

## WYKORZYSTANIE ADAPTACYJNEJ METODY REDUKCJI SZUMÓW NA OBRAZACH CYFROWYCH DO DETEKCJI KRAWĘDZI

### EDGE DETECTION USING ADAPTIVE SMOOTHING OF DIGITAL IMAGES

Jakub Kolecki, Andrzej Wróbel

Katedra Geoinformacji, Fotogrametrii i Teledetekcji Środowiska,  
Wydział Geodezji Górniczej Inżynierii Środowiska, AGH, Kraków

SŁOWA KLUCZOWE: przetwarzanie obrazów cyfrowych, detekcja krawędzi, metody adaptacyjne, lokalna estymacja MNW

STRESZCZENIE: Detekcja krawędzi często stanowi ważny etap przetwarzania obrazów cyfrowych w procedurach automatycznego pomiaru. Metody będące obecnie w użyciu, w większości oparte są na wykorzystaniu filtrów krawędziujących. Wymienić należy tu chociażby filtr LoG (Laplacian of Gaussian) jak również bardziej zaawansowaną metodę Cannyego-Derichea. W niniejszym artykule proponowane jest inne podejście do problemu detekcji krawędzi. Proponowane rozwiązanie jest oparte na wykorzystaniu metod statystyki matematycznej, w szczególności propagacyjno-separacyjnego podejścia do lokalnej metody największej wiarygodności, opracowanego w roku 2006 przez J. Polzehl'a i V. Spokoinego. Istotą metody jest adaptacyjne określenie sąsiedztwa każdego z pikseli, które wykorzystywane jest do estymacji jasności danego piksela. Sąsiedztwo to określane jest poprzez wagi przypisywane pikselom leżącym w pobliżu piksela o estymowanej jasności. Wartości wag zależą nie tylko od odległości od estymowanego piksela, ale też od różnicy jasności między pikselami. Wpływ tych dwóch czynników można dowolnie modyfikować. Prezentowana metoda przeznaczona jest docelowo do redukcji szumów obrazów cyfrowych, jednak po przyjęciu odpowiednich wartości parametrów udaje się osiągnąć wynik zbliżony do wyniku segmentacji. Do ostatecznego wykrycia krawędzi należy w drugim etapie zastosować filtr krawędziujący np. filtr Laplace'a. Dobór parametrów redukcji szumu pozwala kontrolować stopień szczegółowości otrzymywanych konturów. Do weryfikacji metody wykorzystano dwa fragmenty zdjęć lotniczych wykonanych nad obszarami miejskimi. Dokonano porównania proponowanej metody z wynikami działania algorytmu Cannyego-Derichea. Otrzymywane krawędzie są bardziej wygładzone i pozbawione drobnych przerw. Mniej jest również mało istotnych, niepożądanych krawędzi, wykrycia, których nie udało się uniknąć stosując algorytm Cannyego-Derichea.

### 1. WSTĘP

Detekcja krawędzi stanowi ważny etap w wielu procedurach automatycznego pozyskiwania informacji z obrazów cyfrowych. Niestety opracowanie algorytmów, które byłyby w stanie w dużym stopniu zastąpić człowieka jest niezwykle trudne. Człowiek dokonujący ekstrakcji krawędzi w sposób manualny, np. na drodze wektoryzacji, ma możliwość dużo lepszego ich pozyskania niż uczyniłby to algorytm programu komputerowego. Dzieje się tak, dlatego, że dla człowieka zdjęcie stanowi obraz świata rzeczywistego i może on dokonywać interpretacji przebiegu krawędzi wiedząc, jakie

elementy zagospodarowania czy pokrycia terenu przedstawiają konkretne fragmenty obrazu. Program komputerowy w odróżnieniu od człowieka dysponuje jedynie tablicą niosącą informację o rozkładzie jasności na obrazie cyfrowym. Detekcja krawędzi opiera się na badaniu różnic jasności sąsiednich pikseli czy też wyodrębnianiu obszarów o podobnej intensywności. Przykładem może być wykorzystanie filtra LoG, DoG (Gonzalez, Woods, 2008) lub algorytmu Canny'ego-Deriche'a (Deriche, 1987). Obecnie rozwijane są bardziej złożone algorytmy mające na celu detekcje nie tyle samych krawędzi, lecz obiektów konkretnego typu, jak np. dróg. Wykorzystują one, poza obrazem cyfrowym, również dodatkową informację, np. dane o ukształtowaniu terenu (Hinz, Baumgartner, 2003).

Analizując efekty działania algorytmu wykrywającego krawędzie bierzemy pod uwagę pewne cechy, które są pożądane z punktu widzenia dalszego przetwarzania uzyskanego obrazu konturów. W większości przypadków zależy nam na braku obecności mało istotnych krawędzi, lub takich, których wykrycie zawdzięczamy obecności na obrazie szumu. Przykładem może być wykrywanie krawędzi samochodów odfotografowanych na zdjęciu lotniczym w sytuacji, gdy zależy nam na ekstrakcji ulic. Od wykrytych krawędzi oczekujemy braku przerw i w pewnym stopniu zgeneralizowanego przebiegu. Możemy to osiągnąć jeżeli na obrazie wejściowym zredukujemy szum i pozbowimy go mało istotnych szczegółów. Stosowana metoda redukcji szumu powinna uwzględniać występujące na obrazie naturalne różnice jasności, przez co umożliwi uniknięcie rozmycia krawędzi. Okazuje się, że opracowany przez J. Polzehla i V. Spokoinego (Polzehl, Spokoiny, 2000; Polzehl Spokoiny, 2006; Polzehl, Tabelow, 2007) algorytm redukcji szumu obrazu cyfrowego z wykorzystaniem adaptacyjnych wag (adaptive weights smoothing- AWS), po przyjęciu odpowiednich, wysokich parametrów odpowiadających za siłę wygładzenia, pozwala uzyskać obraz, który stanowi punkt wyjściowy do prostej detekcji krawędzi.

## 2. ADAPTACYJNE WYGŁADZANIE OBRAZU Z WYKORZYSTANIEM ALGORTMU AWS

### 2.1. Przebieg algorytmu

Metoda AWS należy do statystycznych metod przetwarzania obrazów cyfrowych. U jej podstaw leży teoria lokalnej estymacji metodą największej wiarygodności (MNW) w modelach nieparametrycznych. Aspekty teoretyczne metody zostały szczegółowo opisane w publikacji jej pomysłodawców (Polzehl, Spokoiny, 2006), a w niniejszym rozdziale zostanie zaprezentowana w sposób zwięzły jedynie sama procedura obliczeniowa.

Zakładamy, że dysponujemy zbiorem danych – pikseli. Piksel „i-ty” oznaczmy poprzez  $X_i$  a jego jasność poprzez  $Y_i$ . Zbiór danych ma zatem postać par  $(X_i, Y_i)$ . Naszym zadaniem jest estymacja funkcji jasności obrazu w każdym jego punkcie  $X_i$ , zapewniająca prawidłowe wychwycenie nieciągłości, będących efektem odfotografowania na obrazie krawędzi. Algorytm realizowany jest iteracyjnie. Numer iteracji oznaczmy symbolem  $k$ . Pierwszym krokiem jest inicjalizacja ( $k = 0$ ). Dla każdego z pikseli  $X_i$  zostaje obliczony zestaw wag, który przypisany jest pikselom  $X_j$  leżącym w pewnym sąsiedztwie  $U_i$  piksela  $X_i$ . Mamy więc tyle zestawów wag, ile jest pikseli, a każdy z nich jest tak liczny, jak liczne jest sąsiedztwo piksela  $X_i$ . Wielkość

sąsiedztwa na każdym z etapów iteracji definiowana jest w oparciu o graniczną odległość euklidesową  $h^{(k)}$  od danego piksela  $X_i$ . Na etapie inicjalizacji przyjmujemy  $h^{(0)} = 1$ . Obliczamy początkowe zestawy wag.

$$w_{ij}^{(0)} = K_{loc} \left( \left| \frac{\rho(X_i, X_j)}{h^{(0)}} \right|^2 \right) = K_{loc} \rho^2(X_i, X_j) \quad (1)$$

$K_{loc}$  jest niemalejącą funkcją wagującą, określoną na przedziale  $[0,1]$  a  $\rho$  odległością euklidesową pikseli  $X_i$  i  $X_j$ . Estymatory funkcji jasności w każdym punkcie  $X_i$  obliczamy z prostego wzoru:

$$\hat{\theta}_i^{(k)} = \frac{S_i^{(k)}}{N_i^{(k)}}, \quad (2)$$

gdzie:

$$S_i^{(k)} = \sum_j w_{ij}^{(k)} Y_j, \quad N_i^{(k)} = \sum_j w_{ij}^{(k)} \quad (3)$$

Na koniec etapu inicjalizacji, jak również na koniec każdego etapu iteracji, obliczamy nowy promień sąsiedztwa stosując ogólny wzór:

$$h^{(k)} = ah^{(k-1)} \quad (4)$$

Pomysłodawcy metody sugerują przyjęcie wartości  $a$  równej  $1.25^{1/2}$ .

Drugim etapem procedury obliczeniowej, obecnym na każdym z etapów iteracji ( $k > 0$ ), jest adaptacja wag. Dla każdego z pikseli  $X_j$  leżących w sąsiedztwie piksela  $X_i$  obliczamy wartości kar, lokacyjnej ( $l_{ij}$ ) oraz statystycznej ( $s_{ij}$ ), które dane są wzorami:

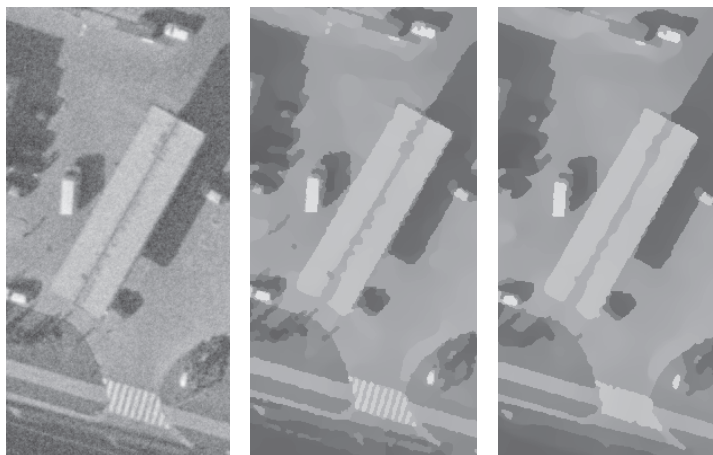
$$l_{ij}^{(k)} = \left| \frac{\rho(X_i, X_j)}{h^{(k)}} \right|^2, \quad s_{ij}^{(k)} = \lambda^{-1} T_{ij}^{(k)} \quad (5)$$

$T$  jest statystyką będącą miarą różnicy estymowanych w iteracji  $k-1$  jasności  $\theta$  dla pikseli  $X_i$  i  $X_j$  (Polzehl, Spokoiny, 2006). Gdy  $T$  rośnie, rośnie również kara  $s$ . Jak wynika ze wzorów (5) i (6), parametr  $\lambda$  pozwala na adaptację wag w zależności od różnic estymowanych jasności. Wagi w iteracji  $k$  obliczamy następująco:

$$w_{ij}^{(k)} = K_{loc}(l_{ij}^{(k)}) K_{st}(s_{ij}^{(k)}) \quad (6)$$

$K_{st}$  to niemalejąca funkcja wagująca.

Trzeci etap algorytmu, również wykonywany podczas każdej iteracji, to lokalna estymacja. Polega ona na obliczeniu estymatorów według równań (2) oraz (3). Czwarty etap, czyli zakończenie, ma miejsce wtedy, gdy spełniony jest warunek  $h^{(k)} \geq h_{max}$ . Wartość  $h_{max}$  jest kolejnym z parametrów procedury i definiuje maksymalny rozmiar sąsiedztwa obejmującego piksele  $X_j$ , którym zostaną przypisane wagi.



Rys. 1. Od lewej: fragment zdjęcia lotniczego w skali 1:10 000, dalej obrazy po przetworzeniu algorytmem AWS przy parametrach  $h_{max} = 20$ ,  $\lambda = 4$  i  $h_{max} = 20$ ,  $\lambda = 8$

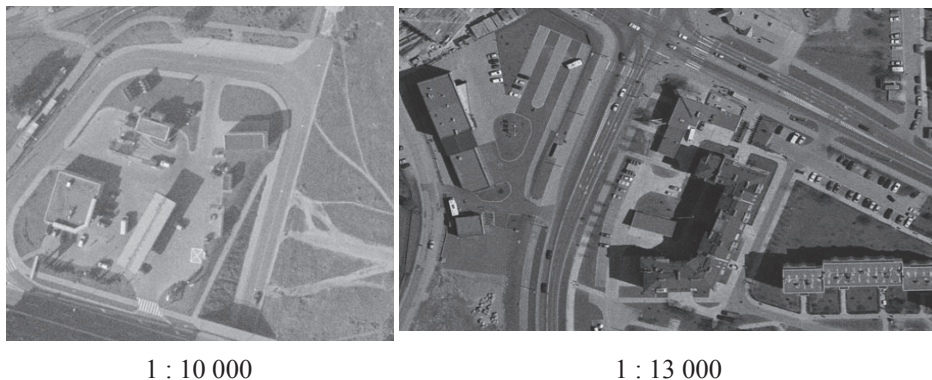
## 2.2. Dobór parametrów algorytmu AWS pod kątem detekcji krawędzi

Algorytm AWS posiada dwa parametry,  $h_{max}$  oraz  $\lambda$ . Wielkość  $\lambda$  odpowiada za siłę wygładzenia. Z licznych prób redukcji szumu na obrazach cyfrowych wynika, że najlepszy efekt wygładzenia uzyskuje się przyjmując  $\lambda$  w granicach przedziału [2,4]. Niższe wartości dają efekt losowej segmentacji, natomiast wyższe mogą skutkować utratą istotnych szczegółów. Najczęściej zależy nam na uzyskaniu konturów o zgeneralizowanym przebiegu, stąd powinniśmy wybierać wysokie wartości parametru  $\lambda$ . Jeżeli zależy nam na uniknięciu krawędziowania drobnych obiektów, powinniśmy przyjąć jeszcze wyższe wartości  $\lambda$ . Równocześnie, wartość parametru  $h_{max}$  powinna być wysoka, tak by wartość funkcji jasności estymować z jak największego sąsiedztwa obejmującego jednolity fragment obrazu. Przykłady obrazów przetworzonych za pomocą algorytmu AWS przedstawiono na rysunku 1. W obu przypadkach udało się uzyskać efekt zbliżony do efektu segmentacji. Po przyjęciu  $\lambda = 8$ , doprowadzono do usunięcia z obrazu drobniejszych szczegółów (pasy na przejściu dla pieszych). W celu przejścia z obrazów przetworzonych algorytmem AWS do obrazów krawędzi można zastosować filtr Laplace'a, a następnie progowanie.

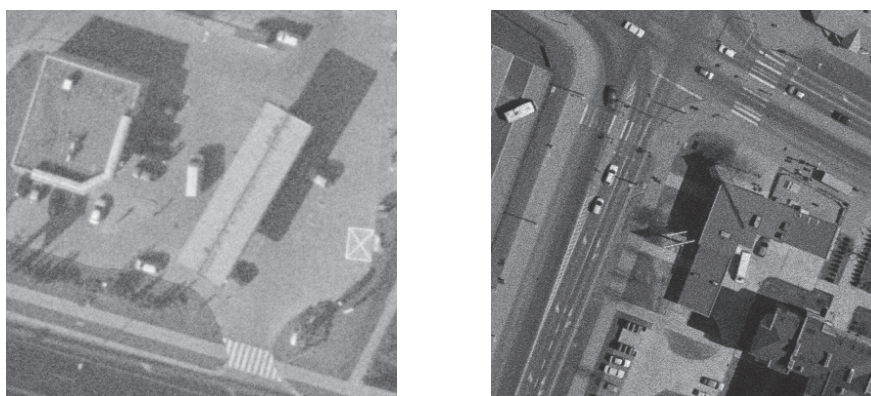
## 3. EKSPERYMENTY NUMERYCZNE

### 3.1. Dane testowe

W celu oceny jakości wykrytych krawędzi, sprawdzono działanie algorytmu na przykładzie dwóch fragmentów zdjęć lotniczych w skali 1:10 000 oraz 1:13 000 (Rys.2). Zdjęcia obejmują swoim zasięgiem obszary miejskie i zawierają wiele rejonów o jednolitej jasności (ulice, dachy budynków). Posiadają też łatwo identyfikowalne krawędzie, ale jednocześnie cechują się wysokim stosunkiem szumu do sygnału. Powiększone fragmenty obrazów pokazano na rysunku 3.



Rys. 29. Obrazy wykorzystywane do testów



Rys. 3. Powiększone fragmenty obrazów z rysunku 2

### 3.2. Algorytm Canny'ego-Deriche'a jako metoda porównawcza

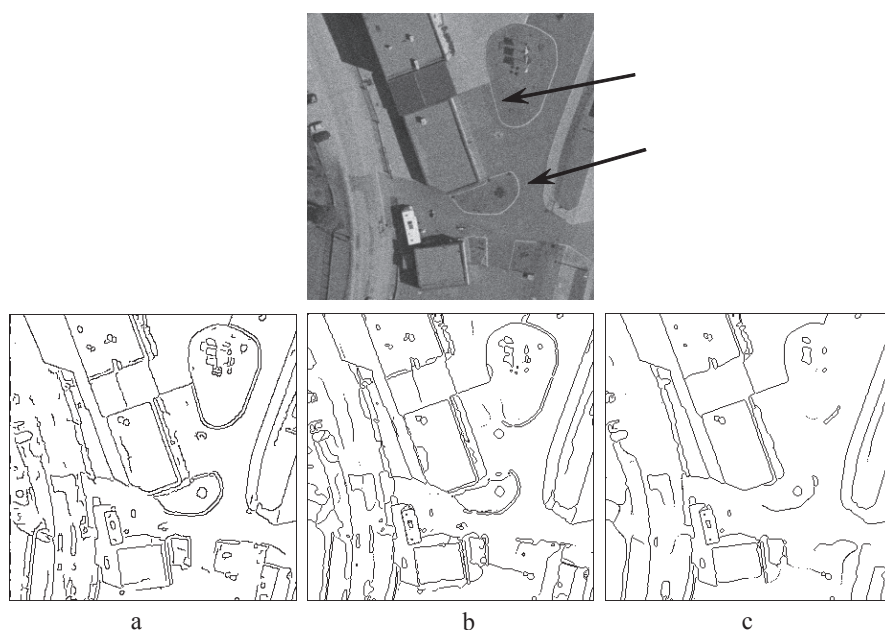
W celu oceny działania algorytmu opartego na redukcji szumów metodą AWS, przeprowadzono porównanie otrzymanych krawędzi z wynikami działania algorytmu Canny'ego-Deriche'a. Filtr Canny'ego-Deriche'a jest filtrem służącym do wykrywania krawędzi na obrazie zaszumionym. Założenia tego algorytmu przedstawił Canny (Canny, 1986) stosując filtr o ograniczonej dziedzinie. Deriche (Deriche, 1987) zmodyfikował propozycję Canny'ego stosując filtr o nieograniczonej dziedzinie.

### 3.3. Wykorzystane oprogramowanie

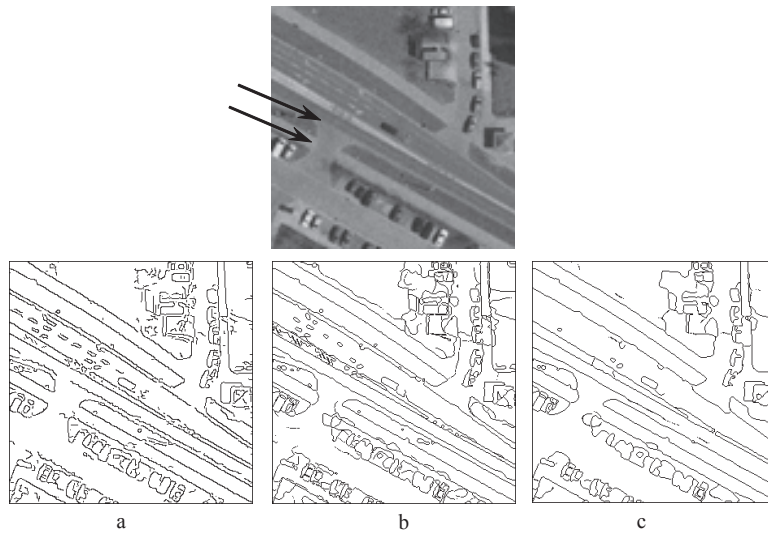
Redukcja szumów na obrazach cyfrowych algorytmem AWS wykonano wykorzystując bibliotekę *adimpro* języka R (Polzehl, Tabelow, 2007; CRAN). Do wykrywania krawędzi filtrem Canny'ego-Deriche'a wykorzystano oprogramowanie przygotowane w ramach Projektu QGAR udostępniane bezpłatnie w internecie (QGAR, 2006) na licencji GNU.

### 3.4. Wyniki

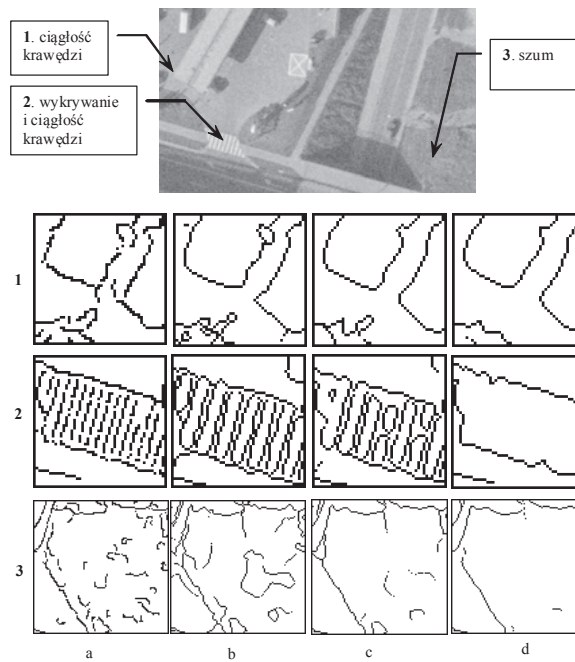
Wyniki działania proponowanego algorytmu oraz algorytmu Canny'ego-Deriche'a zaprezentowano w postaci obrazów. Rysunek 4 ukazuje miejsca, w których algorytm Canny'ego-Deriche'a wykrył krawędzie tam gdzie nie udało się ich wykryć metodą opartą na AWS. Miało to miejsce w przypadku jasnych linii o grubości 1-2 pikseli. Natomiast rysunek 5 pokazuje przewagę algorytmu AWS w miejscach, gdzie różnica jasnością po przeciwnych stronach krawędzi jest niewielka, a także ukazuje wyraźnie lepszą ciągłość wykrytych krawędzi. W obu przypadkach udało się otrzymać krawędzie o jednopikselowej szerokości. Rysunek 6 uwidacznia różnice w ciągłości wykrytych krawędzi oraz eliminację drobnych i mało istotnych konturów, które w przypadku metody Canny'ego-Deriche'a zawdzięczamy nie tylko rosnącym na skarpie roślinom, lecz także silnemu zaszumieniu obrazu. Takie zachowanie algorytmu opartego na AWS, wynika ze skutecznej redukcji szumu. Przykład przejścia dla pieszych uwidacznia wpływ parametru  $\lambda$  na stopień szczegółowości konturów. Dla  $\lambda = 8$  całe przejście dla pieszych zostało obwiedzione krawędzią, podczas gdy dla  $\lambda = 4$  wyróżnione zostały wszystkie pasy. Taki efekt wynika wprost z rezultatu wygładzenia, przedstawionego na rysunku 1. Dla  $\lambda = 4$  wykrycie krawędzi wszystkich pasów pociąga za sobą wykrycie nieistotnych krawędzi w rejonie nr 3. Dla  $\lambda = 6$  wykryto nieistniejące kontury, jak np. te w poprzek dwóch pasów.



Rys. 4. Fragment obrazu testowego w skali 1:13 000 oraz przykład skuteczniejszej detekcji metodą Canny'ego-Deriche'a. a-krawędzie uzyskane algorytmem Canny'ego-Deriche'a, b- metoda oparta na AWS dla  $h_{max}=20$ ,  $\lambda = 4$ , c- metoda oparta na AWS dla  $h_{max}=20$ ,  $\lambda = 8$



Rys. 510. Fragment obrazu testowego w skali 1:13 000 oraz przykład skuteczniejszej detekcji proponowaną metodą. a-krawędzie uzyskane algorytmem Canny’ego-Deriche’a, b- metoda oparta na AWS dla  $h_{max}=20$ ,  $\lambda = 4$ , c- metoda oparta na AWS dla  $h_{max}=20$ ,  $\lambda=8$



Rys. 6. Ciągłość wykrytych krawędzi i szumy: a-algorytm Canny’ego-Deriche’a, b- algorytm oparty na AWS dla  $h_{max} = 20$ ,  $\lambda=4$  c- algorytm oparty na AWS dla  $h_{max} = 20$ ,  $\lambda=6$  d- algorytm oparty na AWS dla  $h_{max} = 20$ ,  $\lambda=8$

#### 4. PODSUMOWANIE

Analiza uzyskanych wyników pokazuje, że AWS jest algorytmem pozwalającym bardzo dobrze estymować jasność na zaszumionym obrazie cyfrowym oraz prawidłowo uwzględniać występujące tu naturalne nieciągłości. Po zastosowaniu parametrów dających efekt bardzo silnego wygładzenia otrzymujemy obraz, na którym wykrycie krawędzi nie stanowi już większego problemu. Zawdzięczamy to skokowej zmianie jasności w miejscu nieciągłości, dającej się łatwo wykryć prostymi operatorami krawędziującymi (Gonzalez, Woods, 2008). Podczas lokalnej estymacji jasności, algorytm AWS bierze pod uwagę szersze otoczenie danego piksela. Badana jest zmienność jasności w obrębie sąsiedztwa piksela, co pozwala na wyodrębnienie dużych obszarów, które cechują się stałym tonem. Identyfikacja krawędzi z wykorzystaniem algorytmu AWS dokonuje się, więc w większym stopniu poprzez wyodrębnianie obszarów o stałej funkcji jasności, a nie w oparciu o badanie jej różnic jak ma to miejsce w przypadku algorytmu Canny'ego-Deriche'a. Poniżej podano najważniejsze zalety analizowanej metody.

1. Otrzymywane krawędzie mają szerokość jednego piksela.
2. Nie zauważono odstępstw od faktycznego położenia konturów na zdjęciu.
3. W przeciwieństwie do algorytmu Canny'ego-Deriche'a otrzymane kontury nie posiadają przerw.
4. W przeciwieństwie do algorytmu Canny'ego-Deriche'a uzyskane krawędzie mają przebieg bardziej zgeneralizowany. Nie zauważono tak licznych zmian kierunku przebiegu krawędzi wynikających z zaszumienia obrazu.
5. Wykrycie krawędzi możliwe jest nawet przy bardzo silnym zaszumieniu i stosunkowo niedużym skoku jasności.
6. Uzyskano całkowitą eliminację krawędzi, które są wynikiem obecności szumu. Nie można tego powiedzieć o krawędziach wykrytych algorytmem Canny'ego-Deriche'a.
7. Procedura wykorzystująca AWS pozwala na proste kontrolowanie szczególności wykrywanych krawędzi poprzez zmianę parametru  $\lambda$ . Daje to możliwość eliminacji krawędzi niechcianych jak np. konturów krzewów na trawniku. Osiągnięcie takiego efektu przy użyciu algorytmu Canny'ego-Deriche'a jest niemożliwe.
8. Plusem jest obecność tylko dwóch parametrów wejściowych procedury.

Natura estymacji jasności poprzez AWS niesie ze sobą niestety pewne mankamenty. W niektórych przypadkach wpływają one negatywnie na obraz wykrytych krawędzi.

1. Przy większych wartościach parametru  $\lambda$ , koniecznych do uzyskania efektu silnego wygładzenia, możliwe jest zatrącenie drobnych obiektów. Efekty takie pokazano na rysunku 4.
2. Zmniejszenie wartości parametru  $\lambda$  prowadzi do osłabienia tego niepożądanego efektu, ale jednocześnie powoduje wzrost ilości konturów mało istotnych lub będących wynikiem szumu, jak w przypadku przejścia dla pieszych 6.
3. Dokonanie wyboru odpowiedniej wartości parametru  $\lambda$  wymaga przeprowadzenia kilku prób. Jest to jednak cecha zaawansowanych metod detekcji konturów, także algorytmu Canny'ego-Deriche'a.



4. Przy niektórych wartościach  $\lambda$  może dojść do wykrycia nieistniejących krawędzi jak w przypadku przejścia dla pieszych (Rys. 6c), gdzie obecne są krawędzie biegnące w poprzek dwóch pasów.
5. Procedura AWS realizowana jest iteracyjnie, wykonywana jest dla każdego piksela i wymaga każdorazowego obliczania wag w obrębie dużego sąsiedztwa. Skutkuje to dużą czasochłonnością, co dyskwalifikuje wykorzystanie metody tam, gdzie liczy się szybkość uzyskania wyniku.

Do ostatniego etapu wykrywania krawędzi opisywaną metodą, używano prostego filtra krawędziującego i zwykłego progowania. Zastąpienie zwykłego progowania, progowaniem z histerezą, tak jak w algorytmie Canny'ego-Deriche'a może być jedną z dróg wzmocnienia analizowanego algorytmu.

Detekcja krawędzi metodami automatycznymi nawet na stosunkowo nieskomplikowanych obrazach niesie ze sobą ryzyko wystąpienia błędów. Perspektywa pełnej automatyzacji tego procesu wydaje się wciąż odległa mimo oczywistego postępu na polu przetwarzania obrazów cyfrowych. Interpretacja obrazu przez człowieka jest bezsprzecznie najpewniejszą, choć niestety bardzo powolną metodą pozyskiwania krawędzi i wciąż stanowi punkt odniesienia dla metod automatycznych.

Artykuł jest efektem pracy wykonanej w ramach badań statutowych AGH nr: 11.11.150.949

## 5. LITERATURA

Canny J., 1986. A Computational Approach for Edge Detection. *IEEE Trans. Pattern Anal. Machine Intel*, vol. 8, no. 6, s. 679-698.

CRAN – The Comprehensive R Archive Network: <http://cran.rakana.com/>

Deriche R., 1987. Using Canny's Criteria to Derive a recursively Implemented Optimal Edge Detector. *International Journal of Computer Vision*.

Gonzalez R. C., Woods R. E., 2008. *Digital Image Processing*. PEARSON Prentice Hall, s. 706-738.

Hinz S., Baumgartner A., 2003. Automatic Extraction of Urban Road Networks from Multi-View Aerial Imagery. *International Journal of Photogrammetry and Remote Sensing*. 58/1-2, s. 83 -98.

Polzehl J., Spokoiny V. G., 2000. Adaptive weights smoothing with applications to image restoration. *Journal of the Royal Statistical Society*, B 62, Part 2, s. 335-354

Polzehl J., Spokoiny V. G., 2006. Propagation-Separation Approach for Local Likelihood Estimation. *Probability Theory and Related Fields*, 135, s. 335-362.

Polzehl J., Tabelow K., 2007. Adaptive Smoothing of Digital Images: The R Package *adimpro*. *Journal of Statistical Software*, vol. 19.

QGAR 2006: <http://www.qgar.org/doc/index.html>

## **EDGE DETECTION USING ADAPTIVE SMOOTHING OF DIGITAL IMAGES**

**KEY WORDS:** digital image processing, edge detection, adaptive approaches, local likelihood estimation

### **Summary**

Edge detection is often a fundamental stage of digital image processing in automatic measurement techniques. A number of approaches for edge detection, such as LoG (Laplacian of Gaussian) filtering and Canny-Deriche algorithm, involve using edge-extracting filters. In this paper we present a new edge detection technique. Our approach is based on statistics, specifically on the propagation-separation approach for local likelihood estimation, which was developed in 2006 by J.Pohlzel and V.Spokoiny. This new approach for local estimation involves adaptive determination of a pixel's neighbourhood, used for estimation of pixel's intensity. This neighbourhood is determined by a set of weights assigned to pixels in the vicinity of the point of estimation. The value of each weight depends not only on the Euclidean distance between the pixels, but also on a differences in the intensity. The influence of these two factors could be modified by changing the parameters of the procedure. The method, as described briefly here, has been mainly designed for image restoration; however, by using a special set of parameters an effect, similar to segmentation, can be achieved. To obtain the final edge image, it is sufficient to use simply one of the edge extracting filters, for example the Laplacian one. The procedure parameters allow to influence sensitivity of the edge detection. The edge detection results were tested on two fragments of frame images of a city. The edges detected were compared with results of the Canny-Deriche algorithm. The edges obtained were smoother and did not show numerous small breaks. In addition, short, less important edges were less likely to appear. These effects were impossible to avoid using the Canny-Deriche approach.

mgr inż. Jakub Kolecki  
e-mail: kolecki@agh.edu.pl  
tel. 012 617 38 26

dr inż. Andrzej Wróbel  
e-mail: awrobel@agh.edu.pl  
tel. 012 617 38 26