Учреждение образования

«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

«ИССЛЕДОВАНИЕ СТЕГАНОГРАФИЧЕСКОГО МЕТОДА НА ОСНОВЕ ПРЕОБРАЗОВАНИЯ НАИМЕНЕЕ ЗНАЧАЩИХ БИТОВ»

Студент: Белицкий В.Д.

ФИТ 3 курс 5 группа

Преподаватель:

Савельева Маргарита Геннадьевна

Минск 2023

Цель: изучение стеганографического метода встраивания/извлечения тайной информации с использованием электронного файла контейнера на основе преобразования наименее значащих битов (НЗБ), приобретение практических навыков программной реализации данного метода

1. **Описание приложения**

Приложение написано на языке программирования C# и позволяет:

* Зашифровать и расшифровать сообщение в изображении с использованием стенографии
* Выбор изображения для зашифрования в нем текста.

1. **Методика выполнения расчетов**

public static Bitmap HideText(string text, Bitmap bmp)

{

State state = State.Hiding;

int charIndex = 0;

int charValue = 0;

long pixelElementIndex = 0;

int zeros = 0;

int R = 0, G = 0, B = 0;

for (int i = 0; i < bmp.Height; i++)

{

for (int j = 0; j < bmp.Width; j++)

{

Color pixel = bmp.GetPixel(j, i);

R = pixel.R - pixel.R % 2;

G = pixel.G - pixel.G % 2;

B = pixel.B - pixel.B % 2;

for (int n = 0; n < 3; n++)

{

if (pixelElementIndex % 8 == 0)

{

if (state == State.FillingWithZeros && zeros == 8)

{

if ((pixelElementIndex - 1) % 3 < 2)

{

bmp.SetPixel(j, i, Color.FromArgb(R, G, B));

}

return bmp;

}

if (charIndex >= text.Length)

{

state = State.FillingWithZeros;

}

else

{

charValue = text[charIndex++];

}

}

switch (pixelElementIndex % 3)

{

case 0:

{

if (state == State.Hiding)

{

R += charValue % 2;

charValue /= 2;

}

}

break;

case 1:

{

if (state == State.Hiding)

{

G += charValue % 2;

charValue /= 2;

}

}

break;

case 2:

{

if (state == State.Hiding)

{

B += charValue % 2;

charValue /= 2;

}

bmp.SetPixel(j, i, Color.FromArgb(R, G, B));

}

break;

}

pixelElementIndex++;

if (state == State.FillingWithZeros)

{

zeros++;

}

}

}

}

return bmp;

Листинг 2.1 – Метод шифрования текста в изображении

public static string ExtractText(Bitmap bmp)

{

int colorUnitIndex = 0;

int charValue = 0;

string extractedText = "";

for (int i = 0; i < bmp.Height; i++)

{

for (int j = 0; j < bmp.Width; j++)

{

Color pixel = bmp.GetPixel(j, i);

for (int n = 0; n < 3; n++)

{

switch (colorUnitIndex % 3)

{

case 0:

{

charValue = charValue \* 2 + pixel.R % 2;

}

break;

case 1:

{

charValue = charValue \* 2 + pixel.G % 2;

}

break;

case 2:

{

charValue = charValue \* 2 + pixel.B % 2;

}

break;

}

colorUnitIndex++;

if (colorUnitIndex % 8 == 0)

{

charValue = ReverseBits(charValue);

if (charValue == 0)

{

return extractedText;

}

char c = (char)charValue;

extractedText += c.ToString();

}

}

}

}

return extractedText;

Листинг 2.2 – Метод извлечения зашифрованного текста из изображения

1. **Результаты работы приложения**

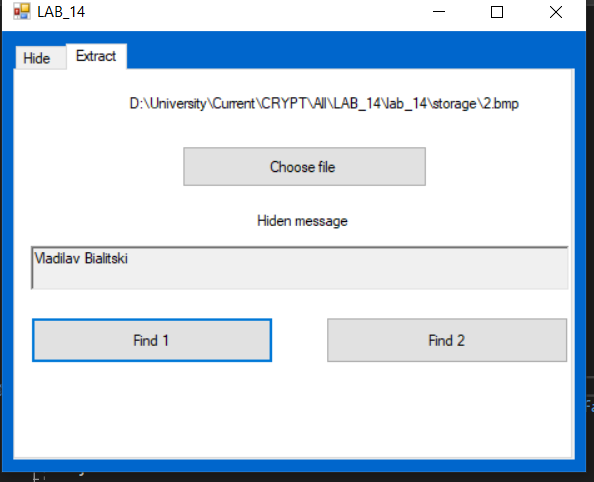


Рисунок 3.1 – результат работы приложения



Рисунок 3.2 – контейнер с без сообщения



Рисунок 3.2 – контейнер с сообщением LSB



Рисунок 3.3 – контейнер с сообщением (через один пиксель)

1. **Визуальный анализ**

Визуальный анализ показывает, что при внедрении сообщения двумя способами, заметить отличия между картинками практически невозможно. Рекомендуется использовать специализированное программное обеспечение.

**Вывод**

В ходе лабораторной работы были приобретены практические навыки программной реализации стеганографического метода извлечения информации с использованием электронного файла-контейнера (bmp) на основе преобразования наименее значащих бит (НЗБ).