Правительство Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ

ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ

«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»

(НИУ ВШЭ)

Московский институт электроники и математики им. А.Н. Тихонова

ОТЧЕТ

О ПРАКТИЧЕСКОЙ РАБОТЕ № 4

по дисциплине «Проектный семинар по информационной безопасности»

«Стеганографическое встраивание информации в пространственную область изображений»

|  |
| --- |
| Студент гр. БИБ222 Малыгин Данила Дмитриевич «07» мая 2023 г. |
| Руководитель Заведующий кафедрой информационной безопасности киберфизических систем  канд. техн. наук, доцент  \_\_\_\_\_\_\_\_\_\_\_\_\_\_О. О. Евсютин  «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2023 г. |

Москва 2023

**СОДЕРЖАНИЕ**

[1 Задание на практическую работу …………………………………………………………… 3](#_Toc134283234)

[2 Краткая теоретическая часть ………………………………………………………………... 3](#_Toc134283235)

[3 Программная реализация ……………………………………………………………………. 4](#_Toc134283236)

[4 Вычислительные эксперименты …………………………………………………………….. 5](#_Toc134283237)

[5 Выводы о проделанной работе ……………………………………………………………… 8](#_Toc134283238)

[7 Список использованных источников ……………………………………………………… 10](#_Toc134283239)

8 Ссылка на репозиторий с программами …………………………………………………... 10

# 1 Задание на практическую работу

Целью данной работы является приобретение навыков программной реализации стеганографического встраивания информации в цифровые изображения.

В рамках практической работы необходимо выполнить следующее:

1) написать программную реализацию одного из следующих стеганографических методов для цифровых изображений: QIM, PVD, NMI.

2) провести вычислительные эксперименты с полученной программной реализацией и сделать выводы об эффективности рассмотренного метода встраивания.

3) подготовить отчет о выполнении работы.

# 2 Краткая теоретическая часть

Стеганография – наука, изучающая тему встраивания информации в разные цифровые объекты, с целью скрытной передачи их третьим лицам так, чтобы сам факт передачи был незаметен. Существуют два принципа, на которых базируется компьютерная стеганография: первый из них – цифровые носители данных, такие как фотография, звук или видео, могут в некой степени видоизменяться без потери их функциональности, второй – неспособность органов чувств человека распознать незначительные изменения цифровых данных.

Контейнер – некий цифровой объект, нужный для встраивания сообщения. В качестве сообщения может использоваться любой вид информации: текст, изображение, аудиосигнал. Одним из наиболее распространенных типов контейнеров для стеганографического встраивания являются цифровые изображения. Также неоспоримым плюсом изображений в роли контейнера является их пространственная избыточность, которая состоит в том, что соседние пиксели имеют примерно одинаковые параметры цвета, то есть сильно похожи друг на друга. Сильнее всего это свойство изображений видно на не текстурированных изображениях.

Существует большое количество методов встраивания вплоть до машинного обучения, которые используют разные свойства изображений. Всех их можно разделить на две группы: методы пространственного встраивания и методы частотного встраивания. Методы пространственного встраивания напрямую изменяют значения пикселей изображения, в то время как методы частотного встраивания вносят изменения в значения частотных коэффициентов, полученных после применения некоторого частотного преобразования к матрице пикселей.

В данной работе будет использоваться метод встраивания PVD. PVD (Pixel Value Difference) – метод разности значений пикселей, один из способов встраивания информации, представленной в двоичной последовательности в изображение. Этот метод активно использует свойство пространственной избыточности изображений при вычислении количества бит встраиваемой информации в пару пикселей.

В процессе встраивания картинка разбивается на пары не пересекающихся между собой пикселей (Pi , Pi + 1). Далее для каждой пары считается их разность: , .

Абсолютное значение разности пары пикселей попадает в один из шести отрезков вида . Эти отрезки заданы заранее и обычно определяются следующим образом: , , , , , . Таким образом, определяется .

Количество битов сообщения, которое может быть встроено в пару пикселей , определяется по формуле (1):

|  |  |
| --- | --- |
|  | (1) |

где nk – количество бит встраиваемой информации в данную пару пикселей, uk и lk – границы отрезка, которые были определены ранее.

Далее фрагмент сообщения длиной битов представляется в виде целого числа и вычисляется новое значение разности для данной пары пикселей по формуле (2):

|  |  |
| --- | --- |
|  | (2) |

Встраивание фрагмента сообщения в пару пикселей осуществляется по формуле (3):

|  |  |
| --- | --- |
|  | (3) |

Извлечение сообщения выполняется по формуле (4):

(4)

# 3 Программная реализация

Данный метод стеганографии был реализован на языке Python с использованием встроенных библиотек Numpy, math, Pillow для удобной работы с изображениями. Возьмем изображение и встроим в него дважды написанный английский алфавит, то есть 52 символа. На рисунке 1 представлено изображение до встраивания, на рисунке 2 – изображение после встраивания:



Изображение 1 – картинка до встраивания.

Изображение выглядит как диаграмма

Автоматически созданное описание

Изображение 2 – картинка после встраивания.

# 4 Вычислительные эксперименты

В целях оценки встраивания проведем вычислительные эксперименты. Наиболее распространенными характеристиками встраивания являются незаметность и емкость.

С целью проверки незаметности посчитаем PSNR, MSE, RMSE, SSIM для изображения 1 и изображения 2. С этой целью была написана программа, которая считает эти показатели на основе значений пикселей двух изображений. Результаты работы представлены на рисунке 3:



Рисунок 3 – метрики встраивания.

Емкость (EC) составила 1,553 бита на один пиксель изображения, что довольно-таки неплохо. SSIM – индекс схожести двух изображений, этот коэффициент всегда лежит в пределах от -1 до 1. Чем выше этот коэффициент, тем более схожи два изображения, в нашем случае видно, что схожесть на высоком уровне и почти равна 1. PSNR наиболее часто используется для измерения уровня искажений изображений. Проще всего его определить через среднеквадратичную ошибку MSE. RMSE используется в случае двух монохромных изображений, то есть в нашем случае. Для используемого изображения 1 максимальное количество битов составляет около 500000 битов. Довольно проблематично сгенерировать такой объем информации. С целью упрощения исходное изображение 1 было сжато до размеров . В этом случае максимальное количество символов, которые кодируются 8 битами, составляет 1013 символов. Внедрим такое количество символов английского языка в наше изображение Результат метрик представлен на рисунке 4:

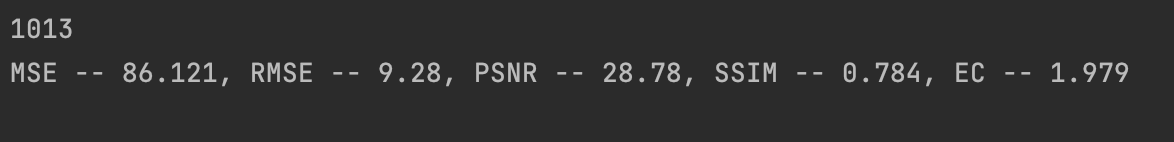
**

Рисунок 4 – метрики встраивания при максимальном объеме встроенной информации.

Теперь в целях экспериментах встроим сообщение меньшего размера чем это и сравним результаты. Метрики встраивания сообщения длиной 100 символов представлены на рисунке 5:



Рисунок 5 – метрики встраивания при встраивании 100 символов.

Видим, что EC осталось без изменений, так как изображение никак не менялось, поэтому и емкость не изменилась. SSIM уменьшается с уменьшением объема встраиваемой информации. Однако RMSE имеет обратную зависимость с объемом встроенной информации. PSNR показывает, что уровень искажения растет при меньшем объеме встраивания. Таким образом, рост объема встраивания стимулирует уменьшение уровня искажения изображения.

В итоге можно сделать вывод, что с увеличением длины сообщения растет незаметность встраивания. Емкость напрямую зависит от размера сообщения, так для одного и того же изображения, только разных размеров емкость возрастает в 500 раз, при изменении размера в 8 раз.

При извлечении информации ошибок не замечено. Проведем эксперимент: встроим в изображение размером сообщение «stegosystem and conteiner». Результаты извлечения видны на рисунке 6:

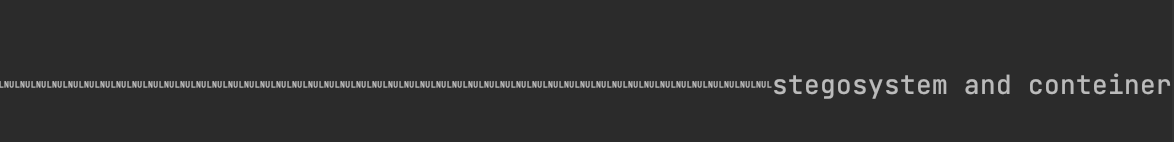


Рисунок 6 – результаты извлечения информации из изображения .

Перед нашим введенным сообщением видны неизвестные знаки. Это объясняется тем, что при извлечении информации в случае применения метода PVD мы не знаем длину сообщения заранее, поэтому приходится проходить по всем парам пикселей, даже по тем, которые не были задействованы при встраивании.

Теперь сравним гистограммы изображения до и после внедрения информации. Гистограммы представлены на рисунке 7 и рисунке 8:

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 7 – гистограмма исходного изображения.

Изображение выглядит как диаграмма

Автоматически созданное описание

Рисунок 8 – гистограмма изображения со встроенной информацией.

Видим, что встраивание не сильно повлияло на распределение пиковых значений частоты определенных пикселей. В основном изменения видны в пикселях с близкими к нулю значениями, на рисунке 8 в этой области образовался пик, которого не было на рисунке 7. Можно сделать вывод, что в этом случае встраивания больше всего подверглись изменениям белые участки изображения, так как именно этот цвет кодируется близким к 0 значениям.

# 5 Выводы о проделанной работе

Таким образом можно сделать следующий вывод: стеганография является неплохим методом сокрытия информации. Чтобы раскрыть стеганографическое изображение, необходимо знать точный метод, которым оно было встроено, что довольно-таки сложно, так как существует большое количество методов внедрения информации в цифровые контейнеры.

Метод PVD является одним из них. Данный метод является хорошей альтернативой методу LSB. Главной отличительной особенностью метода PVD от LSB является его подстраивание под изображение, то есть учитывание гладкости и текстурированности изображения. PVD учитывает границы объектов на изображении и встраивает биты информации именно туда, посредством регулирования количества битов для встраивания в определенную область. Тем самым увеличивается контраст на границах объектов, а гладкие области без перепадов контраста остаются менее тронутыми.

# 7 Список использованных источников

1. Пиковое отношение сигнала к шуму // Wikipedia -- URL: <https://ru.wikipedia.org/wiki/%D0%9F%D0%B8%D0%BA%D0%BE%D0%B2%D0%BE%D0%B5_%D0%BE%D1%82%D0%BD%D0%BE%D1%88%D0%B5%D0%BD%D0%B8%D0%B5_%D1%81%D0%B8%D0%B3%D0%BD%D0%B0%D0%BB%D0%B0_%D0%BA_%D1%88%D1%83%D0%BC%D1%83>

2. SSIM // Wikipedia -- URL: <https://ru.wikipedia.org/wiki/SSIM>

3. Wu D.-C. A steganographic method for images by pixel-value differencing / D.-C. Wu, W.-H. Tsai // Pattern Recognition Letters. – 2003. – Vol. 24, № 9-10. – P. 1613–1626.

4. Атрибуты и методы объекта Image модуля Pillow. // docs-python.ru -- URL: https://docs-python.ru/packages/biblioteka-pillow-python/obekt-image/

# 8. Ссылка на репозиторий с программами

# [https://github.com/fedyarays/crypto\_pr4](https://github.com/fedyarays/crypto_pr43)