МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждение образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Отчет по лабораторной работе №12

Исследование стеганографического метода на основе

преобразования наименее значащих битов

Выполнил студент: Плюто Э. В.

ФИТ 3 курса, 5 группа

Проверил: Савельева М. Г.

Минск 2024

**Практическое задание:**

**1)** Разработать авторское приложение, которое должно реализовывать метод НЗБ.

Реализация функции для сокрытия сообщения в младших битах уровня красного представлена на листинге 1.

|  |
| --- |
| public static string LSB(string texts)  {  byte[] text = Encoding.UTF8.GetBytes(texts);  int charIndex = 0;  int bitIndex = 0;  for (int y = 0; y < image.Height; y++)  { for (int x = 0; x < image.Width; x++) {  if (charIndex < text.Length)  {  Color pixel = image.GetPixel(x, y);  byte Byte = text[charIndex];  int Bit = (Byte >> (7 - bitIndex)) & 0x01;  pixel = Color.FromArgb((pixel.R & 0xFE) | Bit,  pixel.G,  pixel.B);  image.SetPixel(x, y, pixel);  bitIndex++;  if (bitIndex >= 8)  {  bitIndex = 0;  charIndex++;  }}}}  image.Save("C:\\Users\\Erik\\Desktop\\3course\\IB\\LABA13\\LABA13\\LABA13\\" + texts.Substring(0, 5) + ".bmp");  return "C:\\Users\\Erik\\Desktop\\3course\\IB\\LABA13\\LABA13\\LABA13\\" + texts.Substring(0, 5) + ".bmp";  } |

Листинг 1 – Реализация сокрытия сообщения

Реализация функции для извлечения сокрытого сообщения из файла-контейнера представлена на листинге 2.

|  |
| --- |
| public static string GetMessageLSB(Bitmap image, int length)  {  byte[] hiddenMessage = new byte[length];  int bitIndex = 0;  byte Byte = 0;  int charIndex = 0;  for (int y = 0; y < image.Height; y++)  {  for (int x = 0; x < image.Width; x++)  {  Color pixel = image.GetPixel(x, y);  Byte = (byte)((Byte << 1) | pixel.R & 0x01);  bitIndex++;  if (bitIndex >= 8)  {  hiddenMessage[charIndex] = Byte;  Byte = 0;  bitIndex = 0;  charIndex++;  if (charIndex >= length) break;  }  }  if (charIndex >= length)  break;  }  return Encoding.UTF8.GetString(hiddenMessage);  } |

Листинг 2 – Реализация извлечения сообщения

Результат выполнения вышеописанных функций представлен на рисунке 1.

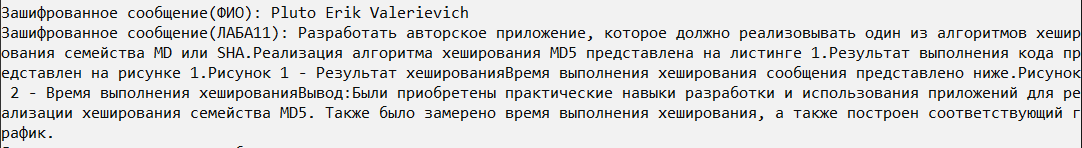


Рисунок 1 – Результат извлечения сообщения

Исходный файл-контейнер (слева) и файл с сообщением (справа) представлены ниже. При визуальном осмотре исходного файла и файла со скрытым сообщением разница не была замечена.



Рисунок 2 – Результат сокрытия сообщения

Реализация функции для создания цветовой матрицы представлена на листинге 3.

|  |
| --- |
| public static void ColorMatrix(Bitmap image, string fileName)  {  Bitmap colorMatrix = new Bitmap(image.Width, image.Height);  for (int y = 0; y < image.Height; y++)  {  for (int x = 0; x < image.Width; x++)  {  Color pixel = image.GetPixel(x, y);  Color matrixPixel = BitColor(GetBit(pixel.R, pixel.G, pixel.B, 5));  colorMatrix.SetPixel(x, y, matrixPixel);  }  } colorMatrix.Save("C:\\Users\\Erik\\Desktop\\3course\\IB\\LABA13\\LABA13\\LABA13\\" + fileName + ".bmp");  } |

Листинг 3 – Функция создания цветовой матрицы

Результат выполнения представлен на рисунке 3.

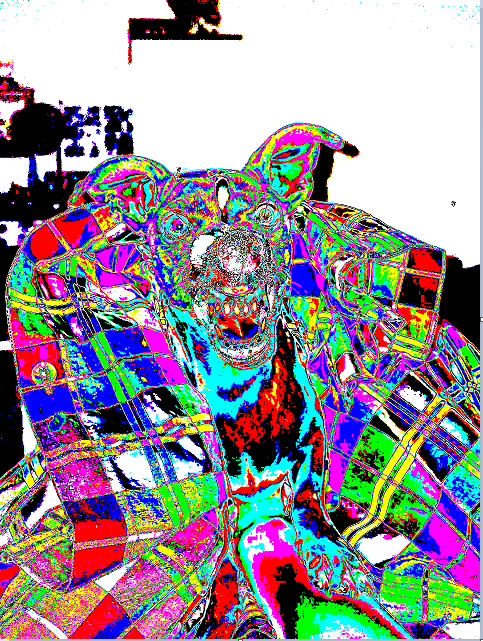


Рисунок 3 – Цветовая матрица

Реализация функции для сокрытия сообщения в младших битах красного, зеленого и синего уровней представлена на листинге 4.

|  |
| --- |
| public static string LSB3(string texts)  {  byte[] text = Encoding.UTF8.GetBytes(texts);  int charIndex = 0;  int bitIndex = 0;  for (int y = 0; y < image.Height; y++)  {  for (int x = 0; x < image.Width; x++)  {  if (charIndex < text.Length)  {  Color pixel = image.GetPixel(x, y);  byte Byte = text[charIndex];  int redBit = (Byte >> (7 - bitIndex)) & 1;  int greenBit = (Byte >> (6 - bitIndex)) & 1;  int blueBit = (Byte >> (5 - bitIndex)) & 1;  //Console.WriteLine(pixel.R + " - " + ((pixel.R & 0xFE) | redBit));  pixel = Color.FromArgb(  (pixel.R & 0xFE) | redBit,  (pixel.G & 0xFE) | greenBit,  (pixel.B & 0xFE) | blueBit  );  image.SetPixel(x, y, pixel);  bitIndex++;  if (bitIndex >= 8)  {  bitIndex = 0;  charIndex++;  }  }  }  }  image.Save("C:\\Users\\Erik\\Desktop\\3course\\IB\\LABA13\\LABA13\\LABA13\\" + texts.Substring(0, 5) + "3.bmp");  return "C:\\Users\\Erik\\Desktop\\3course\\IB\\LABA13\\LABA13\\LABA13\\" + texts.Substring(0, 5) + "3.bmp";  } |

Листинг 4 – Реализация сокрытия сообщения

Реализация функции для извлечения сокрытого сообщения из файла-контейнера представлена на листинге 5.

|  |
| --- |
| public static string GetMessageLSB3(Bitmap image, int length)  {  byte[] hiddenMessage = new byte[length];  int bitIndex = 0;  byte Byte = 0;  int charIndex = 0;  int i = 0;  for (int y = 0; y < image.Height; y++)  {  for (int x = 0; x < image.Width; x++)  {  Color pixel = image.GetPixel(x, y);  if (i == 0)  Byte = (byte)((Byte << 1) | pixel.R & 0x01);  if (i == 1)  Byte = (byte)((Byte << 2) | pixel.G & 0x01);  if (i == 2)  Byte = (byte)((Byte << 3) | pixel.B & 0x01);  Else i = 0;  bitIndex++;  if (bitIndex >= 8)  { hiddenMessage[charIndex] = Byte;  Byte = 0;  bitIndex = 0;  charIndex++;  if (charIndex >= length)  break;}}  if (charIndex >= length)  break; }  return Encoding.UTF8.GetString(hiddenMessage);} |

Листинг 5 – Реализация извлечения сообщения

Результат выполнения вышеописанных функций представлен на рисунке 1.

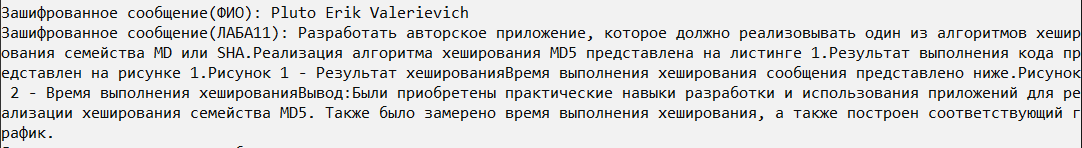


Рисунок 4 – Результат извлечения сообщения

Исходный файл-контейнер (слева) и файл с сообщением (справа) представлены ниже. Как и прошлом случае визуально полученный и исходный файл не отличаются.



Рисунок 5 – Результат сокрытия сообщения

Цветовая матрица полученного файла представлена на рисунке 6.

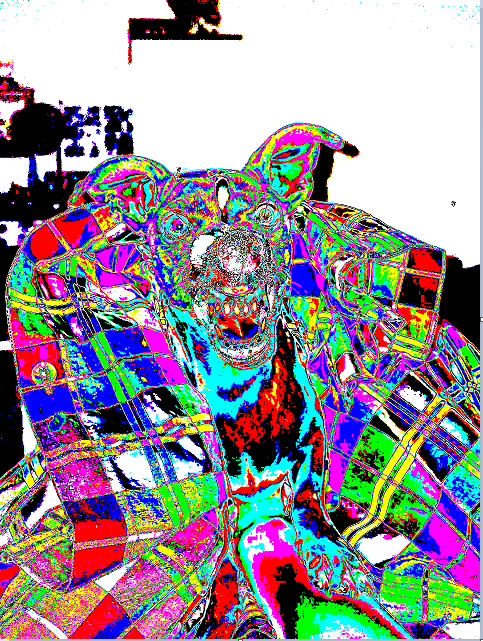


Рисунок 6 – Цветовая матрица

Цветовая матрица также аналогична предыдущей.

**Вывод:**

Были приобретены практические навыки разработки и использования приложений для стеганографического метода встраивания и извлечения тайной информации с использованием электронного файла-контейнера на основе преобразования наименее значащих битов (НЗБ). Был реализован функционал для сокрытия и извлечения тайного сообщения с помощью одного цвета и трех цветов, отображения цветовой матрицы файла.