

Przemysław MAZUREK, Krzysztof OKARMA
ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY W SZCZECINIE,
ul. 26. Kwietnia 10, 71-126 Szczecin

Zastosowanie przetwarzania równoległego w procesorach graficznych do szybkiej estymacji wskaźnika wielorozdzielczego podobieństwa strukturalnego dla celów oceny jakości obrazów cyfrowych

Dr inż. Przemysław MAZUREK

Adiunkt w Katedrze Przetwarzania Sygnałów i Inżynierii Multimedialnej Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie. Absolwent Wydziału Elektrycznego (1998), stopień doktora nauk technicznych uzyskał w 2002 r. na Wydziale Elektrycznym Politechniki Szczecińskiej. Autor ponad 100 publikacji z zakresu cyfrowego przetwarzania sygnałów oraz estymacji kinematyki obiektów.



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

Dr inż. Krzysztof OKARMA

Adiunkt w Katedrze Przetwarzania Sygnałów i Inżynierii Multimedialnej Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie. Absolwent Wydziału Elektrycznego (1999) i Informatyki (2001) Politechniki Szczecińskiej, stopień doktora nauk technicznych uzyskał na Wydziale Elektrycznym (2003), gdzie pełni funkcję prodziekana ds. studenckich. Autor ponad 100 publikacji z zakresu przetwarzania i analizy obrazów oraz sygnałów.



e-mail: okarma@zut.edu.pl

Streszczenie

W artykule zaprezentowano technikę estymacji jakości obrazów opartą na implementacji zmodyfikowanego wskaźnika MS-SSIM z wykorzystaniem technik programowania procesorów graficznych. Ze względu na specyfikę technologii CUDA nie obliczano wszystkich lokalnych wartości wskaźników, jednak te same dane w znacznej części są wykorzystane dla kolejnych poziomów analizy. Weryfikację wyników przeprowadzono dla powszechnie używanej bazy LIVE zawierającej obrazy z pięcioma typowymi rodzajami zniekształceń.

Słowa kluczowe: ocena jakości obrazów cyfrowych, programowanie układów GPGPU, podobieństwo strukturalne.

Application of parallel processing to GPGPUs for fast estimation of the Multi-Scale Structural Similarity index for digital image quality assessment purposes

Abstract

In the paper an efficient method for fast estimation of the image quality is presented. It is based on implementation of the modified Multi-Scale Structural Similarity Index using the CUDA technology and the GPGPU architecture. Considering the architecture specific character, some of the local SSIM index values have not been calculated. Nevertheless, the most part of this information is used at next levels in the multi-scale analysis. The calculation procedure of the local SSIM index for each level is similar to the previously published estimation method of the single scale SSIM index, which uses 32 pixels wide blocks with the 8×8 pixels sliding window. Nevertheless, all the operations are performed using float numbers instead of the unsigned chars. Verification of the results has been performed using the LIVE Image Quality Assessment Database Release 2 containing the images corrupted by five common types of distortions together with their subjective quality evaluations expressed as the Differential Mean Opinion Score (DMOS) values. The obtained estimation results are much better correlated with the subjective quality evaluations than the original MS-SSIM values without using any nonlinear mapping. The average calculation time for a single image from the LIVE database is about 10 ms and is about 30 times shorter than the full MS-SSIM calculation using the typical Matlab implementation.

Keywords: image quality assessment, GPGPU programming, structural similarity.

1. Wstęp

Współczesne techniki przetwarzania i analizy obrazów w znacznej części operują na obrazach o niedoskonaliej jakości, co może być spowodowane zarówno przez zastosowanie stratnej kompresji, jak również zniekształcenia powstające podczas akwizycji lub transmisji obrazów.

Biorąc pod uwagę znaczący wpływ jakości obrazu na efekt działania algorytmu jego przetwarzania lub analizy, można uznać, iż jednym z najistotniejszych kierunków badań związanych z analizą obrazów jest dążenie do zapewnienia wiarygodnej automatycznej oceny ich jakości. Pierwsze porównawcze wskaźniki jakości obrazów [1] oparte były na błędzie średniokwadratowym, jednak charakteryzowały się one bardzo niską korelacją z ocenami subiektywnymi. Z tego względu w ostatnich latach zaproponowano wiele innych technik, które w znacznej mierze stanowią rozwiązanie tego problemu pozwalając m.in. na dużo rzetelniejszą ocenę nowych algorytmów przetwarzania obrazów (np. filtracji), czy też ich stratnej kompresji. W ogromnej większości są to wskaźniki o charakterze porównawczym, głównie ze względu na swoją dużą uniwersalność, której pozbawione są tzw. metody „ślepej oceny jakości” [2] nie wymagające znajomości obrazu oryginalnego (referencyjnego). Zazwyczaj miary takie są czule jedynie na wybrane rodzaje zniekształceń np. efekty kompresji JPEG [3] lub rozmycie [4].

2. Podobieństwo strukturalne w ocenie jakości obrazów

Jednym z najczęściej wykorzystywanych obiektywnych porównawczych wskaźników jakości obrazów jest podobieństwo strukturalne (Structural Similarity – SSIM), co wynika zarówno z jego prostej formy i wygodnej implementacji, jak również ze stosunkowo wysokiej korelacji z ocenami subiektywnymi. Wskaźnik ten, zaproponowany [5] jako modyfikacja wcześniej przedstawionej idei tzw. uniwersalnego wskaźnika jakości obrazu (Universal Image Quality Index – UIQI) [6], doczekał się kilkunastu modyfikacji, spośród których najczęściej wykorzystywana jest wersja wielorozdzielcza (Multi-Scale SSIM) [7].

Podstawowym krokiem algorytmu wyznaczania podobieństwa strukturalnego jest obliczenie lokalnego wskaźnika zgodnie z uproszczoną zależnością:

$$SSIM = \frac{(2\mu_x\mu_y + C_1)}{(\mu_x^2 + \mu_y^2 + C_1)} \cdot \frac{(2\sigma_{xy} + C_2)}{(\sigma_x^2 + \sigma_y^2 + C_2)}, \quad (1)$$

gdzie C_1 oraz C_2 oznaczają wartości stałych współczynników zapobiegających niestabilności wyników dla fragmentów obrazu o stałym kolorze, w szczególności czarnym. Wartości zalecane przez twórców tego wskaźnika wynoszą odpowiednio $(0.01 \times L)^2$ oraz $(0.03 \times L)^2$, gdzie L jest liczbą dostępnych poziomów luminancji, gdyż miara jest tradycyjnie zdefiniowana dla obrazów monochromatycznych (w skali szarości). Pozostałe wielkości występujące we wzorze (1) są obliczane następująco (dla okna przesuwonego o rozmiarze $N \times N$ pikseli):

$$\mu_x = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N x_{ij}, \quad \mu_y = \frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N y_{ij}, \quad (2)$$

$$\sigma_x^2 = \frac{1}{N^2 - 1} \sum_{i=1}^N \sum_{j=1}^N (x_{ij} - \mu_x)^2, \quad \sigma_y^2 = \frac{1}{N^2 - 1} \sum_{i=1}^N \sum_{j=1}^N (y_{ij} - \mu_y)^2, \quad (3)$$

$$\sigma_{xy} = \frac{1}{N^2 - 1} \sum_{i=1}^N \sum_{j=1}^N (x_{ij} - \mu_x)(y_{ij} - \mu_y). \quad (4)$$

Zalecanym przez autorów miarą oknem dwuwymiarowym jest okno Gaussa o rozmiarze 11×11 pikseli, jednak ze względu na zwiększenie wydajności obliczeń możliwe jest również zastosowanie innych funkcji okien, w szczególności okna prostokątnego o rozmiarze 8×8 pikseli, jak proponowano dla wskaźnika UIQI [6]. Rozwiązanie takie zostało również zastosowane we wcześniejszej publikacji [8] dotyczącej estymacji wartości tego wskaźnika jakości z wykorzystaniem GPGPU.

W celu zwiększenia korelacji z ocenami subiektywnymi zaproponowano [7] bardziej złożoną obliczeniowo wielorozdzielczą wersję tego wskaźnika, znaną jako Multi-Scale SSIM, w której część obliczeń związana z określeniem utraty kontrastu oraz struktury obrazu wykonywana jest dla pięciu poziomów. Kolejne poziomy uzyskiwane są w wyniku dwukrotnego zmniejszenia poziomu i pionowego rozmiaru obrazu wraz z filtracją dolnoprzepustową z wykorzystaniem filtru Gaussa. Uzyskane wyniki dla każdego poziomu są następnie mnożone z odpowiednimi wskaźnikami wagowymi w wykładnikach.

Ze względu na konieczność wykonywania dodatkowych operacji zmiennoprzecinkowych z wykorzystaniem arbitralnie dobranych wartości współczynników wagowych, a także konieczność zastosowania funkcji okna dwuwymiarowego, w niniejszej publikacji zdecydowano się na uproszczenie procedury obliczeniowej poprzez rezygnację zarówno ze współczynników wagowych, jak również zastąpienie okna Gaussa oknem prostokątnym o rozmiarze 8×8 pikseli, co jest uzasadnione m.in. wynikami wcześniejszych badań eksperymentalnych [9].

3. Idea szybkiej estymacji jakości obrazów

Podstawowym zadaniem estymacji jakości obrazów jest umożliwienie szybkiej, choć niekoniecznie dokładnej, oceny jakości obrazu. Biorąc pod uwagę fakt, iż „doskonała” obiektywna miara jakości, idealnie skorelowana z ocenami subiektywnymi, jak dotąd nie została opracowana, uznać można, iż właściwie każda znana metoda oceny jakości służy właściwie jedynie do jej, mniej lub bardziej dokładnej, estymacji. Z tego względu, szczególnie dla wskaźników o stosunkowo wysokiej złożoności obliczeniowej, w praktycznych zastosowaniach dużo istotniejsza jest efektywna implementacja aniżeli dokładne wyznaczenie wartości danego wskaźnika. Jedną z możliwości takiej implementacji jest wykorzystanie możliwości obliczeniowych współczesnych procesorów graficznych, co wymaga jednak odpowiedniego przygotowania i organizacji danych. W efekcie możliwe jest dokonanie szybkiej estymacji wartości wskaźnika SSIM lub MS-SSIM, jednak kolejne obliczenia mające na celu wyznaczenie jego dokładnej wartości musiałyby być zaimplementowane w mniej efektywny sposób, co zauważalnie pogorszyłoby wydajność.

Programowalne procesory graficzne pozwalają na równoległe obliczeń poprzez wykorzystanie uniwersalnych jednostek przetwarzania. Podstawowym elementem dla platformy CUDA jest jądro (kernel), będący funkcją napisaną w zmodyfikowanym języku C przypisaną do wątku. Wątki są organizowane w bloki i wykonywane równoległe przez GPGPU. Wątki takie mogą wymieniać dane między sobą w efektywny sposób z wykorzystaniem pamięci współdzielonej znajdującej się wewnątrz GPGPU, która jest dość wygodna do celów realizacji niewielkich buforów różnego rodzaju m.in. kolejek FIFO. Zasadnicza pamięć karty graficznej (Global Memory) znajdująca się na karcie jest jednak bardzo

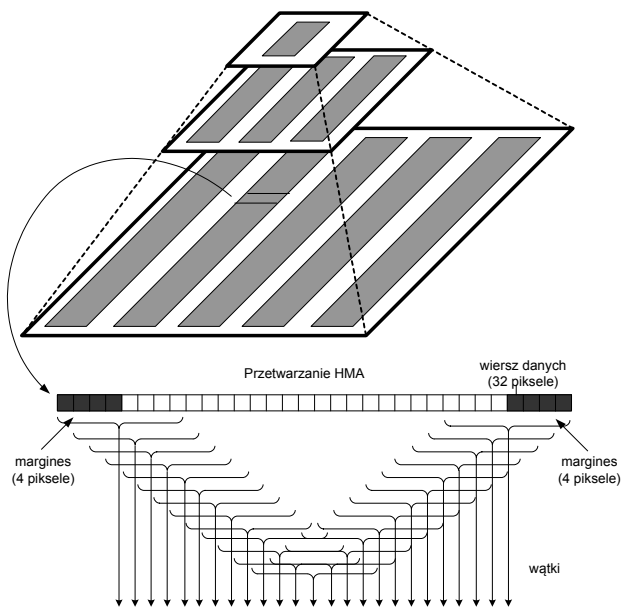
powolna w stosunku do możliwości GPGPU, a zatem wskazane jest tworzenie kodu w taki sposób, aby można było ograniczyć transfery danych pomiędzy GPGPU a tą pamięcią. Warto zwrócić uwagę na fakt, iż najwolniejszy rodzaj transmisji dotyczy dostarczania danych do pamięci karty graficznej z pamięci komputera (Host Memory).

Bloki organizowane są w siatkę (Grid), której elementy są także wykonywane równoległe, ale na kolejność wykonywania przy ograniczonych zasobach programista nie ma wpływu. Zwykle organizacja wątków i bloków jest powiązana bezpośrednio z organizacją danych w pamięci, jednak nie jest to jedyne rozwiązanie, gdyż możliwe jest przydzielenie różnych bloków do wykonywania różnych operacji. Wówczas jeden kernel obsługuje różne algorytmy, które wykonywane są w zależności od określonego bloku (pozyjacji na siatce), co pozwala na lepsze wykorzystanie możliwości obliczeniowych GPGPU bez konieczności maksymalnego zrównoleglenia algorytmu, które może być kłopotliwe w implementacji. Oznacza to, iż typowe wskaźniki efektywności implementacji bazujące na liczbie wykorzystywanych jednocześnie bloków, liczbieostępów do pamięci, konfliktów pamięci itp., nie mają istotnego znaczenia (są jedynie miernikami pozwalającymi na dopracowanie algorytmu). Jedynym znaczącym wskaźnikiem dla określonej aplikacji (zawierającej wiele wirtualnych kerneli wykonywanych równoległe w ramach jednego kernela programowego) jest czas jej wykonywania.

Ze względu na konieczność przechowywania obrazu w pamięci kart graficznej możliwe są różne implementacje wskaźnika MS-SSIM. Szczególnie interesującym rozwiązaniem jest zastąpienie obliczania pełnego wskaźnika MS-SSIM jego estymacją, która jest dostosowana do uwarunkowań wynikających z zastosowanej karty graficznej. Zamiast obliczania wartości wskaźnika dla każdego poziomu, można dokonać ich estymacji dla zestawu pionowych pasów obrazu, co jest szczególnie istotne z punktu widzenia dostępu do pamięci. Równoczesny dostęp do grupy pikseli z adresowaniem uwzględniającym możliwości karty graficznej zapewnia redukcję czasów oczekiwania na dane, co przyspiesza wydajność przetwarzania. W przedstawianej implementacji wykorzystano pasy o szerokości 32 pikseli, co pozwala na wyznaczenie 25 i utratę 7 wartości lokalnych przy szerokości okna SSIM równej 8 pikseli obrazu. Taka organizacja wątków pozwala na przydział stosunkowo niewielkiej liczby jednostek do przetwarzania obrazu. Na pierwszym poziomie przydział ten jest ograniczony możliwościami procesora i wtedy występuje automatyczne zrównoleglenie przetwarzania przez CUDA, natomiast na kolejnych poziomach przydziały jednostek przetwarzania zostają zredukowane w związku ze zmniejszeniem rozmiaru obrazu. W praktyce nie stanowi to problemu wydajnościowego z uwagi na możliwość warunkowego wykonywania innych wirtualnych kerneli. Wykorzystana implementacja bazuje na kilkukrotnym uruchamianiu kerneli dla każdego z poziomów, jak pokazano na rysunku 1.

Obliczenie wskaźnika MS-SSIM, podobnie jak SSIM, dla obrazu monochromatycznego (w skali szarości) wymaga operacji na dwuwymiarowych blokach obrazu (w rozpatrywanym przypadku o rozmiarze 8×8 pikseli), dla których można wykorzystać bufory FIFO oraz rekurencyjną postać wzorów dla każdego poziomu. Dla każdego z nich konieczne jest przechowywanie obrazu w bloku o 32 pikselach szerokości i 8 pikselach wysokości) oraz wyniku FIFO z aktualnymi wartościami średniej. Dane przechowywane są w pamięci jako liczby zmiennoprzecinkowe (float), tryb 8-bitowy może być warunkowo zastosowany jedynie dla pierwszego poziomu w przypadku użycia innego kodu kernela, co pozwala na przyspieszenie obliczeń. Skalowanie obrazu wymaga jednak wyznaczenia wartości przechowywanych na 32 bitach (float) dla wyższych poziomów.

W implementacji wykorzystano platformę CUDA 3.2 dla karty NVidia GeForce 8800 GTS (NVidia G80 GPU - 128 procesorów strumieniowych, taktowanie rdzenia 650MHz, taktowanie jednostek shader 1625MHz, taktowanie pamięci 1944MHz, 256-bitowy interfejs pamięci, interfejs PCI Express x16).



Rys. 1. Model przetwarzania estymaty MS-SSIM z użyciem HMA (Horizontal Moving Average)
 Fig. 1. Model of processing the MS-SSIM estimate using the HMA

Warto zauważyć, iż istotną zaletą algorytmu MS-SSIM jest niskie zapotrzebowanie na pamięć, co umożliwia efektywną implementację przez połączenie wykonywania kodu SSIM z innym kodem przetwarzającym obraz w celu eliminacji wielokrotnych dostępów do pamięci obrazu.

Estymacja wskaźnika SSIM pomija część pikseli [8], jednak metoda estymacji wielorozdzielczej zaproponowana w artykule częściowo je wykorzystuje. Przykładowo dla pasów pionowych 1 i 2 dla pierwszego poziomu wartości pikseli pomiędzy nimi są pomijane, jednak dla następnych poziomów są one używane z uwagi na zmianę rozdzielczości obrazu, co stanowi dużą zaletę proponowanej metody w stosunku do estymacji wskaźnika SSIM.

4. Weryfikacja eksperymentalna

W celu określenia zgodności wyznaczonej estymaty wskaźnika MS-SSIM z ocenami subiektywnymi wykorzystano bazę LIVE Image Quality Database Release 2 [10] zawierającą 982 obrazy (w tym 203 referencyjne) z pięcioma typowymi rodzajami zniekształceń oraz ich oceny subiektywne wyrażone jako wartości DMOS. Uzyskane wskaźniki korelacji liniowej Pearsona z wartościami ocen DMOS dla wskaźnika MS-SSIM oraz jego estymaty wyznaczonej w sposób proponowany w artykule zaprezentowano w tabeli 1. Obliczenia zostały przeprowadzone dla obrazów przekonwertowanych do skali szarości, a uzyskany średni czas estymacji rzędu 10 ms okazał się około 30-krotnie krótszy od średniego czasu obliczenia wskaźnika MS-SSIM w środowisku Matlab.

Tab. 1. Współczynniki korelacji liniowej wskaźnika MS-SSIM z ocenami subiektywnymi (wartościami DMOS)
 Tab. 1. Linear correlation coefficients between the MS-SSIM index values and subjective evaluation (DMOS values)

Korelacja wskaźnika	Zniekształcenie	Kompresja JPEG 2000	JPEG	Szum biały	Rozmycie Gaussa	Transmisja przez kanał Rayleigha	Wszystkie
	Pełny MS-SSIM	0.6995	0.8155	0.7489	0.7023	0.7046	0.4762
Estymata 4-poziomowa	0.8297	0.8311	0.8812	0.8492	0.8134	0.8036	
Estymata 3-poziomowa	0.8264	0.8271	0.8850	0.8385	0.8154	0.7971	
Estymata 2-poziomowa	0.8265	0.8240	0.8951	0.8323	0.8148	0.7879	

Ze względu na ograniczenia rozdzielczości obrazów zamieszczonych w wykorzystywanej bazie oraz specyfikę zastosowanej implementacji wymagającej użycia 32-pikselowych pasów ograniczono liczbę poziomów analizy do czterech. Dla porównania w tabeli 1 przedstawiono również wartości korelacji uzyskane przy zastosowaniu jedynie pierwszych trzech oraz dwóch poziomów.

5. Podsumowanie

Zaproponowana metoda szybkiej estymacji wartości wskaźnika wielorozdzielczego podobieństwa strukturalnego charakteryzuje się znacznym zwiększeniem szybkości obliczeń w porównaniu z typową implementacją oraz, co ważne, jest znacznie lepiej skorelowana z subiektywnymi ocenami jakości obrazów.

Uzyskane wartości korelacji są bardzo wysokie, szczególnie biorąc pod uwagę, iż jest to korelacja liniowa uzyskiwana bez żadnego dodatkowego mapowania nieliniowego, jak ma to miejsce w wielu publikacjach. Uzyskanie wysokiego współczynnika korelacji z wartościami ocen DMOS dla pełnej wersji wskaźnika MS-SSIM wymaga zastosowania takiego mapowania np. w postaci funkcji logistycznej zgodnie z zaleceniami Video Quality Experts Group (VQEG). Takie podejście jest jednak mało efektywne w praktyce ze względu na fakt, iż wymaga ono dodatkowej optymalizacji współczynników dla funkcji opisującej krzywą logistyczną.

Proponowane efektywne podejście do estymacji jakości obrazów może być również zmodyfikowane dla celów szybkiej oceny jakości sekwencji wideo, w szczególności w zastosowaniach związanych z transmisją sekwencji w warunkach zmiennej przepustowości łącza.

6. Literatura

- [1] Eskicioglu A.: Quality Measurement for Monochrome Compressed Images in the Past 25 Years. Proceedings of IEEE International Conference on Acoustics Speech and Signal Processing, pp. 1907–1910, Istanbul, Turkey, 2000.
- [2] Li X.: Blind Image Quality Assessment. Proceedings of IEEE International Conference on Image Processing, pp. 449–452, 2002.
- [3] Meesters L., Martens J.B.: A Single-Ended Blockiness Measure for JPEG-Coded Images. Signal Processing vol. 82 no. 3, pp. 369–387, 2002.
- [4] Marziliano P., Dufaux F., Winkler S., Ebrahimi T.: A No-Reference Perceptual Blur Metric. Proceedings of IEEE International Conference on Image Processing, pp. 57–60, Rochester, USA, 2002.
- [5] Wang Z., Bovik A.C., Sheikh H., Simoncelli E.: Image Quality Assessment: From Error Measurement to Structural Similarity. IEEE Transactions on Image Processing vol. 13 no. 4, pp. 600–612, 2004.
- [6] Wang Z., Bovik A.C.: A Universal Image Quality Index. IEEE Signal Processing Letters vol. 9 no. 3, pp. 81–84, 2002.
- [7] Wang Z., Simoncelli E., Bovik A.: Multi-Scale Structural Similarity for Image Quality Assessment. Proceedings of the 37th IEEE Asilomar Conference on Signals, Systems and Computers, Pacific Grove, California, USA, 2003.
- [8] Mazurek P., Okarma K.: An Efficient Estimation of the Structural Similarity Index Using the GPGPU Programming Techniques. Measurements, Automation and Monitoring vol. 56 no. 7, pp. 668–670, 2010.
- [9] Okarma K.: Influence of the 2-D Sliding Windows on the Correlation of the Digital Image Quality Assessment Results Using the Structural Similarity Approach with the Subjective Evaluation. Electrical Review, vol. 86 no. 7, pp. 109–111, 2010.
- [10] Sheikh H.R., Wang Z., Cormack L., Bovik A.C.: LIVE Image Quality Assessment Database Release 2. <http://live.ece.utexas.edu/research/quality>