

Mirosław FORCZEK

Politechnika Śląska, Instytut Informatyki

ŚLEDZENIE POZYCJI DŁONI NA OBRAZIE WIDEO¹

Streszczenie. Przedstawiony został system śledzenia pozycji dłoni na obrazie wideo. Konstrukcja systemu zakłada wykorzystanie informacji z poprzednich okresów analizy oraz redukcję obszaru przetwarzanego obrazu. Oprócz samego obrazu wejściowego moduł decyzyjny wykorzystuje dwa dodatkowe źródła informacji: predykcję następnego położenia dłoni oraz model barwy obiektu. Uwzględnienie wszystkich dostępnych danych pozwala na lokalne przeszukiwanie obrazu.

HAND POSITION TRACKING ON VIDEO IMAGE

Summary. A system for hand position tracking on video image is shown. The construction of the system assumes use of information from previous analysis stages and reduction of processed image area. Apart from input image, the decision module utilizes two additional information sources: prediction of the next hand position and a hand color model. Taking into account all available data allows for local image searching.

1. Wstęp

Przedstawione dalej zadanie wyznaczenia aktualnej pozycji dłoni stanowi element systemu translacji języka migowego. Wejściem dla systemu jest obraz wideo przedstawiający osobę prezentującą wypowiedzi w języku migowym. Wyznaczenie pozycji dłoni w aktualnej klatce obrazu stanowi pierwszy krok w całym cyklu rozpoznawania znaków języka migowego. Czynność ta musi być powtarzana wielokrotnie dla kolejnych obrazów. Możliwe jest więc

¹ Praca została wykonana w ramach projektu badawczego 8 T11C 007 17, finansowanego przez Komitet Badań Naukowych w latach 1999-2001, a częściowo – w ramach pracy o symbolu BW 446/RAu2/2000

wykorzystanie techniki „śledzenia” w celu redukcji ilości potrzebnych obliczeń [2]. W ogólnym przypadku omawiany tutaj algorytm może dotyczyć śledzenia dowolnego obiektu, którego położenie zmienia się w sposób ciągły w czasie (niekoniecznie dłoni). Algorytm ukierunkowany na wyznaczanie pozycji dłoni może być również użyteczny we wszelkich aplikacjach wykorzystujących nowoczesne interfejsy oparte na wizji – np.: wydawanie poleceń za pomocą prostych gestów.

Głównym celem dla konstruowanego algorytmu jest redukcja złożoności obliczeniowej przez:

- wykorzystanie informacji z poprzednich okresów analizy,
- redukcję wielkości obszaru obrazu podlegającego przetwarzaniu.

Wykorzystanie wiedzy o poprzednich wynikach śledzenia pozwala na predykcję następnego położenia dłoni [1]. Tym samym zwiększana jest szansa na szybsze osiągnięcie celu kolejnego etapu przetwarzania – wyznaczenie nowej lokalizacji dłoni. Dodatkową redukcję ilości przetwarzanej informacji można uzyskać stosując metody analizy obrazu o charakterze lokalnym zamiast globalnym. Metody globalne wymagają poddania całego obrazu wybranemu przekształceniu, podczas gdy metody lokalne przetwarzają tylko niezbędny fragment obrazu. Przykładem metody globalnej może być wykonanie segmentacji całego obrazu, a następnie wybór właściwego obszaru.

Oczywiście, warunkiem zastosowania metody lokalnej jest znajomość przybliżonego położenia interesującego nas obiektu – tak więc predykcja położenia jest ważnym elementem dla lokalnej detekcji dłoni.

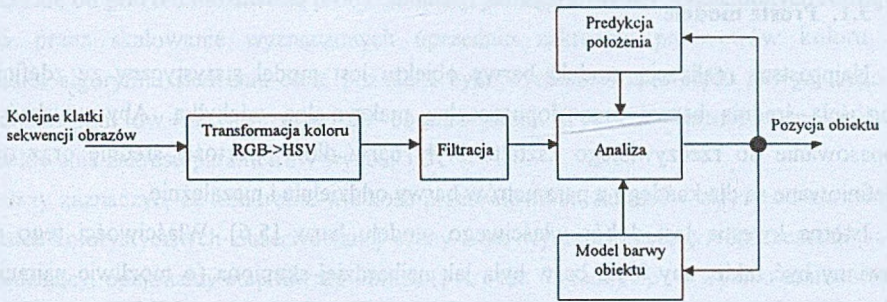
2. Moduł śledzenia

Schemat blokowy algorytmu został przedstawiony na rysunku 1.

Etap wstępnego przetwarzania obrazu obejmuje dwie czynności:

- konwersję barw,
- filtrację w celu usunięcia zakłóceń w obrazie.

Zgodnie z przyjętym założeniem o wykorzystaniu metod o charakterze lokalnym, tylko rzeczywiście przetwarzane fragmenty obrazu podlegają wyżej wymienionym przekształceniom.



Rys. 1. Ogólny schemat algorytmu

Fig. 1. General algorithm scheme

W kolejnym kroku następuje właściwe wyznaczenie pozycji dłoni. Oprócz obrazu wejściowego algorytm śledzenia wykorzystuje dwa dodatkowe źródła informacji:

- przewidywaną pozycję dłoni,
- model barwy dłoni.

Tym sposobem aktywnie wykorzystywane są wyniki poprzednich okresów analizy.

Kolejne rozdziały omawiają poszczególne bloki algorytmu. Rozdział 3 opisuje zagadnienia związane z modelem barwy, rozdział 4 metodę lokalnej detekcji dłoni, wreszcie rozdział 5 omawia predykcję nowego położenia oraz główny algorytm wyznaczania pozycji.

3. Model barwy obiektu

Podstawowym atrybutem (ang. *cue*) wykorzystywanym w pracy algorytmu jest barwa [3,4]. Przez porównywanie barw określone są granice obszaru dłoni. W tym celu algorytm utrzymuje wzorcowy model barwy dłoni, wykorzystywany do porównań [7-9]. Dłoń, jako obiekt o nieregularnych kształtach, oświetlona rozproszonym światłem nie uzyskuje jednego koloru na całej swojej powierzchni. Przeciwnie, poszczególne punkty widocznych powierzchni dłoni przyjmują barwy z określonego ciągłego, ale skończonego podzbioru przestrzeni barw. W przypadku popularnych 3-wymiarowych modeli barw podzbiór ten będzie bryłą o bliżej niesprecyzowanym kształcie. Dlatego konieczne jest posiadanie odpowiedniego wzorca, określającego przynależność danego koloru do zbioru barw dłoni.

3.1. Proste modele

Najprostszą realizacją modelu barwy obiektu jest model statystyczny ze zdefiniowaną wartością średnią barwy oraz dopuszczalną maksymalną odchyłką. Aby uzyskać lepsze dopasowanie do rzeczywistego kształtu bryły barw dłoni, wartości średnie oraz odchyłki zdefiniowane są dla każdego z parametrów barwy oddzielnie i niezależnie.

Istotną kwestią jest dobór właściwego modelu barw [5,6]. Właściwości tego modelu powinny być takie, aby bryła barw była jak najbardziej skupiona (o możliwie najmniejszym stosunku pola powierzchni do objętości bryły). Analiza własności modeli barw oraz kształtów brył barw wykracza poza tematykę niniejszego artykułu. Spośród znanych modeli barw wybrany został model HSV, który posiada zadowalające własności (dobre skupienie bryły barw – obecność parametru H reprezentującego odcień barwy) oraz jest stosunkowo prosty do implementacji (procedury konwersji).

Wyznaczenie wartości średnich c_{avg} i dopuszczalnych odchyłek c_{dev} dla parametrów można przeprowadzić na dwa sposoby. Mając do dyspozycji N próbek kolorów c_i z obszaru dłoni obliczamy potrzebne wskaźniki wg wzorów:

$$c_{avg} = \frac{\sum_{i=0}^{N-1} c_i}{N} \quad c_{dev} = \sqrt{\frac{\sum_{i=0}^{N-1} (c_{avg} - c_i)^2}{N}} \quad (1)$$

Wyznaczone w ten sposób zakresy parametrów koloru definiują aktualny model barwy obiektu. W wyniku zastosowania odchylenia standardowego granice zakresów zostały tak określone, że część próbek pozostaje poza ich granicami.

Alternatywna metoda wyznaczenia wartości c_{avg} oraz c_{dev} opiera się na wykorzystaniu próbek o maksymalnych odchyłkach parametrów w analizowanym zestawie wzorcowym:

$$c_{avg} = \frac{\max(c_i) + \min(c_i)}{2} \quad c_{dev} = \max(c_i) - c_{avg} \quad (2)$$

Taki sposób określenia zakresów parametrów koloru gwarantuje, że wszystkie próbki zestawu wzorcowego znajdują się wewnątrz tego zakresu.

Wizualizacja tak określonego modelu barwy obiektu w przestrzeni barw konkretnego systemu kolorów (RGB, HSV) przyjmie postać graniastoslupa (zakładając, że przestrzeń barw jest 3-wymiarowa i wymiary tworzą układ prostokątny). Jest oczywiste, że forma graniastoslupa jest tylko przybliżeniem rzeczywistego kształtu bryły barw obiektu. W rzeczywistości kształt bryły jest nieregularny, co powoduje, iż klasyfikacja punktów obrazu za pomocą tego modelu będzie obciążona dwoma rodzajami błędów:

- barwy należące do obiektu będą klasyfikowane jako tło,
- barwy nie należące do obiektu będą klasyfikowane jako do niego przynależne.

Zależnie od potrzeb możliwe są próby eliminacji jednego z dwóch wymienionych rodzajów błędów przez skalowanie wyznaczonych uprzednio zakresów parametrów koloru. W przypadku algorytmu śledzenia dłoni pożądane było wyeliminowanie błędu polegającego na odrzuceniu punktów przynależnych do obszaru dłoni. Typowa wartość współczynnika przeskalowania zakresu parametrów wynosi 3-5.

Należy zaznaczyć, że konkretna wielkość przeskalowania zakresów zależy od aktualnych własności kolorystycznych obserwowanej sceny oraz wybranej metody reprezentacji barw. Poprzedzający, opcjonalny etap filtracji obrazu (patrz rys. 1) może np. w wyniku zastosowania filtra dolnoprzepustowego (redukcja szumów) poprawić zbieżność próbek barw dłoni w przestrzeni kolorów. W efekcie - współczynnik przeskalowania może przyjąć mniejszą wartość.

3.2. Modele adaptacyjne

Przedstawiony poprzednio model barwy obiektu posiada dwie istotne wady:

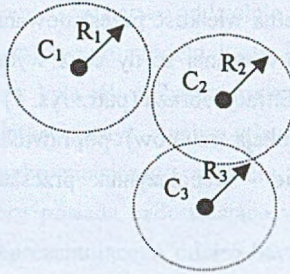
- kształt bryły barw reprezentowanej przez model znacznie odbiega od rzeczywistego,
- model nie uwzględnia problemu fluktuacji parametrów obserwowanej sceny wraz z wpływem czasu (np. skutek ciągłej zmiany warunków oświetlenia sceny).

Model barwy obiektu powinien posiadać możliwość adaptacji swoich parametrów [7-9]. Aktualizacja może być przeprowadzana okresowo lub w sposób ciągły. Głównym problemem przy adaptacji parametrów jest określenie wielkości i kierunku rzeczywistych zmian barwy obiektu. Algorytm nie wykorzystuje zewnętrznego wzorca barwy obiektu, jedyny pozostający do dyspozycji wzorzec to aktualny model barwy oraz charakterystyki bieżących testów.

Aby umożliwić adaptację modelu do powolnych zmian, celowo poszerzone zostają jego zakresy tolerancji parametrów. W wyniku tego model klasyfikuje część barw zbliżonych do barwy dłoni jako również do niej przynależące. Zakładając, iż obserwowana scena charakteryzuje się wysokim kontrastem oraz że detekcja dłoni wykorzystuje metodę zorientowaną lokalnie (patrz punkt 4), takie przypadki błędnej klasyfikacji (na granicy zakresów) powinny stanowić tylko niewielki procent wszystkich testów. Jeśli jednak kolorystyka sceny zaczyna się zmieniać, takich przypadków będzie coraz więcej podczas analizy kolejnych klatek obrazu. Charakterystycznym wskaźnikiem jest zmiana położenia punktu ciężkości analizowanych próbek, który oddala się od środka c_{avg} definiowanego przez model. Wskaźnik ten pozwala na wykrycie niepożądanego stanu, wyznaczenie kierunku i wielkości zmian oraz adaptację parametrów modelu. Metodę adaptacji opartą na położeniu środka ciężkości można łatwo wykorzystać w przedstawionych już modelach statystycznych.

Proponowany tutaj model posiada jednak inną konstrukcję. Na definicję modelu składa się zbiór par zawierających: próbkę barwy oraz jej *zasięg* (ang. *range*). Klasyfikacja koloru

odbywa się przez sprawdzenie, czy jego parametry leżą wewnątrz zasięgu jakiegokolwiek próbki ze zbioru modelu. Kształt bryły barw nie jest więc określony jedną definicją (jak w przypadku poprzednich modeli), ale jest wynikiem aktualnej konfiguracji próbek i ich zasięgów – otrzymujemy model **rozproszony**.



Rys. 2. Reprezentacja bryły barw w modelu rozproszonym

Fig. 2. Representation of color solid in distributed model

Taka organizacja modelu pozwala na swobodne dopasowanie do rzeczywistego kształtu bryły barw obiektu. Dodatkowo poprzez modyfikację zasięgów próbek możliwe jest uzyskanie modelu adaptacyjnego. W tym celu należy wprowadzić efekt „starzenia” się próbek w modelu – ich zasięg z czasem maleje do zera, po czym próbka zostaje usunięta ze zbioru. Jednocześnie aktualnie testowane próbki barw, które zostały sklasyfikowane jako przynależące do barwy dłoni, mogą zostać dodane do zbioru próbek modelu. Aktualizacja próbek może odbywać się w sposób ciągły lub okresowy, np. raz na cały cykl przetwarzania pojedynczej ramki obrazu. W ten sposób uzyskany zostaje efekt „podążania” zawartości zbioru próbek w modelu za aktualną tendencją zmian kolorystyki sceny.

4. Detektor obiektu

Zgodnie z ustaleniami z punktu 1, detekcja obszaru dłoni na obrazie powinna wykorzystywać metodę *zorientowaną lokalnie*. Metoda taka przetwarza jedynie fragment obrazu (możliwie mały) do momentu uzyskania pożądanych wyników. Dodatkowo zastosowanie metody lokalnej jest kluczowym założeniem algorytmu aktualizacji modelu barw (patrz punkt 3.2).

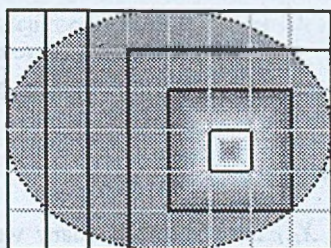
Dla kolejnych klatek obrazu wyznaczana jest aktualna pozycja dłoni. Istotną rolę w tym procesie pełni moduł **detektora obiektu**. Detektor otrzymuje jako parametr wejściowy

sugerowaną pozycję dłoni na obrazie (patrz punkt 5 – predyktor pozycji). Przed modułem detektora postawione są następujące zadania:

- weryfikacja, czy na wskazanej pozycji rzeczywiście znajduje się obiekt,
- określenie wymiarów granicznych obiektu, jeśli jego obecność została potwierdzona.

Proces detekcji oparty jest na klasyfikacji barw poszczególnych punktów obrazu za pomocą znanego modelu barwy dłoni. W pierwszym cyklu weryfikowana jest hipoteza obecności dłoni na wskazanej pozycji. W tym celu punkty leżące bezpośrednio w otoczeniu R punktu x_0 (sugerowana pozycja) poddawane są klasyfikacji. Jeżeli w wyniku klasyfikacji liczba punktów obrazu zaliczonych do koloru dłoni jest większa od założonej dolnej granicy, to wynik testu obecności obiektu jest pozytywny. W praktyce, wielkość otoczenia R określona jest przez promień 2 pikseli wokół punktu x_0 , natomiast dolny próg testu obecności ustalony jest na poziomie 75%.

Jeśli obecność dłoni na sugerowanej pozycji obrazu została potwierdzona, następuje cykl wyznaczenia granic obszaru obiektu. Proces ten jest wykonywany iteracyjnie: w każdej iteracji testowane są punkty leżące na obrzeżu aktualnego obszaru obiektu. Jeśli w wyniku klasyfikacji zostały znalezione nowe punkty przynależące do barwy dłoni, to obszar obiektu jest poszerzany w danym kierunku. Iteracja jest powtarzana aż do zatrzymania się ekspansji obszaru na skutek braku kolejnych punktów klasyfikowanych jako przynależące do barwy dłoni.



Rys. 3. Wyznaczanie granic obiektu przez rozrost obszaru

Fig. 3. Detection of object's bounds by region growing

5. Predyktor pozycji

Moduł **predyktora pozycji** jest odpowiedzialny za wyznaczenie kolejnej bieżącej lokalizacji dłoni. Właściwa predykcja kolejnych punktów jest tylko jednym z zadań wykonywanych przez ten moduł. Oprócz wyznaczenia potencjalnej pozycji dłoni, konieczna jest weryfikacja tej hipotezy za pomocą mechanizmu **detektora** (patrz punkt 4). W przypadku

niepowodzenia konieczne jest wyznaczenie alternatywnej pozycji i powtórzenie cyklu weryfikacji. Istotnymi częściami modułu predyktora są: algorytm koordynacji całego procesu śledzenia obiektu oraz algorytm systematycznego przeszukiwania przestrzeni obrazu.

5.1. Wyznaczanie następczej pozycji

Podczas określania kolejnej pozycji, w której znajduje się śledzony obiekt, możliwe jest wykorzystanie wszelkich dostępnych informacji z poprzednich okresów. Najprostszy model predykcji wykorzystuje tylko informację o poprzedniej lokalizacji X_t , co pozwala wstępnie określić punkt początkowy dla przeszukiwania obrazu. Model predykcji zastosowany w opisywanym systemie śledzenia dłoni wykorzystuje informacje o dwóch ostatnich lokalizacjach obiektu: X_{t-1} oraz X_t . Pozwala to na wyznaczenie ostatniego wektora ruchu obiektu \vec{M}_{t-1} . Wektor ruchu zawiera informację o kierunku ruchu oraz prędkości. Oba rodzaje informacji pozwalają lepiej wyznaczyć kolejną pozycję obiektu. Z braku innych przesłanek można założyć, że bieżący wektor ruchu \vec{M}_t będzie taki jak poprzedni, co pozwala na określenie współrzędnych nowego punktu X_{t+1} :

$$\begin{aligned}\vec{M}_{t-1} &= X_t - X_{t-1} \\ \vec{M}_t &= \vec{M}_{t-1} \\ X_{t+1} &= X_t + \vec{M}_t\end{aligned}\tag{3}$$

Wykorzystanie informacji z bardziej odległych okresów pozwoli na uwzględnienie dynamiki ruchu oraz zastosowanie bardziej wyrafinowanych algorytmów predykcji.

5.2. Przeszukiwanie obrazu

Wyznaczony nowy punkt X_{t+1} musi zostać poddany weryfikacji za pomocą detektora obiektu. Jeśli predykcja punktu X_{t+1} nie była właściwa, wynik testu będzie negatywny. W takim przypadku konieczna jest generacja alternatywnych punktów i powtórzenie weryfikacji. Generowane punkty regularnie pokrywają *obszar zainteresowania* (ang. *attention area*), w którym spodziewane jest odnalezienie obiektu. Obszar zainteresowania jest wycinkiem przestrzeni obrazu – w przypadku opisywanego systemu śledzenia obszar ten jest wnętrzem tzw. *kąta przeszukiwania*, którego wierzchołek umiejscowiony jest w punkcie ostatniej znanej lokalizacji obiektu X_t , oraz jest on symetryczny względem estymowanego wektora ruchu \vec{M}_t .

Wielkość kąta przeszukiwania α_t zależy dynamicznie od estymowanego wektora ruchu:

$$\alpha_t = f(\vec{M}_t)\tag{4}$$

Ponieważ śledzony obiekt materialny charakteryzuje się bezwładnością, można wnioskować, iż w przypadku dużej wartości bezwzględnej wektora ruchu (prędkość) nie zmieni on raczej dynamicznie kierunku ruchu – kąt przeszukiwania może zostać zawężony. Odwrotnie, jeśli prędkość ruchu jest mała, należy się spodziewać większych zmian w kierunku ruchu – kąt przeszukiwania powinien zostać zwiększony. Wielkość kąta przeszukiwania może się zmieniać od 0 do 2π (cała przestrzeń obrazu). Redukcja wartości kąta do 0 wyklucza jakiegokolwiek poszukiwania. Aby tego uniknąć, dolna granica dla kąta przeszukiwania musi być tak ograniczona, aby przy założeniu, iż obiekt rzeczywiście znajduje się na pozycji X_{t+1} , jego cały obszar znalazł się we wnętrzu obszaru przeszukiwania. W tym celu wykorzystywana jest informacja o rozmiarach obiektu D (znana z poprzednich okresów):

$$\alpha_t = f(\vec{M}_t, D) \quad (5)$$

Rozmiar obiektu jest również brany pod uwagę podczas określania kroku S dla siatki generowanych alternatywnych punktów w celu pokrycia całej przestrzeni obszaru zainteresowania:

$$S = g(D) \quad (6)$$

Ponieważ kąt przeszukiwania jest symetryczny względem wektora ruchu, w praktyce wyznaczana jest wielkość $\frac{\alpha_t}{2}$ reprezentująca przewidywaną wielkość odchylenia w kierunku ruchu. Zastosowane w implementacji formuły dla wielkości $\frac{\alpha_t}{2}$ oraz S są następujące:

$$S = \min(\dim X(D), \dim Y(D)) \quad (7)$$

$$\frac{\alpha_{\min}}{2} = \frac{S}{2|\vec{M}_t| - S} \quad (8)$$

$$k = \frac{|\vec{M}_t|}{V_{\max}} \quad (9)$$

$$\frac{\alpha_t}{2} = k \frac{\alpha_{\min}}{2} + \pi(1 - k) \quad (10)$$

Wielkość V_{\max} oznacza maksymalną prędkość obiektu i jest dobierana arbitralnie.

6. Uwagi końcowe

Przedstawiony powyżej system śledzenia dłoni został zaimplementowany oraz wykonane zostały wstępne testy na dostępnych sekwencjach wideo. Przeprowadzone testy wykazały

zdolność systemu do poprawnego śledzenia dłoni przez okres ok. 50 klatek obrazu. Potwierdza to słuszność głównych założeń konstrukcyjnych algorytmu. Nadal jednak pozostają problemy, które należy rozwiązać w dalszym toku badań. Należą do nich:

- wykorzystanie innych modeli barwy obiektu z uwzględnieniem możliwości adaptacji modelu w czasie – celem jest poprawa własności modelu oraz redukcja kosztu obliczeniowego pojedynczego testu z udziałem modelu,
- rozszerzenie metody predykcji kolejnego położenia – w celu uzyskania bardziej dokładnych „trafień” już w pierwszym etapie wyznaczania pozycji,
- poprawa metody estymacji kąta przeszukiwania,
- wykorzystanie innych atrybutów oprócz barwy, np.: ruchu.

LITERATURA

1. Munk K., Granum E.: On the use of context and a priori knowledge in motion analysis for visual gesture recognition. *Gesture and Sign Language in Human-Computer Interaction, Proceedings of GW'97*, Springer 1997.
2. Subutai A.: *A Usable Real-Time 3D Hand Tracker*. Interval Research Corporation.
3. Triesch J., Malsburg Ch.: Robotic gesture recognition. *Gesture and Sign Language in Human-Computer Interaction, Proceedings of GW'97*, Springer 1997.
4. Triesch J., Malsburg Ch.: Robust Classification of Hand Postures against Complex Backgrounds. *IEEE Computer Society Press, Proceedings of the Second International Conference on Automatic Face and Gesture Recognition '96*, Killington, Vermont 1996.
5. Praca zbiorowa pod redakcją J. Zabrodzkiego: *Grafika komputerowa metody i narzędzia*. Wydawnictwo Naukowo-Techniczne. Warszawa 1994.
6. Forczek M.: Modele barw w algorytmach segmentacji obrazów kolorowych. *ZN Pol. Śl. s. Informatyka z. 38*, Gliwice 2000, s. 171-195.
7. Raja Y., Mc Kenna S.J., Gong S.: *Segmentation and Tracking Using Colour Mixture Models*, Dept. Computer Science, Queen Mary and Westfield College, London.
8. Raja Y., Mc Kenna S.J., Gong S.: *Object Tracking using Adaptive Colour Mixture Models*, Dept. Computer Science, Queen Mary and Westfield College, London.
9. Raja Y., Mc Kenna S.J., Gong S.: *Tracking and Segmenting People in Varying Lighting Conditions using Colour*, *Proceedings of FG'98*, Nara, Japan 1998.

Wpłynęło do Redakcji 20 czerwca 2001 r.

Abstract

The task of finding current hand location is presented here. An input to the system is a video image presenting a person showing sentences using deaf language.

The main aim of developed algorithm is reduction of its calculation complexity by: use of information from previous analysis stages and reduction of processed image area.

The pre-processing stage includes: colors conversion and filtering for noise elimination. Only really processed image parts are pre-processed. Apart from input image, the tracing algorithm utilizes two additional information sources: predicted hand position and hand color model.

The basic cue employed in the algorithm is an color. By color comparison, the hand area bounding box is determined. The algorithm keeps hand color model as the pattern, that is used during comparisons. The hand illuminated by a distributed light source, doesn't have a single color on all its surface. A particular points from visible part of surface get colors from finite but continuous subset of colors space. In the case of use of one of popular 3-dimensional color models, the hand colors subset become to a solid of free shape. Thus, it is necessary to have a pattern, which define membership of given color to the hand colors subset.

The simplest realization of object color model is a statistic model with defined average value and maximum deviation. Such model has two drawbacks: the shape of colors solid is highly different from the real one and model doesn't change when object's colors change in time. We need an adaptive model which is defined by set of pairs containing: color sample and its range. The color classification is done by checking if its parameters belong to the range of any sample from model set. The shape of colors solid is defined now by configuration of samples and their ranges – this is a **distributed** model.