

# Zastosowanie techniki Motion Capture

**Michał Grędzia**

mgredzia@mion.elka.pw.edu.pl

Wydział Elektroniki i Technik Informatycznych  
Politechnika Warszawska  
ul. Nowowiejska 15/19  
00-665 Warszawa, Polska

13 czerwca 2008

## Streszczenie

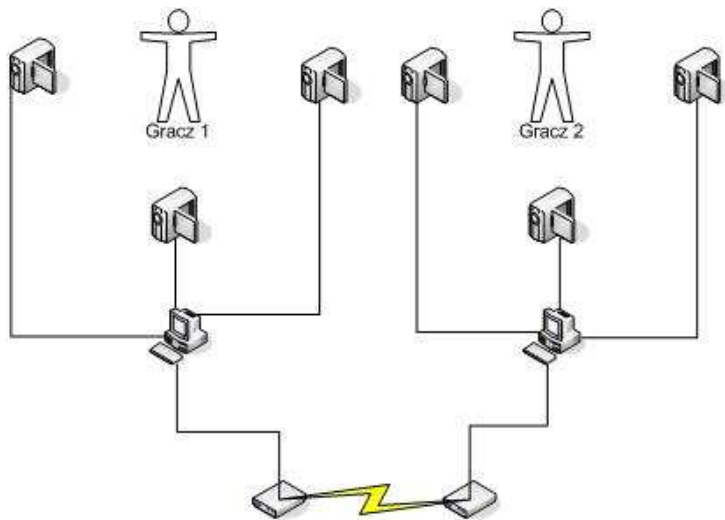
Niniejszy artykuł opisuje metodę implementacji techniki Motion Capture w warunkach domowych. W eseju przedstawione są w sposób ogólny wykorzystane algorytmy przetwarzania obrazów. Dokument jest podzielony na 5 części. Pierwsza opisuje problemy które trzeba pokonać oraz ogólny schemat postępowania. Druga opisuje zastosowaną technikę ( bibliotekę ) umożliwiającą akwizycję obrazu z 2 kamer internetowych, trzecia - przetwarzanie wstępne , czwarta - wykrywanie markerów a ostatnia ich śledzenie.

## Spis treści

<b>1</b>	<b>Wstęp</b>	<b>2</b>
<b>2</b>	<b>Akwizycja obrazu z kamery internetowej</b>	<b>5</b>
<b>3</b>	<b>Przetwarzanie wstępne</b>	<b>5</b>
3.1	Uniezależnienie od jasności . . . . .	5
3.2	Przekształcenia geometryczne . . . . .	6
3.3	Ekstrakcja tła . . . . .	6
<b>4</b>	<b>Detekcja markerów</b>	<b>6</b>
4.1	Detekcja wstępna . . . . .	6
4.2	Algorytm mean shift . . . . .	7
<b>5</b>	<b>Śledzenie markerów</b>	<b>8</b>

## 1 Wstęp

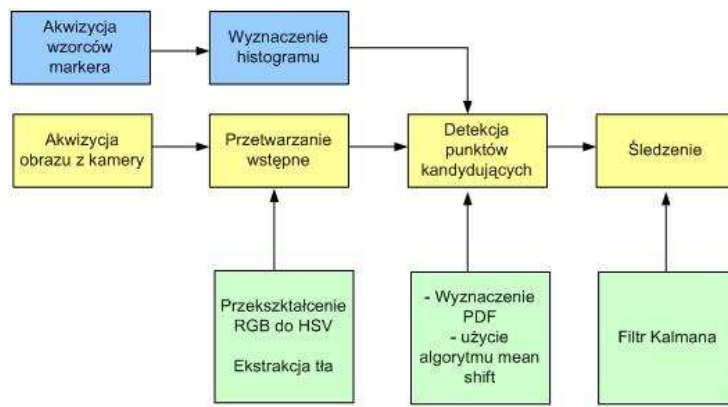
Projekt gry interaktywnej oparty jest na pomysśle połączenia standardowej gry 3D z techniką "motion capture". Dokładniej, gracz jest obserwowany przez co najmniej 3 kamery a jego ruchy śledzone i przełożone na postać 3D. Jest to coś więcej niż tylko sterowanie myszką. Można sterować postacią za pomocą ruchów rąk czy całego korpusu. Gesty jednego gracza widzi gracz drugi który jest po drugiej stronie sieci, gdyż podstawą funkcjonalnością gry będzie możliwość grania przez sieć. W celu rozpoznania postawy, na odpowiednich częściach ciała będą umieszczone kolorowe markery. Z punktu widzenia ekonomicznego oraz samej wygody systemu jest umieszczenie jak najmniejszej liczby kamer oraz wykorzystanie jak najmniejszej liczby maszyn w celu przetwarzania odebranego sygnału. Jedna kamera będzie przekazywać widok z przodu, druga z lewej i wreszcie trzecia z prawej strony postaci. Potrzebna by była również kamera z góry ale ze względów technicznych i ekonomicznych na chwilę obecną nie jest to możliwe do zrealizowania.



Rysunek 1: Konceptualny schemat gry multimedialnej.

Fundamentem techniki "Motion Capture" jest wykrycie "interesujących" punktów i umieszczenie ich w przestrzeni 3D. W pracy magisterskiej proces ten jest podzielony na 4 podstawowe etapy:

- Akwizycja obrazu z kamery
- Przetwarzanie wstępne (preprocessing)
- Detekcja punktów kandydujących
- Śledzenie filtrem Kalmana



Rysunek 2: Schemat implementacji techniki MC

Pierwszym etapem jest akwizycja obrazu z 2 kamer internetowych. Dostęp do obrazu jest możliwy dzięki wykorzystaniu biblioteki OPENCV. Wykorzystanie już 3 kamer wprowadza trudności implementacyjne związane z wydajnością. Problemem jest również ograniczenie biblioteki OPENCV w której można wykorzystać w systemie Windows maksimum 2 kamery. Przed procesem przetwarzania obrazu należy go odpowiednio przygotować. By uniezależnić się od składowej jasności koloru, obraz przetwarzany jest to trybu HSV (Hue Saturation Value). Kolejnym krokiem ulepszającym wykrywanie jest usunięcie z niego części, które nas nie interesują czyli wszystko co nie jest pierwszym planem czyli jest po prostu tłem. Mając już tak przygotowany obraz można przejść do procesu wykrywania markerów. Markery są to punkty odniesienia na ciele które należy wyodrębnić od całej reszty obrazu. Mając oddzielone tło teraz należy oddzielić obszary pierwszego planu które nie są markerami. Wykorzystać do tego można wiele algorytmów.

Jednym z podejść może być wykorzystanie progowania i algorytmu zalewania "flood fill": Ponieważ cechą charakterystyczną markerów (to co powinno wyróżniać je od całej reszty) jest barwa należy wydzielić te obszary obrazu przez progowanie. Progowanie jest funkcją która w zależności od wartości jaką mają 3 składowe piksela przyporządkowuje im jedną z 2 wartości binarnych. Czyli daje nam odpowiedź na pytanie czy piksel spełnia nasze kryterium czy nie. Zastosowanie progowania dla całego obrazu da nam 2 wymiarową funkcję binarną. Mając taką mapę binarną wystarczyłoby pozaznaczać obszary gdzie wynikiem progowania jest wartość pozytywna, wyliczyć powierzchnię tych obszarów oraz ich środki i oznaczyć jako markery. Wydaje się to proste.

Niestety przychodzący z kamery obraz jest mocno zaszumiony a obszary które jak nam się wydaje powinny mieć jednolity kolor tak naprawdę są wielokolorowe. Marker na takim obrazie

jest jedynie zagęszczeniem pikseli mających podobny kolor do tego co byśmy chcieli. W takich przypadkach zastawianie algorytmów bazujących na progowaniu takich jak flood fill nie zdaże egzaminu. Czyli są do rozwiązania 2 problemy: - problem barwy markera - problem szumu

Do rozwiązania pierwszego problemu można wykorzystać metody opierające się na analizie histogramu:

- stworzyć plik obrazu z markerami
- poprzez progowanie wyznaczyć obszary w obrazie wyznaczające tylko obszary markera - wynikiem tej operacji jest maska zaznaczająca na obrazie obszar markera
- wyznaczenie funkcji rozkładu prawdopodobieństwa wystąpienia piksela w obszarze markera - czyli na obrazie otrzymanym poprzez wykonanie operacji logicznej 'and' na obrazie markera i maski otrzymanej po progowaniu
- wykonać rzut histogramu na przychodzącą z kamery ramkę ( w takim przypadku otrzymamy obraz w odcieniu szarości gdzie pikselami o większej jasności są te które częściej występują w obrazie wzorca.)

Mając taki obraz rzutu należy wyznaczyć obszary o największej wartości całki po powierzchni na ograniczonym obrazie. Szybkie znalezienie tych obszarów jest możliwe przez wykorzystanie algorytmu **mean shift**. Mając metody do wyznaczania maksimów można przejść do śledzenia. Efektywne śledzenie można zaimplementować wykorzystując **filtr kalmana** lub algorytm **Condensation**

## 2 Akwizycja obrazu z kamery internetowej

W systemie Windows jest to możliwe przez wykorzystanie biblioteki DirectShow, która obecnie jest częścią pakietu DirectX SDK. Jednym z założeń projektu jest jego przenośność pomiędzy systemami operacyjnymi tak więc wykorzystanie DirectShow temu przeczy. Z pomocą przychodzi biblioteka CVCAM z pakietu OpenCV. CVCAM udostępnia interfejs pod którym kryją się odwołania do procedur przechwytyjących obraz w systemie Windows.

Akwizycja z wykorzystaniem biblioteki CVCAM z pakietu OpenCV oraz przy wykorzystaniu DirectShow. Aplikacja umożliwia akwizycję z maksimum 2 kamer internetowych podłączonych do jednego komputera w rozdzielczości 640x480, Poniższa tabela przedstawia wyniki wydajnościowe przy wykorzystaniu 2 kamer

- Creative VFO420
- Labtec

Liczba kamer	Tryb pracy	Liczba klatek na sekundę FPS
1	Bez przetwarzania	25
1	Z wykrywaniem markerów i śledzeniem	16
2	Bez przetwarzania	15
2	Z wykrywaniem punktów i śledzeniem	10

## 3 Przetwarzanie wstępne

### 3.1 Uniezależnienie od jasności

W celu uniezależnienia się od poziomu jasności otoczenia dobrym krokiem jest przejście do trybu HSV. HSV nawiązuje do sposobu w jakim widzimy, gdzie barwa biała wywodzi się z mieszanki wszystkich kolorów. Czyli barwy postrzegane są jako światło otoczenia.

$$H = \frac{g - b}{\text{delta}} \text{ if } MAX = R$$

$$H = 2 + \frac{b - r}{\text{delta}} \text{ if } MAX = G$$

$$H = 4 + \frac{r - g}{\text{delta}} \text{ else}$$

$$V = MAX$$

$$\text{if}(max! = 0) s = \frac{\text{delta}}{max}$$

$$\text{else } S = 0, H = -1$$

gdzie

$$MAX = \max(r, g, b)$$

$$MIN = \min(r, g, b)$$

$$\text{delta} = \max - \min$$

### 3.2 Przekształcenia geometryczne

Kolejnym problemem z którym trzeba się zmierzyć są zniekształcenia spowodowane krzywizną soczewek kamery. Są to tzw. Zniekształcenia radialne. Kalibracja kamery miałaby na celu znalezienie współczynników zniekształcenia  $k_1$  i  $k_2$ .

$$x' = x + x[k_1(x^2 + y^2) + k_2(x^2 + y^2)]$$

$$y' = y + y[k_1(x^2 + y^2) + k_2(x^2 + y^2)]$$

gdzie  $x'$ ,  $y'$  znormalizowane idealne współrzędne obrazu  $x$ ,  $y$  znormalizowane współrzędne obrazu zniekształconego

### 3.3 Ekstrakcja tła

Ekstrakcja tła dzieli się na 2 fazy:

- faza uczenia
- faza ekstrakcji

W fazie uczącej dla każdego piksela wyznaczana jest średnia jasność piksela oraz średnia kwadratów w  $N$  klatkach:

$$m(x, y) = \frac{S(x, y)}{N}$$

gdzie:

$s(x, y)$  - wartość piksela w punkcie  $x$ ,  $y$   $N$  - liczba klatek uczenia

Decyzja czy punkt należy do tła czy nie zależy od wyniku równania:

$$\text{abs}(m(x, y) - p(x, y)) > \text{thre}$$

gdzie:

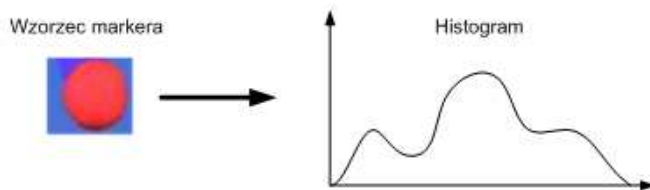
$\text{thre}$  - stały próg

## 4 Detekcja markerów

### 4.1 Detekcja wstępna

Detekcja składa się z 2 etapów wstępnych:

1. Warunkiem koniecznym detekcji jest załadowanie obrazu z wzorcowym markerem lub zdjęcie z wycinkami markera wykonanym w różnym oświetleniu.
2. Wykonanie histogramu obrazu wzorca w składowej Hue.



Rysunek 3: Wizualizacja fazy wstępnej wykrywania obiektów

Detekcja właściwa: Mając histogram składowej  $H$  wzorca markera, na przychodzącej klatce wykonywany jest rzut w płaszczyźnie  $H$ . Dzięki temu otrzymujemy na płaszczyźnie nowo powstałego obrazu rozkład prawdopodobieństwa wystąpienia danego koloru ( $H$ ).

$$K(x, y) = H(P_h(x, y))$$

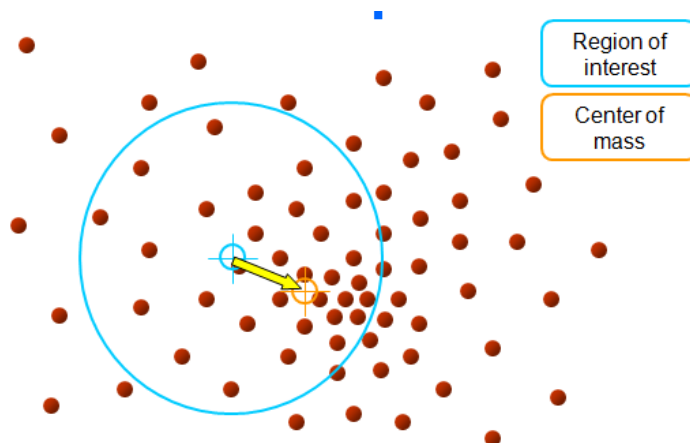
gdzie

$P_h(x, y)$  wartość składowej  $h$  na pozycji  $x, y$

$H(h)$  funkcja histogramu  $h - [0, 180]$

$K(x, y)$  - funkcja rozkładu

## 4.2 Algorytm mean shift



Rysunek 4: Działanie algorytmu mean shift

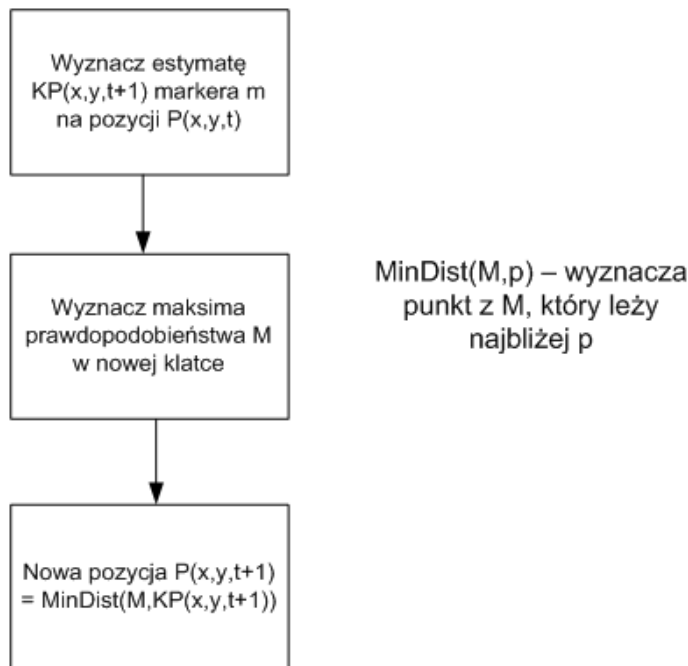
Mając punkt początkowy oraz pewien obszar wokół tego punktu wyznaczany jest wektor przesunięcia w kierunku maksimum funkcji rozkładu wyznaczonej na tym obszarze. Przesunięcie się odbywa iteracyjnie aż wartość tego wektora (jego długość) nie przekroczy pewnego wyznaczonego eksperymentalnie progu.

## 5 Śledzenie markerów

Śledzenie również składa się z 2 etapów:

- Inicjalizacja śledzenia przez znalezienie właściwych markerów
- Śledzenie punktów filtrem Kalmana

Inicjalizacja ma na celu poprawne wyznaczenie markerów a także ich poprawną klasyfikację. **Proces inicjalizacji:** Po- stać powinna stać w miarę nieruchomo. Pozycje maksimumów dodawane są do listy markerów jeżeli ich pozycja w ciągu  $p$  klatek nie zmieni się o  $d$ . Po spełnieniu warunku rozpoczyna się właściwa faza śledzenia. Mając zainicjalizowaną listę markerów  $L(m)$  z pozycjami  $P_i$  za pomocą filtru Kalmana wyznaczone są najbardziej prawdopodobne pozycje  $KP_{i+1}$  markerów w kolejnej klatce. Po otrzymaniu nowej klatki i wyznaczeniu w niej maksimumów  $M$ , wyznaczane są nowe pozycje markerów. Nową pozycją  $P_{i+1}$  markera  $L(m)$  jest maksimum z  $M$  najbliższe estymacie  $KP_{i+1}(m)$ .



Rysunek 5: Schemat procesu śledzenia dla jednego punktu



## Literatura

- [1] Zhengyou Zhang "Flexible camera calibration by viewing a plane from unknown orientations"  
<http://research.microsoft.com/~zhang>
- [2] Daniel Roetenberg, Human Motion Analysis  
<http://www.xsens.com>
- [3] Jelski Krzysztof, Rokita Przemysław  
Sterowanie kursorem w przestrzeni  
trójwymiarowej za pomocą obrazu z 2-kamer
- [4] Yaron Ukrainitz, Bernard Sarel Mean Shift - theory  
and applications