

WOJSKOWA AKADEMIA TECHNICZNA

im. Jarosława Dąbrowskiego

WYDZIAŁ CYBERNETYKI



Steganografia Lab. 2

Student

x

Prowadzący laboratoria:

y

Spis treści

Steganografia.....	1
Lab. 2	1
Treść zadania	2
Kod realizujący zadanie.....	2
Opis rozwiązania.....	6

Treść zadania

Instrukcje

1. Napisać skrypt w programie Matlab wczytujący bitmapę w 24-bitowej głębi koloru.
2. Wykorzystując podmianę najmniej znaczących bitów piksela, ukryć w obrazie tekst wskazany jako parametr odczytany z linii poleceń.
3. Dane powinny zostać ukrywane w formie spirali mającej swój początek w środku obrazu
4. Napisać skrypt odczytujący ukryte w ten sposób dane.

Zadanie zwrócić w postaci sprawozdania opisującego wykonane zadanie oraz kodów źródłowych (m-pliku). Pliki zamieścić w postaci nieskompresowanej.

Kod realizujący zadanie

Skrypt odpowiadający za ukrywanie danego ciągu znaków w obrazie – przedstawiono na stronie kolejnej oraz załączono wraz ze sprawozdaniem pod postacią pliku: „Bartosz_Miazga_Lab2.m”.

```
Bartosz_Miazga_Lab2.m  Bartosz_Miazga_Lab2_odczyt.m  +
1      %wczytanie bitmapy w 24-bitowej głębi koloru
2      image = imread('eminem_lab2.jpg');
3
4      %zmiana rozmiaru wczytanego zdjęcia
5      image = imresize(image, [1200 1200]);
6
7      %pobranie wiadomości od użytkownika
8      message = input('Podaj wiadomosc do ukrycia: ', 's');
9
10     %zamiana na kody ascii
11     ascii_value = uint8(message);
12
13     %konwersja wartości dziesiętnych na binarne
14     binary_message = transpose(dec2bin(ascii_value, 8));
15     binary_message = binary_message(:);
16
17     %zapisanie dlugosci binarnej wiadomosci
18     len_binary_message = length(binary_message);
19
20     %konwersja tablicy char na numeryczna
21     binary_num_message = str2num(binary_message);
22
23     embed_counter = 1;
24     temp_spiral_counter = 0;
25     spiral_counter = 1;
26     pos_y = 601;
27     pos_x = 600;
28     tmp = 1;
29     %przejście po obrazie
30     while embed_counter <= len_binary_message
31         if temp_spiral_counter > 1
32             spiral_counter = spiral_counter+tmp;
33             spiral_counter = spiral_counter*(-1);
34             tmp = tmp*(-1);
35             temp_spiral_counter = 0;
36         end
37         LSB = mod(double(image(pos_x, pos_y)), 2);
38         temp = double(xor(LSB, binary_num_message(embed_counter)));
39         disp('=====')
40         disp(pos_y)
41         disp(pos_x)
42         image(pos_x, pos_y) = image(pos_x, pos_y)+temp;
43         if mod(embed_counter,2) == 1
44             pos_y = pos_y + spiral_counter;
45         end
46         if mod(embed_counter,2) == 0
47             pos_x = pos_x + spiral_counter;
48         end
49         embed_counter = embed_counter+1;
50         temp_spiral_counter = temp_spiral_counter +1;
51     end
52
53     %zapisanie obrazu powstałego w wyniku przekształcen
54     imwrite(image, 'eminem_lab2_stegano.png');
```

Skrypt odpowiadający za wydobywanie ukrytego ciągu znaków z obrazu – przedstawiono poniżej oraz załączono wraz ze sprawozdaniem pod postacią pliku: „Bartosz_Miazga_Lab2_odczyt.m”.

```
Bartosz_Miazga_Lab2.m  Bartosz_Miazga_Lab2_odczyt.m  +
1      %wczytanie zdjęcia z ukrytą wiadomością
2      image2 = imread('eminem_lab2_stegano.png');
3
4      %liczba znaków ukrytej wiadomości
5      chars = input('Podaj długość wiadomości: ');
6
7      %długość wiadomości w bitach
8      message_length = chars * 8;
9
10     embed_counter = 1;
11     temp_spiral_counter = 0;
12     spiral_counter = 1;
13     pos_y = 601;
14     pos_x = 600;
15     tmp = 1;
16     %przejdź po obrazie
17     while embed_counter <= message_length
18         if temp_spiral_counter > 1
19             spiral_counter = spiral_counter+tmp;
20             spiral_counter = spiral_counter*(-1);
21             tmp = tmp*(-1);
22             temp_spiral_counter = 0;
23         end
24         extracted_bits(embed_counter, 1) = mod(double(image(pos_x, pos_y)), 2);
25         if mod(embed_counter,2) == 1
26             pos_y = pos_y + spiral_counter;
27         end
28         if mod(embed_counter,2) == 0
29             pos_x = pos_x + spiral_counter;
30         end
31         embed_counter = embed_counter+1;
32         temp_spiral_counter = temp_spiral_counter +1;
33     end
34
35     %potęgi liczby 2 do odzyskania znaków ascii z binarki
36     binValues = [ 128 64 32 16 8 4 2 1 ];
37
38     %odkodowanie wiadomości
39     binMatrix = reshape(extracted_bits, 8,(message_length/8));
40     textString = char(binValues*binMatrix);
41     disp(textString);
42
```

Poniżej zaprezentowano efekt wykonania obu skryptów, drugi skrypt wypisał wydobytą ze zdjęcia wiadomość. Natomiast pierwszy skrypt wypisuje na wyjście numery kolejnych indeksów, w których ukrywane będą bity wiadomości – pierwszy jest index osi Y, drugi to index osi X. Jak można zaobserwować, skrypt zaczyna działanie w środku obrazu, następnie przechodzi modyfikując albo zmienną x, albo zmienną y, przy czym raz na zmiennych wykonywana jest operacja odejmowania, a raz dodawania, poniżej przedstawiłem również rysunek obrazujący jak wygląda przemieszczanie się skryptu po indeksach.

```
>> Bartosz_Miazga_Lab2
Podaj wiadomosc do ukrycia: Ala ma kota.
```

```
=====
```

```
601
```

```
600
```

```
=====
```

```
602
```

```
600
```

```
=====
```

```
602
```

```
601
```

```
=====
```

```
600
```

```
601
```

```
=====
```

```
600
```

```
599
```

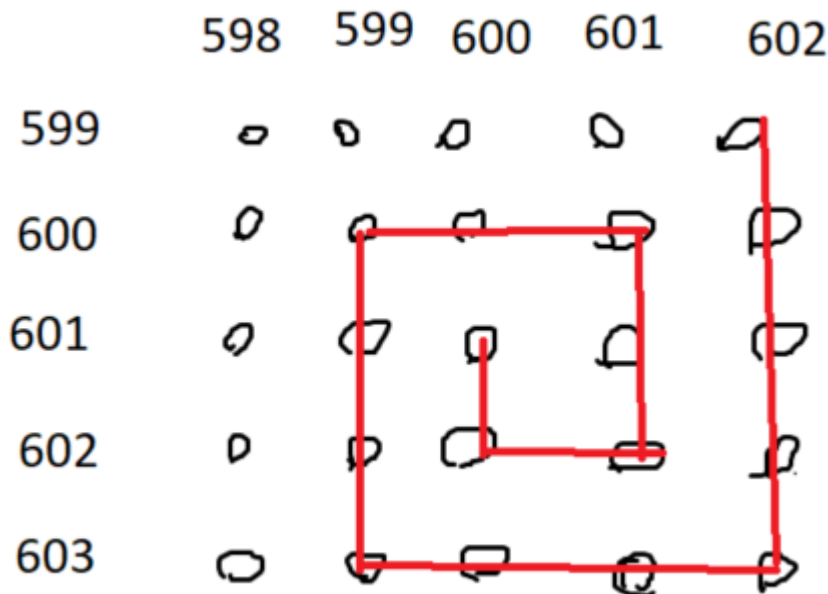
```
=====
```

```
603
```

```
599
```

```
>> Bartosz_Miazga_Lab2_odczyt
Podaj dlugosc wiadomosci: 12
Ala ma kota.
```

Poniżej przedstawiono rysunek obrazujący przemieszczanie się po indeksach w obrazie.



Opis rozwiązania

Zadanie zrealizowane zostało w postaci 2 skryptów: `Bartosz_Miazga_Lab2` oraz `Bartosz_Miazga_Lab2_odczyt`, pierwszy skrypt odpowiada za ukrycie w zdjęciu przekazanej przez użytkownika poprzez konsolę wiadomości, zdjęcie z ukrytą w sobie wiadomością zapisane zostaje pod nazwą „`eminem_lab2_stegano.png`”. Drugi skrypt wczytuje obraz zapisany pod nazwą „`eminem_lab2_stegano.png`” i wydobywa z niego ukrytą wiadomość, na samym końcu wypisując ją na ekran. Wymogiem poprawnego odczytania ukrytej wiadomości jest przekazanie jako parametr wejściowy długości ukrytej wiadomości.

Pierwszy skrypt - działanie:

Do zmiennej `image` wczytany zostaje obraz, który został dostarczony jako obraz, w którym będziemy chcieli ukrywać wiadomość – nazwa obrazu: „`eminem_lab2.jpg`”. Następnie pod zmienną `message` zostaje zapisana pobrana od użytkownika wiadomość, następnie ten ciąg znaków zamieniany jest na ciąg kodów ascii, a w kolejnym kroku ciąg kodów ascii zamieniany jest na ciąg binarny. Ciąg binarny wiadomości ukrywany jest w zdjęciu, zaczynając od piksela o indeksie (600,601) i w następnych krokach ukrywanie następuje w indeksach według algorytmu poruszającego się po spirali, najlepiej sposób poruszania się obrazuje sam kod oraz powyżej załączony rysunek pomocniczy – najpierw ze środka następuje przemieszczenie w dół, a potem w kolejnym kroku w prawo. Na sam koniec działania skryptu zapisany zostaje obraz, który ma w sobie ukrytą wiadomość.

Drugi skrypt – działanie:

Skrypt wczytuje obraz z ukrytą wiadomością, pod zmienną `chars` zostaje zapisana pobrana od użytkownika liczba znaków, z których składa się docelowa wiadomość, którą będzie trzeba „odzyskać”. W kolejnych krokach następuje iteracja, analogiczna do pierwszego skryptu (czyli iterujemy od piksela o indeksach równych (600,601)), w czasie iteracji odczytana zostaje ukryta

wiadomość, a na samym końcu następuje konwersja ukrytej wiadomości z postaci binarnej do postaci zrozumiałej dla użytkownika.