

Adam BOREK

ALGORYTM WYZNACZANIA PRZEMIESZCZEŃ OPTYCZNYCH, WYKORZYSTUJĄCY TRANSFOR- MACJĘ HOUGHA¹

Streszczenie. W pracy zaproponowano algorytm wyznaczania pola przemieszczeń dla dwóch przesłanych z kamery obrazów, wykorzystujący transformację Hougha. Na bazie zaproponowanego algorytmu opracowano także program komputerowy, którym sprawdzono poprawność idei. Uzyskane wyniki okazały się być bardzo obiecujące i zachęcające do dalszych kroków w tym kierunku.

AN ALGORITHM FOR DETERMINATION OF OPTICAL DISPLACEMENT USING HOUGH TRANSFORM

Summary. In the paper an algorithm for determining optical displacements has been proposed. The algorithm uses hough transform and it reduces the difficult problem of looking for correspondences in the image space, to the simpler one of looking for local maxima in appropriate parameter space. The computer program which has been elaborated on the basis of the algorithm was tested with 256x256, 16 gray scale images transmitted by camera through frmae grabber card. The obtained results sum up the paper.

¹Praca była finansowana z grantu wewnętrznego BK - 465.

АЛГОРИТМ ОПРЕДЕЛЕНИЯ ОПТИЧЕСКИХ ПЕРЕМЕЩЕНИЙ, ИСПОЛЬЗУЮЩИЙ ПРЕОБРАЗОВАНИЕ ХОФФА

Резюме. В работе представлен алгоритм определения поля перемещений для 2 изображений из камеры, который использует преобразование Хоффа. На основе предложенного алгоритма разработана также компьютерная программа, которая подтверждает корректность идеи. Полученные результаты многообещающие и вызывают интерес к дальнейшим исследованиям.

1. WSTĘP

Wiele uwagi poświęca się obecnie próbom wykorzystania systemów wizyjnych we wszelkiego rodzaju systemach sterowania. Sytuacja taka stała się możliwa dzięki gwałtownemu wzrostowi mocy obliczeniowej komputerów. Poszukuje się także coraz doskonalszych metod obliczeniowych wykorzystujących w pełni możliwości sprzętu oraz zapewniających dużą dokładność obliczeń.

Wyznaczanie przemieszczeń i przepływu optycznego jest jednym z etapów na drodze do wyznaczania parametrów ruchu z informacji wizyjnej. Na ogół począwszy od dyskretyzacji poprzez konturowanie do segmentacji konturu. Są to etapy stosunkowo mało czasochłonne, lecz kłopotliwe obliczeniowo i nietrudno tutaj o popełnienie błędu. Zastosowanie transformacji Hougha do wyznaczania przemieszczeń optycznych pozwala wyeliminować wiele etapów przetwarzania wstępnego obrazu. Inną zaletą HT jest możliwość łatwej adaptacji na sprzęcie o strukturze równoległej.

Przedstawione fakty powodują, że w literaturze pojawia się coraz więcej prac poświęconych analizie, uogólnieniom i zastosowaniom transformacji Hougha [1]-[7]. Jedną z takich prac, w których wykorzystuje się transformację Hougha do przetwarzania obrazów pobieranych z kamery jest [3]. W pracy tej przedstawiono algorytm wyznaczania prostoliniowych przemieszczeń dla sekwencji złożonych z trzech obrazów.

W niniejszej pracy prezentowany jest algorytm wyznaczania przemieszczeń optycznych wykorzystujących transformację Hougha. Zastosowano tutaj odmianę transformacji Hougha, w której przestrzeń parametrów jest przestrzenią wartości przemieszczenia obiektu na obrazie. Do obliczeń potrzebne są dwa kolejne obrazy z kamery. Obliczanie tej transformacji podobne jest do liczenia korelacji wzajemnej między obrazami. Złożony problem analizy przestrzeni obrazu sprowadzony zostaje do poszukiwania lokalnych maksimów przestrzeni parametrów Hougha.

Układ pracy jest następujący:

- w punkcie 2 przedstawiono podstawowe informacje dotyczące transformacji Hougha. Jest to bardzo skrócone powtórzenie informacji dostępnej w literaturze, ale umożliwia czytelnikowi stykającemu się po raz pierwszy z tym zagadnieniem należyte zrozumienie problemu,
- w punkcie 3 zawarte są zasadnicze wyniki pracy,
- w punkcie 4 przedstawiono wyniki przeprowadzonych obliczeń.

2. TRANSFORMACJA HOUGH

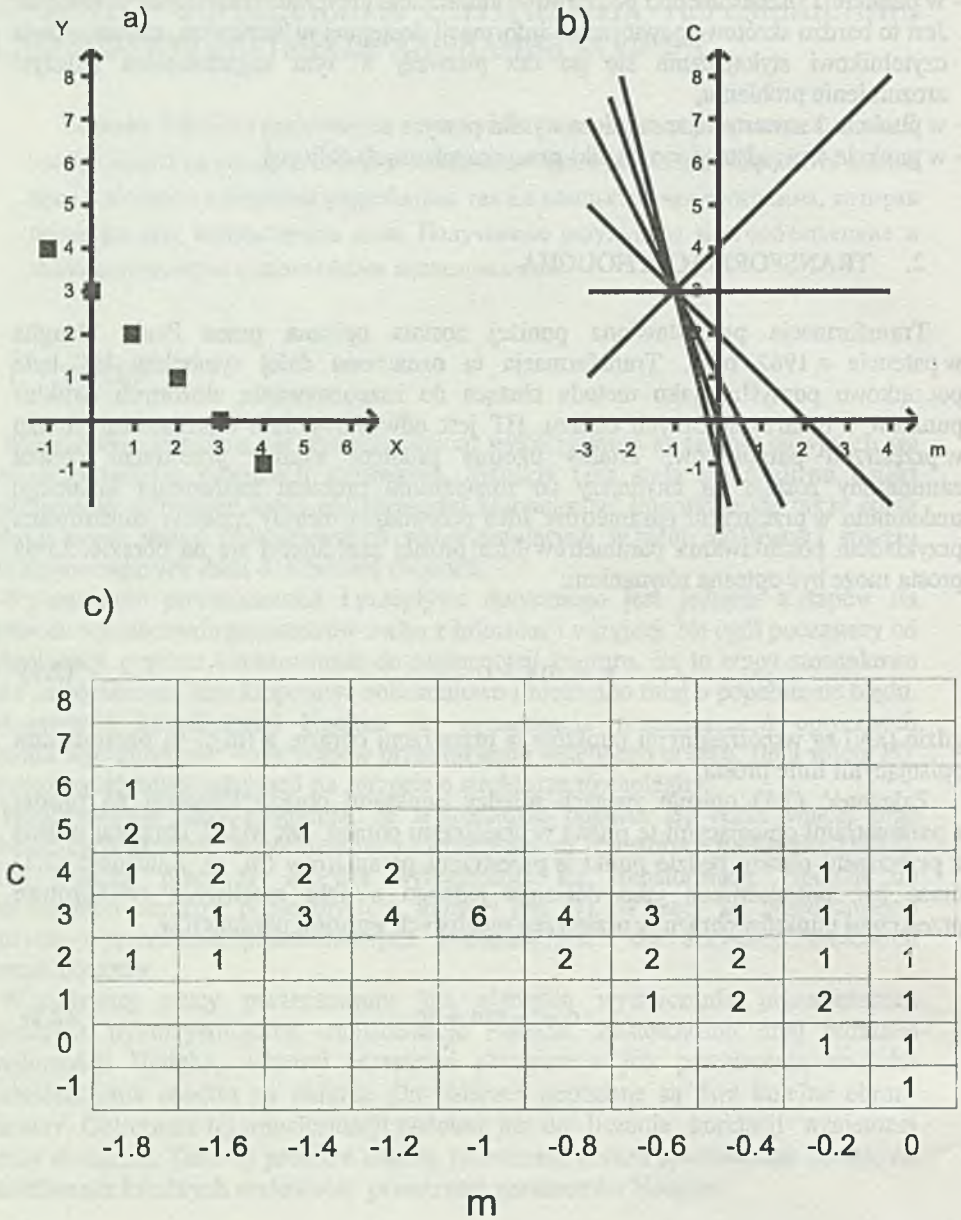
Transformacja przedstawiona poniżej została opisana przez Paula Hougha w patencie z 1962 roku. Transformacja ta oznaczona dalej symbolem HT była początkowo pomyślna jako metoda służąca do rozpoznawania złożonych struktur punktów w binarnych danych obrazu. HT jest odwzorowaniem z przestrzeni obrazu w przestrzeń parametrów. Trudny ogólnie problem analizy przestrzeni obrazu zamieniony zostaje na łatwiejszy do rozwiązania problem znalezienia lokalnego maksimum w przestrzeni parametrów. Idea przewodnia metody zostanie zilustrowana przykładem poszukiwania parametrów linii prostej znajdującej się na obrazie. Linia prosta może być opisana równaniem:

$$y = mx + c \quad (2.1)$$

gdzie (x,y) są współrzędnymi punktów w przestrzeni obrazu, a (m,c) są parametrami opisującymi linię prostą.

Zależność (2.1) opisuje związek między punktami obrazu leżącymi na prostej a parametrami opisującymi tę prostą w przestrzeni obrazu. Jak widać, obrazem prostej z przestrzeni obrazu będzie punkt w przestrzeni parametrów (m, c) . Zależność (2.1) może być potraktowana jako definicja jednego z wielu możliwych odwzorowań przestrzeni punktów obrazu w przestrzeń możliwych wartości parametrów.

$$c = -xm + y \quad (2.2)$$



Rys. 2.1. Przykład wykorzystania transformacji Hougha do detekcji linii: a) punkty w przestrzeni (x, y) , b) przestrzeni parametrów (m, c) , c) przestrzeni akumulatora odpowiadająca b)
 Fig. 2.1. An example of application of Hough transform to line detection a) points in (x, y) space, b) parameter space (m, c) , c) accumulator space corresponding to b)

Wykorzystanie zależności (2.2) do transformacji punktu o współrzędnych (x, y) odpowiada obliczeniu wszystkich możliwych parametrów (m, c) linii prostych przechodzących przez ten punkt. Tak więc każdy punkt (x, y) z przestrzeni obrazu generuje linię prostą w przestrzeni parametrów (m, c) . Proste w przestrzeni parametrów (m, c) odpowiadające punktom kolinearnym z przestrzeni obrazu przecinają się w jednym punkcie. Współrzędne punktu przecięcia określają parametry (m, c) linii prostej łączącej punkty obrazu.

Na rysunku 2.1.a pokazano zbiór punktów obrazu. Linie proste w przestrzeni (m, c) odpowiadające punktom obrazu pokazano na rysunku 2.1.b. Linie te przecinają się w jednym punkcie wyznaczającym parametry linii prostej z rysunku 2.1.a. W praktycznej realizacji przestrzeń parametrów można podzielić na fragmenty uzyskując macierz "akumulatorów". Jeżeli linia w przestrzeni (m, c) przechodzić będzie przez dany akumulator-licznik, to jego stan zwiększa się o jeden. W ten sposób transformacja Hougha podobna jest do procesu głosowania, gdzie każdy punkt głosuje na wszystkie możliwe parametry (m, c) linii prostych przechodzących przez ten punkt. Wynik tego działania przedstawiono na rysunku 2.1.c. Jak widać, złożony problem analizy obrazu zamieniony został w znacznie prostszy problem poszukiwania lokalnego maksimum w przestrzeni parametrów (m, c) .

Ponieważ "m" i "c" mogą zmieniać się w znacznych przedziałach, dlatego znacznie lepszym opisem linii prostej będzie postać normalna tego równania:

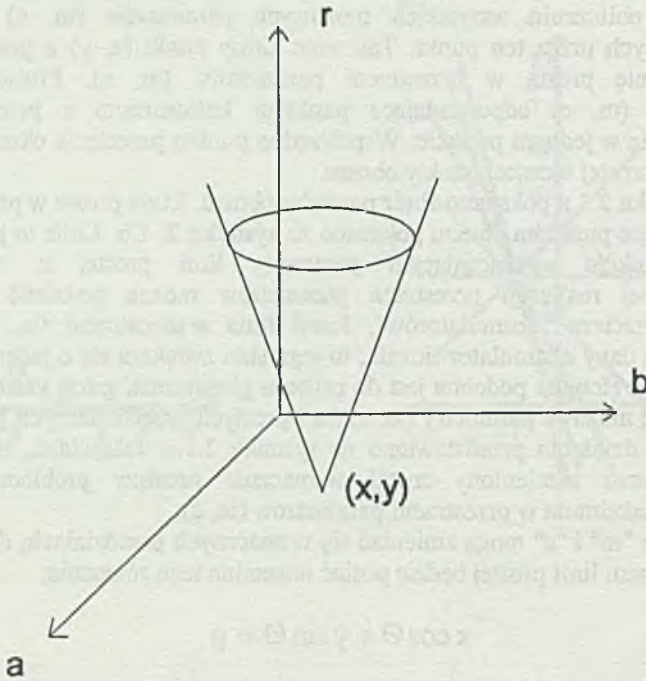
$$x \cos \Theta + y \sin \Theta = p \quad (2.3)$$

Teraz punktowi (x, y) w przestrzeni obrazu odpowiada sinusoida w przestrzeni parametrów (p, Θ) , a linia prosta opisana jest przez punkt przecięcia tych sinusoid.

Transformacja Hougha może być rozszerzona do określania położenia i promienia okręgu na płaszczyźnie obrazu. Okrąg może być opisany równaniem:

$$(x - a)^2 + (y - b)^2 = r^2 \quad (2.4)$$

gdzie (a, b) są współrzędnymi środka okręgu, a "r" jest jego promieniem. Przestrzeń parametrów będzie tu trójwymiarowa. Jeżeli rozpatrzmy punkt o współrzędnych (x, y) , to dowolny punkt w odległości "r" od niego może być środkiem okręgu. Każdy punkt (x, y) w przestrzeni obrazu generuje zatem stożek w przestrzeni parametrów (a, b, r) jak na rysunku 2.2. Metoda może być rozszerzona do wykrywania różnych kształtów. Kształt może być reprezentowany przez listę punktów brzegowych (x_i, y_i) . Obiekty należące do tej samej klasy kształtów mogą być otrzymane przez translację o wektor (u, v) , obrót wokół punktu (x_0, y_0) i zmianę skali "s" obiektu podstawowego.



Rys. 2.2. Każdy punkt z przestrzeni obrazu generuje stożek w przestrzeni parametrów
 Fig. 2.2. Each point of image space generates the cone in parameter space

$$x' = xO + s(x - xO) \cos \Theta - s(y - yO) \sin \Theta + u \quad (2.5)$$

$$y' = yO + s(x - xO) \sin \Theta + s(y - yO) \cos \Theta + v \quad (2.6)$$

Transformacja będzie teraz polegała na znalezieniu parametrów (u , v , Θ , s), które przekształcają wzorec w obiekt znajdujący się na obrazie. Każdy punkt obrazu przekształcony zostaje w hiperpłaszczyznę w przestrzeni parametrów. Liczba hiperpłaszczyzn przecinających się w danym punkcie odpowiada liczbie punktów pokrywających się z kształtem wzorcowym opisanym przez dany układ parametrów.

Zaletami transformacji Hougha są:

- odporność na zakłócenia o charakterze losowym,

- analiza każdego punktu obrazu oddzielnie, więc metoda daje się łatwo zaadaptować do systemów komputerowych o strukturze równoległej (możliwy jest podział między procesory zarówno przestrzeni wejściowej, jak i przestrzeni parametrów),
 - odporność na częściową deformację obrazu i niewielką utratę informacji
- Wady:
- metoda jest bardzo czasochłonna obliczeniowo,
 - wymagania co do wolnej pamięci operacyjnej rosną szybko ze wzrostem wymiarowości problemu i wzrostem zakładanej dokładności obliczeń.

3. ALGORYTM METODY WYZNACZANIA PRZEMIESZCZEŃ OPTYCZNYCH

Prezentowany algorytm pozwala wyznaczyć wektor przemieszczenia odpowiadający obiektowi przemieszczającemu się na scenie. Konieczne jest założenie, że obiekt porusza się ruchem prostoliniowym (nie występuje ruch obrotowy).

Kolejne kroki algorytmu:

1. Pobranie z kamery dwóch obrazów P_0 i P_1 .
2. Konturowanie obrazów P_0 i P_1 . W wyniku tej operacji otrzymuje się dwa obrazy binarne B_0 i B_1 . Do wyznaczania krawędzi zastosowano algorytm uśrednionego gradientu.
3. Usunięcie tła z obrazów B_0 i B_1 . W tym celu oblicza się $A = B_0 \text{ AND } B_1$. W binarnym obrazie A znajduje się część wspólna obrazów B_0 i B_1 , a więc tło obrazów. Następnie wylicza się:

$$O_0 = B_0 \oplus A$$

$$O_1 = B_1 \oplus A$$

Operator " \oplus " oznacza operację logiczną exclusive-or (XOR). Na obrazach O_0 i O_1 znajdują się teraz obserwowane obiekty. Obrazy O_0 i O_1 poddaje się filtracji, aby usunąć zakłócenia w postaci, rozrzuconych po obrazie pojedynczych punktów. Nie zawsze operacja wyznaczania i usuwania tła daje dobre rezultaty (np. w pewnych specyficznych warunkach może być utracona informacja o fragmentach przemieszczających się obiektu). Należy jednak zauważyć, że krok 3 przewidziany jest jedynie do zmniejszenia liczby punktów na obrazie i przyspieszenia transformacji. Można ten krok pominąć, licząc się ze zwiększeniem czasu trwania obliczeń i z powstaniem w punkcie o współrzędnych (0,0) w przestrzeni parametrów lokalnego maksimum. Maksimum to odpowiada nieruchomym elementom obrazu, a więc takim, dla których składowe wektora przemieszczenia są równe zeru.

4. Transformacja Hougha na przefiltrowanych obrazach O_0 i O_1 . Operacja ta jest równoważna w tym przypadku obliczeniu dla każdego punktu przestrzeni parametrów następującego wyrażenia:

$$H[i, j] = \sum_x \sum_y O_0[x, y] O_1[i + y, j + y] \quad (3.1)$$

W praktyce taki sposób wyznaczania przestrzeni parametrów oznacza bardzo dużą czasochłonność obliczeń. Ponieważ obrazy O_0 i O_1 są binarne, dlatego dokładnie taki sam wynik otrzymana się po wykonaniu operacji:

$$\bigvee_{x, y, i, j} O_0[x, y] \neq 0 \quad \bigvee O_1[x + i, y + j] \neq 0 \quad H[i, j] := H[i, j] + 1 \quad (3.2)$$

Operacja opisana powyżej oznacza obliczanie składowych wektora przemieszczenia z każdego punktu obrazu O_0 do każdego punktu obrazu O_1 . Każdy obliczony wektor przemieszczenia zlicza się w dyskretnej przestrzeni parametrów. Dzięki takiemu sposobowi obliczeń są one szybsze i ściśle zależne od liczby punktów na obrazie.

5. Wyszukanie w przestrzeni parametrów maksimumów lokalnych. Możliwe jest teraz wyliczenie z pewnym przybliżeniem środków ciężkości obiektów generujących te maksima. W tym celu oblicza się dla każdego maksimum lokalnego powstałego w punkcie (i, j) następujące wyrażenia:

$$X_c(i, j) = \frac{\sum_x \sum_y O_0[x, y] * O_1[x + i, y + j] * x}{\sum_x \sum_y O_0[x, y] * O_1[x + i, y + j]} \quad (3.3)$$

$$Y_c(i, j) = \frac{\sum_x \sum_y O_0[x, y] * O_1[x + i, y + j] * y}{\sum_x \sum_y O_0[x, y] * O_1[x + i, y + j]} \quad (3.4)$$

Wartości X_c i Y_c można uznać w przybliżeniu za współrzędne środka ciężkości obiektu generującego dane maksimum lokalne. Należy podkreślić, że jeżeli nie przeprowadzono kroku trzeciego algorytmu, to wartości obliczone ze wzorów (3.3) i (3.4) mogą być obarczone znacznym błędem. Algorytm nie wymaga wcześniejszej informacji o kształcie przemieszczającego się obiektu lub jego prędkości. Jak już wcześniej wspomniano, krok trzeci algorytmu można pominąć. Zastosowano go, aby:

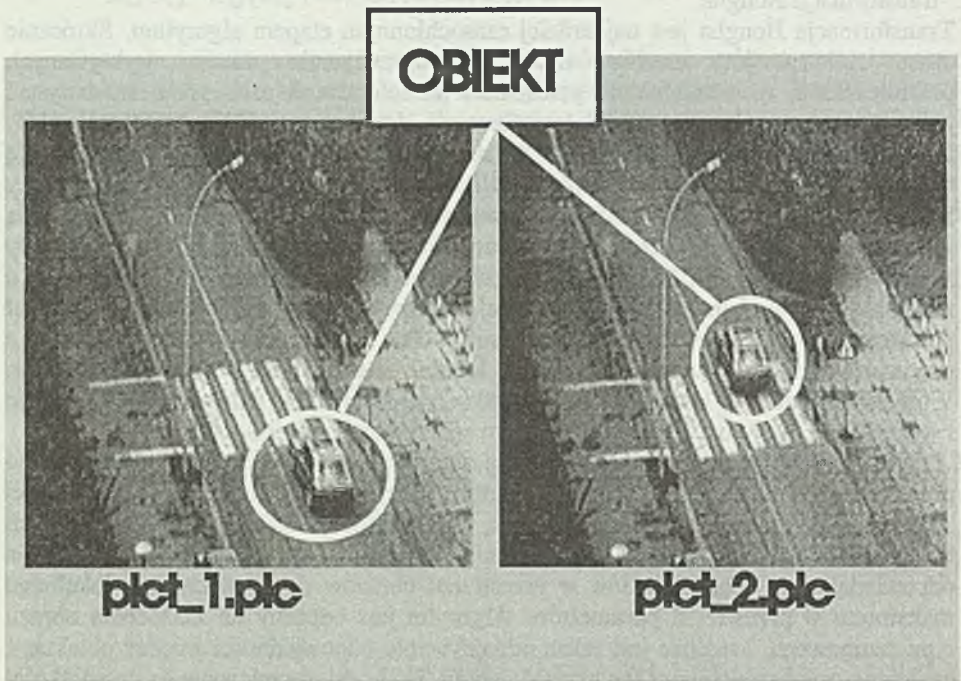
- wyeliminować statyczne elementy obrazu,
- zmniejszyć tło w przestrzeni parametrów,
- umożliwić wyliczenie środków ciężkości obiektów w kroku piątym,
- zmniejszyć ilość punktów na obrazach, co z kolei powoduje skrócenie czasu trwania transformacji Hougha.

Transformacja Hougha jest najbardziej czasochłonnym etapem algorytmu. Skrócenie czasu obliczeń jest możliwe np. przez ograniczenie zakresu wykrywanych przemieszczeń, tj. zmniejszenie przestrzeni parametrów. Można także wykorzystać jedną ze znanych implementacji transformacji Hougha, tzn. FHT, MHT lub AHT. Kolejne zmniejszenie czasu obliczeń można osiągnąć realizując transformację w systemie z wieloma procesorami. Możliwe są dwa podejścia do podziału pracy między procesory. Można podzielić przestrzeń obrazu na fragmenty, które będą przetwarzane przez poszczególne procesory. Po zakończeniu obliczeń trzeba wtedy zebrać przestrzeń parametrów od poszczególnych procesorów, posumować je, a następnie wyszukać maksima lokalne w tej przestrzeni. Inną możliwością jest podział przestrzeni parametrów między procesory. Po zakończeniu obliczeń każdy z procesorów może wyszukać maksimum lokalne w swoim fragmencie parametrów. Wartości maksimów lokalnych ze wszystkich podsystemów należy poddać analizie i wyszukać te, które odpowiadają wektorom rzeczywistym przemieszczeń.

Ważną cechą przedstawionego powyżej algorytmu jest możliwość wyeliminowania procesu segmentacji i dyskretyzacji konturu. Etapy te są konieczne w większości znanych algorytmów tego typu i sprawiają dużo kłopotu. Zaletą algorytmów opartych na transformacji Hougha jest sprowadzenie dość złożonego problemu poszukiwania odpowiadających sobie obiektów w przestrzeni obrazów do poszukiwania lokalnego maksimum w przestrzeni parametrów. Algorytm jest odporny na zakłócenia obrazu typu szumowego. Możliwe jest także odnajdywanie odpowiedności między obiektami częściowo zniekształconymi lub przesłoniętymi. Brak jest ograniczenia co do wielkości przemieszczenia obiektu na obrazie.

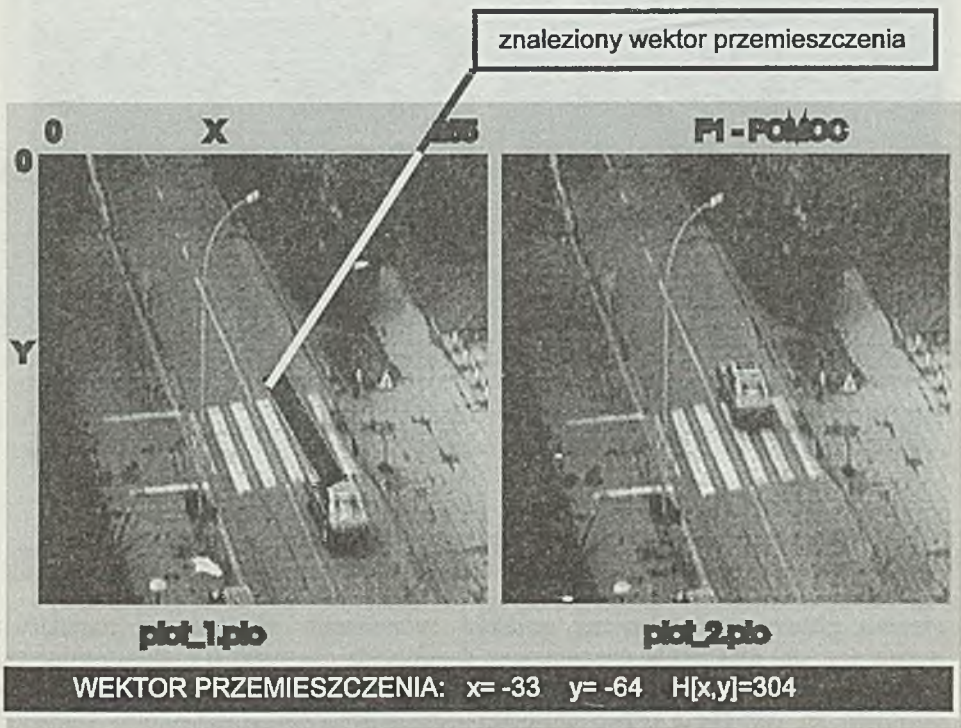
4. PRZYKŁAD

Na podstawie opisanego w punkcie 3 algorytmu powstał program komputerowy, który wyznacza wektory przemieszczeń obiektów poruszających się na obrazie. Program potrafi wyznaczyć przemieszczenie dla trzech obiektów znajdujących się jednocześnie na obrazie. Przetwarzaniu poddano obrazy o wymiarach 256x256 i 16 poziomach szarości.



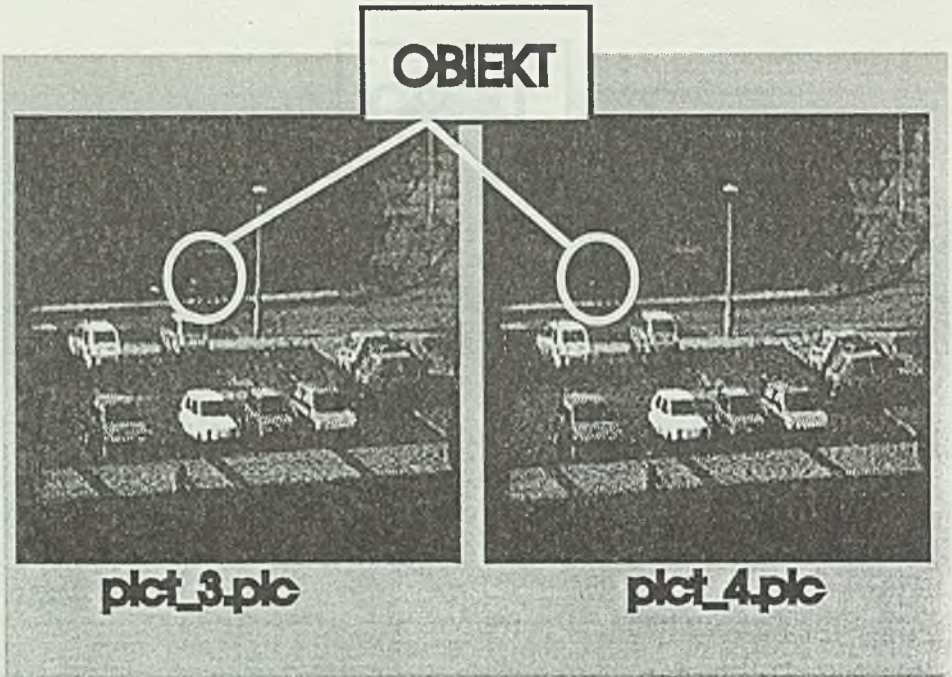
Rys. 4.1. Dwa kolejne zdjęcia jadącego samochodu
Fig. 4.1. Two subsequent images of a moving car

Na rysunku 4.1 przedstawiono obrazy stanowiące dane wejściowe. Widać na nich przemieszczający się samochód. Obrazy te poddane zostały procesowi obliczeniowemu. Na rysunku 4.2 widać te same obrazy z zaznaczonym wektorem przemieszczenia. Wyznaczony wektor przemieszczenia ma składowe $[-33, -64]$ (kierunku osi układu współrzędnych są zaznaczone na rysunku). Obliczenia przeprowadzono także dla różnych parametrów procesu konturowania i filtracji. Uzyskane wyniki oscylowały w przedziałach wartości: $\Delta x \in (-30, -33)$ i $\Delta y \in (-64, -65)$. Próba określenia wektora przemieszczenia bez pomocy komputera jest trudna, ponieważ samochód wygląda nieco inaczej na dwóch kolejnych ujęciach, ale daje takie same rezultaty.

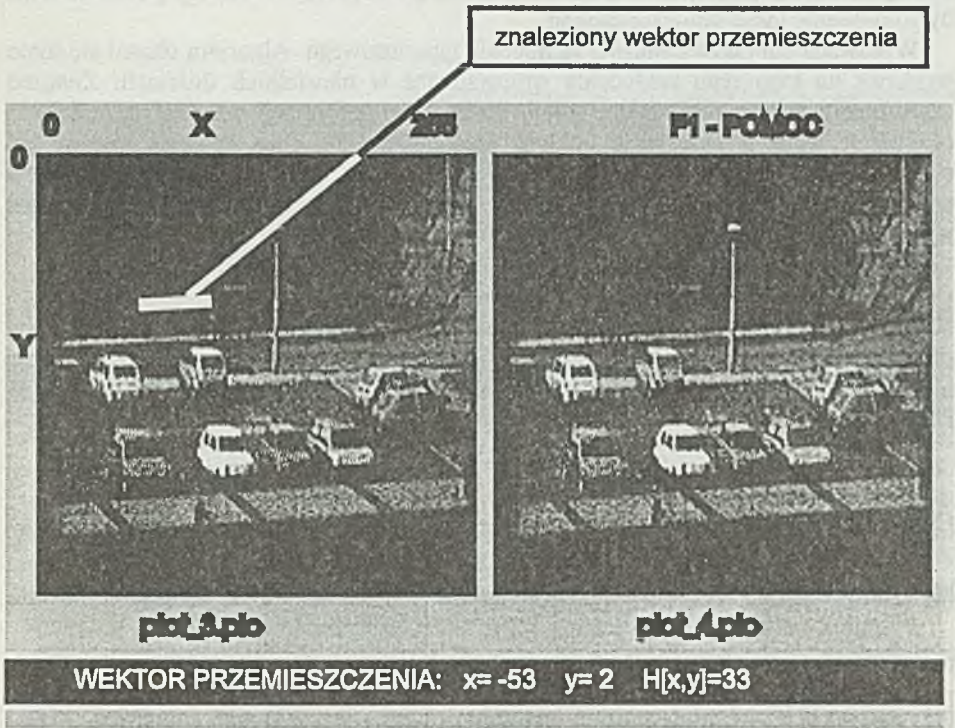


Rys. 4.2. Wyznaczony przez program wektor przemieszczenia
Fig. 4.2. Displacement vector determined by the program

Kolejne dwa zdjęcia Nr 3 i nr 4 przedstawiają podobną sytuację, lecz teraz poruszającym się obiektem jest rowerzysta. Obliczono, że rowerzysta przemieścił się odpowiednio o -41 punktów w osi x i 2 punkty w osi y. Tak jak poprzednio zmieniano parametry procesu konturowania i filtracji uzyskując następujące wyniki: $\Delta x \in (-40, -41)$, $\Delta y \in (2,3)$. Rezultaty obliczeń sprawdzono analizując wybrane fragmenty obrazu, wydają się one być poprawne i dokładne.



Rys. 4.3. Dwa kolejne zdjęcia przemieszczającego się rowerzysty
Fig. 4.3. Two subsequent images of the moving cyclist



Rys. 4.4. Wyznaczony przez program wektor przemieszczenia
 Fig. 4.4. Displacement vector determined by the program

Wzory (3.3) i (3.4) umożliwiają obliczenie środka ciężkości obiektu generującego maksimum w przestrzeni parametrów. Możliwe jest więc narysowanie wektora przemieszczenia o obliczonych składowych zaczepionego w punkcie (x_c, y_c) obrazu wejściowego. Umożliwia to natychmiastową wizualną kontrolę uzyskanych wyników. Ponieważ trudno analizować dokładność przedstawionego algorytmu na zdjęciach rzeczywistego świata, algorytm testowano także z obrazami wygenerowanymi w sposób

sztuczny. Wygenerowane zostały przemieszczające się okręgi, kwadraty i inne bardziej skomplikowane figury geometryczne. Obliczone przez program wektory przemieszczeń były dokładnie takie same jak zadane.

Wprowadzono także sztucznie zakłócenia typu losowego. Algorytm okazał się mało wrażliwy na tego typu zakłócenia wprowadzone w niewielkich ilościach. Znaczne zaszumienie obrazu powodujące utratę dużej ilości informacji wizyjnej powodowało oczywiście błędy w określaniu wektora przemieszczenia. Czas obliczeń zależny był silnie od liczby punktów znajdujących się na obrazie. Zależnie od liczby punktów pozostałej na obrazach po procesie konturowania, usuwania tła i filtracji czas obliczeń wyniósł odpowiednio:

- dla przemieszczającego się samochodu 1 min. 10 sekund do 2 min. 20 sekund,
- dla poruszającego się rowerzysty 20 sekund do 1 minuty.

Najbardziej czasochłonnym etapem algorytmu jest transformacja Hougha. Czas obliczeń przeznaczony na pozostałe etapy, a więc konturowanie, usuwanie tła, filtrację nie przekraczał 3 sekund. Obliczenia przeprowadzono na komputerze IBM PC 486 DX2 50MHz.

LITERATURA

- [1] Illingworth J., Kittler J.: A survey of the Hough transform, *Computer Vision, Graphics and Image Processing*, 44, pp. 87-116, 1988.
- [2] Ballard D.H.: Generalizing the Hough transform to detect arbitrary shapes, *Pattern Recognition*, vol. 13, no.2 pp. 111-122, 1981.
- [3] Biland H.P.: Multiple motion recognition with general Hough transform, *Time-varying image processing and moving object recognition*, Proceedings of the International Workshop, Florence, Italy, 1986.
- [4] Brown C.M.: Inherent bias and noise in the Hough transform, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 5, no 5, september 1983.
- [5] Lavin H. Li., Master R. Le.: Fast Hough Transform, *Computer Vision, Graphics and Image Processing*, 36, pp. 139-161, 1986.
- [6] Illingworth J., Kittler J.: The Adeptive Hough Transform, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 5, september 1987.
- [7] Atiguzzaman M.: Multiresolution Hough Transform, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 11, november 1992.
- [8] Hildreth F.C.: *The measurement of visual motion*, Cambridge, M. I. T. Press, 1984.
- [9] Ordys A., Polański A., Simek K., Świerniak A., Wojciechowski K.: *Zeszyty Naukowe Politechniki Śląskiej*, nr 975, z. 97 1990.

- [10] Hough P.V.C.: Method and means for recognizing complex patterns, U. S. Patent 306954, 1964.

Recenzent: Prof. dr hab. inż. Ryszard Tadeusiewicz

Wpłynęło do Redakcji 14 grudnia 1992 r.

Abstract

In the paper an algorithm for determining optical displacements has been proposed. The algorithm uses Hough transform and it reduces the difficult problem of looking for correspondences in the image space, to the simpler one of looking for local maxima in appropriate parameter space. The algorithm offers several advantages, such as robustness to noise and partial image distortions as well as the possibility of being easily adapted for parallel hardware architecture. The computer program which has been elaborated on the basis of the algorithm was tested using different input data. The result of these experiments were very good. The processed 256x256, 16 gray scale images were transmitted by video recorder through frame grabber card.

The paper contains also some basic information on Hough transform. The obtained results sum up the paper.