Jerzy KOROSTIL, **Łukasz NOZDRZYKOWSKI**
WYDZIAŁ INFORMATYKI, ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY W SZCZECINIE,
ul. Żołnierska 49, 71-210 Szczecin

Propozycja algorytmu steganograficznego z kluczem rozproszonym opartego na funkcji CSF

Prof. dr hab. inż. Jerzy KOROSTIL

Pracuje na Wydziale Informatyki Zachodniopomorskiego Uniwersytetu Technologicznego od 2000r. Obecnie jest kierownikiem Zakładu Bezpieczeństwa Oprogramowania w Katedrze Inżynierii Oprogramowania. Zainteresowania naukowe dotyczą steganografii, bezpieczeństwa sieci komputerowych oraz oprogramowania

e-mail: jkorostil@wi.zut.edu.pl

Mgr inż. Łukasz NOZDRZYKOWSKI

Ukończył studia na Wydziale Informatyki Politechniki Szczecińskiej w 2006r. Obecnie jest doktorantem w Katedrze Inżynierii Oprogramowania Wydziału Informatyki ZUT. Jego zainteresowania naukowe obejmują kryptografię i steganografię oraz bezpieczeństwo sieci komputerowych.

e-mail: admin@hotfix.pl

Streszczenie

W artykule przedstawiono propozycję algorytmu steganograficznego z kluczem rozproszonym dla cyfrowych środowisk graficznych, wykorzystującym znormalizowaną funkcję czułości kontrastu CSF jako kryterium oceny niewidoczności zmian wprowadzanych w obrazie wynikowych. Jako klucz rozproszony rozumie się klucz stosowany na każdym etapie ukrywania wiadomości. Przeprowadzono analizę efektywności proponowanego rozwiązania oraz przedstawiono próbę jego optymalizacji.

Słowa kluczowe: steganografia, obrazy cyfrowe, ukrywanie informacji, CSF.

Proposition of the steganographic algorithm with dispersed key based on the CSF function

Abstract

The paper presents a proposition of the steganographic algorithm with dispersed key for digital images. This algorithm uses the normalized contrast sensitive function CSF. The CSF function is used as a measure criterion of invisibility changes in the output image. Choosing the images to hiding messages is realized according to the image characteristic measures. This parameter determines the invisibility level of changes in the output image. Some exemplary measures are: mean, variance, dispersion from colors histogram and entropy, homogenity from Co-Occurrence Matrix. A dispersed key is used at every stage of the hiding messages process. The proposed algorithm is based on the wavelet transform DWT. The key is used for selection of a container, random selection of the placement, selection of the wavelet function and wavelet coefficients. There was performed an analysis of the proposed solution efficiency. The attempt at optimizing the solution is presented in the paper. The database UCID was used for analysis of the efficiency. The proposed optimization resulted in four-time acceleration of the algorithm.

Keywords: steganography, digital images, information hiding, CSF.

1. Wprowadzenie

Pojęcie steganografii stało się popularne pod koniec lat dziewięćdziesiątych wraz z rosnącą popularnością Internetu. Algorytmy steganograiczne służą do ukrywania wiadomości w innych informacjach, stanowiących kontener dla ukrywanych danych. Przykładem kontenera dla ukrywanej informacji są cyfrowe pliki graficzne oraz dźwiękowe. Ponadto algorytm steganograficzny zapewnia niewidoczność ukrywanej wiadomości w kontenerze.

Szczególnie istotne dla steganografii są kolorowe obrazy graficzne stosowane jako kontener dla ukrywanej wiadomości. Popularność stosowania obrazów w steganografii wiąże się bezpośrednio z ich popularnością w Internecie, gdzie obok tekstu są one głównym elementem stron internetowych. Według raportów Netcraft [1] dla listopada 2010 roku, w Internecie znajduje się około 250 milionów witryn. Według statystyk światowych [2] w 2009 roku było 1,7 miliarda Internautów. W samej Polsce wg GUS [3] w 2008 roku 45.7% gospodarstw posiadało dostęp do szerokopasmowego Internetu (populacja Polski wynosiła 38,1 miliona ludzi). Biorąc pod uwagę, że WWW stanowiło w 2009 roku 52% ruchu [4] (analiza dla 110 ISP) i tendencja jest rosnąca, zatem strony internetowe stanowią dobre medium do przesyłania w obrazach na nich umieszczanych dodatkowych informacji. Szczególnie użytecznym kanałem mogą być hostingi plików graficznych, jak serwis imgur.com, gdzie godzinowo wysyłanych jest do 4 tysięcy obrazów, a średni pojemność obrazka wynosi 184.33 KB, zaś całkowity transfer sięga 9 TB (dane na listopad 2010 [5]).

2. Niewidoczność ukrywania wiadomości

Użycie algorytmu steganograficznego dla cyfrowego środowiska graficznego nie może powodować występowania widocznych zmian w obrazach wynikowych. Opracowanie metody oceny niewidoczności zmian pomiędzy obrazem źródłowym, a wynikowym pozwala na zbudowanie metody steganograficznej ukrywającej wiadomości w miejscach, gdzie zmiany pozostaną niewykrywalne.

W pracy [6] przedstawiono metodę oceny niewidoczność zmian w kolorowych obrazach cyfrowych bazującą na postrzeganiu zmian kontrastu przez ludzkie oko. Stosując szum uzyskiwano efekt jaki powoduje działanie algorytmu steganograficznego. To pozwalało na symulowanie działania dowolnego algorytmu.

Metodę wyznaczania zmian w obrazach opracowano wykorzystując znormalizowaną funkcję czułości kontrastu H(f) określaną zależnością Mannosa i Sakrisona [6]:

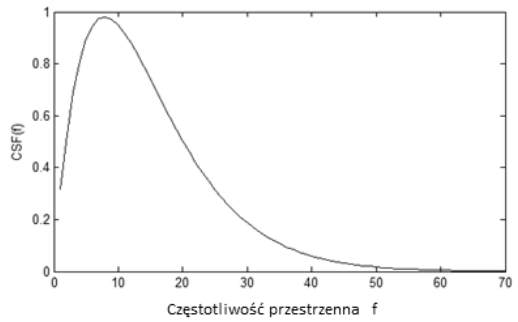
$$H(f) = 2.6*(0.0192 + 0.114*f)*e^{-(0.114*f)^{1.1}}, \qquad (1)$$

gdzie: f jest częstotliwością zmian wyrażoną w cyklach na stopień i jest utożsamiana z częstotliwością prążkową zmian przy obserwacji w jednym stopniu patrzenia.

Zastosowanie funkcji wrażliwości kontrastu pozwala wyznaczyć wrażliwość ludzkiego oka na różne częstotliwości wizualne, gdzie im wyższa częstotliwość zmian impulsów, tym gorsze jest rozpoznawanie wzorców i zmiany stają się niewidoczne.

Wykres funkcji wrażliwości kontrastu przedstawia rysunek 1.

Funkcja CSF osiąga szczytową wartość dla częstotliwości $f = 8$. Całkowicie nierozpoznawalne dla ludzkiego oka są częstotliwości leżące powyżej 60 cykli na stopień [7]. Oko ludzkie słabo rozpoznaje także zmiany kontrastu poniżej granicy 8 cykli na stopień. Odpowiada to obserwacji większych obiektów z bliska, gdzie w kącie jednego stopnia nie występuje zmiana kontrastu. Wyznaczając niewidoczność zmian należy także określić standardowe warunki obserwacji. Przyjmuje się, że odległość obserwacji wynosi 80 centymetrów od monitora.



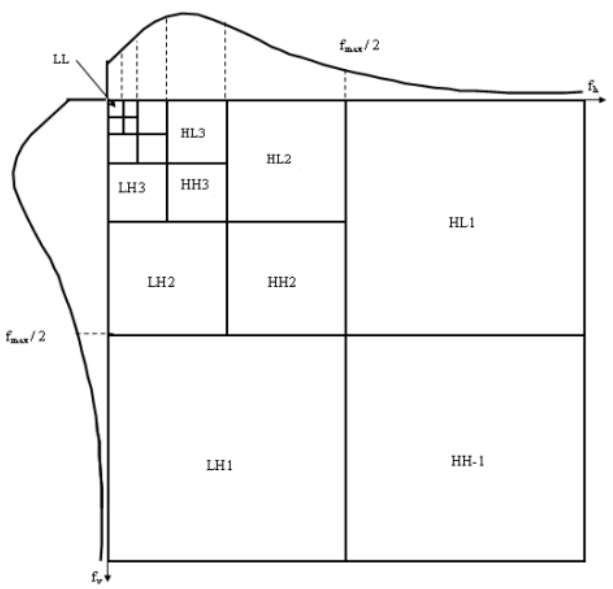
Rys. 1. Funkcja czułości kontrastu
Fig. 1. Contrast sensitive function

We wzorze (1) $f = f_n * f_s$ jest częstotliwością środkową wyrażaną w cyklach na stopień. Parametr f_n jest znormalizowaną częstotliwością przestrzenną, zaś f_s częstotliwością próbkowania wyznaczaną z zależności:

$$f_s = \frac{2v * \tan(0.5^{\circ}) * r}{0.0254}, \qquad (2)$$

We wzorze (2) parametr v określa odległość obserwacji wynoszącą około 0.8 metra od monitora, zaś r jest rozdzielczością monitora wyrażoną w pikselach na cal.

Ponieważ prezentowany w niniejszym artykule steganograficzny algorytm ukrywania informacji w obrazach wykorzystuje dziedzinę transformaty falkowej, więc zaproponowano przeniesienie obliczeń do tej dziedziny. W pracy [8] przedstawiono przeniesienie funkcji wrażliwości kontrastu do dziedziny falkowej. Przedstawia to rysunek 2. Maksymalną częstotliwość f wyznacza się zgodnie z kryterium Nyquista



Rys. 2. Funkcja CSF w odniesieniu do dekompozycji falkowej
Fig. 2. CSF function in relation to the wavelet decomposition

Zgodnie z tym, najbardziej zauważalne zmiany występują w aproksymacji niskopoziomowej, a najmniejsze na poziomie detali. Budując algorytm steganograficzny należy zadbać, aby dane były ukrywane na poziomie detali, w których zmiany będą najmniej zauważalne. Jednocześnie zauważa się, że najmniej zmiany są widoczne w dekompozycji diagonalnej.

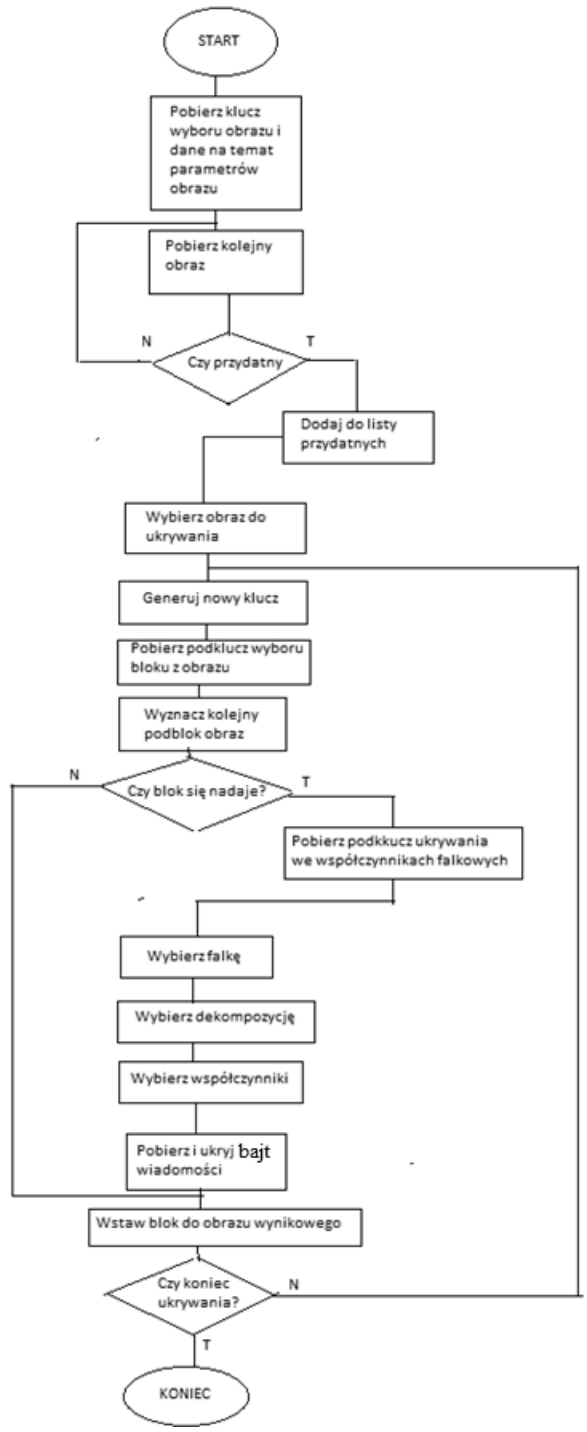
3. Proponowany algorytm steganograficzny

W artykule proponuje się realizację algorytmu steganograficznego z kluczem opartym o transformatę falkową DWT połączonym z wyborem miejsc ukrywania informacji.

Wykorzystywana jest także metoda oceny niewidoczności zmian oparta o funkcję czułości kontrastu CSF.

W proponowanym algorytmie używany jest rozproszony klucz, co oznacza, że klucz działa w wielu miejscach algorytmu na każdym etapie ukrywania wiadomości.

Ogólny schemat proponowanego algorytmu przedstawia rysunek 3.



Rys. 3. Propozycja algorytmu steganograficznego
Fig. 3. The proposed steganographic algorithm

W proponowanym algorytmie początkowo następuje wybór obrazu z bazy obrazów na podstawie fragmentu klucza. Klucz ten określa jakie parametry musi spełnić obraz, aby ukryć w nim wiadomość. Parametry te są miarami charakteryzującymi dany obraz. W jaki sposób miary te wpływają na niewidoczność zmian w obrazie wynikowym, przedstawiono w pracy [9].

W pracy [9] wykonano analizę istotności szeregu miar wyznaczanych dla kolorowych obrazów cyfrowych wyliczanych na podstawie histogramu kolorów oraz macierzy zdarzeń (Co-Occurrence Matrix) [6]. Dla histogramu kolorów były to średnia, wariancja, dyspersja, kurtoza oraz energia, zaś dla macierzy zdarzeń kontrast, entropia oraz jednorodność. Pod uwagę była brana także liczba odcieni czerwieni.

Za miękkiej redukcji atrybutów warunkowych z teorii zbiorów przybliżonych w oparciu o względne prawdopodobieństwo reguł użytecznych, wykonano analizę istotności charakterystycznych miar obrazów. To pozwoliło określić w jakim stopniu wielkości tych miar wpływają na ukrywanie danych z zadanym poziomem niewidoczności.

W bazie przechowuje się informacje o miarach charakterystycznych dla wszystkich podbloków obrazu o rozmiarze kąta patrzenia jednego stopnia. Dla obserwacji z odległości 0.8 metra i monitora 19 cali, rozmiar takiego bloku wyniósłby 47x47 pikseli. Obraz powinien zostać podzielony na takie bloki, a w bazie przechowywana by była informacja o wszystkich miarach charakterystycznych dla każdego bloku. Poprzez informacje o wpływie tych miar na niewidoczność zmian określa się dalej szacunkową liczbę bloków, w których ukrywana wiadomość pozostanie niewidoczna. Dane te są trzymane w sposób zakodowany w bazie, aby nie było konieczności ciągłego obliczania parametrów w procesie ukrywania kolejnych wiadomości.

Ponieważ w bazie obrazów, wiele z obrazów może spełnić wymogi niewidoczności, a zatem kolejna część klucza określa dokładnie numer obrazu użytego jako kontener dla ukrywanej informacji.

Następnie na wybranym obrazie wykonywane jest ukrywanie wiadomości. Także tutaj obraz jest dzielony na bloki odpowiadające jednemu stopniowi patrzenia, który jest wymagany przy użyciu funkcji czułości kontrastu CSF. Jeżeli poziom zmian w bloku znajdzie się powyżej progu rozpoznawania zmian kontrastu człowieka, to blok taki jest klasyfikowany jako przydatny do ukrycia w nim wiadomości. Wszystkie bloki tworzą razem mapę pozwoleń, czyli miejsc, gdzie będzie ukrywana wiadomość.

Mapa pozwoleń tworzy graf, po którym przejście odbywa się w kolejności określonej przez algorytm pseudolosowy z kluczem. Na tym etapie należy zapewnić to, że po wprowadzeniu wiadomości, mapa pozwoleń pozostanie niezmieniona i będzie można ją wygenerować podczas odczytywania wiadomości.

Dane są ukrywane w kolejnych blokach losowanych przez algorytm wyboru. Początkowo blok taki jest dekomponowany przy użyciu pojedynczej dekompozycji falkowej. Rodzaj zastosowanej funkcji falkowej zależny jest od kolejnej części klucza. W wyniku zastosowanej dekompozycji otrzymuje się cztery podobrazy: jeden aproksymacji i trzy detali. Ponieważ ludzkie oko jest czułe na poszczególne barwy przestrzeni RGB w stosunku 3:6:1 oraz, że metody kompresji szczególnie mocno ingerują w barwę niebieską, zatem zdecydowano się wykorzystywać do ukrywania jedynie barwę czerwoną.

Kolejny fragment klucza określa, w którym podpaśmie detali wiadomość będzie ukrywana, zaś następny fragment klucza określa dokładną lokalizację zmienianych współczynników. Właściwe ukrywanie bajtów wiadomości odbywa się poprzez podmianę wartości współczynników. Na koniec przeprowadzana jest odwrotna transformata falkowa. Proces odczytywania wiadomości jest procesem analogicznym, z tym, że w miejscu ukrywania następuje odczytanie wiadomości.

W proponowanym algorytmie wykorzystano ponadto zmianę klucza przed każdym jego pełnym użyciem, tzn przed ukryciem kolejnego bajtu wiadomości. W tym celu klucz rozszerza się o ziarno, nieużywane w samym procesie ukrywania. Przed kolejnym użyciem pełnego klucza, klucz jest modyfikowany poprzez użycie funkcji mieszającej z dodatkowym ziarnem dającym dodatkowy poziom zabezpieczenia. Ziarno to ma za zadanie dać dodatkowe bezpieczeństwo w przypadku odgadnięcia klucza na którymkolwiek etapie ukrywania i w przypadku użycia bezpiecznego algorytmu mieszającego zabezpieczyć klucz przed wnioskowaniem wprzód i wstecz.

4. Analiza efektywności proponowanego algorytmu

W celu zbadania możliwości proponowanego algorytmu wykorzystano testową bazę UCID [10]. Baza ta składa się obecnie z 1320 kolorowych obrazów w rozmiarze 512x384 piksele dla orientacji poziomej oraz obrazów 384x512 pikseli dla orientacji pionowej. Baza ta zawiera zdjęcia w formacie TIFF pochodzące z aparatu Minolta Dimage 5, który pozwala na zapisywanie zdjęć nieprzetworzonych przez elektronikę aparatu w postaci bez kompresji [11]. Znajdują się w niej zdjęcia zarówno robione na dworze, jak i w pomieszczeniach.

Wszystkie zdjęcia były wykonywane na ustawieniach automatycznych aparatu, co pozwoliło autorom stworzyć bazę zdjęć wykonywanych przez przeciętnego użytkownika aparatu. Aparat samodzielnie dobierał parametry ekspozycji, kontrastu oraz balans bieli.

Baza UCID zapewnia użytkownikom [11, 12]:
- standardowy zestaw danych do badań efektywności kompresji
- standardowy zestaw danych dla oceny stopnia kompresji
- trudny zbiór danych do oceny algorytmów wyszukiwania obrazów
- zbiór danych do oceny technik kompresji obrazów i kwantyzacji kolorów
- sprawdzony zestaw danych dostępny dla badań naukowych.

Dla bazy UCID uzyskano średnie wartości miar charakteryzujących obraz, które przedstawiono w tabeli 1.

Tab. 1. Miary charakterystyczne dla bazy UCID
Tab. 1. Characteristics measures for the UCID database

Atrybut	Obraz
Liczba odcieni czerwieni	60.4
Średnia	72.3637
Wariancja	5188.68
Dyspresja	71.6531
Energia histogramu	0.0644
Kontrast	0.3721
Entropia macierzy zdarzeń	0.3721
Jednorodność	0.3721

Zauważono, że baza UCID posiada obrazy o zmniejszonym poziomie co do liczby odcieni barwy czerwonej w stosunku do obrazów pozyskiwanych z formatu JPEG. Charakterystyczny dla tej bazy jest także fakt, że wszystkie obrazy posiadają podobne poziomy liczby poszczególnych odcieni oraz średnie histogramu kolorów dla wszystkich barw, tj całej przestrzeni RGB. Potwierdza to fakt, że obrazy te wcześniej nie były w jakikolwiek sposób kompresowane.

W trakcie badań wybierane były wszystkie obrazy z bazy, przy czym obrazy zapisane w orientacji pionowej były obracane do orientacji poziomej. Ponadto w trakcie ukrywania stosowany był losowy klucz wyznaczającym losowo miejsca ukrycia wiadomości, falkę użytą do dekompozycji oraz współczynniki wstawiania wiadomości.

Przeprowadzono następujące analizy:

a) z wykorzystaniem maksymalnej pojemności obrazu, tj próbie ukrycia wiadomości we wszystkich współczynnikach falkowych.
b) z wykorzystaniem losowego klucza określającego minimalny poziom zmian kontrastu jaki musi zaistnieć, aby móc ukryć wiadomość.

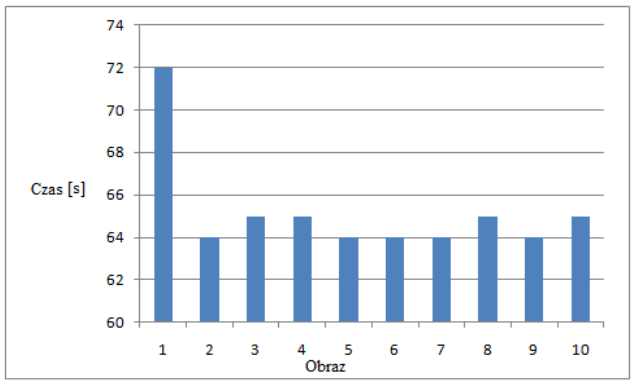
W bazie UCID każdy z obrazów posiada 196608 pikseli. Stosując prezentowany algorytm dla przypadku, gdy klucz pozwala ukrywać wiadomość we wszystkich możliwych współczynnikach, otrzymano średnią liczbę zmian kontrastu wynoszącą 14026.5

oraz średnią liczbę miejsc ukrycia wiadomości 49152 współczynników.

W przypadku zastosowania pełnego klucza pseudolosowego, średnia liczba miejsc ukrycia wiadomości wynosiła 555 współczynników, zaś liczba zmian kontrastu wynosiła 2779.3.

Przy założeniu ukrywania bajtu wiadomości w jednym współczynniku, w typowej realizacji algorytmu ukrywanych było średnio 550 bajtów wiadomości. W najlepszym wypadku ukrywanych było 49 KB danych.

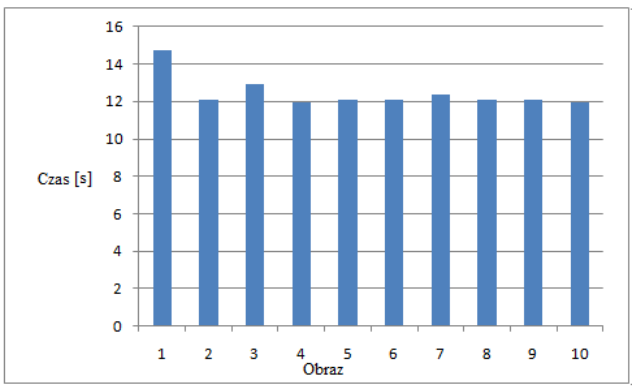
Przeprowadzono także analizę szybkości działania proponowanego rozwiązania. Średni czas realizacji algorytmu bez optymalizacji przedstawia wykres na rysunku 4. Do badań wykorzystano komputer klasy PC z procesorem Intel Core 2 Duo E8200 (2.66 GHz) z pamięcią 4 GB DDR2 1066 MHz. Algorytm został zrealizowany w środowisku Matlab.



Rys. 4. Czas wykonania bez optymalizacji
Fig. 4. Execution time without optimization

Średni czas wykonania algorytmu steganograficznego dla 10 kolejnych obrazów z bazy UCID wynosił 65 sekund. Na każdym etapie następowało wyszukiwanie następnych współrzędnych współczynników, w których ukrywana była wiadomość. Początkowa anomalia w postaci dłuższego wykonania algorytmu związana była z koniecznością przeprowadzenia operacji przydzielenia pamięci oraz inicjacji danych. W pozostałych przypadkach średni czas wykonania był podobny.

Ponieważ czasy realizacji były długie, zaproponowano więc optymalizację algorytmu. Zaproponowano jednokrotne generowanie grafu przejść na początku wszystkich operacji, a do jego generowania proponuje się używać algorytmu losowania z odrzucaniem. Ponadto rozwinięto macierz prostokątną z kolejnymi koordynatami do przejścia do tablicy jednowymiarowej. Ułatwia to proces wyszukiwania kolejnych współrzędnych współczynników. W wyniku tych operacji uzyskano przyspieszenie pokazane na rysunku 5.



Rys. 5. Czas wykonania po optymalizacji
Fig. 5. Execution time after optimization

Dzięki optymalizacji uzyskano czterokrotne przyspieszenie, a wiadomość była ukrywa w średnim czasie wynoszącym 12.1 sekundy. Wyliczono także średni czas wyszukiwania kolejnych współczynników dla obrazów z bazy UCID. Średni czas pełnego wyszukiwania wszystkich współczynników wynosił 11.3 sekundy. To oznacza, że w przypadku zastosowania środowiska Matlab czas potrzebny na ukrywanie wiadomości wynosił 0.6 sekundy. Implementacja programowa powinna dodatkowo ten czas skrócić. Uznaje się zatem, że realizacja algorytmu ukrywania była optymalna. Przyspieszenie wymagane jest dla algorytmów wyszukiwania miejsc dla ukrywania kolejnych bajtów wiadomości. Jest to problem związany z przeszukiwaniem grafów i nie leży w zakresie niniejszej pracy.

5. Wnioski

W artykule przedstawiono propozycję nowego algorytmu steganograficznego dla cyfrowych środowisk graficznych wykorzystującego rozproszony klucz. Wykonano analizę jego efektywności określającą czasy wykonywania ukrywania wiadomości w sposób niewidoczny dla ludzkiego oka oraz przedstawiono sposoby na optymalizację czasów wykonań tych operacji.

Ponieważ potencjalny atakujący musiałby sprawdzać wszystkie obrazy w sieci pod kątem ukrycia w nich wiadomości oraz chcąc odczytać wiadomość, więc w przypadku średniej wielkości witryn proces ten stałby się trudny do wykonania. Prowadzone równolegle badania na jednym z portali komputerowych w Polsce wykazały, że łączny czas wykonywania tych operacji dla tej witryny bez optymalizacji wyniósłby 4554160 sekund, zaś po optymalizacji 847,77 tysięcy sekund. Gdyby zastosować szybkie wyszukiwanie zbliżające się do zera, czas ten wyniósłby 42 tysiące sekund. Czasy te są podane dla witryny posiadającej liczbę 70064 obrazów JPG. Średnia wielkość obrazka wynosiła 31,44 KB.

6. Literatura

[1] http://news.netcraft.com/archives/2010/11/05/november-2010-web-server-survey.html (dostęp listopad 2010).
[2] http://www.newwebsitebuilders.net/Astounding%20Internet%20World%20Stats%20and%20Online%20Growth%20Exposure.pdf (dostęp listopad 2010).
[3] http://www.stat.gov.pl/cps/rde/xbcr/gus/PUBL_oz_maly_rocznik_statystyczny_2009.pdf (dostęp listopad 2010).
[4] Labovitz C., Iekel-Johnson S., McPherson D., Oberheide J., Jahanian F.: Internet Inter-Domain Traffic, SIGCOMM'10, August 30 – September 3, 2010, New Delhi, India.,
[5] http://imgur.com/stats/ (dostęp listopad 2010).
[6] Korostil J., Nozdrzykowski Ł.: The security level of a particular blind steganographic systems. ADVANCED COMPUTER SYSTEMS, October 14-16, 2009, Międzyzdroje, Poland 2009. Elektronika nr 11/2009.
[7] Pyka K.: Uwarunkowania fizjologiczne i techniczne wpływające na percepcję obrazu obserwowanego na ekranie monitora; http://home.agh.edu.pl/~zfiit/publikacje_pliki/Pyka_2005b.pdf (dostęp listopad 2010).
[8] Noureddine Moumkine, Ahmed Tamtaoui, Abdellah Ait Ouahman: Integration of the Contrast Sensitive Fumction into Wavelet Codec. ISCCSP 2006, Marrakech, Maroc 13-15 March 2006.
[9] Nozdrzykowski Ł.: Analiza istotności miar obrazu wpływających na zmniejszenie zniekształceń spowodowanych steganograficznym ukrywaniem wiadomości w obrazach w oparciu o funkcję CSF. Metody Informatyki Stosowanej 3/2009. Szczecin 2009.
[10] http://vision.doc.ntu.ac.uk/ (dostęp maj 2011).
[11] Schaefer G., Stich M.: UCID - An Uncompressed Colour Image Database (dostęp maj 2011).
[12] Picard R.: Content access for image/video coding: The fourth criterion, Tech. Rep. 195, MIT Media Lab, 1994.

otrzymano / received: 06.12.2011
przyjęto do druku / accepted: 03.01.2012 *artykuł recenzowany / revised paper*