



Poznań, 21-23 czerwca 2017

WYZNACZANIE KONTURÓW OBSZARÓW POKRYCIA SYGNAŁEM Z WYKORZYSTANIEM BIBLIOTEKI PRZETWARZANIA OBRAZÓW OPENCV

SIGNAL COVERAGE AREA CONTOUR GENERATION WITH OPENCV LIBRARY

Streszczenie: Znajomość konturów obszarów pokrycia sygnałem użytecznym jak i zakłócającym, odgrywa istotną rolę w procesie planowania i optymalizacji zasięgu sieci radiowych oraz w gospodarce widmem elektromagnetycznym. O ile wizualizacja zasięgów jest zadaniem prostym oraz szybkim, to znajomość wartości numerycznych jest zagadnieniem czasochłonnym i nie trywialnym. Znaczne przyspieszenie obliczeń można uzyskać wykorzystując funkcje biblioteki OpenCV, stosowanej w procesie przetwarzania obrazów.

Abstract: Knowledge of the coverage areas borders of the usable and interfering signals, plays an important role in planning and optimizing radio networks, and in the electromagnetic spectrum engineering. As far as scope visualization is simple and fast, knowledge of the numerical values is a time consuming and not trivial task. Significant acceleration of the calculation can be obtained using the OpenCV library functions, used in the processing of the images.

Słowa kluczowe: Obszar zasięgu sygnału, generowanie konturów izolinii, OpenCV.

Keywords: Signal coverage area, isoline contour generation, OpenCV.

1. WSTĘP

W procesie planowania i optymalizacji sieci radiowych znajomość konturów obszaru pokrycia sygnałem użytecznym odgrywa istotną rolę przy wyznaczaniu procentowego pokrycia planowanego obszaru, czy w wyznaczaniu procentowego pokrycia ludnościowego na danym obszarze. Znajomość obszaru pokrycia sygnałem użytecznym, jak i zakłócającym, wymagana jest również w analizach kompatybilności elektromagnetycznej systemów w procesie zarządzania widmem elektromagnetycznym. Zagadnienie wyznaczania konturów linii stałej wartości (izolinii) jest dobrze znane^{[1][4][5][6][7]}. Problem polega na czasie ich wykonania. Jeżeli w procesie planowania i optymalizacji sieci radiowych wykonywanych jest wiele iteracji, to czas wykonania każdej iteracji jest elementem składowym całkowitej sumy obliczeń. W celu przyspieszenia procesu generowania konturów obszaru zasięgu sieci, postanowiono wykorzystać funk-

cje biblioteki OpenCV, stosowanej w procesie przetwarzania obrazów, związane z rozpoznawaniem konturów.

2. BIBLIOTEKA OpenCV

Biblioteka OpenCV^[2], oparta na kodzie otwartym, zapoczątkowana przez firmę Intel, jest biblioteką wieloplatformową. Można z niej korzystać zarówno na platformie Linux, Mac OS X, Windows, iOS oraz Android. Posiada interfejsy dla języków programowania C, C++, Java czy Python.

Istnieje również wieloplatformowa wersja biblioteki „opakowującej” (wrapper) OpenCV - Emgu CV^[3], pozwalająca na wykorzystanie biblioteki OpenCV na platformie Microsoft .NET. Biblioteka Emgu CV kompilowana jest w środowisku programistycznym Visual Studio, Xamarin Studio czy Unity. Dostępne w niej funkcje wywoływane są bezpośrednio w językach kompatybilnych z platformą .NET (C#, VB, VC++, IronPython, ...), a powstałe oprogramowanie uruchamiane jest w systemach Windows, Linux, Mac OS X, iOS, Android oraz Windows Phone.

Biblioteki OpenCV i Emgu CV dostępne są zarówno w wersji otwartej (licencja GPL), jak i w wersji komercyjnej, w której do pracy wielowątkowej, wykorzystywane są pakiety Intel TBB i IPP. Biblioteki wykorzystują zarówno procesor główny (CPU) jak i procesory graficzne, z wykorzystaniem standardów CUDA i OpenCL.

3. PROBLEM

Obliczenia rozkładu natężenia pola, lub innego parametru pochodnego (np. prawdopodobieństwa wystąpienia sygnału o poziomie, na podstawie wyznaczonych wartości, dla znanej wartości medianowej, odchylenia standardowego i określonego procent pokrycia obszaru), prowadzone są na obszarze prostokątnym, z określoną rozdzielczością poziomą i pionową. Natężenia pola wyznaczane jest w punktach siatki - wynikiem obliczeń jest prostokątna macierz o N wierszach i M kolumnach, której elementy a_{ij} zawierają obliczone wartości. Interesuje nas wyznaczenie konturu obszaru dla stałej wartości poszukiwanego parametru.

4. ALGORYTM

Dla potrzeb generowania konturów z treści obrazu biblioteka OpenCV oferuje funkcję `FindContours`, która wyznacza kontury z obrazu binarnego. Obraz binarny dany jest w postaci pojedynczego kanału 8 bitowego. Niezerowe piksele traktowane są, jako wartości równe jedności (1), zerowe pozostają zerami. Obraz w takiej postaci otrzymujemy, poddając standardowy obraz kolorowy konwersji do obrazu w skali szarości, korzystając z klasy `Image<Gray,byte>` biblioteki, dla której parametrem wejściowym konstruktora jest kolorowa mapa bitowa. Z klasy tej wywoływana jest funkcja `ThresholdBinary`, której argumentem jest wartość poziomu szarości, dla której następuje detekcja krawędzi konturu.

Wygenerowane kontury, dla których wartości współrzędnych punktów wyrażone są we współrzędnych obrazka, można w prosty sposób konwertować do układu współrzędnych wymaganego w dalszych obliczeniach.

W ogólnej postaci algorytm ma postać:

1. Generuj monochromatyczny („szary”) obraz, którego „szare” prostokąty odpowiadają punktom siatki, dla których spełnione jest wymagane kryterium,
2. Wywołaj funkcję `FindContours` pakietu OpenCV, w celu wyznaczenia konturów obszaru,
3. Konwertuj współrzędne punktów konturów otrzymanych w kroku 2, z układu graficznego do wymaganego układu odniesienia.

Zaletą funkcji `FindContours` jest to, że zwracany jest wskaźnik do konturów, a pierwszy kontur jest konturem najbardziej zewnętrznym (kontur główny). Pozostałe kontury stanowią brzegi obszarów rozdzielnych z konturem głównym, lub są obszarami wewnętrznymi (tzw. „dziurami”) w konturze głównym lub innych konturach rozdzielnych. W parametrze, `hierarchy`, funkcji `FindContours`, zwracana jest topologia rozpoznanych w treści obrazu konturów. Dla k -tego konturu element $[0,k,0]$ parametru `hierarchy` jest indeksem do kolejnego konturu, zaś element $[0,k,1]$ indeksem do poprzedniego konturu na tym samym poziomie hierarchii. Element $[0,k,2]$ jest indeksem konturu wewnętrznego („dziury”), natomiast element $[0,k,3]$ to indeks konturu rodzica danego konturu. Dla każdego z powyższych elementów wartość równa -1 wskazuje brak indeksu. Na podstawie tych parametrów, możliwe jest stworzenie odpowiedniej struktury danych opisującej obszar zasięgu.

Z uwagi na to, że algorytm detekcji krawędzi wykrywa obecność krawędzi po przekroczeniu zadanej wartości (ang. *threshold*), należy tak ustawić rozmiar obszaru odpowiadającemu pojedynczemu elementowi siatki, aby detekcja była poprawna. W wyniku przeprowadzonych testów ustalono, że minimalny obszar odpowiadający punktowi danych siatki, umożliwiający poprawną detekcję krawędzi, to obszar o rozmiarze 3×3 pikseli. Dla siatki $N \times M$ danych, rozmiar obrazu wynosi

$3N \times 3M$ pikseli (lub bajtów, gdyż obraz jest monochromatyczny).

Dodatkowo, generowany obraz powinien być odwrotnością obrazu pokrycia obszaru – w przeciwnym wypadku, jako obszar najbardziej zewnętrzny otrzymamy obszar, którego kontur jest krawędzią obszaru.

5. IMPLEMENTACJA

Implementację algorytmu wykonano z wykorzystaniem języka C# na 64-bitowej platformie Windows 10 w środowisku programistycznym Visual Studio 2015 Update 3. Wykorzystano bibliotekę Emgu CV w wersji 3.1.0.

W implementacji zdefiniowano dodatkową klasę, `GrayImage`, w której zapisywane są obszary odpowiadające punktom siatki, dla których spełnione jest wymagane kryterium. Funkcja `GetStream` tej klasy zwraca strumień danych, będących treścią monochromatycznego obrazu, który stanowi parametr wejściowy konstruktora klasy `Image<Gray,byte>` biblioteki Emgu CV.

W celu przyspieszenia czasu generowania obrazu monochromatycznego, wykonywanego w pętli podwójnej (dla każdego wiersza, dla każdej kolumny), proces ten odbywa się z wykorzystaniem obliczeń równoległych dla pętli zewnętrznej.

W następnym kroku wywołana zostaje funkcja `CvInvoke.FindContours` biblioteki Emgu CV, która wywołuje funkcję `FindContours` biblioteki OpenCV, generującą kontury.

Na podstawie topologii obrazu, generowane są kontury obszarów rozdzielnych zawierające „dziury”. Kontury te konwertowane są z układu współrzędnych obrazu, do układu geograficznego (długość i szerokość geograficzna). Kontury te wykorzystywane są do wizualizacji wyników analizy, lub do dalszych obliczeń - przykładowo, wyznaczania procentowego pokrycia obszaru objętego usługą (np. obszar województwa, reprezentowany przez obszary gmin), bądź też procentowego pokrycia ludnościowego tego obszaru.

Rejestrowany jest czas wykonania funkcji generowania konturów obszaru do porównania z czasami innych algorytmów.

6. WYNIKI

W referacie przedstawiono wyniki obliczeń przeprowadzonych dla dwóch hipotetycznych sieci jednocześnie:

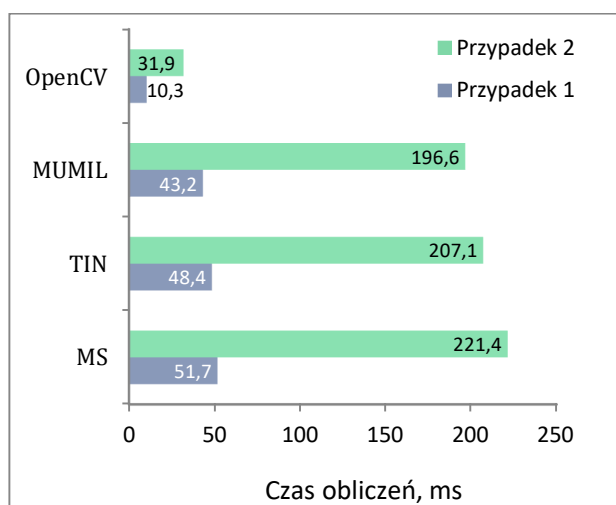
- lokalnej DVB-T2 o przepływności rzędu 40 Mb/s,
- ogólnokrajowej DVB-T o przepływności rzędu 24 Mb/s.

W obu przypadkach obliczenia przeprowadzono dla trybu odbioru stałego, z zastosowaniem kierunkowej anteny odbiorczej, o charakterystyce poziomej zgodnej z ITU-R BT.419, dla 95% prawdopodobieństwa pokrycia obszaru sygnałem.

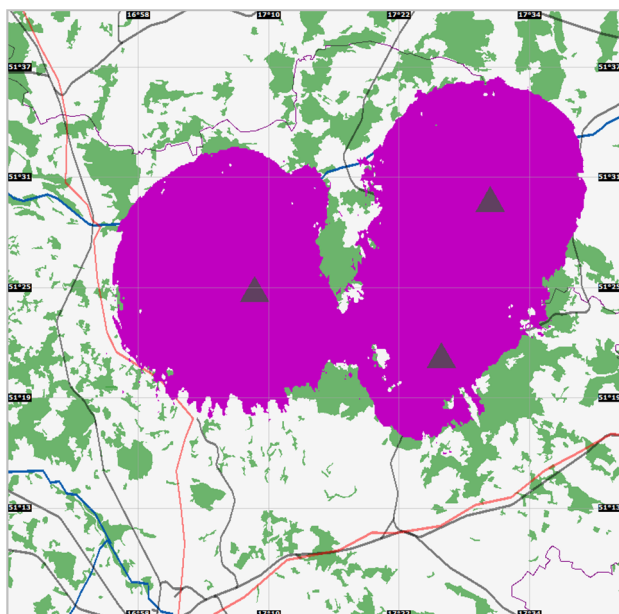
W Tabeli 1 zamieszczono czasy wyznaczania konturów obszarów pokrycia sygnałem. Wyników w postaci wykresu przedstawiono na Rys. 1. Obszary pokrycia sygnałem dla analizowanych przypadków przedstawiono odpowiednio na Rys. 2 i 3.

Tab. 1. Czasy wyznaczania konturów obszaru dla różnych metod

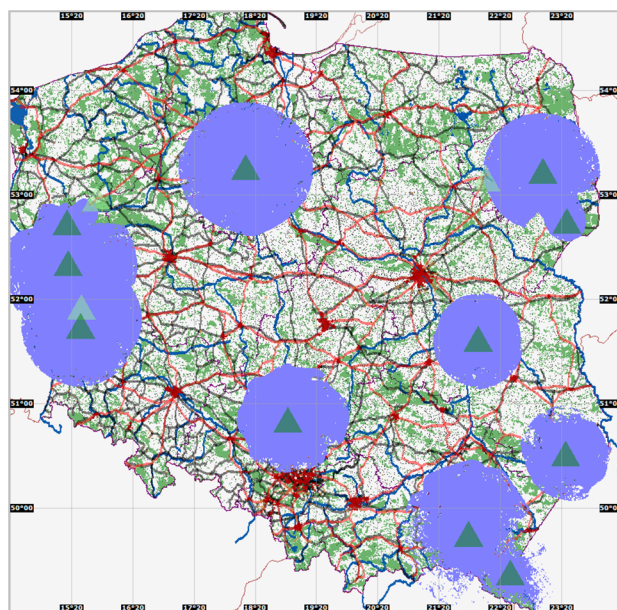
Metoda	Czas wyznaczania konturów w milisekundach	
	Przypadek 1	Przypadek 2
MS	60,0	221,4
TIN	51,7	207,1
MUMIL	43,2	196,6
OpenCV	10,3	31,9



Rys. 1. Wykres czasów obliczeń konturów obszaru



Rys. 2. Obszar sygnału użytecznego hipotetycznej jedno-częstotliwościowej sieci lokalnej DVB-T2



Rys. 3. Obszar sygnału użytecznego hipotetycznej jedno-częstotliwościowej sieci krajowej DVB-T

7. WNIOSKI

Wyniki testów czasów obliczeń metody wyznaczania konturów obszaru, z wykorzystaniem funkcji rozpoznawania konturów biblioteki OpenCV, przedstawione w referacie, w porównaniu do innych metod, wskazują jednoznacznie, że metoda ta jest kilkakrotnie szybsza od najszybszej z konwencjonalnych testowanych metod, niezależnie od stopnia złożoności obszaru.

Metoda ta została zastosowana w oprogramowaniu planowania i optymalizacji sieci radiowych używanym w Pracowni Inżynierii i Gospodarki Widma Zakładu Kompatybilności Elektromagnetycznej Instytutu Łączności, Państwowego Instytutu Badawczego.

LITERATURA

- [1] <http://paulbourke.net/papers/conrec/>
- [2] <http://opencv.org/>
- [3] <http://www.emgu.com>
- [4] Gold G. M.. 1997. „Automated contour mapping using triangular element data structures and an interpolant over each irregular triangular domain”. *ACM SIGGRAPH Computer Graphics* 11 (2) : 170–175
- [5] McLain D. H.. 1974. „Drawing contours from arbitrary data points”. *Computer Journal*. 17 (4): 318–324
- [6] Sutcliffe D. C.. 1980. „Contouring over rectangular and skewed rectangular grids-An introduction”. *Mathematical Methods in Computer Graphics and Design*. 39–62
- [7] Zhao Jingchang, Bai Runcai, Liu Guangwei, Liu Wei. 2014. „Fast isolines generation algorithm based on TIN”. *Computer Engineering and Applications*. 50 (24) : 10–15