Akademia Górniczo-Hutnicza
im. Stanisława Staszica w Krakowie

Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej

KATEDRA AUTOMATYKI

PRACA INŻYNIERSKA

JAKUB GOLA, ZBIGNIEW TEKIELA

METODY INICJALIZACJI MODELU TŁA

PROMOTOR:
dr hab. inż. Marek Gorgoń

Kraków 2012

OŚWIADCZENIE AUTORA PRACY

OŚWIADCZAM, ŚWIADOMY ODPOWIEDZIALNOŚCI KARNEJ ZA POŚWIADCZENIE NIEPRAWDY, ŻE NINIEJSZĄ PRACĘ DYPLOMOWĄ WYKONAŁEM OSOBIŚCIE I SAMODZIELNIE, I NIE KORZYSTAŁEM ZE ŹRÓDEŁ INNYCH NIŻ WYMIENIONE W PRACY.

. . . . . .

PODPIS

AGH

University of Science and Technology in Krakow

Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering

DEPARTMENT OF AUTOMATICS

BACHELOR OF SCIENCE THESIS

JAKUB GOLA, ZBIGNIEW TEKIELA

BACKGROUND MODEL INITIALIZATION METHODS

SUPERVISOR:
Marek Gorgoń Ph.D

Krakow 2012

Serdecznie dziękuję . . . tu ciąg dalszych podziękowań np. dla promotora, żony, sąsiada itp.

Spis treści

1. Wstęp

W dzisiejszych czasach systemy wizyjne znajdują coraz więcej zastosowań w życiu codziennym. Wraz ze wzrostem mocy obliczeniowej współczesnych komputerów oraz coraz niższych cen kamer i sprzętu wizyjnego można zaobserwować dynamiczny rozwój tej dziedziny informatyki. Są one używane między innymi w systemach inteligentnego obszarowego sterowania ruchem, monitoringu przestrzeni miejskiej oraz w bardziej zaawansowanych rozwiązaniach CCTV, jak na przykład systemy śledzenia potencjalnych zagrożeń na lotniskach lub w miejscach publicznych. We wszystkich wyżej wymienionych zastosowaniach kluczową kwestią jest oddzielenie pierwszego planu od tła. Odseparowanie dynamicznych elementów obrazu od statycznej scenerii jest podstawą działania wszystkich algorytmy śledzących.

Pomimo ciągłego rozwoju algorytmów stosowanych do inicjalizacji modelu tła nadal istnieją sytuacje w których algorytmy te nie są w stanie wygenerować poprawnych rezultatów. Największe problemy sprawiają zmienne warunki oświetlenia oraz drobne ruchy obiektów tła (np. liście na wietrze). Kolejnym niepożądanym zjawiskiem jest wtapianie się obiektów z pierwszego planu w tło, gdy pozostają one przez długi czas nieruchome. Nieustannie trwają prace nad stworzeniem algorytmów eliminujących lub minimalizujących skutki wyżej wymienionych zjawisk. Próby te zostały opisane między innymi w [BVC10] oraz [RSL09]. Metody te zaliczają się do klasy metod przestrzennych (blokowych), a ich zaimplementowanie i porównanie będzie stanowić jeden z celów niniejszej pracy. Następnie zostaną one skonfrontowane z metodami operującymi na poziomie pikseli - opisanymi w [SG99] oraz [WS06].

2. Metody operujące na poziomie pikseli

2.1. Mixture of Gaussians (MOG)

2.1.1. Wprowadzenie

Tło sceny zawiera wiele dynamicznych obiektów jak poruszane na wietrze gałęzie i liście drzew czy krzewów. Zmiany w obrazie nimi wywołane powodują, że wartość intensywności danego piksela w czasie mogą się bardzo różnić od siebie. Z tego powodu wykorzystanie pojedynczego przybliżenia rozkładu prawdopodobieństwa przy użyciu krzywej Gaussa daje złe rezultaty. Zamiast tego w opisywanej metodzie użyto podejścia bazującego na wykorzystaniu kilku rozkładów Gaussa o różnych parametrach w celu zamodelowania takich zmian [SG99].

Standardowe adaptacyjne modele tła polegają na tworzeniu aproksymacji tła, które jest podobne do obecnej statycznej sceny za wyjątkiem miejsc w których odbył się ruch. Podejście takie jest efektywne w sytuacjach gdy obiekty poruszają się stale, a tło jest widoczne znaczną część czasu, jednakże nie sprawdza się ono dla scen zawierających dużo poruszających się obiektów, a w szczególności, gdy obiekty te poruszają się powoli. Nie potrafi także poradzić sobie z tłami posiadającymi rozkład dwumodalny, powoli odtwarza tło jeśli zostanie ono odkryte i ma jeden ustalony próg dla całej sceny.

W opisywanej metodzie zamiast modelować wartości wszystkich pikseli jako wyłącznie jeden typ rozkładu, wartości dla każdego piksela modelowane są mieszanka rozkładów Gaussa. Bazując na trwałości i wariancji każdego użytego do mieszanki rozkładu Gaussa określane jest które z nich mogą odnosić się do kolorów tła. Wartości pikseli, które nie mieszczą się w żadnym rozkładzie tła uznawane są za piksele należące do pierwszego planu. Taki stan pikseli utrzymuje się do czasu kiedy zaczną wystarczająco dobrze pasować do któregoś z rozkładów tła.

Opisywana metoda dostosowuje się aby radzić sobie ze zmianami oświetlenia, powtarzalnymi ruchami elementów tła, powolnie poruszającymi się elementami pierwszego planu oraz wprowadzaniu lub usuwaniu elementów ze sceny. Powolnie poruszające się elementy potrzebują więcej czasu aby wpasować się w tło, ponieważ rozkład do którego pasują ma większą wariancję niż tło. Powtarzające się zmiany również są uwzględniane, a model rozkładu tła jest utrzymywany nawet jeśli jest chwilowo zamieniony przez inny rozkład, co prowadzi do szybszego odtworzenia tła w przypadku usunięcia obiektów ze sceny. Metoda ta wykorzystuje dwa podstawowe parametry:

a – stała uczenia

T – porcja danych jaka powinna być uwzględniona w tle

Zakładając, że każdy piksel obrazu będzie pochodził z jednej płaszczyzny przy zmiennym świetleniu, to do wydzielenia tła z takiego obrazu wystarczy użycie pojedynczej adaptacyjnej aproksymacji rozkładem Gaussa dla każdego piksela. Jednakże w rzeczywistych warunkach na obrazie występuje wiele powierzchni oraz zmienne oświetlenie, co sprawia, że potrzebne staje się użycie kilku adaptacyjnych rozkładów Gaussa. W opisywanej metodzie użyta jest mieszanka kilku takich rozkładów o różnych parametrach. Za każdym razem, gdy parametry rozkładu są aktualizowane, następuje proces oceny dostępnych rozkładów w celu określenia tego najbardziej prawdopodobnego.

2.1.2. Opis metody

2.1.2.1. Model mieszanki

Niech kolejne wartości danego piksela w czasie nazywają się historią piksela. Zatem historia piksela jest to seria wartości danego piksela, gdzie dla obrazów w skali szarości są to wartości skalarne, a dla obrazów kolorowych są to wektory. O danym pikselu $\{x_0, y_0\}$ w danej chwili czasu t, można powiedzieć, że znana jest jego historia

$$\{X_1, \ldots, X_t\} = \{I(x_0, y_0, i) : 1 \leq i \leq t\} \tag{2.1}$$

gdzie I jest sekwencją ramek.

Niedawna historia dla każdego piksela modelowana jest mieszaniną K rozkładów Gaussa. Prawdopodobieństwo zaobserwowania obecnego piksela określa się wzorem

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}) \tag{2.2}$$

gdzie K jest ilością rozkładów, $\omega_{i,t}$ jest oszacowaną wagą i-tego rozkładu w mieszance w chwili t, $\mu_{i,t}$ jest średnią wartością i-tego rozkładu w mieszance w chwili t, $\Sigma_{i,t}$ jest macierzą kowariancji i-tego rozkładu w mieszance w chwili t,η jest funkcją gęstości prawdopodobieństwa Gaussa

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t - \mu_t)^T \Sigma^{-1} (W_t - \mu_t)} \tag{2.3}$$

gdzie K określane jest przez dostępną pamięć oraz moc obliczeniową jednostki na której wykonywany jest algorytm. Najczęściej spotykana wartość mieści się w zakresie od 3 do 5. W celu zmniejszenia ilości obliczeń przyjmuje się, że macierz kowariancji ma postać:

$$\Sigma_{k,t} = \sigma_k^2 I \tag{2.4}$$

Takie rozwiązanie zakłada, że wartości dla poszczególnych kolorów składowych każdego piksela mają taką samą wariancję.Założenie to obciążone jest pewnymi błędami, lecz pozwala ominąć odwracanie macierzy, co jest zadaniem bardzo kosztownym, za cenę mniejszej precyzji.

Zatem rozkład ostatnio obserwowanych wartości dla każdego piksela w obrazie opisany jest mieszanką rozkładów Gaussa. Nowa wartość piksela będzie na ogół reprezentowana przez jeden z głównych składników mieszanki.

Do obliczenia wartości nowego piksela użyta została metoda on-line K-means approximation. Każda wartość piksela jest sprawdzana pod kątem dopasowania do jednego z K rozkładów. Warunkiem przynależności jest wartość w zakresie do 2,5 odchylenia standardowego z danego rozkładu. Zmiana opisanego wcześniej progu ma niewielki wpływ na wydajność algorytmu. Opisany sposób wyboru odpowiednich pikseli jest bardzo użyteczny dla obszarów z różnym oświetleniem, ponieważ obiekty znajdujące się w zacienionych obszarach mają mniejszy szum niż obiekty znajdujące się w jaśniejszych regionach.

W przypadku gdy żaden rozkład nie został dopasowany do danego piksela, to rozkład z najmniejszą wagą jest zastępowany nowym z wartością piksela jako nową wartością oczekiwaną, dużą wariancją i niską wagą.

Wcześniejsze wagi K rozkładów w czasie t są aktualizowane wg wzoru

$$\omega_{k,t} = (1 - \alpha)\omega_{k,t-1} + \alpha(M_{k,t}) \tag{2.5}$$

gdzie α to tempo uczenia, a $M_{k,t}$ jest 1 dla dopasowanego rozkładu i 0 dla pozostałych rozkładów. Po dokonaniu aproksymacji wagi są normalizowane. $\frac{1}{\alpha}$ oznacza stałą czasową określającą prędkość z jaką parametry rozkładów się zmieniają.

Dla rozkładów niedopasowanych do danego piksela wartości μ i σ pozostają niezmienione. Dla rozkładów, które zostały dopasowane wartości μ i σ obliczane są następująco:

$$\mu_t = (1 - \rho)\mu_{t-1} + \rho X_t \tag{2.6}$$

$$\sigma_t^2 = (1 - \rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T(X_t - \mu_t) \tag{2.7}$$

przy czym

$$\rho = \alpha\eta(X_t|\mu_k, \sigma_k) \tag{2.8}$$

Jedną z największych zalet opisanego rozwiązania jest fakt, że kiedy jakiś obiekt zostanie dodany do tła, to nie niszczy on istniejącego modelu tła. Oryginalny kolor tła zostaje zachowany w mieszance do czasu aż stanie się najmniej prawdopodobnym kolorem oraz zostanie zaobserwowany nowy kolor. Zatem jeśli obiekt pozostanie nieruchomy wystarczająco długo aby stać się częścią tła, a następnie się poruszy, to rozkład opisujący poprzednie tło ciągle istnieje w tymi samymi μ i σ^2, ale mniejszym ω przez co może zostać szybko ponownie dołączony do modelu tła.

2.1.2.2. Estymacja modelu tła

Podczas gdy parametry modelu mieszanki dla każdego piksela zmieniają się, należy określić które rozkłady z mieszanki dają największe prawdopodobieństwo bycia wygenerowanymi przez procesy tła. Z heurystycznego punktu widzenia najbardziej interesujące są rozkłady, które dają najlepsze dopasowanie i najmniejszą wariancję

W celu wybrania odpowiednich rozkładów należy uszeregować je według wartości ω/σ. Wartość ta zwiększa się zarówno przy zwiększeniu dopasowania, jak i przy zmniejszeniu wariancji. W praktyce tak uszeregowana lista daje zbiór rozkładów, gdzie najbardziej prawdopodobni kandydaci znajdują się na

początku, a najmniej prawdopodobni na końcu. Pierwsze rozkłady wybrane jako model tła wybiera się za pomocą wzoru

$$B = argmin_b \left(\sum_{k=1}^{b} \omega_k > T \right) \tag{2.9}$$

gdzie T jest miarą minimalnej ilości danych jaka powinna być prana pod uwagę. Rozwiązanie takie bierze pod uwagę najlepiej dostosowany rozkład dotąd, aż pewna porcja T danych jest rozważona. Jeśli T jest jest wartością małą, to tedy model tła zazwyczaj jest unimodalny. Jeśli T jest wartością większą, multimodalny rozkład wywołany powtarzalnymi ruchami w tle może skutkować uwzględnieniem w modelu tła więcej niż jednego koloru. Skutkuje to efektem przezroczystości, który pozwala modelowi przyjmować dwa lub więcej oddzielnych kolorów.

2.2. Średnia z bufora ramek

2.2.1. Wprowadzenie

Średnia z bufora ramek jest jednym z najprostszych możliwych algorytmów generacji tła. Algorytm ten opiera się na wyliczaniu średniej arytmetycznej dla każdej pozycji piksela na obrazie spośród ramek zgromadzonych w buforze.

2.2.2. Opis metody

Wartość dla każdego piksela tła wyliczana jest ze wzoru.

$$B(x, y) = \frac{1}{n} \sum_{i=1}^{n} P(x_i, y_i) \tag{2.10}$$

Gdzie $B(x, y)$ oznacza obecną wartość piksela tła w miejscu (x,y), n oznacza rozmiar bufora, $P(x_i, y_i)$ oznacza wartość piksela na pozycji (x,y) w i-tej ramce w buforze.

2.3. Aproksymacja średniej przy użyciu parametru alfa

2.3.1. Wprowadzenie

Algorytm ten jest próbą aproksymacji średniej z bufora ramek. Jest on modyfikacją poprzedniego algorytmu i daje podobne rezultaty. Jednakże w przeciwieństwie do swojego pierwowzoru nie wykorzystuje on bufora oraz wymaga mniejszej ilości obliczeń.

2.3.2. Opis metody

Wartość dla każdego piksela tła wyliczana jest ze wzoru.

$$B_n(x, y) = P_n(x, y) * \alpha + B_{n-1}(x, y) * (1 - \alpha) \tag{2.11}$$

Gdzie $B_n(x, y)$ oznacza obecną wartość piksela tła w miejscu (x,y), $P_n(x, y)$ oznacza wartość piksela na pozycji (x,y) w obecnej ramce, B_{n-1} oznacza wartość piksela na pozycji (x,y) w poprzednim modelu tła

3. Metody blokowe (przestrzenne)

3.1. Wprowadzenie

Metody przestrzenne, w przeciwieństwie do liniowych, rozważają ramkę wideo jako grupę rozłącznych bloków o określonym rozmiarze oraz współrzędnych, lokalizujących jednoznacznie dany blok w danej ramce. Każdy blok jest określony przez jego wagę oraz piksele, które zawiera. Dodatkowo, dla całej sekwencji, dla każdej lokalizacji blokowej (czyli miejsca na obrazie, gdzie znajdują się bloki o określonych współrzędnych) jest utrzymywana tzw. grupa kandydatów - bloków, które z pewnym prawdopodobieństwem należą do tła. Waga danego bloku z tej grupy określa, jak często w sekwencji wideo pojawiał się blok podobny do niego (tzn. spełniający określone kryteria, omówione w sekcji 3.2.1)

Założenie tej klasy algorytmów polega na przypuszczeniu, iż blok który pojawia się najczęściej w danej sekwencji wideo jest najlepszym kandydatem na bycie blokiem tła. Bloki o tych samych współrzędnych są porównywane między sobą w kolejnych ramkach sekwencji wideo (np. przy użyciu współczynnika korelacji poszczególnych pikseli czy też sumy wartości bezwzględnych z różnic odpowiadających sobie pikseli), po czym w zależności od wyniku porównania następuje aktualizacja któregoś z bloków w grupie kandydatów lub też dodanie nowego bloku do tej grupy. W ostatnim etapie tej grupy algorytmów następuje odtworzenie tła na podstawie grup kandydatów przyporządkowanych do każdego bloku.

W tym rozdziale zostanie omówiona zasada działania tej rodziny algorytmów, jak i również zostaną przedstawione dwie metody, stosujące do wyboru najlepszego bloku z grupy kandydatów odpowiednio transformatę DCT oraz rekursywną transformatę Hadamarda.

3.2. Algorytm działania

Za [RSL09] przyjęto następujące oznaczenia, używane w dalszej części niniejszej pracy:

– W,H - odpowiednio szerokość i wysokość ramki

– I_f - ramka nr f

– $B_f(i,j)$ - blok ramki f o współrzędnych (i,j)

– $b_f(i,j)$ blok $B_f(i,j)$ po wektoryzacji

– $R(i,j)$ -zbiór kandydatów (grupa kandydatów) dla lokalizacji blokowej o współrzędnych (i,j)

– $r_k(i,j)$ - k-ty blok z grupy kandydatów $R(i,j)$

– $W_k(i,j)$ - waga k-tego bloku z grupy kandydatów $R(i,j)$

– μ_{r_k}, μb_f - średnia z elementów bloków odpowiednio r_k oraz b_f

– σ_{r_k}, σb_f - odchylenie standardowe z elementów bloków odpowiednio r_k oraz b_f

Każdą z opisywanych metod można podzielić na trzy zasadnicze fazy

1. Kolekcjonowanie kandydatów - bloków, które mogą się zawierać w tle
2. Częściowa rekonstrukcja tła
3. Estymacja tła na podstawie grup kandydatów

W kolejnych sekcjach każda z faz zostanie szczegółowo omówiona.

3.2.1. Kolekcjonowanie kandydatów

W tej fazie następuje obróbka każdej kolejnej ramki sekwencji wideo. Każda ramka f jest dzielona na bloki $B_f(i,j)$ o rozmiarze $N \cdot N$ każdy. Następnie każdy blok $B_f(i,j)$ jest zamieniany na N^2 wymiarowy wektor $b_f(i,j)$ poprzez łączenie ze sobą kolejnych wierszy. Następnie, każdy blok poddany wektoryzacji($b_f(i,j)$ jest porównywany z każdym blokiem $r_k(i,j)$ znajdującym się w grupie kandydatów dla danej lokalizacji blokowej ($R(i,j)$). Jeśli blok nie jest podobny do żadnego z kandydatów, zostaje dodany jako nowy kandydat z początkową wagą równą jeden. W przeciwnym wypadku, każdy podobny blok i jego waga są aktualizowane wg następujących wzorów:

$$r_k(i,j) = \frac{r_k(i,j)W_k(i,j) + b_f(i,j)}{W_k(i,j) + 1} \tag{3.1}$$

$$W_k(i,j) = W_k(i,j) + 1 \tag{3.2}$$

3.2.1.1. Kryteria podobieństwa bloków

Kluczowe dla prawidłowego działania algorytmu jest dobranie odpowiednich kryteriów podobieństwa bloków. Najcześciej w tym celu używa się współczynnika korelacji oraz współczynnika MAD[1]. Są one wyliczane nastepująco:

$$T_{corr} = \frac{(r_k(i,j) - \mu_{r_k}(i,j)^T (b_f(i,j) - \mu_{b_f})}{\sigma_{r_k}\sigma_{b_f}} \tag{3.3}$$

$$T_{MAD} = \sum_{n=0}^{N^2-1} \left| b_{f_n}(i,j) - r_{k_f}(i,j) \right| \tag{3.4}$$

Współczynnik korelacji T_{corr} odpowiada za podobieństwo bloków między sobą i zazwyczaj wymaga się, aby był powyżej pewnej wartości (np. 0.8) w celu uznania bloków za podobne. Jednakże często bywa on niewystarczający, gdyż często może zdarzyć się, iż dwa całkowicie niepodobne bloki będą miały bardzo wysoki współczynnik korelacji. W celu minimalizacji tego zjawiska wprowadzono współczynnik T_{mad}, który musi być odpowiednio niski dla dwóch bloków, by zostały uznane za podobne.

[1] MAD - Median of Absolute Differences

3.2.2. Częściowa rekonstrukcja tła

Po wyliczeniu kandydatów dla wszystkich lokalizacji blokowych następuje częściowa rekonstrukcja tła. Polega ona na znalezieniu wszystkich lokalizacji blokowych mających tylko jednego kandydata i przypisaniu ich wartości do odpowiednich bloków rekonstruowanego tła.

3.2.3. Estymacja brakującego tła

W celu przeprowadzenia estymacji tła zostało wprowadzone pojęcie superbloku, definiowanego jako klaster o wymiarach $2 \cdot 2$ bloki.

B	C	D
A	X	E
H	G	F

Rysunek 3.1: Blok X oraz jego 8-punktowe otoczenie

Przykładowo, dla bloku X z powyższego rysunku możemy wyróżnić następujące superbloki:{B,C,A,X}, {C,D,X,E}, {A,X,H,G} oraz {X,E,G,F}. W tej fazie algorytmu dla każdego superbloku który zawiera 3 bloki wypełnione tłem jest szacowany czwarty, brakujący blok.Estymacja odbywa się w dziedzinie częstotliwości. Każda z omawianych metod skupia się na analizie wysokich częstotliwości, gdyż to one odpowiadają za zmienność obrazu. Faza kończy się gdy całe tło zostanie zrekonstruowane. Omawiane dwa algorytmy różnią się doborem transformaty.

3.2.3.1. Estymacja z wykorzystaniem DCT

Metoda ta została zaproponowana w [RSL09]. W tej metodzie dla każdego superbloku tworzone są dwie różne wersje transformaty.

1. blok X jest zerowany, natomiast brana jest pod uwagę zawartość sąsiednich bloków. Na superbloku jest przeprowadzana dwuwymiarowa dyskretna transformata kosinusowa, a jej współczynniki są zapisywane w macierzy C o wymiarach $M \cdot M$. Współczynnik DC macierzy C (o współrzędnych (0,0)) jest ustawiany na 0, przez co pod uwagę zostanie wzięte tylko przestrzenne zróżnicowanie wartości poszczególnych pikseli.

2. bloki otaczające X są zerowane, natomiast X jest inicjalizowany kolejnymi wartościami r_k ze zbioru kandydatów R dla danej lokalizacji blokowej. Powstaje więc k wersji superbloku. Na każdym z superbloków jest przeprowadzana 2D DCT, a jej współczynniki są zachowywane w macierzy D_k, gdzie k to numer kolejnego bloku ze zbioru kandydatów. Tak jak poprzednio, współczynnik DC macierzy D_k jest ustawiany na zero.

Należy zauważyć, iż w wyniku zastosowania dwóch przeciwstawnych masek superbloku (tj. zerowania określonych bloków do niego należących) wartości pikseli w obszarze wysokich częstotliwości będą przeciwstawne w macierzach C oraz D_k. Mają one również tendencję do redukowania się przy dodawaniu tych macierzy. Istnieją jednak przypadki, gdy tak się nie dzieje i w macierzach C oraz D_k nie ma elementów o wysokich częstotliwościach - zdarza się to gdy wartości pikseli w niewyzerowanych blokach sa bliskie zeru. Aby temu zapobiec, analizujemy średnią pikseli μ_k bloku r_k - jeśli jest wyższa lub równa 128, odpowiednie bloki w obu wersjach superbloku są zerowane, a jeśli średnia jest niższa - wszystkie piksele tych bloków są ustawiane na 255. Korekta ta zapewnia, że w każdym superbloku obszar wysokich częstotliwości nie będzie pusty.

Własność redukowania się wysokich częstotliwości przy dodawaniu dwóch komplementarnych superbloków wykorzystano przy tworzeniu funkcji kosztu, wyznaczającej najlepszy blok do uzupełnienia brakującego tła. Funkcja ta ma następującą postać

$$cost(k) = \left(\sum_{v=0}^{M-1} \sum_{u=0}^{M-1} |C(v,u) + D_k(v,u)| \right) \lambda_k \tag{3.5}$$

$$\lambda_k = e^{-\alpha\omega_k} \tag{3.6}$$

gdzie $a \in \langle 0, 1 \rangle$, $\omega_k = \frac{W_k}{\sum_{k=0}^{L-1} W_k}$, przy czym W_k jest wagą elementu r_k, a L jest ilością elementów zbioru r_k. Współczynnik α jest dobierany zazwyczaj eksperymentalnie i określa, jak duży wpływ na wynik funkcji kosztu ma waga danego bloku(czyli tak naprawdę częstość jego występowania w sekwencji wideo). Blok o najniższej wartości funkcji kosztu (czyli taki, który najlepiej redukuje sumę wysokich częstotliwości w macierzach C i D) zostaje dodany jako najbardziej wiarygodna estymacja tła.

3.2.3.2. Estymacja z wykorzystaniem rekursywnej transformaty Hadamarda

Metoda została przedstawiona po raz pierwszy w [BVC10] W tej metodzie każda ramka jest dzielona na bloki o rozmiarze $16 \cdot 16$ pikseli. Do przeprowadzenia trzeciego etapu tej metody jest wykorzystywana dyskretna transformata Hadamarda, będąca generalizacją transformaty Fouriera. Opiera się ona na macierzach Hadamarda H, definiowanych rekursywnie w następujący sposób:

$$H_1 = [1] \tag{3.7}$$

$$H_2N = \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix} \tag{3.8}$$

Transformatę F bloku X o wymiarach $2N \cdot 2N$ wyraża się jako

$$F = MXM \tag{3.9}$$

gdzie M to macierz Hadamarda rzędu 2N. Bardzo użyteczną własnością tej transformaty jest możliwość rozbicia jej na sumę transformat rzędu niższego. Przykładowo, po rozbiciu bloku X na podbloki A,B,C,D o wymiarach $N \cdot N$ transformatę można obliczyć ze wzoru

$$F = \begin{bmatrix} H & H \\ H & -H \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix} \begin{bmatrix} H & H \\ H & -H \end{bmatrix} \tag{3.10}$$

gdzie H to macierz Hadamarda rzędu H. Wyrażając macierz F jako:

$$F = \begin{bmatrix} f_{1,1} & f_{1,2} \\ f_{2,1} & f_{2,2} \end{bmatrix} \tag{3.11}$$

otrzymujemy:

$$\begin{cases} f_{1,1} = HAH + HBH + HCH + HDH \\ f_{1,2} = HAH - HBH + HCH - HDH \\ f_{2,1} = HAH + HBH - HCH - HDH \\ f_{2,2} = HAH - HBH - HCH + HDH \end{cases} \tag{3.12}$$

Stosując wzory 3.11 oraz 3.12 można obliczyć macierz Hadamarda rzędu 2N z czterech macierzy rzędu N, co będzie pomocne w przy wyliczaniu transformat superbloków. Tak samo jak w metodzie DCT, dla każdego superbloku tworzone są 2 wersje transformat: pierwsza - z wyzerowanym blokiem X, biorąca pod uwagę zawartość bloków sąsiednich oraz druga - z wyzerowanymi sąsiednimi blokami, biorąca pod uwagę tylko zawartość bloku X. Funkcja kosztu, określająca który blok jest najlepszym kandydatem do bycia blokiem tła, jest identyczna jak we wzorze 3.5. Algorytm wprowadza również poprawkę na integralność wybranego bloku z resztą tła - jeśli jego średni gradient wzdłuż przynajmniej dwóch krawędzi jest większy niż γ, blok jest odrzucany, po czym następuje analiza kolejnego bloku z grupy kandydatów z najmniejszą wartością funkcji kosztu. Współczynnik γ jest dobierany eksperymentalnie.

4. Metodologia badań

4.1. Użyte sekwencje

W niniejszej pracy do oceny jakości algorytmów inicjalizacji tła użyto pierwszych 200 ramek następujących sekwencji video:

1. clip_01.mpg - sekwencja zarejestrowana przez kamerę umieszczoną na ścianie budynku C3 AGH w pochmurny dzień, przedstawiająca ludzi chodzących po chodniku oraz część jezdni ulicy Czarnowiejskiej wraz z poruszającymi się na niej samochodami

2. clip_02.mpg - jak w pkt. 1, lecz sekwencja jest w gorszej jakości

3. clip_03.mpg - jak w pkt. 1, lecz przy zmiennych warunkach oświetleniowych

4. clip_04.mpg - jak w pkt. 1, lecz ze stałym cieniem

5. fountain.mpg - sekwencja przedstawiająca na pierwszym planie działającą fontannę na tle parkingu z poruszającymi się samochodami

6. highway.mpg - sekwencja zarejestrowana przez kamerę umieszczoną nad autostradą

7. pedestrians.mpg - sekwencja zarejestrowana przez kamerę umieszczoną na dworcu kolejowym, przedstawiająca poruszających się po peronie pasażerów

8. sidewalk.mpg - sekwencja zarejestrowana przez kamerę umieszczoną nad przejściem dla pieszych; na obrazie występują mocne drgania kamery.

4.2. Użyte metryki

W celu porównania skuteczności algorytmów dla każdej z wyżej wymienionych sekwencji zostały wyliczone modele tła z użyciem różnych metod i współczynników.

Dla metod wykorzystujących transformatę DCT oraz rekursywną transformatę Hadamarda(Walsha) obliczenia zostały wykonane dla wartości współczynnika $T_1 = \{0.6, 0.7, 0.8, 0.9\}$ oraz $T_2 = \{10, 20\}$.

Tam, gdzie jest to możliwe została wybrana ramka referencyjna, najlepiej oddająca tło w danej sekwencji. Następnie obliczone modele tła zostały przyrównane do tejże ramki następującymi metrykami:

1. Błąd średniokwadratowy (Mean Squared Error, MSE)

$$MSE(I_r, I_b) = \frac{1}{N * M} \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} (I_r(i,j) - I_b(i,j))^2 \tag{4.1}$$

Gdzie I_r to ramka referencyjna, I_b to obliczony model tła, N to wysokość, a M to szerokość ramki

2. Procent podobieństwa o zadanym progu

Współczynnik ten określa, ile pikseli w wygenerowanym modelu tła jest podobnych (z zadanym progiem) do odpowiadających pikseli w ramce referencyjnej

$$P(I_r, I_b) = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} F(I_r(i,j), I_b(i,j)) \tag{4.2}$$

$$F(p_r, p_b) = \begin{cases} 1, \; gdy \, |p_r - p_b| < (1-\alpha) * 255 \\ 0, \; gdy \, |p_r - p_b| \geq (1-\alpha) * 255 \end{cases} \tag{4.3}$$

Gdzie $\alpha \in< 0,1 >$ oznacza wymagany próg podobieństwa piksela, I_r to ramka referencyjna, I_b to obliczony model tła, N to wysokość, a M to szerokość ramki, p_r to wartość piksela ramki referencyjnej, p_b to wartość piksela wygenerowane modelu tła.

3. Obraz różnicowy

Metryka ta jest głównie przeznaczona do wizualnej oceny efektywności algorytmu

$$AbsDiff(I_r, I_b) = |I_r - I_b| \tag{4.4}$$

4. Histogram

Metryka ta określa ilość pikseli o danej wartości, posłuży ona do wizualnej oceny różnic obrazów.

$$H(i, I) = \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} p(i, I(x,y)) \tag{4.5}$$

$$p(i, px) = \begin{cases} 1, \; gdy \, px = i \\ 0, \; gdy \, px \neq i \end{cases} \tag{4.6}$$

5. Histogram różnicowy

Metryka ta określa różnicę ilości pikseli o danej wartości

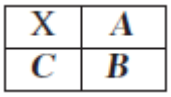
$$H(i, I_1, I_2) = \left(\sum_{x=0}^{N-1} \sum_{y=0}^{M-1} p(i, I_1(x,y)) \right) - \left(\sum_{x=0}^{N-1} \sum_{y=0}^{M-1} p(i, I_2(x,y)) \right) \tag{4.7}$$

$$p(i, px) = \begin{cases} 1, \; gdy \, px = i \\ 0, \; gdy \, px \neq i \end{cases} \tag{4.8}$$

5. Słownik użytych pojęć

blok wydzielona kwadratowa cześć ramki o rozmiarach $N * N$, zawierająca oprócz pikseli informację o wadze superbloku

superblok grupa 4 sąsiadujących ze sobą bloków, tworząca większy blok(superblok) o rozmiarach $2N * 2N$



Rysunek 5.1: Blok X oraz jego otoczenie, tworzące razem z nim superblok

grupa kandydatów

lokalizacja blokowa

Bibliografia

[BVC10] D. Baltieri, R. Vezzani, and R. Cucchiara. Fast background initialization with recursive hadamard transform. In *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*, pages 165 –171, 29 2010-sept. 1 2010.

[RSL09] V. Reddy, C. Sanderson, and B.C. Lovell. An efficient and robust sequential algorithm for background estimation in video surveillance. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 1109 –1112, nov. 2009.

[SG99] C. Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, volume 2, pages 2 vol. (xxiii+637+663), 1999.

[WS06] Hanzi Wang and David Suter. A novel robust statistical method for background initialization and visual surveillance. In P. Narayanan, Shree Nayar, and Heung-Yeung Shum, editors, *Computer Vision – ACCV 2006*, volume 3851 of *Lecture Notes in Computer Science*, pages 328–337. Springer Berlin / Heidelberg, 2006.