

Wojciech Bieniecki\*, Michał Węgrzyn\*, Szymon Grabowski\*,  
Katarzyna Kościelska-Kasprzak\*\*, Dominika Drulis-Fajdasz\*\*,  
Oktawia Mazanowska\*\*, Marian Klinger\*\*

## **Zastosowanie algorytmów progowania adaptacyjnego do segmentacji barwnych obrazów mikroskopowych w badaniu metodą ELISPOT**

### **1. Wprowadzenie**

Zespoły Katedry Informatyki Stosowanej Politechniki Łódzkiej oraz Katedry i Kliniki Nefrologii i Medycyny Transplantacyjnej Akademii Medycznej we Wrocławiu prowadzą prace nad systemem automatycznej analizy obrazów mikroskopowych metodą ELISPOT [8, 9]. Nasze wcześniejsze badania koncentrowały się na przetwarzaniu obrazów monochromatycznych, obecnie staramy się jednak wykorzystać informację barwną. W skład tworzonego systemu SpotView wchodzi algorytmy wykrywania obszaru zainteresowania, algorytmy przetwarzania wstępnego obrazów monochromatycznych, algorytmy segmentacji obrazów monochromatycznych oraz algorytmy analizy ilościowej obrazu [10, 12]. W niniejszej pracy zaprezentowano podejście do segmentacji obrazu poprzez progowanie składowych barwy, metodę będącą szczególnym przypadkiem segmentacji przestrzeni barw.

### **2. Przegląd metod klasyfikacji pikseli**

Klasyfikacja pikseli może służyć jako procedura wstępna do segmentacji przez łączenie obszarów. Jej celem jest przyporządkowanie każdego badanego punktu do pewnej określonej klasy. Klasyfikacja może być przeprowadzona w sposób nadzorowany lub nienadzorowany. Klasyfikacja nienadzorowana określana jest także terminami: klasteryzacja, grupowanie, taksonomia, analiza skupień [1, 5]. Polega ona na wydzieleniu w przestrzeni cech spójnych obszarów, tzn. klastrów (skupień) pikseli. Zwykle przestrzeń cech wiąże się z wybraną przestrzenią barw (RGB, HSV), dodatkowo można przypisać pikselowi takie

---

\* Katedra Informatyki Stosowanej, Politechnika Łódzka

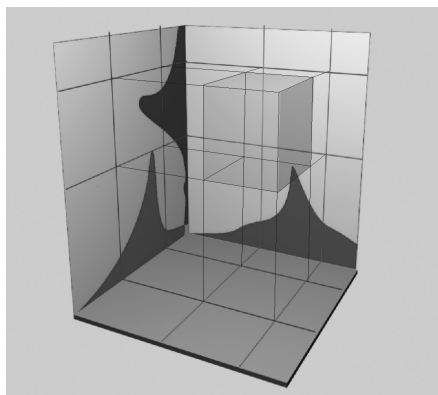
\*\* Katedra i Klinika Nefrologii i Medycyny Transplantacyjnej, Akademia Medyczna we Wrocławiu

cechy, jak jasność lub cechy kontekstowe, czyli związane z jego sąsiedztwem (wariancja jasności, kontrast, parametry tekstury). Klasy, które się definiuje, to najczęściej tło oraz klasy poszukiwanych obiektów (w zależności od badanego obrazu). W badaniu ELISPOT należy wyróżnić trzy klasy obiektów: czerwone, fioletowe i niebieskie. Zadaniem klasyfikatora jest takie wydzielenie skupień, by w jak największym stopniu pokrywały się one z poszukiwanymi obiektami (w tym jedno z nich z tłem).

Najprostszą i najszybszą metodą klasteryzacji jest klasyczne progowanie, przypisujące wartość 1 pikselom o jasności (lub innym atrybucie) większej lub równej pewnej wartości progowej  $T$ , a 0 pozostałym. Trudność polega na wyznaczeniu takiej wartości  $T$ , aby algorytm był skuteczny, tj. dobrze dyskryminował obiekty względem tła. W przypadku progowania globalnego,  $T$  jest ustalone arbitralnie dla całego obrazu najczęściej na podstawie charakterystyk globalnych. Wymienić należy tu algorytmy bazujące na kombinacji liniowej średniej oraz odchylenia standardowego obrazu źródłowego [7], algorytm Otsu [2], metodę iteracyjnego doboru progu isodata [6, 11] bazującą na średnich jasnościach tła, metody statystyczne: SIS [3], działające na podstawie średniej, na podstawie mediany, na podstawie środka przedziału jasności.

Wymienione podejścia można zastosować w odniesieniu do całej powierzchni obrazu lub tylko fragmentu; w drugim przypadku mówimy o tzw. progowaniu lokalnym. Progowanie lokalne jest szczególnie użyteczne np. dla obrazów z nierównym rozkładem oświetlenia.

Progowanie obrazów barwnych jest operacją bardziej złożoną od progowania obrazów monochromatycznych. Wynika to z faktu, że w obrazach monochromatycznych każdy piksel opisany jest tylko pojedynczą wartością (jasności). W obrazach barwnych (RGB) występują trzy wartości jasności: jedna dla każdej składowej – czerwonej, zielonej i niebieskiej. Dopiero wartości wszystkich trzech składowych dają nam informacje o barwie badanego punktu. Metodologia procesu progowania obrazów jest bardzo podobna do progowania obrazów monochromatycznych z tą różnicą, że musimy zbadać trzy rozkłady jasności i ewentualne powiązania między nimi (rys. 1).



**Rys. 1.** Progowanie w przestrzeni RGB

### 3. Algorytmy progowania obrazów barwnych w badaniu ELISPOT

Do naszych badań wybrano trzy algorytmy: algorytm Bernsena [4], oraz dwie metody bazujące na przebiegu histogramu. Wybór metod był związany z rodzajem obrazów, które podlegały badaniu. Metoda Bernsena została zaimplementowana zgodnie z oryginalnymi założeniami (z niewielką modyfikacją), natomiast metody bazujące na histogramie zostały opracowane na nowo. Poniżej zostaną przedstawione teoretyczne założenia tych algorytmów, sposób ich implementacji oraz napotkane problemy i ich rozwiązania.

#### 3.1. Metoda Bernsena

Metoda ta jest prosta w teorii i implementacji. Dla badanego obszaru wyznaczanego w obrębie obrazu za wartość proggu przyjmuje się średnią pomiędzy wartością minimalną a wartością maksymalną w tym obszarze (wzór (1)). Chociaż w zasadzie możliwe jest użycie tej metody globalnie, to na ogół stosuje się ją w progowaniu lokalnym, i wyżej wymienione wartości wyznacza się dla okna o ustalonym rozmiarze lub wręcz indywidualnie dla każdego piksela.

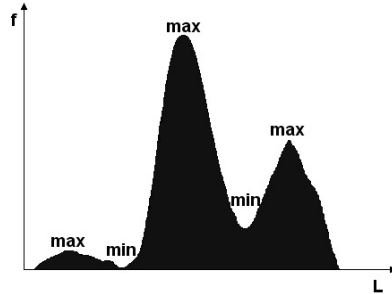
$$T(i, j) = \frac{1}{2} \cdot \left[ \max_{p \in M_{ij}} (f(i, j)) + \min_{p \in M_{ij}} (f(i, j)) \right] = \frac{1}{2} \cdot [l_{\max}(i, j) + l_{\min}(i, j)] \quad (1)$$

gdzie:  $M_{ij}$  jest oknem o rozmiarach  $b \times b$  z pikselem  $(i, j)$  w centrum. Symbole  $l_{\max}$  i  $l_{\min}$  oznaczają wartości maksimum i minimum dwuwymiarowej funkcji  $f$  jasności obrazu w obrębie  $M_{ij}$ .

Metoda ta wymaga sprawdzenia jednorodności badanego obszaru. Należy bowiem uwzględnić fakt, że badany obszar może nie zawierać szukanych obiektów, natomiast progowanie w tym obszarze rozdzieli go na dwa podzbiory. Aby wykluczyć taką sytuację, wyznaczenie proggu wykonuje się dopiero wtedy, gdy okaże się, że warunek jednorodności obszaru nie jest spełniony. W wersji oryginalnej metody [4] sprawdza się kontrast według wzoru  $C(i, j) = l_{\max}(i, j) - l_{\min}(i, j)$ : jeśli jest zbyt mały, to przyjmuje się, że obszar jest jednorodny i rezygnuje się z dalszych działań, w przeciwnym zaś razie przystępuje się do wyznaczania proggu. Podejście to ma zasadniczą wadę: jest bardzo wrażliwe na pojedyncze zakłócenia (np. tzw. „gorące” piksele). Nawet jeden taki piksel w badanym obszarze zwykle „wymusi” podział skądinąd jednorodnego obszaru na dwa. Wprowadziliśmy modyfikację polegającą na zastąpieniu powyższego wzoru odchyleniem standardowym  $\sigma$  dla badanego obszaru. Obszar uznany jest za jednorodny wtedy i tylko wtedy, gdy  $\sigma$  jest mniejsze od pewnej ustalonej eksperymentalnie wartości. Rzecz jasna, takie kryterium jest dużo mniej wrażliwe na pojedyncze nietypowe piksele.

### 3.2. Metody bazujące na przebiegu histogramu

Obydwie metody zastosowane w tej pracy korzystają z występowania w histogramie lokalnych maksimum i minimum: tzw. wznórz (*peaks*) i dolin (*valleys*) (rys. 2).



Rys. 2. Maksima i minima w histogramie

Algorytm poszukuje ciągu trzech punktów: maksimum – minimum – maksimum. Miejsca występowania takich sekwencji świadczą, że występują tam dwie klasy obiektów o zróżnicowanych jasnościach, co sugeruje, że jest to miejsce, gdzie należy umieścić szukany próg. Niestety, może występować kilka takich miejsc. Dobrym rozwiązaniem jest wybór ciągu, w którym średnia wartości dwóch maksimum jest największa.

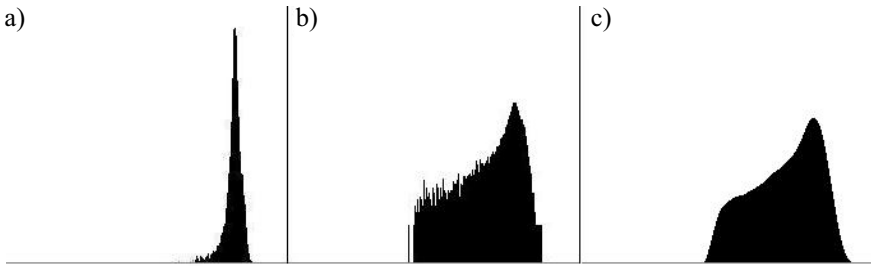
Pierwszy z wariantów został nazwany *Peaks & Valleys (valley)*, gdyż za próg jest przyjmowane miejsce występowania minimum w grupie trzech wyznaczonych punktów, o których była mowa wcześniej. Druga z metod, *Peaks & Valleys (half)*, wyznacza próg w połowie odległości między dwoma maksimum z grupy trzech wyznaczonych punktów.

Na rysunku 3 przedstawiony jest powiększony fragment jednego z obrazów, które są podane badaniu za pomocą różnych algorytmów zaimplementowanych w aplikacji SpotView. Dla tego fragmentu wyznaczony został histogram (rys. 4) oraz różne jego modyfikacje.

Dla fragmentu z rysunku 3 żadna operacja przeprowadzona na wyznaczonym histogramie nie przyniosła oczekiwanego efektu. Nawet po logarytmowaniu oryginalnego histogramu oraz jego uśrednieniu nie jesteśmy w stanie zaobserwować punktów, które spełniłyby warunek występowania minimum pomiędzy dwoma maksimumami. Dzieje się tak, ponieważ granica pomiędzy obiektami a tłem jest bardzo niewyraźna wskutek rozmycia. Rozwiązaniem tego problemu jest użycie dla danego obrazu filtru *Unsharp Mask*.

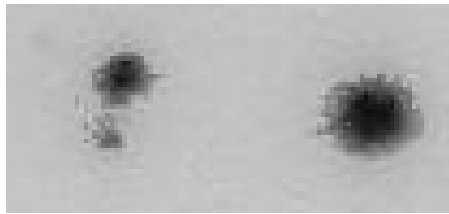


Rys. 3. Powiększony fragment obrazu bez filtracji wstępnej

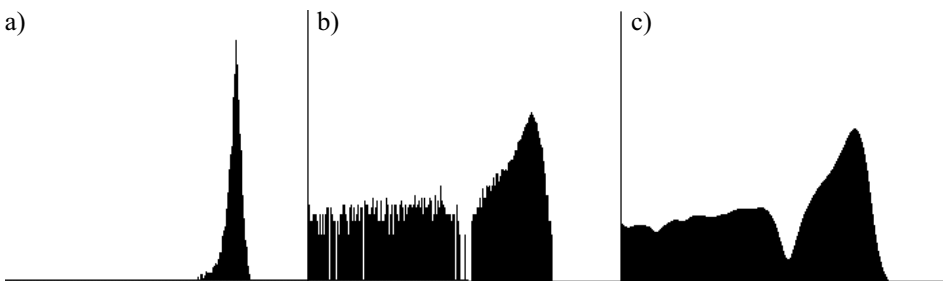


**Rys. 4.** Histogram obrazu bez filtru wyostrzającego: a) histogram normalny; b) histogram logarytmiczny; c) histogram logarytmiczny wygładzony

Jak widać na rysunkach 5 i 6, zastosowanie filtru wyostrzającego poprawiło widoczność obiektów, a także zwiększyło kontrast pomiędzy tymi obiektami a tłem. W przypadku obrazu przefiltrowanego, logarytmowanie jego histogramu wprowadza już istotne zmiany. Ponieważ histogram jest postrzępiony i mało czytelny, przeprowadzona została operacja jego uśredniania. Ostateczny przebieg histogramu (rys. 6c) pozwala bez problemu dostrzec miejsce minimum otoczonego dwoma maksimumi.



**Rys. 5.** Powiększony fragment obrazu po zastosowaniu filtru wyostrzającego

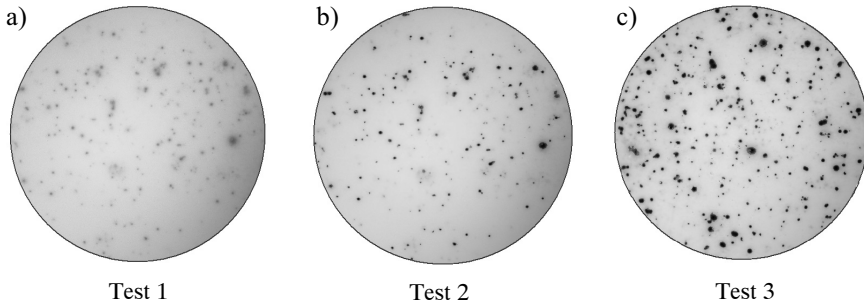


**Rys. 6.** Histogram obrazu z filtrem wyostrzającym: a) histogram normalny; b) histogram logarytmiczny; c) histogram logarytmiczny wygładzony

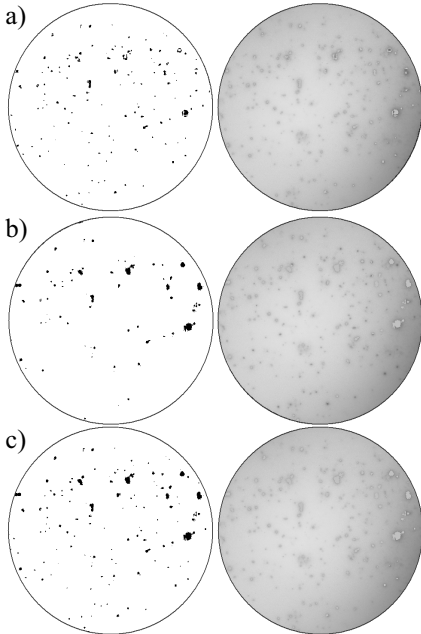
#### 4. Testy algorytmów

Zostały wykonane eksperymenty na trzech obrazach testowych: *test1*, *test2* i *test3*. Dla obrazu *test1* została zastosowana jedynie metoda Bernsena, ponieważ jest to obraz nie-

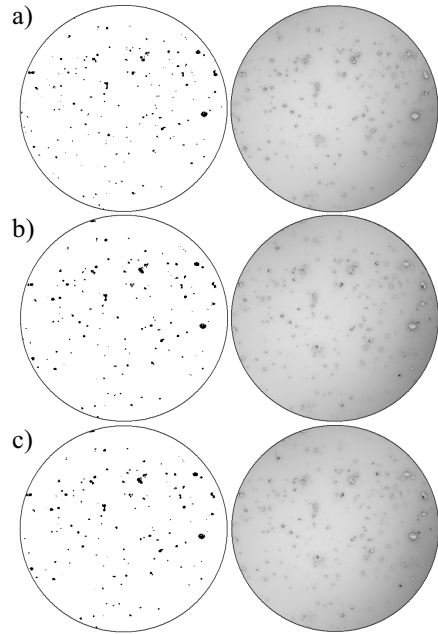
poddany filtracji i algorytmy bazujące na histogramie nie są w stanie dobrze poradzić sobie z jego progowaniem. Obrazy *test2* i *test3* zostały poddane filtrowaniu i dla nich można było porównać wyniki działania wszystkich trzech metod (rys. 7). Przedstawiono wyniki progowania i półprogowania lokalnego dla przebiegu pojedynczego z użyciem masek o rozmiarach  $15 \times 15$ ,  $75 \times 75$  oraz przebiegu podwójnego z użyciem tych samych masek (rys. 8–14, tab. 1). Przez półprogowanie rozumiemy nałożenie na obraz wejściowy czarnych punktów tam, gdzie został wykryty obiekt.



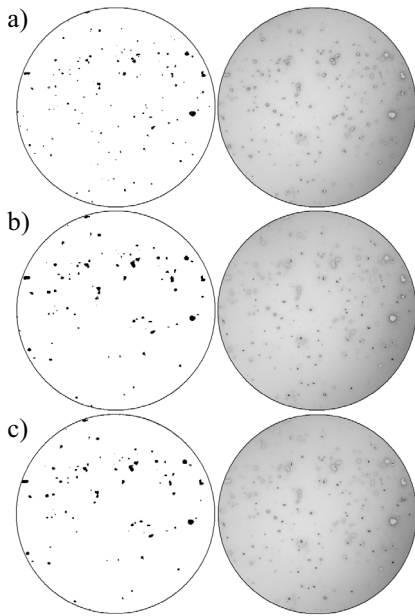
**Rys. 7.** Obraz testowy ( $2272 \times 1704$  pikseli) oryginalny (a) i przetworzony wstępnie (b, c)



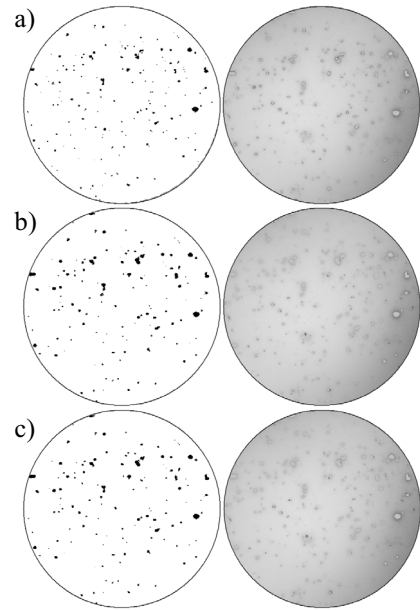
**Rys. 8.** Obraz *test1*, metoda Bernsena:  
a)  $15 \times 15$ ; b)  $75 \times 75$ ; c) dwuprzebiegowo



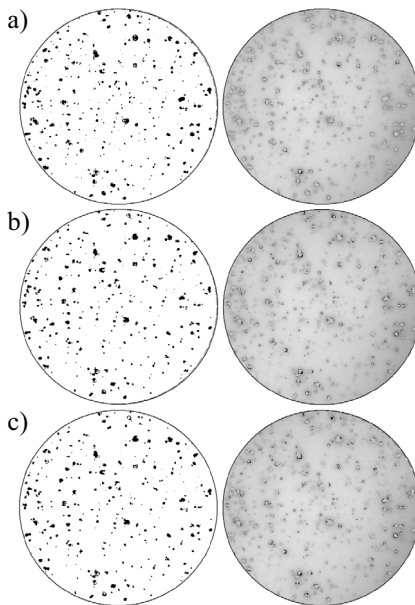
**Rys. 9.** Obraz *test2*, maska  $15 \times 15$ :  
a) Bernsen, b) PV (*half*), c) PV (*valley*)



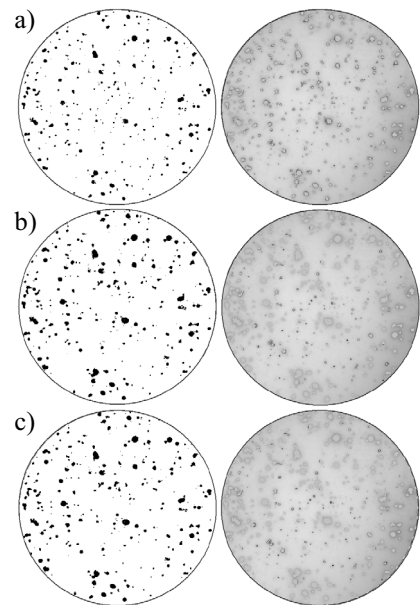
**Rys. 10.** Obraz *test2*, maska  $75 \times 75$ :  
a) Bernsen; b) PV (*half*); c) PV (*valley*)



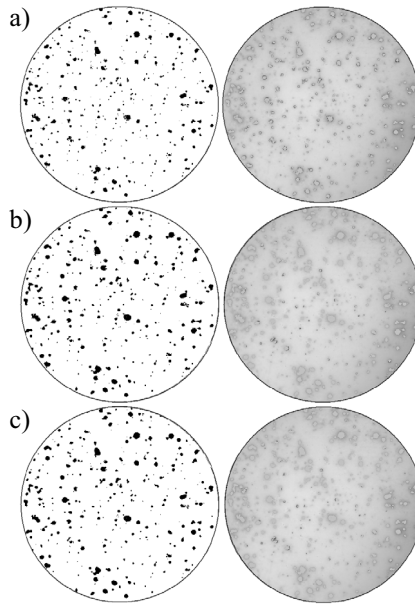
**Rys. 11.** Obraz *test2*, dwuprzbiegowo:  
a) Bernsen; b) PV (*half*); c) PV (*valley*)



**Rys. 12.** Obraz *test3*, maska  $15 \times 15$ :  
a) Bernsen; b) PV (*half*); c) PV (*valley*)



**Rys. 13.** Obraz *test3*, maska  $75 \times 75$ :  
a) Bernsen; b) PV (*half*); c) PV (*valley*)



**Rys. 14.** Obraz *test3*, dwuprzebiegowo:  
a) Bernsen; b) PV (*half*); c) PV (*valley*)

**Tabela 1**  
Czasy progowania [s] dla poszczególnych obrazów  
(AMD Athlon 1500+ 1,32 GHz, 256 MB RAM)

		Bernsen	PV ( <i>half</i> )	PV ( <i>valley</i> )
Test1	15×15	2,91	X	X
	75×75	2,88	X	X
	dwuprzebiegowo	5,14	X	X
Test2	15×15	2,95	5,67	5,66
	75×75	2,93	3,23	3,40
	dwuprzebiegowo	5,22	8,24	8,23
Test3	15×15	3,02	5,78	5,87
	75×75	2,99	3,30	3,36
	dwuprzebiegowo	5,37	8,41	8,55

## 5. Podsumowanie

Wszystkie metody przy odpowiednich ustawieniach dają bardzo zbliżone i bardzo zadowalające wyniki. Obrazy testowe bez przetworzenia wstępnego nie mogły być segmentowane algorytmami bazującymi na histogramie. Obrazy wynikowe pozwoliły dostrzec, że



użycie małej maski ( $15 \times 15$ ) powoduje, że niektóre większe elementy są wykrywane niedokładnie i w ich wnętrzu dają się zauważyć pewne źle sklasyfikowane miejsca. Użycie większej maski ( $75 \times 75$ ) spowodowało znaczące poprawienie tego mankamentu, jednak ta maska gorzej poradziła sobie z wykryciem bardzo małych obiektów. Spostrzeżenia te odnoszą się do wszystkich zastosowanych metod. Kompromisem jest przeprowadzenie dwóch przebiegów, używając obydwu masek. Operacja taka daje bardzo dobry efekt, ponieważ łączy zalety działania tych masek i niweluje ich wady. Niestety, operacja ta jest bardziej kosztowna obliczeniowo.

Obydwie metody bazujące na histogramie dają rezultaty porównywalne, jednak lepsze niż metoda Bernsena. Wykryte obiekty są pełniejsze i dokładniej rozpoznane. W metodzie Bernsena widoczne jest zjawisko pozostawiania niewykrytego brzegu obiektu.

Jeżeli chodzi o czas działania algorytmów, to zdecydowanie najlepiej wypada metoda Bernsena. Największa różnica jest widoczna dla małych masek. Jest to spowodowane tym, że metody bazujące na histogramie dokonują wielu operacji na tym histogramie i dla każdego okna muszą wykonać taką samą ich liczbę, a zatem im mniejsze okno, tym więcej operacji trzeba wykonać dla danego obszaru. Widać to dokładnie, porównując działanie metody *Peaks & Valleys (half)* dla masek  $15 \times 15$  i  $75 \times 75$ . Czas dla większej maski jest wyraźnie mniejszy.

## Literatura

- [1] Hanson A.R., Riseman E.R., *Segmentation of natural scenes*. Computer Vision Systems, Hanson and Riseman (Eds), Academic Press, NJ, 1978, 129–163.
- [2] Otsu N., *A threshold selection method from gray-level histograms*. IEEE Transactions on Systems, Man and Cybernetics, vol. SMC-9, Jan. 1979, 62–66.
- [3] Kittler J., Illingworth J., Foglein J., *Threshold selection based in a simple image statistic*. Computer Vision, Graphics and Image Processing, vol. 30, 1985, 125–147.
- [4] Bernsen J., *Dynamic thresholding of grey-level images*. Proceedings 8th International Conference on Pattern Recognition, Paris 1986, 1251–1255.
- [5] Celenk M., *A color clustering technique for color image segmentation*. Computer Vision, Graphics, and Image Processing, 52, 1990, 145–170.
- [6] Young I.T., Gerbrands J.J., van Vliet L.J., *Fundamentals of Image Processing*. The Netherlands at the Delft University of Technology, 1998.
- [7] Ritter G.X., Wilson J.N., *Computer Vision Algorithms in Image Algebra*. London, CRC Press 2001.
- [8] Gebauer B.S., Hricik D.E., Atallah A., Bryan K., Riley J., Tary-Lehmann M., Greenspan N.S., Dejeló C., Boehm B.O., Hering B.J., Heeger P.S., *Evolution of the enzyme-linked immunosorbent spot assay for post-transplant alloreactivity as a potentially useful immune monitoring tool*. Am. J. Transplant., vol. 2, 2002, 857–866.
- [9] Hricik D.E., Rodriguez V., Riley J., Bryan K., Tary-Lehmann M., Greenspan N., Dejeló C., Schullak J.A., Heeger P.S., *Enzyme linked immunosorbent spot (ELISPOT) assay for interferon-gamma independently predicts renal function in kidney transplant recipients*. Am. J. Transplant., vol. 3, 2003, 878–884.

- [10] Wójcicki D., Bieniecki W., Grabowski Sz., Kościelska-Kasprzak K., *Algorytmy przetwarzania wstępnego obrazów mikroskopowych w badaniu aktywności wydzielniczej limfocytów*. XIII konf. „Sieci i systemy informatyczne”, Łódź 2005, 573–580.
- [11] Strzecha K., Fabijańska A., Sankowski D., *Nowe algorytmy segmentacji w wysokotemperaturowym przemysłowym systemie analizy obrazów*. Automatyka (półrocznik AGH), t. 10, z. 3, 2006, 283–297.
- [12] Bieniecki W., Grabowski S., Kościelska-Kasprzak K., Drulis-Fajdasz D., Mazanowska O., Klinger M., *An Algorithm for Smart ROI Detection in ELISPOT Examination Images*. XV konf. „Sieci i systemy informatyczne”, Łódź 2007, 133–136.