrednim wnioskiem wynikającym z twierdzenia HC jest to, że lokalny i globalny warunek Markowa są ze sobą równoważne [Bes74], tzn.

$$\frac{P(X_i = x_i, X_j = x_j, \dots, X_s = x_s \mid \text{pozostałe piksele})}{P(X_i = x_i, X_j = x_j, \dots, X_s = x_s \mid \text{sąsiedzi } i, j, \dots, s)} \quad (2.14)$$

Dla przykładu przyjmijmy, że mamy do czynienia wyłącznie z klikami drugiego rzędu. Dzięki twierdzeniu 1 wzór (2.12) sprowadza się do postaci:

$$\frac{P(X_i = x_i \mid \text{sąsiedzi } i)}{P(X_i = 0 \mid \text{sąsiedzi } i)} = \exp \left\{ x_i \left[G_i(x_i) + \sum_j x_j G_{i,j}(x_i, x_j) \right] \right\}, \quad (2.15)$$

przy czym j jest sąsiadem i . Ponieważ, zgodnie z przyjętymi założeniami, x_i może przyjmować wartości ze skończonego zbioru L , więc korzystając z równania (2.15) i dokonując prostych przekształceń, można wyprowadzić wzór na prawdopodobieństwo warunkowe pojawienia się informacji x_i w pikselu i , czyli wzór charakterystyki lokalnej pola Markowa. Przyjmując

$$f_i(x_i) = x_i G_i(x_i) + x_i \sum_j x_j G_{i,j}(x_i, x_j), \quad \text{przy czym } j \text{ jest sąsiadem } i, \quad (2.16)$$

otrzymujemy

$$P(X_i = x_i \mid \text{sąsiedzi } i) = \frac{\exp[f_i(x_i)]}{\sum_{z_i \in L} \exp[f_i(z_i)]}, \quad (2.17)$$

przy czym L - zbiór stanów pola Markowa.

Korzystając ze wzoru (2.13) można zdefiniować następujące klasy pól Markowa:

- pola jednorodnie,
- pola izotropowe.

Definicja 7 Pole Markowa nazywane jest polem jednorodnym jeśli wszystkie funkcje $G_{i,j,\dots,K}(x_i, x_j, \dots, x_K)$ nie zależą od $\{i, j, \dots, K\}$, a zależą tylko od wartości x_i, x_j, \dots, x_K .

Definicja 8 Pole Markowa, dla którego wszystkie funkcje $G_{i,j,\dots,K}(x_i, x_j, \dots, x_K)$, dla klik o jednakowej liczebności, są identyczne, nazywa się polem izotropowym.

Zakładając izotropowość pola lub jego jednorodność, uzyskuje się znaczne uproszczenie modelu, a tym samym i obliczeń.

W przedstawionym opisie matematycznym pól Markowa za pomocą charakterystyk lokalnych niektórzy autorzy zauważają pewne wady [GG84], a mianowicie:

- łączny rozkład prawdopodobieństwa nie daje się wyrazić za pomocą wzoru matematycznego,
- trudno jest stwierdzić kiedy zadany zbiór funkcji $g(x_i | x_j)$, przy czym $i \neq j$, $\{x_1, \dots, x_M\} \in \Omega$, stanowi zbiór charakterystyk lokalnych pewnego pola Markowa.

W pracach [Bes74, Bes86] przedstawiono kilka przykładów charakterystyk lokalnych pola Markowa spełniających założenia twierdzenia HC. Jednym z częściej używanych jest rozkład dwumianowy [CJ83, por87], w którym parametr Θ_{ij} tego rozkładu zależy od wartości funkcji jasności w pikselach sąsiednich oraz spełniony jest dla niego warunek:

$$\Theta_{ij} = \frac{\exp(T_{ij})}{1 + \exp(T_{ij})}, \quad (2.18)$$

przy czym T_{ij} (przy założeniu że rząd pola Markowa wynosi 1) jest równe:

$$T_{ij} = a(i, j) + b_{11}(i, j)(x_{i,j-1} + x_{i,j+1}) + b_{12}(i, j)(x_{i-1,j} + x_{i+1,j}). \quad (2.19)$$

W takim przypadku charakterystyka lokalna pola Markowa przybiera postać:

$$P(X_{ij} = x_{ij} | sąsiedzi(i, j)) = \binom{L_{max}}{x_{ij}} \Theta_{ij}^{x_{ij}} (1 - \Theta_{ij})^{L_{max} - x_{ij}}, \quad (2.20)$$

przy czym L_{max} – wartość maksymalnej jasności obrazu modelowanego polem Markowa.

Zakładając izotropowość i jednorodność takiego dwumianowego pola, liczba nieznanymi parametrów redukuje się do dwóch. Gdy rząd pola jest większy niż 1, to w zaprezentowanym modelu, wzrasta liczba parametrów b o liczbę nowych rodzajów klik pojawiających się w modelu sąsiedztwa.

ROZKŁAD GIBBSA

W [DE87] przedstawiono następującą definicję:

Definicja 9 Niech \mathcal{S} będzie systemem sąsiedztwa, wg definicji 5, dziedziny obrazu S , przy czym $S = [0, \dots, N - 1] \times [0, \dots, N - 1]$. Pole losowe $\mathbf{X} = \{X_{ij}\}$ z systemem sąsiedztwa \mathcal{S} o łącznej gęstości prawdopodobieństwa w postaci:

$$P(\mathbf{X} = \mathbf{x}) = \frac{1}{Z} \exp[-W(\mathbf{x})], \quad (2.21)$$

przy czym:

$$W(\mathbf{x}) = \sum_{c \in \mathcal{C}} V_c(\mathbf{x}) - \text{energia obrazu } \mathbf{x},$$

$V_c(\mathbf{x})$ – energia klik² c ,
 $Z = \sum_{\mathbf{x} \in \Omega} \exp[-W(\mathbf{x})]$ – funkcja rozdziału (stała normalizacji),
 jest polem losowym o rozkładzie Gibbsa z systemem sąsiedztwa S .

Energia klik $V_c(\mathbf{x})$ jest taką funkcją określoną na obrazie \mathbf{x} , która wszędzie poza pikselami należącymi do klik c musi przyjmować wartość zero. Przedstawiony rozkład Gibbsa należy do wykładniczej rodziny rozkładów [DE87]. W wielu opracowaniach funkcję energii W definiuje się poprzez nową funkcję $W'(\mathbf{x})$ taką, że $W(\mathbf{x}) = \frac{1}{T}W'(\mathbf{x})$, przy czym T jest parametrem zwanym temperaturą przez analogię z fizycznymi układami opisanymi rozkładem Gibbsa. Taką definicję funkcji energii uzasadnia fakt, że gdy jest ona wykorzystywana w algorytmach wyzarczania, to poprzez powolną zmianę parametru T znajdowane jest globalne minimum energii [GG84].

W [GG84] podano, że wartość parametru T odpowiada za uwypuklenie lub spłaszczenie maksimum rozkładu Gibbsa. Gdy T jest duże, to postać $P(\mathbf{X})$ zbliża się do rozkładu jednostajnego na Ω , gdy T maleje, uwypuklają się mody tego rozkładu.

Charakterystyki lokalne rozkładu Gibbsa wyrażone są wzorami [GG84]:

$$P(X_i = x_i | x_j, j \neq i) = \frac{P(\mathbf{X} = \mathbf{x})}{\sum_{\mathbf{x}_i \in \Omega} P(\mathbf{X} = \mathbf{x}_i)}, \quad (2.22)$$

przy czym: $i \in S$, $x_i = \mathbf{x}$ poza pikselem i .

Chociaż dla rozkładu Gibbsa łatwo można podać ogólny wzór określający funkcję łącznego rozkładu, to zazwyczaj w praktyce określenie wartości parametru Z jest niemożliwe. Jednak nie wzorze (2.22) parametr Z ulega redukcji, a dzięki temu, gdyby były znane wszystkie energie klik, to równanie (2.22) nadawało by się do wykorzystania w obliczeniach komputerowych. Często przyjmuje się, że energie klik mają postać [DE87]:

$$V_c(x) = \begin{cases} -\zeta & \text{dla klik kilkupikselowych jeśli wszystkie } x_{ij}, (i, j) \in c, \text{ są równe} \\ \zeta & \text{dla klik kilkupikselowych w pozostałych przypadkach} \\ \alpha_k & \text{dla klik jednopikselowych gdy } x_{ij} = l_k, l_k \in L, \end{cases} \quad (2.23)$$

przy czym ζ może przyjmować różne wartości dla różnego rodzaju klik.

W pracy [GG84] zajęto się właściwościami łańcuchów losowych, w przypadku których stanami są obrazy \mathbf{x} należące do pewnej przestrzeni obrazów Ω . Zakłada się, że obrazy \mathbf{x} opisane są pewnym rozkładem Gibbsa. W pracy tej sformułowano i udowodniono trzy twierdzenia dotyczące takich łańcuchów losowych.

Twierdzenie 2 (Relaksacja) Niech n_t , przy czym $t = 1, 2, \dots$, będzie pewnym ciągiem, według którego przeglądane są piksele należące do dziedziny obrazu S . Niech w każdym kroku w aktualnym pikselu będzie dokonywana próba zmiany przyporządkowanej temu pikselowi jasności w następujący sposób: losujemy liczbę z rozkładu charakterystyki lokalnej danego

²W literaturze anglojęzycznej $V_c(x)$ określane jest mianem funkcji potencjału klik c . Jednak, jak wiadomo z fizyki, energia nie może być sumą potencjałów, dlatego w niniejszej pracy zdecydowano się na wprowadzenie pojęcia energii klik.

piksla. Otrzymujemy obraz \mathbf{x}' mogący się różnić od aktualnego obrazu \mathbf{x} tylko w jednym pikslu. Liczymy stosunek:

$$q = \frac{P(\mathbf{X} = \mathbf{x}')}{P(\mathbf{X} = \mathbf{x})}.$$

Jeśli $q > 1$, to nową jasność przyporządkowujemy aktualnemu pikslowi. W przeciwnym wypadku losujemy liczbę z rozkładu równomiernego na $[0, 1]$. Jeśli q jest większe bądź równe wylosowanej liczbie, to aktualnemu pikslowi przyporządkowujemy nową jasność, w przeciwnym przypadku nie dokonujemy żadnych zmian. Założmy, że każdy piksel występuje w ciągu n_i nieskończenie często. Wtedy, dla dowolnego obrazu początkowego $\mathbf{x}_0 \in \Omega$ i dla każdego $\mathbf{x} \in \Omega$ rozkład graniczny

$$\lim_{t \rightarrow \infty} P(\mathbf{X}(t) = \mathbf{x} \mid \mathbf{X}(0) = \mathbf{x}_0)$$

jest rozkładem Gibbsa.

W twierdzeniu 2 podano sposób znajdowania obrazów będących realizacjami pewnego pola losowego Gibbsa. Niestety sposób ten wymaga przeprowadzenia symulacji łańcucha losowego. Teoretycznie symulacja taka powinna być nieskończenie długa. W praktyce liczbę kroków symulacji można ograniczyć, ale i tak wydaje się to być procedura bardzo czasochłonna.

Aby sformułować następne twierdzenie przyjmijmy: $W^* = \max_{\mathbf{x} \in \Omega} W(\mathbf{x})$, $W_* = \min_{\mathbf{x} \in \Omega} W(\mathbf{x})$.

Twierdzenie 3 (Wyżarzanie) Założmy, że istnieje naturalna liczba $\tau \geq M$ taka, że dla każdego $t = 0, 1, 2, \dots$ zachodzi:

$$S \subseteq \{n_{t+1}, n_{t+2}, \dots, n_{t+\tau}\},$$

przy czym: n_t jest sekwencją jak w twierdzeniu 2. Niech $T(t)$ będzie dodatnią, malejącą funkcją temperatury, taką że:

$$T(t) \rightarrow 0 \text{ jeśli } t \rightarrow \infty,$$

$$T(t) \geq \frac{M\Delta}{\log t} \text{ dla wszystkich } t \geq t_0 \text{ dla pewnego } t_0 \geq 2,$$

przy czym: M - liczba pikseli należących do dziedziny obrazu, $\Delta = W^* - W_*$. Przy tych założeniach dla dowolnego obrazu początkowego $\mathbf{x}_0 \in \Omega$ i dla każdego obrazu $\mathbf{x} \in \Omega$ rozkład graniczny

$$\lim_{t \rightarrow \infty} P(\mathbf{X}(t) = \mathbf{x} \mid \mathbf{X}(0) = \mathbf{x}_0)$$

jest rozkładem równomiernym na Ω_0 , przy czym $\Omega_0 = \{\mathbf{x} \in \Omega : W(\mathbf{x}) = \min_{\mathbf{y} \in \Omega} W(\mathbf{y})\}$.

W twierdzeniu 3 podany jest sposób estymowania najbardziej prawdopodobnej realizacji rozkładu Gibbsa, tzn. *mody* tego rozkładu. Metoda oparta jest na symulacji łańcucha losowego, a więc jest bardzo czasochłonna.

Następne twierdzenie dotyczy właściwości ergodyczności łańcucha losowego, którego stanami są obrazy opisane pewnym rozkładem Gibbsa.

Twierdzenie 4 (Ergodyczność) Załóżmy, że istnieje liczba naturalna $\tau \geq M$ taka, że

$$S \subseteq \{n_{t+1}, n_{t+2}, \dots, n_{t+\tau}\}$$

dla każdego t , przy czym n_t jest ciągiem jak w Twierdzeniu 3. Przy tym założeniu, dla każdej funkcji Y określonej na Ω i dla dowolnego obrazu początkowego $\mathbf{x}_0 \in \Omega$, zależność:

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{t=1}^n Y(\mathbf{X}(t)) = \int_{\Omega} Y(\omega) d\pi(\omega)$$

zachodzi z prawdopodobieństwem równym 1.

Ergodyczności łańcucha losowego, którego stanami są obrazy, w zastosowaniach praktycznych nie jest dotychczas wykorzystywana. Jednak dzięki tej właściwości możliwym jest estymowanie obrazu będącego wartością średnią pewnego rozkładu Gibbsa.

RÓWNOWAŻNOŚĆ POMIĘDZY POLAMI MARKOWA A ROZKŁADEM GIBBSA

Okazuje się, że modelowanie obrazów za pomocą pól Markowa oraz ich opis przy użyciu rozkładu Gibbsa są ze sobą równoważne. W pracy [GG84] podano następujące twierdzenie:

Twierdzenie 5 (Równoważność) Niech \mathcal{S} będzie pewnym systemem sąsiedztwa pewnego pola losowego. Zmienna losowa \mathbf{X} opisuje pewne pole Markowa z systemem sąsiedztwa \mathcal{S} , wtedy i tylko wtedy, gdy $P(\mathbf{X})$ ma rozkład prawdopodobieństwa Gibbsa z systemem sąsiedztwa \mathcal{S} .

Dzięki takiemu twierdzeniu wzory charakterystyk lokalnych pola Markowa można wyrazić za pomocą energii klik [GG84]:

$$P(X_i = x_i \mid x_j \neq i) = \frac{\exp[-\frac{1}{T} \sum_{c: i \in c} V_c(\mathbf{x})]}{\sum_{x_i \in L} \exp[-\frac{1}{T} \sum_{c: i \in c} V_c(\mathbf{x}_i)]}. \quad (2.24)$$

Rozkłady Gibbsa i pola Markowa używane są wymiennie. Czasami spotyka się również w literaturze określenie: *pole losowe Markowa-Gibbsa*.

Jeżeli wybrano już postać pola Markowa-Gibbsa następnym etapem jest przeprowadzenie *estymacji* wartości parametrów tego pola, zwanej również w niniejszej rozprawie *identyfikacją*.

ESTYMACJA PARAMETRÓW PÓL MARKOWA-GIBBSA

Problem określenia wartości parametrów wybranego do obliczeń modelu jest typowy dla wszystkich zagadnień, dla rozwiązania których konstruuje się jakiegokolwiek modele. W każdym zastosowaniu ma on bardzo duże znaczenie, gdyż faktycznie odpowiada za dopasowanie budowanego modelu do sytuacji, która jest nim opisywana. Oczywiście koncepcja

modelu musi odpowiadać modelowanemu zjawisku, gdyż w innym przypadku identyfikacja może nie bardzo mieć sens.

W przypadku modeli statystycznych, identyfikacja sprowadza się do estymacji wartości parametrów poszczególnych rozkładów, zwanych charakterystykami lokalnymi. Statystyka matematyczna oferuje wiele różnych metod estymacji parametrów rozkładów losowych [Zie90, Bob86]. Zazwyczaj oszacowanie wartości parametru polega na obliczeniu wartości pewnej statystyki określonej na próbie, według jakiegoś wcześniej przyjętego kryterium, na przykład największej wiarygodności czy minimum wariancji. Niestety, praktycznie wszystkie metody statystyki matematycznej korzystają z założenia, że poszczególne obserwacje są od siebie niezależne. W przypadku pól Markowa nie jest to prawdą, gdyż już przy samej ich definicji zakłada się zależność próbek informacji związanych z sąsiednimi pikslami.

Innym założeniem statystyki matematycznej dotyczącym obserwacji jest przyjęcie reprezentatywności próby [Zie90]. W zadaniach wizji komputerowej spełnienie tego założenia nie jest takie oczywiste. Z ogromnej przestrzeni obrazów mamy najczęściej do dyspozycji tylko jedną próbkę, która wcale nie musi być reprezentatywną dla całego rozkładu pola Markowa. O ile w metodach wizji komputerowej pierwszą z wymienionych trudności udaje się jakoś pokonać, o tyle drugi problem jest najczęściej nierozwiązywalny. Zazwyczaj przyjmuje się *ad hoc*, że dostępna realizacja jest reprezentatywna dla całego rozkładu. Niekiedy problem reprezentatywności rozwiązuje się poprzez opracowywanie algorytmów z iteracyjną identyfikacją parametrów [Pie92a, Vei91]. W metodach wizji komputerowej do estymacji parametrów wykorzystuje się metody:

- kodingów [Bes74, CJ83],
- maksimum pseudowiarogodności [Bes86],
- najmniejszych kwadratów [Cha88, DE87],
- korzystające ze szczególnych właściwości niektórych rozkładów losowych [Cha88],
- adaptacyjne [LD89].

Metoda kodingów opiera się na podziale dostępnych danych obrazowych na segmenty, w których znajdują się takie piksele, że próbki informacji z nimi związane są między sobą niezależne. Na każdym z takich segmentów dokonuje się estymacji poszukiwanych parametrów, na przykład metodą największej wiarygodności. Następnie dokonuje się uśrednienia odpowiednich wyestymowanych wartości ze wszystkich segmentów (kodingów). Przykłady wykorzystania tej metody można znaleźć w pracach [Bes74, CJ83].

Metoda maksimum pseudowiarogodności jest właściwie modyfikacją metody maksimum wiarygodności [Zie90]. Obliczenia są prowadzone tak samo jak w metodzie maksimum wiarygodności, ale wykorzystuje się próbki informacji z wszystkich pikseli, tak jakby były między sobą niezależne. Świadomie rezygnuje się z optymalności otrzymanego w ten sposób rozwiązania. W praktyce okazuje się, że mimo takiego uproszczenia, metoda prowadzi do całkiem dobrych rezultatów. Metoda jest bardzo często wykorzystywana w wielu zastosowaniach [Bes86, Cha88, Pie92a].

Metoda najmniejszych kwadratów czasami jest używana samodzielnie [DE87], ale częściej jako metoda uzupełniająca w połączeniu z którąś z pozostałych. W szczególności korzysta się z niej, jako metody pomocniczej dla metod z następnej grupy.

Metody korzystające ze szczególnych właściwości niektórych rozkładów losowych zazwyczaj wykorzystują własności rozkładu normalnego [Cha88]. Dużo zależy od pomysłowości samego autora, ale znalezione w ten sposób estymatory spełniają najczęściej warunki optymalności.

Metody adaptacyjne są zazwyczaj połączeniem etapu estymacji parametrów pola Markowa i etapu znajdowania mody tego rozkładu. W ogromnej większości przypadków wygląda to tak, że algorytm symulowanego wyżarzania jest przerywany co pewien czas w celu dokonania nowej estymacji parametrów pola, na przykład metodą maksimum pseudowiarygodności, po czym znowu wyżarzanie jest kontynuowane, ale przy wykorzystaniu już nowych wartości parametrów. W literaturze podano dowody, że takie postępowanie prowadzi do znalezienia globalnego minimum energii rozkładu Gibbsa [LD89].

Jeżeli znane są już wartości parametrów wybranego pola Markowa-Gibbsa, możliwym staje się estymowanie parametrów pozycyjnych jego rozkładu [por87]. Jednym z częściej poszukiwanych parametrów pozycyjnych jest realizacja pola Markowa-Gibbsa o największym prawdopodobieństwie, zwana **wartością modalną**, lub po prostu **modą** rozkładu pola Markowa-Gibbsa.

ALGORYTMY ESTYMACJI MODY POLA MARKOWA-GIBBSA

W komputerowej analizie obrazów [GG84, Bes86, CJ83] moda wydaje się być jednym z najważniejszych parametrów rozkładu losowego Markowa-Gibbsa. Konstrukcja wielu algorytmów komputerowej analizy obrazów oparta jest na estymacji tego parametru. Ze względu na właściwość nieprzyczynowości omawianych modeli, estymacja mody wymaga różnych zabiegów.

Większość znanych metod korzysta z technik relaksacyjnych, co oznacza, że przeprowadzana jest symulacja pewnego łańcucha losowego, którego stanami są obrazy. Odrębną grupę stanowią metody opierające się na programowaniu dynamicznym.

Poniżej przedstawiono trzy znane z literatury algorytmy estymacji mody rozkładu losowego pola Markowa-Gibbsa.

Symulowane wyżarzanie – stochastyczny algorytm estymacji mody rozkładu pola Markowa. Nazwa *symulowane wyżarzanie* wywodzi się z fizyki [KGV83]. Algorytm symulowanego wyżarzania podawany jest w literaturze jako przykład *relaksacji stochastycznej* [Jin90, GG84]. Metoda została opracowana dla rozkładu Gibbsa, ale dzięki twierdzeniu 5 można ją również wykorzystać do znajdowania mody rozkładu pola Markowa. Podstawy matematyczne stanowią twierdzenia 2, 3. W pracy [GG84] metoda ta została nazwana *impulsatorem Gibbsa* i w tej wersji zostanie poniżej przedstawiona.

Na początku ustalany jest pewien sposób przeglądania, krok po kroku, wszystkich pikseli należących do obrazu. W praktyce, często przyjmuje się kolejność naturalną, tzn. wierszami z lewa na prawo. Przejrzenie całego obrazu określa się nazwą iteracji. Następnie wybierana jest pewna dodatnia, malejąca funkcja temperatury w zależności od numeru iteracji. Dokonany wybór musi gwarantować optymalność, lub przynajmniej suboptymalność, otrzymanego później rozwiązania. Wartość parametru temperatury z funkcji gęstości rozkładu Gibbsa jest obniżana tylko przed rozpoczęciem nowej iteracji, a podczas samej iteracji pozostaje stała. W każdym kolejnym kroku algorytmu próbuje się dokonać zmiany jasności w aktualnym pikselu. Algorytm dla piksla i w kroku k przedstawia się następująco:

1. Losujemy liczbę z rozkładu warunkowego $P(X_i = x_i \mid x_j, j \neq i)$ obrazu \mathbf{x}_k według wzoru (2.24). W ten sposób otrzymujemy obraz \mathbf{x}'_k , mogący się różnić od poprzedniego tylko wartością w jednym pikselu i .
2. Liczymy stosunek:

$$q = \frac{P(\mathbf{X} = \mathbf{x}'_k)}{P(\mathbf{X} = \mathbf{x}_k)} = \frac{\exp[-\frac{1}{T} \sum_{c_i \in \mathcal{C}} V_c(\mathbf{x}'_k)]}{\exp[-\frac{1}{T} \sum_{c_i \in \mathcal{C}} V_c(\mathbf{x}_k)]}. \quad (2.25)$$

3. Jeśli stosunek q jest większy od 1, to obraz \mathbf{x}'_k podstawiamy za \mathbf{x}_k i przechodzimy do kroku $k + 1$.
4. Jeśli q jest mniejsze bądź równe 1, to losujemy liczbę z rozkładu równomiernego na przedziale $[0, 1]$. Gdy q jest większe bądź równe wylosowanej liczbie, to obraz \mathbf{x}'_k podstawiamy za \mathbf{x}_k i przechodzimy do kroku $k + 1$, w przeciwnym przypadku przechodzimy do kroku $k + 1$ bez żadnych zmian.

Zgodnie z twierdzeniami 2 i 3 opisane czynności muszą być powtórzone nieskończenie wiele razy dla danej temperatury. Twierdzenia 2 i 3 gwarantują, że postępując w przedstawiony sposób, zostanie znaleziona moda rozkładu Gibbsa. Ponieważ algorytm zaliczany jest do grupy algorytmów stochastycznych, więc dopuszcza lokalnie zarówno zmniejszenie jak i zwiększenie minimalizowanej funkcji energii. To powoduje, że wartość energii kolejno generowanych obrazów nie dąży do lokalnego minimum, a znajdowane jest minimum globalne [GG84, Bes86]. W praktyce, liczbę iteracji ogranicza się co najwyżej do kilku tysięcy [GG84] lub nawet kilkunastu [Cha88], chociaż globalna optymalność rozwiązania w ostatnim przypadku może być zakwestionowana.

Zaletą zaprezentowanej metody jest jej globalna optymalność, wadą czas dochodzenia do rozwiązania. Badania przeprowadzone przez niektórych autorów pokazały, że w porównaniu z innymi, szybszymi metodami, minimum energii znalezione dzięki symulowanemu wyżarzaniu jest rzeczywiście globalne [HMB91].

ICM – deterministyczny algorytm estymacji mody rozkładu pola Markowa. Metoda została zaproponowana przez Besagą w pracy [Bes86] w roku 1986. Jest typowym przykładem algorytmu relaksacji deterministycznej [Jin90]. W wolnym tłumaczeniu nazwę ICM można podać jako „iteracyjnie poszukiwane mody rozkładu warunkowego”. W przeciwieństwie do symulowanego wyżarzania, najczęściej wynikiem jej działania jest lokalne maksimum funkcji rozkładu prawdopodobieństwa pola Markowa, ale za to czas

dochodzenia do rozwiązania jest stosunkowo krótki. Wielu autorów korzysta z tej metody jawnie, inni się do niej w swoich pracach odwołują [Pie92b]. Metoda została wprowadzona przy okazji odszumiania obrazów, dlatego też w jej opisie pojawiają się takie pojęcia, jak obraz zaszumiony i obraz niezaszumiony. Poniżej przedstawiono opis algorytmu ICM.

Założenia wstępne:

- obraz niezaszumiony modelowany jest wielowymiarową zmienną losową X ($X = \{X_1, \dots, X_M\}$), o której zakłada się, że ma rozkład pola Markowa-Gibbsa,
- obraz zaszumiony modelowany jest wielowymiarową zmienną losową Y , przy czym $Y = \{Y_1, \dots, Y_M\}$,
- każda zmienna losowa Y_i ma tę samą funkcję gęstości warunkowego rozkładu prawdopodobieństwa $f(y_i | x_i)$, zależną tylko od x_i .
- zmienne Y_i są warunkowo niezależne,

Poszukiwana jest estymata obrazu niezaszumionego X . Oznaczmy przez $P(X_i = x_i | x_{\partial i})$ charakterystykę lokalną pola Markowa X w pikselu i , przy czym $x_{\partial i}$ oznacza wartość funkcji jasności we wszystkich pikselach należących do sąsiedztwa piksela i . Algorytm ICM polega na maksymalizacji w każdym pikselu prawdopodobieństwa $P(X_i = x_i | y, x_j, j \neq i)$. Okazuje się, że dzięki przyjętym założeniom [Bes86]:

$$P(X_i = x_i | y, x_j, j \neq i) \propto f(y_i | x_i)P(X_i = x_i | x_{\partial i}). \quad (2.26)$$

Algorytm: Niech k oznacza numer iteracji algorytmu, iteracja – przejście wszystkich pikseli obrazu, oraz niech \hat{x} oznacza estymatę obrazu niezaszumionego z poprzedniej iteracji:

1. Znajdź początkowy obraz \hat{x} , np. poprzez maksymalizację $f(y_i | x_i)$ względem x_i w każdym pikselu i . Wyzeruj wartość licznika iteracji ($it = 0$).
2. Rozpocznij nową iterację ($it = it + 1$). Wyzeruj licznik pikseli ($pk = 0$).
3. Wybierz kolejny piksel ($pk = pk + 1$). Załóżmy, że jest to piksel i -ty. W pikselu tym znajdź taką wartość \hat{x}_i , która maksymalizuje:

$$P(X_i = x_i | y, \hat{x}_j, j \neq i) \propto f(y_i | x_i)P(X_i = x_i | \hat{x}_{\partial i}). \quad (2.27)$$

4. Jeśli aktualny piksel nie był ostatnim ($pk \neq M$, przy czym M – liczba pikseli obrazu) skocz do 3.
5. Jeśli zakończona iteracja nie była ostatnią założoną ($it \neq it_{max}$) skocz do 2.
6. KONIEC.

W przypadku algorytmu ICM najczęściej liczba trzydziestu iteracji jest wystarczająca aby znaleźć estymatę mody pola Markowa.

Znajdowanie mody pola Markowa przy wykorzystaniu metod programowania dynamicznego. Programowanie dynamiczne jest dość często używane w zadaniach optymalizacji. Jest to właściwie numeryczna metoda rozwiązywania zadań sterowania optymalnego, wynikająca bezpośrednio z zasady optymalności. W pracy [DE87] zaproponowano zastosowanie metody programowania dynamicznego do znajdowania mody rozkładu pola Markowa, w celu dokonania segmentacji tego obrazu. W prezentowanym poniżej algorytmie zakłada się, że szum ma rozkład normalny o wartości średniej zero i wariancji σ^2 oraz, że jest to szum addytywny. Przyjmuje się, że dziedziina obrazu jest równa $S = [0, \dots, N - 1] \times [0, \dots, N - 1]$, N - liczba kolumn równa liczbie wierszy, $M = N \cdot N$ - liczba pikseli obrazu. Ponadto zakłada się, że liczba klas segmentacji jest stała i równa K . Metoda maksymalizuje logarytm wiarygodności a posteriori postaci:

$$\ln[P(\mathbf{X} = \mathbf{x} \mid \mathbf{Y} = \mathbf{y})] \propto \ln[P(\mathbf{X})] + \ln[P(\mathbf{Y} = \mathbf{y} \mid \mathbf{X} = \mathbf{x})] \quad (2.28)$$

względem \mathbf{x} , przy czym: \mathbf{X} - modelowane pole Markowa klas, \mathbf{y} - obraz obserwowany. Oznaczmy:

$$S_k = \{(i, j) \in S : X_{ij} = k\},$$

$Q = \{q_1, q_2, \dots, q_K\}$ - zbiór jasności przyporządkowanych poszczególnym klasom,

$C^{r-1,r} = \{c : c \text{ jest kliką której piksele należą tylko do kolumny } r \text{ lub tylko do kolumn } r \text{ oraz } r - 1\}$,

$$S_k^r = \{(i, r) : X_{ir} = k, 1 \leq i \leq N\}.$$

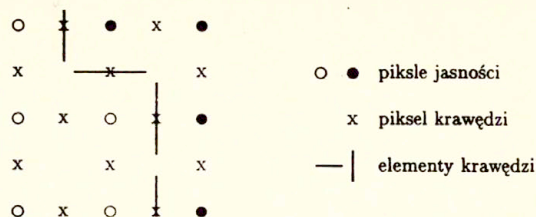
Jeśli przegląda się obraz po kolei, kolumna po kolumnie, to logarytm wiarygodności a posteriori wyraża się relacjami:

$$l_0 = -\ln Z - \frac{M}{2} \ln(2\pi\sigma^2)$$

$$l_r = l_{r-1} - \sum_{c \in C^{r-1,r}} V_c(\mathbf{x}) - \sum_{k=1}^K \sum_{(i,l) \in S_l^r} \frac{1}{2\sigma^2} (y_{il} - q_k)^2$$

przy czym: $r = 1, \dots, N$. Oczywiście $l_N = \ln[P(\mathbf{X} = \mathbf{x}, \mathbf{Y} = \mathbf{y})]$. Wartość l_0 nie ma wpływu na znajdowane optymalne rozwiązanie, a więc można przyjąć, że $l_0 = 0$ [DE87]. Algorytm programowania dynamicznego z przedstawionym wskaźnikiem jakości, po jednokrotnym przejściu obrazu, powoduje znalezienie globalnego maksimum wiarygodności a posteriori modelowanego pola Markowa. Niestety dla $N > 4$ algorytm jest praktycznie nie do zastosowania ze względu na zbyt duży nakład obliczeń. Z tego powodu w [DE87] zaproponowano wersję suboptymalną, nazwaną *algorytmem programowania dynamicznego na pasmie*. Algorytm wygląda następująco:

1. Wybierz liczbę wierszy D stanowiącą szerokość pasma, $2 \leq D \leq 4$.
2. Podstaw $I = 1$, przy czym I - kolejny numer wiersza.
3. Zastosuj algorytm programowania dynamicznego na pasmie złożonym z wierszy od I do $I + D - 1$.



Rys. 2.9: Przykład realizacji binarnego, hierarchicznego pola Markowa.

4. Jeśli $I + D - 1 = N$ zachowaj segmentację wierszy od I do $I + D - 1$, potem STOP.
5. Zachowaj segmentację wiersza I .
6. Podstaw $I = I + 1$.
7. Wróć do kroku 3.

Eksperymenty przeprowadzone w pracy [DE87] pokazały że omówiony algorytm, dla przypadków, w których spełnione są wymienione założenia, prowadzi do całkiem dobrych rezultatów.

2.2.5 Hierarchiczne pola Markowa

Hierarchiczne pola Markowa zostały wprowadzone już w pracy [GG84]. Zasadnicza różnica w stosunku do modeli omawianych dotychczas polega na tym, że rozkładem pola Markowa modelowany jest nie tylko obraz, który można zobaczyć, tzw. *górną warstwę*, ale również niewidoczne pole cech, tzw. *dolną warstwę*, ściśle związane z obserwowanym obrazem. Najczęściej dolną warstwę stanowi pole krawędzi i dlatego na jego przykładzie zostaną omówione hierarchiczne modele Markowa.

Górną warstwę stanowi pole Markowa modelujące obserwowaną funkcję jasności. Dla tego pola ustala się model sąsiedztwa, wybiera się postacie charakterystyk lokalnych oraz estymuje się wartości ich parametrów. Wymienione czynności są identyczne, jak dla modeli niehierarchicznych (jednowarstwowych).

Dolną warstwę stanowi pole Markowa krawędzi międzypikselowych określone w następujący sposób: pomiędzy dwoma elementami w jednym wierszu pola górnego występuje pionowy element krawędzi, jeśli różnica jasności przyporządkowanych tym dwóm pikselom jest odpowiednio duża. Podobnie, pomiędzy dwoma elementami w jednej kolumnie znajduje się poziomy element krawędzi, jeśli różnica dwóch odpowiednich jasności jest duża. Jeśli pole losowe jasności jest obrazem o $N \times N$ pikselach, to liczba pozycji pola krawędzi wynosi $2N(N - 1)$. Dla tak zdefiniowanego pola wybiera się następnie charakterystyki lokalne i oszacowuje ich parametry.

Na rys. 2.9 przedstawiono przykład realizacji hierarchicznego, binarnego pola Markowa. Zamalowane okręgi oznaczają piksele, którym przyporządkowana jest wartość 1 funkcji jasności, natomiast okręgi puste oznaczają piksele, w których funkcja jasności ma wartość 0.

Zależność pomiędzy dwoma wymienionymi warstwami jest definiowana poprzez odpowiednie określenie funkcji energii rozkładu Gibbsa. Różni autorzy radzą sobie rozmaicie z tym problemem. W pracy [GG84] funkcję energii pola hierarchicznego określono jako sumę

$$W_1(\mathbf{x} | \mathbf{l}) + W_2(\mathbf{l}), \quad (2.29)$$

przy czym: \mathbf{l} – pole krawędzi, a $W_1(\mathbf{x} | \mathbf{l})$ – energia warunkowa. W ogólnym przypadku *energia warunkowa* jest energią pola Markowa-Gibbsa określoną w taki sposób, że wartości energii poszczególnych klik są modyfikowane w zależności od dostępnej z zewnątrz informacji. W pracy [GG84] funkcja energii warunkowej $W_1(\mathbf{x} | \mathbf{l})$ zdefiniowana została następująco: jeśli pomiędzy dwoma pikselami pola opisującego funkcję jasności (warstwy górnej), w polu dolnym nie występuje element krawędziowy, to energia kliku ma taką wartość, jak została wybrana w trakcie modelowania warstwy górnej. Jeśli powyższy warunek nie jest spełniony, to energia takiej kliku z warstwy górnej przyjmuje wartość obojętną, np. jedną drugą wartości różnicy pomiędzy maksymalną i minimalną energią.

Przyjęcie warunkowej postaci energii nie jest jedynym sposobem modelowania wzajemnego wpływu warstw górnej i dolnej. W pracy [Cha88], we wzorze na całkowitą energię, zamiast składnika warunkowej energii $W_1(\mathbf{x} | \mathbf{l})$, wstawiono zwykłą energię pola opisującego funkcję jasności wraz z dodatkowym składnikiem energii zwany *energiją zgodności*. Energia zgodności odpowiada za efekt synfazowy.

Jak pokazały wyniki eksperymentów zamieszczone w [GG84, Cha88] obydwa sposoby uwzględnienia wzajemnego wpływu warstw górnej i dolnej prowadzą do dobrych wyników.

Hierarchiczne modele Markowa są bardziej skomplikowane niż pola jednowarstwowe. Jednak ze względu na skłonność modeli niehierarchicznych do przeglądania poprawiających obrazów, tzn. do tworzenia dużych obszarów jednolitej jasności [Bes86], najczęściej w praktycznych zastosowaniach z modeli hierarchicznych nie można zrezygnować.

2.2.6 Ukryte pola Markowa

W ukrytych polach Markowa-Gibbsa występują dwa rodzaje informacji:

1. pierwszym jest dostępna obserwacja, którą w przypadku komputerowej analizy obrazów jest najczęściej funkcja jasności obrazu,
2. drugim jest pewna cecha, przyporządkowywana poszczególnym pikselom analizowanego obrazu.

Oczywiście z matematycznego punktu widzenia obydwie informacje mogą przybierać postać wektorów.

Rozkład cechy modelowany jest polem Markowa. Realizacje tego pola nie są „obserwowane”, stąd też pole takie nazywa się *połem ukrytym*. O realizacjach pola ukrytego wiadomo tylko, że są w pewien sposób związane z dostępną obserwacją. Sposób wpływu

obserwacji na rozkład losowego pola ukrytego musi zostać określony w trakcie budowy modelu.

Dla ustalenia uwagi oznaczmy modelowane pole ukryte, opisujące rozkład pewnej cechy, przez X , a pole dostępnej obserwacji przez Y . W ogólnym przypadku zmienna losowa Y nie musi mieć rozkładu pola Markowa-Gibbsa. W praktyce zakłada się, że zmienna losowa Y przyjmuje tylko jedną realizację, tzn. rozważa się jedynie zdarzenie $Y = y$. Na wstępie budowy modelu ukrytego pola Markowa-Gibbsa należy wybrać system sąsiedztwa oraz energie poszczególnych klik. Następnie trzeba określić sposób wpływu obserwacji y na rozkład ukrytego pola X . Istnieją dwie zasadnicze możliwości [Car92]:

1. kryterium MAP – maksimum prawdopodobieństwa a posteriori, tzn. poszukuje się takiej realizacji \hat{x} pola X , która maksymalizuje rozkład a posteriori, czyli

$$\hat{x} = \arg \max P(X = x | Y = y). \quad (2.30)$$

2. kryterium MPM – wartości modalne brzegowych rozkładów a posteriori, tzn. poszukuje się realizacji \hat{x}_i takich, że:

$$\hat{x}_i = \arg \max P(X_i = x_i | Y = y). \quad (2.31)$$

Kryterium MAP jest wybierane znacznie częściej, dlatego na jego przykładzie zostanie omówiona ta część procesu budowy ukrytego pola Markowa, która dotyczy uwzględnienia wpływu obserwacji y na rozkład pola X .

Z twierdzenia Bayesa oraz faktu, że $P(Y = y)$ jest stałe, wynika, że maksymalizacja z równania (2.30) jest równoważna z maksymalizacją łącznego rozkładu prawdopodobieństwa $P(X = x, Y = y)$, ponieważ:

$$P(X = x | Y = y) \propto P(Y = y | X = x)P(X = x) = P(X = x, Y = y). \quad (2.32)$$

Rozkład $P(Y = y | X = x)$ zwany jest również *modelem obserwacji* [Car92]. Z równania (2.32) wynika, że jeśli model obserwacji jest rozkładem Gibbsa, to łączny rozkład prawdopodobieństwa $P(X = x, Y = y)$ też jest rozkładem Gibbsa. Zanim zostaną omówione dalsze szczegóły, wprowadźmy najpierw kilka pojęć pomocniczych.

- *Energiją ukrytego pola Markowa* nazywamy sumę wszystkich energii klik tego pola, ale bez uwzględnienia wpływu obserwacji, tzn. pola o rozkładzie prawdopodobieństwa $P(X = x)$.
- *Całkowitą energią ukrytego pola Markowa* określa się sumę energii wszystkich klik tego pola, z uwzględnieniem wpływu obserwacji, tzn. pola o rozkładzie $P(X = x | Y = y)$.
- *Energiją lokalną ukrytego pola Markowa w pewnym pikselu* nazywamy sumę energii wszystkich klik, do których należy ten piksel, ale bez uwzględnienia wpływu obserwacji.
- *Całkowitą energią lokalną ukrytego pola Markowa w pewnym pikselu* określa się sumę energii wszystkich klik, do których należy ten piksel, z uwzględnieniem wpływu obserwacji.

Innymi słowy przymiotnik *całkowita* informuje, że dane pojęcie odnosi się do rozkładu ukrytego pola Markowa z uwzględnionym wpływem obserwacji. Na zasadzie analogii można wprowadzić pojęcie *całkowanej charakterystyki lokalnej ukrytego pola Markowa*, która jest warunkowym rozkładem prawdopodobieństwa cechy związanej z pojedynczym pikselem pod warunkiem cech przyporządkowanych pikselom sąsiednim oraz obserwacji dostępczej dla danego piksela i wszystkich pikseli sąsiednich.

Powróćmy do uwzględnienia wpływu obserwacji y na rozkład pola X . W przypadku kryterium MAP, oraz gdy model obserwacji jest rozkładem Gibbsa, uwzględnienie wpływu obserwacji wymaga podania wyrażenia opisującego całkowitą energię ukrytego pola Markowa X . Podobnie jak w przypadku pól hierarchicznych tak samo dla modeli ukrytych istnieją dwa sposoby uwzględnienia wpływu obserwacji:

- za pomocą energii zgodności dodawanej do energii ukrytego pola Markowa, oraz
- za pomocą warunkowej energii pola ukrytego X .

Zazwyczaj wykorzystanie ukrytych pól Markowa polega na poszukiwaniu mody rozkładu tego pola z uwzględnieniem wpływu obserwacji, tzn. na minimalizacji całkowitej energii ukrytego pola Markowa.

Wykorzystanie modeli ukrytych wiąże się najczęściej ze znacznie większym nakładem obliczeniowym niż w przypadku pól jawnych jednowarstwowych. Identyfikacja modelu wymaga, poza estymacją parametrów charakterystyk lokalnych pola ukrytego, również estymacji parametrów modelu obserwacji. Zazwyczaj algorytmy wykorzystujące ukryte pola Markowa, równoległe z poszukiwaniem mody np. rozkładu a posteriori, dokonują iteracyjnej estymacji wszystkich parametrów, tzn. parametrów całkowitych charakterystyk lokalnych [Vei91, ABP92, Pie92a, Pie92b].

W publikacjach zajmujących się wykorzystaniem pól Markowa w komputerowej analizie obrazów często trudno jest rozróżnić, którego rodzaju modelu dany artykuł dotyczy: jednowarstwowego, hierarchicznego czy ukrytego. Wielu autorów, korzystających w swoich algorytmach z modeli pól Markowa, nie używa nazwy *ukryte pola Markowa*, chociaż je stosuje. Praktycznie wszystkie algorytmy detekcji różnych cech w obrazach, bądź segmentacji, korzystające z pól Markowa faktycznie korzystają z pól ukrytych [BL90, LB90, HB90a, HB90b, ZC90]. Przykładem braku spójności w nazewnictwie może być algorytm ICM, wprowadzony w [Bes86] i omówiony w niniejszym rozdziale. W przedstawionej wersji algorytm właściwie dotyczy pól ukrytych, a w całym artykule [Bes86] nawet się o tym nie wspomina.

Z matematycznego punktu widzenia hierarchiczne pola Markowa różnią się tym od pól ukrytych, że w przypadku tych pierwszych dwa źródła informacji opisywane są zmiennymi losowymi o rozkładzie pola Markowa-Gibbsa i poszukuje się „optymalnych” realizacji tych zmiennych jednocześnie. W przypadku pól ukrytych tylko jedno źródło informacji modelowane jest zmienną losową o rozkładzie pola Markowa-Gibbsa i tylko „optymalna” realizacja tego pola jest poszukiwana.

Rozdział 3

Ogólna metoda budowy modeli statystycznych opierających się na polach Markowa

Rozdział poświęcony jest metodzie budowy modeli statystycznych będących polami Markowa, a przeznaczonych do rozwiązywania problemów wizji komputerowej. Proponowana metoda jest uporządkowanym zbiorem pytań, na które trzeba udzielić odpowiedzi podczas modelowania jakiegoś problemu komputerowej analizy obrazów. Metoda nie rozstrzyga kwestii do czego pola Markowa mogą być wykorzystane. Jeżeli tylko wiadomo co ma być opisane matematycznie, co jest informacją wejściową oraz jakie są ograniczenia, to metoda ta wydaje się być niezastąpiona, szczególnie gdy badaczowi brakuje intuicji oraz głębszej znajomości teorii pól Markowa.

3.1 Metoda budowy modeli opartych na polach Markowa

W każdej sytuacji, w której przystępuje się do budowy jakiegoś modelu matematycznego trzeba sobie dokładnie zdawać sprawę z tego co chcemy zamodelować. W przypadku pól losowych na wstępie konieczne należy rozstrzygnąć dwie kwestie:

- Jak wygląda dziedzina budowanego pola ?
- Co jest zbiorem stanów tego pola ?

W obydwu przypadkach potencjalnie istnieje wiele możliwości. W algorytmach wizji komputerowej dziedzina pola losowego najczęściej, aczkolwiek nie zawsze, pokrywa się z dziedziną obrazu rozumianego jako funkcja jasności [Cha88, GG84]. Z matematycznego punktu widzenia jest to zazwyczaj pewien podzbiór kraty [DE87]. Ze względu na to, że obrazy komputerowe są ściśle związane ze sprzętem, dzięki któremu powstają, wygodniej jest przedstawić dziedzinę pola losowego w postaci tablicy o ustalonych rozmiarach, będących prawie zawsze potęgą liczby 2. Jeżeli modelowaniu poddawana jest sekwencja czasowa obrazów, wówczas dziedzina takiego pola będzie reprezentowana przez trójwymiarową tablicę, gdzie trzeci wymiar odpowiada zmiennej czasu.

Pytanie o zbiór stanów pola losowego można sformułować również w następujący sposób:

Rozkład jakiej lokalnej cechy jest przedmiotem modelowania za pomocą pola Markowa?

Jeżeli np. polem Markowa opisuje się funkcję jasności, to zbiór stanów takiego pola jest tożsamy ze zbiorem wartości tej funkcji jasności [Cha88]. Kiedy indziej polem Markowa może być opisane pole etykiet przyporządkowanych poszczególnym pikslom. Wtedy w najprostszym przypadku zbiór stanów takiego pola losowego jest zbiorem dwuelementowym. Można sobie również wyobrazić sytuacje, w których każdemu pikselowi pola losowego przyporządkowana jest wielowymiarowa zmienna losowa, co w konsekwencji powoduje, że elementy zbioru stanów pola losowego pochodzą z przestrzeni o odpowiednim wymiarze. Taki właśnie przypadek jest opisany w rozdziale 5.

Po udzieleniu odpowiedzi na powyższe dwa zasadnicze pytania należy z kolei rozstrzygnąć kwestie dodatkowe. Przede wszystkim konieczne jest wybranie *typu pola Markowa*. W rozdziale 2.2 omówiono trzy możliwości: pola jednowarstwowe jawne, pola hierarchiczne oraz pola ukryte. Właściwy wybór związany jest z rodzajem dostępnej informacji wejściowej (np. może być to sekwencja obrazów z gradacją szarości) oraz jej wpływem na modelowaną cechę. Często przy rozstrzyganiu tej kwestii przydaje się intuicja badacza. Z każdym typem modelu związane są pewne grupy zastosowań, ale o tym będzie mowa w rozdziale 3.2.

Wybranie modelu hierarchicznego albo ukrytego zwiększa liczbę pytań, na które trzeba udzielić odpowiedzi w czasie procesu modelowania.

Jeżeli zdecydowano się na model hierarchiczny, koniecznym staje się *wybranie warstwy lub warstw dolnych* oraz udzielenie odpowiedzi jaką rolę mają one spełniać. W ogromnej większości znanych z literatury zastosowań modeli hierarchicznych warstwą dolną jest

pole krawędzi międzypikselowych [GG84, Cha88, KS93]. Warstwa krawędzi „zabezpieczona” poszukiwaną, tzn. optymalną w pewnym sensie realizację pola Markowa przed przegłędzeniem [Bes86].

Wybranie modelu ukrytego narzuca konieczność określenia *informacji zewnętrznej* zwanej czasami *obserwacją* [Wrz93] oraz opisanie jej wpływu na rozkład modelowanej cechy, czyli na rozkład ukrytego pola Markowa.

Kolejną kwestią jest wybór *rzędu pola*. Ma on bardzo duży wpływ na złożoność danego pola Markowa, a tym samym na jego użyteczność w praktycznych zastosowaniach. Jak pokazano to w rozdziale 2.2 wraz ze wzrostem rzędu rośnie bardzo szybko liczba klik należących do systemu sąsiedztwa. W niektórych zastosowaniach [Cha88] wybrano najmniejszy z możliwych rzędów, tzn. jeden, jednak w ogromnej większości dominuje rząd dwa. Czasami, aby uprościć obliczenia, wyklucza się niektóre kliki przyjmując a priori, że ich energia jest równa zero [LB90]. Jeżeli zdecydowano się na model wielowarstwowy, wówczas należy wybrać rząd dla każdej z warstw, który nie koniecznie musi być jednakowy dla każdej z nich.

Następnym problemem jest *wyбір postaci energii pola* co, jak zostało pokazane w rozdziale 2.2, jest równoznaczne z określeniem postaci lokalnych warunkowych rozkładów prawdopodobieństwa – charakterystyk lokalnych. Sposób postępowania bardzo istotnie zależy od rodzaju wybranego modelu. Najłatwiej jest dla modeli jednowarstwowych jawnych, chociaż i tutaj istnieje kilka możliwości. Stosując podejście typowe dla pól Markowa zakłada się, że charakterystyka lokalna przyjmuje postać jednego ze znanych rozkładów prawdopodobieństwa [CJ83]. Wybierając podejście typowe dla rozkładu Gibbsa określa się energię klik, co determinuje rozkład charakterystyk lokalnych i oczywiście energię całkowitą [DE87]. W przypadku modeli hierarchicznych podobnych wyborów trzeba dokonać dla każdej z warstw. Dodatkowo jednak należy określić wpływ jednej warstwy na drugą, patrz rozdział 2.2. Dla modeli ukrytych koniecznym staje się uwzględnienie wpływu obserwacji. Częstym rozwiązaniem, podobnie jak dla modeli hierarchicznych, jest dodatkowy składnik energii.

Udzielenie odpowiedzi na powyższe pytania kończy w zasadzie koncepcyjną część budowy modelu matematycznego opartego na polach Markowa. W większości zastosowań, należy następnie dopasować model do posiadanych danych, czyli przeprowadzić *identyfikację modelu*. Czasami przeprowadzenie identyfikacji jest jedynym przeznaczeniem modelu [CJ83]. Niekiedy wartości parametrów wybiera się a priori, np. gdy model jest używany do generacji tekstur, lub gdy parametry odpowiadają tylko za spójność bądź niespójność przestrzenną [LB90]. W ogólnym przypadku identyfikacja jest procesem trudnym i często wymaga pokonania wielu dodatkowych zagadnień teoretycznych [KS93, Cha88].

Dotychczas najczęstszym zastosowaniem wielowymiarowej zmiennej losowej o rozkładzie pola Markowa, oczywiście poza identyfikacją tekstur, jest estymacja mody tego rozkładu. Poszukiwanie mody rozkładu losowego opisanego polem Markowa wymaga dokonania ostatniego wyboru w prezentowanej metodzie, tzn. zdecydowania się na algorytm estymacji tej mody. Ważniejsze możliwości omówiono w rozdziale 2.2.

Po tak szczegółowym omówieniu proponowanej metody, poniżej została ona przedstawiona w formie zwartej.

METODA BUDOWY MODELI OPARTYCH NA POLACH MARKOWA:

1. Wybierz dziedzinę pola Markowa.
2. Określ zbiór jego stanów.
3. Zdecyduj się na typ pola korzystając z następujących możliwości:
 - (a) jednowarstwowe jawne,
 - (b) hierarchiczne (wielowarstwowe),
 - (c) ukryte,
4. Jeśli wybrano model hierarchiczny, określ dolne warstwy.
5. Jeśli wybrano model ukryty, określ co jest obserwacją.
6. Wybierz rząd pola, wszystkich jego warstw w przypadku modeli hierarchicznych.
7. Określ wyrażenie energii pola losowego, tzn. wybierz energię wszystkich klik – podejście Gibbsa, lub postacie charakterystyk lokalnych – podejście Markowa.
8. W przypadku modeli hierarchicznych określ wzajemny wpływ poszczególnych warstw oraz wybierz rozkład pola, którego moda będzie poszukiwana, np. rozkład a posteriori.
9. W przypadku modeli ukrytych określ sposób wpływu obserwacji na pole ukryte, np. kryterium MAP lub MPM, co jest równoznaczne z określeniem całkowitej energii pola ukrytego.
10. Jeśli cel, dla którego budowany jest model tego wymaga, przeprowadź identyfikację pola, tzn. estymację jego parametrów.
11. Wybierz algorytm estymacji mody.

3.2 Zastosowania modeli opartych na polach Markowa

Załóżmy, że jest już zbudowany pewien model statystyczny. Pojawia się pytanie, jak można z takiego modelu skorzystać? Pytanie to jest całkiem nietrywialne i w zasadzie wykracza poza zakres komputerowej analizy obrazów. Z teorii modelowania wiadomo, że modele matematyczne można podzielić na trzy następujące grupy [Kur]:

- fizyczne,
- koncepcyjne,
- „czarnej skrzynki”.

Najogólniejszymi są oczywiście fizyczne, przy budowie których wykorzystuje się znajomość uniwersalnych praw fizycznych i relacji szczegółowych (np. związków fizycznych

czy konstytutywnych) obowiązujących dla danego zjawiska czy sygnału. Jeśli uda się zbudować taki model i nie będzie on zbyt skomplikowany, to jego wykorzystanie może być bardzo szerokie. Oczywiście oczekuje się, że model taki zachowuje swoją poprawność nie tylko dla danych na podstawie których był identyfikowany. Dzięki tej właściwości można na przykład przewidywać, tzn. predykować przyszłe stany modelowanego sygnału.

Modele konceptualne są bardziej uproszczone. Opierają się, w pewnym stopniu, na znajomości praw fizycznych rządzących modelowanym zjawiskiem, ale przy ich budowie dokonuje się wielu uproszczeń. Obszar ich poprawności jest najczęściej niewiele większy od obszaru danych, na podstawie których przeprowadzono identyfikację.

Najmniej ogólnymi są modele „czarnej skrzynki”. Z góry zakłada się ich postać, a dopiero na podstawie posiadanych danych stara się je dopasować do modelowanego zjawiska. Często takie modele są poprawne tylko dla danych, na podstawie których zostały zbudowane.

Do której z wymienionych grup można zaliczyć modele pól Markowa? Są one modelami statystycznymi wykorzystującymi pojęcie zmiennej losowej, a za tym przecież kryje się niewiedza o mechanizmie modelowanego zjawiska. Jednak mimo to, nie należy ich zaliczać do modeli typu „czarnej skrzynki”, gdyż w przeciwieństwie do pierwszych modeli statystycznych używanych w komputerowej analizie obrazów, w polach Markowa zawarta jest pewna wiedza o właściwościach sygnałów, jakimi są obrazy. Podstawowym założeniem, na którym oparta jest koncepcja pól Markowa, jest występowanie zależności pomiędzy próbkami informacji przyporządkowanymi sąsiednimi pikselom obrazu. Z tego powodu pola Markowa należy raczej zaliczyć do grupy modeli konceptualnych, co w dużym stopniu determinuje możliwości korzystania z nich.

Dzięki swoim właściwościom pola Markowa są używane bardzo często do restauracji obrazów [GG84, Cha88].

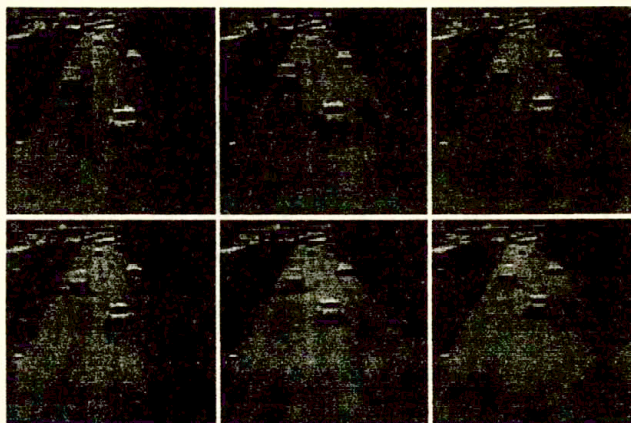
W niektórych zastosowaniach jest wykorzystywana zdolność pól Markowa do „przeglądania” modelowanej funkcji [KS93, CSC⁺93, CSC⁺, CSN⁺94]. Ma to miejsce na przykład w obrazach prążkowych powstałych jako efekt metod plamkowych stosowanych w analizie odkształceń. W przypadku tych obrazów pożądanym jest uciąglenie linii obserwowanych w zarejestrowanych obrazach, tak aby łatwiej można było zlokalizować maksima funkcji jasności. Przeglądanie modelowanej powierzchni polega na skłonności do tworzenia na nieskończonych obrazach nieskończenie wielkich obszarów o jednolitej jasności [Bes86]. Gdy w konkretnym zastosowaniu właściwość ta jest niepożądana, buduje się hierarchiczne modele pól Markowa z warstwą krawędzi, która ma uniemożliwić przeglądanie poprawianego obrazu [GG84, JW91]. Czasami też, w takim wypadku korzysta się z ukrytych pól Markowa.

Innymi ciekawymi zastosowaniami pól Markowa są: generacja oraz identyfikacja tekstur [CJ83], znajdowanie potoku optycznego [HB90a, HB90b], detekcja krawędzi [ZC90], detekcja masek obiektów ruchomych [LB90, BL90].

Rozdział 4

Detekcja zmian w obrazach – badania porównawcze

Metody detekcji masek obiektów ruchomych proponowane w następnych rozdziałach wymagają wcześniejszego znalezienia masek zmian w obrazach. Dla modeli opartych na polach Markowa omawianych w rozdziałach 5 oraz 10 maski zmian zachodzących pomiędzy kolejnymi obrazami sekwencji czasowej stanowią podstawę do znalezienia masek poruszającego się obiektu. Z tego powodu w niniejszym rozdziale zajęto się najpopularniejszymi, znanymi z literatury algorytmami wykrywania zmian czasowych zachodzących między kolejnymi obrazami sekwencji. Wybrane metody przetestowano przy użyciu kilku sekwencji obrazów przedstawiających poruszające się samochody. Wyniki przeprowadzonych eksperymentów oraz wnioski z nich wypływające zebrano na końcu tego rozdziału.



Rys. 4.1: Sekwencja A: ramki telewizyjne o numerach 1, 6, 11, 16, 21 oraz 26. Sekwencja otrzymana za pomocą zapisu filmowego na taśmie magnetycznej. Poruszający się obiekt jest przeciętnie jaśniejszy od tła.

4.1 Wybór metod detekcji zmian w obrazie

W wybranym w niniejszej rozprawie podejściu do detekcji ruchu, a przedstawionym w rozdziale 2.1.3, maski zmian w obrazie odgrywają bardzo ważną rolę. Na ich podstawie znajdowane są maski poruszającego się obiektu. W rozdziale 2.1.2 omówiono kilka znanych z literatury metod detekcji zmian w obrazach. Każdy z autorów przedstawiał swoją metodę jako niezawodną, ze szczególnym uwzględnieniem wybranych przez siebie przykładów. Nasuwa się jednak pytanie, który z algorytmów i dla jakich sekwencji obrazów prowadzi do najlepszych rezultatów. Ponieważ brak jest w dostępnej literaturze takiego porównania, dlatego zdecydowano się na przeprowadzenie odpowiednich badań eksperymentalnych. Porównano ze sobą:

1. *Test modułu różnicy jasności przyporządkowanych pikseli w dwóch kolejnych obrazach sekwencji* (MR). Jest to najprostsze podejście do problemu, odpowiadające wartości modułu pochodnej względem czasu w przypadku sygnałów ciągłych. Metoda MR jako jedyna nie uwzględnia wpływu jasności pikseli sąsiednich.
2. *Test Hsu-Nagela-Rekersa* (HNR) [HNR84] w dwóch wersjach:
 - (a) korzystającej z liniowego modelu funkcji jasności (HNRlin),
 - (b) korzystającej z kwadratowej aproksymacji funkcji jasności (HNRkw).
3. *Test Nieniewskiego-Pathaka* (NP) [NP91, NP90] wykorzystujący transformację Walsh'a funkcji jasności.

4. *Test Skifstada-Jaina* (SJ) [SJ89], który został utworzony na podstawie modelu zacienienia – wg jego autorów jedno z rozwiązań o zmniejszonej wrażliwości na zmiany oświetlenia.

4.2 Testowe sekwencje obrazów

Testowe sekwencje obrazów A, B, C przedstawiono odpowiednio na rys. 4.1, 4.12, 4.18. Wszystkie zostały utworzone w następujący sposób. Najpierw poruszające samochody zarejestrowano na taśmie video standardu Hi8 mm przy użyciu kamery umieszczonej na statywie. Następnie poszczególne klatki telewizyjne zdigitalizowano wykorzystując do tego celu standardowy przetwornik. Otrzymano sekwencje obrazów o rozdzielczości 512×512 pikseli i 256 poziomach szarości. Ze względu na to, że w przypadku europejskiego standardu telewizyjnego zapisywanych jest 25 obrazów na sekundę, do komputerowej analizy wybrano tylko niektóre kadry; co piąty w przypadku sekwencji A oraz co czwarty dla sekwencji B i C. Następnie w wyselekcjonowanych obrazach ustalono okno o rozmiarach 256×256 pikseli. Położenie okna było stałe dla całej sekwencji. Do analizy ruchu przeznaczono jedynie sekwencję takich spreparowanych klatek 256×256 pikseli. Wszystkie sekwencje składają się z sześciu obrazów, co stanowi wystarczającą długość dla oceny jakości testowanych metod. Obrazy sekwencji A, B, C umieszczono na rys. 4.1, 4.12, 4.18 w następującej kolejności:

1	2	3
4	5	6

Z punktu widzenia komputerowej analizy obrazów sekwencje różnią się między sobą relacją pomiędzy jasnością tła oraz poruszającego się obiektu. Sekwencja A przedstawia obiekt, który w zasadzie jest jaśniejszy od tła. Ponieważ obiektem tym jest samochód, to w rzeczywistości jasność poruszającego się obiektu jest zarówno jaśniejsza jak i ciemniejsza od tła, np. okna samochodu lub jego cień.

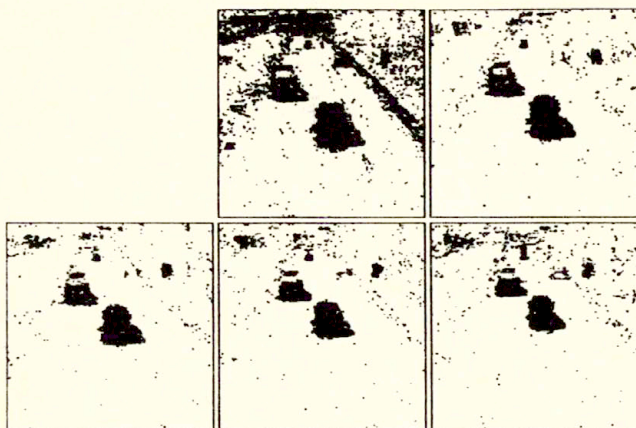
W przypadku sekwencji B występuje relacja odwrotna, tzn. jasność obiektu jest przeważnie ciemniejsza od tła. Również i tutaj można zaobserwować cień.

Sekwencja C jest podobna do A, z tym, że różnica jasności pomiędzy białymi częściami samochodu a tłem jest o wiele większa.

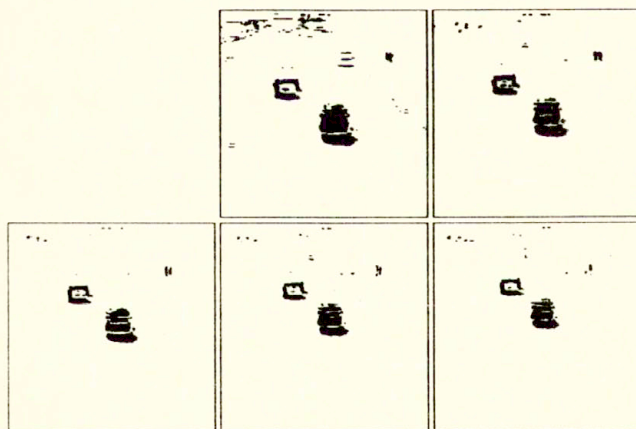
4.3 Wyniki przeprowadzonych eksperymentów

Przy porównywaniu ze sobą obrazów otrzymywanych przy użyciu różnych metod trudno jest posługiwać się obiektywnym kryterium oceny ich jakości. Zazwyczaj najważniejszą rolę odgrywa subiektywny odbiór obserwatora. Gdy system wizyjny zastosowany jest do konkretnego zadania jak np. paletyzacja detali, to najlepszym testem metody, również tej niskiego poziomu, jest niezawodność całego systemu wyrażona np. przy pomocy prawdopodobieństwa popełnienia błędu. W przypadku oceny jakości masek zmian można sformułować dwa kryteria:

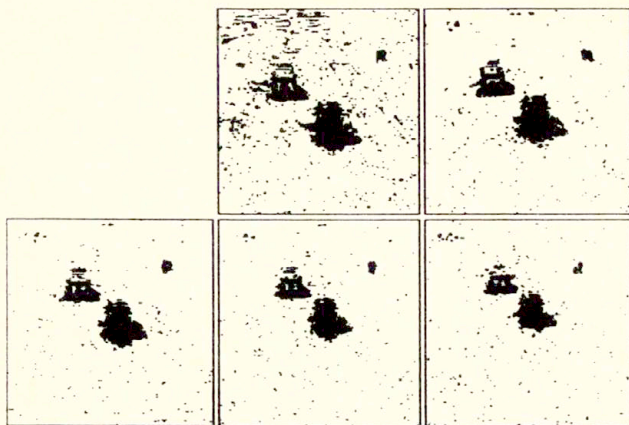
- spójność znalezionej maski,



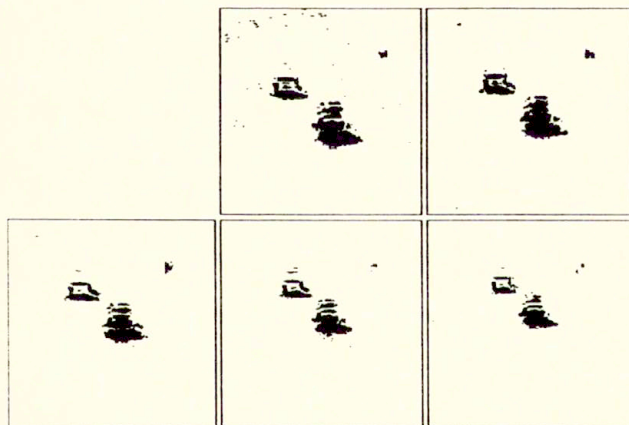
Rys. 4.2: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą MR, próg=10. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



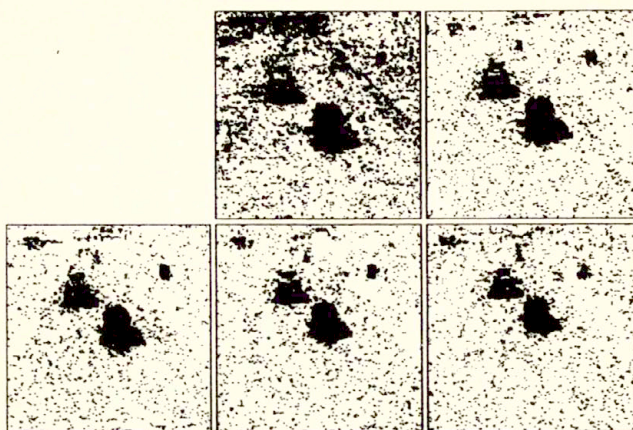
Rys. 4.3: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą MR, próg=40. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



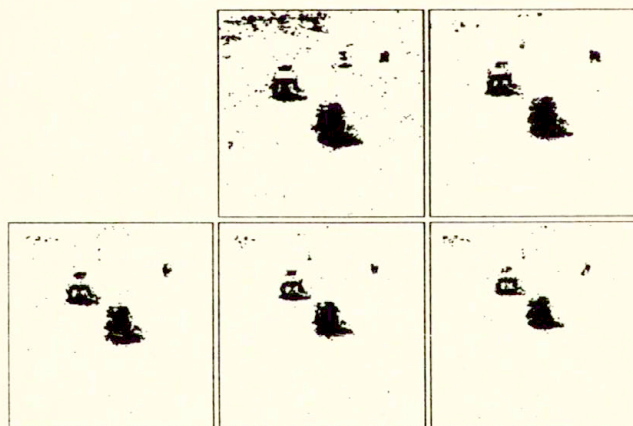
Rys. 4.4: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą HNRlin, próg=50. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



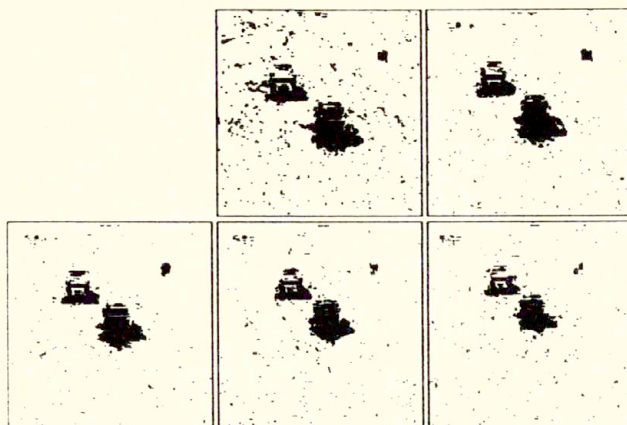
Rys. 4.5: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą HNRlin, próg=200. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



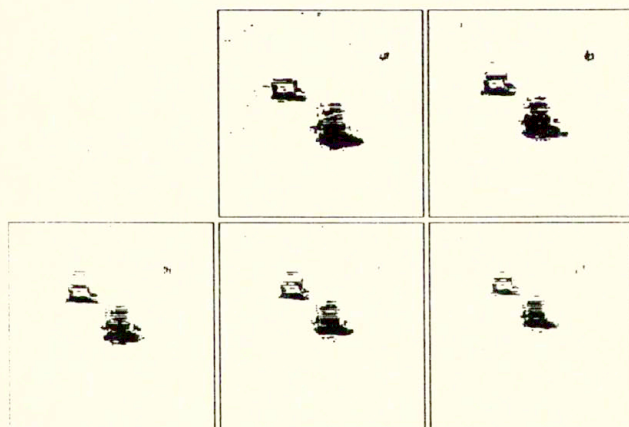
Rys. 4.6: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą HNRkw, próg=50. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



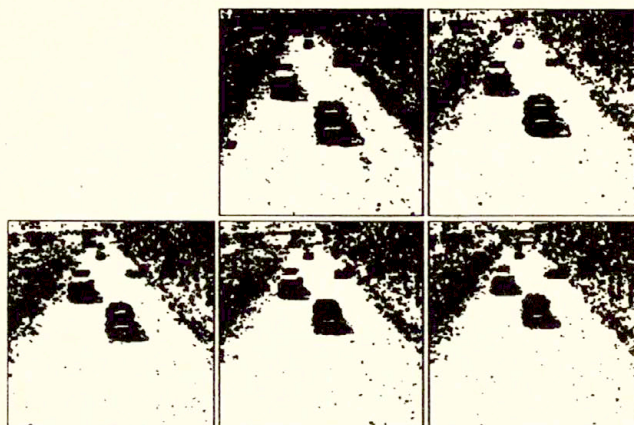
Rys. 4.7: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą HNRkw, próg=400. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



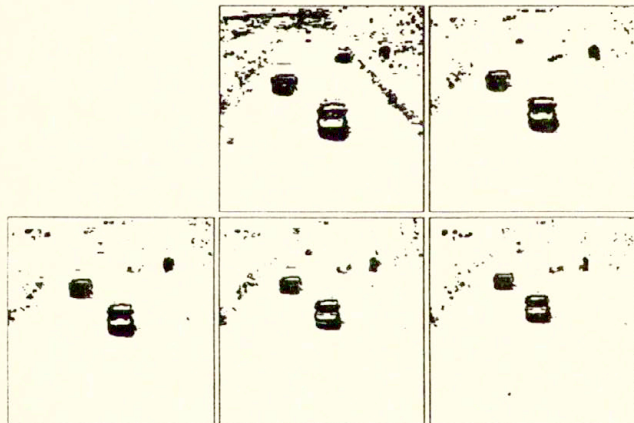
Rys. 4.8: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą NP, próg=25. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



Rys. 4.9: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą NP, próg=100. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



Rys. 4.10: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą SJ, próg=0.001. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



Rys. 4.11: Maski zmian w obrazie dla sekwencji A z rys. 4.1, otrzymane metodą SJ, próg=0.01. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.

- brak błędnie zaetykietowanych pikseli, tzn. tych którym przyporządkowano etykiety ruch zamiast tła lub na odwrót – błędnie zaetykietowanych pikseli obiektu.

Na takiej podstawie zostaną porównane wybrane metody.

Każda z sekwencji A, B, C została przebadana w podobny sposób. Okno testowe dla metod HNR w wersji liniowej jak i kwadratowej oraz dla testu SJ miało rozmiar 3×3 piksele, natomiast dla metody NP – 4×4 piksele. W przypadku algorytmów HNRlin, HNRkw, SJ, NP przy wyniku testu etykiety przypisywano tylko jednemu pikselowi okna, tzn. środkowemu dla HNRlin, HNRkw, SJ oraz jednemu z wewnętrznych pikseli okna testowego dla metody NP.

Wyniki przeprowadzonych eksperymentów przedstawiono w postaci sekwencji znalezionych masek zmian w obrazie. Sekwencji 6 obrazów z gradacją szarości odpowiada 5 masek zmian w obrazie. Maski te na wszystkich rysunkach zostały umieszczone w następującej kolejności:

$$\begin{array}{cc} \{1, 2\} & \{2, 3\} \\ \{3, 4\} & \{4, 5\} \quad \{5, 6\} \end{array}$$

przy czym np. $\{1, 2\}$ oznacza maskę zmian czasowych zachodzących pomiędzy pierwszym i drugim obrazem analizowanej sekwencji.

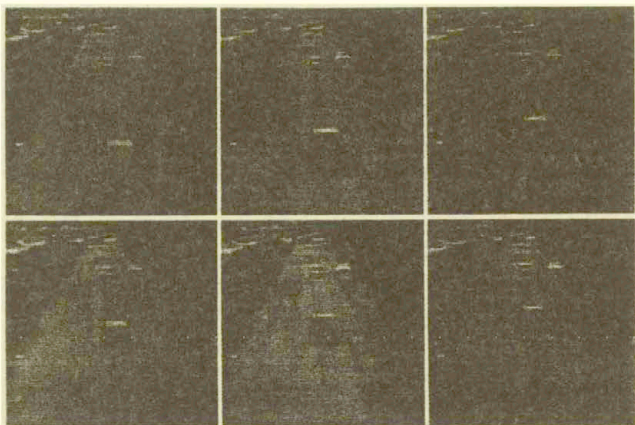
Badania wszystkich wybranych metod zostały przeprowadzone przy różnych wartościach progu. Na rysunkach przedstawiono tylko najważniejsze rezultaty. Szczegółowo wyniki eksperymentów zostaną omówione na przykładzie sekwencji A.

Na rys. 4.2-4.3 pokazano rezultaty zastosowania najprostszej i najszybszej metody, tzn. testu modułu różnicy jasności. Maski z rys. 4.2 to efekt małej wartości progu, w przeciwieństwie do tych z rys. 4.3 – gdzie próg był czterokrotnie większy. W obydwu przypadkach znalezione maski zmian w obrazie są dalekie od tego co widzi oko ludzkie. Zarówno maski z rys. 4.2 jak i 4.3 są niespójne, przy czym o wiele mniejsze braki daje się zauważyć dla masek wykrytych przy małym progu. Inaczej wygląda problem z drugim sformułowanym kryterium oceny jakości obrazów, tzn. z liczbą pikseli tła źle zaetykietowanych. Piksele takie występują zarówno w obrazach z rys. 4.2 jak i 4.3, z tym, że wraz ze wzrostem progu liczba tych pikseli wyraźnie maleje.

Na rys. 4.4, 4.5 przedstawiono maski zmian w obrazie otrzymane dzięki testowi HNR z liniowym, a na rys. 4.6, 4.7 z kwadratowym modelem funkcji jasności. Wnioski, które można na ich podstawie wyciągnąć są bardzo podobne jak w przypadku MR, tzn. wraz ze wzrostem progu, maski coraz bardziej tracą spójność, ale również maleje liczba źle zaetykietowanych pikseli tła. Ogólne wrażenie jest jednak nie najlepsze. Gdyby jako poziom odniesienia przyjąć tę samą liczbę pikseli tła, którym błędnie przypisano etykiety ruch, to maski najbardziej spójne otrzymywane są dzięki testowi HNRkw, w następnej kolejności jest HNRlin, a na końcu, czego należało się spodziewać, test modułu różnicy jasności. Zaobserwowane różnice nie są jednak bardzo istotne, a koszt obliczeniowy metod HNR jest o wiele większy.

Na rys. 4.8, 4.9 pokazano wyniki zastosowania testu NP. Porównując te maski z poprzednimi można zauważyć ich mniejszą spójność, ale jednocześnie metoda wykrywa mniejszą liczbę pikseli tła, którym przypisano ruch.

Rezultaty zastosowania metody Skifstada-Jaina pokazano na rys. 4.10, 4.11. W [SJ89] napisano, że metoda SJ wykrywa zmiany fizycznej struktury przedstawianych w obrazie



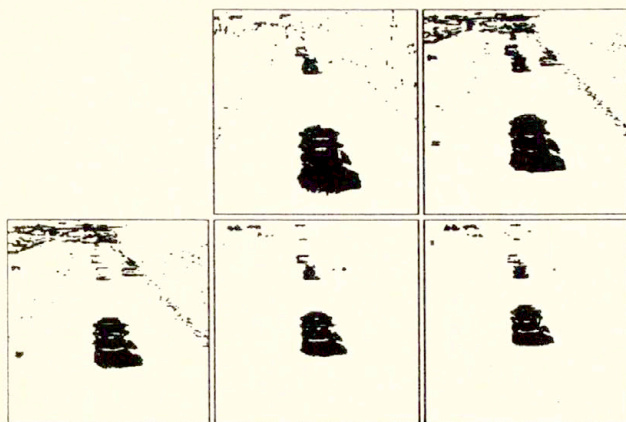
Rys. 4.12: Sekwencja B: ramki telewizyjne o numerach 1, 5, 9, 13, 17 oraz 21. Sekwencja otrzymana za pomocą zapisu filmowego na taśmie magnetycznej. Poruszający się obiekt jest przeciętnie ciemniejszy od tła.

powierzchni. W przypadku analizowanej sekwencji wydawać by się mogło, że cień samochodu nie powinien zmieniać fizycznej struktury powierzchni obrazu jezdni, czyli ruch cienia nie powinien być wykrywany. Okazuje się jednak, że tak nie jest. Cień za poruszającym się samochodem został wykryty zarówno przy małej jak i dużej wartości progu.

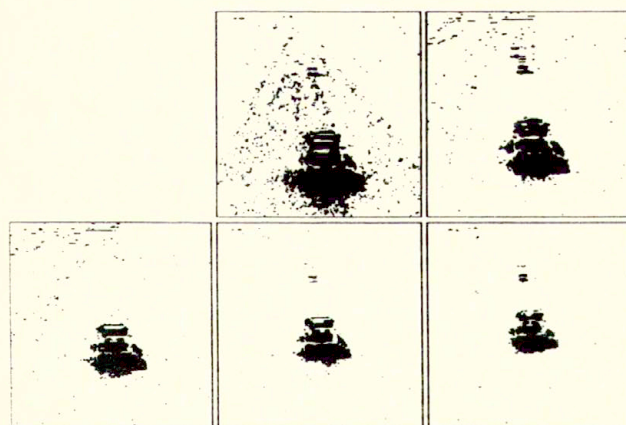
Mała wartość progu prowadzi jednocześnie do bardzo dużej liczby fałszywie zaetykietowanych pikseli tła, por. rys. 4.10. Jeżeli jednak treść sekwencji testowej byłaby inna, tzn. byłaby widoczna tylko jezdnia i jadące samochody, to przy małych wartościach progu metoda wydaje się dawać najlepsze maski zmian w obrazie.

Podobne serie eksperymentów przeprowadzono dla sekwencji B oraz C. Na rys. 4.13 – 4.17 pokazano przykładowe maski zmian w obrazie otrzymane dla sekwencji B, a na rys. 4.19–4.23 dla sekwencji C. Charakter otrzymanych masek jest w zasadzie taki sam jak dla sekwencji A. Zarówno dla poruszających się obiektów, które są ciemniejsze jak i wyraźnie jaśniejsze od tła, znalezione maski nie są spójne, a liczba źle zaetykietowanych pikseli tła zależy od przyjętej wartości progu.

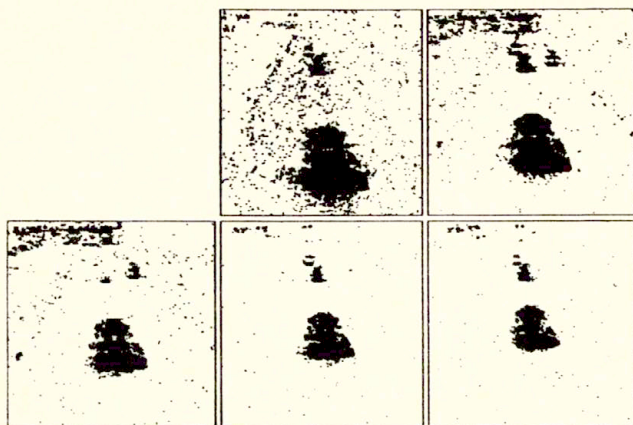
Przeglądając się uważnie przedstawionym maskom zmian w obrazie, szczególnie tym, które zostały znalezione dzięki testom HNR, można zauważyć, że liczba pikseli jezdni, którym przypisano błędnie etykietę ruch nie jest jednakowa dla całej sekwencji, por. rys. 4.4, 4.14, 4.15, 4.20. Liczba tych pikseli rośnie, gdy obiekt jest bliżej kamery. Najprawdopodobniej przyczyną takiego zjawiska są układy automatyki zastosowanej kamery telewizyjnej.



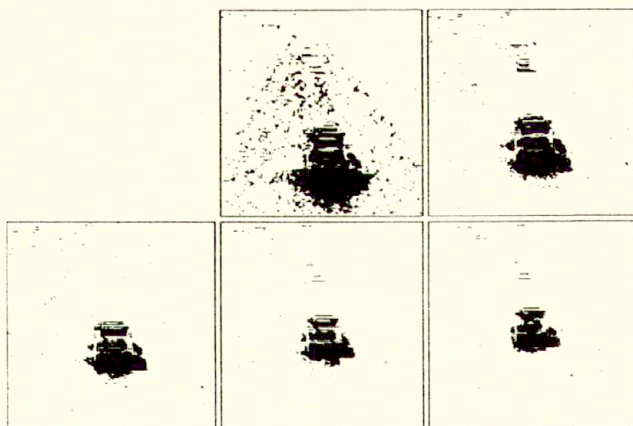
Rys. 4.13: Maski zmian w obrazie dla sekwencji B z rys. 4.12, otrzymane metodą MR, próg=20. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



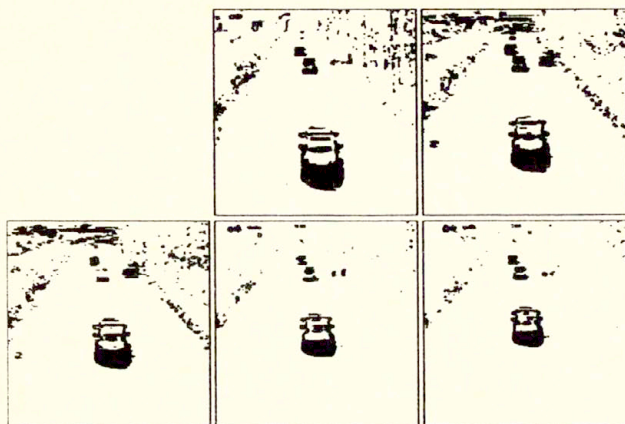
Rys. 4.14: Maski zmian w obrazie dla sekwencji B z rys. 4.12, otrzymane metodą HNRlin, próg=100. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



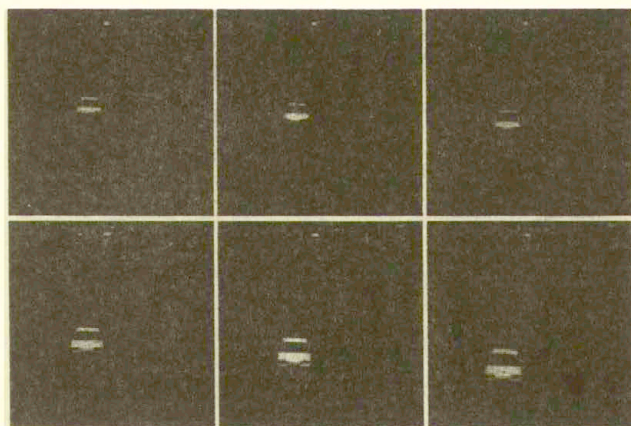
Rys. 4.15: Maski zmian w obrazie dla sekwencji B z rys. 4.12, otrzymane metodą HNRkw, próg=200. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



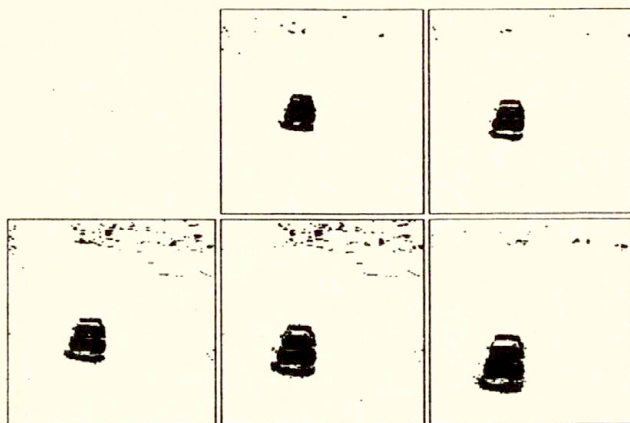
Rys. 4.16: Maski zmian w obrazie dla sekwencji B z rys. 4.12, otrzymane metodą NP, próg=50. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



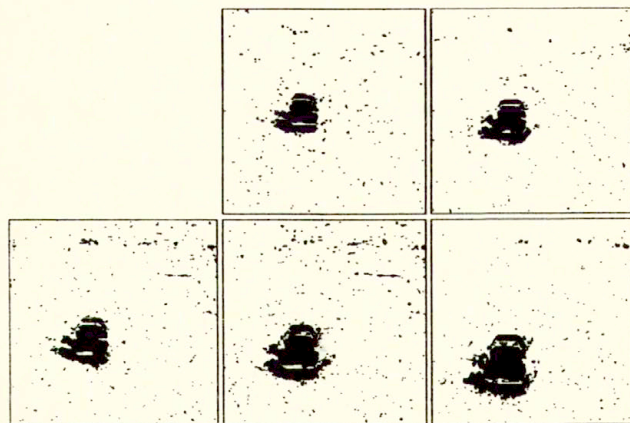
Rys. 4.17: Maski zmian w obrazie dla sekwencji B z rys. 4.12, otrzymane metodą SJ, próg=0.01. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



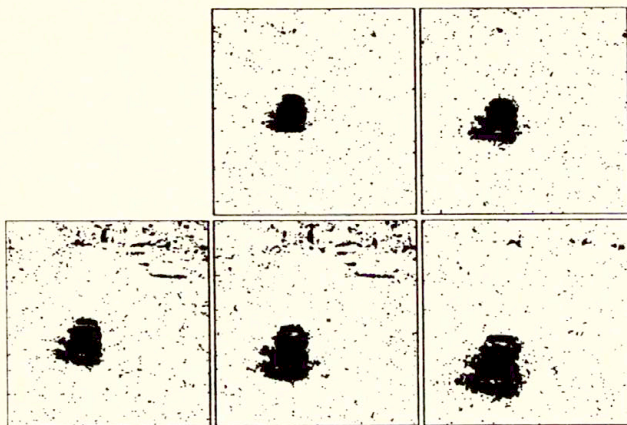
Rys. 4.18: Sekwencja C: ramki telewizyjne o numerach 1, 5, 9, 13, 17 oraz 21. Sekwencja otrzymana za pomocą zapisu filmowego na taśmie magnetycznej. Poruszający się obiekt jest przeciętnie znacznie jaśniejszy od tła.



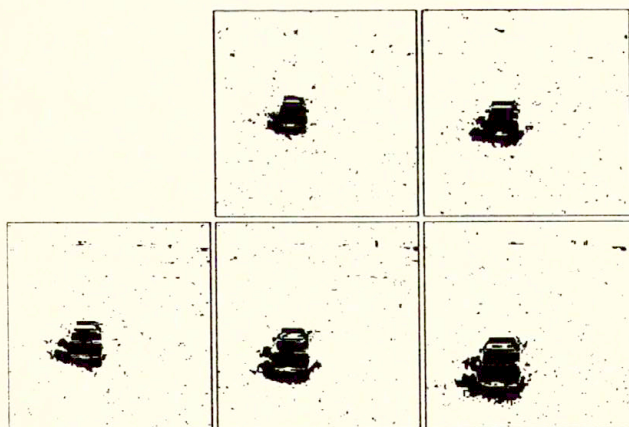
Rys. 4.19: Maski zmian w obrazie dla sekwencji C z rys. 4.18, otrzymane metodą MR, próg=20. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



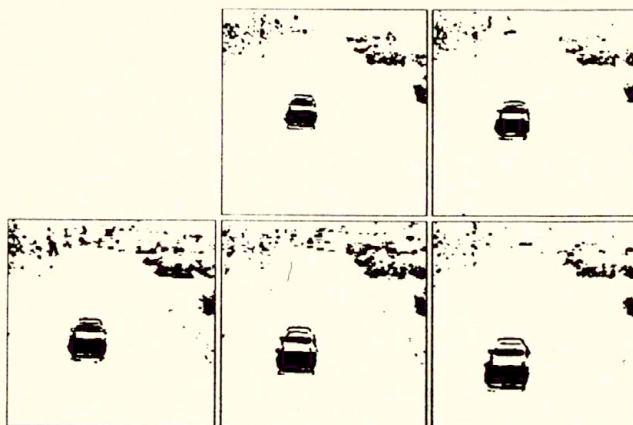
Rys. 4.20: Maski zmian w obrazie dla sekwencji C z rys. 4.18, otrzymane metodą HNRlin, próg=50. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



Rys. 4.21: Maski zmian w obrazie dla sekwencji C z rys. 4.18, otrzymane metodą HNRkw, próg=100. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



Rys. 4.22: Maski zmian w obrazie dla sekwencji C z rys. 4.18, otrzymane metodą NP, próg=25. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.



Rys. 4.23: Maski zmian w obrazie dla sekwencji C z rys. 4.18, otrzymane metodą SJ, próg=0.02. Kolorem czarnym zaznaczono piksele, w których została wykryta zmiana, a kolorem białym piksele, w których zmiany nie wykryto.

4.4 Wnioski

Najważniejsze wnioski wynikające z przeprowadzonych eksperymentów są następujące:

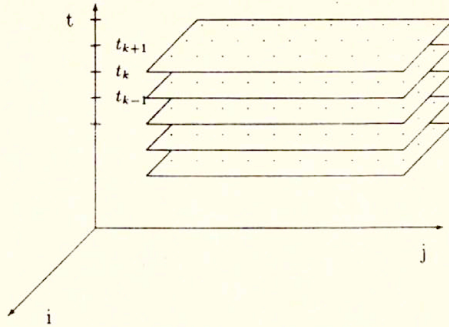
1. Wszystkie metody (łącznie z modulem różnicy jasności) prowadzą do bardzo podobnych rezultatów.
2. Żadna z metod nie znajduje idealnej lub zbliżonej do idealnej maski zmian w obrazie, tzn. znalezione maski są niespójne oraz występują źle zaetykietowane piksele. Obydwa kryteria jakości wykluczają się wzajemnie, tzn. poszukiwania pełniejszej maski, prowadzą do wykrywania jako ruchomych piksli tła.
3. Najbardziej spójne maski można znaleźć przy pomocy metod HNRkw, SJ oraz HNRlin.
4. Najmniejszą liczbę piksli tła zaetykietowanych jako ruch, przy porównywalnej spójności znalezionych masek zmian w obrazie, daje metoda NP.
5. Żadna z metod nie jest w stanie odseparować poruszającego się obiektu od cienia, przy czym zgodnie z oczekiwaniami, najlepsze rezultaty można otrzymać dzięki testowi SJ.
6. Niezależnie od wzajemnej relacji pomiędzy jasnościami tła i obiektu każda z metod daje podobne efekty.

Przeprowadzone eksperymenty nie pozwoliły jednoznacznie określić, która z testowanych metod jest najlepsza. Zaskakująco dobre rezultaty otrzymuje się dzięki metodzie modułu różnicy, która jest najprostsza i jednocześnie bardzo szybka. W zależności od treści sekwencji, a szczególnie od treści tła, niektóre metody są nieznacznie lepsze od pozostałych.

Rozdział 5

Zastosowanie pól Markowa w detekcji i śledzeniu ruchu

Rozdział jest poświęcony budowie modeli matematycznych opierających się na czasowo-przestrzennych polach losowych oraz wykorzystaniu tych modeli w detekcji i śledzeniu ruchu. Jako przykład pól losowych wybrano pola Markowa-Gibbsa. Zadanie postawione przed systemem komputerowej analizy obrazów polega na detekcji i śledzeniu ruchu obiektów pojawiających się w polu widzenia kamery, przedstawionych za pomocą sekwencji obrazów z gradacją szarości, tzn. na znalezieniu sekwencji masek poruszającego się obiektu, odpowiadających poszczególnym obrazom sekwencji wejściowej. Z kilku znanych podejść do tak postawionego problemu wybrano to, które nie wymaga wyznaczania potoku optycznego, a opiera się jedynie na sekwencji wejściowej oraz maskach zmian w obrazie, znalezionych za pomocą jednej ze znanych metod. W [BL90, LB90] Lalande i Bouthemy zastosowali ukryte pole Markowa-Gibbsa do detekcji masek obiektu ruchomego, w którym wykorzystali dwuwymiarowy wektor obserwacji, tzn. maskę zmian w obrazie oraz różnicę jasności przyporządkowanych pojedynczemu pikselowi w dwóch kolejnych obrazach. W rozdziale 5.2 przedstawiono, zgodnie z wymogami metody z rozdziału 3, model opracowany przez Lalande'a i Bouthemy'ego, który nazwano od pierwszych liter ich nazwisk modelem LB. W podrozdziale 5.3 zaproponowano do detekcji masek obiektu ruchomego nowy model będący czasowo-przestrzennym ukrytym polem Markowa-Gibbsa, który jest lepszy od modelu LB, bo dopuszcza wszystkie możliwości zaetykietowania pojedynczego piksela w dwóch kolejnych obrazach sekwencji. Nowy model korzysta z takiego samego, jak model LB, dwuwymiarowego wektora obserwacji. W podrozdziale 5.4 opisano implementacje algorytmów estymowania mody modelu LB oraz modelu nowego, tzn. relaksację deterministyczną oraz symulowane wyzarcanie. W podrozdziale 5.4.2 zaproponowano bardzo efektywną modyfikację algorytmu relaksacji deterministycznej, pozwalającą na skrócenie czasu estymowania mody rozkładu pola Markowa-Gibbsa nawet do dwudziestu razy.



Rys. 5.1: Nastęstwo obrazów.

5.1 Pola Markowa jako modele matematyczne wykorzystywane do detekcji masek obiektów ruchomych

Załóżmy, że mamy dostępną sekwencję obrazów przedstawiającą poruszający się obiekt. Zadanie polega na dokonaniu segmentacji każdego z nich, tak aby oddzielić od siebie ruchomy obiekt i statyczne tło. Można tego dokonać np. poprzez przyporządkowanie każdemu pikselowi jednej z dwóch etykiet, tzn. etykiety „ruch” albo etykiety „tło”. Ze względu na istnienie szumu, pożądanym wydaje się zastosowanie jakiegoś modelu statystycznego. Pola Markowa, z powodu uwzględniania zależności pomiędzy wartościami jasności sąsiednich pikseli, wydają się znakomicie nadawać do tego celu.

Dla uproszczenia zapisu w dalszej części rozprawy zostaną wprowadzone następujące oznaczenia:

$f_k(i, j)$ – wartość funkcji jasności k -tego obrazu sekwencji w pikselu (i, j) ,

L – zbiór etykiet, przyporządkowywanych pojedynczemu pikselowi,

E_k – pole losowe etykiet dla k -tego obrazu,

e_k – realizacja losowego pola etykiet E_k ,

$E_k(i, j)$ – rozkład losowy etykiet w pikselu (i, j) w obrazie k ,

$e_k(i, j)$ – etykieta przyporządkowana pikselowi (i, j) w obrazie k ,

$o_k(i, j)$ – różnica jasności $o_k(i, j) = f_{k+1}(i, j) - f_k(i, j)$,

$\bar{o}_k(i, j)$ – etykieta maski zmian zachodzących pomiędzy obrazami $k, k + 1$ w pikselu (i, j) .

Zagadnienie konstrukcji algorytmu korzystającego z pola Markowa, przeznaczonego do detekcji i śledzenia ruchu, zostanie szczegółowo omówione wg kolejnych punktów metody zaprezentowanej w rozdziale 3.

- Pierwszą kwestią, którą należy rozstrzygnąć w trakcie budowy pola Markowa jest jego **dziedzina**. Na rys. 5.1 schematycznie przedstawiono sekwencję obrazów. Oś t

jest osią czasu, kolejne obrazy indeksowane są liczbami naturalnymi. Osie i oraz j określają współrzędne poszczególnych pikseli należących do dziedziny obrazu. Przedstawiona przestrzeń jest przestrzenią dyskretną. Położenie poszczególnych pikseli może być określone za pomocą uporządkowanej trójki liczb naturalnych (i, j, k) . Oznaczmy dziedzinę pojedynczego obrazu przez S . Ponieważ wszystkie obrazy sekwencji mają jednakową dziedzinę, dziedzinę całej sekwencji można oznaczyć przez $S^K \subset Z^3$, przy czym: K – długość sekwencji. Dziedzina zmiennej losowej o rozkładzie pola Markowa nie musi być konieczne pewnym podzbiorem Z^3 . Możliwe jest wybranie podzbioru przestrzeni Z^2 pokrywającego się z dziedziną pojedynczego obrazu. W takim przypadku jasności we wszystkich pikselach o tych samych współrzędnych przestrzennych i różnych współrzędnych czasowych można modelować używając rozkładów wielowymiarowych zmiennych losowych. Wymiar zmiennej może być równy co najwyżej długości analizowanej sekwencji. W ogólnym przypadku istnieje wiele możliwości wybrania dziedziny. Jedną z nich jest sytuacja, w której dziedzinę stanowi S^K . Takie rozwiązanie, aczkolwiek koncepcyjnie poprawne, w praktyce często nie nadaje się do zastosowania z następujących powodów:

- bardzo długie sekwencje zużywałyby wręcz niewyobrażalną ilość pamięci komputera,
- niemożliwa staje się implementacja dostarczająca wyników on-line.

Sytuacją przeciwną do opisanej powyżej, ograniczającą rozmiar dziedziny do niezbędnego minimum, jest każdorazowe modelowanie polem Markowa tylko dwóch kolejnych obrazów sekwencji. W ten sposób znikają wymienione problemy, ale maleje prawdopodobieństwo znalezienia rozwiązania optymalnego.

- O wiele mniej kłopotów jest z dokonaniem wyboru zbioru stanów L pola Markowa. W rozważanej sytuacji zbiór L jest zbiorem dwuelementowym. Najczęściej przyjmuje się, że $L = \{a, b\}$, przy czym: etykieta a oznacza, że piksel należy do poruszającego się obiektu, a etykieta b – piksel zaliczony został do statycznego tła. Jeżeli pole Markowa zostanie skonstruowane tak, że w każdym pikselu należącym do jego dziedziny mamy do czynienia z rozkładami n -wymiarowymi (n – liczba modelowanych obrazów), to wówczas zbiorem stanów pola losowego będzie przestrzeń L^n , a możliwymi realizacjami w pikselu będą n -elementowe ciągi etykiet ze zbioru L .
- Kolejnym krokiem jest wybranie typu pola Markowa. Z trzech możliwości wymienionych w rozdziale 2.2 właściwie tylko modele ukryte nadają się do opisanie rozważanej sytuacji. Zasadniczym powodem jest to, że zmienną losową o rozkładzie pola Markowa opisujemy czasowo-przestrzenny rozkład etykiet, którego realizacje nie są bezpośrednio dostępne, a informację wejściową stanowią jedynie wartości funkcji jasności.
- Wybór modelu ukrytego determinuje konieczność określenia obserwacji wpływającej na rozkład ukrytego pola. Mając na uwadze detekcję ruchu, jako obserwację należałoby wybrać pewną lokalną miarę zachodzących zmian, jak np. różnicę wartości jasności w danym pikselu w kolejnych obrazach. Możliwe jest również wykorzystanie

wektorowej obserwacji, tzn. np. wspomnianej różnicy wartości jasności w pikslu oraz maski zmian w obrazie.

- Następnym krokiem jest określenie **rzędu pola Markowa**, czyli wybranie rodzajów klik, dla których funkcja energii może przyjmować wartości różne od zera. W większości przypadków przyjmowany jest rząd równy 2, który gwarantuje uwzględnienie wpływu wszystkich ośmiu najbliższych położonych sąsiadów danego piksla.
- Kolejnym etapem jest wybór wyrażenia analitycznego dla energii pola losowego, a tym samym postaci charakterystyk lokalnych, co jest równoznaczne z wybraniem funkcji określających energie wszystkich dopuszczalnych dla danego rzędu klik. Zastosowanie modelu ukrytego dodatkowo implikuje konieczność określenia **kryterium wpływu wybranej obserwacji na rozkład pola**, czyli podania całkowitej energii pola ukrytego. Jeżeli dla konkretnego pola Markowa mamy do czynienia w każdym pikslu z n -wymiarowymi rozkładami prawdopodobieństwa, to wówczas pojawia się konieczność określenia zależności pomiędzy rozkładami brzegowymi w pikslach o tych samych współrzędnych przestrzennych i różnych czasowych. Można to uczynić np. poprzez wybranie klik czasowych oraz postaci energii czasowej.
- Wybór postaci energii pola kończy etap budowy modelu matematycznego, jednak energia pola Markowa określona jest z dokładnością do pewnego wektora parametrów Θ . W takiej sytuacji pojawia się konieczność **identyfikacji** tego wektora, czyli estymacji współrzędnych wektora Θ , dla których zbudowany model najlepiej, względem pewnego kryterium, opisuje konkretną, analizowaną sekwencję obrazów. Właśnie ta czynność wydaje się być jedną z najtrudniejszych. Wartości niektórych parametrów można estymować na podstawie pewnego rozumowania heurystycznego. Niestety prawie zawsze identyfikacja parametrów wymaga znajomości wyniku detekcji ruchu, do której przecież chcemy ich użyć. Problem w pewnym stopniu rozwiązuje podejście iteracyjne, tzn. przeplatana estymacja parametrów i poszukiwanie minimum energii z poprawianymi za każdym razem estymatami parametrów [BPM93, ABP92, Pie92a, Pie92b]. Takie podejście do problemu jest zazwyczaj bardzo czasochłonne, a przez to wykorzystanie jego do detekcji ruchu jest bardzo utrudnione.
- Ostatnią kwestią pozostającą do rozstrzygnięcia jest wybór **algorytmu poszukiwania minimum energii**, co jest równoznaczne z estymowaniem mody zmiennej losowej o rozkładzie pola Markowa. Kilka możliwości zostało przedstawionych w rozdziale 2.2 oraz w [Car92]. Ze względu na czas obliczeń godnymi uwagi wydają się wszelkie algorytmy nadające się do implementacji równoległych, np. szeregowo-równoległe implementacje oparte na kodingach opisanych przez Besaga [CJ83, Bes74] lub sposób podobny do przedstawionego w [HMB91]. Szczególnie godnym polecenia jest algorytm relaksacji deterministycznej, np. podobny do wprowadzonego w [Bes86].

5.2 Model Lalande'a-Bouthemy'iego (LB)

Model matematyczny opracowany przez Bouthemy'iego i Lalande'a został opublikowany wraz z przykładowymi efektami jego użycia w [BL90, LB90]. Model ten zostanie obecnie przedstawiony z wykorzystaniem metody konstrukcji modelu matematycznego korzystającego z pól Markowa przedstawionej w rozdziale 3¹.

Dziedzina pola losowego

W modelu LB pole losowe zostało określone na przestrzeni 2D, tzn. dziedzina pola pokrywa się z dziedziną pojedynczego obrazu. Lalande i Bouthemy zamodelowali polem Markowa rozkład etykiet odpowiadających każdemu dwóm obrazom sekwencji. W efekcie wartościami pola losowego są dwuwymiarowe zmienne losowe, opisujące rozkład etykiet przyporządkowywanych pojedynczemu pikselowi w dwóch kolejnych obrazach sekwencji. Dokonane wybory spowodowały, że algorytm detekcji maski obiektu ruchomego składa się z wielu kroków. W każdym kroku poszukiwana jest optymalna realizacja pola etykiet odpowiadających jedynie dwóm kolejnym obrazom analizowanej sekwencji. Ze względu na wybór najmniejszej z możliwych dziedzin, detekcja maski obiektu ruchomego w każdym kroku nie może odbywać się bez uwzględnienia wyników z kroku poprzedniego. Z tego powodu Lalande i Bouthemy zdecydowali się na dwuetapowe poszukiwanie maski obiektu ruchomego dla każdego obrazu k sekwencji wejściowej. Pierwszy z tych etapów można nazwać *predykcją* a drugi *korekcją*. I tak, w kroku k poszukiwana jest najbardziej prawdopodobna realizacja pola etykiet jednocześnie dla obrazów k i $k+1$, tzn. dokonywana jest korekcja maski k -tej i predykcja maski $k+1$ -szej. Znalezione etykiety dla k -tego obrazu (efekt korekcji) są już realizacją ostateczną, natomiast etykiety dla obrazu o numerze $k+1$ (efekt predykcji) będą użyte do inicjacji korekcji maski $k+1$ -szej w $k+1$ -szym kroku algorytmu.

Zbiór stanów, typ oraz rząd pola Markowa

Dokonany wybór dziedziny pola spowodował, że zbiór jego stanów jest zbiorem cztero-elementowym $L^2 = \{(a, a), (a, b), (b, a), (b, b)\}$, przy czym a - etykieta przyporządkowana pikselowi należącemu do obiektu ruchomego, b - etykieta piksela tła. Elementami tego zbioru są uporządkowane pary etykiet a, b , przypisywanych pojedynczemu pikselowi w dwóch kolejnych chwilach czasowych. W konsekwencji łączny rozkład prawdopodobieństwa etykiet w pikselu zawsze będzie pewnym rozkładem czteropunktowym.

Z powodów opisanych w rozdziale 5.1 model LB jest ukrytym polem Markowa opisującym rozkład etykiet a, b .

Zgodnie z definicją kliki z rozdziału 2 para pikseli z dwóch różnych obrazów sekwencji, gdy pole Markowa jest określone na przestrzeni 2D a wartościami tego pola są dwuwymiarowe zmienne losowe, nie jest kliką. Jednak w celu skrócenia nazwa używanych pojęć, w dalszej części rozprawy będzie stosowane **rozszerzone pojęcie kliki**, tzn. zbiór pikseli należące do dwóch kolejnych obrazów sekwencji też będzie określane mianem kliki. Klika taka będzie nazywana *kliką czasową*. Dodatkowo w dalszej części rozprawy pojawi się

¹W porównaniu z symbolami wprowadzonymi w pracach Lalande'a i Bouthemy'iego część symboli zmieniła swoje znaczenie

jeszcze pojęcie *kliki przestrzennej*. Pod tą nazwą będzie rozumiany zbiór złożony z pikseli jednego obrazu. Dla przykładu w modelu LB, gdy polem Markowa będzie opisany rozkład etykiet przyporządkowywanych obrazom o numerach k i $k+1$, to w takim przypadku będziemy mieli do czynienia z klikami przestrzennymi obrazu k , z klikami przestrzennymi obrazu $k+1$ oraz z klikami czasowymi.

Przyjęto, że rząd pola jest równy 2, ale wykluczono wszystkie kliki jedno, trzy oraz czterepikselowe, poprzez przyjęcie dla nich zerowej wartości energii. Pozostawiono jedynie kliki dwupikselowe. Ponieważ w każdym pikselu mamy do czynienia z uporządkowaną parą etykiet, to dla modelu sąsiedztwa drugiego rzędu energia kliki złożonej z etykiety $e_k(i, j)$ i jednej z etykiet $e_{k+1}(i \pm 1, j \pm 1)$ może przyjmować w ogólnym przypadku wartość niezerową. Jednak Lalonde Bouthemy założyli, że tylko energia kliki $\{e_k(i, j), e_{k+1}(i, j)\}$ może być niezerowa. Innymi słowy przyjęto, że rozkład prawdopodobieństwa etykiety $e_k(i, j)$ zależy tylko od etykiet $e_k(i \pm 1, j \pm 1)$ oraz od etykiet $e_{k+1}(i, j)$ i $e_{k-1}(i, j)$, przy czym zależność od etykiety $e_{k-1}(i, j)$ jest uwzględniana podczas predykcji k -tej maski obiektu, a wpływ etykiety $e_{k+1}(i, j)$ jest brany pod uwagę w trakcie korekcji tej maski. Dzięki takim założeniom, w przypadku modelu LB, zależność pomiędzy brzegowymi rozkładami prawdopodobieństwa w pikselach o tych samych współrzędnych przestrzennych i różnych czasowych jest określona za pomocą energii przypisywanej klikom czasowym.

Obserwacja

Lalonde i Bouthemy wybrali dla swojego modelu obserwację będącą dwuwymiarowym wektorem, którego współrzędnymi są:

- maska zmian w obrazie $\bar{o}_k = \{\bar{o}_k(i, j)\}$,
- różnica jasności $o_k = \{o_k(i, j)\}$.

Chociaż autorzy tego wyrażnie nie napisali, taki dobór obserwacji dla modelu wykorzystywanego do detekcji ruchu ma głębokie uzasadnienie koncepcyjne. Gdyby obserwacją była sama różnica funkcji jasności, to etykiety ruchu mogłyby być przypisane izolowanym pikselom, w których wspomniana różnica osiągnęła wystarczająco dużą wartość, np. z powodu zwykłego szumu elektronicznego. Natomiast maska zmian otrzymana za pomocą takich metod jak metoda Skifstada [SJ89], czy testy zaprezentowane w [HNR84], przedstawia zmiany lokalne jakie zaszły pomiędzy dwoma kolejnymi obrazami, ale z uwzględnieniem wpływu jasności pikseli sąsiednich. Taka obserwacja w pewnym stopniu zmniejsza prawdopodobieństwo błędnego przyporządkowania etykiet ruchu.

Przyjęto, że wpływ wybranej obserwacji na rozkład ukrytego pola etykiet jest opisany za pomocą kryterium maksymalnego prawdopodobieństwa rozkładu a posteriori (MAP) oraz założono, że rozkład a posteriori pola etykiet $\{e_k, e_{k+1}\}$ jest rozkładem Gibbsa.

Postać energii

W przypadku modelu LB całkowita energia pola Markowa² jest sumą czterech składników:

$$W_k = W_{s_1} + W_{s_2} + W_t + W_z, \quad (5.1)$$

przy czym: W_k – całkowita energia pola etykiet dla obrazów k i $k+1$, W_{s_1} – energia przestrzenna pola etykiet e_k , W_{s_2} – energia przestrzenna pola etykiet e_{k+1} , W_t – energia czasowa pola etykiet $\{e_k, e_{k+1}\}$, W_z – energia zgodności pola $\{e_k, e_{k+1}\}$ z obserwacją o_k .

Lalande i Bouthemy założyli, że wpływ obserwacji o_k na rozkład pola ukrytego opisany jest za pomocą składnika energii W_z , a obserwacji o_k , za pomocą warunkowej postaci energii czasowej W_t , w której o_k spełnia rolę warunku dla tej energii.

Z postaci równania (5.1) wynika postać całkowitej energii lokalnej w pikselu (i, j) :

$$U_{ij} = U_{ij s_1} + U_{ij s_2} + U_{ijt} + U_{ijz}, \quad (5.2)$$

przy czym: $U_{ij s_1}$ – lokalna energia przestrzenna piksela (i, j) k -tej maski, $U_{ij s_2}$ – lokalna energia przestrzenna piksela (i, j) $k+1$ -szej maski, U_{ijt} – lokalna energia czasowa realizacji $(e_k(i, j), e_{k+1}(i, j))$, U_{ijz} – energia zgodności realizacji $(e_k(i, j), e_{k+1}(i, j))$ z obserwacją $o_k(i, j)$.

Każda z energii przestrzennych $U_{ij s_1}$, $U_{ij s_2}$ jest sumą ośmiu energii dwupikslowych klik przestrzennych, do których należy piksel (i, j) . Lalande i Bouthemy w swoim modelu przyjęli jednakową postać energii wszystkich czterech rodzajów dwupikslowych klik przestrzennych, a mianowicie:

$$V_{c_2 s} = \begin{cases} \beta_s & \text{gdy etykiety dla dwóch pikseli są różne,} \\ -\beta_s & \text{gdy etykiety dla dwóch pikseli są identyczne,} \end{cases} \quad (5.3)$$

przy czym: $c_2 s$ oznacza przestrzenną klikę dwupikslową, $V_{c_2 s}$ oznacza energię przestrzennej klikki dwupikslowej, β_s – stała większa od zera. Jak wiadomo, im energia klikki ma wartość bardziej ujemną, tym bardziej taka realizacja jest prawdopodobna (preferowana) w optymalnej realizacji całego pola. Im energia klikki przyjmuje wartość bardziej dodatnią, tym bardziej realizacja taka ma mniejsze szanse na wystąpienie. Wyrażenie opisujące każdą z energii $U_{ij s}$ można przedstawić w postaci:

$$U_{ij s} = \sum V_{c_2 s}(i, j). \quad (5.4)$$

przy czym: $V_{c_2 s}(i, j)$ to energia przestrzennej klikki dwupikslowej, do której należy piksel (i, j) . Niech Θ_s oznacza tę część wektora parametrów Θ , która odpowiada tylko za energię przestrzenną. Ze względu na istnienie w przyjętym modelu czterech rodzajów dwupikslowych klik przestrzennych, wektor parametrów Θ_s przyjmuje postać:

$$\Theta_s = [\beta_s, \beta_s, \beta_s, \beta_s]^T. \quad (5.5)$$

² Aby zachować zgodność z nazwami wprowadzonymi w rozdziale 2 powinno być napisane *całkowita energia ukrytego pola Markowa*, jednak w celu skrócenia nazw przymiotnik *ukryte* będzie niekiedy pomijany w dalszej części rozprawy, gdyż tylko ukryte pola Markowa będą dalej rozważane.

Energia czasowa $U_{ijt}[e_k(i, j), e_{k+1}(i, j) | \bar{o}_k(i, j)]$ ma postać energii warunkowej, przy czym warunkiem jest etykieta $\bar{o}_k(i, j)$ maski zmian w obrazie. Każdemu pikselowi dziedziny pola odpowiada dwuetykietowa klika czasowa. Energia takiej kliki czasowej przyjmuje wartości według tabelki:

$$U_{ijt}[e_k(i, j), e_{k+1}(i, j) | \bar{o}_k(i, j)] = \begin{cases} \begin{array}{|c|c|c|} \hline & \bar{o}_k = 0 & \bar{o}_k = 1 \\ \hline (b, b) & -\beta_t & \beta_t \\ \hline (a, b) & \beta'_t & \beta'_t \\ \hline (b, a) & \beta_t & -\beta_t \\ \hline (a, a) & \beta_t & -\beta_t \\ \hline \end{array} , & (5.6) \end{cases}$$

przy czym: β_t oraz β'_t – stałe większe od zera. $\bar{o}_k = 0$ oznacza, że zmian w pikselu (i, j) pomiędzy obrazami k oraz $k+1$ nie wykryto, a $\bar{o}_k = 1$ odpowiada wykryciu takich zmian. Oznaczając przez Θ_t tę część wektora parametrów, która opisuje energię czasową, wektor Θ_t przyjmuje postać:

$$\Theta_t = [-\beta_t, \beta_t, \beta'_t, \beta'_t, \beta_t, -\beta_t, \beta_t, -\beta_t]^T. \quad (5.7)$$

Do określenia energii zgodności U_{ijz} została zdefiniowana pomocnicza funkcja o postaci³:

$$\psi[e_k(i, j), e_{k+1}(i, j)] = \begin{cases} m_0 & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (b, b) \\ m_1 & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (a, b) \text{ lub } (b, a) \\ m_2 & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (a, a). \end{cases} \quad (5.8)$$

Założono, że rozkład prawdopodobieństwa zgodności realizacji $(e_k(i, j), e_{k+1}(i, j))$ z obserwacją $o_k(i, j)$ jest rozkładem normalnym z wartością średnią m_0, m_1 albo m_2 oraz wariancją σ^2 . W takim przypadku lokalna energia zgodności opisana jest równaniem:

$$U_{ijz} = \frac{1}{2\sigma^2} \{o_k(i, j) - \psi[e_k(i, j), e_{k+1}(i, j)]\}^2. \quad (5.9)$$

Oznaczmy przez Θ_z tę część wektora Θ , która opisuje energię zgodności. Wektor Θ_z przyjmuje postać:

$$\Theta_z = [m_0, m_1, m_2, \sigma^2]^T. \quad (5.10)$$

Estymacja parametrów modelu

Wektor parametrów Θ ma postać:

$$\Theta = [\Theta_s, \Theta_t, \Theta_z]^T. \quad (5.11)$$

Łącznie model LB jest określony z dokładnością do 16 parametrów. Skróty opis identyfikacji modelu LB zamieszczony w pracach [BL90, LB90] sugeruje, że Lalande i Bouthemy

³W stosunku do oryginalnych oznaczeń Lalande'a i Bouthemy'iego symbole m_1 i m_2 zostały ze sobą zamienione

dobrali wartości większości parametrów swojego modelu w sposób heurystyczny. Dotyczy to parametrów energii przestrzennej i czasowej oraz parametru m_0 energii zgodności. I tak przyjęli: $\beta_s = 10$, $\beta_t = 100$, $\beta'_t = 1000$, $m_0 = 0$. Konsekwencją dokonanych wyborów jest ograniczenie do trzech liczby parametrów odpowiadających za dopasowanie modelu LB do analizowanej sekwencji obrazów. O estymacji parametru σ^2 Lalande i Bouthemy napisali tylko, że należy ją przeprowadzić raz na początku przetwarzania sekwencji obrazów. W przypadku parametrów m_1 i m_2 zaproponowali skorzystanie z metod lokalnej estymacji, bądź zasugerowali oszacowanie wartości tych parametrów tylko raz, na początku analizy sekwencji wejściowej.

W [BL90, LB90] nie napisano na temat doboru wartości tych parametrów nic więcej, nawet nie podano wartości parametrów m_1 , m_2 , σ^2 , które posłużyły do znalezienia prezentowanych tam sekwencji masek. Dobór odpowiednich wartości wymienionych parametrów wcale nie jest zagadnieniem oczywistym, gdyż wymaga wiedzy a priori o analizowanej sekwencji. Szerzej ten problem zostanie poruszony podczas omawiania wyników badań eksperymentalnych.

Algorytm poszukiwania minimum energii pola Markowa

Lalande i Bouthemy w swojej implementacji wykorzystali do tego celu algorytm relaksacji deterministycznej, który wymaga poprawnej inicjacji. Ze względu na skrótowy charakter prac [BL90, LB90] nie przedstawiono żadnych informacji ułatwiających implementację. Skorzystanie z modelu LB wymaga odtworzenia nieopisanych przez autorów szczegółów. Praca taka została wykonana, a jej rezultaty zamieszczono w rozdziale 5.4.

5.3 Nowy model z dwuwymiarowym wektorem obserwacji

Przedstawiony w rozdziale 5.2 model matematyczny, wydaje się bardzo dobrze nadawać do znajdowania masek obiektu ruchomego. Lalande i Bouthemy wybrali najmniejszą z możliwych dziedzin, ułatwiając w ten sposób implementację działającą w trybie zbliżonym do on-line. Aby uwzględnić wpływ zarówno przeszłości jak i przyszłości na poszukiwane rozwiązanie, wprowadzono dwuetapowe poszukiwanie maski obiektu ruchomego dla k -tego obrazu, tzn. predykcję i korekcję. Można jednak zadać pytanie:

Czy model Lalande'a-Bouthemy'ego jest modelem optymalnym, w którym nie można nic poprawić?

Po głębszej analizie daje się zauważyć pewne niedociągnięcia procesu modelowania. Zastanawiającą jest bardzo duża wartość energii czasowej przypisana realizacji (a, b) odpowiadającej sytuacji (ruch, tło)⁴, wykluczająca ją praktycznie w sposób deterministyczny. Skądinąd wiadomo, że algorytmy detekcji zmian w obrazie najlepiej wykrywają sytuacje

⁴Podobnie jak w rozdziale 2 jeśli pojawia się w tekście określenie np. sytuacja (ruch, tło), to oznacza to tylko sytuację teoretyczną, tzn. informacja tego typu jest niedostępna w trakcie zastosowania opisywanego modelu do analizy ruchu przy wykorzystaniu rzeczywistych sekwencji obrazów.

przejściowie tzn. (tło, ruch) lub (ruch, tło), a to by sugerowało zmianę wartości energii czasowej realizacji $(a, b \mid \bar{o}_k = 1)$ na ujemną.

Zastanówmy się jednak przez chwilę nad konsekwencjami wynikającymi z takiego przyjęcia energii dla realizacji (a, b) . Najpierw jednak założymy, że maska zmian \bar{o}_k jest idealna, tzn. rozróżnione są w niej prawidłowo sytuacje (ruch, tło), (tło, ruch), (ruch, ruch) od sytuacji (tło, tło). W k -tym kroku algorytmu, tzn. kiedy przeprowadzana jest korekcja maski k -tej jednocześnie z predykcją maski $k+1$ -szej, jeśli detekcja zmian w obrazach poprzedzająca detekcję masek obiektu ruchomego byłaby przeprowadzona bezbłędnie, to wtedy do maski obiektu odpowiadającej k -temu obrazowi powinny zostać zaliczone piksele, w których zmiany spowodowały sytuacje (ruch, tło) oraz (ruch, ruch), a do $k+1$ -szej maski obiektu, w których zmiany spowodowały sytuacje (ruch, ruch) oraz (tło, ruch). W rzeczywistości w czasie detekcji ruchu nie wiadomo, która z sytuacji: (ruch, tło), (tło, ruch) czy (ruch, ruch) spowodowała zmiany, więc w trakcie etykietowania mogą być popełniane błędy. Błędnie zaetykietowane piksele podczas predykcji powinny zostać „poprawione” podczas korekcji. Praktycznie deterministyczne wykluczenie realizacji (a, b) jest przyczyną błędów popełnianych podczas predykcji. Spróbujmy to wyjaśnić bardziej wyczerpująco.

Jeśli na podstawie maski zmian \bar{o}_k wiadomo, że w pikselu (i, j) zaszły zmiany, to podczas detekcji ruchu para etykiet $(e_k(i, j), e_{k+1}(i, j))$ powinna zostać wybrana z realizacji (a, b) , (b, a) , (a, a) . Jednak realizacja (a, b) jest zabroniona ze względu na dużą wartość β_t , więc pozostają tylko realizacje (b, a) oraz (a, a) . Wybranie realizacji (b, a) , gdy w rzeczywistości mamy do czynienia z sytuacją (ruch, tło), jest niewskazane, gdyż w ten sposób w czasie korekcji maski k -tej przypisywane byłyby etykiety b pikselom obiektu ruchomego. Ponieważ błędy popełniane byłyby w trakcie korekcji, to nie byłoby już ich kiedy poprawić. Oznacza to, że pozostaje tylko realizacja (a, a) , czyli pikselom należącym do tła w obrazie $k+1$ -szym błędnie przypisywane są etykiety a , w konsekwencji czego maska $k+1$ obiektu znaleziona w czasie predykcji jest co najmniej zgodna z maską zmian. Jeżeli w konkretnej sekwencji sytuacja (ruch, tło) jest liczna, to konsekwencją praktycznie deterministycznego wykluczenia realizacji (a, b) może być dość znaczne wydłużenie czasu obliczeń, gdyż fałszywe etykiety a przypisane w czasie predykcji muszą być następnie usunięte w trakcie korekcji.

Co jest przyczyną takiego a nie innego przyjęcia energii klik czasowych przez Lalanda'a i Bouthemy'iego. Odpowiedź można znaleźć analizując postać funkcji (5.8). Autorzy tworząc tę funkcję dokonali uproszczenia w etapie modelowania, łącząc ze sobą realizacje (a, b) i (b, a) . Chociaż taki model jest prostszy, jednak w sposób niedokładny modeluje rzeczywistość.

Przedstawiony tok rozumowania prowadzi do wniosku, że zmiana wyrażenia opisującego energię w modelu LB, tak aby rozróżnić realizacje (a, b) i (b, a) , jest w pełni uzasadniona. Z tego powodu w niniejszej rozprawie proponowany jest nowy model z dwuwymiarowym wektorem obserwacji, który charakteryzuje się taką właśnie cechą.

W proponowanym modelu większość założeń jest identyczna jak w modelu LB. Dla obydwu modeli identyczne są: dziedzina, zbiór stanów, typ, rząd, wyrażenie opisujące energię przestrzenną oraz obserwacja wpływająca na rozkład ukrytego pola losowego. Identyczne są również wartości przypisane parametrom β_s , β_t oraz m_0 . Zmianie ulegają:

1. wyrażenie opisujące energię zgodności, oraz

2. tablica określająca warunkową energię klik czasowych.

Z tego powodu w niniejszym rozdziale nie zostanie powtórzony cały tok rozumowania wymagany w trakcie konstrukcji modelu korzystającego z pól Markowa, a omówione będą tylko te kroki, w których zaproponowano zmiany.

Pierwsza zmiana dotyczy funkcji (5.8). Funkcja ta ma teraz postać:

$$\psi[e_k(i, j), e_{k+1}(i, j)] = \begin{cases} m_0 & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (b, b) \\ m_1 & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (a, b) \\ m_2 & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (b, a) \\ m_3 & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (a, a). \end{cases} \quad (5.12)$$

Dawny parametr m_2 stał się teraz parametrem m_3 . Modyfikacja powoduje powiększenie wymiaru przestrzeni, na której określony jest wektor Θ_z z 4 do 5 tzn.:

$$\Theta_z = [m_0, m_1, m_2, m_3, \sigma^2]^T. \quad (5.13)$$

Bez zmian pozostaje założenie, że rozkład prawdopodobieństwa zgodności realizacji pary etykiet $(e_k(i, j), e_{k+1}(i, j))$ z obserwacją $o_k(i, j)$ jest w każdym pikselu rozkładem normalnym, ale przyjmuje się, że możliwymi wartościami średnimi są m_0, m_1, m_2 albo m_3 , a wariancją jest, tak jak poprzednio, σ^2 .

Drugą modyfikacją, która wynika bezpośrednio z tej pierwszej, jest konieczność zmian energii przypisywanych realizacji (a, b) , a przedstawionych w tablicy opisującej energię klik czasowych. Proponuje się zlikwidowanie parametru β_t^i i zastąpienie go parametrem β_t z odpowiednim znakiem. Ponieważ realizacja (a, b) , odpowiadająca sytuacji (ruch, tło), jest jedną z lepiej przedstawianych w masce zmian w obrazie, więc proponuje się przypisanie realizacji (a, b) takiej energii, która preferowałaby tę realizację w przypadkach uzasadnionych. Zmodyfikowana tabela określająca energię klik czasowych ma postać:

$$U_{i,j,t}[e_k(i, j), e_{k+1}(i, j) | \bar{o}_k(i, j)] = \begin{cases} \begin{array}{|c|cc|} \hline & \bar{o}_k = 0 & \bar{o}_k = 1 \\ \hline (b, b) & -\beta_t & \beta_t \\ (a, b) & \beta_t & -\beta_t \\ \hline (b, a) & \beta_t & -\beta_t \\ (a, a) & \beta_t & -\beta_t \\ \hline \end{array} \\ \end{cases} \quad (5.14)$$

przy czym: β_t stała większa od zera. Wektor Θ_t przybiera postać:

$$\Theta_t = [-\beta_t, \beta_t, \beta_t, -\beta_t, \beta_t, -\beta_t, \beta_t, -\beta_t]^T. \quad (5.15)$$

Wprowadzone modyfikacje spowodowały, że za dopasowanie proponowanego modelu do analizowanej sekwencji obrazów odpowiadają cztery parametry: m_1, m_2, m_3 oraz σ^2 .

Jako algorytm poszukiwania mody pola Markowa, ze względu na czas obliczeń, proponuje się algorytm relaksacji deterministycznej.

5.4 Algorytmy estymacji mody pola Markowa-Gibbsa przeznaczonego do detekcji masek obiektów ruchomych

5.4.1 Relaksacja deterministyczna

Zarówno w przypadku modelu LB jak i nowego modelu zaproponowanego w rozdziale 5.3, estymacja mody pola Markowa, opisującego rozkład etykiet a, b dla obrazów o numerach k oraz $k + 1$, polega na znalezieniu minimum całkowitej energii ukrytego pola Markowa-Gibbsa W_k , określonej równaniem (5.1). Jedną z możliwości znalezienia realizacji $\{e_k, e_{k+1}\}$ zapewniającej minimum energii W_k jest zastosowanie algorytmu relaksacji deterministycznej. Krok k -ty takiego algorytmu przedstawiono poniżej.

1. *Znajdź maski inicjujące, tzn. maskę e_k^0 inicjującą korekcję maski k oraz maskę e_{k+1}^0 inicjującą predykcję maski $k + 1$ -szej. Wyzeruj licznik iteracji ($it = 0$).*
2. *Rozpocznij nową iterację ($it = it + 1$). Wyzeruj licznik pikseli ($pk = 0$).*
3. *Wybierz kolejny piksel ($pk = pk + 1$) należący do dziedziny pola losowego. Załóżmy, że jest to piksel (i, j) . Z czterech możliwych realizacji (a, a) , (a, b) , (b, a) , (b, b) charakterystyki lokalnej w pikselu (i, j) wybierz tę, która minimalizuje wartość energii U_{ij} określoną równaniem (5.2).*
4. *Jeśli piksel (i, j) nie był ostatnim ($pk \neq M$, przy czym M – liczba pikseli należących do dziedziny pola losowego) skocz do 3.*
5. *Jeśli zakończona iteracja nie była ostatnią założoną ($it \neq it_{max}$) oraz w zakończonej iteracji (o numerze it) dokonano jakiegokolwiek zmiany etykiet w stosunku do iteracji poprzedniej (o numerze $it - 1$) skocz do 2.*
6. **KONIEC**

Zbieżność algorytmu do lokalnego minimum gwarantuje addytywny charakter całkowitej energii pola ukrytego W_k , gdyż dzięki temu minimalizacja energii U_{ij} nigdy nie powoduje wzrostu energii W_k .

W przedstawionym algorytmie jako kryterium stopu zastosowano alternatywę dwóch warunków, tzn. osiągnięcie minimum lokalnego energii oraz wykonanie zadanej z góry liczby iteracji. Drugi warunek jest zabezpieczeniem przed niestabiłością np. spowodowaną złym doбором wartości parametrów stosowanego modelu.

Algorytm zapewnia znalezienie minimum energii W_k , ale może to być oczywiście minimum lokalne. Z tego powodu wymagana jest inicjacja obydwu poszukiwanych masek. Maską e_k^0 , inicjującą korekcję maski k -tej, może być maska otrzymana po predykcji w kroku poprzednim, tzn. $k - 1$. Wyjątek stanowi maska odpowiadająca pierwszemu obrazowi analizowanej sekwencji, ponieważ w jej przypadku przeprowadzana jest jedynie korekcja, a tym samym maska e_k^0 musi być dostarczona z zewnątrz. Maską inicjującą predykcję e_{k+1}^0 może zostać znaleziona przy użyciu maski e_k^0 , dostępnych obserwacji o_k oraz \bar{o}_k . W każdym pikselu (i, j) etykieta $e_{k+1}^0(i, j)$, musi minimalizować ograniczoną energię

lokalną o postaci:

$$U_{ij}^0 = U_{ijz} + U_{ijt}, \quad (5.16)$$

przy czym U_{ij}^0 – ograniczona energia lokalna. Ograniczona energia lokalna jest całkowitą energią lokalną pola U_{ij} , w której zrezygnowano ze składników przestrzennych.

Przedstawiony algorytm zapewnia znalezienie minimum energii W_k poprzez maksymalizację prawdopodobieństwa charakterystyk lokalnych. Z tego powodu można go nazwać algorytmem ICM. Należy jednak pamiętać, że różni się on od oryginalnego algorytmu ICM wprowadzony w pracy [Bes86]. W algorytmie z pracy [Bes86] w każdym pikselu pola losowego przeprowadza się maksymalizację rozkładu warunkowego postaci⁵:

$$P(X_i = x_i | \mathbf{y}, \mathbf{x}_{S_i}), \quad (5.17)$$

przy czym: \mathbf{y} – obserwacja, X_i – wartość pola Markowa w pikselu i , \mathbf{x}_{S_i} – realizacja pola \mathbf{X} bez i -tego piksela.

Stosując tę samą notację algorytm deterministycznej relaksacji przedstawiony w niniejszym rozdziale opiera się na maksymalizacji prawdopodobieństwa warunkowego w postaci:

$$P(X_i = x_i | y_i, x_{\partial i}), \quad (5.18)$$

przy czym: $X_i = (e_k(i), e_{k+1}(i))^T$, $\mathbf{y}_i = (\bar{o}_k(i), o_k(i))^T$, $x_{\partial i}$ – wszyscy przestrzenni sąsiedzi piksela i .

Jeżeli w dalszej części rozprawy pojawi się nazwa ICM, to pod tą nazwą będzie rozumiany algorytm relaksacji deterministycznej przedstawiony w niniejszym rozdziale.

5.4.2 Modyfikacja przyspieszająca algorytm relaksacji deterministycznej

Algorytm relaksacji deterministycznej pozwala na względnie szybkie osiągnięcie lokalnego minimum energii. Najczęściej liczba iteracji nie przekracza 30. Jeżeli jednak sekwencja składa się dużej liczby obrazów, to nawet liczba 30 iteracji potrzebna do znalezienia masek obiektu dla każdego dwóch obrazów sekwencji, może w sumie prowadzić do dość długich czasów obliczeń. Przeprowadzone eksperymenty dla sekwencji składających się z sześciu obrazów o rozmiarach 256×256 pikseli i 256 poziomach szarości przy użyciu algorytmu relaksacji deterministycznej (maksymalna założona liczba iteracji była równa 25) wymagały około 40 minut do przeprowadzenia detekcji masek obiektu ruchomego (komputer klasy IBM AT 486). Zastanówmy się czy nie można jakoś tego przyspieszyć?

Jeżeli pole Markowa jest dobrze „dopasowane” do danych, to algorytm relaksacji deterministycznej jest w stanie znaleźć jego realizację o minimalnej energii. Zanim jednak minimum zostanie osiągnięte, w każdej iteracji w każdym pikselu trzeba czterokrotnie (dla każdej realizacji (a, a) , (a, b) , (b, a) oraz (b, b)) obliczyć wartość energii U_{ij} . Za każdym razem wykonywanych jest kilkanaście sumowań oraz conajmniej po jednym dzieleniu i mnożeniu (dzielenie i mnożenie wymagane jest do obliczenia lokalnej energii zgodności

⁵W przedstawionym wzorze zastosowano oznaczenia z pracy [Bes86]. Między innymi dla uproszczenia notacji przyjęto, że piksele obrazu ustawiono w pewien ciąg, dzięki czemu pojawia się tylko jeden indeks.

U_{ijz}). Ponieważ algorytm relaksacji deterministycznej jest algorytmem deterministycznym, więc zmiana etykiet w pikselu aktualnym dokonywana jest tylko na podstawie realizacji etykiet w pikselach sąsiednich. Jeżeli lokalnie dla pewnej grupy pikseli osiągnięto rozkład optymalny, to w trakcie kolejnych iteracji przyporządkowane etykiet nie będą już zmieniane. Aby skrócić czas obliczeń należy unikać wykonywania niepotrzebnych operacji. Można tego dokonać wprowadzając dwuwymiarową tablicę znaczników binarnych, która rozmiarem odpowiada dziedzinie pola losowego, a jak wiadomo w przypadku rozważanych modeli dziedzina jest podzbiorem przestrzeni 2D. W każdej iteracji, w każdym pikselu należy uaktualniać zawartość tablicy znaczników, zaznaczając fakt dokonania bądź niedokonania zmian etykiet. W każdej iteracji, poza pierwszą, można skorzystać z takiej tablicy w następujący sposób: na początku każdego kroku algorytmu relaksacji deterministycznej należy najpierw sprawdzić, czy któremukolwiek sąsiadowi zmieniono ostatnio etykiety, tzn. czy ustawiony jest którykolwiek ze znaczników. Jeśli odpowiedź jest negatywna, to można przejść do następnego kroku algorytmu bez wykonywania zbędnych obliczeń. W przeciwnym przypadku trzeba wykonać wszystkie wymagane obliczenia.

Jak pokazały przeprowadzone eksperymenty taka prosta modyfikacja polegająca na wprowadzeniu tablicy znaczników pozwala skrócić czas obliczeń nawet do 20 razy. W przypadku tej samej sekwencji co dla algorytmu relaksacji deterministycznej bez modyfikacji, czas obliczeń po wprowadzeniu modyfikacji wynosił poniżej dwóch minut, przy wykorzystaniu tego samego komputera.

5.4.3 Symulowane wyżarzanie

W przypadku implementacji algorytmu symulowanego wyżarzania trzeba rozwiązać wstępnie kilka problemów, a mianowicie:

- wybór funkcji temperatury,
- sposób znajdowania charakterystyki lokalnej,
- sposób losowania par etykiet będących realizacjami charakterystyki lokalnej.

Dobór funkcji temperatury. Optymalna funkcja zmian temperatury zaproponowana w [GG84] nie nadaje się do implementacji komputerowych. Jednak w tej samej pracy zaproponowano funkcję suboptymalną o postaci:

$$T(n) = \frac{C}{\ln(1+n)} \quad (5.19)$$

przy czym: n - nr iteracji, $n = 1, \dots, n_{max}$, C - pewna stała większa od zera. W pracy [GG84] zaproponowano: $C = 3$ i $n_{max} = 300$. Taka funkcja zmiany temperatury była odpowiednia, ale do modelu zaproponowanego w tej pracy.

Jednym z założeń, na którym opiera się symulowane wyżarzanie, jest przyjęcie jednakowych prawdopodobieństw każdej realizacji etykiet w pierwszej iteracji, tzn. temperatura $T(1)$ musi być odpowiednio wysoka. Należy więc tak dobrać wartość stałej C , aby ten warunek w pierwszej iteracji był spełniony. Temperatura w pierwszej iteracji wynosi:

$$T(1) = \frac{C}{\ln 2} \quad (5.20)$$

Dla uproszczenia opisu oznaczmy realizacje etykiet (a, a) , (a, b) , (b, a) , (b, b) odpowiednio przez e_1 , e_2 , e_3 , e_4 . Przeanalizujemy wyrażenie opisujące rozkład prawdopodobieństwa charakterystyki lokalnej modelu LB oraz modelu nowego na przykładzie kombinacji e_1 . W takiej sytuacji mamy:

$$P[(E_k(i, j), E_{k+1}(i, j)) = e_1 | e_{\partial(i, j)}] = \frac{\exp\left[-\frac{U_{ij}(e_1)}{T}\right]}{\exp\left[-\frac{U_{ij}(e_1)}{T}\right] + \exp\left[-\frac{U_{ij}(e_2)}{T}\right] + \exp\left[-\frac{U_{ij}(e_3)}{T}\right] + \exp\left[-\frac{U_{ij}(e_4)}{T}\right]} \quad (5.21)$$

Dzieląc licznik i mianownik prawej strony równania (5.21) przez $\exp\left[-\frac{U_{ij}(e_1)}{T}\right]$ otrzymujemy:

$$P[(E_k(i, j), E_{k+1}(i, j)) = e_1 | e_{\partial(i, j)}] = \frac{1}{1 + \exp\left[\frac{\Delta U_2}{T}\right] + \exp\left[\frac{\Delta U_3}{T}\right] + \exp\left[\frac{\Delta U_4}{T}\right]} \quad (5.22)$$

Gdyby każdy z wykładników funkcji wykładniczych mianownika był w przybliżeniu równy 0, to prawdopodobieństwo wystąpienia kombinacji etykiet e_1 byłoby w przybliżeniu równe $\frac{1}{4}$. Powtarzając takie rozumowanie dla następnych kombinacji etykiet zauważymy, że prawdopodobieństwo wystąpienia każdej z kombinacji opisane jest podobnym wyrażeniem. Bazując na postaci tych wzorów można określić algorytm doboru wartości stałej C dla konkretnego pola Markowa. Algorytm ten przedstawiony jest poniżej.

1. Oblicz maksymalną i minimalną wartość całkowitej energii lokalnej, bez uwzględniania wpływu temperatury, tzn. przyjmując $T = 1$, jaka może wystąpić dla rozważanego pola, dla dowolnej realizacji etykiet w aktualnym pikselu i jego sąsiedztwie. $\Delta U = ?$.
2. Dobierz taką temperaturę $T(1)$ aby: $T(1) = 0.1 | \Delta U |$.
3. Oblicz stałą C ze wzoru: $C = T(1) \ln 2$.

W przedstawionym algorytmie występuje próg równy 0.1. Oczywiście temperaturę $T(1)$ należało by tak dobrać aby w równaniu:

$$P[(E_k(i, j), E_{k+1}(i, j)) = e_1 | e_{\partial(i, j)}] = \frac{1}{1 + \exp\left[\frac{\Delta U_2}{T(1)}\right] + \exp\left[\frac{\Delta U_3}{T(1)}\right] + \exp\left[\frac{\Delta U_4}{T(1)}\right]} \quad (5.23)$$

wykładnik każdej funkcji wykładniczej był równy zero. W takim przypadku prawdopodobieństwo pojawienia się kombinacji e_1 wynosiłoby dokładnie $\frac{1}{4}$, ale wtedy temperatura $T(1)$ powinna dążyć do nieskończoności.

Charakterystyka lokalna. W każdym pikselu może wystąpić jedna z czterech realizacji etykiet (a, a) , (a, b) , (b, a) lub (b, b) . Ustalenie rozkładu charakterystyki lokalnej trzeba przeprowadzać w każdym pikselu oddzielnie ze względu na jego sąsiedztwo. Korzystając czterokrotnie z równania (5.2), obliczamy prawdopodobieństwa wystąpienia każdej z realizacji. Oznaczmy te prawdopodobieństwa przez p_1 , p_2 , p_3 oraz p_4 .

Algorytm losowania pary etykiet z charakterystyki lokalnej. Przyjmijmy, że uporządkowaliśmy możliwe realizacje. Losowanie pary etykiet z charakterystyki lokalnej można przeprowadzić następująco:

1. Oblicz prawdopodobieństwa p_1, p_2, p_3, p_4 wystąpienia poszczególnych realizacji w danym pikselu (korzystając ze wzoru (5.2)).
2. Wylosuj liczbę l z rozkładu równomiernego na przedziale $[0, 1]$.
3. Jeśli $l > p_1$ skocz do 5.
4. Wylosowano realizację nr 1. STOP.
5. Jeśli $l > p_2$ skocz do 7.
6. Wylosowano realizację nr 2. STOP.
7. Jeśli $l > p_3$ skocz do 9.
8. Wylosowano realizację nr 3. STOP.
9. Wylosowano realizację nr 4. STOP.

Poniżej przedstawiono tę część algorytmu symulowanego wyżarzania, która dotyczy pojedynczego piksla.

1. Weź nowy piksel.
2. Wylosuj nową realizację pary etykiet $(e_k(i, j), e_{k+1}(i, j))$ w aktualnym pikselu według algorytmu przedstawionego powyżej. Niech e_{ij} oznacza aktualną kombinację etykiet, natomiast e'_{ij} oznacza kombinację nowo wylosowaną.
3. Jeśli w całkowitej charakterystyce lokalnej e'_{ij} ma większe prawdopodobieństwo wystąpienia (mniejszą całkowitą energię lokalną) niż e_{ij} to zaakceptuj nową kombinację i skocz do 1.
4. Oblicz wartość liczby q będącej stosunkiem prawdopodobieństwa nowej kombinacji do starej.
5. Wylosuj liczbę l z rozkładu równomiernego na $[0, 1]$.
6. Jeśli $q \geq l$ zaakceptuj nową kombinację.
7. Skocz do 1

Rozdział 6

Kryterium oceny dokładności masek obiektów ruchomych

W niniejszym rozdziale wprowadzono dwie statystyki określone na znajdujących masek obiektu ruchomego. Wartości tych statystyk mogą posłużyć do obiektywnej oceny dokładności przeprowadzonej detekcji. Statystyki te zostały wykorzystane w następnych rozdziałach do porównania efektów otrzymywanych przy użyciu modelu LB oraz nowego modelu z dwuwymiarowym wektorem obserwacji.

6.1 Ocena dokładności masek obiektów ruchomych

W przypadku komputerowych metod przetwarzania obrazów występuje problem oceny dokładności znajdujących efektów. Jeżeli oceniane są obrazy otrzymywane przy użyciu tylko jednej metody, to można zastosować *kryterium subiektywnej percepcji*, tzn. klasyfikację znalezionych obrazów do grup: bardzo dobry, dobry lub zły. Wykorzystanie subiektywnej percepcji do porównania ze sobą obrazów otrzymywanych za pomocą różnych metod, przeznaczonych do tego samego celu, jest praktycznie niemożliwe. Zasadniczym tego powodem jest to, że kryterium subiektywnej percepcji w dużym stopniu zależy od opinii oceniających ludzi, a te mogą się przecież bardzo różnić między sobą. Z taką właśnie sytuacją mamy do czynienia w przypadku masek obiektu ruchomego znajdujących za pomocą modelu LB oraz nowego modelu z dwuwymiarowym wektorem obserwacji. Z tego powodu pojawiła się konieczność opracowania nowego, obiektywnego kryterium oceny dokładności znajdujących masek obiektu ruchomego.

Dobrym testem do oceny dokładności znalezionych masek obiektu mogą być wartości dwóch statystyk określonych na tych maskach, tzn.

1. statystyka $\neg a$ będąca liczbą pikseli faktycznie należących do tła, którym błędnie przypisano etykietę a ,
2. statystyka $\neg b$, czyli liczba rzeczywistych pikseli obiektu błędnie zakwalifikowanych jako tło.

Aby ocenić efekty znajdujące za pomocą różnych metod detekcji ruchu wystarczy zastosować te metody dla tej samej testowej sekwencji obrazów, a następnie obliczyć dla każdej znalezionej maski wartości obydwu statystyk. Pozostaje oczywiście problem wyboru testowej sekwencji obrazów. Nie może być to sekwencja naturalna, gdyż brak wiedzy o dokładnym położeniu obiektu uniemożliwiłby obliczenie statystyk $\neg a$ oraz $\neg b$. Raczej powinna być to sekwencja utworzona w sposób sztuczny, w przypadku której znane jest dokładne położenie obiektu w kolejnych obrazach. Jednocześnie nie może być to sekwencja zbyt „łatwa” z punktu widzenia analizy ruchu, tzn. taka, dla której można przeprowadzić detekcję maski poruszającego się obiektu poprzez zastosowanie np. progowania histogramów [Ś191].

W rozdziale 6.2 przedstawiono cztery testowe sekwencje obrazów przeznaczone do oceny dokładności masek obiektu ruchomego znajdujących przy użyciu modelu LB oraz nowego modelu z dwuwymiarowym wektorem obserwacji.

6.2 Testowe sekwencje obrazów

Aby ocenić maski obiektu ruchomego znajdujące przy użyciu różnych metod stworzono cztery sztuczne sekwencje obrazów o rozdzielczości 256×256 pikseli i 256 odcieniach szarości, przedstawiające poruszający się kwadrat o rozmiarach 80×80 pikseli. W przypadku każdej z sekwencji przyjęto, że jasność tła jest stała, a jasność obiektu zmienia się liniowo w pewien wybrany dla każdej sekwencji sposób. Sekwencje zostały zsumowane addytywnym białym szumem o wartości średniej 0 i wariancji 225. Stworzone sekwencje oznaczono K,

L, M, N. Każda z nich składa się z sześciu obrazów. Sekwencje pokazane są odpowiednio na rysunkach 6.1, 6.3, 6.5, 6.7.

Przyjęcie liniowej zamiast stałowartościowej funkcji jasności poruszającego się obiektu miało na celu uniknięcie przeprowadzania testów za pomocą tzw. „trudnych”, ze względu na analizę ruch, sekwencji obrazów, tzn. takich dla których lokalna miara zmian w przypadku sytuacji (tło, tło) oraz (ruch, ruch) ma identyczne wartości. Detekcja masek obiektów ruchomych w przypadku sekwencji „trudnych” zostanie omówiona w rozdziale 10.

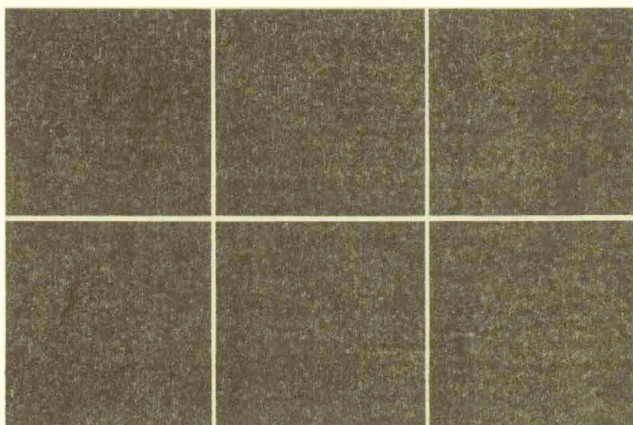
W każdej z testowych sekwencji K, L, M, N obiekt porusza się w kierunku prawego dolnego narożnika. Położenie kwadratu w odpowiadających sobie obrazach w przypadku wszystkich sekwencji jest identyczne. Sekwencje przedstawiają szybki jednostajny ruch, co oznacza, że względne zmiany położenia w kolejnych obrazach są duże. W szczególności wektor przemieszczenia ma stałą wartość – 40 pikseli w poziomie i 30 pikseli w pionie. W wszystkich sekwencjach przed zaszumieniem jasność obiektu zmienia się w sposób liniowy, przy czym zakres zmian dla sekwencji K, L, M jest identyczny i wynosi 20 poziomów szarości, tzn. mniej niż 10 procent całego możliwego zakresu funkcji jasności. Dla sekwencji N zakres zmian jasności obiektu wynosi aż 40 poziomów szarości. Wzajemne relacje pomiędzy jasnością tła oraz obiektu dla sekwencji K, L, M, N zostały pokazane odpowiednio na rys. 6.2, 6.4, 6.6, 6.8.

Sekwencje K oraz L różnią się między sobą relacją pomiędzy jasnością obiektu i tła. W sekwencji K poruszający się obiekt jest zawsze jaśniejszy od tła (przed dodaniem szumu), przy czym różnica pomiędzy jasnością tła i minimalną jasnością obiektu wynosi 20 poziomów szarości, tzn. tyle samo co zakres zmian funkcji jasności obiektu. W przypadku sekwencji L obiekt jest zawsze ciemniejszy od tła, ale różnica pomiędzy jasnością tła i maksymalną jasnością obiektu ponownie wynosi 20 poziomów szarości. Sekwencja M jest właściwie pewną modyfikacją sekwencji K. W obydwu sekwencjach obiekt przed zaszumieniem jest jaśniejszy od tła, z tym że w przypadku sekwencji K obiekt porusza się częścią jaśniejszą do przodu, przeciwnie niż w przypadku sekwencji M. Sekwencja N przedstawia ruch obiektu, który jest zarówno jaśniejszy jak i ciemniejszy od tła.

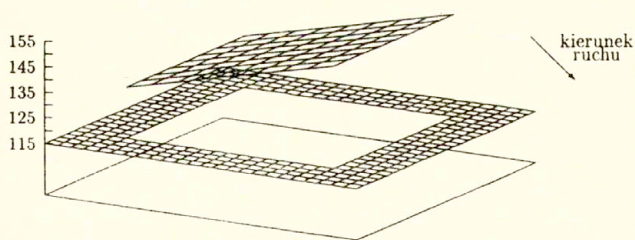
6.3 Estymacja parametrów modelu LB oraz nowego modelu z dwuwymiarowym wektorem obserwacji dla sekwencji testowych

Celowość doboru sekwencji testowych przedstawionych w rozdziale 6.2 można wyjaśnić przeprowadzając estymację parametrów m_1 , m_2 dla modelu LB oraz m_1 – m_3 w przypadku nowego modelu z dwuwymiarowym wektorem obserwacji. Ponieważ dostępna jest pełna wiedza o każdej z testowych sekwencji, możliwym więc staje się estymowanie parametrów metodą uśrednienia obserwacji o_k dla każdej z realizacji. Wartości parametrów modelu LB, obliczone w taki sposób, przy wykorzystaniu sekwencji niezaszumionych, przedstawiono w tab. 6.1, a dla nowego modelu w tab. 6.2.

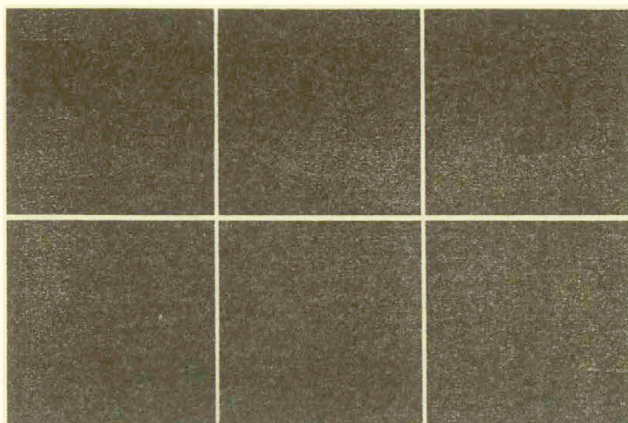
Zaprezentowane tablice tłumaczą sposób wyboru sekwencji testowych. Estymaty parametrów m_1 modelu LB dla sekwencji K oraz L są identyczne, chociaż sekwencje K oraz L wyraźnie się różnią między sobą. W przypadku nowego modelu powyższy problem nie



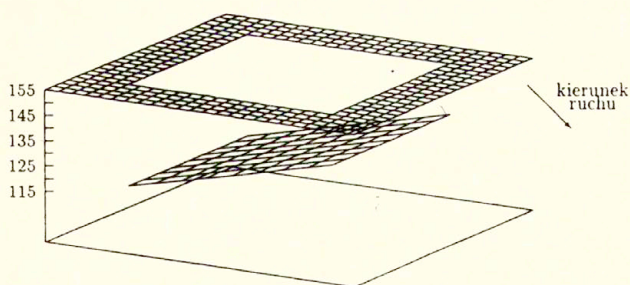
Rys. 6.1: Sekwencja K. Jasność obiektu (przed zaszumieniem) zmienna liniowo w zakresie od 135 do 155, jasność tła jest równa 115.



Rys. 6.2: Wzajemna relacja pomiędzy jasnością tła i obiektu dla sekwencji K z rys. 6.1.



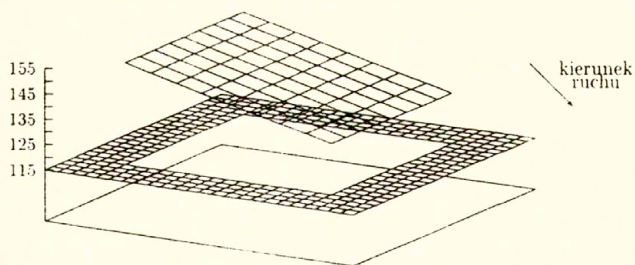
Rys. 6.3: Sekwencja L. Jasność obiektu (przed zaszumieniem) zmienna liniowo w zakresie od 115 do 135, jasność tła jest równa 155.



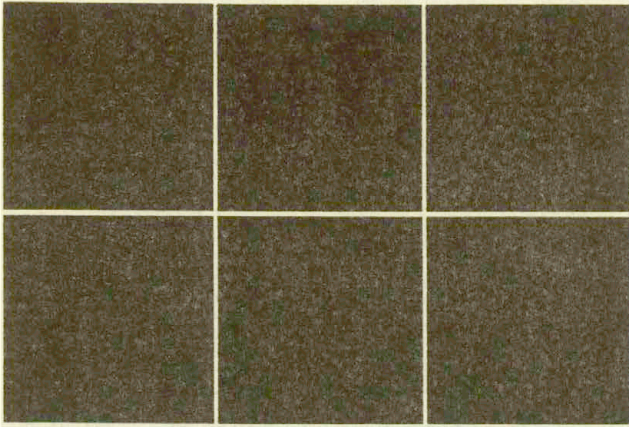
Rys. 6.4: Wzajemna relacja pomiędzy jasnością tła i obiektu dla sekwencji L z rys. 6.3.



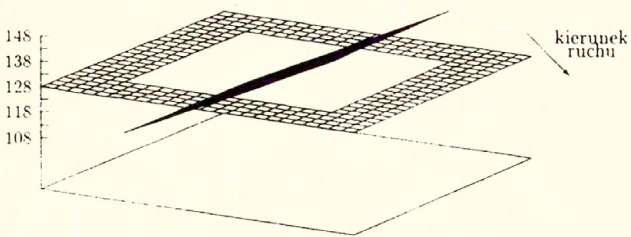
Rys. 6.5: Sekwencja M. Jasność obiektu (przed zaszumieniem) zmienna liniowo w zakresie od 115 do 135, jasność tła jest równa 155.



Rys. 6.6: Wzajemna relacja pomiędzy jasnością tła i obiektu dla sekwencji M z rys. 6.5.



Rys. 6.7: Sekwencja N. Jasność obiektu (przed zaszumieniem) zmienna liniowo w zakresie od 108 do 148, jasność tła jest równa 128.



Rys. 6.8: Wzajemna relacja pomiędzy jasnością tła i obiektu dla sekwencji N z rys. 6.7.

	\hat{m}_1	\hat{m}_2
	(a, b) lub (b, a)	(a, a)
sekwencja K	2.27	-10
sekwencja L	2.27	-10
sekwencja M	-2.27	10
sekwencja N	4.55	-20

Tab. 6.1: Estymaty parametrów m_1 oraz m_2 modelu LB dla niezaszumionych sekwencji K, L, M, N. Parametry zostały estymowane metodą uśrednienia różnicy σ_k dla odpowiednich realizacji.

	\hat{m}_1	\hat{m}_2	\hat{m}_3
	(a, b)	(b, a)	(a, a)
sekwencja K	-27.73	32.27	-10
sekwencja L	32.27	-27.73	-10
sekwencja M	-32.27	27.73	10
sekwencja N	4.55	4.55	-20

Tab. 6.2: Estymaty parametrów m_1 , m_2 oraz m_3 nowego modelu z dwuwymiarowym wektorem obserwacji dla niezaszumionych sekwencji K, L, M, N. Parametry zostały estymowane metodą uśrednienia różnicy σ_k dla odpowiednich realizacji.

występuje. Wartości parametrów m_1 oraz m_2 odpowiadające za sytuacje przejściowe, są w przypadku wszystkich sekwencji różne.

Rozdział 7

Badania eksperymentalne modelu LB

W niniejszym rozdziale przedstawiono wyniki badań eksperymentalnych modelu LB. W pracach [BL90, LB90] zaprezentowano jedynie skąpe efekty detekcji ruchu otrzymane dzięki zastosowaniu tego modelu. Badania, których wyniki zamieszczono w niniejszy rozdziale, zostały przeprowadzone przede wszystkim pod kątem oceny użyteczności modelu LB do detekcji masek obiektów ruchomych. Z tego powodu gruntownie przebadano wrażliwość tego modelu na wartości parametrów, jak również, korzystając z kryterium wprowadzonego w rozdziale 6, oceniono dokładność znajdowanych masek obiektu ruchomego. Na podstawie otrzymanych wyników określono kryteria, którymi można się kierować w trakcie doboru parametrów modelu. Dodatkowym powodem przeprowadzenia eksperymentów była chęć porównania modelu znanego z literatury z nowym modelem z dwuwymiarowym wektorem obserwacji zaproponowanym w niniejszej rozprawie.

7.1 Rola poszczególnych składników energii modelu LB

W pracach [BL90, LB90] zaproponowano takie same wartości współrzędnych wektorów Θ_s , Θ_t , które w niniejszej rozprawie zostały przedstawione w rozdziale 5.2. Na ich podstawie zostaną przeanalizowane obecnie wzajemne relacje energii $U_{ijs} = U_{ijs_1} + U_{ijs_2}$ z energią czasową U_{ijt} . Przyjęcie $\beta_s = 10$ spowodowało, że energia poszczególnych klik przestrzennych może przybierać tylko wartości 10 albo -10 . W konsekwencji wartość każdej z lokalnych energii przestrzennych należy do przedziału $[-80, 80]$, a wartość całej energii przestrzennej U_{ijs} nigdy nie będzie mniejsza niż -160 i większa od 160 . Sytuacja optymalna, tzn. $U_{ijs} = -160$ ma miejsce wtedy, gdy w dwóch kolejnych obrazach pikseli oraz wszystkim jego sąsiadom przestrzennym przyporządkowane są takie same etykiety. W najbardziej niekorzystnym przypadku $U_{ijs} = 160$. Odpowiada to sytuacji, w której pikseli w kolejnych obrazach przyporządkowano przeciwne etykiety niż wszystkim jego sąsiadom przestrzennym.

Wg [BL90, LB90] parametry określające energię klik czasowych mają wartości: $\beta_t = 100$ oraz $\beta'_t = 1000$. Oznacza to, że wartość lokalnej energii czasowej może zawierać się w przedziale $[-100, 1000]$. Tak jak zostało to już wspomniane w rozdziale 5, para etykiet, dla której energia czasowa przybiera wartość 1000 praktycznie jest w sposób deterministyczny wykluczona, co może być przyczyną błędów popełnianych w trakcie predykcji. Wszystkie pozostałe przypadki dla klik czasowych dają wartości energii czasowej zbliżone do zakresu zmian energii przestrzennej. Maksymalna co do modułu wartość U_{ijs} jest o 60% większa od energii czasowej, a to oznacza, że w modelu LB spójność przestrzenna ma większy priorytet niż czasowa.

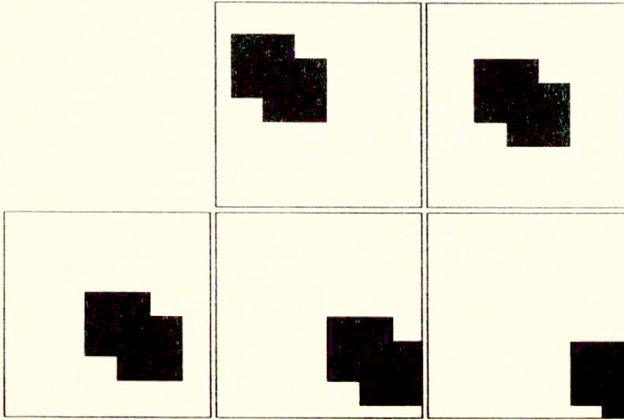
7.2 Badania eksperymentalne – sekwencja K

We wszystkich eksperymentach jako obserwację \bar{o}_k wybrano idealne maski zmian w obrazie z rys. 7.1, a inicjację korekcji pierwszej maski obiektu przeprowadzono idealną maską z rys. 7.2. Ze względu na czas obliczeń, do estymowania mody modelu LB, wybrano algorytm deterministycznej relaksacji opisany w rozdziale 5.4, przy czym piksele w każdej iteracji były przemiatane z wykorzystaniem kodinów [Bes74, Kur], dzięki czemu uniknięto narzuconej niejednorodności [Car92] znajdujących masek.

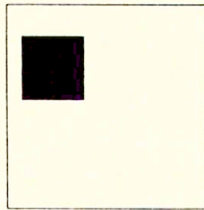
Ocenę jakości znajdujących masek obiektu ruchomego przeprowadzono przy użyciu statystyk $\neg a$, $\neg b$ określonych na tych maskach. Jako kryterium stabilności procesu detekcji ruchu wybrano liczbę iteracji, po której dla każdej z masek osiągnany jest stan stabilny. Ponieważ maska odpowiadająca pierwszemu obrazowi sekwencji jest nietypowa, gdyż poddawana jest tylko korekcji, więc dodatkowo zdecydowano się na porównanie wartości statystyk $\neg a$, $\neg b$ dla pierwszej i drugiej maski obiektu ruchomego.

Na początku postawiono sobie za cel wykrycie masek obiektu dla sekwencji K z rys. 6.1. Wartości parametrów m_1 oraz m_2 przyjęto zgodnie z tab. 6.1. Parametrowi σ^2 przypisano wartość wariancji dodanego szumu, tzn. 225.

Znalezione maski obiektu przedstawiono na rys. 7.3. Detekcja ruchu została zakończona sukcesem. Jedynie dla ostatniego obrazu znaleziona maska nie przypomina obiektu



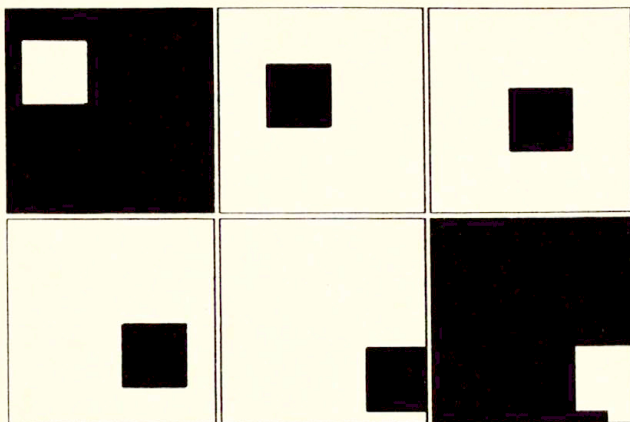
Rys. 7.1: Idealne maski zmian czasowych dla sekwencji K, L, M, N.



Rys. 7.2: Idealna maska obiektu ruchomego dla pierwszego obrazu sekwencji K, L, M, N, używana w algorytmie relaksacji deterministycznej do inicjacji korekcji maski tego obrazu.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = \hat{m}_1 = 2.27, m_2^0 = \hat{m}_2 = -10, \sigma_0^2 = 225$								
	0	11	3	0	9	0	21/21	
m_1	$m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
-30	0	10	2	0	52	0	21/21	$m_1 \ll m_2^0$
-20	0	12	3	0	76	0	21/21	$m_1 < m_2^0$
-11	0	12	3	0	125	0	21/21	$m_1 \leq m_2^0$
-9	0	16	3	0	9	0	21/21	$m_1 \geq m_2^0$
30	0	7	3	0	4	0	21/21	$m_1 \gg m_2^0$
m_2	$m_1 = m_1^0, \sigma^2 = \sigma_0^2$							
-30	0	16	3	0	11	0	21/21	$m_2 \ll m_1^0$
1	0	9	3	0	4	0	21/21	$m_2 \leq m_1^0$
10	0	471	25*	0	551	0	25*/21	$m_2 > m_1^0$
20	0	788	25*	0	862	0	25*/21	$m_2 \gg m_1^0$
σ^2	$m_1 = m_1^0, m_2 = m_2^0$							
22.5	0	11	3	0	9	0	21/21	$U_z \times 10$
2.25	5	255	4	7	232	0	5/4	$U_z \times 100$
2250	0	11	3	0	9	0	21/21	$U_z \times 0.1$

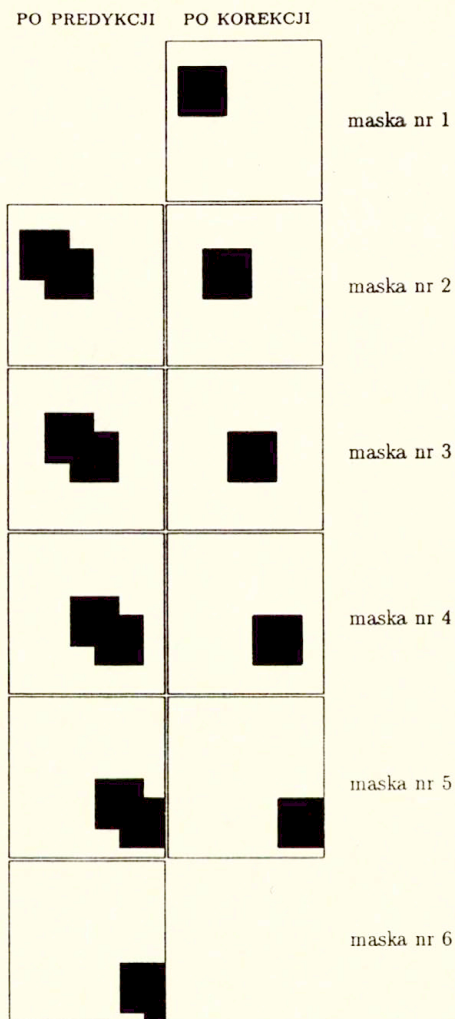
Tab. 7.1: Wpływ zmian parametrów modelu LB na wyniki detekcji masek obiektu ruchomego odpowiadających sekwencji K z rys. 6.1. Grupa kolumn oznaczonych MASKA 1 zawiera wyniki wyłącznie korekcji, ponieważ odnosi się do maski znalezionej dla pierwszego obrazu sekwencji K. W grupie kolumn oznaczonych MASKA 2 zebrano wyniki dotyczące maski odpowiadającej drugiemu obrazowi sekwencji K, tzn. maski, która była poddana zarówno predykcji jak i korekcji. W kolumnach oznaczonych $\neg a$, $\neg b$ zebrano wartości statystyk, będących podstawą kryterium do oceny dokładności znalezionych masek, wprowadzonego w rozdziale 6. W kolumnie oznaczonej k_1 zebrano liczby iteracji algorytmu relaksacji deterministycznej, w implementacji kodingowej, po których maska odpowiadająca pierwszemu obrazowi sekwencji K nie ulegała dalszym zmianom. W kolumnie oznaczonej p_2 podano analogiczną liczbę iteracji ale odnoszącą się do predykcji maski dla drugiego obrazu sekwencji K. W kolumnie oznaczonej k_2/p_3 podano liczbę iteracji potrzebnych do przeprowadzenia korekcji maski drugiej i predykcji maski trzeciej. Parametry m_1^0, m_2^0 zostały przyjęte wg tab. 6.1, tzn. są to wartości obliczone dla sekwencji niezasumionej. Brak wskaźnika 0 przy parametrze oznacza, że jest on zmieniany arbitralnie. Symbol * przy liczbie iteracji oznacza, że nawet w tej iteracji nie osiągnięto stanu stabilnego.



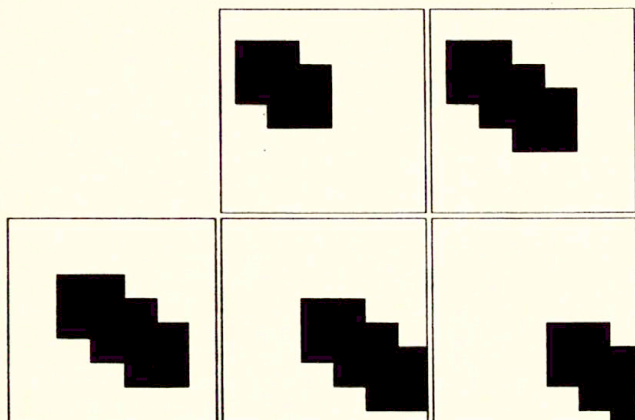
Rys. 7.3: Maski obiektu ruchomego znalezione dla sekwencji K – model LB, relaksacja deterministyczna. Parametry: $m_1 = \hat{m}_1 = 2.27$ (estymowane), $m_2 = \hat{m}_2 = -10$ (estymowane), $\sigma^2 = 225$. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.

a tylko maskę zmian. Przyczyną jest to, że dla ostatniego obrazu przeprowadzany jest jedynie etap predykcji. Wartości statystyk dla sekwencji masek z rys. 7.3 przedstawione są w pierwszym wierszu tab. 7.1. Dokładność znalezionych masek wydaje się być dobra. Tylko 11 pikselom obiektu błędnie przypisano etykietę b po korekcji pierwszej maski. Podobny problem wystąpił po predykcji i korekcji maski nr 2. Ani jednego piksela tła nie zakwalifikowano błędnie jako należącego do obiektu ruchomego. Jest to najprawdopodobniej skutkiem wykorzystania idealnych masek zmian w obrazie.

Na rys. 7.4 porównano ze sobą maski obiektu ruchomego otrzymane po predykcji, a następnie poprawione po korekcji. Analizując je można stwierdzić, że teoretyczne rozważania przedstawione w rozdziale 5.3. dotyczące błędnie przypisywanych etykiet a w czasie predykcji, są prawdziwe. Dodatkowym potwierdzeniem sformułowanego w rozdziale wniosku są wyniki przedstawione na rys. 7.5, 7.6, 7.7. Na rys. 7.5 przedstawione są maski inicjujące predykcję wszystkich, poza pierwszą, masek obiektu ruchomego. Z powodu wykluczenia realizacji (a, b) maska inicjująca predykcję jest za każdym razem nadzbiorem odpowiedniej maski zmian w obrazie. Na rys. 7.6 pokazane są zmiany maski po kolejnych iteracjach predykcji. Na rysunku tym bardzo wyraźnie widoczne jest „mozolne” usuwanie zbędnych etykiet przypisanych w trakcie inicjacji. Jednocześnie maska znaleziona po predykcji jest całkowicie zgodna z maską zmian, czego powodem ponownie jest wykluczenie realizacji (a, b) . Na rys. 7.7 pokazane są również zmiany znajdowanej maski, ale tym razem po kolejnych iteracjach korekcji. Wyraźnie widoczne są konsekwencje predykcji przeprowadzonej „na wyrost”, jak poprzednio z powodu wykluczenia realizacji (a, b) .



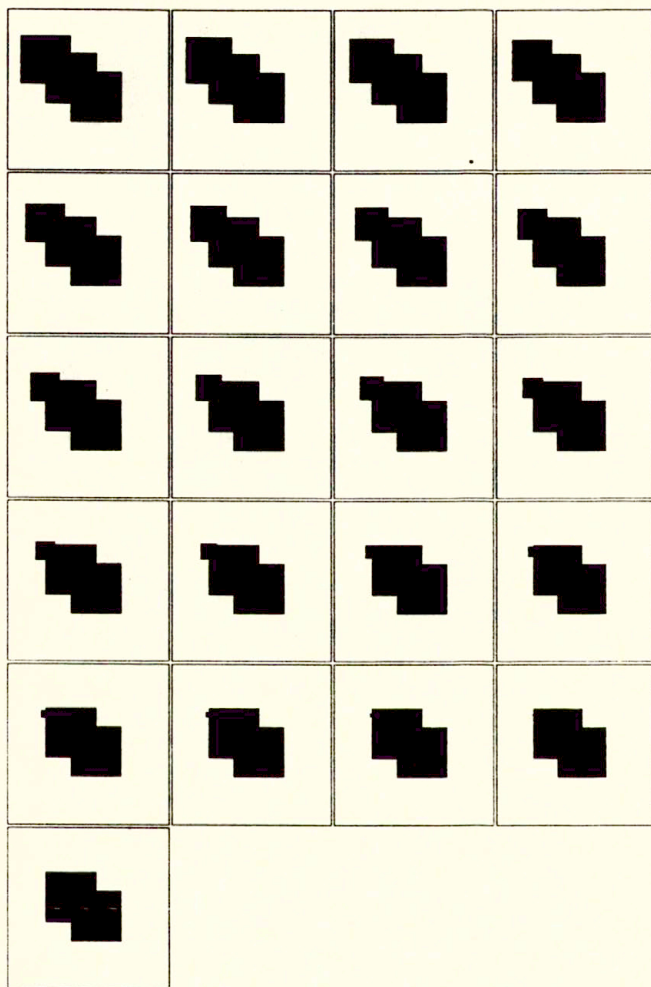
Rys. 7.4: Porównanie masek obiektu znalezionych jedynie po predykcji (lewo) a następnie poprawionych w trakcie korekcji (prawo) dla sekwencji K, model LB, algorytm relaksacji deterministycznej. Wartości parametrów jak dla rys. 7.3 (estymowane). Maskę nr 1 poddana została tylko korekcji, a maskę nr 6 jedynie predykcji.



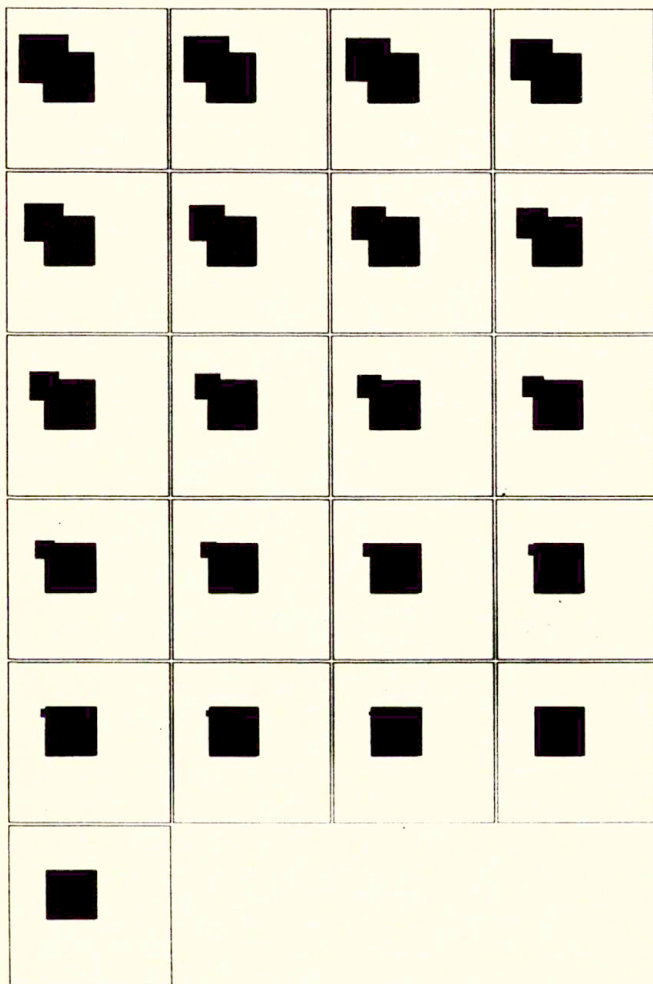
Rys. 7.5: Maski inicjujące predykcję masek odpowiadających obrazom 2-6 sekwencji K, znalezione wg metody inicjacji przedstawionej w rozdziale 5.4, model LB, algorytm relaksacji deterministycznej. Wartości parametrów jak dla rys. 7.3 (estymowane).

Konsekwencje odtwarzania maski zmian w trakcie predykcji widoczne są także w kolumnie k_2/p_3 tab. 7.1, w której podano liczbę iteracji potrzebnych do osiągnięcia minimum energii podczas korekcji maski 2 i jednoczesnej predykcji maski 3. Jak było już wspomniane, analizowana sekwencja przedstawia ruch szybki, czyli pikseli z sytuacji przejściowych jest względnie dużo. Dla maski pierwszej, dla której nie przeprowadzono predykcji i jednocześnie brak jest źle przydzielonych etykiet a , stan stabilny osiągany jest już po trzech iteracjach. W przypadku maski nr 2 korekcja trwa aż 21 iteracji, gdyż musiała być usunięta duża liczba fałszywych etykiet a .

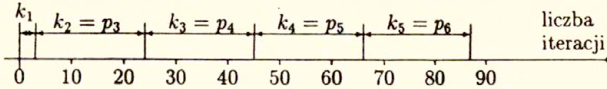
Jednym z parametrów pozwalających ocenić użyteczność metody dla zastosowań praktycznych jest czas obliczeń potrzebny do znalezienia rozwiązania. Ponieważ czas wyrażony w jednostkach bezwzględnych jest zależny od konkretnego systemu komputerowego, lepiej jest określać go za pomocą liczby iteracji. Rys. 7.8 obrazuje czas trwania poszczególnych kroków metody (krokiem i określa poszukiwanie optymalnej realizacji etykiet dla obrazów i oraz $i + 1$) wymaganych do znalezienia masek z rys. 7.3. Kolumny k_1, p_2 tab. 7.1 informują, że jeżeli tylko algorytm jest stabilny, to pierwszy krok metody ($i = 1$) trwa najczęściej o wiele krócej niż pozostałe. Chociaż predykcja p_2 najczęściej nie wymaga żadnej iteracji, to jednak musi być ona wykonywana aż do ustabilizowania się korekcji k_1 , gdyż czas trwania kroku i jest równy maksymalnej z liczb k_i oraz p_{i+1} . Mała liczba iteracji wymagana dla kroku pierwszego wynika z jego nietypowości, gdyż korekcja maski pierwszej w przeprowadzonych eksperymentach była inicjowana idealną maską. Pozostałe kroki (w przypadkach stabilnych) trwają najczęściej „aż” 21 iteracji. Oznacza to, że rezultaty z rys. 7.3 zostały otrzymane po 87 iteracjach, co też zostało przedstawione na rys. 7.8.



Rys. 7.6: Kolejne iteracje wyniku predykcji maski obiektu odpowiadającej trzeciemu obrazowi sekwencji K, model LB, algorytm relaksacji deterministycznej. Wartości parametrów jak dla rys. 7.3 (estymowane).



Rys. 7.7: Kolejne iteracje wyniku korekcji maski odpowiadającej drugiemu obrazowi sekwencji K, model LB, algorytm relaksacji deterministycznej. Wartości parametrów jak dla rys. 7.3 (estymowane).



Rys. 7.8: Rysunek obrazujący czas obliczeń wyrażony za pomocą liczby iteracji algorytmu relaksacji deterministycznej, potrzebnych do znalezienia masek z rys. 7.3 – model LB. Symbole k_i oraz p_i odnoszą się odpowiednio do korekcji i predykcji maski odpowiadającej i -temu obrazowi sekwencji K. Łączny czas obliczeń wynosi 87 iteracji.

Implementacja algorytmu relaksacji deterministycznej z normalnym przeglądaniem piksli, tzn. wiersz po wierszu kolumna po kolumnie, pozwala na skrócenie czasu obliczeń. Przeprowadzając eksperymenty analogiczne do tych z rys. 7.3, ale zmieniając tylko sposób przeglądania piksli na normalny, korekcja zarówno maski nr 1 jak i 2 zajęła tylko 5 iteracji. Jednak w celu uniknięcia efektów powodowanych przez normalne przeglądanie piksli [Car92] we wszystkich badaniach, których wyniki zamieszczone zostały w niniejszej rozprawie, stosowano konsekwentnie implementację kodującą algorytmu relaksacji deterministycznej.

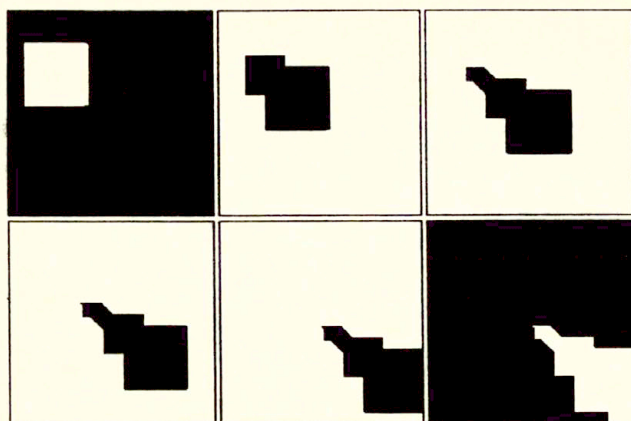
Na rys. 7.9 przedstawiono rezultat, gdy z góry ograniczono do ośmiu w każdym kroku liczbę iteracji algorytmu relaksacji deterministycznej. Widać, że fałszywe etykiety przypisane podczas predykcji nie zostały usunięte w czasie korekcji, a przez to propagują w następnych krokach.

Następną serię eksperymentów przeprowadzono w celu sprawdzenia wrażliwości modelu na wartość parametru m_1 . Sytuacją wyjściową była sytuacja z rys. 7.3. Wartość m_1 zmieniano tak, aby uwzględnić możliwe wszystkie sytuacje, tzn. $m_1 \ll \hat{m}_2$, $m_1 \approx \hat{m}_2$ oraz $m_1 \gg \hat{m}_2$. Przykładowe maski przedstawiono na rys. 7.10 przy $m_1 \leq \hat{m}_2$. Widoczne są spore braki piksli obiektu. Pełne wyniki przedstawiono w pierwszej części tab. 7.1. Wynika z nich, że najlepszą maskę otrzymano nie tylko gdy $m_1 > \hat{m}_2$ (tak jest w sytuacji wyjściowej), ale gdy dodatkowo m_1 ma wartość dużo większą niż wartość estymowana \hat{m}_2 .

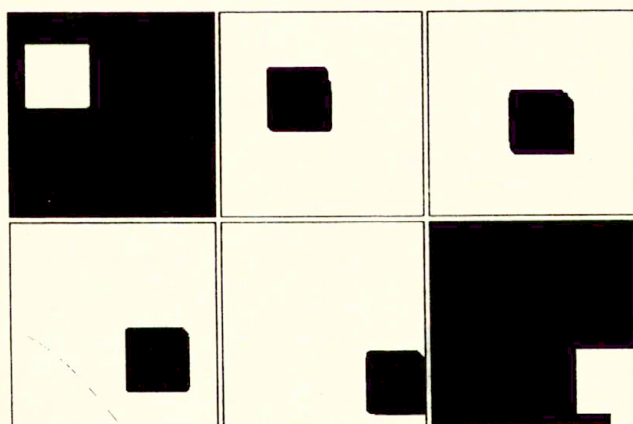
Z kolei przebadano wrażliwość modelu na wartość m_2 – druga część tab. 7.1. Znowu rezultaty są zadowalające, gdy tylko $\hat{m}_1 > m_2$. W przypadkach, w których $m_2 > \hat{m}_1$ algorytm nie znajduje minimum energii, a otrzymane maski (por. rys. 7.11) tylko w pewnym stopniu przypominają obiekt.

Trzecia seria eksperymentów, ilustrowana końcową częścią tab. 7.1, miała na celu zbadanie wpływu energii zgodności na rezultaty detekcji. Można zwiększyć udział energii zgodności k razy poprzez zmniejszenie k razy parametrów β_s , β_t i β'_t . Można tego również dokonać zmniejszając k razy wartość parametru wariancji σ^2 energii zgodności. Zaprezentowane w tab. 7.1 wyniki sugerują, że 10-krotne zmniejszenie lub zwiększenie poziomu energii zgodności nie ma najmniejszego wpływu na otrzymane rezultaty, w przypadku gdy wyjściowe wartości parametrów m_1^0 , m_2^0 zapewniały stabilność rozwiązania. Natomiast 100-podwyższenie udziału energii zgodności prowadzi już do złych wyników, patrz np. rys. 7.12.

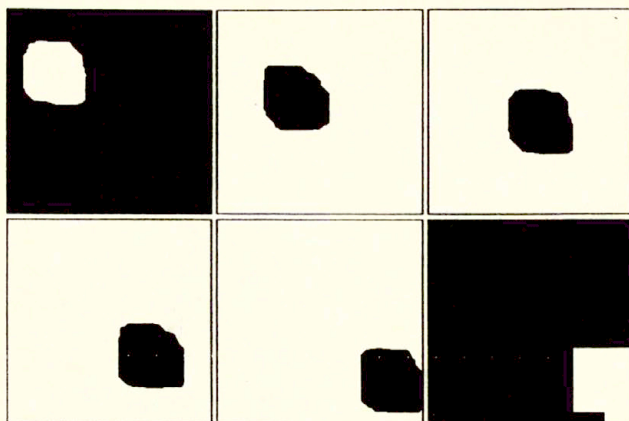
Reasumując eksperymenty przeprowadzone dla sekwencji K można stwierdzić, że w zasadzie (z dokładnością do kilkunastu piksli) wartości parametrów m_1 i m_2 nie mają większego wpływu na otrzymywane wyniki (do takiego wniosku doszli również Lalande



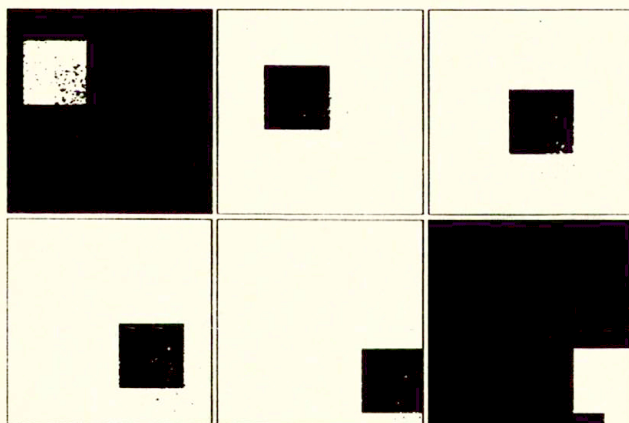
Rys. 7.9: Maski obiektu ruchomego z sekwencji K, model LB, algorytm relaksacji deterministycznej. Wartości parametrów jak dla rys. 7.3. Ze względu na zbyt małą liczbę (8) iteracji algorytmu relaksacji deterministycznej nie zostało osiągnięte lokalne minimum energii. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.



Rys. 7.10: Maski obiektu ruchomego z sekwencji K, model LB, algorytm relaksacji deterministycznej, $m_1 = -11$, $m_2 = \hat{m}_2 = -10$ (estymowane), $\sigma^2 = 225$, ($m_1 \leq \hat{m}_2$). Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.



Rys. 7.11: Maski obiektu ruchomego z sekwencji K, model LB, algorytm relaksacji deterministycznej, $m_1 = \hat{m}_1 = 2.27$ (estymowane), $m_2 = 20$, $\sigma^2 = 225$, ($m_2 \gg \hat{m}_1$). Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.



Rys. 7.12: Maski obiektu ruchomego z sekwencji K, model LB, algorytm relaksacji deterministycznej, $m_1 = \hat{m}_1 = 2.27$ (estymowane), $m_2 = \hat{m}_2 = -10$ (estymowane), $\sigma^2 = 2.25$, (energia zgodności ma 100 krotnie większy udział w energii pola niż w sytuacji przedstawionej na rys.7.3). Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = \hat{m}_1 = 2.27, m_2^0 = \hat{m}_2 = -10, \sigma^2 = 225$								
	0	695	25*	0	308	0	25*/21	
m_1	$m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
-30	0	4	1	0	6	0	21/21	$m_1 \ll m_2^0$
-20	0	11	4	0	6	0	21/21	$m_1 < m_2^0$
-11	0	18	6	0	6	0	21/21	$m_1 \leq m_2^0$
-9	0	1091	25*	0	620	0	25*/21	$m_1 \geq m_2^0$
30	0	179	12	0	103	0	25*/21	$m_1 \gg m_2^0$
m_2	$m_1 = m_1^0, \sigma^2 = \sigma_0^2$							
-30	0	1193	25*	0	631	0	25*/21	$m_2 \ll m_1^0$
1	0	476	25*	0	209	0	25*/21	$m_2 \leq m_1^0$
10	0	23	7	0	20	0	21/21	$m_2 > m_1^0$
20	0	79	20	0	40	0	21/21	$m_2 \gg m_1^0$
σ^2	$m_1 = m_1^0, m_2 = m_2^0$							
22.5	21	932	25*	31	649	0	25*/21	$U_z \times 10$
2.25	4327	4540	5	4320	4529	1	5/4	$U_z \times 100$
2250	0	695	25*	0	308	0	25*/21	$U_z \times 0.1$

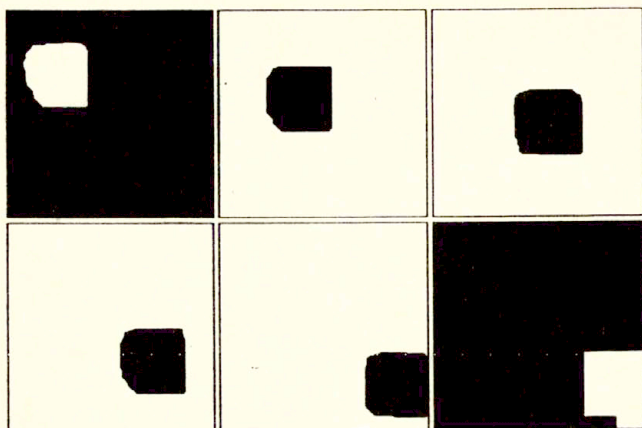
Tab. 7.2: Wpływ zmian parametrów modelu LB na wyniki detekcji masek obiektu ruchomego odpowiadających sekwencji L z rys. 6.3. Znaczenie poszczególnych kolumn jest identyczne jak dla tab. 7.1.

oraz Bouthemey). Ważna jest wzajemna relacja tych dwóch parametrów. W przypadku sekwencji K okazało się, że musi być spełniony warunek $m_1 > m_2$. Dobór poziomu energii zgodności, tzn. wartości parametru σ^2 , jeżeli korzysta się z idealnych masek zmian w obrazie, nie będzie miał większego wpływu na wynik detekcji.

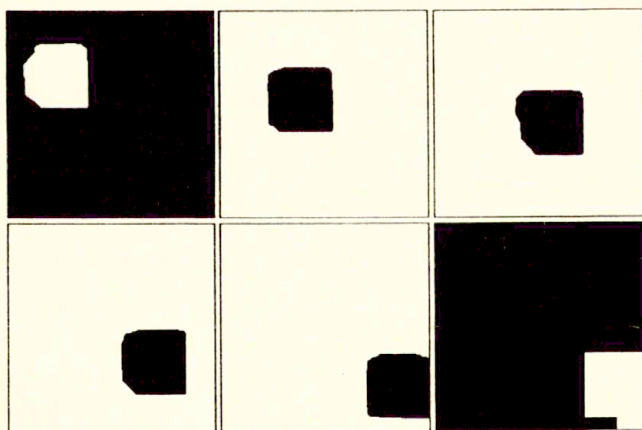
7.3 Badania eksperymentalne – sekwencje L,M,N

Podobną serię eksperymentów przeprowadzono dla sekwencji L z rys. 6.3. Estymowane wartości parametrów \hat{m}_1 i \hat{m}_2 są identyczne dla sekwencji K i L. Rezultaty przeprowadzonych badań przedstawione są w tab. 7.2, a znalezione maski na rys. 7.13. Od razu widać, że wyniki otrzymane dla dokładnych wartości parametrów są niedobre. Algorytm relaksacji deterministycznej nawet po 25 iteracjach nie osiągnął minimum energii, chociaż parametry modelu zostały wyestymowane korzystając z sekwencji niezasmuconej. Badania przeprowadzone przy różnych wartościach parametrów m_1 i m_2 sugerują, że w dobre wyniki uzyskiwane są, gdy $m_1 < m_2$, tzn. przeciwnie niż w przypadku sekwencji K. Oczywiście zwiększanie udziału energii zgodności przy źle dobranych parametrach m_1 i m_2 prowadzi do ogromnych absurdów.

Dla sekwencji M również przeprowadzono analogiczną serię eksperymentów. Rezultaty poszukiwania masek, przy dokładnych estymatach parametrów, przedstawiono na



Rys. 7.13: Maski obiektu ruchomego z sekwencji L, model LB, algorytm relaksacji deterministycznej, $m_1 = \hat{m}_1 = 2.27$ (estymowane), $m_2 = \hat{m}_2 = -10$ (estymowane), $\sigma^2 = 225$. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.



Rys. 7.14: Maski obiektu ruchomego z sekwencji M, model LB, algorytm relaksacji deterministycznej, $m_1 = \hat{m}_1 = -2.27$ (estymowane), $m_2 = \hat{m}_2 = 10$ (estymowane), $\sigma^2 = 225$. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.

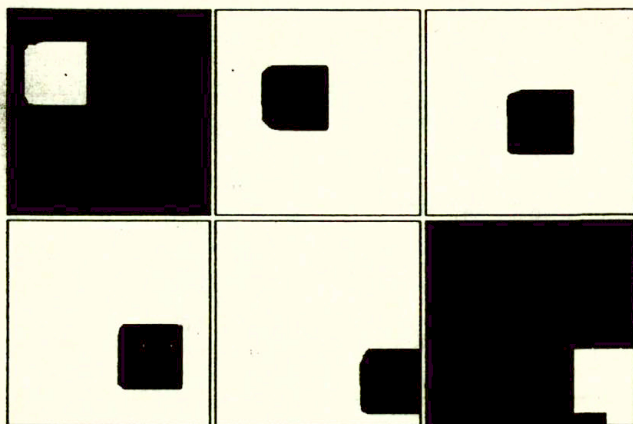
	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = \hat{m}_1 = -2.27, m_2^0 = \hat{m}_2 = 10, \sigma_0^2 = 225$								
	0	618	25*	0	281	0	25*/21	
m_1	$m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
-20	0	256	25*	0	243	0	25*/21	$m_1 \ll m_2^0$
0	0	678	25*	0	267	0	25*/21	$m_1 < m_2^0$
9	0	991	25*	0	288	0	25*/21	$m_1 \leq m_2^0$
11	0	6	2	0	4	0	21/21	$m_1 \geq m_2^0$
50	0	6	2	0	4	0	21/21	$m_1 \gg m_2^0$
m_2	$m_1 = m_1^0, \sigma^2 = \sigma_0^2$							
-30	0	89	20	0	57	0	21/21	$m_2 \ll m_1^0$
-4	0	24	6	0	15	0	21/21	$m_2 \leq m_1^0$
-1	0	256	25*	0	243	0	25*/21	$m_2 > m_1^0$
20	0	960	25*	0	288	0	25*/21	$m_2 \gg m_1^0$
σ^2	$m_1 = m_1^0, m_2 = m_2^0$							
22.5	0	946	25*	0	835	0	25*/21	$U_z \times 10$
2.25	4318	4591	5	4304	4586	1	5/4	$U_z \times 100$
2250	0	601	25*	0	265	0	25*/21	$U_z \times 0.1$

Tab. 7.3: Wpływ zmian parametrów modelu LB na wyniki detekcji masek obiektu ruchomego odpowiadających sekwencji M z rys. 6.5. Znaczenie poszczególnych kolumn jest identyczne jak dla tab. 7.1.

rys. 7.14, a wyniki całej serii eksperymentów w tab. 7.3. I znowu obliczone wartości parametrów \hat{m}_1 i \hat{m}_2 nie sprawdzily się. Podobnie jak dla sekwencji L, aby otrzymać akceptowalne wyniki, parametry muszą spełniać zależność $m_1 > m_2$.

Analogicznie jak dla sekwencji K, L, M, przebadano użyteczność modelu LB dla sekwencji N przedstawiającej obiekt, który jest zarówno jaśniejszy jak i ciemniejszy od tła. Znalezione maski obiektu ruchomego przedstawione są na rys. 7.15, a wyniki całej serii eksperymentów zebrano w tab. 7.4. Na podstawie rys. 7.15 oraz tab 7.4 można dojść do wniosku, że w przypadku modelu LB trudno jest znaleźć tak dobre, jak np. dla sekwencji K, maski obiektu ruchomego, który jest zarówno jaśniejszy jaki i ciemniejszy od tła.

W celu interpretacji otrzymanych wyników przeprowadźmy analizę co zachodzi lokalnie w czasie działania algorytmu relaksacji deterministycznej. W pewnym uproszczeniu można powiedzieć, że energia zgodności decyduje o etykietach, gdy pozostałe składniki energii lokalnej są w równowadze. Nazwijmy *przypadkiem przestrzennie obojętnym* sytuację, w której czterech sąsiadów piksla ma przyporządkowaną etykietę przeciwną, niż cztery pozostałe. W takim przypadku przypisanie piksłowi dowolnej z dwóch etykiet nie spowoduje żadnej różnicy w wartości energii przestrzennej. Decydującą rolę musi odegrać suma energii czasowej i zgodności. Przeanalizujemy energię czasową. Jeżeli danemu piksłowi odpowiada etykieta $\bar{o}_k = 1$ maski zmian, co oznacza, że w danym piksle wykryto



Rys. 7.15: Maski obiektu ruchomego z sekwencji N, model LB, algorytm relaksacji deterministycznej, $m_1 = \hat{m}_1 = 4.55$ (estymowane), $m_2 = \hat{m}_2 = -20$ (estymowane), $\sigma^2 = 225$. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.

zmiany, to w takim przypadku istnieją dwie możliwości zaetykietowania danego piksla: (b, a) oraz (a, a) , ale obie realizacje mają przypisane jednakowe wartości energii. Oznacza to, że decydującą rolę musi odegrać energia zgodności.

Wg toku rozumowania przedstawionego w rozdziale 5.3 przy opisie nowego modelu realizacja (a, b) powinna być zastąpiona realizacją (a, a) . Jednak w niektórych przypadkach zamiast (a, a) przyporządkowywana jest realizacja (b, a) . Pojawia się pytanie, jaki warunek musi spełniać energia zgodności, aby wykluczyć błędne przypisywanie etykiet (b, a) ? Odpowiedź jest łatwo znaleźć, gdy mamy pewną wiedzę a priori o analizowanej sekwencji, szczególnie dotyczącą jasności tła i obiektu. Zaczniemy od sekwencji K, ale dla uproszczenia zapomnijmy o dodanym szumie. W sytuacjach, którym w rzeczywistości powinna odpowiadać realizacja (a, b) , różnica o_k ma zawsze wartość mniejszą do zera. Aby zapewnić większą wartość energii zgodności dla sytuacji przejściowej w stosunku do sytuacji (ruch, ruch) wystarczy, aby parametr $m_1 > m_2$. Do takich wniosków też doszliśmy analizując wyniki eksperymentów przeprowadzonych dla sekwencji K. Różnice pomiędzy sytuacjami, gdy ten warunek jest spełniony, wynikają z obecności szumu o dużej wariancji dodanego do obrazu.

W przypadku sekwencji L mamy do czynienia z odwrotną sytuacją, tzn. tam gdzie powinna być przyporządkowana realizacja (a, b) różnica o_k ma zawsze wartość dodatnią, czyli poprawne działanie algorytmu będzie miało miejsce, gdy $m_1 < m_2$. Niestety estymaty \hat{m}_1 , \hat{m}_2 , które zostały obliczone poprzez zwykłe uśrednienie spełniają relację przeciwną. Stąd wynikają złe rezultaty przedstawione na rys. 7.13.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = \hat{m}_1 = 4.55, m_2^0 = \hat{m}_2 = -20, \sigma_0^2 = 225$								
	0	141	16	0	176	0	25*/21	
m_1	$m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
-40	0	4	1	0	6	0	21/21	$m_1 \ll m_2^0$
-21	0	8	2	0	10	0	21/21	$m_1 \leq m_2^0$
-19	0	467	25*	0	300	0	25*/21	$m_1 \geq m_2^0$
0	0	205	15	0	215	0	25*/21	$m_1 \geq m_2^0$
20	0	44	8	0	101	0	24/21	$m_1 \gg m_2^0$
m_2	$m_1 = m_1^0, \sigma^2 = \sigma_0^2$							
-40	0	431	25*	0	300	0	25*/21	$m_2 \ll m_1^0$
0	0	36	8	0	49	0	24/21	$m_2 \leq m_1^0$
10	0	222	25*	0	103	0	21/21	$m_2 > m_1^0$
40	0	760	25*	0	426	0	25*/21	$m_2 \gg m_1^0$
σ^2	$m_1 = m_1^0, m_2 = m_2^0$							
22.5	21	817	25*	10	631	0	25*/20	$U_z \times 10$
2.25	992	3642	5	995	3634	3	5/3	$U_z \times 100$
2250	0	141	16	0	176	0	25*/21	$U_z \times 0.1$

Tab. 7.4: Wpływ zmian parametrów modelu LB na wyniki detekcji masek obiektu ruchomego odpowiadających sekwencji N z rys. 6.7. Znaczenie poszczególnych kolumn jest identyczne jak dla tab. 7.1.

Dla sekwencji M przyczyna znajdowania złych masek jest podobna jak dla sekwencji L, z tym, że tym razem σ_k jest ujemne, a obliczone parametry \hat{m}_1 i \hat{m}_2 spełniają relację $\hat{m}_1 < \hat{m}_2$.

Na podstawie przeprowadzonego rozumowania można sformułować pewne ogólniejsze wnioski odnoszące się do modelu LB zastosowanego do detekcji ruchu.

- Przyjęta postać energii zgodności $U_{i,z}$, ze względu na łączenie realizacji (a, b) oraz (b, a) , oparta jest na zbyt dużym uproszczeniu modelowania.
- Energia zgodności pełni rolę decydującą w sytuacjach przestrzennie obojętnych i czasowo niejednoznacznych (taka sama wartość energii przyporządkowana różnym, dopuszczalnym realizacjom kliki czasowej).
- W celu poprawnego dobrania parametrów energii zgodności konieczna jest wiedza o wzajemnej relacji jasności tła i obiektu.
- Przy użyciu modelu LB trudnym wydaje się znalezienie maski obiektu, który jest zarówno jaśniejszy jak i ciemniejszy od tła.

Wyjaśnienia wymaga również zaobserwowane zjawisko, że w przypadku analizowanych sekwencji, algorytm relaksacji deterministycznej potrzebuje znacznie więcej iteracji

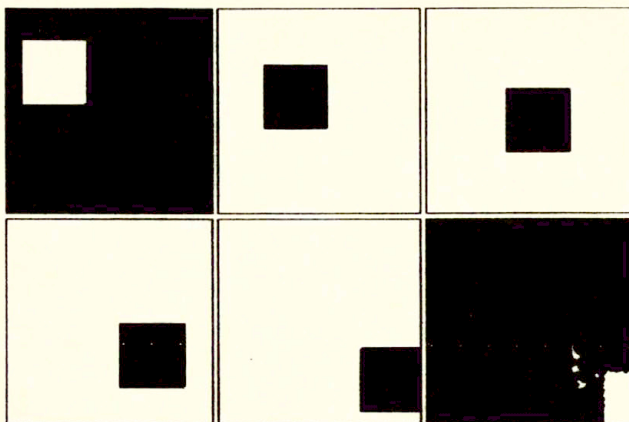
do osiągnięcia minimum energii przy przeglądaniu piksli po kodingach, niż przy przemiataniu normalnym (kolumna po kolumnie, wiersz po wierszu). Falszywe etykiety a przypisane podczas predykcji są usuwane w czasie korekcji dzięki energii zgodności. Z rozważań przeprowadzonych poprzednio wynika, że ma to miejsce szczególnie w sytuacjach przestrzennie obojętnych, czyli krótko mówiąc na brzegach maski obiektu. W implementacji z normalnym przemiataniem piksli po wyrzuceniu piksła, który znajdował się w sytuacji przestrzennie obojętnej, sąsiad najprawdopodobniej będzie znajdował się też w takiej sytuacji, czyli będzie kandydatem do wyrzucenia. Można powiedzieć, że następuje propagacja sytuacji przestrzennie obojętnej. W implementacji kodingowej niestety tego nie ma, a więc dochodzenie do minimum potrafi trwać nawet do 21 iteracji.

Zaprezentowana powyżej seria eksperymentów zdaje się potwierdzać wniosek sformułowany w trakcie rozważań teoretycznych przedstawionych w rozdziale 5.3, że przypisanie w definicji energii zgodności jednego parametru realizacjom (a, b) oraz (b, a) powoduje wydłużenie czasu obliczeń, gdyż podczas predykcji przypisywanych jest wiele „falszywych” etykiet, które następnie mozolnie muszą być usuwane podczas korekcji.

Rozdział 8

Badania eksperymentalne nowego modelu z dwuwymiarowym wektorem obserwacji

W niniejszym rozdziale przedstawiono wyniki badań eksperymentalnych nowego modelu z dwuwymiarowym wektorem obserwacji. Eksperymenty przeprowadzono w sposób analogiczny do tych z rozdziału 7. Podstawowym celem przeprowadzonych badań była ocena użyteczności nowego modelu z dwuwymiarowym wektorem obserwacji do detekcji masek obiektów ruchomych. Wykorzystanie identycznych sekwencji wejściowych jak w przypadku modelu LB pozwoliło na porównanie wyników otrzymywanych przy użyciu obydwu modeli.



Rys. 8.1: Maski obiektu ruchomego znalezione dla sekwencji K – nowy model z dwuwymiarowym wektorem obserwacji, relaksacja deterministyczna. Parametry: $m_1 = \hat{m}_1 = -27.73$ (estymowane), $m_2 = \hat{m}_2 = 32.27$ (estymowane), $m_3 = \hat{m}_3 = -10$ (estymowane), $\sigma^2 = 225$. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.

8.1 Badania eksperymentalne – sekwencje K,L,M,N

Dla zaproponowanego w niniejszej rozprawie nowego modelu z dwuwymiarowym wektorem obserwacji przeprowadzono analogiczne serie eksperymentów jak dla modelu LB. Ponownie testom poddano sekwencje K, L, M, N. Wyestymowane dokładne wartości parametrów \hat{m}_1 , \hat{m}_2 oraz \hat{m}_3 przedstawiono w tab. 6.2. Również i w przypadku tych eksperymentów wykorzystano idealne maski zmian w obrazie przedstawione na rys. 7.1, a korekcję maski pierwszej inicjowano idealną maską przedstawioną na rys. 7.2.

Podobnie jak dla modelu LB testowanie każdej sekwencji przeprowadzono w kilku etapach. Po pierwsze znaleziono maski przy estymowanych wartościach parametrów. Każdy następny etap polegał na zmianie jednego wybranego parametru w szerokim zakresie, gdy pozostałe parametry przyjmowały wartości estymowane. Etapy miały na celu zbadanie wrażliwości modelu na niepoprawną estymację poszczególnych parametrów.

Wyniki badań przeprowadzonych dla sekwencji K, L, M, N zebrano w odpowiednio w tab. 8.1–8.4. Każda z analizowanych sekwencji jest inna, co w pełni odzwierciedlają estymowane wartości parametrów \hat{m}_1 , \hat{m}_2 , \hat{m}_3 . Dzięki temu oraz dzięki faktowi, że model dokładniej opisuje analizowane zjawisko, eksperymenty przeprowadzone w przypadku każdej sekwencji, przy estymowanych wartościach parametrów, zakończyły się sukcesem. Oznacza to, że znalezione maski wiernie odwzorowują poruszający się obiekt, por. rys. 8.1, 8.7, 8.8, 8.9 lub odpowiednie wiersze tab. 8.1–8.4, co nie zawsze miało miejsce

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = \hat{m}_1 = -27.73, m_2^0 = \hat{m}_2 = 32.27, m_3^0 = \hat{m}_3 = -10, \sigma_0^2 = 225$								
	0	4	3	0	1	9	5/14	
m_1	$m_2 = m_2^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
-50	0	5	3	0	1	18	6/11	$m_1 < m_1^0$
-35	0	4	3	0	7	22	3/14	$m_1 < m_1^0$
-22	0	4	22	0	1	9	6/10	$m_1^0 < m_1 < m_3^0$
-12	0	4	3	0	151	12	19/18	$m_1^0 < m_1 \leq m_3^0$
-11	0	4	3	0	151	12	18/17	$m_1^0 < m_1 \leq m_3^0$
-9	0	4	3	0	1	21	24/11	$m_3^0 \leq m_1 < m_2^0$
0	0	4	3	0	1	21	4/10	$m_3^0 < m_1 < m_2^0$
20	0	4	3	0	1	3	3/3	$m_3^0 < m_1 < m_2^0$
27	0	4	3	0	1	3	3/3	$m_3^0 < m_1 \leq m_2^0$
50	0	7	3	0	4	1	10/10	$m_2^0 < m_1$
m_2	$m_1 = m_1^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
-50	48	1	24	0	19	9	5/9	$m_2 < m_1^0$
-30	46	4	23	0	29	9	5/9	$m_2 \leq m_1^0$
-11	0	6	3	0	7	9	10/17	$m_1^0 < m_2 \leq m_3^0$
-9	0	6	5	0	6	9	10/16	$m_3^0 \leq m_2 < m_2^0$
0	0	6	3	0	6	9	9/15	$m_1^0 < m_2 < m_3^0$
10	0	6	3	0	1	9	9/15	$m_3^0 < m_2 < m_2^0$
20	0	4	3	0	1	9	9/15	$m_3^0 < m_2 < m_2^0$
50	0	1	1	0	1	9	5/13	$m_2^0 \ll m_2$
m_3	$m_1 = m_1^0, m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
-50	0	4	3	0	1948	8	9/10	$m_3 < m_1^0$
-30	0	4	3	0	1852	25*	16/15	$m_3 \leq m_1^0$
-20	0	4	3	0	1	23	3/11	$m_1^0 < m_3 < m_3^0$
0	0	1	1	0	1	15	9/18	$m_3^0 < m_3 < m_2^0$
5	0	1	1	0	55	16	24/25*	$m_3^0 < m_3 < m_2^0$
15	0	1	1	0	901	24	25*/14	$m_3^0 < m_3 < m_2^0$
30	0	1	1	0	1204	5	25*/11	$m_3^0 < m_3 \leq m_2^0$
50	0	6	3	0	1898	16	18/18	$m_2^0 < m_3$
σ^2	$m_1 = m_1^0, m_2 = m_2^0, m_3 = m_3^0$							
22.5	0	3	2	5	6	20	3/13	$U_z \times 10$
2.25	1485	218	3	1514	257	5	3/4	$U_z \times 100$
2250	0	4	3	0	1	9	5/14	$U_z \times 0.1$

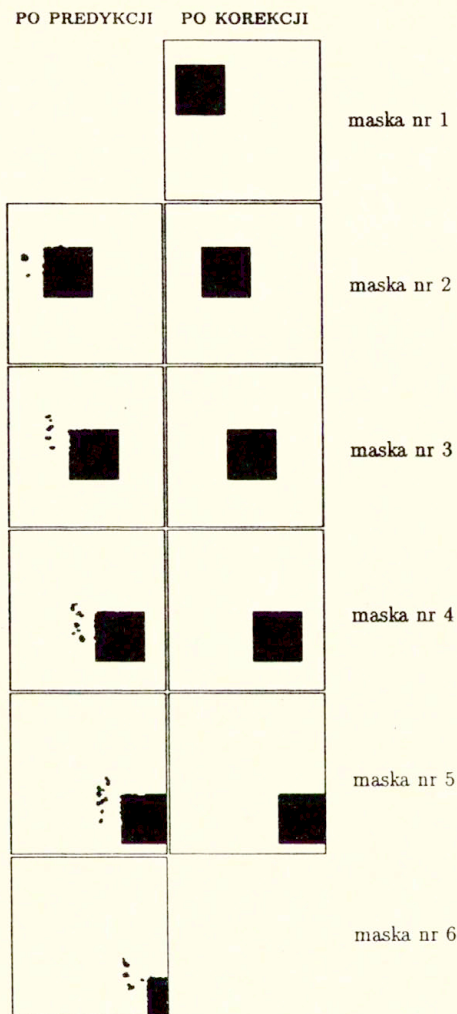
Tab. 8.1: Wpływ zmian parametrów nowego modelu z dwuwymiarowym wektorem obserwacji na wyniki detekcji masek obiektu ruchomego odpowiadających sekwencji K z rys. 6.1. Znaczenie poszczególnych kolumn jest identyczne jak dla tab. 7.1.

dla modelu LB.

Porównując znalezione maski dla sekwencji K dzięki zastosowaniu modeli LB i nowego, por. rys. 7.3 oraz 8.1, można zauważyć, że maski znalezione dzięki modelowi proponowanemu w niniejszej rozprawie są lepszej jakości – trzy z czterech narożników obiektu wykryto poprawnie, co nigdy nie miało miejsca dla modelu LB. Ponadto rys. 8.2, 8.3, 8.4, 8.5, które są analogami rys. 7.4, 7.5, 7.6, 7.7, również świadczą o przewadze proponowanego modelu. Podczas predykcji wykrywanych jest mniej „falszywych” etykiet a . Przyglądając się uważnie rys. 8.5, 8.4 można dojść do wniosku, że w przypadku nowego modelu najistotniejsze zmiany poszukiwanej maski dokonywane są w pierwszych dwóch lub trzech iteracjach w przypadku predykcji oraz pierwszej iteracji w przypadku korekcji, a to z kolei oznacza, że można ograniczyć liczbę iteracji algorytmu relaksacji deterministycznej, co nieznacznie tylko pogorszy znajdowaną maskę obiektu.

Inną cechą przemawiającą na korzyść zaproponowanego modelu może być czas obliczeń. Na rys. 8.6 w sposób analogiczny do rys. 7.8 przedstawiono długości poszczególnych kroków metody poszukiwania maski ruchomego obiektu dla sekwencji K, przy wyestymowanych wartościach parametrów. Kryterium stopu było identyczne w przypadku obydwu modeli. Metoda korzystająca z nowego modelu z dwuwymiarowym wektorem obserwacji wymagała jedynie 56 iteracji do zakończenia detekcji ruchu dla całej sekwencji, tzn. o 35 procent mniej niż przy użyciu modelu LB. Ponieważ jednak w przypadku nowego modelu wymagana jest większa liczba operacji dla obliczenia wartości energii lokalnej U_{ij} , więc w efekcie czas obliczeń był tylko o około 10 procent krótszy niż dla modelu LB. Liczba iteracji pierwszego kroku metody jest większa dla nowego modelu niż dla modelu LB. Jednak o ogólnej liczbie iteracji wymaganych dla przeprowadzenia detekcji ruchu dla całej sekwencji decydują przede wszystkim następane kroki, które są krótsze niż analogiczne kroki dla modelu LB. Analiza kolumn opisujących liczbę iteracji dla maski pierwszej i drugiej przedstawionych w tab. 8.1–8.4 oraz porównanie ich z odpowiednimi kolumnami z tab. 7.1–7.4 prowadzą do wniosku, że w najgorszym przypadku liczba iteracji dla obydwu modeli jest co najwyżej równa, natomiast w ogromnej większości przypadków jest prawie o 50 procent mniejsza na korzyść modelu nowego. Przyczyną tak znacznego zmniejszenia liczby wymaganych iteracji jest znacznie lepsza predykcja maski. W modelu LB w trakcie predykcji praktycznie jest otrzymywana maska zmian, co jest efektem dużej wartości energii przypisanej realizacji (a, b) . W nowym modelu takie zjawisko praktycznie nie występuje, por. rys. 8.4, 7.6. Ze względu na szum pewnym grupom pikseli tła przyporządkowywane są fałszywe etykiety a , ale są one o wiele mniej liczne niż dla modelu LB. Ponieważ w przypadku nowego modelu przypisywanych jest podczas predykcji znacznie mniej błędnych etykiet, dlatego ich usuwanie podczas korekcji wymaga mniejszej liczby iteracji.

Ocenę wrażliwości nowego modelu na zmianę wartości parametrów można przeprowadzić po przeanalizowaniu tab. 8.1–8.4. W przypadku parametru m_1 największe kłopoty pojawiają się gdy $m_1 \approx \hat{m}_3$. Dla sekwencji K, dla której estymowane wartości parametrów spełniają warunek $\hat{m}_1 < \hat{m}_3$, najmniejszą dokładność detekcji otrzymuje się gdy $m_1 \leq \hat{m}_3$. Również liczba iteracji potrzebna do osiągnięcia minimum energii ma jedną z największych wartości gdy $m_1 \approx \hat{m}_3$. W przypadku sekwencji L oraz M sytuacja jest bardzo podobna, z tą różnicą, że dla L estymowane parametry spełniają warunek $\hat{m}_1 > \hat{m}_3$ i w takiej sytuacji najgorszy wynik detekcji otrzymywany jest gdy $m_1 \geq \hat{m}_3$. Wartość parametru m_1 ma



Rys. 8.2: Porównanie masek obiektu znalezionych jedynie po predykcji (lewo) a następnie poprawionych w trakcie korekcji (pravo) dla sekwencji K, nowy model z dwuwymiarowym wektorem obserwacji. Wartości parametrów jak dla rys. 8.1 (estymowane). Maską nr 1 poddana została tylko korekcji, a maska nr 6 jedynie predykcji.

	MASKA 1			MASKA 2				UWAGI
	-a	-b	k ₁	-a	-b	p ₂	k ₂ /p ₃	
$m_1^0 = \hat{m}_1 = 32.27, m_2^0 = \hat{m}_2 = -27.73, m_3^0 = \hat{m}_3 = -10, \sigma_0^2 = 225$								
	0	7	4	0	1	8	2/4	
m_1	$m_2 = m_2^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
-50	0	10	4	0	6	2	12/12	$m_1 \ll m_2^0$
-30	0	10	4	0	3	5	8/8	$m_1 < m_2^0$
-12	0	7	4	0	1	17	21/12	$m_2^0 < m_1 < m_3^0$
-11	0	7	4	0	1	17	21/12	$m_2^0 < m_1 < m_3^0$
-9	0	7	4	0	85	15	25*/25*	$m_3^0 \leq m_1 < m_1^0$
0	0	7	4	0	1	9	9/10	$m_2^0 < m_1 < m_1^0$
20	0	7	4	0	1	3	2/4	$m_3^0 \ll m_1 < m_1^0$
27	0	7	4	0	1	3	2/4	$m_3^0 \ll m_1 < m_1^0$
30	0	7	4	0	1	3	2/4	$m_3^0 \ll m_1 < m_1^0$
50	0	7	4	0	1	14	3/6	$m_1^0 < m_1$
m_2	$m_1 = m_1^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
-50	0	1	1	20	1	8	2/4	$m_2 \ll m_2^0$
-25	0	7	4	0	1	8	2/4	$m_2^0 < m_2 < m_3^0$
-11	0	13	6	0	1	8	2/4	$m_2^0 < m_2 \leq m_3$
-9	0	4	2	0	10	8	4/4	$m_3^0 \leq m_2 < m_1^0$
0	0	3	2	0	8	8	3/4	$m_3^0 < m_2 < m_1^0$
20	0	3	2	0	6	8	4/4	$m_3^0 < m_2 < m_1^0$
30	0	3	2	0	6	8	4/4	$m_1^0 < m_2$
50	0	3	2	0	1	8	2/4	$m_1^0 \ll m_2$
m_3	$m_1 = m_1^0, m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
-50	0	201	25*	0	129	12	25*/25*	$m_3 \ll m_2^0$
-30	0	9	5	0	14	5	5/4	$m_2^0 \leq m_3 < m_3^0$
-20	0	1	1	0	1	3	2/4	$m_2^0 < m_3 < m_3^0$
0	0	8	4	0	1	17	2/7	$m_2^0 < m_3 < m_1^0$
5	0	9	5	0	1	15	2/7	$m_2^0 < m_3 < m_1^0$
15	0	14	7	0	1	10	4/13	$m_2^0 \ll m_3 < m_1^0$
30	0	14	7	0	1	9	6/14	$m_2^0 < m_3 \leq m_1^0$
50	0	14	7	0	1993	18	6/10	$m_1^0 < m_3$
σ^2	$m_1 = m_1^0, m_2 = m_2^0, m_3 = m_3^0$							
22.5	0	7	4	0	5	3	2/5	$U_z \times 10$
2.25	1987	236	7	1999	315	3	5/5	$U_z \times 100$
2250	0	7	4	0	1	8	2/4	$U_z \times 0.1$

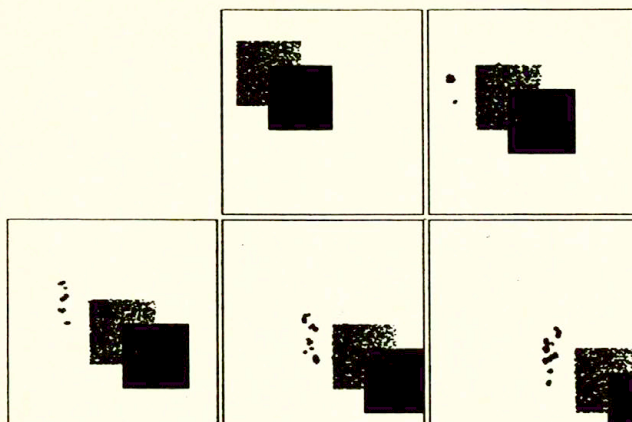
Tab. 8.2: Wpływ zmian parametrów nowego modelu z dwuwymiarowym wektorem obserwacji na wyniki detekcji masek obiektu ruchomego odpowiadających sekwencji L z rys. 6.3. Znaczenie poszczególnych kolumn jest identyczne jak dla tab. 7.1.

	MASKA 1			MASKA 2				UWAGI
	-a	-b	k ₁	-a	-b	p ₂	k ₂ /p ₃	
$m_1^0 = \hat{m}_1 = -32.27, m_2^0 = \hat{m}_2 = 27.73, m_3^0 = \hat{m}_3 = 10, \sigma_0^2 = 225$								
	0	3	2	0	1	8	1/3	
m_1	$m_2 = m_2^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
-50	0	3	2	0	1	8	2/8	$m_1 < m_1^0$
-30	0	3	2	0	1	8	3/4	$m_1^0 < m_1 < m_3^0$
0	0	3	2	0	1	8	19/19	$m_1^0 < m_1 < m_3^0$
9	0	3	2	0	186	19	25*/25*	$m_1^0 < m_1 \leq m_3^0$
10	0	3	2	0	1412	2	25*/25*	$m_1^0 < m_1 = m_3^0$
11	0	3	2	0	1	16	11/15	$m_3^0 \leq m_1 < m_2^0$
20	0	3	2	0	1	15	8/24	$m_3^0 < m_1 < m_2^0$
25	0	3	2	0	1	10	6/8	$m_3^0 < m_1 \leq m_2^0$
30	0	6	2	0	4	10	6/7	$m_2^0 \leq m_1$
50	0	6	2	0	4	2	11/11	$m_2^0 \ll m_1$
m_2	$m_1 = m_1^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
-50	0	1	1	0	1	8	1/3	$m_2 \ll m_1^0$
-30	0	1	1	0	3	8	3/3	$m_1^0 \leq m_2 \ll m_3^0$
0	0	1	1	0	3	8	2/3	$m_1^0 < m_2 < m_3^0$
9	0	1	1	0	3	8	2/3	$m_1^0 \ll m_2 \leq m_3^0$
11	0	3	2	0	1	8	1/3	$m_3^0 \leq m_2 \ll m_2^0$
20	0	3	2	0	1	8	1/3	$m_3^0 < m_2 < m_2^0$
25	0	3	2	0	1	8	1/3	$m_3^0 \ll m_2 \leq m_2^0$
50	0	3	2	0	1	8	1/3	$m_2^0 \ll m_2$
m_3	$m_1 = m_1^0, m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
-50	0	18	6	0	1972	15	5/10	$m_3 \ll m_1^0$
-30	0	12	5	0	12	15	9/16	$m_1^0 \leq m_3 \ll m_3^0$
-5	0	3	2	0	1	8	2/12	$m_1^0 < m_3 < m_3^0$
0	0	3	2	0	1	7	2/8	$m_1^0 < m_3 < m_3^0$
15	0	3	2	0	1	8	3/4	$m_3^0 < m_3 < m_2^0$
30	0	17	5	0	10	5	6/6	$m_2^0 \leq m_3$
50	0	32	9	0	289	15	25*/25*	$m_2^0 \ll m_3$
σ^2	$m_1 = m_1^0, m_2 = m_2^0, m_3 = m_3^0$							
22.5	0	3	2	0	1	6	1/3	$U_z \times 10$
2.25	2052	244	7	2008	231	3	4/3	$U_z \times 100$
2250	0	3	2	0	1	8	1/3	$U_z \times 0.1$

Tab. 8.3: Wpływ zmian parametrów nowego modelu z dwuwymiarowym wektorem obserwacji na wyniki detekcji masek obiektu ruchomego odpowiadających sekwencji M z rys. 6.5. Znaczenie poszczególnych kolumn jest identyczne jak dla tab. 7.1.

	MASKA 1			MASKA 2				UWAGI
	$-a$	$-b$	k_1	$-a$	$-b$	p_2	k_2/p_3	
$m_1^0 = \hat{m}_1 = 4.55, m_2^0 = \hat{m}_2 = 4.55, m_3^0 = \hat{m}_3 = -20, \sigma_0^2 = 225$								
	0	4	3	0	3	15	2/9	
m_1	$m_2 = m_2^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
-40	0	141	16	0	199	7	24/18	$m_1 \ll m_3^0$
-22	0	140	16	1	1265	10	25*/15	$m_1 \leq m_3^0$
-18	0	4	3	0	1293	10	23/23	$m_3^0 \leq m_1 \ll m_2^0$
-5	0	4	3	0	59	9	6/9	$m_3^0 < m_1 < m_2^0$
10	0	4	3	0	3	9	2/8	$m_2^0 < m_1$
30	0	5	3	0	3	10	3/10	$m_2^0 \ll m_1$
m_2	$m_1 = m_1^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
-40	0	1	1	0	1	15	2/9	$m_2 < m_3^0$
-22	0	3	2	0	1	15	2/9	$m_2 \leq m_3^0$
-18	0	11	5	0	3	15	2/9	$m_3^0 \leq m_2$
-5	0	4	3	0	3	15	2/9	$m_3^0 < m_2 < m_1^0$
10	0	1	1	0	3	15	2/9	$m_1^0 < m_2$
30	0	1	1	0	3	15	2/9	$m_1^0 \ll m_2$
m_3	$m_1 = m_1^0, m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
-40	0	11	5	0	295	17	5/17	$m_3 < m_3^0$
-25	0	4	3	0	55	9	2/25*	$m_3 \leq m_3^0$
-15	0	1	1	0	3	9	2/8	$m_3^0 < m_3 \ll m_2^0$
0	0	2	1	0	3	9	3/12	$m_3^0 < m_3 < m_2^0$
2	0	2	1	0	3	8	3/12	$m_3^0 < m_3 \leq m_2^0$
10	0	1	1	0	1997	9	3/6	$m_2^0 < m_3$
30	0	1	1	0	2000	9	3/9	$m_2^0 \ll m_3$
σ^2	$m_1 = m_1^0, m_2 = m_2^0, m_3 = m_3^0$							
22.5	21	6	5	10	3	11	3/9	$U_z \times 10$
2.25	993	5	5	996	24	3	4/4	$U_z \times 100$
2250	0	4	3	0	3	15	2/9	$U_z \times 0.1$

Tab. 8.4: Wpływ zmian parametrów nowego modelu z dwuwymiarowym wektorem obserwacji na wyniki detekcji masek obiektu ruchomego odpowiadających sekwencji N z rys. 6.7. Znaczenie poszczególnych kolumn jest identyczne jak dla tab. 7.1.



Rys. 8.3: Maski inicjujące predykcję masek odpowiadających obrazom 2–6 sekwencji K, znalezione wg metody inicjacji przedstawionej w rozdziale 5.4, nowy model z dwuwymiarowym wektorem obserwacji, algorytm relaksacji deterministycznej. Wartości parametrów jak dla rys. 8.1 (estymowane).

duży wpływ na liczbę wymaganych iteracji do osiągnięcia minimum energii, por. kolumna k_2/p_3 tab. 8.1–8.4. O ile dobrana przypadkowo wartość parametru m_1 najczęściej nie zmienia znacząco jakości znajdowanych masek obiektu, o tyle liczba iteracji wymaganych do znalezienia rozwiązania optymalnego może się zmieniać nawet siedmiokrotnie.

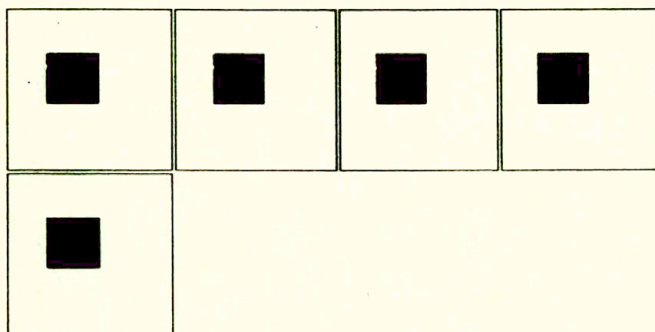
Wpływ parametru m_2 na dokładność znajdowanej maski jest podobny do wpływu parametru m_1 (jako dokładność maski rozumie tak jak poprzednio małe wartości statystyk $\neg a$ oraz $\neg b$). Dla sekwencji K stosunkowo najgorsze rezultaty otrzymuje się, gdy $m_2 < \hat{m}_1$ – pojawiają się nawet etykiety ruchu przypisane pikselom tła. Gdy $m_2 \approx \hat{m}_3$ w przypadku tej sekwencji zwiększa się nieco liczba wymaganych iteracji. Dla sekwencji L oraz M czas obliczeń właściwie jest niezależny od wartości parametru m_2 . Traktując problem ogólniej, właściwie w przypadku każdej z analizowanych sekwencji K, L, M, N dokładność znalezionych masek nie zależy od wartości parametru m_2 , gdy tylko m_1 oraz m_3 są dobrane poprawnie.

Najwrażliwszym wydaje się być proponowany model na wartość parametru m_3 i to praktycznie dla każdej z sekwencji testowych. Jak pokazały wyniki przeprowadzonych badań, wartość tego parametru może się zmieniać w niewielkim przedziale wokół wartości dokładnej.

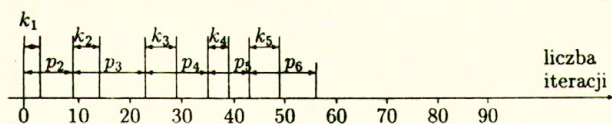
Parametr σ^2 odpowiadający za procentowy udział energii zgodności względem energii przestrzennej i czasowej ma praktycznie taki sam wpływ jak dla modelu LB. Dla żadnej z testowych sekwencji dziesięciokrotne zmniejszenie poziomu tej energii nie miało wpływu na dokładność przeprowadzanej detekcji, a stukrotne zwiększenie udziału tej ener-



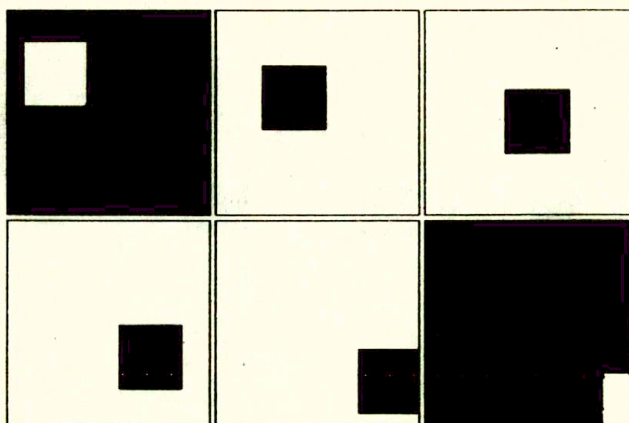
Rys. 8.4: Kolejne iteracje wyniku predykcji maski obiektu odpowiadającej trzeciemu obrazowi sekwencji K, nowy model z dwuwymiarowym wektorem obserwacji, algorytm relaksacji deterministycznej. Wartości parametrów jak dla rys. 8.1 (estymowane).



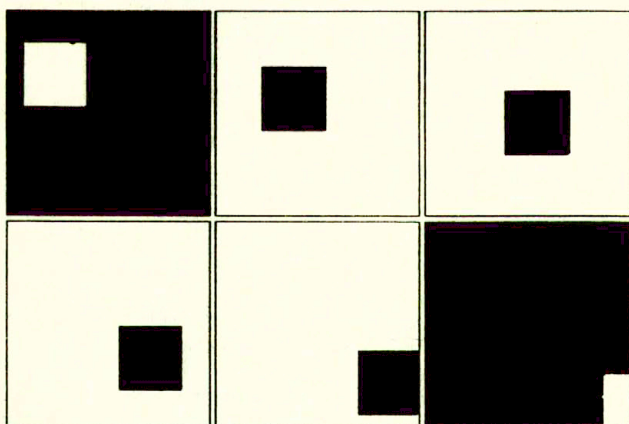
Rys. 8.5: Kolejne iteracje wyniku korekcji maski odpowiadającej drugiemu obrazowi sekwencji K, nowy model z dwuwymiarowym wektorem obserwacji, algorytm relaksacji deterministycznej. Wartości parametrów jak dla rys. 8.1 (estymowane).



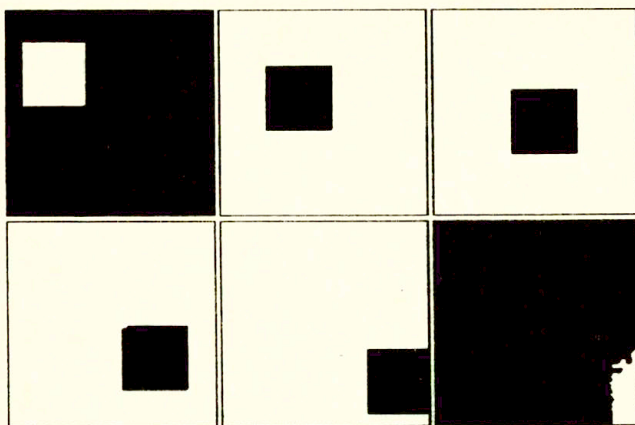
Rys. 8.6: Rysunek obrazujący czas obliczeń wyrażony za pomocą liczby iteracji algorytmu relaksacji deterministycznej, potrzebnych do znalezienia masek z rys. 8.1 – nowy model z dwuwymiarowym wektorem obserwacji. Symbole k_i oraz p_i odnoszą się odpowiednio do korekcji i predykcji maski odpowiadającej i -temu obrazowi sekwencji K. Łączny czas obliczeń wynosi 56 iteracji, który jest o 35 procent krótszy od czasu jaki wymagany jest przy korzystaniu z wersji podstawowej, por. rys. 7.8.



Rys. 8.7: Maski obiektu ruchomego z sekwencji L, nowy model z dwuwymiarowym wektorem obserwacji, algorytm relaksacji deterministycznej, $m_1 = \hat{m}_1 = 32.27$ (estymowane), $m_2 = \hat{m}_2 = -27.73$ (estymowane), $m_3 = \hat{m}_3 = -10$ (estymowane), $\sigma^2 = 225$. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.



Rys. 8.8: Maski obiektu ruchomego z sekwencji M, nowy model z dwuwymiarowym wektorem obserwacji, algorytm relaksacji deterministycznej, $m_1 = \hat{m}_1 = -32.27$ (estymowane), $m_2 = \hat{m}_2 = 27.73$ (estymowane), $m_3 = \hat{m}_3 = 10$ (estymowane), $\sigma^2 = 225$. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.



Rys. 8.9: Maski obiektu ruchomego z sekwencji N, nowy model z dwuwymiarowym wektorem obserwacji, algorytm relaksacji deterministycznej, $m_1 = \hat{m}_1 = 4.55$ (estymowane), $m_2 = \hat{m}_2 = 4.55$ (estymowane), $m_3 = \hat{m}_3 = -20$ (estymowane), $\sigma^2 = 225$. Maski pierwsza i ostatnia przedstawione są w negatywie ze względu na niekompletność procesu ich detekcji, pierwsza to efekt wyłącznie korekcji a ostatnia wyłącznie predykcji.

gii zawsze prowadzi do absurdów. Jednak podniesienie poziomu energii dziesięciokrotnie w przypadku sekwencji K oraz L nieznacznie pogorszyło wykrywaną maskę, a dla sekwencji M tak samo nieznacznie ją poprawiło.

O przewadze modelu zaproponowanego w niniejszej rozprawie w stosunku do modelu LB mogą również świadczyć znalezione maski obiektu, który jest zarówno ciemniejszy jak i jaśniejszy od tła (sekwencja N), por. rys. 8.9 oraz 7.15. Dzięki nowemu modelowi znalezione maski są pełniejsze, por. również tab. 8.4 i 7.4.

8.2 Porównanie nowego modelu z dwuwymiarowym wektorem obserwacji z modelem LB

Badania eksperymentalne, których wyniki przedstawiono w niniejszym rozdziale, pokazały, że zarówno model Lalande'a i Bouthemy'iego jak i nowy model z dwuwymiarowym wektorem obserwacji dobrze realizują cel, do którego zostały stworzone. W dużej liczbie przypadków umożliwiają one znalezienie masek obiektu o dobrej jakości, nawet przy dużym poziomie szumu.

Wyniki przeprowadzonych eksperymentów w pełni potwierdzają teoretyczne rozważania z rozdziału 5.3, które stały się podstawą do stworzenia nowego modelu z dwuwymiarowym wektorem obserwacji.

1. Model zaproponowany w niniejszej rozprawie łatwiej i dokładniej można dopasować do danych.
2. W konsekwencji można otrzymać lepsze maski inicjujące predykcję, tzn. o mniejszej liczbie fałszywych etykiet.
3. Dzięki lepiej przeprowadzonej predykcji zmniejsza się liczba iteracji algorytmu relaksacji deterministycznej wymaganych do zakończenia z sukcesem detekcji maski obiektu ruchomego.
4. Dzięki pełniejszemu dopasowaniu proponowanego modelu do analizowanej sekwencji wykrywane maski obiektu są lepszej jakości — trzy z czterech narożników kwadratu, których nigdy nie udało się wykryć przy pomocy modelu LB, są praktycznie zawsze znajdowane dzięki zastosowaniu nowego modelu.
5. Skutkiem dokładniejszego modelowania, gdy tylko dostępna jest pewna wiedza a priori o jasnościach poruszającego się obiektu, jest możliwość łatwiejszego przeprowadzenia identyfikacji parametrów nowego modelu.
6. Metoda detekcji ruchu korzystająca z nowego modelu lepiej nadaje się dla sekwencji, w których jasność obiektu jest zarówno powyżej jak i poniżej jasności tła.

Zalety nowego modelu wynikają przede wszystkim z dokładnego przeprowadzenia etapu jego budowy. Niewiele większa złożoność nowego modelu w stosunku do modelu LB ma pomijalne znaczenie, a korzyści, które można osiągnąć dzięki zastosowaniu tego modelu wydają się być istotne. Zastosowanie nowego modelu z dwuwymiarowym wektorem obserwacji zamiast modelu LB daje następujące korzyści:

1. Estymacja parametrów modelu staje się łatwiejsza.
2. Czas obliczeń ulega skróceniu.
3. Zwiększa się dokładność znajdowanych masek obiektu, dzięki umożliwieniu znajdowania narożników należących do obiektu ruchomego.

Dla porównania w dodatku zamieszczono wyniki podobnej serii eksperymentów dla modelu LB oraz nowego modelu z dwuwymiarowym wektorem obserwacji gdy jako obserwację przyjęto wartość bezwzględną różnicy jasności kolejnych obrazów. Ogólnie rzecz ujmując moduł różnicy jasności jest gorszym źródłem obserwacji niż sama różnica. Wartości poszczególnych parametrów bardzo się do siebie zbliżają, przez co dobór poprawnych wartości jest utrudniony.

Rozdział 9

Przykładowe wykorzystanie pól Markowa w detekcji i śledzeniu ruchu

W niniejszym rozdziale przedstawiono przykłady zastosowania nowego modelu z dwuwymiarowym wektorem obserwacji wprowadzonego w rozdziale 5 do przeprowadzenia detekcji i śledzenia ruchu w przypadku rzeczywistych sekwencji obrazów. Szczegółowo omówiono zagadnienie doboru parametrów nowego modelu. Przedstawiono także sumacyjną metodę detekcji masek obiektu ruchomego.

9.1 Detekcja i śledzenie ruchu w przypadku sekwencji rzeczywistych

Detekcja ruchu w przypadku sekwencji sztucznych wymaga jedynie przeprowadzenia dwóch etapów, tzn. znalezienia masek zmian w obrazie oraz przeprowadzenia właściwej detekcji ruchu, tj. detekcji masek obiektu ruchomego. Drugi z wymienionych etapów polega przede wszystkim na odrzuceniu części pikseli znalezionej wcześniej maski zmian tak, aby w przypadku konkretnej maski obiektu ruchomego pozostały tylko te piksele, które przedstawiają ten obiekt (suma logiczna masek obiektu ruchomego odpowiadających dwóm kolejnym obrazom analizowanej sekwencji jest maską zmian zachodzących pomiędzy tymi obrazami).

W przypadku sekwencji rzeczywistych liczba etapów musi jednak ulec zwiększeniu. Wyniki badań różnych detektorów zmian w obrazie, zamieszczone w rozdziale 4, pokazały bardzo duże zaszumienie masek znajdowanych przy pomocy różnych detektorów. Manipulując progami poszczególnych metod można zmniejszyć wpływ szumu, ale jednocześnie obszar przedstawiający zmiany spowodowane ruchem obiektów traci swoją spójność. Zarówno w przypadku modelu LB, jak i nowego modelu z dwuwymiarowym wektorem obserwacji, maska zmian w obrazie \bar{o}_k pełni bardzo ważną rolę, por. wartości energii czasowej U_{ij} dla obydwu modeli, tzn. równania (5.6) oraz (5.14). Ewentualne braki w tej masce bezwarunkowo będą miały odbicie w znajdowanej masce obiektu. Z tego powodu w detekcji zmian trzeba mieć na uwadze spójność obszarów przedstawiających ruch. Dlatego koniecznym staje się często przefiltrowanie znalezionej maski zmian w obrazie.

Ze względu na wykorzystanie we właściwej detekcji ruchu algorytmu relaksacji deterministycznej, który dochodzi jedynie do lokalnego minimum energii, niekiedy znalezione maski obiektu ruchomego mogą być w pewnym stopniu zaszumione. W takim przypadkach również one muszą zostać przefiltrowane.

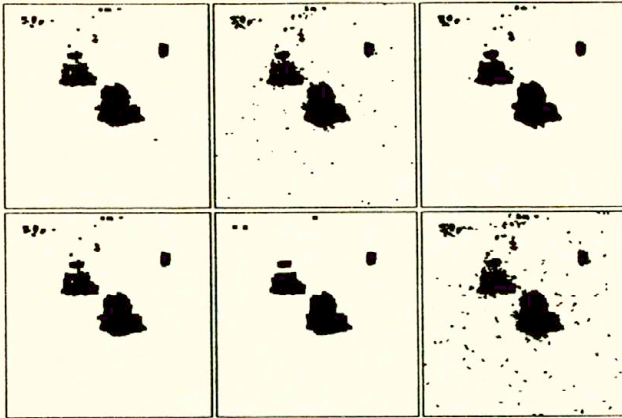
W konsekwencji detekcja i śledzenie ruchu dla sekwencji rzeczywistych wymagają przeprowadzenia następujących czynności:

1. detekcji zmian w obrazie,
2. filtracji znalezionych masek zmian,
3. właściwej detekcji ruchu, tj. detekcji masek obiektów ruchomych,
4. filtracji znalezionych masek obiektów ruchomych.

Detekcja zmian w obrazie została szczegółowo przedstawiona w rozdziałach 2.1. 4. Właściwej detekcji ruchu poświęcono rozdziały 5, 7, 8. Filtracje masek zmian w obrazie oraz masek obiektów ruchomych sprowadzają się do filtracji obrazów binarnych. Z tego powodu obydwaj etapy omówiono łącznie w rozdziale 9.2.

9.2 Filtracja masek

Zarówno filtracja masek zmian w obrazie jak i masek obiektów ruchomych przede wszystkim polega na usunięciu szumu, przy założeniu, że szum stanowią izolowane piksele.



Rys. 9.1: Maska zmian zachodzących pomiędzy obrazami 3 oraz 4 sekwencji A z rys. 4.1. Maskę tę najpierw znaleziono przy pomocy metody Hsu-Nagel-Rekers z kwadratowym modelem funkcji jasności (próg=50), por. rys. 4.6. Następnie została ona przefiltrowana metodami: kontur-wnętrze (1,1), test mediany – okno 3×3 (1,2), test mediany – okno 5×5 (1,3), , filtr morfologiczny typu otwarcie-domknięcie z elementem strukturującym 3×3 (2,1), filtr morfologiczny typu otwarcie-domknięcie z elementem strukturującym 5×5 (2,2), filtr morfologiczny typu otwarcie-domknięcie z czterema elementami strukturującymi (2,3). Numeracja zdjęć: (1, 1) (1, 2) (1, 3)
(2, 1) (2, 2) (2, 3)

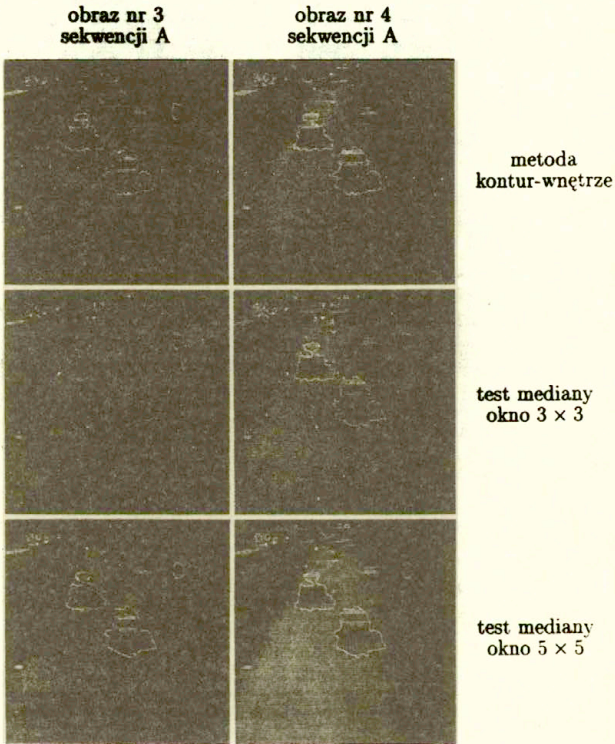
Ponadto pożądaną cechą metody filtracji jest uzupełnianie niewielkich lokalnych braków masek.

W przeprowadzonych badaniach wykorzystano trzy grupy metod:

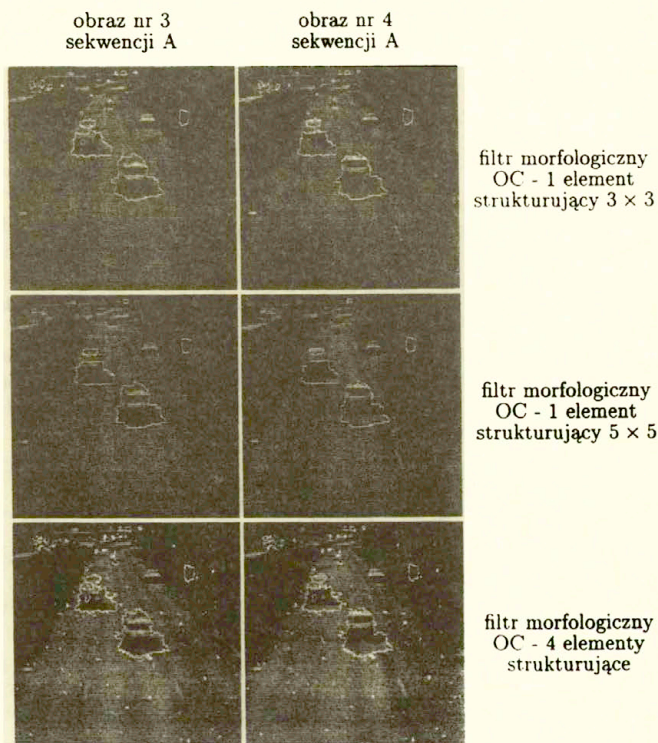
- prostą metodę *kontur-wnętrze*,
- filtry medianowe o różnej wielkości okna testowego.
- filtrację morfologiczną o jednym lub kilku elementach strukturujących [Nie92, Nie94].

Metoda kontur-wnętrze polega na usuwaniu pikseli, które ani nie należą do wnętrza, ani do konturu, przy czym przyjmuje się, że piksel należy do wnętrza, gdy jest zaetykietowany identycznie jak ośmiu jego sąsiadów, natomiast pikselem konturu jest ten, który ma w swoim otoczeniu zarówno piksele należące do tła jak i do wnętrza. Przy użyciu tej metody niewielkie braki maski nigdy nie będą wypełniane, a ponadto wszystkie „wąsy” o grubości 1–2 pikseli zostaną usunięte.

W algorytmach filtracji medianowej dla obrazów binarnych danemu pikselowi przypisuje się taką etykietę, jaką ma większość z jego sąsiadów. W zależności od wielkości okna testowego otrzymuje się różne efekty filtracji. Skorzystanie z tej metody może doprowadzić do usunięcia niewielkich braków maski.



Rys. 9.2: Wyniki filtracji maski zmian (HNRkw próg=50) zachodzących pomiędzy obrazami 3 oraz 4 sekwencji A, po nałożeniu na te obrazy. Efekt filtracji metodami: kontur-wnętrze (wiersz nr 1), test mediany - okno 3 x 3 (wiersz nr 2), test mediany - okno 5 x 5 (wiersz nr 3).



Rys. 9.3: Wyniki filtracji maski zmian (HNRkw próg=50) zachodzących pomiędzy obrazami 3 oraz 4 sekwencji A, po nałożeniu na te obrazy. Efekt filtracji metodami: filtr morfologiczny typu otwarcie-domknięcie (OC) z elementem strukturującym 3 × 3 (wiersz nr 1), filtr morfologiczny typu otwarcie-domknięcie (OC) z elementem strukturującym 5 × 5 (wiersz nr 2), filtr morfologiczny typu otwarcie-domknięcie (OC) z czterema elementami strukturującymi (wiersz nr 3).

relacja pomiędzy jasnością obiektu i tła	znak m_1	znak m_2
obiekt średnio jaśniejszy od tła	-	+
obiekt średnio ciemniejszy od tła	+	-

Tab. 9.1: Zależność znaku parametrów m_1 oraz m_2 modelu zaproponowanego w niniejsze rozprawie do detekcji i śledzenie ruchu.

Stosując operacje morfologiczne otwarcia i domknięcia, oraz odpowiednio dobrane elementy strukturalne, można przeprowadzać różne rodzaje filtracji. W ogólności metody opierające się na morfologii matematycznej umożliwiają filtrację lepszej jakości niż mediana.

Wybrane metody zastosowano do poprawy jakości masek zmian z rys. 4.6 odpowiadających sekwencji A z rys. 4.1. Wyniki eksperymentów przedstawiono na rys. 9.1-9.4. Na rys. 9.1 pokazano maskę zmian zachodzących pomiędzy obrazami 3 oraz 4 sekwencji A, przefiltrowaną sześcioma różnymi metodami. Na rys. 9.2, 9.3 kontur wszystkich masek z rys. 9.1 nałożono na obrazy 3 oraz 4 sekwencji A.

Na podstawie przedstawionych wyników można stwierdzić, że najlepsze dla oka wygładzenie otrzymuje się za pomocą testu morfologicznego typu otwarcie-domknięcie (OC) oraz testu mediany z oknem 5×5 . Zupełnie dobre rezultaty można uzyskać również przy użyciu metody kontur-wnętrze.

9.3 Dobór parametrów pola Markowa

Detekcja ruchu przy wykorzystaniu sekwencji rzeczywistych oraz modelu zaproponowanego w niniejszej rozprawie wymaga przeprowadzenia identyfikacji tego modelu. Wartości większości parametrów dobrano w sposób heurystyczny. Za dopasowanie modelu do analizowanej sekwencji odpowiadają parametry: m_1 , m_2 , m_3 oraz σ^2 . Na podstawie wyryków badań zamieszczonych w rozdziałach 7 oraz 8 można stwierdzić, że najmniej kłopotów sprawia znalezienie wartości parametru σ^2 . Wartość jego nie ma praktycznie żadnego wpływu na efekt detekcji, jeśli tylko nie jest zbyt mała. Liczba 225 wydaje się być odpowiednia.

O wiele większe trudności mogą wystąpić w przypadku parametrów odpowiadających za sytuacje przejściowe, tzn. m_1 oraz m_2 . Wartości tych parametrów w sposób istotny zależą od relacji pomiędzy jasnością obiektu ruchomego i statycznego tła, a precyzyjniej tylko tej jego części, która jest zakrywana lub odkrywana przez poruszający się obiekt. Dysponując taką wiedzą a priori jak to, czy obiekt jest średnio jaśniejszy czy ciemniejszy od tła, bez kłopotów można określić znak obydwu parametrów, co przedstawiono w tab. 9.1

Załóżmy chwilowo, że ta część funkcji jasności, która odpowiada obiektowi ruchomemu, ma postać funkcji liniowej. W takim przypadku wartości parametrów m_1 oraz m_2 zależą zarówno od relacji pomiędzy kierunkiem gradientu funkcji jasności obiektu i kierunkiem ruchu oraz od zmiany położenia obiektu w sąsiednich obrazach sekwencji. W tab. 9.2, 9.3 przedstawiono wartości parametrów m_1 , m_2 dla kwadratu o liniowej funkcji jasności,

przes. pikseli	współczynnik nachylenia płaszczyzny jasności obiektu							
	$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{2}$	1	2	4	8
2	0	0	0	-0.5	-0.5	-1	-2	-4
5	0	-0.2	-0.6	-1.2	-2	-4	-8	-16.6
10	-0.2	-0.6	-1.2	-2.5	-4.5	-9	-18.2	-37.2
20	-0.6	-1.2	-2.5	-5	-9.5	-19	-38.6	-78.6
40	-1.2	-2.5	-5	-10	-19.5	-39.5	-79.3	-

Tab. 9.2: Zależność współczynnika m_1 nowego modelu z dwuwymiarowym wektorem obserwacji w przypadku sekwencji z poruszającym się kwadratem o liniowej funkcji jasności. Wartości parametru przedstawiono w funkcji wielkości przesunięcia tego obiektu wyrażonej w pikselach oraz tangensa kąta nachylenia funkcji jasności kwadratu. Kierunek ruchu jest zgodny z kierunkiem gradientu funkcji jasności. Minimalna jasność obiektu jest na poziomie tła.

którego jasność zmienia się od poziomu tła do pewnej wartości maksymalnej (obiekt jaśniejszy do tła), gdy kierunek ruchu jest zgodny z kierunkiem gradientu funkcji jasności. Wartości tych współczynników podano w zależności od współczynnika nachylenia funkcji jasności obiektu oraz jego przesunięcia pomiędzy sąsiednimi obrazami (wyrażonego w pikselach). Podane wartości obliczono poprzez uśrednienie różnicy o_k dla sytuacji (ruch,tło) i (tło,ruch).

Z tab. 9.2, 9.3 można również korzystać przy określaniu dokładnych parametrów dla innych sekwencji z poruszającym się kwadratem o liniowo zmiennej jasności. W takich przypadkach należy kierować się dodatkowo następującymi regułami:

- Gdy różnica pomiędzy minimalną jasnością obiektu i tłem jest różna od zera, wówczas parametr o dodatniej wartości należy zwiększyć o tę różnicę, a parametr mniejszy od zera odpowiednio zmniejszyć o taką samą wartość.
- Jeżeli kwadrat porusza się w kierunku przeciwnym do kierunku gradientu funkcji jasności, wówczas wartości parametrów m_1 oraz m_2 zamieniają się ze sobą.
- Zmiana relacji wzajemnej jasności tła i obiektu również powoduje zamianę wartości parametrów m_1 oraz m_2 .

Tab. 9.2, 9.3 mogą być również przydatne dla sekwencji rzeczywistych. Załóżmy, że obiekt jest średnio jaśniejszy od tła, jak np. w przypadku sekwencji A z rys. 4.1. Jeżeli lokalne przesunięcie obiektu nie jest zbyt duże (od kilku do kilkunastu pikseli), to można przyjąć, że jasność przedniej części obiektu, tzn. tej jego części, która przykrywa tło, ma lokalnie charakter funkcji liniowej, zazwyczaj o kierunku gradientu przeciwnym do kierunku ruchu. Funkcja jasności przedniej części obiektu decyduje o wartości parametru m_2 . Aby określić jego wartość można skorzystać z tab. 9.2, zmieniając oczywiście znak. Tylna część obiektu, tzn. ta która odkrywa tło, decydująca o wartości parametru m_1 , przy niewielkich przesunięciach też może być opisana funkcją liniową, ale o kierunku gradientu zgodnym z kierunkiem ruchu. Ponownie można skorzystać z tab. 9.2.

przes. pikseli	współczynnik nachylenia płaszczyzny jasności obiektu							
	$\frac{1}{16}$	$\frac{1}{8}$	$\frac{1}{4}$	$\frac{1}{2}$	1	2	4	8
2	5	10	20	39.5	79.5	159	238	236
5	5	9.8	19.4	38.8	78	156	232	223.4
10	4.8	9.4	18.8	37.5	75.5	151	221.8	202.8
20	4.4	8.8	17.5	35	70.5	141	201.4	161.4
40	3.8	7.5	15	30	60.5	120.5	160.7	-

Tab. 9.3: Zależność współczynnika m_2 nowego modelu w przypadku sekwencji z poruszającym się kwadratem o liniowej funkcji jasności. Wartości parametru przedstawiono w funkcji wielkości przesunięcia tego obiektu wyrażonej w pikselach oraz tangensa kąta nachylenia funkcji jasności kwadratu. Kierunek ruchu jest zgodny z kierunkiem gradientu funkcji jasności. Minimalna jasność obiektu jest na poziomie tła.

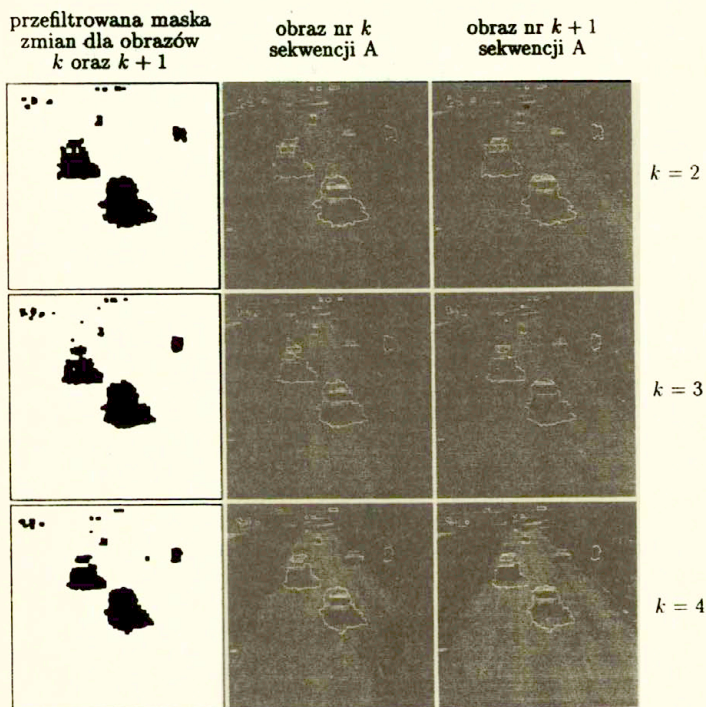
Określenie wartości parametru m_3 może sprawić o wiele więcej kłopotu. Praktycznie niemożliwym jest skorzystanie z aproksymacji funkcjami o znanej postaci środkowej części obiektu. Jednocześnie badania, których wyniki zamieszczono w rozdziale 8, pokazały wrażliwość modelu proponowanego w niniejszej rozprawie na wartość parametru m_3 . Analizując tab. 8.1–8.4 można zauważyć, że przyjmując $m_3 = 0$ otrzymano zawsze algorytm stabilny. Można więc wysnuć wniosek, że właściwa wartość m_3 znajduje się w okolicy zera. Precyzyjniejsze jej określenie wymaga jednak znajomości funkcji jasności obiektu ruchomego. Ważną cechą parametru m_3 jest to, że jego wartość nie zależy od relacji pomiędzy jasnością tła i obiektu, a tylko od funkcji jasności samego obiektu.

9.4 Wyniki eksperymentów

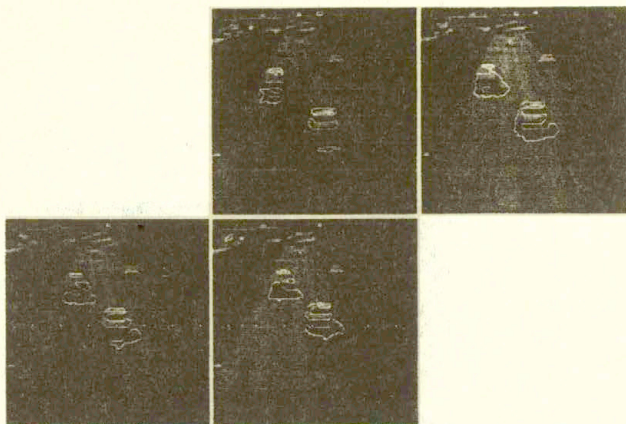
Aby przeprowadzić detekcję ruchu dla sekwencji rzeczywistych koniecznym jest znalezienie maski inicjującej korekcję maski obiektu ruchomego odpowiadającej pierwszemu obrazowi analizowanej sekwencji. Nie dysponując pełną wiedzą a priori, nie można praktycznie takiej maski znaleźć. Korekcja maski dla pierwszego obrazu może jednak zostać zainicjowana maską zmian zachodzących pomiędzy obrazami pierwszym i drugim. We wszystkich eksperymentach, których rezultaty są prezentowane w niniejszym rozdziale, zastosowano taki właśnie sposób inicjacji korekcji maski odpowiadającej pierwszemu obrazowi sekwencji.

Po przeprowadzeniu detekcji dla całej sekwencji wejściowej, maski obiektu odpowiadające pierwszemu i ostatniemu obrazowi powinny zostać odrzucone, ze względu na niepełny proces ich otrzymywania, gdyż maska nr 1 poddawana jest jedynie korekcji, a maska ostatnia tylko predykcji. Z tego powodu przy prezentacji wyników detekcji ruchu zrezygnowano z pokazywania masek obiektu odpowiadających pierwszemu oraz szóstemu obrazowi poszczególnych sekwencji.

Na rys. 9.5 pokazano sekwencję masek obiektu znalezionych przy pomocy modelu proponowanego w niniejszej rozprawie, a odpowiadających obrazom 2-5 sekwencji A. Na



Rys. 9.4: Maski zmian zachodzących między obrazami (2,3), (3,4) oraz (4,5) sekwencji A. znalezione przy użyciu metody HNRkw (próg=50), następnie poprawione filtrem morfologicznym typu otwarcie-domknięcie z elementem strukturującym 3×3 . W pierwszej kolumnie pokazano kolejne maski zmian, a w kolumnach drugiej oraz trzeciej ich porównanie z odpowiednimi obrazami sekwencji A.

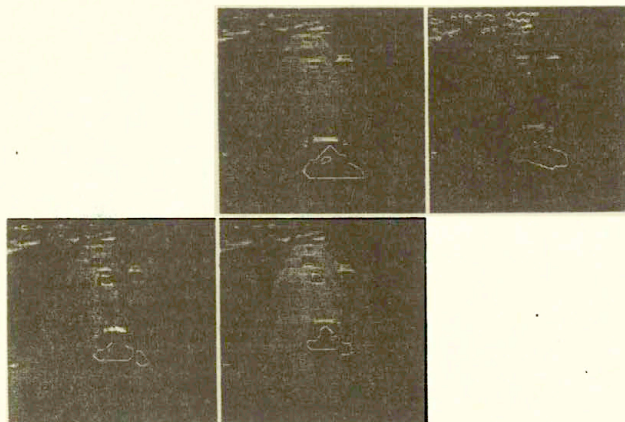


Rys. 9.5: Maski obiektu znalezione dla obrazów 2–5 sekwencji A (nowy model z dwuwymiarowym wektorem obserwacji, parametry jak dla sekwencji K) nałożone na odpowiednie obrazy sekwencji A. Wykorzystano maski zmian z rys. 9.4.

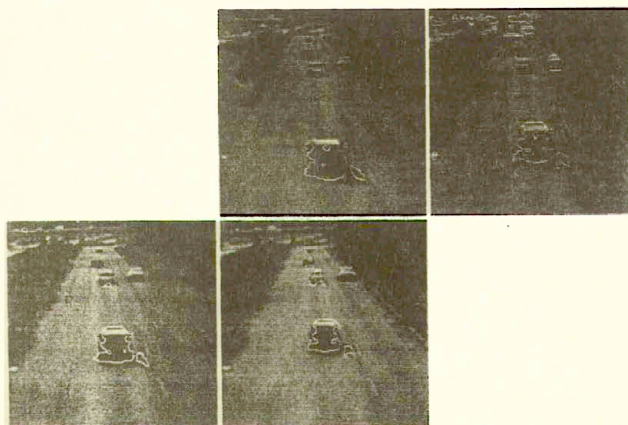
rys. 9.4 przedstawiono maski zmian, które zostały użyte do detekcji ruchu z rys. 9.5. Parametry pola losowego zostały tak dobrane, aby wykryć obiekty jaśniejsze od tła. Z tego powodu najprawdopodobniej maski samochodu znajdującego się na pierwszym planie nie są spójne. Poza samochodem wykryty został również jego cień. W przypadku samochodu z drugiego planu właściwie wykryta została tylko tylna część samochodu. Przyczyną zapewne jest bardzo mały kontrast pomiędzy jaśniejszą częścią tego samochodu a tłem. Nieruchomy samochód z trzeciego planu, w sposób zgodny ze stanem faktycznym, został zaliczony do statycznego tła. Poruszające się samochody na dalszych planach praktycznie w ogóle nie zostały znalezione, najprawdopodobniej ze względu na ich niewielkie rozmiary (wyrażone w pikselach).

Na rys. 9.6 oraz 9.7 przedstawiono efekt detekcji ruchu dla sekwencji B, przy czym na rys. 9.6 pokazano maski znalezione, gdy wartości parametrów energii zgodności były takie jak dla sekwencji L (obiekt ciemniejszy od tła), a na rys. 9.7, gdy wartości parametrów były takie same jak w przypadku sekwencji N (poruszający się obiekt zarówno ciemniejszy jak i jaśniejszy od tła). Na pierwszy rzut oka wydawać by się mogło, że samochód jest ciemniejszy od tła. Jednak dach tego samochodu, z powodu oświetlenia promieniami słonecznymi, jest jaśniejszy niż jezdnia. Z tego powodu maski z rys. 9.7 są lepszej jakości.

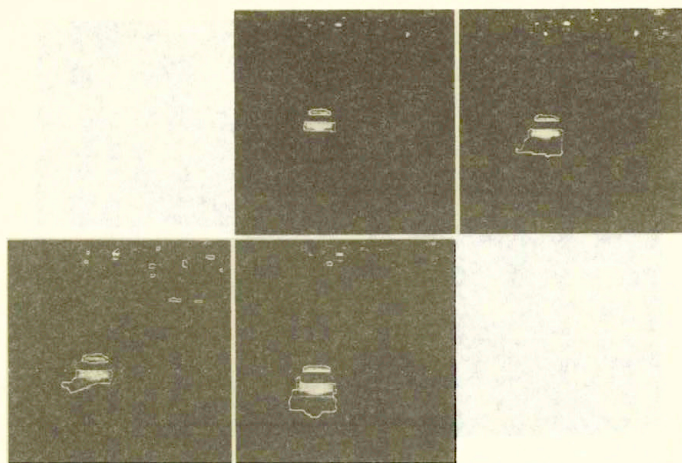
Na rys. 9.8 przedstawiono efekt detekcji ruchu przeprowadzonej dla sekwencji C. Wartości parametrów przyjęto takie, jak dla sekwencji K. Ponownie detekcja ruchu zakończona została sukcesem. W przypadku większości masek, poza poruszającym się samochodem, został znaleziony również jego cień.



Rys. 9.6: Maski obiektu znalezione dla obrazów 2-5 sekwencji B (nowy model z dwuwymiarowym wektorem obserwacji, parametry jak dla sekwencji L (obiekt jest ciemniejszy od tła), nałożone na odpowiednie obrazy sekwencji B. Skorzystano z masek zmian otrzymanych przy pomocy metody HNRkw (próg=100), które zostały następnie przefiltrowane metodą kontur-wnętrze.



Rys. 9.7: Maski obiektu znalezione dla obrazów 2-5 sekwencji B (nowy model z dwuwymiarowym wektorem obserwacji, parametry jak dla sekwencji N (obiekt jest zarówno ciemniejszy jak i jaśniejszy od tła), nałożone na odpowiednie obrazy sekwencji B. Skorzystano z masek zmian otrzymanych dzięki metodzie HNRkw (próg=100), które następnie przefiltrowano metodą kontur-wnętrze.

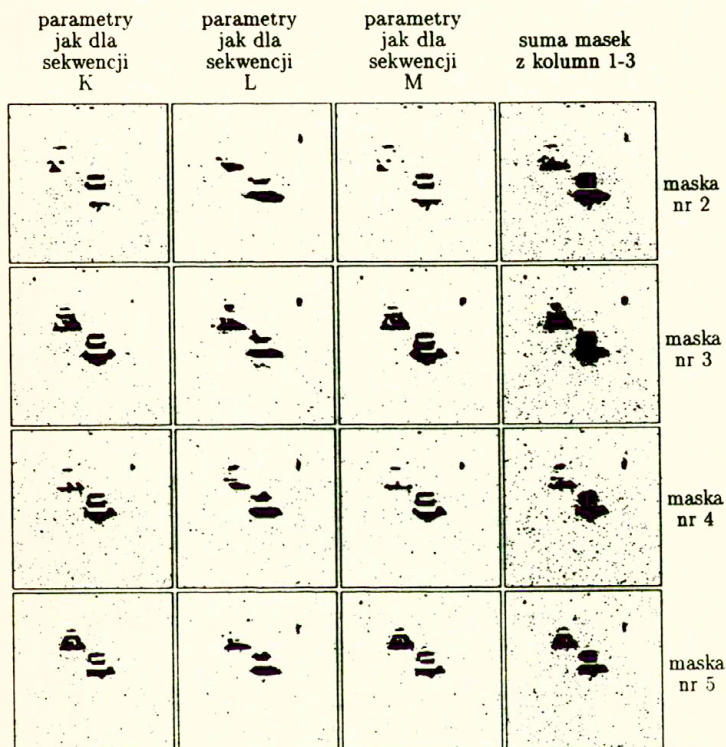


Rys. 9.8: Maski obiektu znalezione dla obrazów 2-5 sekwencji C (nowy model, parametry jak dla sekwencji K), nałożone na odpowiednie obrazy sekwencji C. Skorzystano z masek zmian otrzymanych dzięki metodzie HNRkw (próg=50), które następnie poprawiono filtrem morfologicznym typu otwarcie-domknięcie z elementem strukturującym 3×3 .

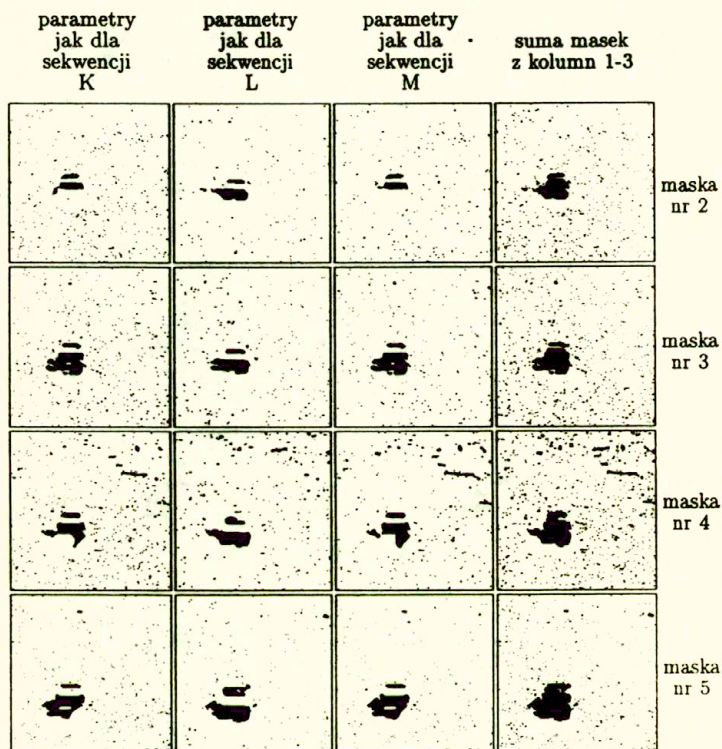
9.5 Sumacyjna metoda detekcji maski obiektu ruchomego

Estymacja parametrów modelu zaproponowanego w niniejszej rozprawie w przypadku sekwencji rzeczywistych sprawia pewną trudność. Pewnym sposobem rozwiązanie tego problemu może być *sumacyjna metoda detekcji maski obiektu ruchomego*, będąca konsekwencją następującego toku rozumowania. W przedstawionych w rozdziale 5 modelach korzystających z pól Markowa, maski obiektu są znajdowane na podstawie masek zmian. W pewnym uproszczeniu można przyjąć, że w trakcie działania algorytmu usuwane są z maski zmian te piksele, które nie należą do obiektu w danej chwili czasu. Żle dobrane parametry energii zgodności mogą powodować jedynie wyrzucenie zbyt dużej liczby pikseli, między innymi tych, które w rzeczywistości należą do obiektu ruchomego. Przy różnych wartościach parametrów m_1 , m_2 , m_3 różne piksele obiektu mogą zostać usunięte. Przeprowadzając detekcję ruchu przy kilku zestawach parametrów, a następnie sumując znalezione maski, można otrzymać maskę obiektu ruchomego, która będzie niegorsza od masek składowych.

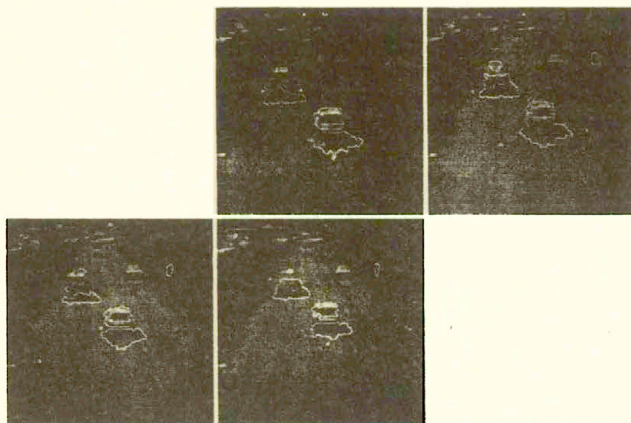
Wykorzystanie sumacyjnej metody detekcji maski obiektu ruchomego wymaga wybrania kilku zestawów parametrów m_1 - m_3 . Jedną z możliwości jest zastosowanie zestawów parametrów dla obiektu o liniowo zmiennej funkcji jasności różniących się relacją pomiędzy jasnością tła i obiektu oraz relacją pomiędzy kierunkiem gradientu funkcji jasności a kierunkiem ruchu.



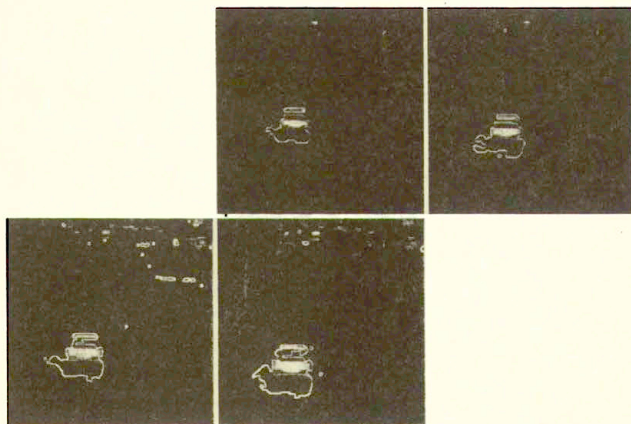
Rys. 9.9: Zastosowanie sumacyjnej metody detekcji maski obiektu ruchomego dla sekwencji A. W kolejnych wierszach przedstawiono maski obiektu odpowiadające obrazom 2-5 sekwencji A. W kolumnach 1-3 przedstawiono maski znalezione dzięki zastosowaniu modelu proponowanego w niniejszej rozprawie przy wartościach parametrów m_1 , m_2 , m_3 odpowiednio jak dla sekwencji K (kolumna pierwsza), L (kolumna druga), M (kolumna trzecia). W kolumnie czwartej przedstawiano maskę obiektu będącą sumą masek z kolumn 1-3.



Rys. 9.10: Zastosowanie sumacyjnej metody detekcji maski obiektu ruchomego dla sekwencji C. W kolejnych wierszach przedstawiono maski obiektu odpowiadające obrazom 2-5 sekwencji C. W kolumnach 1-3 przedstawiono maski znalezione dzięki zastosowaniu modelu proponowanego w niniejszej rozprawie przy wartościach parametrów m_1 , m_2 , m_3 odpowiednio jak dla sekwencji K (kolumna pierwsza), L (kolumna druga), M (kolumna trzecia). W kolumnie czwartej przedstawiano maskę obiektu będącą sumą masek z kolumn 1-3.



Rys. 9.11: Maski obiektu z rys. 9.9, znalezione przy pomocy metody sumacyjnej, przefiltrowane metodą kontur-wnętrze, a następnie nałożone na obrazy 2-5 sekwencji A.



Rys. 9.12: Maski obiektu z rys. 9.10, znalezione przy pomocy metody sumacyjnej, przefiltrowane metodą kontur-wnętrze, a następnie nałożone na obrazy 2-5 sekwencji C.

Przykładowe wykorzystanie zaprezentowanej metody w przypadku sekwencji A przedstawiono na rys. 9.9, 9.11, a przypadku sekwencji C na rys. 9.10, 9.12. Dla obydwu sekwencji przeprowadzono detekcję ruchu przy trzech zestawach parametrów m_1 , m_2 , m_3 . Wykorzystano estymowane wartości parametrów znalezione dla sekwencji K, L, M. Rys. 9.9, 9.10 przedstawiają maski obiektu odpowiadające obrazom 2-5 obydwu sekwencji, a rys. 9.11, 9.12 pokazują sumacyjną maskę obiektu, przefiltrowaną metodą kontur-wnętrze, a następnie nałożoną na odpowiednie obrazy sekwencji A oraz C.

Rozdział 10

Ukryte pole Markowa z trójwymiarowym wektorem obserwacji, przeznaczone do detekcji masek obiektów ruchomych

W rozdziale zaprezentowano model oparty na czasowo-przestrzennym polu Markowa, przeznaczony do detekcji masek obiektów ruchomych. Najważniejszą cechą proponowanego ukrytego pola Markowa jest zastosowanie w nim trójwymiarowego wektora obserwacji o składowych: etykieta maski zmian zachodzących pomiędzy dwoma kolejnymi obrazami sekwencji, różnica jasności przyporządkowanych pikselowi w dwóch kolejnych obrazach sekwencji oraz wartość jasności w pikselu. Proponowany model pozwala, w czasie detekcji maski obiektu ruchomego, na uzupełnianie ewentualnych niespójności masek zmian, mogących występować w przypadku pewnej grupy sekwencji rzeczywistych. W końcowej części rozdziału przedstawiono przykłady zastosowania nowego modelu.

10.1 Analiza ruchu obiektów o lokalnie stałej funkcji jasności

Zarówno w modelu LB, jak również w nowym modelu z dwuwymiarowym wektorem obserwacji zaproponowanym w rozdziale 5, maska zmian w obrazie odgrywa bardzo istotną rolę. Bezspornie świadczą o tym przyjęte w obydwu modelach wartości energii klik czasowych. Wyniki eksperymentów zamieszczone w rozdziale 4 pokazały, że detekcja spójnych masek zmian w obrazie sprawia ogromne trudności. Niespójność masek występuje przede wszystkim w tych obszarach, które w obydwu porównywanych obrazach sekwencji odpowiadają obiektowi ruchomemu, tzw. sytuacja (ruch, ruch), a ściślej gdy jasność obiektu w tych obszarach jest stała lub zbliżona do stałej. Z punktu widzenia komputerowej analizy ruchu sekwencje przedstawiające ruch takich obiektów są bardzo niewygodne. Przyczyną kłopotów jest praktycznie taka sama wartość lokalnej miary zmian, np. różnicy jasności, dla sytuacji (tło, tło) oraz (ruch, ruch). Przykłady „trudnych” sekwencji zaprezentowano na rys. 10.1 oraz 10.2.

W sekwencji D z rys. 10.1 przedstawiono powolny ruch obrotowy wokół osi pionowej głowy modelki z lewej strony. W trakcie detekcji zmian praktycznie występuje tylko sytuacja (ruch, ruch). Dodatkowo dużą część poruszającej się głowy stanowi twarz, która w obrazie komputerowym przedstawiona jest za pomocą pikseli o zbliżonych wartościach funkcji jasności.

Sekwencja E z rys. 10.2 przedstawia ruch o charakterze translacji. Cechą charakterystyczną poruszającego się obiektu są duże obszary o lokalnie stałej jasności.

Na rys. 10.3, 10.4 pokazano maski zmian znalezione dla obydwu sekwencji za pomocą metody HNRLin. Maski zmian dla sekwencji D składają się z wielu „plam” – daleko im do tego co jest w stanie dostrzec oko ludzkie. W przypadku sekwencji E znalezione maski zmian są pełniejsze, ale ta ich część, która odpowiada sytuacji (ruch, ruch) jest niespójna.

Wyniki detekcji ruchu przeprowadzonej dla obydwu sekwencji przy użyciu masek zmian z rys. 10.3, 10.4 pokazano na rys. 10.5, 10.6 dla sekwencji D oraz na rys. 10.7, 10.8 dla sekwencji E. W przypadku sekwencji D zarówno model LB jak i nowy model z dwuwymiarowym wektorem obserwacji dają bardzo złe wyniki detekcji ruchu obiektu. Maski obiektu znalezione dla sekwencji E mają wyraźne braki, zgodne z brakami zastosowanych masek zmian.

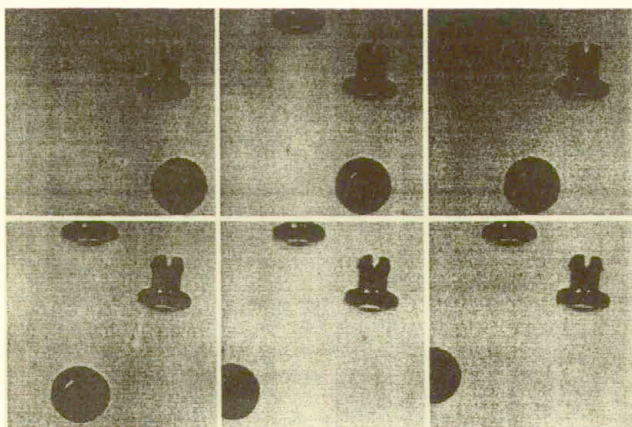
W tym miejscu można zadać następujące pytanie:

Czy możliwe jest, przy wykorzystaniu czasowo-przestrzennych pól Markowa, znalezienie masek obiektu ruchomego, w których uzupełnione byłyby braki zastosowanych masek zmian?

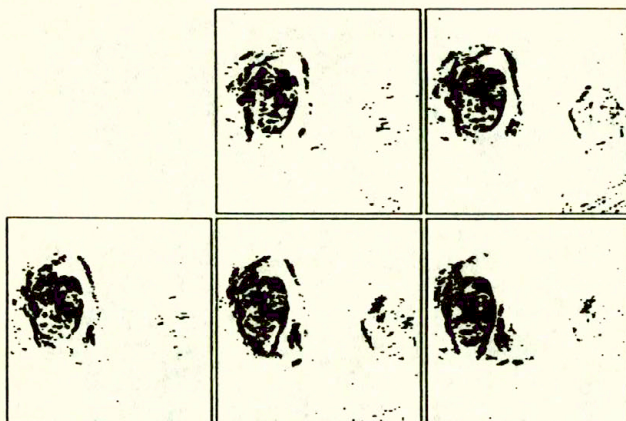
Odpowiedź jest twierdząca ale przy założeniu, że obserwacja wpływająca na ukryte pole Markowa ulegnie poszerzeniu. W przypadku sekwencji o stałej lub prawie stałej funkcji jasności obiektu, w celu poprawnego rozróżnienia pomiędzy sytuacjami (ruch, ruch) oraz (tło, tło), należy, poza różnicą jasności przyporządkowanych pikselowi w kolejnych obrazach sekwencji, dodatkowo wykorzystać jeszcze jasność danego piksla. Aby to dokładniej wyjaśnić założymy, że pewna sekwencja przedstawia ruch obiektu o stałej jasności, przy czym tło również ma stałą jasność. Dla takiej sekwencji różnica jasności dla sytuacji (ruch, ruch) oraz (tło, tło) jest równa zero, ale jasności tła i obiektu są różne.



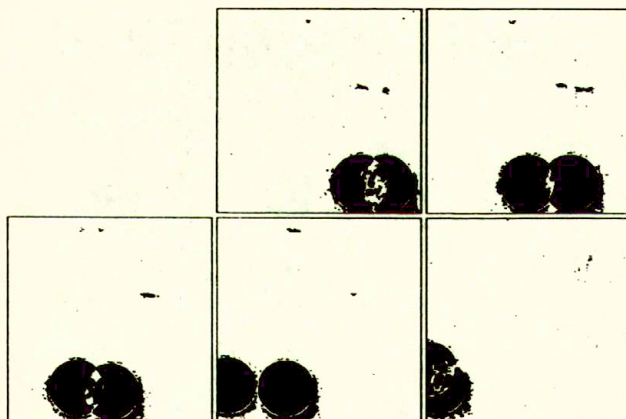
Rys. 10.1: Sekwencja D przedstawiająca ruch obrotowy wokół pionowej osi głowy modelki siedzącej z lewej strony.



Rys. 10.2: Sekwencja E przedstawiająca ruch o charakterze translacji.



Rys. 10.3: Maski zmian zachodzących pomiędzy kolejnymi obrazami sekwencji D z rys. 10.1. Maski te znaleziono za pomocą metody HNRlin (próg=50).



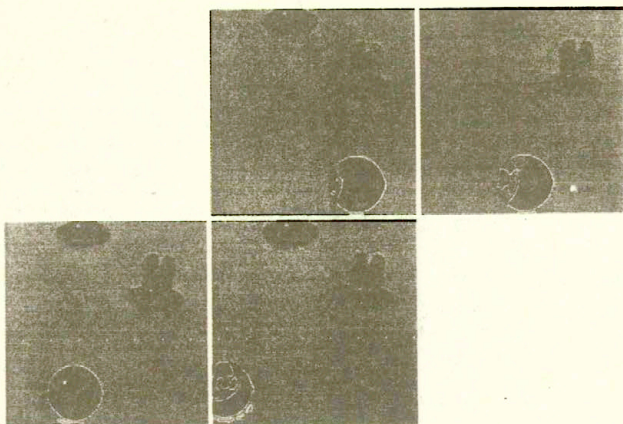
Rys. 10.4: Maski zmian zachodzących pomiędzy kolejnymi obrazami sekwencji E z rys. 10.2. Maski te zostały znalezione za pomocą metody HNRlin (próg=50).



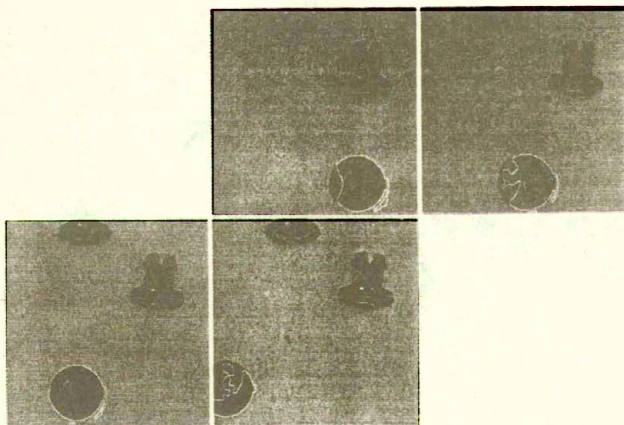
Rys. 10.5: Maski obiektu ruchomego nałożone na odpowiednie obrazy sekwencji D, otrzymane dzięki zastosowaniu nowego modelu z dwuwymiarowym wektorem obserwacji. ($m_1 = -21.3$, $m_2 = 39.3$, $m_3 = 2.4$, $\sigma^2 = 225$). Wykorzystano maski zmian z rys. 10.3.



Rys. 10.6: Maski obiektu ruchomego nałożone na odpowiednie obrazy sekwencji D, otrzymane dzięki zastosowaniu modelu LB ($m_1 = 9.1$, $m_2 = -2.4$, $\sigma^2 = 225$). Wykorzystano maski zmian z rys. 10.3.



Rys. 10.7: Maski obiektu ruchomego nałożone na odpowiednie obrazy sekwencji E, otrzymane dzięki zastosowaniu nowego modelu z dwuwymiarowym wektorem obserwacji, ($m_1 = 124$, $m_2 = -122$, $m_3 = -4$, $\sigma^2 = 225$). Wykorzystano maski zmian z rys. 10.4.



Rys. 10.8: Maski obiektu ruchomego nałożone na odpowiednie obrazy sekwencji E, otrzymane dzięki zastosowaniu modelu LB ($m_1 = -2.27$, $m_2 = 10$, $\sigma^2 = 225$). Wykorzystano maski zmian z rys. 10.4.

W przypadku analizy ruchu bazującej na czasowo-przestrzennym ukrytym polu Markowa dodatkowe uwzględnienie jasności piksla jako źródła informacji prowadzi do powiększenia o jeden wymiar wektora obserwacji wpływającej na rozkład pola ukrytego. W następnym rozdziale przedstawiono właśnie taki model.

10.2 Ukryte pole Markowa z trójwymiarowym wektorem obserwacji

Przystępując do budowy nowego pola Markowa należałoby udzielić odpowiedzi na wszystkie pytania, zgodnie z metodą wprowadzoną w rozdziale 3. W proponowanym poniżej modelu zakłada się, że dziedzina, zbiór stanów, typ, rząd, wyrażenie opisujące energię przestrzenną oraz wartości parametrów β_s i β_t są identyczne jak w przypadku modelu LB, czy nowego modelu z dwuwymiarowym wektorem obserwacji z rozdziału 5. Bez zmian więc pozostaje również podział poszukiwania maski obiektu ruchomego na dwa etapy: na predykcję oraz korekcję. Proponowane zmiany dotyczą jedynie:

1. wyrażenia opisującego energię zgodności, oraz
2. tablicy określającej warunkową energię klik czasowych.

W modelu wprowadzonym przez Lalonde'a-Bouthemy'ego, jak również w modelu zaproponowanym w rozdziale 5, wykorzystano dwa źródła obserwacji wpływające na rozkład ukrytego pola Markowa: maskę zmian oraz różnicę jasności przyporządkowanej pojedynczemu pikslowi w kolejnych obrazach sekwencji. W obydwu modelach przyjęto, że maska zmian \bar{o}_k wpływa jedynie na energię czasową, natomiast wpływ różnicy jasności o_k uwzględniono w energii zgodności.

W modelu proponowanym w niniejszym rozdziale zakłada się korzystanie z trzech źródeł obserwacji:

1. maski zmian $\bar{o}_k(i, j)$ zachodzących pomiędzy obrazami o numerach k i $k + 1$,
2. różnicy $o_k(i, j)$ jasności piksla (i, j) w obrazach k -tym oraz $k + 1$ -szym,
3. jasności $f_k(i, j)$ przyporządkowanej pikslowi (i, j) w obrazie k -tym.

Tak jak w przypadku poprzednich modeli przyjmuje się, że maska zmian wpływa jedynie na energię czasową ukrytego pola Markowa. Zakłada się, że obserwacje $o_k(i, j)$ oraz $f_k(i, j)$ tworzą dwuwymiarowy wektor, będący częścią trójwymiarowego wektora obserwacji. Wpływ takiego dwuwymiarowego wektora na energię ukrytego pola Markowa opisany jest pojedynczym składnikiem energii, zwanym, tak jak poprzednio, energią zgodności. Aby podać wyrażenie opisujące energię zgodności U_{ij} , zdefiniujemy pomocniczą funkcję:

$$\psi[e_k(i, j), e_{k+1}(i, j)] = \begin{cases} [m_{01}, m_{02}, \sigma_{01}^2, \sigma_{02}^2, \rho_0]^T & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (b, b) \\ [m_{11}, m_{12}, \sigma_{11}^2, \sigma_{12}^2, \rho_1]^T & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (a, b) \\ [m_{21}, m_{22}, \sigma_{21}^2, \sigma_{22}^2, \rho_2]^T & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (b, a) \\ [m_{31}, m_{32}, \sigma_{31}^2, \sigma_{32}^2, \rho_3]^T & \text{gdy } (e_k(i, j), e_{k+1}(i, j)) = (a, a), \end{cases} \quad (10.1)$$

przy czym:

- m_{01} – wartość średnia różnicy jasności o_k dla realizacji (b, b) ,
- m_{02} – wartość średnia jasności f_k dla realizacji (b, b) ,
- σ_{01}^2 – wariancja różnicy jasności o_k dla realizacji (b, b) ,
- σ_{02}^2 – wariancja jasności f_k dla realizacji (b, b) ,
- ρ_0 – współczynnik korelacji różnicy o_k i jasności f_k dla realizacji (b, b) ,
- m_{11} – wartość średnia różnicy jasności o_k dla realizacji (a, b) ,
- m_{12} – wartość średnia jasności f_k dla realizacji (a, b) ,
- σ_{11}^2 – wariancja różnicy jasności o_k dla realizacji (a, b) ,
- σ_{12}^2 – wariancja jasności f_k dla realizacji (a, b) ,
- ρ_1 – współczynnik korelacji różnicy o_k i jasności f_k dla realizacji (a, b) ,
- m_{21} – wartość średnia różnicy jasności o_k dla realizacji (b, a) ,
- m_{22} – wartość średnia jasności f_k dla realizacji (b, a) ,
- σ_{21}^2 – wariancja różnicy jasności o_k dla realizacji (b, a) ,
- σ_{22}^2 – wariancja jasności f_k dla realizacji (b, a) ,
- ρ_2 – współczynnik korelacji różnicy o_k i jasności f_k dla realizacji (b, a) ,
- m_{31} – wartość średnia różnicy jasności o_k dla realizacji (a, a) ,
- m_{32} – wartość średnia jasności f_k dla realizacji (a, a) ,
- σ_{31}^2 – wariancja różnicy jasności o_k dla realizacji (a, a) ,
- σ_{32}^2 – wariancja jasności f_k dla realizacji (a, a) ,
- ρ_3 – współczynnik korelacji różnicy o_k i jasności f_k dla realizacji (a, a) .

Jest to funkcja wektorowa w odróżnieniu od analogicznych funkcji dla modeli z dwuwymiarowymi wektorami obserwacji, gdzie funkcje te były funkcjami skalarnymi.

Przyjmuje się, że rozkład prawdopodobieństwa zgodności realizacji pola Markowa $(e_k(i, j), e_{k+1}(i, j))$ z obserwacją $[o_k(i, j), f_k(i, j)]^T$ ma postać dwuwymiarowego rozkładu normalnego z parametrami $[m_{11}, m_{12}, \sigma_{11}^2, \sigma_{12}^2, \rho_1]^T$, przy czym l jest równe albo 0, albo 1, albo 2, albo 3. W takim przypadku lokalna energia zgodności opisana jest równaniem [Mor90]:

$$U_{ijz} = \frac{1}{2(1-\rho_l^2)} \left[\left(\frac{o_k(i, j) - m_{11}}{\sigma_{11}} \right)^2 - 2\rho_l \left(\frac{o_k(i, j) - m_{11}}{\sigma_{11}} \right) \left(\frac{f_k(i, j) - m_{12}}{\sigma_{12}} \right) + \left(\frac{f_k(i, j) - m_{12}}{\sigma_{12}} \right)^2 \right] \quad (10.2)$$

Wymiar wektora parametrów energii zgodności Θ_z jest równy 20.

Dzięki wprowadzeniu dodatkowego źródła obserwacji możliwa jest modyfikacja wartości energii klikli czasowej przypisanej realizacji (a, a) , gdy $\bar{o}_k(i, j) = 0$, tzn. gdy nie wykryto zmian w pikslu (i, j) . W przypadku sekwencji „trudnych” realizacja (a, a) powinna mieć takie samo prawdopodobieństwo wystąpienia, gdy maska zmian w obrazach informuje o zachodzących zmianach ($\bar{o}_k(i, j) = 1$), jak i wtedy gdy zmiany nie zostały

wykryte ($\bar{o}_k(i, j) = 0$). Zmodyfikowana energia czasowa ma postać:

$$U_{i,t}\{e_k(i, j), e_{k+1}(i, j) \mid [\bar{o}_k(i, j), f_k(i, j)]^T\} = \left\{ \begin{array}{c|cc} & \bar{o}_k = 0 & \bar{o}_k = 1 \\ \hline (b, b) & -\beta_t & \beta_t \\ (a, b) & \beta_t & -\beta_t \\ (b, a) & \beta_t & -\beta_t \\ \hline (a, a) & -\beta_t & -\beta_t \end{array} \right\}, \quad (10.3)$$

przy czym: β_t stała większa od zera. Jako algorytm poszukiwania mody pola Markowa również proponuje się algorytm relaksacji deterministycznej.

10.3 Przykłady zastosowania pola Markowa z trójwymiarowym wektorem obserwacji do detekcji masek obiektów ruchomych

Model wprowadzony w rozdziale 10.2 określony jest z dokładnością do 20 parametrów. Jest to o wiele więcej niż w przypadku modelu LB (3 parametry) czy nowego modelu z dwuwymiarowym wektorem obserwacji zaproponowanego w rozdziale 5 (4 parametry). Pojawia się pytanie jak przeprowadzić identyfikację takiego modelu? Przy tak dużej liczbie parametrów raczej niemożliwym jest dobieranie ich wartości metodą prób i błędów. Wydaje się, że jedyną rozsądną metodą identyfikacji jest przeprowadzenie *uczenia modelu*. W przypadku ukrytego pola Markowa uczenie może być rozumiane w następująco: najpierw użytkownik musi przeprowadzić ręczną segmentację sekwencji trenującej, tzn. stworzyć, metodą zaznaczania pojedynczych pikseli, binarną maskę obiektu ruchomego dla każdego obrazu sekwencji wejściowej. Następnie, mając dostępną sekwencję wejściową oraz znaną „ręcznie” sekwencję masek, obliczyć dla każdej z czterech możliwych realizacji (a, a) , (a, b) , (b, a) oraz (b, b) wartości średnie oraz wariancje z próby różnicy o_k i jasności f_k , jak również współczynnik korelacji różnicy o_k i jasności f_k . Przedstawione estymaty są w przypadku dwuwymiarowego rozkładu normalnego estymatami największej wiarygodności [Mor90].

Za pomocą „uczenia modelu” znaleziono wartości parametrów dla sekwencji testowych z rys. 10.1 oraz 10.2. W przypadku sekwencji D parametry te wyniosły:

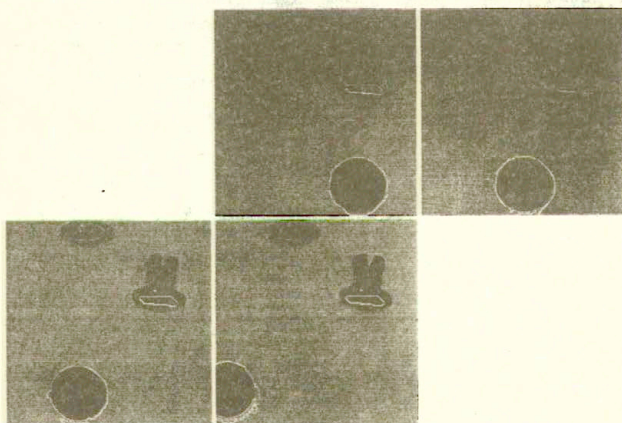
$$\begin{array}{llllll} m_{01} = -0.05 & m_{02} = 49.12 & \sigma_{01}^2 = 38.93 & \sigma_{02}^2 = 2534.91 & \rho_0 = -0.08 \\ m_{11} = -21.28 & m_{12} = 24.63 & \sigma_{11}^2 = 273.06 & \sigma_{12}^2 = 264.28 & \rho_1 = -0.85 \\ m_{21} = 39.3 & m_{22} = 17.8 & \sigma_{21}^2 = 1465.6 & \sigma_{22}^2 = 617.38 & \rho_2 = -0.43 \\ m_{31} = -2.37 & m_{22} = 87.53 & \sigma_{31}^2 = 1454.66 & \sigma_{32}^2 = 1797.26 & \rho_3 = -0.41 \end{array}$$

Dla sekwencji E otrzymano:

$$\begin{array}{llllll} m_{01} = 0.14 & m_{02} = 144.17 & \sigma_{01}^2 = 2.09 & \sigma_{02}^2 = 978.94 & \rho_0 = -0.18 \\ m_{11} = 123.88 & m_{12} = 28.24 & \sigma_{11}^2 = 1265.62 & \sigma_{12}^2 = 1268.94 & \rho_1 = -0.9991 \\ m_{21} = -121.65 & m_{22} = 152.44 & \sigma_{21}^2 = 1153.03 & \sigma_{22}^2 = 1.41 & \rho_2 = 0.05 \\ m_{31} = -4.03 & m_{22} = 30.79 & \sigma_{31}^2 = 2174.69 & \sigma_{32}^2 = 1025.29 & \rho_3 = -0.69. \end{array}$$



Rys. 10.9: Maski obiektu ruchomego nałożone na odpowiednie obrazy sekwencji D, otrzymane dzięki zastosowaniu modelu z trójwymiarowym wektorem obserwacji, przy wykorzystaniu masek zmian z rys. 10.3.



Rys. 10.10: Maski obiektu ruchomego nałożone na odpowiednie obrazy sekwencji E, otrzymane dzięki zastosowaniu modelu z trójwymiarowym wektorem obserwacji, przy wykorzystaniu masek zmian z rys. 10.4.

Następnie dla obydwu sekwencji testowych przeprowadzono przy użyciu modelu z trójwymiarowym wektorem obserwacji detekcję masek obiektu ruchomego, stosując znalezione wartości parametrów. W obydwu przypadkach zastosowano te same maski zmian, jak w eksperymentach z modelami LB oraz nowym modelem z dwuwymiarowym wektorem obserwacji.

Na rys. 10.9 przedstawiono maski obiektu znalezione dla sekwencji D, a na rys. 10.10 maski obiektu znalezione dla sekwencji E. W przypadku obydwu sekwencji wyniki są rewelacyjne. Ogromna większość braków maski zmian w obrazie została uzupełniona, czego nigdy nie udało się uzyskać przy pomocy obydwu modeli z dwuwymiarowym wektorem obserwacji. W przypadku sekwencji D wykryty został również obiekt statyczny, charakteryzujący się taką samą jasnością jak obiekt ruchomy. Nie należy się jednak temu dziwić, gdyż już w maskach zmian z rys. 10.3 zostały w tym miejscu wykryte zmiany, najprawdopodobniej spowodowane efektami oświetleniowymi.

Rozdział 11

Podsumowanie

11.1 Wnioski

Wszystkie cele postawione na początku niniejszej rozprawy zostały zrealizowane.

1. Nowy model statystyczny w dwuwymiarowym wektorze obserwacji, przeznaczony do detekcji masek obiektów ruchomych, został opracowany i zaprezentowany w rozdziale 5. Cechą charakterystyczną tego modelu jest, tak jak założono we wstępie, uwzględnienie wszystkich możliwości etykietowania pojedynczego piksla. Dzięki tej właściwości model lepiej opisuje rzeczywistość, a więc łatwiej można go dopasować do danych wejściowych.
2. Model statystyczny z trójwymiarowym wektorem obserwacji został wprowadzony w rozdziale 10. Jest to pierwszy model statystyczny przeznaczony do detekcji masek obiektów ruchomych, w którym wykorzystano trójwymiarowy wektor obserwacji. Największą zaletą zaproponowanego modelu jest to, że znajdowane są przy jego pomocy maski obiektów ruchomych o lepszej jakości, niż w przypadku modeli z dwuwymiarowym wektorem obserwacji. Zbudowany model pozwala na znajdowanie dobrych masek obiektów ruchomych nawet dla obiektów o lokalnie stałej funkcji jasności. Komputerowa analiza ruchu takich obiektów znanymi metodami sprawiała dotychczas wiele kłopotów.
3. Metoda budowy modeli statystycznych korzystających z pól Markowa została opracowana i przedstawiona w rozdziale 3. Użyteczność tej metody potwierdzono wykorzystując ją do budowy dwóch nowych modeli przeznaczonych do detekcji masek obiektów ruchomych, tzn. nowego modelu z dwuwymiarowym wektorem obserwacji (rozdział 5) oraz modelu z trójwymiarowym wektorem obserwacji (rozdział 10). Niewątpliwą zaletą opracowanej metody jest jej ogólność. Stosując tę metodę można konstruować modele statystyczne przeznaczone do rozwiązywania różnych problemów komputerowej analizy obrazów.
4. Kryterium oceny dokładności masek obiektów ruchomych znajdujących za pomocą metod statystycznych wprowadzono w rozdziale 6. Dzięki zaproponowanemu kryterium możliwym stało się obiektywne porównanie różnych metod detekcji masek obiektów ruchomych.
5. Wyniki badań eksperymentalnych modelu Lalande'a-Bouthemy'ego znanego z literatury, potwierdzające użyteczność tego modelu do detekcji masek obiektów ruchomych, przedstawiono w rozdziale 7. W szczególności na podstawie przeprowadzonych dla potrzeb niniejszej rozprawy badań (tab. 7.1, 7.2, 7.3, 7.4) stwierdzono, że w przypadku modelu zaproponowanego przez Lalande i Bouthemy'iego:
 - wartość parametru σ^2 jest nieistotna, o ile tylko nie jest za mała. Dzięki temu można przyjąć jedną wartość tego parametru dla dużej grupy różnych czasowych sekwencji obrazów.
 - Wartości parametrów m_1 oraz m_2 nie mogą być estymowane metodą uśredniania różnic jasności dla sekwencji trenujących.

- Wartości parametrów m_1 , m_2 nie są istotne, a liczy się tylko ich wzajemna relacja, tzn. który z nich jest większy.
 - Przy doborze wartości m_1 , m_2 należy kierować się informacją o wzajemnej relacji jasności poruszającego się obiektu i tła.
6. Wyniki badań nowego modelu z obserwacją dwuwymiarową zamieszczone w rozdziale 8 (tab. 8.1, 8.2, 8.3, 8.4) pozwoliły na stwierdzenie bardzo dużej użyteczności tego modelu do detekcji masek obiektów ruchomych. Okazało się, że estymacja parametrów modelu, opierająca się na wartościach średnich różnicy jasności, w każdym przypadku daje poprawne wartości parametrów modelu, czyli estymacja parametrów jest łatwiejsza niż w przypadku modelu znanego z literatury.
 7. Kompleksowa metoda detekcji i śledzenia ruchu obiektów przedstawionych za pomocą sekwencji obrazów z gradacją szarości została wprowadzona w rozdziale 9, a jej użyteczność potwierdziły zamieszczone tam wyniki badań eksperymentalnych.
 8. Metoda detekcji masek obiektów ruchomych za pomocą sumowania masek znajdowanych przy różnych wartościach parametrów pola Markowa, razem z przykładami świadczącymi o jej użyteczności, została zaprezentowana w rozdziale 9. Zaletą tej metody jest **niezależenie**, w dużym stopniu, detekcji masek obiektów ruchomych od treści analizowanych sekwencji obrazów, dzięki stosowaniu kilku zestawów wybranych wcześniej parametrów. Praktyczne wykorzystanie tej metody znacznie zmniejsza uciążliwość etapu estymacji parametrów modelu.
 9. Modyfikacja znanego z literatury algorytmu relaksacji deterministycznej została opracowana a następnie opisana w rozdziale 5. Niewątpliwą zaletą proponowanej modyfikacji jest, w przypadku detekcji ruchu, 20-krotne skrócenie czasu obliczeń.
 10. Eksperymenty przedstawione w rozdziale 4, a mające na celu wybór optymalnego detektora zmian w obrazie, nie pozwoliły na jednoznaczny wybór takiego detektora. Okazało się, że większość algorytmów detekcji zmian czasowych prowadzi do porównywalnych rezultatów. Ubocznym efektem przeprowadzonych eksperymentów jest określenie kryteriów, którymi należy się kierować w trakcie detekcji zmian w obrazach, tak aby uzyskać maski zmian w obrazie o dobrych właściwościach.
 11. Pakiet programów do detekcji ruchu został stworzony, a jego opis zamieszczono w Dodatku B.

Dzięki zrealizowaniu wszystkich zamierzonych celów **bezsprzecznie można stwierdzić**, że słuszność tezy rozprawy została **wykazana**, gdyż pokazano, że detekcja i śledzenie ruchu przeprowadzane przy użyciu czasowo-przestrzennego modelowania obrazów mogą być u efektywne poprzez zmianę wyrażenia energii pola Markowa. Odpowiednio dokonane zmiany wyrażenia energii znanego z literatury modelu opartego na ukrytych polach Markowa, stały się podstawą do zbudowania dwóch różnych modeli statystycznych, charakteryzujących się lepszymi właściwościami niż model dotychczasowy. I tak:

1. rozszerzenie wyrażenia energii, tak aby były uwzględnione wszystkie możliwości etykietowania piksla pozwoliło na:

- szybsze znajdowanie masek obiektu ruchomego,
- detekcję dokładniejszych masek,
- łatwiejszą estymację parametrów, tzn. metodą uśredniania na podstawie sekwencji trenujących.

2. rozszerzenie wyrażenia energii o dodatkowe źródło obserwacji umożliwiło uzupełnianie braków informacji, powstałych w trakcie detekcji zmian w obrazie.

Zaprezentowana rozprawa niewątpliwie potwierdziła ogromną użyteczność statystycznych modeli pól Markowa do rozwiązywania zadań komputerowej analizy obrazów, a tym samym udowodniła, że rosnąca popularność tych modeli wśród badaczy zajmujących się wizją komputerową jest w pełni uzasadniona.

11.2 Kierunki dalszych badań

Chociaż zaprezentowana rozprawa stanowi zamkniętą całość, to jednak nie zostały poruszone w niej wszystkie problemy dotyczące wybranego podejścia do detekcji i śledzenia ruchu obiektów znajdujących się w polu widzenia kamery. Do przebadania pozostają następujące zagadnienia:

1. Przebadanie wpływu liczby modelowanych jednocześnie polem Markowa kolejnych obrazów sekwencji na dokładność znajdowanych masek obiektu ruchomego.
2. Zwiększenie liczby klik czasowych w modelu sąsiedztwa pola Markowa opisującego rozkład etykiet a , b .
3. Przebadanie parametrów energii zgodności modelu z trójwymiarowym wektorem obserwacji, w celu znalezienia tych, które mają niewielki wpływ na dokładność znajdowanych masek.
4. Ocena wrażliwości modelu z trójwymiarowym wektorem obserwacji na wartości jego parametrów.

Bibliografia

- [ABP92] O. Allagnat, J. M. Boucher, W. Pieczynski. Hidden Markov fields and unsupervised segmentation of images. In *Proceedings of ICPR 92*, The Hague, The Netherlands, September 1992.
- [AKM93] T. Aach, A. Kaup, R. Mester. Statistical model-based change detection in moving video. *Signal Processing*, 31:165-180, 1993.
- [AN88] J. K. Aggarwal, N. Nadhakumar. On the computation of motion from sequences of images - a review. *Proceedings of IEEE*, 76(8):917-935, 1988.
- [AS92] E. Antoniu, I. Sandau. On pros and cons of implementing '80s theory of edge detection. *Machine Graphics & Vision*, 1(1/2):255-260, 1992.
- [AT90] J. Aloimonos, D. Tsakiris. Tracking in a complex visual environment. In *Proceedings of European Conference on Computer Vision*, str. 249-258, Antibes, France, April 1990. Springer-Verlag.
- [Bes74] J. Besag. Spatial interaction and the statistical analysis of lattice systems. *Journal of Royal Statistical Society series B*, 36:192-236, 1974.
- [Bes86] J. Besag. On the statistical analysis of dirty pictures. *Journal of Royal Statistical Society series B*, 48:259-302, 1986.
- [BHM86] A. C. Bovik, T. S. Huang, D. C. Jr. Munson. Nonparametric tests for edge detection in noise. *Pattern Recognition*, 19(3):115-219, 1986.
- [BL90] P. Bouthemy, P. Lalande. Detection and tracking of moving objects based on a statistical regularization method in space and time. In *Proceedings of European Conference on Computer Vision*, str. 307-311, Antibes, France, April 1990.
- [Bla90] J. M. Black. A model for the detection of motion over time. In *Proceedings of International Conference on Computer Vision*, Osaka, Japan, December 1990. IEEE Computer Society Press.
- [Bob86] D. Bobrowski. *Probabilistyka w zastosowaniach technicznych*. Wydawnictwa Naukowo-Techniczne, Warszawa, 1986.
- [Bou89] P. Bouthemy. A maximum likelihood framework for determining moving edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(5):499-511, 1989.

- [BPM93] B. Braathen, W. Pieczynski, P. Masson. Global and local methods of unsupervised Bayesian segmentation of images. *Machine Graphics & Vision*, 2(1):39-52, 1993.
- [Can86] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679-698, 1986.
- [Car92] J. M. Carstensen. *Description and Simulation of Visual Texture*. PhD thesis, IMSOR, Lyngby Technical University, Denmark, 1992.
- [Cha88] B. Chalmond. Image restoration using an estimated Markov model. *Signal Processing*, 15(2):115-129, September 1988.
- [CJ83] G. R. Cross, A. K. Jain. Markov random fields texture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 5(1):25-39, January 1983.
- [CSC⁺] L. Chmielewski, M. Skłodowski, W. Cudny, M. Nieniewski, A. Kuriański, B. Michalski. Fringe image enhancing in the light wavelength stepping method. Zgłoszony do publikacji w *Machine Graphics & Vision*.
- [CSC⁺93] L. Chmielewski, M. Skłodowski, W. Cudny, M. Nieniewski, A. Kuriański, B. Michalski. Fringe image processing in the white light wavelength stepping method. Sprawozdanie z wykonania grantu KBN 8 8055 91 02, ZBS EPSILON, kwiecień 1993.
- [CSN⁺94] L. Chmielewski, M. Skłodowski, M. Nieniewski, A. Kuriański, B. Michalski. Nonstandard methods of enhancing fringe images. In *Proceedings of 5th MICROCROCOMPUTER School*, str. 209-223, Zakopane, luty 1994.
- [DA92] E. L. Dagless, A. T. Ali. Real-time vision-aided traffic monitoring. *Applied Mathematics and Computer Science*, 3(1):125-134, 1992. Proceedings of the ACEP Workshop, Borowice, Poland 1992.
- [DE87] H. Derin, H. Elliott. Modeling and segmentation of noisy and textured images using Gibbs random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(1):39-55, January 1987.
- [DN82] L. Dreschler, H. H. Nagel. Volumetric model and 3d trajectory of a moving car derived from monocular tv frame sequence of a street scene. *Computer Graphics and Image Processing*, 20:199-228, 1982.
- [EG92] M. El-Gabali. Restoration of images corrupted by non-causal blur and noise using Markov random field model. In *Proceedings of International AM SE Conference „Applications of Signals, Data, Systems Methodologies to Engineering Problems”*, str. 37-48, Alexandria, Egypt, December 1992.
- [EGK92] M. El-Gabali, S. Khalaf. Efficient implementation of hierarchically ordered Markov random field models on single instruction multiple data machine architecture. In *Proceedings of International AM SE Conference „Applications of*

Signals, Data, Systems Methodologies to Engineering Problems", str. 107-120, Alexandria, Egypt, December 1992.

- [FB90a] E. François, P. Bouthemy. Derivation of qualitative information in motion analysis. *Image and Vision Computing*, 8(4):279-287, November 1990.
- [FB90b] E. François, P. Bouthemy. Multiframe-based identification of mobile components of a scene with moving camera. Technical Report 1368, IRISA, December 1990.
- [Fel68] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume 1. John Wiley & Sons Inc., New York, London, Sydney, 3d edition, 1968.
- [GG84] S. Geman, D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721-741, November 1984.
- [GG91] D. Geiger, F. Girosi. Parallel and deterministic algorithms from MRF's: surface reconstruction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:401-412, May 1991.
- [HB90a] F. Heitz, P. Bouthemy. Multimodal estimation of discontinuous optical flow using Markov random fields. Technical Report 561, IRISA, November 1990.
- [HB90b] F. Heitz, P. Bouthemy. Multimodal estimation of discontinuous optical flow using Markov random fields. In *Proceedings of 10th International Conference on Pattern Recognition (Computer Vision Conference)*, Atlantic City, June 1990.
- [HJ83] S. M. Haynes, R. Jain. Detecting of moving edges. *Computer Vision Graphics and Image Processing*, 21:345-367, 1983.
- [HMB91] F. Heitz, E. Memin, P. Bouthemy. Markov random fields models and parallel algorithms for 2D motion analysis. In *Proceedings of 13th IMACS World Congress on Computation and Applied Mathematics*, Dublin, July 1991.
- [HNR84] Y. Z. Hsu, H. H. Nagel, G. Rekers. New likelihood test method for change detection in image sequence. *Computer Vision Graphics and Image Processing*, 26:73-106, 1984.
- [Hor89] B. K. P. Horn. *Robot Vision*. Mir, Moskwa, 1989. wydanie rosyjskie.
- [HS80] M. Hassner, J. Sklansky. The use of Markov random fields as models of texture. *Computer Graphics and Image Processing*, 12:357-370, 1980.
- [HW81] R. M. Haralick, L. Watson. A facet model for image data. *Computer Graphics and Image Processing*. 15:113-129, 1981.
- [IT93] P. Iliev, L. Tsekov. Motion detection using image histogram sequence analysis. *Signal Processing*, 30:373-384, 1993.

- [Jai82] R. Jain. Segmentation of moving observer frame sequences. *Pattern Recognition Letters*, 1:115-120, 1982.
- [Jar94] J. Jarnicki. A probabilistic approach to image motion fields. In *Proceedings of 5th MICROCOMPUTER School*, str. 35-42, Zakopane, luty 1994.
- [Jin90] K. Jinyu. Deterministic relaxation and simulated annealing for image restoration. In *Proceedings of International Conference on Signal Processing*, str. 987-990, Beijing, China, October 1990.
- [JN90] A. K. Jain, S. G. Nadabar. MRF model-based segmentation of range images. In *Proceedings of International Conference on Computer Vision*. IEEE Computer Society Press, December 1990.
- [JW91] F. C. Jeng, J. W. Woods. Compound Gauss-Markov random fields for image estimation. *IEEE Transactions on Signal Processing*, 39(3):683-697, March 1991.
- [Kan80] L. N. Kanal. Markov mesh models. *Computer Graphics and Image Processing*, 12:683-697, 1980.
- [KD90] J. Konrad, E. Dubois. A comparison of stochastic and deterministic solution methods in Bayesian estimation of 2-d motion. In *Proceedings of European Conference on Computer Vision*, Antibes, France, April 1990. Springer-Verlag.
- [KGV83] S. Krikpatrick, C. D. Jr. Gelatt, M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4595):671-688, May 1983.
- [KL86] S. M. Kay, G. J. Lemay. Edge detection using the linear model. *IEEE Transactions on Acoustics Sound and Signal Processing*, 34(5):1221-1227, 1986.
- [Kor93] A. Korzyńska. Using Markovian models to the description of cells' movement. In *Proceedings of the American-Polish Workshop on Image Processing in Medicine and Biology*, Warszawa, maj 1993.
- [KRS2] L. Kitchen, A. Rosenfeld. Grey-level corner detection. *Pattern Recognition Letters*, 1:95-102, 1982.
- [KS93] A. Kuriański, M. Skłodowski. Enhancement of speckle images with the use of statistical hierarchical Markov field models. *Machine Graphics & Vision*, 2(4):309-338, 1993.
- [Kur] A. Kuriański. Pola Markowa w komputerowej analizie obrazów. Materiały II Krajowego Seminarium Metod Matematycznych Analizy Obrazów Prążkowych zorganizowane przez OC PTM, Częstochowa 26-27 marca 1993 - w druku.
- [Kur92] A. Kuriański. Time-varying corners in practical applications - discussion. *Machine Graphics & Vision*, 1(3):527-536, 1992.

- [Kur93] A. Kuriański. Improvement of edge images. In *Proceedings of the International Workshop on Intelligent Robotic Systems '93*, str. 234–243, Zakopane, lipiec 1993.
- [Lac87] Lacroix. Pixel labeling in a second-order Markov mesh. *Signal Processing*, 12(1):59–82, 1987.
- [LB90] P. Lalande, P. Bouthemy. A statistical approach to the detection and tracking of moving objects in an image sequence. In *Proceedings of 5th ESPC EUSIPCO 90*, Barcelona, September 1990.
- [LD89] S. Lakshmanan, H. Derin. Simultaneous parameter estimation and segmentation of Gibbs random fields using simulated annealing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(8):799–813, August 1989.
- [LHS87] J. S. J. Lee, R. M. Haralick, L. G. Shapiro. Morphologic edge detection. *IEEE Journal of Robotics and Automation*, 3(2):142–156, April 1987.
- [MAG88] S. A. Mahmoud, M. S. Afifi, R. J. Green. Recognition and velocity computation of large moving objects in images. *IEEE Transactions on Acoustics Sound and Signal Processing*, 36:1790–1791, November 1988.
- [Mah91a] S. A. Mahmoud. Motion analysis of multiple moving objects using Hartley transform. *IEEE Transactions on Systems, Man and Cybernetics*, 21(1):280–287, 1991.
- [Mah91b] S. A. Mahmoud. A new technique for velocity estimation of large moving objects. *IEEE Transactions on Signal Processing*, 39(3):741–743, 1991.
- [MB87] D. W. Murray, B. F. Buxton. Scene segmentation from visual motion using global optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(2):220–228, March 1987.
- [MC93] H. Mori, N. M. Charkari. Shadow and rhythm as sign patterns of obstacle detection. In *Proceedings of IEEE International Symposium on Industrial Electronics*, str. 271–277, Budapest, Hungary, June 1993.
- [Mor90] D. F. Morrison. *Wielowymiarowa analiza statystyczna*. Państwowe Wydawnictwa Naukowe, Warszawa, 1990.
- [Nag82] H. H. Nagel. On change detection and displacement vector estimation in image sequences. *Pattern Recognition Letters*, 1:55–59, 1982.
- [Nag83] H. H. Nagel. Displacement vectors derived from second-order intensity variations in image sequences. *Computer Vision Graphics and Image Processing*, 21:85–117, 1983.
- [NB86] V. S. Nalwa, T. O. Binford. On detecting edges. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):699–714, 1986.

- [Nie92] M. Nieniewski. Statistical investigation of 2D morphological filters for image processing. *Machine Graphics & Vision*, 1(3):495-526, 1992.
- [Nie93] M. Nieniewski. Detekcja ruchu w obrazach sceny roboczej. Sprawozdanie z wykonania grantu KBN 8 8521 91 02 Zakład Badań Podstawowych Elektrotechniki MPIH i PAN w Instytucie Elektrotechniki, kwiecień 1993.
- [Nie94] M. Nieniewski. Mathematical morphology in image processing. In *Proceedings of 5th MICROCOMPUTER School*, str. 72-95, Zakopane, luty 1994.
- [NP90] M. Nieniewski, P. Pathak. Change detection in image sequences using Walsh functions and the likelihood test method. In *Proceedings of International Conference on Signal Processing*, str. 995-998, Beijing, China, October 1990.
- [NP91] M. Nieniewski, P. Pathak. Change detection in two images using Walsh functions and the likelihood test method. *Engineering Transactions*, 39(2):139-161, 1991.
- [Pac85] A. Pacut. *Prawdopodobieństwo Teoria Modelowanie w probabilistyce technicznej*. Wydawnictwa Naukowo-Techniczne, Warszawa, 1985.
- [Pav87] T. Pavlidis. *Grafika i przetwarzanie obrazów. Algorytmy*. Wydawnictwa Naukowo-Techniczne, Warszawa, 1987.
- [Pie92a] W. Pieczynski. Parameter estimation in the case of hidden data. In *Proceedings of 16th Biennial Symposium on Communications*, Kingston, Canada, May 1992.
- [Pie92b] W. Pieczynski. Statistical image segmentation. *Machine Graphics & Vision*, 1(1/2):261-281, 1992.
- [Pom93] D. A. Pomerleau. Input recognition reliability estimation. In *Proceedings of the International Workshop on Intelligent Robotic Systems '93*, str. 101-106, Zakopane, lipiec 1993.
- [por87] *Poradnik inżyniera - matematyka tom 2*. Wydawnictwa Naukowo-Techniczne, Warszawa, 1987.
- [Pra91] W. K. Pratt. *Digital Image Processing*. John Wiley & Sons, New York, 2d edition, 1991.
- [QT91] Qian, Titterton. Pixel labeling for three-dimensional scenes based on Markov mesh models. *Signal Processing*, 22(3):313-328, March 1991.
- [RC92] A. Rattarangsi, R. T. Chin. Scale-based detection of corners of planar curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(4):430-449, April 1992.
- [SJ84] A. M. Shan, R. Jain. Detecting time-varying corners. *Computer Vision Graphics and Image Processing*, 28:345-355, 1984.

- [SJ87] I. K. Sethi, R. Jain. Finding trajectories of feature points in a monocular image sequence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(1):56-73, 1987.
- [SJ89] K. Skifstad, R. Jain. Illumination independent change detection for real world image sequences. *Computer Vision Graphics and Image Processing*, 46:387-399, 1989.
- [ŚI91] A. Śluzek. *Komputerowa analiza obrazów*. Wydawnictwa Politechniki Warszawskiej, Warszawa, 1991.
- [SS90] V. Salari, I. K. Sethi. Feature point corespondence in the presence of occlusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):87-91, 1990.
- [TF91] R. Tadeusiewicz, M. Flasiński. *Rozpoznawanie obrazów*. Państwowe Wydawnictwa Naukowe, Warszawa, 1991.
- [TQD86] C. W. Therrien, T. F. Quatieri, D. E. Dudgeon. Statistical model-based algorithms for image analysis. *Proceedings of IEEE*, 74(4):532-551, April 1986.
- [Tru80] H. J. Trussel. The relationship between image restoration by the maximum a posteriori method and a maximum entropy method. *IEEE Transactions on Acoustics Sound and Signal Processing*, 28(1):114-117, February 1980.
- [Tsu93] S. Tsugawa. Vision-based vehicles in japan: The machine vision systems and driving control systems. In *Proceedings of IEEE International Symposium on Industrial Electronics*, str. 278-285, Budapest, Hungary, June 1993.
- [Vei91] A. Veijanen. A simulation-based estimator for hidden Markov random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(8):825-830, August 1991.
- [VP92] A. Verri, T. Poggio. Motion field and optical flow: Qualitative properties. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(3):490-498, March 1992.
- [WHA92] J. Weng, T. S. Huang, N. Ahuja. Motion and structure from line correspondences: Closed-form solution, uniqueness, and optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(3):318-336, March 1992.
- [Wrz93] A. Wrzokowicz. Niejawne modele Markowa HMM w automatycznym rozpoznawaniu mowy. Technical Report 32, IPPT PAN, Warszawa, 1993.
- [ZC90] J. Zerubia, R. Chellappa. Mean field annealing using compound Gauss-Markov fields for edge detection and image restoration. Technical Report 1295, INRIA, October 1990.
- [Zie90] R. Zieliński. *Siedem wykładów wprowadzających do statystyki matematycznej*. Państwowe Wydawnictwa Naukowe, Warszawa, 1990.

Dodatek

Dodatek A

Moduł różnicy jako źródło obserwacji dla nowego modelu z dwuwymiarowym wektorem obserwacji oraz modelu LB

W rozdziale przedstawiono wyniki badań analogicznych do tych z rozdziałów 7 i 8, ale jako obserwację przyjęto moduł różnicy jasności piksła w kolejnych obrazach sekwencji, a nie jak poprzednio samą różnicę. Skorzystanie z absolutnej wartości różnicy jasności powoduje ograniczenie informacji, ponieważ tracona jest wiedza o kierunku zmian jakie zachodzą pomiędzy kolejnymi obrazami analizowanej sekwencji.

A.1 Badania modelu LB

Przyjmując modul różnicy jasności jako obserwację, przeprowadzono cztery serie eksperymentów, po jednej dla każdej z sekwencji K, L, M, N. Wartości parametrów m_1 oraz m_2 estymowano w analogiczny sposób jak te z rozdziału 6. Okazało się, że w przypadku sekwencji K, L, M znalezione wartości parametrów m_1 , m_2 są identyczne. Wyniki eksperymentów zebrano w tab. A.1-A.4.

	MASKA 1			MASKA 2				UWAGI
	$-a$	$-b$	k_1	$-a$	$-b$	p_2	k_2/p_3	
$m_1^0 = 30, m_2^0 = 10$ - wartości estymowane, $\sigma_0^2 = 225$								
	0	12	3	0	52	0	21/21	
m_1	$m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
0	0	6	2	0	7	0	21/21	$m_1 \ll m_2^0$
1	0	6	2	0	7	0	21/21	$m_1 \ll m_2^0$
9	0	8	2	0	9	0	21/21	$m_1 \leq m_2^0$
11	0	12	3	0	125	0	21/21	$m_1 \geq m_2^0$
100	0	1230	25°	0	1184	0	25°/21	$m_1 \gg m_2^0$
m_2	$m_1 = m_1^0, \sigma^2 = \sigma_0^2$							
20	0	10	2	0	28	0	21/21	$m_2 < m_1^0$
25	0	7	2	0	19	0	21/21	$m_2 < m_1^0$
35	0	75	11	0	43	0	21/21	$m_2 > m_1^0$
50	0	214	25°	0	218	0	25°/21	$m_2 \gg m_1^0$
σ^2	$m_1 = m_1^0, m_2 = m_2^0$							
22.5	0	91	8	0	108	0	21/21	$U_z \times 10$
2.25	180	3743	6	329	3671	4	11/10	$U_z \times 100$
2250	0	12	3	0	52	0	21/21	$U_z \times 0.1$

Tab. A.1: Wpływ zmian parametrów modelu LB na wyniki detekcji masek obiektu ruchomego dla sekwencji K z rys. 6.1. Obserwacją modul różnicy jasności, a nie jak w tab. 7.1 różnica jasności. Znaczenie kolumn identyczne jak dla tab. 7.1.

Dla sekwencji K oraz N otrzymano całkiem dobre rezultaty przy estymowanych wartościach parametrów.

A.2 Badania nowego modelu z dwuwymiarowym wektorem obserwacji

Dla nowego modelu przeprowadzono również serię eksperymentów przyjmując jako obserwację modul różnicy jasności. Pełne rezultaty badań dla sekwencji K, L, M, N przedstawiono w tab. A.5-A.8.

	MASKA 1			MASKA 2				UWAGI
	-a	-b	k ₁	-a	-b	p ₂	k ₂ /p ₃	
$m_1^0 = 30, m_2^0 = 10$ - wartości estymowane, $\sigma_0^2 = 225$								
	0	111	15	0	83	0	25*/21	
m_1	$m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
0	0	4	1	0	8	0	21/21	$m_1 \ll m_2^0$
1	0	4	1	0	8	0	21/21	$m_1 < m_2^0$
9	0	6	2	0	13	0	21/21	$m_1 \leq m_2^0$
11	0	193	13	0	116	0	25*/21	$m_1 \geq m_2^0$
100	0	7	2	0	4	0	21/21	$m_1 \gg m_2^0$
m_2	$m_1 = m_1^0, \sigma^2 = \sigma_0^2$							
20	0	85	12	0	31	0	21/21	$m_2 < m_1^0$
25	0	70	12	0	31	0	21/21	$m_2 \leq m_1^0$
35	0	75	10	0	30	0	21/21	$m_2 > m_1^0$
50	0	154	23	0	112	0	21/21	$m_2 \gg m_1^0$
σ^2	$m_1 = m_1^0, m_2 = m_2^0$							
22.5	0	223	13	0	215	0	25*/21	$U_z \times 10$
2.25	317	4212	11	485	4272	3	9/9	$U_z \times 100$
2250	0	111	15	0	83	0	25*/21	$U_z \times 0.1$

Tab. A.2: Wpływ zmian parametrów modelu LB na wyniki detekcji masek obiektu ruchomego dla sekwencji L z rys. 6.3. Obserwacją modul różnicy jasności, a nie jak w tab. 7.2 różnica jasności. Znaczenie kolumn identyczne jak dla tab. 7.1.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = 30, m_2^0 = 10$ - wartości estymowane, $\sigma_0^2 = 225$								
	0	66	10	0	64	0	23/21	
m_1	$m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
0	0	4	1	0	6	0	21/21	$m_1 \ll m_2^0$
1	0	4	1	0	6	0	21/21	$m_1 \ll m_2^0$
9	0	8	3	0	6	0	21/21	$m_1 \leq m_2^0$
11	0	112	18	0	254	0	25*/21	$m_1 \geq m_2^0$
100	0	8	2	0	4	0	21/21	$m_1 \gg m_2^0$
m_2	$m_1 = m_1^0, \sigma^2 = \sigma_0^2$							
20	0	32	8	0	24	0	21/21	$m_2 < m_1^0$
25	0	32	8	0	24	0	21/21	$m_2 \leq m_1^0$
35	0	38	6	0	25	0	21/21	$m_2 > m_1^0$
50	0	142	18	0	57	0	21/21	$m_2 \gg m_1^0$
σ^2	$m_1 = m_1^0, m_2 = m_2^0$							
22.5	0	260	24	0	193	0	24/21	$U_z \times 10$
2.25	374	4186	7	395	4161	0	8/9	$U_z \times 100$
2250	0	66	10	0	64	0	23/21	$U_z \times 0.1$

Tab. A.3: Wpływ zmian parametrów modelu LB na wyniki detekcji masek obiektu ruchomego dla sekwencji M z rys. 6.5. Obserwacją moduł różnicy jasności, a nie jak w tab. 7.3 różnica jasności. Znaczenie kolumn identyczne jak dla tab. 7.1.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = 10, m_2^0 = 20$ - wartości estymowane, $\sigma_0^2 = 225$								
	0	17	3	0	9	0	21/21	
m_1	$m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
0	0	13	3	0	9	0	21/21	$m_1 \ll m_1^0$
5	0	16	3	0	9	0	21/21	$m_1 < m_1^0$
19	0	31	5	0	18	0	21/21	$m_1 \leq m_1^0$
21	0	17	3	0	27	0	21/21	$m_1 \geq m_1^0$
50	0	7	2	0	11	0	21/21	$m_1 \gg m_1^0$
m_2	$m_1 = m_1^0, \sigma^2 = \sigma_0^2$							
0	0	128	12	0	197	0	25*/21	$m_2 < m_2^0$
9	0	49	7	0	102	0	21/21	$m_2 \leq m_2^0$
11	0	16	3	0	9	0	21/21	$m_2 \geq m_2^0$
40	0	36	5	0	18	0	21/21	$m_2 \gg m_2^0$
σ^2	$m_1 = m_1^0, m_2 = m_2^0$							
22.5	0	17	3	0	9	0	21/21	$U_z \times 10$
2.25	568	90	11	507	61	4	25*/25*	$U_z \times 100$
2250	0	17	3	0	9	0	21/21	$U_z \times 0.1$

Tab. A.4: Wpływ zmian parametrów modelu LB na wyniki detekcji masek obiektu ruchomego dla sekwencji N z rys. 6.7. Obserwacją moduł różnicy jasności, a nie jak w tab. 7.4 różnica jasności. Znaczenie kolumn identyczne jak dla tab. 7.1.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_0^0 = 27.73, m_2^0 = 32.27, m_3^0 = 10$ - wartości estymowane, $\sigma_0^2 = 225$								
	1	6	3	0	344	9	13/13	
m_1	$m_2 = m_2^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
0	0	6	3	0	51	3	7/7	$m_1 < m_3^0$
9	0	6	3	0	9	10	5/5	$m_1 \leq m_3^0$
11	1	6	3	0	1943	10	4/4	$m_3^0 \leq m_1 \ll m_1^0$
20	1	6	3	0	1331	10	15/13	$m_3^0 < m_1 < m_1^0$
25	1	6	3	0	484	12	12/15	$m_3^0 < m_1 \leq m_1^0$
30	1	6	3	0	131	10	9/15	$m_1^0 \leq m_1 \leq m_2^0$
35	0	6	3	0	45	22	5/12	$m_2^0 \leq m_1$
50	0	7	4	0	9	18	5/11	$m_2^0 \ll m_1$
100	0	12	3	0	21	4	21/21	$m_2^0 \ll m_1$
m_2	$m_1 = m_1^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
0	21	1	11	0	83	9	24/22	$m_2 \ll m_3^0$
9	16	1	8	0	182	9	11/20	$m_2 \leq m_3^0$
11	1	6	3	0	308	9	7/8	$m_3^0 \leq m_2 \ll m_1^0$
20	0	6	3	0	323	9	9/9	$m_3^0 < m_2 < m_1^0$
25	0	6	3	0	295	9	11/10	$m_3^0 \ll m_2 \leq m_1^0$
30	1	6	3	0	379	9	8/9	$m_1^0 \leq m_2 \ll m_2^0$
35	1	6	3	0	294	9	14/21	$m_2^0 \leq m_2$
50	1	1	1	0	231	9	10/20	$m_2^0 < m_2$
100	7	1	4	0	110	9	12/20	$m_2^0 \ll m_2$
m_3	$m_1 = m_1^0, m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
0	1	6	3	0	1481	19	5/13	$m_3 \ll m_3^0$
15	1	4	2	0	44	21	8/15	$m_3^0 < m_3 < m_1^0$
20	1	4	2	0	20	23	3/11	$m_3^0 < m_3 < m_1^0$
25	0	1	1	0	6	11	4/10	$m_3^0 < m_3 \leq m_1^0$
30	0	1	1	1	1982	25*	3/15	$m_1^0 \leq m_3 \leq m_2^0$
35	0	1	1	2	1992	11	4/7	$m_2^0 \leq m_3$
50	2	1	1	2	1995	8	3/10	$m_2^0 < m_3$
100	1	1	1	1	1998	1	2/2	$m_2^0 \ll m_3$
σ^2	$m_1 = m_1^0, m_2 = m_2^0, m_3 = m_3^0$							
22.5	1	10	3	0	345	14	13/13	$U_z \times 10$
2.25	527	52	6	725	467	6	6/6	$U_z \times 100$
2250	1	6	3	0	344	9	13/13	$U_z \times 0.1$

Tab. A.5: Wpływ zmian parametrów nowego modelu z dwuwymiarowym wektorem obserwacji na wyniki detekcji masek obiektu ruchomego dla sekwencji K z rys. 6.1. Obserwacją moduł różnicy jasności, a nie jak w tab. 7.1 różnica jasności. Znaczenie kolumn identyczne jak dla tab. 7.1.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = 32.27, m_2^0 = 27.73, m_3^0 = 10$ – wartości estymowane, $\sigma_0^2 = 225$								
	0	7	3	0	1	10	7/9	
m_1	$m_2 = m_2^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
0	0	124	14	0	83	4	11/7	$m_1 < m_3^0$
9	0	7	3	0	107	6	11/4	$m_1 \leq m_3^0$
11	1	7	4	1	1973	9	4/9	$m_3^0 \leq m_1 \ll m_2^0$
20	1	7	3	0	1427	11	25*/25*	$m_3^0 < m_1 < m_2^0$
25	0	7	3	0	547	12	11/13	$m_3^0 < m_1 \leq m_2^0$
30	0	7	3	0	34	10	11/16	$m_2^0 \leq m_1 \leq m_1^0$
35	0	7	3	0	1	15	7/9	$m_1^0 \leq m_1$
50	0	7	3	0	1	18	6/14	$m_1^0 \ll m_1$
100	0	8	3	0	83	4	25*/21	$m_1^0 \ll m_1$
m_2	$m_1 = m_1^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
0	0	1	1	2	3	10	5/8	$m_2 \ll m_3^0$
9	0	1	1	2	3	10	5/8	$m_2 \leq m_3^0$
11	0	15	5	0	8	10	7/8	$m_3^0 \leq m_2 \ll m_2^0$
20	0	8	3	0	1	10	7/8	$m_3^0 < m_2 < m_2^0$
25	0	7	3	0	1	10	7/9	$m_3^0 \ll m_2 \leq m_2^0$
30	0	7	3	0	1	10	7/9	$m_2^0 \leq m_2 \ll m_1^0$
35	0	1	1	0	66	10	5/8	$m_1^0 \leq m_2$
50	0	1	1	6	63	10	7/8	$m_1^0 < m_2$
100	0	1	1	10	1	10	6/8	$m_1^0 \ll m_2$
m_3	$m_1 = m_1^0, m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
0	0	8	3	1	616	10	20/20	$m_3 \ll m_3^0$
15	0	1	1	0	1	15	4/13	$m_3^0 < m_3 < m_2^0$
20	0	1	1	0	1	15	3/9	$m_3^0 < m_3 < m_2^0$
25	0	1	1	0	1	9	4/12	$m_3^0 < m_3 \leq m_2^0$
30	0	9	5	0	14	9	6/14	$m_2^0 \leq m_3 \leq m_1^0$
35	0	9	5	1	1996	13	5/7	$m_1^0 \leq m_3$
50	0	9	5	1	2007	18	5/10	$m_1^0 < m_3$
100	0	9	5	1	2007	5	5/5	$m_1^0 \ll m_3$
σ^2	$m_1 = m_1^0, m_2 = m_2^0, m_3 = m_3^0$							
22.5	0	7	3	0	5	10	7/17	$U_z \times 10$
2.25	2059	40	5	2221	33	4	5/6	$U_z \times 100$
2250	0	7	3	0	1	10	7/9	$U_z \times 0.1$

Tab. A.6: Wpływ zmian parametrów nowego modelu z dwuwymiarowym wektorem obserwacji na wyniki detekcji masek obiektu ruchomego dla sekwencji L z rys. 6.3. Obserwacją moduł różnicy jasności, a nie jak w tab. 7.2 różnica jasności. Znaczenie kolumn identyczne jak do tab. 7.1.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = 32.27, m_2^0 = 27.73, m_3^0 = 10$ - wartości estymowane, $\sigma_0^2 = 225$								
	0	3	2	0	25	9	9/15	
m_1	$m_2 = m_2^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
0	0	75	12	0	57	3	13/7	$m_1 < m_3^0$
9	0	74	12	0	57	7	11/7	$m_1 \leq m_3^0$
11	0	3	2	0	1908	13	4/8	$m_3^0 \leq m_1 < m_2^0$
20	0	3	2	0	1352	14	7/11	$m_3^0 < m_1 < m_2^0$
25	0	3	2	0	604	17	11/13	$m_3^0 < m_1 \leq m_2^0$
30	0	3	2	0	24	10	11/12	$m_2^0 \leq m_1 \leq m_1^0$
35	0	3	2	0	1	9	9/12	$m_1^0 \leq m_1$
50	0	3	2	0	1	18	5/13	$m_1^0 \leq m_1$
100	0	38	12	0	77	3	24/20	$m_1^0 \leq m_1$
m_2	$m_1 = m_1^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
0	14	1	7	0	3	9	8/15	$m_2 \leq m_3^0$
9	8	1	4	0	3	9	8/15	$m_2 \leq m_3^0$
11	0	3	2	0	34	9	10/15	$m_3^0 \leq m_2 < m_2^0$
20	0	3	2	0	25	9	9/15	$m_3^0 < m_2 < m_2^0$
25	0	3	2	0	25	9	9/15	$m_3^0 \leq m_2 \leq m_2^0$
30	0	3	2	0	23	9	9/15	$m_2^0 \leq m_2 \leq m_1^0$
35	0	3	2	0	77	9	9/20	$m_1^0 \leq m_2$
50	0	3	2	0	60	9	10/15	$m_1^0 < m_2$
100	4	3	2	0	1	9	9/15	$m_1^0 \leq m_2$
m_3	$m_1 = m_1^0, m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
0	0	3	2	0	695	11	12/14	$m_3 \leq m_3^0$
15	0	3	2	0	1	12	9/13	$m_3^0 < m_3 < m_2^0$
20	0	3	2	0	1	11	6/9	$m_3^0 < m_3 < m_2^0$
25	0	3	2	0	1	15	6/12	$m_3^0 < m_3 \leq m_2^0$
30	0	17	5	0	10	15	9/16	$m_2^0 \leq m_3 \leq m_1^0$
35	3	19	6	3	1991	19	4/9	$m_1^0 \leq m_3$
50	3	19	6	2	1991	15	4/10	$m_1^0 < m_3$
100	3	19	6	2	1987	6	4/4	$m_1^0 \leq m_3$
σ^2	$m_1 = m_1^0, m_2 = m_2^0, m_3 = m_3^0$							
22.5	0	3	2	0	1	9	12/15	$U_z \times 10$
2.25	2104	27	4	2242	60	4	7/7	$U_z \times 100$
2250	0	3	2	0	25	9	9/15	$U_z \times 0.1$

Tab. A.7: Wpływ zmian parametrów nowego modelu z dwuwymiarowym wektorem obserwacji na wyniki detekcji masek obiektu ruchomego dla sekwencji M z rys. 6.5. Obserwacją moduł różnicy jasności, a nie jak w tab. 7.3 różnica jasności. Znaczenie kolumny identyczne jak dla tab. 7.1.

	MASKA 1			MASKA 2				UWAGI
	$\neg a$	$\neg b$	k_1	$\neg a$	$\neg b$	p_2	k_2/p_3	
$m_1^0 = 10, m_2^0 = 10, m_3^0 = 20$ - wartości estymowane, $\sigma_0^2 = 225$								
	0	5	3	0	43	16	4/14	
m_1	$m_2 = m_2^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
0	0	5	3	0	8	11	3/8	$m_1 < m_2^0$
8	0	5	3	0	9	13	3/21	$m_1 \leq m_2^0$
12	0	5	3	0	60	16	4/12	$m_2^0 \leq m_1 < m_3^0$
15	0	5	3	0	97	17	4/12	$m_2^0 < m_1 < m_3^0$
25	0	10	3	0	505	10	11/20	$m_3^0 < m_1$
40	0	14	4	0	2	7	7/7	$m_3^0 \ll m_1$
m_2	$m_1 = m_1^0, m_3 = m_3^0, \sigma^2 = \sigma_0^2$							
0	0	4	3	0	5	16	6/14	$m_2 \ll m_1^0$
8	0	4	3	0	39	16	3/14	$m_2 \leq m_1^0$
12	0	11	5	0	43	16	4/14	$m_2^0 \leq m_2 < m_3^0$
15	0	11	5	0	51	16	6/14	$m_2^0 < m_2 < m_3^0$
25	0	3	2	0	70	16	9/14	$m_3^0 < m_2$
40	0	1	1	0	16	16	10/14	$m_3^0 \ll m_2$
m_3	$m_1 = m_1^0, m_2 = m_2^0, \sigma^2 = \sigma_0^2$							
0	0	1	1	0	2000	3	1/4	$m_3 \ll m_1^0$
8	0	6	3	0	1982	5	3/4	$m_3 \leq m_1^0$
12	0	4	3	0	8	14	3/10	$m_1^0 < m_3 < m_3^0$
15	0	4	3	0	8	14	3/14	$m_1^0 < m_3 < m_3^0$
25	0	11	5	0	111	17	6/12	$m_2^0 < m_3$
40	0	1	1	2	1781	10	11/11	$m_2^0 \ll m_3$
σ^2	$m_1 = m_1^0, m_2 = m_2^0, m_3 = m_3^0$							
22.5	0	5	3	0	43	16	4/14	$U_z \times 10$
2.25	558	6	11	513	53	12	13/9	$U_z \times 100$
2250	0	5	3	0	43	16	4/14	$U_z \times 0.1$

Tab. A.8: Wpływ zmian parametrów nowego modelu z dwuwymiarowym wektorem obserwacji na wyniki detekcji masek obiektu ruchomego dla sekwencji N z rys. 6.7. Obserwacją modul różnicy jasności, a nie jak w tab. 7.4 różnica jasności. Znaczenie kolumn identyczne jak dla tab. 7.1.

Dla sekwencji K, przy estymowanych wartościach parametrów, otrzymano złe rezultaty, przeciwnie niż w przypadku sekwencji L oraz M. Niestety model stał się bardzo wrażliwy na wartości parametru m_1 . Jednocześnie wzrosła wrażliwość na wartość parametru m_3 . W przypadku sekwencji L wpływ parametru m_2 jest niezauważalny, dla sekwencji M akceptowalny, a dla sekwencji K praktycznie całkowicie negatywny. W przypadku parametru σ^2 odpowiadającego za poziom energii zgodności, zastosowanie jako obserwacji modułu różnicy jasności nie wprowadziło praktycznie żadnych zmian. Jeśli tylko model, przy estymowanych wartościach parametrów, pozwalał na znalezienie stabilnego rozwiązania, to zmiany parametru σ^2 o rząd wielkości praktycznie nie mają żadnego wpływu.

Dodatek B

Opis pakietu programów do detekcji ruchu

Chęć przeprowadzenia badań modelu Lalande'a-Bouthemy'iego oraz nowych modeli zaproponowanych w rozdziałach 5, 10 stworzyła konieczność opracowania pakietu programów do przetwarzania obrazów, przeznaczonego do detekcji masek obiektów ruchomych. W niniejszym dodatku opisana jest koncepcja oraz podstawowe założenia poszczególnych programów pakietu. Dość szczegółowo opisano sposób wymiany informacji pomiędzy programami.

B.1 Koncepcja pakietu

Metody detekcji i śledzenia ruchu (detekcji masek obiektu ruchomego) omawiane w niniejszej rozprawie składają się z dwóch zasadniczych etapów, a mianowicie:

- detekcji maski zmian zachodzących pomiędzy dwoma kolejnymi obrazami sekwencji,
- właściwej detekcji maski poruszającego się obiektu.

Przeprowadzone eksperymenty pokazały, że w przypadku sekwencji rzeczywistych dodatkowo konieczna jest filtracja masek zmian oraz znalezionych masek poruszającego się obiektu.

Detekcję zmian zachodzących pomiędzy kolejnymi obrazami sekwencji umożliwia program *CHANGE*. Wstępną filtrację masek zmian w obrazie i ostateczną filtrację masek obiektu można przeprowadzić za pomocą programu *MEDIAN* lub programu *BINFILTR*. Do znajdowania masek obiektów ruchomych służą: program *ICM*, będący implementacją metody opartej na modelu LB, program *ICM2*, którego algorytm poszukiwania masek obiektu ruchomego korzysta z nowego modelu z dwuwymiarowym wektorem obserwacji, zaproponowanego w rozdziale 5 oraz program *ICM4* będący implementacją metody korzystającej z wprowadzonego w rozdziale 10 modelu z trójwymiarowym wektorem obserwacji.

Poza wymienionymi programami powstał kilka programów pomocniczych takich jak *GREY2MSK* – pomagający użytkownikowi w ręcznym znajdowaniu maski obiektów ruchomych, *SHOW_CON* – nakładający obrys znalezionej maski binarnej na źródłowy obraz z gradacją szarości oraz *GAUSS2D* znajdujący wartości parametrów dwuwymiarowych rozkładów normalnych – rozkładów zgodności dla programu *ICM4*.

B.2 Pliki tekstowe używane do wymiany informacji pomiędzy programami

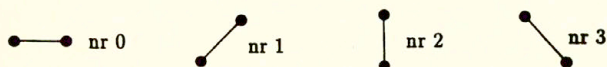
W pakiecie używane są pliki tekstowe o dwóch różnych formatach.

Format 1 służy do opisu przetwarzanej sekwencji. W pierwszym wierszu umieszczana jest długość sekwencji. W kolejnych wierszach wymienione są nazwy kolejnych obrazów sekwencji. Oczywiście konsekwencją jest to, że liczba wierszy w takim pliku zależy od długości sekwencji.

Format 2 odnosi się do pliku parametrów pola Markowa. Ponieważ badania detekcji ruchu zostały zainspirowane pracami Lalande'a-Bouthemy'ego, parametry modelu są zbliżone do tych zaprezentowanych w literaturze, jednak z wieloma rozszerzeniami. Wszystkie parametry pola losowego używanego do detekcji ruchu można podzielić na trzy grupy:

- parametry energii przestrzennej,
- parametry energii czasowej,
- parametry energii zgodności.

Ponieważ przyjęto rząd pola Markowa równy 2, więc istnieją cztery rodzaje klik dwupik-slowych. Uwaga, dla potrzeb systemu komputerowego kliki dwupikslowe zostały ponumerowane w następujący sposób:



Cztery pierwsze wiersze pliku parametrów są przeznaczony na bezwzględne wartości parametrów β , dla kolejnych klik od 0 do 3; dla modelu LB w pliku parametrów należy w czterech kolejnych wierszach wpisać te same liczby.

Następnych osiem wierszy to parametry energii czasowej. Energia czasowa ma postać energii warunkowej w zależności od etykiety maski zmian $\bar{\alpha}_k$. W pliku parametrów należy więc podać w przedstawionej poniżej kolejności wartości energii dla poszczególnych realizacji:

1. - $(b, b, 0)$,
2. - $(b, b, 1)$,
3. - $(a, b, 0)$,
4. - $(a, b, 1)$,
5. - $(b, a, 0)$,
6. - $(b, a, 1)$,
7. - $(a, a, 0)$,
8. - $(a, a, 1)$,

przy czym: 0 - brak zmian, 1 - zmiany są.

Kolejne wiersze to parametry energii zgodności. Liczba tych parametrów zależy od wersji modelu. W modelu LB realizacjom (a, b) oraz (b, a) odpowiada jeden parametr energii zgodności (wersja 1.0 algorytmu - program ICM). W wersji 2.0 (program ICM2) realizacje (a, b) i (b, a) są rozróżniane, a to powoduje, że pliki parametrów programów implementujących poszczególne wersje algorytmów nie są wymienne. W wersji 4.0 (program ICM4) liczba parametrów energii zgodności jest znacznie większa, dlatego plik z parametrami jest dłuższy.

Poniżej przedstawiono kolejność parametrów energii zgodności dla wersji 1.0, tzn. dla metody opartej na modelu LB:

- m_0 - wartość średnia dla realizacji (b, b) .
- σ_0^2 - wariancja dla realizacji (b, b) ,
- m_1 - wartość średnia dla realizacji (a, b) oraz (b, a) ,
- σ_1^2 - wariancja dla realizacji (a, b) oraz (b, a) .
- m_2 - wartość średnia dla realizacji (a, a) ,
- σ_2^2 - wariancja dla realizacji (a, a) .

Jeżeli, tak jak zostało to przedstawione w rozdziale 5, używany jest jeden parametr σ^2 , to w wierszach odpowiadającym parametrom σ_0^2 , σ_1^2 oraz σ_2^2 należy wpisać tę samą wartość. UWAGA w pracach [BL90, LB90] parametry m_1 i m_2 są ze sobą zamienione.

Ponizej przedstawiono wartość parametrów energii zgodności dla wersji 2.0, tzn. dla metody opartej na nowym modelu z dwuwymiarowym wektorem obserwacji:

- m_0 - wartość średnia dla realizacji (b, b) ,
- σ_0^2 - wariancja dla realizacji (b, b) ,
- m_1 - wartość średnia dla realizacji (a, b) ,
- σ_1^2 - wariancja dla realizacji (a, b) ,
- m_2 - wartość średnia dla realizacji (b, a) ,
- σ_2^2 - wariancja dla realizacji (b, a) ,
- m_3 - wartość średnia dla realizacji (a, a) ,
- σ_3^2 - wariancja dla realizacji (a, a) .

Ponownie, jeżeli, tak jak zostało to przedstawione w rozdziale 5, używany jest jeden parametr σ^2 , to wierszach odpowiadającym parametrom $\sigma_0^2, \sigma_1^2, \sigma_2^2$ oraz σ_3^2 należy wpisać tę samą wartość.

Ponizej przedstawiono wartość parametrów zgodności dla wersji 4.0, tzn. dla metody opartej na modelu z trójwymiarowym wektorem obserwacji:

- m_{01} - wartość średnia różnicy o_k dla realizacji (b, b) ,
- m_{02} - wartość średnia jasności f_k dla realizacji (b, b) ,
- σ_{01}^2 - wariancja różnicy o_k dla realizacji (b, b) ,
- σ_{02}^2 - wariancja jasności f_k dla realizacji (b, b) ,
- ρ_0 - współczynnik korelacji o_k z f_k dla realizacji (b, b) ,
- m_{11} - wartość średnia różnicy o_k dla realizacji (a, b) ,
- m_{12} - wartość średnia jasności f_k dla realizacji (a, b) ,
- σ_{11}^2 - wariancja różnicy o_k dla realizacji (a, b) ,
- σ_{12}^2 - wariancja jasności f_k dla realizacji (a, b) ,
- ρ_1 - współczynnik korelacji o_k z f_k dla realizacji (a, b) ,
- m_{21} - wartość średnia różnicy o_k dla realizacji (b, a) ,
- m_{22} - wartość średnia jasności f_k dla realizacji (b, a) ,
- σ_{21}^2 - wariancja różnicy o_k dla realizacji (b, a) ,
- σ_{22}^2 - wariancja jasności f_k dla realizacji (b, a) ,
- ρ_2 - współczynnik korelacji o_k z f_k dla realizacji (b, a) ,
- m_{31} - wartość średnia różnicy o_k dla realizacji (a, a) ,
- m_{32} - wartość średnia jasności f_k dla realizacji (a, a) ,
- σ_{31}^2 - wariancja różnicy o_k dla realizacji (a, a) ,
- σ_{32}^2 - wariancja jasności f_k dla realizacji (a, a) ,
- ρ_3 - współczynnik korelacji o_k z f_k dla realizacji (a, a) .

Wszystkie pliki tekstowe używane bądź tworzone przez poszczególne programy pakietu mają na sztywno określone nazwy, a mianowicie:

- *LISTA.LST* plik opisujący wejściową sekwencję obrazów z gradacją szarości, przedstawiającą ruchomy obiekt - format 1,

- *PARAMS.PAR* plik zawierający parametry modelu (uwaga na wersję algorytmu) - format 2,
- *LISTATC.LST* plik opisujący sekwencję masek zmian tworzoną przez program o nazwie *CHANGE*, a używaną przez inne programy - format 1,
- *LISTAMSK.LST* plik opisujący sekwencję znalezionych masek - format 1.

B.3 Opis zasadniczych programów wchodzących w skład pakietu

Każdy z programów dostarczany jest w trzech wersjach: w wersji podstawowej, wersji na procesor 286 i minimum 3MB pamięci i w wersji na procesor 386 i minimum 4MB wolnej pamięci. Wersja podstawowa pozwala na przetwarzanie co najwyżej sekwencji trzech obrazów (skompilowana została pod BORLAND C wersja 3.1). Wersja 286 została skompilowana przy użyciu DOS-extendera LITE286. Pozwala ona na przetwarzanie sekwencji o długości do 20 obrazów. Niestety programy skompilowane w tej wersji działają znacznie wolniej niż pozostałe. Wersja 386 została skompilowana pod ZORTECH C wersja 3.1. Daje ona możliwość przetwarzania sekwencji do 40 obrazów.

W całym pakiecie zakłada się, że wczytywane obrazy zapisane są w standardzie *TIFF* oraz, że rozmiar ich wynosi dokładnie 256×256 pikseli.

Programy mają dwa tryby pracy:

- **konwersacyjny** - możliwe jest oglądanie wczytywanych obrazów, ręczne ustawianie poszczególnych parametrów, jak również oglądanie obrazów wynikowych. W niektórych programach wypisywane są informacje o obliczeniach cząstkowych.
- **wsadowy** - program wywoływany w pliku do przetwarzania wsadowego systemu MS-DOS. W tym trybie wymagane jest podanie dodatkowych parametrów za nazwą wywoływanego programu. Wymagane parametry zostały opisane poniżej razem z krótkim omówieniem każdego programu.

UWAGA: Programy w wersji 386 nie działają w trybie konwersacyjnym.

Każdy z programów ma możliwość sporządzania protokołu z wykonywanych operacji (w trybie batchowym opcja ta ustawiona jest domyślnie). Protokół dopisywany jest do pliku o ustalonej nazwie *PROTOKOL.TXT*.

W przypadku wykrycia błędu każdy z programów zapisuje komunikat o błędzie w pliku o nazwie *ERROR.TXT*.

Wywołanie każdego z programów z parametrem */?* wyświetla na ekranie krótką informację o sposobie korzystania z niego.

B.3.1 Program CHANGE

Program ten znajduje sekwencję masek zmian w obrazie odpowiadających przetwarzanej sekwencji, o której informacja zapisana jest w pliku *LISTA.LST*. Ponieważ maska zmian

czasowych odpowiada dwóm kolejnym obrazom, dlatego obraz wykrytych zmian jest zapisany pod nazwą pierwszego z dwóch porównywanych obrazów z dołączonymi na końcu dwoma literami – *tc* (zmiany czasowe). Program poza, odpowiednimi obrazami, tworzy również plik *LISTATC.LST*. W programie zaimplementowano pięć różnych metod detekcji maski zmian. Dla czterech z tych metod znajdowane zmiany mogą być zaznaczane na obrazach wyjściowych w dwojaki sposób:

- po wykryciu w masce zmian zaznaczane są wszystkie piksele okna,
- po wykryciu zmian tylko środkowy piksel jest zaznaczany.

W trybie wsadowym wyboru dokonuje się pisząc jedno ze słów: *jeden* albo *całe*.

Najprostszą z dostępnych metod detekcji zmian jest metoda modułu różnicy dwóch wartości funkcji jasności w danym pikselu (wymagany parametr: wartość progu).

Dwie kolejne metody to testy liniowy i kwadratowy Hsu-Nagela-Rekera. Poza wartością progu metody te wymagają podania wielkości okna testowego (zakłada się, że jest to okno kwadratowe).

Następną metodą jest test Skifstada-Jaina. Metoda ta wymaga podania wartości progu (w przybliżeniu równej 0) oraz wielkości okna testowego.

Ostatnia metoda to test Nieniewskiego-Pathaka oparty na rozkładzie funkcji jasności na funkcje Walsh'a (na sztywno przyjmuje się wielkość okna testowego na 4×4). Wymaganym parametrem jest wartość progu.

B.3.2 Program MEDIAN

Program ten filtruje metodą mediany sekwencję masek, których nazwy zapisane są w pliku *LISTA.LST*. Poprawne wywołanie wymaga podania liczby nieparzystej z przedziału [3, 9] określającej wielkość okna filtru medianowego.

B.3.3 Program BINFILTR

Program jest implementacją algorytmu przedstawionego w rozdziale 9. W obrazie wyjściowym pozostawiane są tylko te piksele, które należą do konturu lub do wnętrza. Program może działać zarówno w trybie wsadowym jak i konwersacyjnym. W trybie wsadowym wymagane jest podanie dwóch parametrów: nazwy obrazu filtrowanego oraz nazwy pod jaką ma być zapisany obraz przefiltrowany. Niestety nie istnieją wersje na procesory 286 oraz 386.

B.3.4 Program ICM

Program powyższy jest implementacją algorytmu ICM (wersja 1.0 – dla modelu LB). Jednym z podstawowych parametrów jest liczba iteracji w każdym kroku jaka ma być wykonana dla każdego z obrazów. Kolejny parametr określa co jest obserwacją: *roznica* czy *moduł*. Trzeci z parametrów określa sposób skanowania obrazów: *kodig* czy *normal*. Program wymaga czterech plików o sztywno ustalonych nazwach: *PARAMS.PAR* – plik

z parametrami, *LISTA.LST* – nazwy obrazów sekwencji wejściowej, *LISTATC.LST* – nazwy obrazów sekwencji masek zmian, *MASKA1.TIF* – maska inicjująca korekcję maski obiektu dla pierwszego obrazu sekwencji.

B.3.5 Program ICM2

Powyższy program jest implementacją algorytmu ICM w wersji 2.0 (nowy model z dwuwymiarowym wektorem obserwacji). Plik parametrów *PARAMS.PAR* różni się w stosunku do programu ICM. Poza tym wszystko jest identyczne.

B.3.6 Program ICM4

Powyższy program jest implementacją algorytmu ICM w wersji 4.0 (nowy model z trójwymiarowym wektorem obserwacji). Plik parametrów *PARAMS.PAR* ma inną postać niż w przypadku programów ICM oraz ICM2. Poza tym wszystko jest identyczne.

B.4 Opis pomocniczych programów wchodzących w skład pakietu

B.4.1 Program GREY2MSK

Program pomaga w utworzeniu obrazu binarnego odpowiadającego wejściowemu obrazowi z gradacją szarości. Obraz binarny, czyli maska, tworzona jest przez użytkownika. Po wczytaniu obrazu z gradacją szarości należy wybrać okno, w którym będzie tworzona maska obiektu. Następnie okno to zostanie powiększone i pokazane w prawej części ekranu. Klawiszami strzałek można przesuwać kursor. Zaznaczenie pojedynczego piksla odbywa się poprzez wciśnięcie klawisza ENTER. W razie potrzeby odznaczanie piksla również odbywa się za pomocą tego klawisza. Możliwe jest zarówno zaznaczanie całych wierszy jak i kolumn za pomocą klawiszy funkcyjnych. Po zakończeniu tworzenia maski można ją następnie zapisać na dysku wybierając odpowiednią opcję programu.

B.4.2 Program SHOW_CON

Program przeznaczony jest do nakładania konturu z obrazu binarnego na obraz z gradacją szarości. Nakładany obraz jest filtrowany. Filtracja może być dokonywana za pomocą prostego filtru kontur-wnętrze opisanego w rozdziale 9 lub może ona tylko polegać na usuwaniu izolowanych piksli.

B.4.3 Program GAUSS2D

Program oblicza wartości parametrów zgodności, tzn. średnie, wariancje oraz korelacje, dla modelu z trójwymiarowym wektorem obserwacji, korzystając z metody największej wiarygodności [Mor90]. Program wymaga istnienia na dysku w aktualnej kartotece dwóch plików *LISTA.LST* będącego spisem wejściowej sekwencji obrazów z gradacją szarości oraz

LISTAMSK.LST będącym spisem masek binarnych odpowiadających obrazom sekwencji z gradacją szarości (maski te mogą być np. znalezione za pomocą programu *GREY2MSK*). Ze względu na dużą ilość wymaganej pamięci, obydwie sekwencje nie mogą być dłuższe niż 10 obrazów. Efektem działania programu jest plik *WYNIK.TXT*, w którym wypisane są wartości wszystkich znalezionych parametrów.



56612