

Przemysław MAZUREK

ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY W SZCZECINIE, Katedra Przetwarzania Sygnałów i Inżynierii Multimedialnej
26. Kwietnia 10, 71-126, Szczecin

Redukowanie czasu obliczeń transformaty Hougha z wykorzystaniem preselekcji implementowane na GPGPU

Dr inż. Przemysław MAZUREK

Adiunkt w Katedrze Przetwarzania Sygnałów i Inżynierii Multimedialnej Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie. Autor ponad 150 artykułów z zakresu cyfrowego przetwarzania sygnałów, estymacji ruchu, biopomiarów oraz przetwarzania obrazów biomedycznych.

e-mail: przemyslaw.mazurek@zut.edu.pl



Streszczenie

W artykule przedstawiono algorytm rzadkiego próbkowania okręgu dla transformaty Hougha, w celu preselekcji okręgów do dalszego przetwarzania. Rozwiązanie pozwala na skrócenie czasu obliczeń, rzędu dwóch razy dla implementacji z wykorzystaniem programowanej karty graficznej (GPGPU). Może ono być zastosowywane do wykrycia jednokolorowej kryzy wokół lustra sferycznego lub półsferycznego służącego do pomiaru oświetlenia. Dzięki temu można analizować obraz wideo zawierający pomiary z różnych obszarów.

Słowa kluczowe: Transformata Hougha, Próbnik światła, Pomiary oświetlenia, GPGPU

Reducing Computation Time of Hough Transform Implemented on GPGPU using Preselection of Circles

Abstract

The Hough Transform (HT) is very powerful algorithm for the detection of shapes. It is applied for the detection of the flange of light probe (Fig.1). GPGPU processing is necessary for HT computation. Proposed algorithm (Fig.3) is based on the preselection of the circles for further computation (Fig.2). Selection of the circles based on a limited set of pixel samples allows reduction of the processing time up to two times. The reduction is dependent on the image content, but the test shows that are less influences by the number of tested radiuses. About one second of the processing time is necessary, so larger images need tens of seconds for computation (Fig.4). This algorithm uses exhaustive search over all positions and radiuses. One of the interesting applications of the HT is the estimation of position and diameter of circle related to the flange of light probe. The light probes are used for computer graphics applications and they allow application of the real world measurements of light using the Image Based Lighting (IBL) technique.

Keywords: Hough transform, Light probe, Light measurements, GPGPU

1. Wprowadzenie

Transformata Hougha (HT – ang. Hough Transform) jest algorytmem wykorzystywanym do detekcji kształtów [1], także zasumionych. Najczęściej wykorzystywana jest ona do detekcji linii oraz okręgów. HT estymuje dopasowanie kształtu, położenie, orientację, skalę, w zależności od realizacji. Algorytm jest bardzo wymagający obliczeniowo, ponieważ testowane są wszystkie przypadki. Estymacja parametrów okręgu jest bardziej wymagająca, ponieważ konieczne jest analizowanie okręgów o różnych promieniach.

Jednym z interesujących zastosowań HT w analizie obrazu jest estymacja obszaru, w którym znajduje się próbnik światła. Zwykle jest to sferyczne lub półsferyczne lustro, odbijające oświetlenie otoczenia [2]. Ekstrakcja tego kołowego obszaru, a następnie

transformacja do postaci półsfery umożliwia wykorzystanie informacji o wektorach światła na potrzeby grafiki i animacji komputerowej – IBL (ang. Image-Based Lighting). Na Rys. 1 przedstawiono półsferyczny próbnik oświetlenia z niebieską kryzą.



Rys. 1. Próbnik oświetlenia

Fig. 1. Light probe

Ponieważ w obrazie jest tylko jeden próbnik to zadaniem jest znalezienie maksimum w przestrzeni transformaty H:

$$H_{\max}(x, y, r) = \max H(x, y, r) \quad (1)$$

Transformata Hougha realizuje operację:

$$H(x, y, r) = \sum_i I(x + r \cdot \sin(\alpha_i), y + r \cdot \cos(\alpha_i)) \quad (2)$$

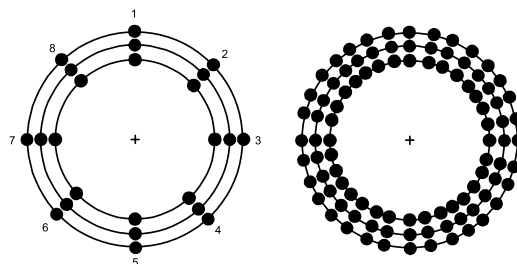
gdzie: I – obraz, x, y – środek okręgu, r – promień, i – indeks kąta, α – kąt. W przypadku zastosowania inwersji [3] przestrzeni poszukiwań zadaniem jest znalezienie minimum:

$$H_{\min}(x, y, r) = \min_r H(x, y, r) \quad (3)$$

Rozwiązanie to jest o tyle dobre, że w trakcie akumulacji (3) można odrzucić okręgi, dla których wartość tymczasowa jest za duża.

2. Preselekcja okręgów

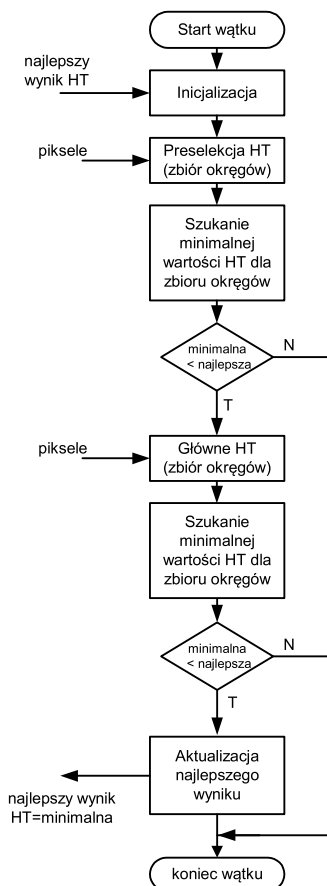
Proponowana metoda preselekcji okręgów bazuje na rzadszym próbkowaniu obrazu (Rys.2). Wybór okręgów, a dokładnie zestawu, dokonują się podobnie jak dla metody [3] jak na Rys. 3.



Rys. 2. Preselekcja i zwykłe próbkowanie HT (pokazano trzy okręgi)

Fig. 2. Preselection and main HT sampling (three circles shown)

Proponowany algorytm nie próbuje okręgu w sposób optymalny, ponieważ ponownie te same punkty używane są podczas głównej estymacji.

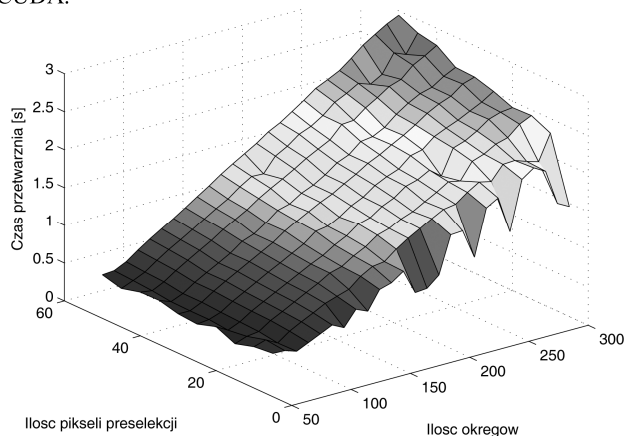


Rys. 3. Algorytm preselekcji dla HT

Fig. 3. Algorithm of preselection based HT

3. Implementacja oraz czas przetwarzania

Zasadniczą kwestią jest wpływ wyboru liczby punktów na okręgu. Zbyt duża prowadzi do zmniejszenia szybkości, mała natomiast wpływa na jakość pracy. Wyznaczenie minimalnej liczby punktów jest trudne, ponieważ zależy ona od zawartości obrazu. Z doświadczeń wynika, że powinno być to kilkanaście punktów. Na Rys. 4 przedstawiono czas obliczeń dla obrazu o rozdzielczość 512x512 pikseli. Wykorzystano kartę graficzną z procesorem G80 (Nvidia GeForce 8800GTS). Oprogramowanie napisano z wykorzystaniem CUDA w języku C z rozszerzeniem CUDA.



Rys. 4. Czas obliczeń dla preselekcji HT

Fig. 4. Computation time for the preselection based HT

W implementacji wykorzystano niestandardowy przydział wątków od pikseli obrazu. Wątki przydzielane są do piksela, który stanowi środek zestawu okręgów. Wynikiem pośrednie dla danego bloku są synchronizowane przez porównanie wartości.

4. Wnioski

Liczba analizowanych okręgów wpływa w przybliżeniu liniowo na czas obliczeń. Inaczej jest z ilością punktów (pikseli) okręgu, które są równomiernie rozmieszczone. Dla małej ilości okręgów zysk nie jest zauważalny, jednak dla dużej liczby okręgów wyraźnie występuje zmniejszenie czasu obliczeń, nawet dwukrotnie. Pokazuje to, że zaproponowane rozwiązanie jest znacznie bardziej wydajne od rozwiązań [3], a także od zwykłej metody realizującej transformatę Hougha.

Wyniki zależą także od zawartości obrazu. Badania wpływu zawartości samego obrazu na czas obliczeń, wymaga zastosowania metody Monte Carlo (testowania dla bardzo dużej liczby różnych obrazów zarejestrowanych w różnych warunkach). W praktyce należy spodziewać się, że zysk algorytmu jest zmienny, jednak czas obliczeń jest skrócony, względem zwykłej metody.

5. Literatura

- [1] Hough P.V.C., Arbor A.: Method and Means for Recognizing Complex Patterns, US Patent no. 3,069,654, 1962.
- [2] Debevec P.: Rendering Synthetic Objects into Real Scenes: Bridging Traditional and Image-based Graphics with Global Illumination and High Dynamic Range Photography, SIGGRAPH 98, 189-198, 1998.
- [3] Mazurek P.: Optimalizacja transformaty Hougha dla okręgu z wykorzystaniem techniki odrzucenia implementowana na GPGPU, Measurement Automation and Monitoring, vol.58, no.7, 587-589, 2012