Piotr WIŚNIEWSKI, Marek R. OGIELA
AGH - AKADEMIA GÓRNICZO-HUTNICZA W KRAKOWIE, KATEDRA AUTOMATYKI I INŻYNIERII BIOMEDYCZNEJ
Al. A. Mickiewicza 30, 30-059 Kraków

Wybrane zagadnienia implementacyjne zaawansowanych algorytmów kryptografii wizualnej

Mgr inż. Piotr WIŚNIEWSKI

Doktorant na wydziale Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej Akademii Górniczo – Hutniczej w Krakowie w dyscyplinie Automatyka i Robotyka. Prowadzi badania nad algorytmami podziału sekretu i ich zastosowaniach w utajnianiu informacji obrazowych.

e-mail: wpiotr@agh.edu.pl

Prof. dr hab. Marek R. OGIELA

Profesor zwyczajny na Akademii Górniczo – Hutniczej w Krakowie. Prowadzi badania nad kognitywnymi systemami informacyjnymi nowej generacji, a także kryptografią i podziałem sekretów. Jest członkiem wielu renomowanych towarzystw naukowych, a także autorem ponad 250 publikacji o zakresie międzynarodowym.

e-mail: mogiela@agh.edu.pl

Streszczenie

W publikacji scharakteryzowano dwa zaawansowane algorytmy kryptografii wizualnej: tzw. rozszerzoną kryptografię wizualną, w której istnieje możliwość umieszczenia fałszywych danych w sekretnych częściach obrazu, oraz skuteczną technikę szyfrowania obrazów kolorowych. Zaprezentowano także przykładową aplikację umożliwiającą użycie jednej z czterech metod oraz odczyt utajnionej informacji. Otrzymane wyniki działania opisywanych algorytmów potwierdzają wysoką skuteczność tego rodzaju technik kryptograficznych oraz zasadność ich praktycznego zastosowania.

Słowa kluczowe: podział sekretu, kryptografia wizualna, szyfrowanie informacji obrazowej.

Selected aspects of implementation of advanced visual cryptography algorithms

Abstract

The paper presents two advanced visual cryptography algorithms: extended visual cryptography (EVCS) [2], where it is possible to place false information in shares of an image, and an effective encryption method for color images. This publication also presents an example application, which allows its user to execute one of four visual cryptography algorithms and to decrypt the secret information by joining shares. The first section shows the purpose of encrypting image data and main advantages of visual cryptography. In the second section an idea of secret sharing is presented. The third section describes visual cryptography techniques, where it is possible to create meaningful shares by encoding false information into shadow images. An example of (2,4)-threshold EVCS scheme is presented for a secret binary image. The fourth section presents a visual cryptography algorithm suitable for color images, which was analysed using a sample image. In the fifth section a visual cryptography system is proposed and its main features are presented, including procedure descriptions and obtained execution times of encryption algorithms. The summary presents the advantages of advanced visual cryptography algorithms and the utility of the application created for research. The obtained results of executing the described algorithms confirm the efficiency of these cryptographic techniques and the validity of its practical applications.

Keywords: secret sharing, visual cryptography, image encryption.

1. Wprowadzenie

W związku z rosnącym zapotrzebowaniem na duże zasoby pamięciowe oraz coraz większą przepustowość łączy sieciowych, przesyłane dane tekstowe coraz częściej zastępowane są przez obrazy i inne pliki multimedialne.

Szyfrowanie danych obrazowych wykorzystywane jest w sytuacjach takich jak: przesyłanie sygnału telewizyjnego, archiwizacja obrazów medycznych, transmisja planów i schematów o znaczeniu strategicznym itp. Szeroko stosowane algorytmy kryptograficzne wymagają często bardzo skomplikowanych obliczeń zarówno podczas procesu szyfrowania, jak i odzyskiwania utajnionej informacji. Nieco inaczej jest w przypadku kryptografii wizualnej, która jest bezpiecznym sposobem utajniania informacji, polegającym na utworzeniu z obrazu wejściowego określonej liczby nieczytelnych części. Ważną zaletą opisywanej techniki jest możliwość odczytania sekretnych danych bez konieczności wykonywania złożonych operacji matematycznych. Od adresata przekazu nie wymaga się także znajomości wykorzystanej metody, gdyż jedyną czynnością niezbędną do odzyskania zakodowanego obrazu jest fizyczne połączenie ze sobą części powstałych w wyniku procesu szyfrowania.

Zobrazowanie sekretu oraz sprawdzenie poprawności metod najczęściej zapewnia system kryptograficzny lub aplikacja, która umożliwia utajnienie dowolnej informacji wizualnej wczytywanej z pliku. W niniejszej publikacji zostanie również opisany autorski system podziału informacji obrazowej, bazujący na implementacji kilku wybranych algorytmów kryptografii wizualnej.

2. Metody podziału sekretu

Charakteryzując techniki kryptografii wizualnej warto podkreślić, że dzielenie sekretu polega na podzieleniu tajnej informacji pomiędzy pewną grupę uczestników protokołu, spośród których każdy będzie posiadał jakąś jej część, nazywaną udziałem lub cieniem. Sekret może być odczytany wyłącznie w przypadku, gdy odpowiednia liczba cieni zostanie ze sobą połączona. Jeżeli udziały są rozdzielone nie jest możliwe uzyskanie jakiejkolwiek wskazówki dotyczącej zakodowanej informacji, innymi słowy każdy pojedynczy udział jest bezużyteczny [7].

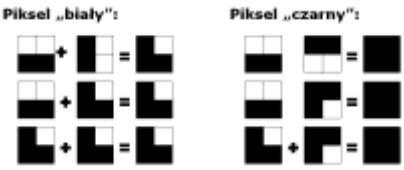
Opisywana w tej publikacji kryptografia wizualna opiera się na metodzie dzielenia sekretu i polega na podzieleniu szyfrowanego obrazu na części, z których każda jest niemożliwa do odczytania i sprawia wrażenie losowego szumu. W podstawowym przypadku stosuje się schemat (k, n)-progowy, w którym, zgodnie z powyższą definicją, obraz dzielony jest na n udziałów i jego ponowne odczytanie jest całkowicie niemożliwe dopóki nie nałoży się na siebie co najmniej $k < n$ części [7]. W dalszej części zaprezentowane zostaną wybrane algorytmy kryptografii wizualnej przeznaczone do podziału obrazów kolorowych lub umieszczania fałszywych informacji w sekretnych częściach składowych.

3. Techniki podziału z fałszywą informacją ukrywaną w sekretnych udziałach

Autorzy pracy [6] przytaczają przykład, w którym w miejsce nieczytelnych udziałów, do odczytania zaszyfrowanej informacji wykorzystuje się nałożenie na siebie dwóch obrazów o konkretnym znaczeniu.

W przypadku schematu (2, 2), w obydwóch udziałach kolor biały reprezentowany jest przez dwa czarne subpiksele, a piksel czarny przez trzy (rys. 1). Przy takich założeniach każdy udział

może być obrazem przedstawiającym dowolny obiekt i pomimo tego nie dawać żadnej informacji o zakodowanym obrazie. Jedyną wadą tej metody jest pogorszenie kontrastu wynikające z mniejszej różnicy względnej pomiędzy reprezentacją barw w porównaniu z klasycznym algorytmem.



Rys. 1. Przykład łączenia dwóch udziałów zawierających dane. Kolor piksela wyjściowego nie zależy od kolorów pikseli udziałów, lecz od sposobu nałożenia ich na siebie
Fig. 1. An example of stacking two shares containing data. Resulting pixel color does not depend on pixel colors in shares, but on their superimposition

W pracy [2] przedstawiono inny algorytm rozszerzonej kryptografii wizualnej (ang. Extended Visual Cryptography Scheme) w którym zakodowana informacja może być odczytana tylko w przypadku połączenia określonych kombinacji udziałów.

Do analizy algorytmu wybrano obraz przedstawiony na rys. 2 oraz schemat progowy (2, 4) z liczbą subpikseli m równą 9.

Rys. 2. Zadany obraz binarny zawierający informację tekstową
Fig. 2. Binary input image containing text

Przebieg metody jest następujący:

1. Definiuje się n obrazów binarnych o wymiarach obrazu wejściowego, które posłużą do tworzenia udziałów zawierających fałszywą informację. Przykładowe obrazy mogą zawierać numery: 607905, 140604, 140524, 609044.
2. Należy określić strukturę dostępu [3], to znaczy parę zbiorów kombinacji dozwolonych (ang. qualified sets) i zabronionych (ang. forbidden sets) oznaczaną przez (Γ_{Qual}, Γ_{Forb}). W analizowanym przypadku założono, że zakodowana informacja ma być czytelna po połączeniu udziałów 1 i 4 oraz 1, 2 i 3, zatem dozwolone będą wszystkie podzbiory zawierające te kombinacje: Γ_{Qual} = {{1,4}, {1,2,4}, {1,2,3}, {1,3,4}, {1,2,3,4}}, a pozostałe pary udziałów stanowić będą kombinacje zabronione.
3. Wyznacza się macierze bazowe S_0 i S_1 rozmiaru $n \times (m-2)$ w taki sposób, żeby sumy wierszy odpowiadające danym kombinacjom dozwolonym dawały niezerową różnicę względną, zaś w przypadku kombinacji zabronionych różnica ta powinna wynosić 0. Dzięki temu, w przypadku połączenia ze sobą udziałów ze zbioru Γ_{Forb} nie będzie możliwe rozróżnienie reprezentacji pikseli czarnych i białych.

$$S_1 = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$S_0 = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}. \quad (1)$$

4. Tworzy się 2^n macierzy binarnych $D^{c1\ldots c\,n}$ o rozmiarach $(n \times 2)$, gdzie c_i jest kolorem odpowiadającego piksela w i-tym obrazie z fałszywą informacją. W przypadku piksela białego (oznaczanego literą w) w danym wierszu macierzy umieszcza się wartości [1 0] lub [0 1] , natomiast w przypadku piksela czarnego (oznaczanego literą b) umieszcza się każdorazowo wartości [1 1]. Przykładowo, jeżeli odpowiadające piksele wszystkich czterech obrazów są białe, macierz D ma postać:

$$D^{wwww} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (2)$$

5. W wyniku złączenia macierzy bazowych z macierzami $D^{c1\ldots c\,n}$ powstaje 2^{2n} macierzy $S^{c1\ldots c\,n}$, gdzie c oznacza kolor odpowiadającego piksela w obrazie wejściowym. Przykładowo, jeżeli piksele obrazu wejściowego oraz fałszywych obrazów 2 i 3 są czarne, a pozostałe mają barwę białą, macierz S przedstawia wyrażenie (3):

$$S_b^{wbbw} = [S_1 \quad D^{wbbw}] = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3)$$

6. W wyniku dowolnych permutacji kolumn macierzy $S^{c1\ldots c\,n}$ powstają macierze $_{Sc}{}^{c1\ldots c\,n}$, w których każda suma wierszy odpowiadająca kombinacjom dozwolonym, przy pikselach obrazów fałszywych o kolorach $c_1 \ldots c_n$, wyznacza reprezentację piksela w kolorze c.

Działanie metody sprawdzono z zadanym obrazem wejściowym (rys. 2) oraz czterema obrazami fałszywymi. Otrzymano cztery różne udziały zaprezentowane na rys. 3:

Rys. 3. Cztery udziały z fałszywymi informacjami
Fig. 3. Four shares containing false information

Następnie przeprowadzono łączenie udziałów, które potwierdziło działanie metody i poprawność napisanego algorytmu. Połączono udziały ze zbioru kombinacji dozwolonych.

Rys. 4. Rezultat połączenia udziałów 1 i 4
Fig. 4. Result of joining shares 1 and 4

Najlepszy kontrast i tym samym największą czytelność rozszyfrowanej informacji zapewnia suma udziałów 1 i 4, co wynika z największej różnicy względnej pomiędzy reprezentacjami białych i czarnych pikseli. Po złączeniu trzech pierwszych udziałów uzyskany tekst jest wciąż w pełni czytelny.

W celu zobrazowania szczególnych właściwości rozszerzonej kryptografii wizualnej zsumowano także udziały ze zbioru kombinacji zabronionych. Otrzymane wyniki nie zawierały żadnej informacji lub informację nieczytelną, będącą wynikiem pomieszania fałszywych informacji z poszczególnych udziałów (rysunek 5).

Rys. 5. Rezultat połączenia udziałów 2 i 4
Fig. 5. Result of joining shares 2 and 4

4. Szyfrowanie obrazów kolorowych

Istnieje wiele algorytmów kryptografii wizualnej, które pozwalają na szyfrowanie obrazów kolorowych. Szczególnie interesujące wydają się rozwiązania przedstawione w [5], ze względu na stosunkowo dobre odwzorowanie obrazu wejściowego oraz spełnienie głównych założeń, to znaczy brak konieczności wykonywania złożonych operacji matematycznych podczas odczytywania informacji.

W opisywanym przypadku wykorzystuje się subtraktywny model barw (CMY). Aby skorzystać z wybranego algorytmu należy w pierwszej kolejności rozłożyć obraz wejściowy na trzy binarne obrazy odpowiadające kolorom podstawowym, a następnie przekształcić je do postaci półtonowej, na przykład za pomocą metody Bayera opisanej w [4].

Rys. 6. Obraz wykorzystany do analizy algorytmu
Fig. 6. Image used for algorithm analysis

Szyfrowania obrazu kolorowego można dokonać poprzez zastosowanie jednego z trzech opisanych w [5] algorytmów. Do dalszej analizy wybrano metodę I, której przebieg przedstawiono poniżej:

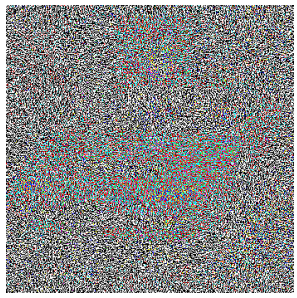
- Dla każdego piksela P_{ij} wyznacza się maskę rozmiaru (2×2), w której losowo rozmieszcza się dwa czarne subpiksele (pozostałe dwa są białe). Oznacza to, że każdemu pikselowi obrazu wejściowego przyporządkowana zostaje losowa reprezentacja białego piksela w schemacie dla czterech subpikseli (Tabela 1). Wyznaczona maska wizualnie nie różni się niczym od udziałów wykorzystywanych w algorytmach klasycznych.
- Następnie, na podstawie składowych półtonowych obrazu wejściowego tworzone są trzy udziały. Przykładowo, dla składowej żółtej (Y) zachodzi warunek:
  a. jeżeli wartość piksela Y_{ij} wynosi 1 (piksel jest żółty), to żółte piksele umieszcza się na pozycjach odpowiadających zerowym wartościom subpikseli maski,
  b. jeżeli wartość piksela Y_{ij} wynosi 0 (piksel jest biały), to żółte piksele umieszcza się na pozycjach odpowiadających niezerowym wartościom subpikseli maski.
- Każda ze składowych jest obrazem binarnym, zatem w analizie teoretycznej można przyjąć, że wartość 1 odpowiada subpikselowi danej barwy, a wartość 0 subpikselowi białemu. Zarówno maska jak i wszystkie udziały mają wymiary dwukrotnie większe od obrazu wejściowego.

Tab. 1. Rozkład i rekonstrukcja czerwonego piksela. Na podstawie [5]
Tab. 1. Red pixel decomposition and reconstruction. Based on [5]

Piksel wejściowy					
			$P_{ij} = (0, 1, 1)$		
Rozkład na kolory podstawowe					
			$C_{ij} = 0$	$M_{ij} = 1$	$Y_{ij} = 1$
Tworzenie udziałów					
Maska	Udział 1 (C)		Udział 2 (M)		Udział 3 (Y)
Odczyt utajnionego piksela					

Po wykonaniu tych kroków, aby odczytać zaszyfrowaną informację, należy nałożyć na siebie maskę oraz trzy udziały. Obiekt z obrazu wejściowego staje się rozpoznawalny już po nałożeniu na siebie maski i dowolnego udziału kolorowego (rys. 7). Suma dwóch lub trzech udziałów kolorowych bez użycia maski nie daje żadnej informacji o zdjęciu (przykład widoczny na rysunku 8).

Rys. 7. Maska + udział C
Fig. 7. Mask + share C



Rys. 8. Udział C + udział M
Fig. 8. Share C + share M

Uzyskanie pełnej informacji o kolorach możliwe jest dopiero po zsumowaniu maski oraz wszystkich trzech udziałów (rys. 9). Mimo pewnej utraty jakości w stosunku do obrazu wejściowego jakość otrzymanego wyniku jest na zadowalającym poziomie (tzn. obraz jest całkowicie czytelny, a obiekt w pełni rozpoznawalny).

Rys. 9. Zrekonstruowany obraz
Fig. 9. Reconstructed image

5. System realizujący kryptografię wizualną

Do szyfrowania obrazów stosuje się różnego rodzaju algorytmy kryptografii wizualnej, w zależności od wybranego schematu progowego i skali kolorów obrazu wejściowego [1]. Dostępne publicznie implementacje programowe ograniczają się jednak przeważnie do symulacji jednej metody i działają poprawnie wyłącznie dla niewielkich obrazów wejściowych.

Mając na uwadze powyższe uwarunkowania, utworzono aplikację pozwalającą na szyfrowanie zadanej informacji z wykorzystaniem kilku różnych algorytmów i znajdującą zastosowanie zarówno dla obrazów monochromatycznych, jak i kolorowych. Sformułowano następujące wymagania:

- obsługa obrazów w standardzie HD1080,
- wybór obrazu wejściowego z pliku,
- zapis udziałów oraz rozszyfrowanej informacji w pliku,
- wybór algorytmu,
- odczyt zakodowanej informacji z plików,
- funkcjonowanie aplikacji bez konieczności instalowania dodatkowego oprogramowania.

W realizowanej implementacji wprowadzono podział metod odpowiadający ich zastosowaniom:

1. Funkcje dla obrazów w skali szarości (sprawdzenie warunków wykonania metody, utworzenie udziałów w schemacie (2, 2), dithering, uruchomienie schematu (2, 2), uruchomienie metody dla obrazów w skali szarości, uruchomienie metody EVCS).
2. Funkcje dla obrazów kolorowych (sprawdzenie warunków wykonania metody, rozkład obrazu w schemacie CMY, utworzenie udziałów od kolorów podstawowych, uruchomienie metody dla obrazów kolorowych).
3. Funkcje pomocnicze (losowa permutacja kolumn macierzy, przekształcenie obrazu do zapisu binarnego, przekształcenie obrazu do zapisu 8-bitowego).

Niektóre funkcje dla obrazów w skali szarości stanowią także etapy szyfrowania obrazów kolorowych. W celu optymalizacji kodu, w programie uwzględniono dostęp do tych funkcji z drugiej grupy metod.

Aplikacja została napisana w powszechnie stosowanym języku C++, z wykorzystaniem edytora Qt Creator 2.8.1 pracującego w środowisku Qt 4.8.4. Działania na obrazach wykonano w oparciu o bibliotekę CImg, a operacje na macierzach korzystając z biblioteki Eigen. Poprawność wykonanych algorytmów oraz otrzymywanych wyników sprawdzono poprzez uruchomienie podobnych skryptów w programie Matlab. Program skompilowano z użyciem zintegrowanego z edytorem kompilatora MinGW 4.4.0 w systemie Windows 7 i 8.

Główne okno systemu, które zawiera cały interfejs użytkownika niezbędny do przeprowadzenia szyfrowania zaprezentowano na rysunku 10.

Rys. 10. Główne okno aplikacji
Fig. 10. Application main window

System ten daje użytkownikowi dostęp do dwóch głównych działań: szyfrowania obrazu oraz odczytu zakodowanej informacji z niezależnych plików. Poniżej opisano kolejne czynności, jakie należy wykonać aby przeprowadzić zadaną procedurę.

Szyfrowanie
1. Załadowanie obrazu z pliku.
2. Wybór algorytmu szyfrującego.
   Dostępne są cztery algorytmy: obrazy binarne w schemacie (2, 2), rozszerzona kryptografia wizualna dla dwóch udziałów, obrazy w skali szarości, obrazy kolorowe.
3. Uruchomienie szyfrowania.
   Wybrany algorytm uruchamia się poprzez naciśnięcie przycisku *Run*. Szyfrowanie nie zostanie rozpoczęte, jeżeli skala kolorów obrazu nie jest zgodna z zaznaczonym algorytmem. W przypadku algorytmu EVCS konieczne jest załadowanie z plików dwóch obrazów z fałszywymi informacjami, które muszą być binarne, a ich wymiary powinny być równe szerokości i wysokości obrazu wejściowego.
4. Wynik.
   Po wykonaniu algorytmu szyfrującego w katalogu zapisywane są udziały. Wynik nałożenia na siebie udziałów wyświetla się w oknie obrazu wyjściowego i zapisywany jest do pliku *result.bmp*.

Odczytanie zakodowanej informacji
1. Wczytanie udziałów.
   Uruchomienie procedury deszyfrującej odbywa się po wyborze opcji *Decrypt*. W pierwszej kolejności należy określić liczbę udziałów, które zostaną na siebie nałożone. Następnie należy wybrać jeden z udziałów. Operacja powtarza się do czasu wczytania tylu części, ile określono w oknie dialogowym.
2. Wynik.
   Procedura odczytania informacji działa w taki sam sposób niezależnie od skali kolorów wczytanych udziałów. Ich suma zapisywana jest do pliku *secret.bmp*.

Działanie programu sprawdzono uruchamiając aplikację dla różnych zestawów obrazów testowych: o rozmiarze VGA (640 × 480) oraz HD1080 (1920 × 1080). Szyfrowanie wszystkich danych przebiegło pomyślnie. Odczytane informacje obrazowe i tekstowe były w pełni czytelne, a na podstawie pojedynczego udziału nie można było uzyskać żadnych danych dotyczących obrazu wejściowego. Obraz wyjściowy został wygenerowany poprawnie, co potwierdził zerowy wynik algorytmu obliczającego różnicę pomiędzy wyjściem danej metody a informacją odczytaną po nałożeniu na siebie udziałów z wczytanych plików.

Utworzona aplikacja pozwala także określić czasy wykonania poszczególnych algorytmów w zależności od rozmiaru obrazu. Do tego celu wykorzystano bibliotekę *Chrono*. Zestawienie przykładowych czasów kodowania przedstawia Tabela 2. W nawiasach podano dla porównania wyniki uruchomienia analogicznych skryptów w programie Matlab.

Tab. 2. Czasy wykonania algorytmów szyfrujących, s
Tab. 2. Execution times of encryption algorithms (in seconds)

Algorytm	Obraz VGA	Obraz HD1080
Monochromatyczny (2,2)	0,596 (4,413)	3,964 (28,815)
Rozszerzony (2,2)	0,693 (8,401)	4,76 (57,892)
Skala szarości	0,608 (4,548)	4,206 (81,8)
Kolor	1,126 (7,387)	7,936 (155,148)

Otrzymane dane dowodzą, że opracowany system bazujący na dedykowanych bibliotekach do przetwarzania obrazów znacząco poprawia szybkość szyfrowania zwłaszcza w przypadku zdjęć o dużym rozmiarze. Najdłuższe czasy wykonania, wyższe o około 100% w porównaniu z podstawowym algorytmem, zaobserwowano dla obrazu kolorowego. Taka różnica wynika z konieczności powtórzenia kilku operacji i większej liczby zmiennych wykorzystywanych podczas działania. Efektywność obliczeniowa stworzonego systemu pozwala na jego wykorzystanie w podziale sekretów obrazowych składających się nie tylko z pojedynczych zobrazowań, ale również i sekwencji obrazów.

6. Podsumowanie

Celem prowadzonych badań było porównanie efektywności oraz skuteczności szyfrowania obrazów za pomocą technik kryptografii wizualnej. Efekty przeprowadzonych doświadczeń potwierdzają zalety kryptografii wizualnej, do których należy przede wszystkim zaliczyć bezpieczeństwo, czyli brak możliwości odczytania tajnej informacji za pomocą pojedynczej części sekretu, ale także efektywność obliczeniową tego rodzaju metod. Warto również podkreślić łatwość rozszyfrowania danych przy spełnieniu warunku wymaganej liczby udziałów. Adresat przekazu może uzyskać sekretny obraz dysponując minimalnymi mocami obliczeniowymi.

Szczególnie interesującą metodą jest tzw. rozszerzona kryptografia wizualna, w której udziały zawierają fałszywe informacje, a jednocześnie wyglądają podobnie do rozszyfrowanych obrazów. Ciekawą cechą takich metod jest także możliwość zdefiniowania dozwolonych i zabronionych kombinacji sekretnych części, co pozwala na utworzenie różnych poziomów dostępu do informacji pośród uczestników procesu dzielenia sekretu.

Utworzona na użytek prowadzonych badań aplikacja pozwoliła wykazać, że czasy wykonania algorytmów kryptografii wizualnej są zadowalające nawet dla obrazów o dużym rozmiarze, co pozwala na bezproblemowe szyfrowanie zdjęć wysokiej jakości. Powstały program jest także odporny na nieprawidłowe dane wejściowe i pozwala na realizację zadań podziału sekretnych danych dla dowolnej liczby uczestników protokołu współdzielenia informacji.

7. Literatura

[1] Anjum A., Revenkar P.S., Gandhare W.Z.: Survey of Visual Cryp-tography schemes. International Journal of Security and Its Applications. Vol. 4, no. 2, s. 49-56, 2010.
[2] Ateniese G. et al.: Extended Capabilities for Visual Cryptography. Theoretical Computer Science, vol. 250, no. 1-2, s. 143-161, 2001.
[3] Ateniese G. et al.: Visual Cryptography for General Access Structures. Information and Computation, vol. 129, no. 2, s. 86-106, 1996.
[4] Crocker L., Boulay P., Morra M.: Digital halftoning. Dostępny w Intenecie: http://www.efg.com/Lab/Library/ImageProcessing.
[5] Hou Y.C.: Visual Cryptography for color images. Pattern Recognition, vol. 36, s. 1619-1629, 2003.
[6] Naor M., Shamir A.: Visual Cryptography. Advances in Cryptology – Eurocrypt '94, vol. 950, s. 1-12, 1994.
[7] Ogiela M., Ogiela U.: Lingwistyczne schematy progowe w inteligentnym zarządzaniu sekretnymi danymi. PAK, vol. 57, s. 315-319, 2011.

otrzymano / received: 28.09.2014
przyjęto do druku / accepted: 03.11.2014 *artykuł recenzowany / revised paper*