Министерство образования Республики Беларусь

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет информационных технологий

Кафедра информатики и веб-дизайна

Отчет к лабораторной работе № 11:

«ИССЛЕДОВАНИЕ СТЕГАНОГРАФИЧЕСКОГО МЕТОДА НА ОСНОВЕ ПРЕОБРАЗОВАНИЯ НАИМЕНЕЕ ЗНАЧАЩИХ БИТОВ»

Выполнила:

студентка 3 курса 12 группы

Шелех А. А.

Вариант: 14

Преподаватель:

Ржеутская Н.В.

Минск 2022

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

**Задачи:**

* Закрепить теоретические знания из области стеганографического преобразования информации, моделирования стеганосистем, классификации и сущности методов цифровой стеганографии.
* Изучить алгоритм встраивания/извлечения тайной информации на основе метода НЗБ (LSB – Least Significant Bit), получить опыт практической реализации метода.
* Разработать приложение для реализации алгоритма встраивания/извлечения тайной информации с использованием электронного файла-контейнера на основе метода НЗБ.
* Познакомиться с методиками оценки стеганографической стойкости метода НЗБ.
* Результатывыполнения лабораторной работыоформить в виде описания разработанного приложения, методики выполнения экспериментов с использованием приложения и результатов эксперимента.

**1 Теоретические сведения**

Определение 1. Стеганографическая система (stegosystem, стегосистема или стеганосистема – в русскоязычной тематической литературе используются оба сокращения) – совокупность средств и методов, которые используются для формирования скрытого канала передачи (или хранения) информации.

При этом скрытый канал организуется на базе и внутри открытого канала с использованием особенностей восприятия информации. «Скрытость» канала передачи тайной информации отличает стеганографию от криптографии: в первом случае тайной является сам факт наличия канала (передачи информации).

Определение 2. Абстрактно стеганографическая система обычно определяется как некоторое множество отображений одного пространства (множества возможных сообщений М) в другое пространство (множество возможных стеганосообщений S), и наоборот.

Определение 3. Стеганографической системой ∑ будем называть совокупность сообщений M, контейнеров C, ключей K, стеганосообщений (заполненных контейнеров) S и преобразований (прямого F и обратного F–1), которые их связывают:

∑ = (M, C, K, S, F, F–1).

Как видим, сущностью рассматриваемой системы является тайное хранение или передача одной информации в другой информации, которая является открытой.

Большинство исследований в предметной области посвящено использованию в качестве стеганоконтейнеров изображений (текст также можно рассматривать как изображение). Это обусловлено следующими причинами:

• относительно большим объемом цифрового представления изображений, что позволяет внедрять большой объем данных;

• заранее известным размером контейнера, отсутствием ограничений, накладываемых требованиями реального времени;

• наличием в большинстве реальных изображений текстурных областей, имеющих шумовую структуру и хорошо подходящих для встраивания информации;

• слабой чувствительностью человеческого глаза к незначительным изменениям цветов изображения, его яркости, контрастности, содержанию в нем шума, искажениям вблизи контуров;

• хорошо разработанными в последнее время методами цифровой обработки изображений.

2 Практическая часть

В лабораторной было представлено приложение, в котором реализован метод НЗБ, при этом реализовано два варианта осаждаемого сообщения, сформированные цифровые матрицы.

Основные функции программы представлены на рисунке 2.2.

private void btnWriteTextInPic\_Click(object sender, EventArgs e)

{

String Message\_Binary = "";

Int32 k = 0;

Color color;

Int32 red;

Int32 green;

Int32 blue;

String Red\_Binary;

String Green\_Binary;

String Blue\_Binary;

foreach (char c in tbInputText.Text)

{

Message\_Binary = Message\_Binary + (Convert.ToString(c, 2).PadLeft(13, '0'));

}

bmpCoded = new Bitmap(bmpOriginal);

Int32 Text\_Len = tbInputText.Text.Length;

String Text\_Len\_binary = Convert.ToString(Text\_Len, 2).PadLeft(bmpOriginal.Width, '0');

for (int i = 0; i < bmpCoded.Width ; i++)

{

color = bmpCoded.GetPixel(i, 0);

Red\_Binary = Convert.ToString(color.R, 2).PadLeft(8, '0');

Red\_Binary = ReplaceLastBitOfByte(Red\_Binary, Text\_Len\_binary[i]);

red = Convert.ToInt32(Red\_Binary, 2);

bmpCoded.SetPixel(i, 0, Color.FromArgb(red, color.G, color.B));

}

for (int i = 0; i < bmpCoded.Width && k < Message\_Binary.Length; i++)

{

for (int j = 1; j < bmpCoded.Height && k < Message\_Binary.Length; j++)

{

color = bmpCoded.GetPixel(i, j);

Red\_Binary = Convert.ToString(color.R, 2).PadLeft(8, '0');

Red\_Binary = ReplaceLastBitOfByte(Red\_Binary, Message\_Binary[k]);

red = Convert.ToInt32(Red\_Binary, 2);

k++;

if (k >= Message\_Binary.Length)

{

bmpCoded.SetPixel(i, j, Color.FromArgb(red, color.G, color.B));

break;

}

Green\_Binary = Convert.ToString(color.G, 2).PadLeft(8, '0');

Green\_Binary = ReplaceLastBitOfByte(Green\_Binary, Message\_Binary[k]);

green = Convert.ToInt32(Green\_Binary, 2);

k++;

if (k >= Message\_Binary.Length)

{

bmpCoded.SetPixel(i, j, Color.FromArgb(red, green, color.B));

break;

}

Blue\_Binary = Convert.ToString(color.B, 2).PadLeft(8, '0');

Blue\_Binary = ReplaceLastBitOfByte(Blue\_Binary, Message\_Binary[k]);

blue = Convert.ToInt32(Blue\_Binary, 2);

k++;

bmpCoded.SetPixel(i, j, Color.FromArgb(red, green, blue));

}

}

int jj = bmpCoded.Height - 1;

for (int i = 0; i < bmpCoded.Width ; i++)

{

color = bmpCoded.GetPixel(i, jj);

Green\_Binary = Convert.ToString(color.G, 2).PadLeft(8, '0');

if(i%2 == 0)

{

Green\_Binary = ReplaceLastBitOfByte(Green\_Binary, '0');

}

else

{

Green\_Binary = ReplaceLastBitOfByte(Green\_Binary, '1');

}

green = Convert.ToInt32(Green\_Binary, 2);

bmpCoded.SetPixel(i, jj, Color.FromArgb(color.R, green, color.B));

}

pbCodedPic.Image = bmpCoded;

for (int y = 0; y < bmpCoded.Height; y++)

{

for (int x = 0; x < bmpCoded.Width; x++)

{

redArr[x, y] = Convert.ToString(bmpCoded.GetPixel(x, y).R, 2).PadLeft(8, '0');

greenArr[x, y] = Convert.ToString(bmpCoded.GetPixel(x, y).G, 2).PadLeft(8, '0');

blueArr[x, y] = Convert.ToString(bmpCoded.GetPixel(x, y).B, 2).PadLeft(8, '0');

}

}

Bitmap colorBitmap = new Bitmap(bmpCoded.Width, bmpCoded.Height);

for (int y = 0; y < bmpCoded.Height; y++)

{

for (int x = 0; x < bmpCoded.Width; x++)

{

colorBitmap.SetPixel(x, y, Color.FromArgb(Convert.ToInt32(redArr[x, y][7].ToString()) \* 255, Convert.ToInt32(greenArr[x, y][7].ToString()) \* 255, Convert.ToInt32(blueArr[x, y][7].ToString()) \* 255));

}

}

colorImage2.Image = colorBitmap;

}

Вывод

В ходе выполнения лабораторной работы были изучены стенографического метода встраивания/извлечения информации с использованием электронного контейнера.