Академия логистики и транспорта

УДК На правах рукописи

**ДАЙЫРБАЕВА ЭЛЬМИРА НУРБЕККЫЗЫ**

**Разработка и исследование стеганографических алгоритмов, ориентированных на внедрение скрытой информации в изображения**

8D06254 –Радиотехника, электроника и телекоммуникации

Диссертация на соискание степени

доктора философии (PhD)

Научный консультант:

PhD, ассоц. проф.

Тойгожинова А.Ж.

PhD, ассоц. проф.

Еримбетова А.С.

Зарубежный научный консультант:

д.т.н., доцент

Нечта И.В.

(г. Новосибирск, РФ)

Алматы -2023

**СОДЕРЖАНИЕ**

[**НОРМАТИВНЫЕ ССЫЛКИ** 3](#_Toc137145106)

[**ОПРЕДЕЛЕНИЯ** 4](#_Toc137145107)

[**ОБОЗНАЧЕНИЯ И СООКРАЩЕНИЯ** 6](#_Toc137145108)

[**ВВЕДЕНИЕ** 7](#_Toc137145109)

[1 МЕТОДЫ СТЕГАНОГРАФИИ И СТЕГОНАЛИЗА 15](#_Toc137145110)

[1.1 Обзор предметной области 15](#_Toc137145111)

[1.2 Обзор существующих методов внедрения данных в изображения 18](#_Toc137145112)

[1.2.1 LSB метод 20](#_Toc137145113)

[1.2.2 Алгоритм дискретного косинусного преобразования (DCT) 25](#_Toc137145114)

[1.2.3 Вейвлет –преобразование 26](#_Toc137145115)

[1.3 Практическое применение методов стеганографии 28](#_Toc137145116)

[1.4 Обзор известных методов стегоанализа 33](#_Toc137145117)

[1.4.1 RS – метод 35](#_Toc137145118)

[1.3.2 SPAM метод 37](#_Toc137145119)

[1.3.3 Метод Хи-квадрат 37](#_Toc137145120)

[1.5 Выводы по разделу 40](#_Toc137145121)

[2 ИНТЕРПОЛЯЦИЯ ИЗОБРАЖЕНИЯ В СТЕГАНОГРАФИИ 42](#_Toc137145122)

[2.1 Анализ задачи 42](#_Toc137145123)

[2.2 Понятие интерполяции 43](#_Toc137145124)

[2.3. Общие сведения об интерполяции Лагранжа 45](#_Toc137145125)

[2.4 Метод NMI 46](#_Toc137145126)

[2.5 Метод INMI 49](#_Toc137145127)

[2.6 Стегоанализ INMI метода 55](#_Toc137145128)

[2.7 Выводы по разделу 61](#_Toc137145129)

[3 ПРИМЕНЕНИЕ КРИВЫХ БЕЗЬЕ В СТЕГАНОГРАФИИ 62](#_Toc137145130)

[3.1 Кривая Безье 62](#_Toc137145131)

[3.2 Анализ существующих методов, базирующихся на кривой Безье 66](#_Toc137145132)

[3.3 Описание предлагаемого метода встраивания LIBC5 72](#_Toc137145133)

[3.4 Стегоанализ нового метода LIBC5 79](#_Toc137145134)

[3.5 Выводы по разделу 81](#_Toc137145135)

[4 ОПИСАНИЕ ПРОГРАММНЫХ СРЕДСТВ 82](#_Toc137145136)

[**ЗАКЛЮЧЕНИЕ** 93](#_Toc137145137)

[**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** 95](#_Toc137145138)

[ПРИЛОЖЕНИЕ А АВТОРСКОЕ СВИДЕТЕЛЬСТВО (3 шт.) 107](#_Toc137145139)

[ПРИЛОЖЕНИЕ В АКТ ВНЕДРЕНИЯ 110](#_Toc137145140)

# **НОРМАТИВНЫЕ ССЫЛКИ**

В настоящей диссертации использованы нормативные ссылки на следующие стандарты:

Утверждены приказом МОН РК от 31 марта 2011 года №127 Правила присвоения степеней.

ГОСТ 7.32-2017 Система стандартов по информации, библиотечная и издательская работа. Отчет о научно-исследовательской работе. Структура и правила оформления.

ГОСТ 8.417-2002 государственная система обеспечения единства измерений. Единицы величины.

РИ-АЛТ-36 Рабочая инструкция «Положение о диссертационном совете».

РИ-АЛТ-67 «Порядок оформления и написания докторской диссертации».

# **ОПРЕДЕЛЕНИЯ**

В настоящем диссертации использованы следующие термины с соответствующими определениями:

Boosting - (усиление) [ансамблевый](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D1%81%D0%B0%D0%BC%D0%B1%D0%BB%D0%B5%D0%B2%D0%BE%D0%B5_%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5) метод [машинного обучения](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D0%BE%D0%B5_%D0%BE%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5), применяется главным образом для уменьшения [смещения](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%BB%D0%B5%D0%BC%D0%BC%D0%B0_%D1%81%D0%BC%D0%B5%D1%89%D0%B5%D0%BD%D0%B8%D1%8F%E2%80%93%D0%B4%D0%B8%D1%81%D0%BF%D0%B5%D1%80%D1%81%D0%B8%D0%B8) (погрешности оценки), а также дисперсии в [*обучении с учителем*](https://ru.wikipedia.org/wiki/%D0%9E%D0%B1%D1%83%D1%87%D0%B5%D0%BD%D0%B8%D0%B5_%D1%81_%D1%83%D1%87%D0%B8%D1%82%D0%B5%D0%BB%D0%B5%D0%BC).

DCT - это один из наиболее распространенных методов внедрения информации в изображения, использующий коэффициенты ряда Фурье в jpg файле.

RS – метод стегоанализа, основаный на применении двойной статистики, полученной из пространственных корреляций в изображениях, автор J. Fridrich.

SPAM (Statistic-based Pixel Adjacency Model) - это метод стегоанализа, который использует статистическую модель соседних пикселей для обнаружения скрытых сообщений в мультимедийных данных.

Вейвлет (от wavelet - малая волна, пульсация, также всплеск, часто - вейвлет) - математическая функция, которая анализирует различные частотные составляющие данных.

Встроенное (скрытое) сообщение – это сообщение, встроенное в контейнер.

Гауссовский шум — это [статистический шум](https://ru.wikipedia.org/w/index.php?title=%D0%A1%D1%82%D0%B0%D1%82%D0%B8%D1%81%D1%82%D0%B8%D1%87%D0%B5%D1%81%D0%BA%D0%B8%D0%B9_%D1%88%D1%83%D0%BC&action=edit&redlink=1), имеющий [плотность вероятности](https://ru.wikipedia.org/wiki/%D0%9F%D0%BB%D0%BE%D1%82%D0%BD%D0%BE%D1%81%D1%82%D1%8C_%D0%B2%D0%B5%D1%80%D0%BE%D1%8F%D1%82%D0%BD%D0%BE%D1%81%D1%82%D0%B8), равную плотности вероятности [нормального распределения](https://ru.wikipedia.org/wiki/%D0%9D%D0%BE%D1%80%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5), также известного как [Гауссовское](https://ru.wikipedia.org/wiki/%D0%93%D0%B0%D1%83%D1%81%D1%81%D0%BE%D0%B2%D1%81%D0%BA%D0%BE%D0%B5_%D1%80%D0%B0%D1%81%D0%BF%D1%80%D0%B5%D0%B4%D0%B5%D0%BB%D0%B5%D0%BD%D0%B8%D0%B5).

Голограмма – объёмное изображение, воспроизведённое интерференцией волн с некоторой поверхности.

Заполненный контейнер – контейнер, содержащий секретное сообщение.

Интерполяция — в вычислительной математике нахождение неизвестных промежуточных значений некоторой функции, по имеющемуся дискретному набору её известных значений, определенным способом.

Контейнер – это безобидный не внешний вид объект данных (файл), предназначенный для хранения внутри себя секретного сообщения.

Ошибка второго рода (β-ошибка, ложноотрицательное заключение) — ситуация, когда принята неверная нулевая гипотеза.

Ошибка первого рода (𝛼-ошибка, ложноположительное заключение) — ситуация, когда отвергнута верная нулевая гипотеза (об отсутствии связи между явлениями или искомого эффекта).

Пустой контейнер - это контейнер без встроенного секретного сообщения.

Сообщение – это любая информация, представленная в двоичном виде и подлежащая скрытой передаче.

Стеганографическая система или стегосистема – это совокупность средств и методов, которые используются для формирования скрытого канала передачи информации.

Стеганография – это наука, изучающая методы передачи секретного сообщения.

Стегоанализ – процесс выявления факта передачи скрытой информации в контейнере.

Стегоконтейнер – то же, что и заполненный контейнер.

Хи-квадрат (χ²) - это статистический метод (критерий Пирсона), который используется для проверки выборки на принадлежность к генеральной совокупности.

Цифровой водяной знак (Digital Watermark) – это внедренное в контейнер сообщение, предназначенное для идентификации автора, пользователя или целостности контейнера.

# **ОБОЗНАЧЕНИЯ И СООКРАЩЕНИЯ**

|  |  |
| --- | --- |
| DCT | Дискретное косинусное преобразование |
| GLM | Gray Level Modification (Изменение уровня серого) |
| INMI | Improved neighbor mean interpolation |
| LIBC5 | Lagrange interpolation Bezier curve 5 |
| LSB | Least Significant Bit (Наименее значимый бит) |
| NMI | Neighbor mean interpolation |
| PSNR | Peak signal-to-noise ratio (Пиковое [отношение сигнала к шуму](https://ru.wikipedia.org/wiki/%D0%9E%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/%D1%88%D1%83%D0%BC)) |
| PVD | Pixel value differencing (Разница в значениях пикселей) |
| RS | Regular–Singular (регулярно-сингулярный) |
| SPAM | Statistic-based Pixel Adjacency Model |
| АЦП | Аналого-цифровое преобразование |
| ЦАП | Цифро – аналоговое преобразование |
| ЦВЗ | Цифровой водяной знак |

# **ВВЕДЕНИЕ**

**Общая характеристика работы.**

Настоящая работа направлена на обеспечение безопасности пользователей телекоммуникационных сети и, в частности, заключается в создании новых и совершенствовании существующих средств защиты информации в глобальной сети Интернет. В диссертации исследуются методы организации тайного канала связи методами стеганографии.

Так в обычный файл, например, фотографию, может быть встроена скрытая информация, которая будет доступна только получателю. Посторонний не только не сможет получить доступ к сообщению, но даже и заподозрить его существование. В качестве такого файла (контейнера) могут выступать изображения, видео-, аудио- и текстовые файлы.

В отличие от криптографии, где необходима устойчивость шифра к взлому, в стеганографии главным требованием является «скрытность» - низкая вероятность обнаружения факта существования тайного канала связи. Дополнительным требованием может быть устойчивость к искажению контейнера третьим лицом, например, для изображения – это масштабирование, добавление шумов и др.

Существует множество методов и алгоритмов, используемых в стеганографии, большинство из них базируются на методах теории вероятности, математической статистики и статистических свойствах контейнера. Основной принцип внедрения – это внесение малых изменений контейнера так, чтобы они были незаметны для третьих лиц. В то же время любое изменение выполняется в соответствии со скрываемым сообщением.

Некоторые из них ориентированы на скрытие информации в определенных типах изображений, например, в изображениях с высоким разрешением или в видео. Другие методы используют разные способы внедрения информации в изображения, например, изменение цветовых компонентов пикселей или использование маскировки.

Одним из ключевых аспектов в разработке стеганографических алгоритмов является их оценка и анализ. Для этого используются различные критерии качества, такие как устойчивость, объём, незаметность (скрытность) и трудоемкость внедрения. Оценка и анализ позволяют определить эффективность алгоритма и убедиться в том, что он может быть использован для скрытия информации в конкретных практических условиях.

Настоящая диссертационная работа посвящена разработке новых и эффективных методов внедрения и обнаружения скрытых данных в изображениях. Так для внедрения успешно используются методы интерполяции, базирующиеся на кривой Безье. Также был проведен анализ существующих методов внедрения и была показана их уязвимость, что является основанием для их дальнейшего совершенствования.

**Актуальность темы.**

В связи с бурным развитием компьютерной техники резко возрос объем передаваемых данных в сети Интернет. Наиболее популярным контентом интернет-сайтов являются: аудио- и видео файлы, изображения и текст. Особенностью цифрового контента является его почти нулевая стоимость воспроизводства. Действительно скопировать файл не составляет большого труда.

В этой связи возникает ряд проблем. Во-первых, должностные лица, имеющие доступ к персональным данным клиентов (паспорта, номера банковских счетов и телефонов, и др.) способны тайно отправлять по электронной почте за пределы организации посторонним лицам. На черном рынке такая информация ценится и используется как для рассылки рекламы, так и для совершения противоправных действий. Во-вторых, разработанная программа или созданная фотография может использоваться третьим лицом, которое выдает эти объекты интеллектуальной деятельности под своим авторством. В-третьих, файл, проданный конкретному пользователю, может быть нелегально скопирован и продаваться как пиратский контент, нанося прямой финансовый урон изготовителю.

Указанные проблемы имеют решения в области стеганографии. Так современные требования по информационной безопасности предприятия вынуждают создавать системы анализа передаваемого трафика. В дополнение анализируется отправляемая почта. Очевидно, что объем трафика огромный и не может быть проанализирован человеком, но может проводиться автоматическом режиме. Так при обнаружении утечек персональных данных выполняется стегоанализ всей отправлявшейся почты организации для установления нарушителя.

Для подтверждения своего права владения над цифровым контентом производитель встраивает в изображение или программу специальное сообщение – цифровой водяной знак, который содержит, например его ФИО. При необходимости этот ЦВЗ может быть извлечен и продемонстрирован в суде, что позволит установить истинное авторство.

Для противодействия пиратскому копированию стеганография предлагает методы цифровых отпечатков пальцев, которые в отличие от ЦВЗ содержат информацию о конечном потребителе. То есть в каждую продаваемую копию встраивается ЦОП, содержащий ФИО потребителя. Если будет перехвачена пиратская версия файла, то достаточно извлечь прочитать ЦОП, что является во многих западных странах основанием для возбуждения следственных действий в отношении нарушителя.

Очевидно, что злоумышленник знает о всех особенностях и возможностях стеганографии. Естественным образом он будет стараться противодействовать им. Например, он может начать преобразовывать изображение (масштабировать, добавлять шум и др.) в надежде на то, что внедренный ЦВЗ или ЦОП (цифровые отпечатки пальцев) будет искажен или полностью уничтожен. Именно по этой причине возникает острая необходимость создания новых и совершенствование и известных методов внедрения. Анализ существующих алгоритмов позволит выявить потенциальную уязвимость стегосистемы и посодействовать дальнейшей эволюции стеганографических методов.

Основными характеристиками методов внедрения в изображение являются: вероятность обнаружения факта внедрения, объем внедрения (вычисляемый от максимально допустимого объема контейнера). Для стегоанализа применяются классические ошибки первого и второго рода: пропуска внедрения и ложного срабатывания.

Исследование стеганографических алгоритмов является важным направлением в области информационной безопасности телекоммуникационных систем. Разработка новых алгоритмов и их исследование позволяют улучшить качество стеганографических методов, сделать их более надежными и защищенными от различных атак.

В настоящее время проводится множество исследований, по проблемам информационной безопасности. Проводятся ежегодные конференции, с каждым годом растет число публикаций, посвященных данной проблеме. В этом направлении науки работают многие казахстанские, российские, и зарубежные ученые: Г.Б. Абдыкеримова, С.Е. Нысанбаева, Н.А. Капалова, И.В. Нечта, И.Н. Оков, Б.Я. Рябко, И.В. Туринцев, А.Н. Фионов, Р. Бергмар (R. Bergmar), К. Кашен (C. Cachin), М. Чапман (M. Chapman), Ц. Чень (J. Chen), Дж. Фридрич (J. Fridrich), и др. Автором диссертации был проведен анализ основных отечественных и зарубежных источников за более чем 10 последних лет. Список этих источников отражен в тексте диссертации. Основные работы, с которыми производилось сопоставление результатов диссертации, принадлежат таким специалистам как К. Джанг (K.Jung), Ц. Чень (J. Chen), Дж. Ю (Z. Yu), Мерзлякова Е.Ю., Евсютин О.

**Данная диссертационная работа** **направлена** на создание научно-технического задела в области информационно-коммуникационных технологий и на получение новых знаний, позволяющих осуществлять анализ и эффективно работать в области стеганографии с применением интерполяции.

**Целью диссертационной работы** является создание новых и эффективных методов внедрения и обнаружения скрытых данных в изображении. Для достижения этой цели необходимо решить следующие **задачи**:

1. Изучение существующих методов стеганографии и анализ их преимуществ и недостатков. Эта задача включает в себя обзор литературы по теме, изучение основных методов стеганографии и оценку их преимуществ и недостатков.
2. Разработка новых стеганографических алгоритмов, ориентированных на внедрение секретной информации в изображения, с учетом современных требований к безопасности и эффективности. Для решения этой задачи необходимо провести теоретический анализ и выбрать оптимальный подход к разработке новых стеганографических алгоритмов, затем реализовать эти алгоритмы и провести их тестирование.
3. Исследование известных алгоритмов на устойчивость к атакам, а также на возможность обнаружения скрытой информации. Для решения этой задачи необходимо провести тестирование разработанных алгоритмов на различных датасетах и оценить их устойчивость к атакам, таким как изменение размера изображения, сжатие, фильтрация и другие. Также необходимо провести анализ возможности обнаружения скрытой информации с помощью стеганализа.
4. Оценка эффективности реализованных алгоритмов по сравнению с существующими методами стеганографии. Для решения этой задачи необходимо провести сравнительный анализ разработанных алгоритмов с существующими методами стеганографии и оценить их эффективность эффективность по устойчивости к атакам.
5. Разработка программной реализации новых стеганографических алгоритмов и создание пользовательского интерфейса. Для решения этой задачи необходимо разработать программную реализацию разработанных алгоритмов, которая будет позволять пользователям внедрять секретную информацию в изображения. Также необходимо разработать удобный и интуитивно понятный пользовательский интерфейс для работы с программой.

**Объектом исследования** являются методы сокрытия информации в контейнерах, представляющие собой цифровые изображения, а также методы выявления наличия скрытой в таких контейнерах информации.

**Предмет исследования** являются методы оценки стойкости стеганографических систем, основанные на перестановках элементов пространства сокрытия, различные характеристики стегоконтейнеров.

**Методы исследования** – в процессе проведения исследований были использованы методы стеганографии и алгоритмы сжатия.

**Новизна работы**. Новизна и оригинальность работы заключается в том, что в ней впервые:

* Проведен обзор существующих, актуальных методов встраивания и обнаружения скрытых данных в изображения. В настоящее время нет исследований, анализирующих устойчивость подобных алгоритмов к анализу;
* Проведен стегоанализ методов INMI и базирующихся на кривой Безье, и выявлены их уязвимости.
* Разработан новый метод встраивания секретных сообщении в изображения используя кривую Безье – LIBC5 (по пяти точкам), являющийся устойчивым к лучшим современным алгоритмам стегоанализа;
* Впервые проведен сравнительный анализ алгоритмов внедрения NMI и INMI, в котором выявлена уязвимость обоих алгоритмов к RS анализу.

**Положения, выносимые на защиту**

1. Метод на основе кривой Безье является устойчивым к анализу RS и сопоставим по стойкости и ёмкости со стегосистемой способа перестановок для растровых изображений;
2. Метод INMI и базирующийся на кривой Безье являются уязвимыми к RS и SPAM анализу.
3. Метода внедрения LIBC5 является устойчивым к RS анализу.
4. Для рассматриваемых INMI и базирующийся на кривой увеличение объема встроенной информации повышает вероятность успешного стегоанализа.
5. Методы NMI и INMI уязвимы к RS анализу.

**Научно-практическая значимость работы.**

Принимая во внимание то, что наиболее распространенным типом файлов, передаваемых в сети Интернет, являются картинки, то настоящее исследование ориентировано на внедрение скрытой информации в изображения. Полученные результаты могут быть использованы в системах скрытой передачи данных по каналам связи. Предложенные алгоритмы стегоанализа могут быть реализованы на аппаратном уровне в телекоммуникационном оборудовании. Предложенные методы внедрения сообщений не увеличивают объем передаваемого трафика в сети.

Имеется акт о внедрении полученных научных результатов диссертационной работы в ТОО «Алматыэлектротранс» (г.Алматы, Казахстан).

**Личный вклад автора** заключается в том, что основные результаты экспериментов были получены соискателем. Постановка задач и обсуждение результатов проводились совместно с научными консультантами.

**В публикациях, выполненных в соавторстве,** вклад автора состоит в анализе источников, постановке эксперимента и представлению результатов.

**Соответствие паспорту образовательной программы 8D06254 –Радиотехника, электроника и телекоммуникации.** Проведенное исследование соответствует области профессиональной деятельности: – средства защиты информации в телекоммуникационных системах. Стеганография исследует методы построения тайных каналов связи на базе существующей Интернет сети.

**Достоверность и обоснованность полученных результатов** подтверждаются наличием публикаций в журналах дальнего зарубежья с импакт-фактором и в изданиях, рекомендованных Комитетом по обеспечению качества в сфере образования и науки МНВО РК, и в трудах международных научных конференции ближнего и дальнего зарубежья. Экспериментальные данные, полученные в процессе выполнения работы хорошо согласуется с данными, полученными на основе нового предложенного метода.

**Апробация диссертационной работы.** Результаты, полученные в диссертационной работе опубликованы, докладывались и обсуждались:

**Публикации с импакт-фактором по базе данных Thomson Reuters или в изданиях, входящих в международную научную базу данных Scopus:**

* **Daiyrbayeva E.,** Yerimbetova A., Nechta I., Merzlyakova E, Toigozhinova A., Turganbayev A. A Study of the Information Embedding Method into Raster Image Based on Interpolation. //J. Imaging. – 2022. –№ 8(288).
* **Daiyrbayeva E.,** Yerimbetova A., Toigozhinova A., Maratov Z., Sambetbayeva M. Learning steganography with a strip transform//6th International Conference on Computer Science and Engineering (UBMK). – Ankara, 2021 –P.209-212.

**Публикации в изданиях, рекомендованных КОКСОНВО РК:**

* **Дайырбаева Э.Н.,** Мурзин Ф.А., Липская М.А. Скрытие информации в изображениях // Вестник КазАТК им. М. Тынышпаева. –Алматы, 2020. – №1 (112). –Б.283-290.
* **Дайырбаева Э.Н.,** Липская М.А., Тойгожинова А.Ж. Суреттерді өңдеуде стрип-әдісті пайданалу жолдары мен нәтижелері // Вестник КазНИТУ. –Алматы, 2020. –№5. –Б. 279-284.
* **Дайырбаева Э.Н.,** Липская М.А., Тойгожинова А.Ж, Нугуманов Ш.Е. Сандық және компьютерлік стеганографиялардың сипаттамалары мен мүмкіншіліктеріне шолу // Вестник КазАТК. –Алматы, 2020. – №3 (114). –Б. 246-252.
* **Дайырбаева Э.Н.,** Еримбетова А.С., Тұрғанбаев А.Ж., Тойгожинова А.Ж., Нурланбек А.Д. Интерполяция арқылы ақпараттарды жасыру жолдарына талдау. //ҚазККА хабаршысы. –Алматы, 2022. – №3. – Б. 376-383.
* Yerimbetova A., **Daiyrbayeva E.,** Cherikbayeva L. Embedding hidden information in images based on bicubic interpolation //Известия НАН РК. Серия «Информатика». –Алматы, 2023. –№ (1). –P. 50-63

**Публикации в сборниках тезисов и докладов:**

* **Daiyrbayeva E.,** Yerimbetova A., Toigozhinova A., Maratov Z., Sambetbayeva M. // Learning steganography with a strip transform. 6th International Conference on Computer Science and Engineering (UBMK). – Ankara, 2021. – P.209-212.
* **Дайырбаева Э.Н**., Еримбетова А.С. Роль метрики в обработке изображений // МНСК-2021 Информационные технологии Материалы 59-й Международной научной студенческой конференции. –Новосибирск, 2021. - С.53
* **Дайырбаева Э.Н.** Использование RS анализа в стеганографии // МНСК-2022 Информационные технологии Материалы 60-ой Международной научной студенческой конференции. – Новосибирск, 2022. – С.9
* **Дайырбаева Э.Н.**, Еримбетова А.С. Исследование и внедрение встраивания скрытой информации в изображения на основе интерполяции // Российская конференция с международным участием Распределенные информационно-вычислительные ресурсы (DICR-2022), сборник трудов Россия. – Новосибирск, 2022. – С. 45-48.

Диссертационная работа частично выполнена в соответствии с планами научно-исследовательской работы (НИР): «Разработка методов и алгоритмов для многомерных данных в задачах обработки изображений и компьютерной лингвистике» 2020-2022 гг., ГФ АР 08857179.

**Публикации.** По материалам диссертационной работы опубликовано 10 работ: 5 в журналах из перечня КОКСОНВО РК для опубликования основных результатов диссертации на соискание ученой степени PhD и 1 статья в журнале дальнего зарубежья с импакт-фактором, входящих в международные информационные ресурсы Web of Science (Clarivate Analitycs, США) и Scopus (Elesiver, Нидерланды); 4 работ в материалах Международных научных конференций.

**Свидетельства о внесении записи в Государственный реестр прав на объекты авторского права.**

**-** Еримбетова А.С., Дайырбаева Э.Н., Маратов Ж.С. Программа для передачи срытых данных с помощью объекта изображения. Авторское свидетельство. № 20614 от 30.09.2021 г.

- Дайырбаева Э.Н., Еримбетова А.С., Маратов Ж.С., Турганбаев А.Ж. Программа встраивания информации в цифровые изображения с применением интерполяции. Авторское свидетельство № 25573 от 28.04. 2022 г.

- Дайырбаева Э.Н., Еримбетова А.С., Турганбаев А.Ж. Программа встраивания информации в цифровые изображения с применением кривой Безье. Авторское свидетельство № 34297 от 04.04. 2023 г.

**Объем и структура диссертации.** Диссертационная работа состоит из введения, 4 разделов, заключения и списка использованных источников из 168 наименовании, содержит 110 страниц основного компьютерного текста, включая 46 рисунков, 21 формул и 18 таблиц.

**Во введении** раскрыты актуальность, конкретизированы проблемы, связанные с исследуемой темой. Приведены цель и задачи исследования, научная новизна и практическая ценность работы, методы исследования.

**В первой главе** диссертации представлены обзор существующих, актуальных методов встраивания и обнаружения скрытых данных в изображения. Было выявлено, что методы стеганографии имеют широкое применение в области защиты авторских прав и отслеживания траектории распространения цифрового объекта. Применение стегоанализа в области обеспечения безопасности объектов критической телекоммуникационной инфраструктуры в части контроля за передачей конфиденциальных данных.

**Во второй главе** диссертации проведен анализ эффективности методов внедрения скрытых сообщений в изображение: INMI и NMI. Представлено программное обеспечение для внедрения скрытой информации в изображения с использованием известных ранее алгоритмов стеганографии. Приведены результаты по оценке устойчивости рассматриваемых алгоритмов к атакам, использующим изменение цветовой палитры. Также отражены сравнение полученных результатов с другими существующими методами стеганографии.

**В третьей главе** диссертации исследованы существующие методы, базирующиеся на кривой Безье. Были проанализированы преимущества и недостатки кривых Безье по их порядкам. Также был дан предложенный новый метод LIBC5, который был использован в сокрытии данных для создания стегоконтейнера, который содержал секретные данные. Были приведены итоговые сравнительные результаты RS анализа методов INMI и LIBC5.

**В четвертой главе** диссертации представлено полное описание разработанного ПО и схемы её работы. Данное описание поможет будущим исследователям при проведении экспериментов и дальнейшем совершенствовании методов стеганографии.

**В заключении** отражены основные результаты и выводы диссертационной работы.

# 1 МЕТОДЫ СТЕГАНОГРАФИИ И СТЕГОНАЛИЗА

* 1. Обзор предметной области

Стеганография – это наука, изучающая методы скрытой передачи информации. Она включает в себя различные методы и подходы для передачи конфиденциальной информации таким образом, чтобы факт её передачи оставался незамеченным для посторонних. Для достижения этой цели стеганография использует различные техники, такие как внедрение сообщения в изображения, звуковые файлы, видео, текстовые документы, и т.д.

В современном мире, в связи с бурным резким увеличением передаваемого трафика в сети Интернет и развитием компьютерной техники невидимые чернила и бумага были заменены гораздо более универсальными и практичными контейнерами для сокрытия сообщений: цифровыми документами, изображениями, видео- и аудиофайлами. До тех пор, пока электронный документ содержит не относящуюся к восприятию или избыточную информацию, он может использоваться в качестве “прикрытия”, чтобы скрыть секретные сообщения [1-10].

Классическая задача стеганографии формулируется следующим образом. Пусть имеется два заключенных, сидящих в различных ячейках камеры: Алиса и Боб, их задача обсудить детали плана побега, но так чтобы охранник Ева не заподозрила о готовящемся побеге. Алиса может отправлять Бобу любые невинные сообщения (контейнеры) по открытому каналу связи. Ева способна проводить анализ этого контейнера. При помощи методов стеганографии Алиса встраивает секретное послание в контейнер и отправляет его Бобу. Например, она может скрыть сообщение внутри изображения, используя методы, такие как замена наименее значимых битов (LSB) или метод альфа-канала (alpha channel).

Когда Боб получает это изображение, он может использовать те же методы, чтобы извлечь и расшифровать скрытое сообщение. Ева не обладает некоторыми секретными параметрами алгоритма, что не дает ей возможности получить секретное сообщение. Даже проведя статистический анализ, Ева не способна однозначно утверждать о наличии внедрения. Таким образом, Алиса и Боб могут обмениваться секретными сообщениями без вызова подозрений у третьих лиц.

Чтобы защитить свою переписку от Евы, Алиса и Боб могут использовать различные меры. Например, они могут использовать шифрование для защиты своих сообщений перед тем, как спрятать их в данных, а затем использовать стеганографию, чтобы скрыть зашифрованные сообщения в изображении или других данных [11-15].

На рис. 1.1, изображена схема передачи секретных сообщений, описанная выше. С помощью специального алгоритма Алиса встраивает секретное сообщение в контейнер, получая стегоконтейнер, и передает его Бобу. Благодаря особенностям алгоритма стеганографии, Ева, перехватив контейнер, не сможет однозначно утверждать о наличии факта внедрения. Боб без труда извлечет и прочитает секретное сообщение. Таким образом, задача сокрытия факта передачи секретного сообщения от третьих лиц будет выполнена.

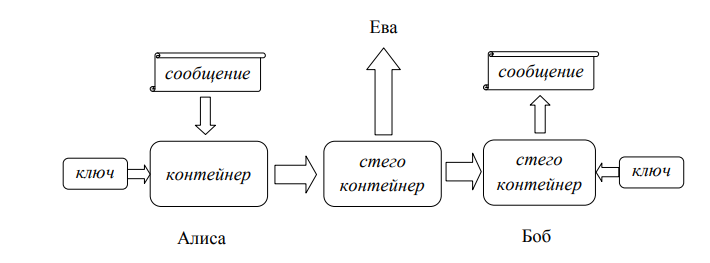


Рисунок 1.1 - Схема передачи секретного сообщения

Метод стеганографии должен предусматривать этап предварительной обработки сообщения: сжатия или шифрования, непосредственно этап внедрения и этап извлечения секретного сообщения. Согласно принципу Керкгофса стегоаналитик (Ева) заранее знает о том какой алгоритм внедрения может быть применен. Это допущение справедливо, так как большинство программ внедрения общедоступны и могут быть скачаны из Интернета. Тем не менее свойства алгоритмов таковы, что внедренное сообщение не меняет статистические свойства контейнера и Ева не может проводить стегоанализ эффективнее чем простое угадывание. В ряде случаем допускается, что Ева обладает неограниченными вычислительными возможностями. Следует отметить, что используемый контейнер не должен быть общедоступным. Например, любое изменение текста Библии будет очевидно вызывать подозрение. Таким образом в качестве контейнера возможно применение личных фотографий, которые могут иметь различное качество сжатия. Причем шумы изображения будут помогать маскировать присутствие секретного сообщения.

В настоящее время проводится множество исследований, по проблемам информационной безопасности. С каждым годом растет число публикаций, посвященных методам стеганографии и стегоанализа. В этом направлении науки работают многие российские и зарубежные ученые: И.Н. Оков, Б.Я. Рябко, И.В. Туринцев, А.Н. Фионов, И.В. Нечта, Р. Бергмар (R. Bergmar), К. Кашен (C. Cachin), М. Чапман (M. Chapman), Ц. Чень (J. Chen), Дж. Фридрич (J. Fridrich), и др.

При работе с стеганографическими системами [16-17] будут использоваться следующие основные понятия.

*Стеганографическая система* или стегосистема – это совокупность средств и методов, которые используются для формирования скрытого канала передачи информации.

*Сообщение* – это любая информация, подлежащая скрытой передаче. В качестве сообщения может использоваться любой вид информации: изображение, текст, аудиосигнал.

*Встроенное (скрытое) сообщение* – это сообщение, встроенное в контейнер.

*Контейнер* – это любая информация, предназначенная для сокрытия сообщения.

Под пустым контейнером будем понимать контейнер без встроенного сообщения. Заполненный контейнер – содержащий встроенную информацию. При выборе вида контейнера нужно обратить внимание на надёжность стегосистемы и возможность обнаружения факта передачи скрытого сообщения. По размеру (протяжённости) *контейнеры* можно разделить на два типа: потоковые и ограниченной длины.

Отличительной характеристикой потокового контейнера является то, что невозможно определить его начало и конец. В таком контейнере биты информации, используемые для скрытия сообщения, включаются в общий поток в реальном масштабе времени и выбираются с помощью специального генератора, задающего расстояния между ними. Также в этом виде контейнера самая большая трудность для получателя – определить, когда начинается скрытое сообщение.

При использовании второго вида контейнера, т.е. ограниченной длины отправитель заранее знает размер файла и может выбрать скрывающие биты в подходящей псевдослучайной последовательности. С другой стороны, такие контейнеры имеют ограниченный объём, и встраиваемое сообщение иногда может не поместиться в файл-контейнер. Другой недостаток заключается в том, что расстояния между скрывающими битами равномерно распределены между наиболее короткими и наиболее длинными заданными расстояниями, в то время как истинный случайный шум будет иметь экспоненциальное распределение длин интервала. На практике чаще всего используются контейнеры ограниченной длины как наиболее распространённые и доступные [18-20].

В зависимости от вида информации, используемой для встраивания сообщений, контейнеры могут быть визуальные, звуковые и текстовые. Визуальный контейнер представляет собой картинку или фотографию, в которой для встраивания сообщений используются небольшие изменения яркости заранее определённых точек растра изображения.

Звуковой контейнер представляет собой речевой или музыкальный сигнал, в котором для встраивания сообщений используются замена младших бит аудиосигнала на секретное сообщение, что практически не отражается на качестве звука.

Текстовый контейнер представляет собой текстовый файл, подготовленный к печати на принтере, в котором для встраивания сообщений используются небольшие изменения стандартов печати (расстояния между буквами, словами и строками, размеры букв, строк и др.).

При выборе того или иного вида контейнера необходимо иметь в виду, что при увеличении объёма встраиваемого сообщения снижается надёжность стегосистемы (при неизменном размере контейнера). Таким образом, используемый в стегосистеме контейнер накладывает ограничения на размер встраиваемого сообщения.

Задача стегоанализа [21] заключается в обнаружении факта внедрения в контейнер и оценке параметров внедрения (объём, позиция и др.). Так как внедряемое сообщение предварительно шифруют, то встраиваемое сообщение выглядит как псеводослучайная двоичная последовательность (по свойству шифра).

В настоящей работе рассматриваются существующие и новые методы внедрения и производится оценка их эффективности при помощи известных методов стегоанализа.

# 1.2 Обзор существующих методов внедрения данных в изображения

Наиболее часто передаваемым файлом по сети Интернет является изображение и видеофайлы. Видеофайл может в свою очередь рассматриваться как набор чередующихся картинок. Изображения в подавляющем числе случаев представлены в виде Jpg файла, использующим сжатие данных с потерями.

В общем случае изображение на мониторе представляет собой матрицу цветных пикселей. Цвет пикселя представлен в памяти компьютера в виде числа. Незначительно искажение такого цвета (числа) не заметно для глаза наблюдателя, что используется при внедрении. Изменение цвета может проводиться для одиночного пиксела либо для целой группы пикселей, путем снижения яркости. В первом случае возникает статистическая неоднородность (шум) картинки, которая бывает и в обычной фотографии. Во втором случае группа пикселей понижает свою яркость равномерно и создается впечатление естественной игры освещения.

Обычно информацию встраивают не в каждый возможный пиксел, а в некоторые случайно выбранные (например, не более 5% от общего числа). Такое требование возникает потому, что последние сообщение статистические выглядит случайным, а любые извлечения из картинки статистически взаимосвязаны. Поэтому выбирают случайные пикселы и встраивают туда информацию, что внешне похоже на тепловой шум матрицы фотоаппарата.

Рассмотрим наиболее распространенные методы внедрения, встречающиеся в научной литературе.

1. Least Significant Bit (LSB) [22] - один из самых простых методов стеганографии. Данные встраиваются в наименее значимый бит каждого пикселя изображения. Этот метод очень уязвим для стеганализа, так как изменение наименее значимого бита не сильно влияет на качество изображения.
2. Методы на основе стеганографических алгоритмов: Эти методы используют различные стеганографические алгоритмы для внедрения данных в изображения. Они более безопасны, чем методы, основанные на изменении младших битов, и обладают более высокой устойчивостью к атакам [23-25].
3. Masking and Filtering [26] – этот метод использует маскирование и фильтрацию, чтобы встроить данные в изображение. Изображение разбивается на блоки, и данные встраиваются в эти блоки, используя маскирование и фильтрацию.
4. Phase Coding [27] – этот метод использует фазовый кодировщик для встраивания данных в изображение. Фаза каждого пикселя изменяется на определенную величину, что позволяет внедрить данные.
5. Алгоритм разность – модификации (DM) [28] - этот метод основан на изменении значений пикселей изображения. Вместо внедрения данных в наименее значимый бит пикселя, он изменяет значения пикселей, чтобы встроить данные.
6. Алгоритм дискретного косинусного преобразования (DCT) [29]- этот метод использует дискретное косинусное преобразование для разложения изображения на компоненты частоты. Данные затем встраиваются в наименее значимые коэффициенты преобразования.
7. Вейвлет – преобразование [30] – этот метод использует вейвлет-преобразование для разложения изображения на компоненты частоты. Данные затем встраиваются в наименее значимые коэффициенты преобразования.
8. Методы на основе стеганографических алгоритмов: Эти методы используют различные стеганографические алгоритмы для внедрения данных в изображения. Они более безопасны, чем методы, основанные на изменении младших битов, и обладают более высокой устойчивостью к атакам.

Это лишь некоторые из методов внедрения данных в изображения. Каждый метод имеет свои преимущества и недостатки, и выбор метода зависит от конкретных требований [31-33].

# 1.2.1 LSB метод

Одним из самых распространенных и широко используемых методов стеганографии является метод встраивания в младшие биты (LSB). На сегодняшний день его продолжают применять для встраивания скрытых данных в цифровые изображения, фильмы и аудиозаписи [34].

Согласно алгоритму, каждый пиксель в изображении представлен битами, которые определяют цвет или яркость этого пикселя. Наименее значимый бит (LSB) содержит наименее значащую информацию и обычно не влияет на качество изображения. Именно поэтому метод LSB можно использовать для внедрения дополнительной информации, не заметной для глаза.

Внедрение данным методом может производиться в каждый канал RGB изображения. Причем все каналы равнозначны по объему и устойчивости к анализу. По этой причине исследователи в своих статьях рассматривают только черно-белые изображения (т.н. gray scale). В таком случае, во-первых, изображения становятся меньше по размеру, что упрощает обработку, и, во-вторых, делает метод внедрения универсальным: без привязки к конкретному цветовому слою.

Недостатком метода LSB является то, что он уязвим для атак стеганализа. Для обнаружения скрытых данных в изображении, необходимо проанализировать наличие изменений в наименее значимых битах. Также внедрение большого количества данных может привести к существенному снижению качества изображения.

В целом, метод LSB прост в использовании и может быть эффективным для внедрения небольшого объема данных в изображение. Он широко используется в различных приложениях [35-45], таких как внедрение водяных знаков, но для передачи большого объема данных лучше использовать более сложные методы стеганографии.

Авторы [46] представили метод LSB и предложили надежный подход к сокрытию данных с высоким уровнем безопасности, использующий обрезку изображений и стеганографию по наименьшему значащему биту (LSB). Новый подход, предложенный авторами основан на разделении секретного текстового сообщения на четыре части и извлечении четырех фрагментов из цветного изображения контейнер с определенными секретными координатами. Каждая часть сообщения встраивается в обрезку изображения с использованием предопределенной секретной последовательности. При проведении эксперимента был использован PSNR и трудоемкость, представленную в относительных единицах для сравнения с другими существующими результатами при известных методах.

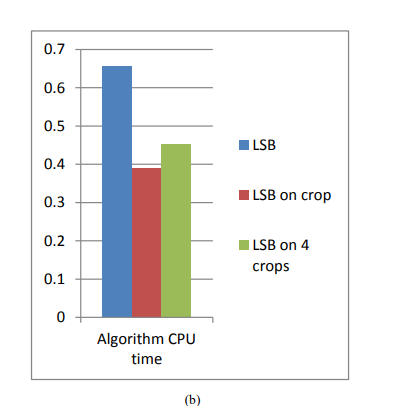
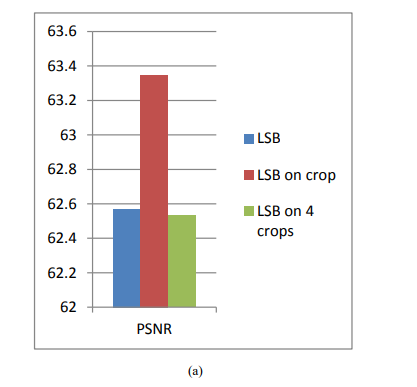


Рисунок 1.2 - (a) PSNR для изображения stego, (b) процессорное время в секундах [46]

По результатам исследований можно увидеть, что предлагаемый подход PSNR и трудоемкость находятся в пределах того же диапазона, что и другие аналогичные подходы, однако он оказался более безопасным (рис.1.2).

В статье [47] исследователи также предложили эффективный подход к стеганографии для сокрытия информации в изображении в серой шкале. Авторы в своих исследованиях сравнили новый метод с двумя более известными методами PVD и GLM. Если сравнить по количеству внедрение данных согласно таблице 1.1, тогда можно увидеть, что предложенный подход авторами позволяет внедрить больше данных, чем существующий метод GLM.

Таблица 1.1 – Размеры сообщения для внедрения [47]

|  |  |  |  |
| --- | --- | --- | --- |
| **Изображение** | **Размер изображения** | **Объем внедряемых данных** | |
| **GLM** | **PVD** |
| Lena | 128 x128 | 2048 | 2493 |
| 256 x256 | 8192 | 10007 |
| 512x512 | 32768 | 40017 |
| 1024x1024 | 131072 | 160604 |
| Baboon | 128 x128 | 2048 | 2560 |
| 256 x256 | 8192 | 10211 |
| 512x512 | 32768 | 40990 |
| 1024x1024 | 131072 | 163724 |
| Peppers | 128 x128 | 2048 | 2443 |
| 256 x256 | 8192 | 9767 |
| 512x512 | 32768 | 39034 |
| 1024x1024 | 131072 | 156308 |

Авторами PSNR был использован для оценки качества стего-изображения после встраивания секретного сообщения в контейнер. Также был применен формула для вычисления сходства методов и результатов. Экспериментальные результаты показали эффективность предложенного метода по сравнению с другими методами. С точки зрения объема данных метод PVD был лучшим после предложенного метода, а метод GLM был последним. Измерения сходства показали, что метод GLM был лучшим. Предложенный метод дал результаты, близкие к GLM, но метод PVD дал худшие результаты (рис.1.3 –рис.1.6)

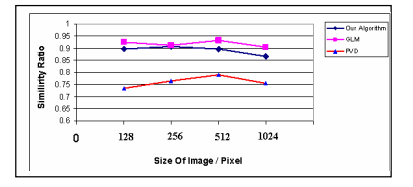


Рисунок 1.3 – Меры сходства для изображения Lena [47]

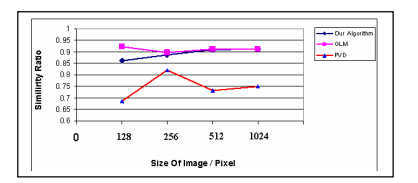


Рисунок 1.4 – Меры сходства для изображения Baboon Image [47]

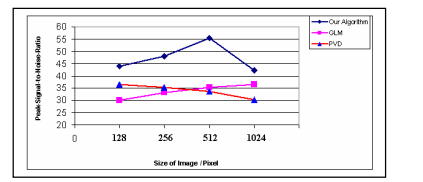


Рисунок 1.5 - PSNR для изображения Lena [47]

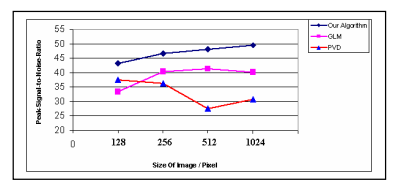


Рисунок 1.6 - PSNR для изображения Baboon Image [47]

В целом, [47] является важным вкладом в область стеганографии и может быть полезной для разработчиков, исследователей и всех, кто интересуется безопасностью и скрытым хранением информации в изображениях.

В статье [48] авторы применяют для определения и оценки длины встроенного сообщения два метода стегоанализа: RS и Хи-квадрат. Для секретной связи устойчивость стеганографии к стеганоанализу очень важна с точки зрения информационной безопасности. Также исследователями предлагается стеганографический метод обратимого преобразования гистограммы, основанный на функции LSB, для противодействия статистическому стегоанализу. Большинство начальных значений RS-атаки находятся в пределах ±6%, но некоторые начальные значения обнаружения x2 превышают 10%.

Работа [49] является одной из первых, посвященных методам внедрения данных в изображения с использованием LSB-метода. В ней описывается метод внедрения данных в изображения, по мере того, как пользователь увеличивает позицию встраиваемого бита в изображение контейнер для сокрытия зашифрованного секретного сообщения, пиковое отношение сигнал/шум (PSNR) линейно уменьшается, а качество изображения stego снижается (рис.1.7).

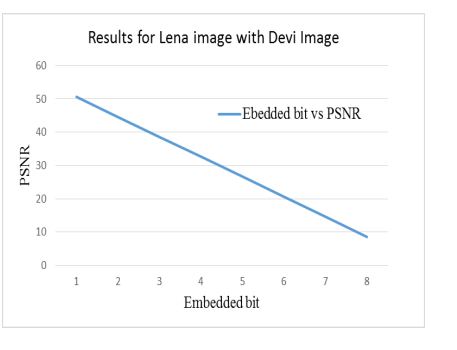


Рисунок 1.7 –Результаты эксперимента по картинке Lena [49]

В работах [50-53] представляет обзор и сравнение различных методов внедрения данных в изображения с использованием LSB – метода. В работах рассмотрены различные подходы, включая методы изменения младших битов, методы на основе стеганографических алгоритмов и методы на основе преобразования вейвлетов. Авторы демонстрируют, что предложенный метод улучшает устойчивость и безопасность метода внедрения данных.

В целом, метод LSB остается одним из наиболее популярных и широко используемых методов стеганографии, и его применение и анализ продолжаются и в настоящее время. Сделав обзор существующих работ по методу LSB можно сделать следующие выводы:

1) более ранние стеганографические схемы, которые использовали метод LSB, фокусировались на внедрении информации с помощью изменений цветов пикселей исходного изображения. Очевидно, что чем меньше внедряется информация, тем будет меньше статистических искажений контейнера и тем более безопасной (устойчивой к анализу) становится стеганографическая схема.

2) современные возможности в стегоанализе четко показывают, что внедрение информации в младшие биты с последовательным либо случайным размещением сообщения успешно обнаруживается даже для коротких сообщений (более 5% от допустимого объема внедрения).

# 1.2.2 Алгоритм дискретного косинусного преобразования (DCT)

Одним из перспективных методов стеганографии является внедрение информации в jpg изображения. Известно, что формат jpg хранит изображение как запись пространственной волны периодического сигнала. Оси абсциссы и ординаты соответствуют позиции пикселя на мониторе, а значение по оси аппликат соответствует цвету изображения. Вся картинка представляет собой набор квадратов мелких изображений 8х8. Сигнал записывается в виде коэффициентов ряда Фурье. Так как ряд Фурье бесконечный для конечных сигналов (картинок), то сохраняют только первые 64 коэффициента этого ряда.

Стеганографический метод внедрения, базирующийся на дискретном косинусном преобразовании (DCT) — это один из наиболее распространенных методов внедрения информации в изображения. Алгоритм DCT основан на расчете косинусных функций для частотных компонент изображения и преобразовании их в векторы коэффициентов, которые представляют изображения. Данные коэффициенты хранятся в матрице, которая представляет изображение.

Для внедрения информации в изображение с использованием алгоритма DCT, сначала выбирается некоторое количество коэффициентов из матрицы DCT и изменяется в соответствии с информацией, которую мы хотим скрыть. Затем производится обратное DCT-преобразование, в результате которого мы получаем измененную версию исходного изображения.

Из множества обзоров литературы за последние 10 лет, которые были проанализированы в рамках исследования, можно выделить несколько ключевых направлений и достижений, которые были получены в DCT стеганографии.

Первая группа статей [54-62] фокусировалась на исследовании производительности методов стеганографии на основе DCT. Некоторые исследователи сравнивали эффективность методов, основанных на DCT с другими методами, такими как wavelet-перобразование или descrete Fourier transform. Другие исследования включали анализ устойчивости методов на основе DCT к атакам и исследование скрытости информации.

Вторая группа статей [63-72] фокусировалась на разработке новых алгоритмов стеганографии на основе DCT. Например, некоторые исследователи предложили использовать двойное DCT для скрытия информации, что улучшило качество стеганографии по сравнению с обычным DCT. Другие исследования предложили использовать кластеризацию для определения наиболее эффективных коэффициентов DCT для скрытия информации.

Третья группа статей [73-75] была посвящена разработке алгоритмов извлечения скрытой информации на основе DCT. Некоторые исследования предложили использовать машинное обучение, чтобы повысить эффективность извлечения информации, а другие предложили использовать алгоритмы восстановления изображений для улучшения качества извлекаемой информации.

В целом, исследования за последние 10 лет показывают, что DCT продолжает быть важным инструментом в стеганографии, и его эффективность может быть улучшена с помощью новых разработок и усовершенствований. Данный метод обладает рядом преимуществ, среди которых высокая эффективность внедрения информации, низкий уровень потерь изображения и простота реализации. Однако, он также обладает и недостатками, связанными с тем, что в ряде случаев просто заметить изменения в матрице DCT, что позволяет с обнаружить скрытую информацию.

Таким образом, метод внедрения информации в изображения с использованием алгоритма DCT имеет свои преимущества и недостатки и может использоваться в рамках стеганографии для скрытой передачи информации в изображениях.

# 1.2.3 Вейвлет –преобразование

Цифровая обработка изображений продолжает развиваться и сегодня. Во многих промышленных и научно-прикладных областях существуют различные задачи цифровой обработки изображений. Многие вопросы имеют полное решение. Это относится к задачам фильтрации, сегментации, распознавания объектов на изображениях, обработки мультимедийной информации и оценки цифрового телевизионного сигнала [76,77]. Чтобы выполнить цифровую обработку изображений, необходимо преобразовать непрерывный (аналоговый) сигнал изображения в цифровую матрицу.

Это преобразование включает в себя выполнение двух преобразований. Первое преобразование — это замена реального непрерывного изображения набором выборок в дискретные моменты времени, такое преобразование называется выборкой. Второй — это преобразование непрерывного набора значений сигнала изображения в набор квантованных значений, такое преобразование называется квантованием. Основные этапы цифровой обработки изображений показаны на рис. 1.8:

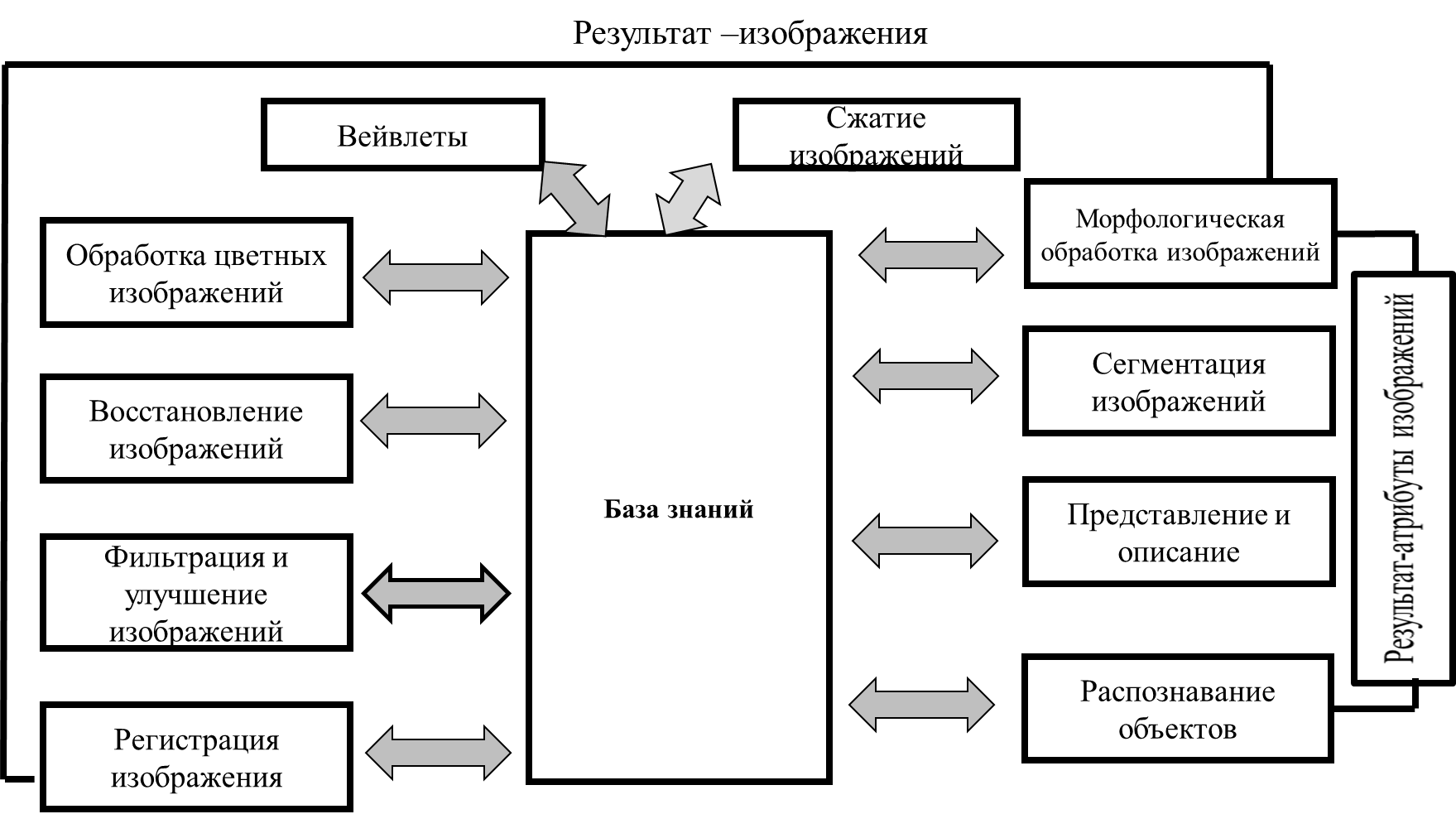


Рисунок 1.8 - Основные этапы обработки цифровых изображений

Анализируя литературу в этой области, можно выделить следующее: практически полное отсутствие методов встраивания устойчивых к сжатию мультимедийных данных. Одним из преобразований, допускающих такое встраивание, является дискретное вейвлет-преобразование [78,79]. Как известно, набор вейвлетов в их временном или частотном представлении может аппроксимировать сложный сигнал или Вейвлет-преобразование является одним из методов внедрения информации в стеганографии. Оно используется для разложения изображений на различные частотные компоненты, которые затем могут быть использованы для внедрения секретной информации.

Новые эффективные методы обработки изображений стали возможны с развитием теории вейвлетов, которые, по сравнению с преобразованием Фурье, позволяют нам с гораздо большей точностью представлять мельчайшие особенности функций, изображений и сигналов, вплоть до разрывов первого рода (скачков), с их привязкой ко времени или пространственные координаты. Термин "вейвлет" был введен Алексом Гроссманом и Жаном Морле в середине 1980-х годов в связи с анализом сейсмических и акустических сигналов. В настоящее время вейвлеты используются в задачах распознавания изображений, при обработке и синтезе различных сигналов, при анализе изображений различной природы, для сжатия больших объемов информации.

Вейвлет (от wavelet - малая волна, пульсация, также всплеск, часто - вейвлет) [80] - математическая функция, которая анализирует различные частотные составляющие данных. График функции выглядит как волнообразные колебания с амплитудой, уменьшающейся до нуля вдали от начала координат. Идеи теории вейвлетов возникли, когда появилось достаточное количество рядов экспериментальных данных, обработка которых стандартным и хорошо разработанным методом преобразования Фурье показала свои ограничения для нахождения в них закономерностей. Стремительное развитие компьютерных технологий также сыграло свою роль, что позволило численно решать такие задачи, которые раньше были просто недоступны. Практически важные вейвлеты традиционно определяются как функции одной действительной переменной с действительными значениями. В зависимости от математической модели (структуры, объема, структурирующего поля возможных значений и преобразований) различают дискретные и непрерывные вейвлеты. Поскольку разложение сигналов в вейвлет-базисе осуществляется с использованием арифметики с плавающей запятой, возникают ошибки, величина которых зависит от степени аппроксимации сигнала.

Рассмотрим пример внедрения информации в коэффициенты дискретного вейвлет-преобразования изображения. Для этого сначала производится разложение изображения на коэффициенты дискретного вейвлет-преобразования. Затем происходит выбор коэффициентов, в которые будет внедрена секретная информация.

Внедрение информации происходит путем изменения выбранных коэффициентов на небольшую величину в зависимости от значения битов секретной информации. Это изменение незначительно для человеческого глаза, но позволяет внести секретную информацию в изображение без заметных с точки зрения восприятия искажений человеческим глазом.

После внедрения информации необходимо произвести обратное вейвлет-преобразование для получения измененного изображения. Важно заметить, что при этом происходит некоторое сжатие изображения, что может привести к потере качества. Поэтому при выборе метода стеганографии необходимо учитывать не только стойкость информации, но и ее визуальное качество [81-86].

# 1.3 Практическое применение методов стеганографии

Применение стеганографии имеет мирное и полезное применение – встраивание цифровых водяных знаков. Цифровой водяной знак (Digital Watermark) — это невидимая информация, встроенная в цифровое изображение, видео или звуковой файл для различных целей, таких как: защита авторских прав, аутентификация, мониторинг и отслеживание [87]. Например, автор встраивает свои идентификационные данные в каждую продаваемую копию программы, изображения или иного файла. Если будет перехвачен аналогичный файл под авторством третьего лица, то истинный автор может извлечь внедренный ЦВЗ и доказать свое авторство в судебном порядке. Стеганографические алгоритмы позволяют свободно встраивать и извлекать ЦВЗ.

Существует множество исследований, посвященных разработке новых методов встраивания водяных знаков и анализу их эффективности [88]. К требованиям ЦВЗ относят устойчивость – свойство, позволяющее сохранять информацию даже после графических преобразований контейнера, например, масштабирования или сохранения в другом формате файла.

В работе авторов [89] для оценки систем водяных знаков были применены различные методы атак (атаки на удаление, атаки с шумоподавлением и сжатием с потерями, модуляционные атаки, атаки на усреднение и геометрические атаки), что обеспечивает автоматизированный и объективный анализ существенных методов водяных знаков для выбранных области применения. Ключевыми моментами исследований были алгоритмы нанесения водяных знаков, которые устойчивые к геометрическим искажениям.

В работе [90] исследователи продемонстрировали универсальную модель водяных знаков для приложений с цифровыми изображениями (с графическим интерфейсом). В представленном модели определили набор возможных входов, выходов и функций компонентов, изучив схемы нанесения водяных знаков, предложенные для различных приложений для обработки изображений (рис.1.8):

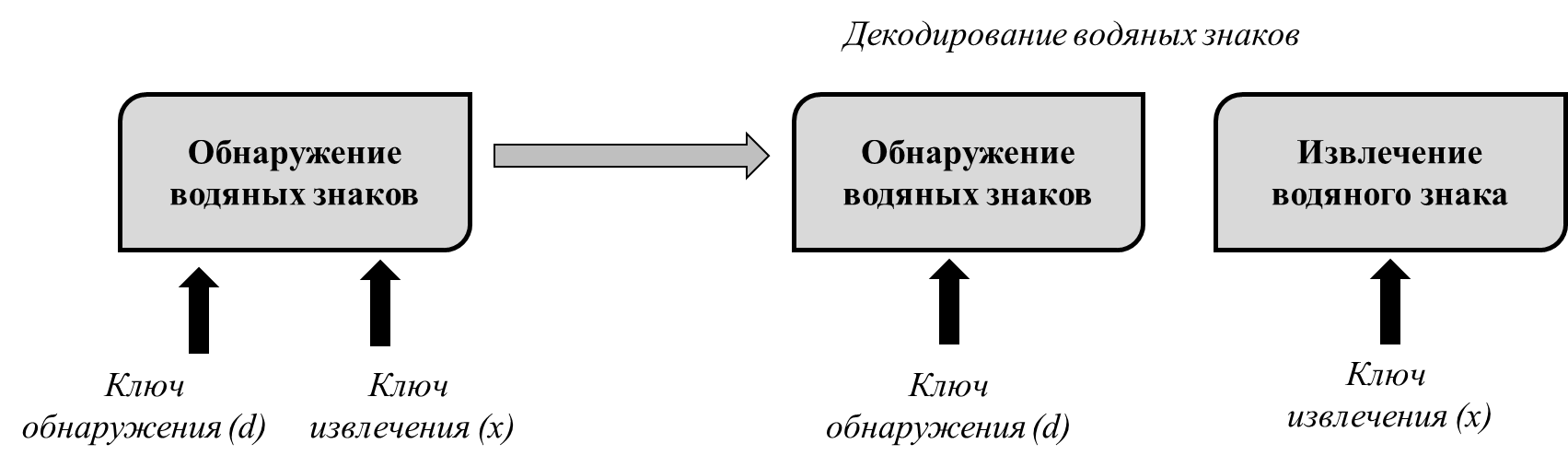


Рисунок 1.9 - Развертывание ключей в функции обнаружения базовой модели [90]

Применяя предложенную модель с подходящими входами, выходами и функциональными свойствами, можно охарактеризовать и описать все возможные варианты схем нанесения водяных знаков на цифровые изображения (например, для проведения необходимого вычислительного анализа). В данной статье авторами был прописан четкие описания предложенной модели в виде таблицы 1.2:

Таблица 1.2 - Нанесение водяных знаков на цифровые изображения: его формальная модель, фундаментальные свойства и возможные атаки [90]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Используемая модель** | **Цели** | **Входы и выходы** | **Функции компонентов** | **Теория** | **Ограничения** |
| Предложенный | Обеспечить средства для систематического развития и, таким образом, разработать единую и более реалистичную теорию нанесения водяных знаков на цифровые изображения | Изображение-данные (с различными свойствами, например, оригинал, с водяными знаками и т.д.,) | Генератор ключей | Цифровая обработка изображений и сигналов | Может не подходить для изучения схем стеганографии |
|  |  | Водяной знак | Генерация водяных знаков | Криптография |  |
|  |  | Сообщение | Встраивание водяного знака |  |  |
|  |  | Ключ (для каждой функции) | Обнаружение водяного знака |  |  |
|  |  |  | Извлечение водяного знака |  |  |

Предложенная модель позволяет унифицировать обработку всех практически значимых вариантов нанесения водяных знаков на цифровые изображения

В исследовании [91] предлагается новый метод создания водяных знаков на изображениях, основанный на инвариантных областях изображения. Понятие инвариантные области понимается, как самоадаптивные участки изображения, которые деформируются при геометрических преобразованиях. Авторы для разработки водяных знаков рассмотрели три различных метода определения инвариантной области, основанные на масштабно-пространственном представлении изображения. В каждую инвариантную область водяной знак встраивается после геометрической нормализации в соответствии с формой области; путем привязки водяных знаков с помощью инвариантных областей можно легко получить устойчивость к геометрическим преобразованиям.

В экспериментах был использован шаблон водяного знака длиной 1024 и только 50 инвариантных областей из 50 при обнаружении водяного знака. Экспериментальные результаты показывают, что предложенный метод устойчив к различным этапам обработки изображений, включая геометрические преобразования, обрезку, фильтрацию и сжатие в формате JPEG.

В работе [92] предлагается защищенный алгоритм создания водяных знаков на изображениях, устойчивых к атаки коалиции. Так группа пользователей может объединиться и сравнить свои копии защищенных файлов. Поскольку ЦВЗ в каждой копии отличается, то все найденные различия могут быть искажены. Тем не менее предлагается встраивать такой знак, который будет устойчив к подобной атаке.

Водяной знак создавался как независимый и идентично распределенный гауссовский случайный вектор, который незаметно вставляется в виде расширенного спектра в наиболее значимые с точки зрения восприятия спектральные компоненты данных. По мнению авторов, вставка водяного знака в этом режиме делает водяной знак устойчивым к операции обработки сигналов (такие как сжатие с потерями, фильтрация, ЦАП и АЦП, повторная обработка и т.д.) и обычные геометрические преобразования при условии, что исходное изображение доступно и что его можно успешно сопоставить с преобразованным водяным знаком изображение. В этих случаях детектор водяных знаков однозначно идентифицирует владельца.

Авторы также отмечают что, использование гауссовского шума обеспечивает высокую устойчивость к атакам с несколькими документами или сговору. В статье даны экспериментальные результаты: применение этого метода к цветным изображениям с точки зрения авторов несложно. Наиболее распространенным преобразованием цветного изображения является преобразование его в черно-белое. Однако устойчивость к определенному цветовому изображению следует изучить процедуры обработки.

В книге представлены [93] обзор форматов хранения различных типов носителей, сжатия данных, а также механизмов и методов защиты этих типов носителей. Материалы, представленные в этой книге, дают всесторонний обзор различных аспектов управление медиа-активами и цифровыми правами. Целью книги являются: изложить соответствующие теоретические основы и последние результаты исследований в этой области.

В следующей статье [94] исследованы ситуаций, в которых требуется встроить в изображение несколько водяных знаков. Авторы в своем исследовании предлагают эффективный подход к внедрению двойных водяных знаков путем расширения алгоритмов создания одиночных водяных знаков для нанесения водяных знаков с цифрами и логотипом соответственно. Идея алгоритма заключается в том, если потребители хотят достичь нескольких целей, например, защитить авторские права на один продукт у нескольких владельцев на разных этапах или проверить целостность контента и защитить авторские права одновременно, то они должны использовать множественные водяные знаки, то есть встраивать более одного водяного знака в одно и то же мультимедийное изображение. Авторы в своих исследованиях применяли три схемы для встраивания двойного водяного знака, как показано на рис. 1.10-1.11.

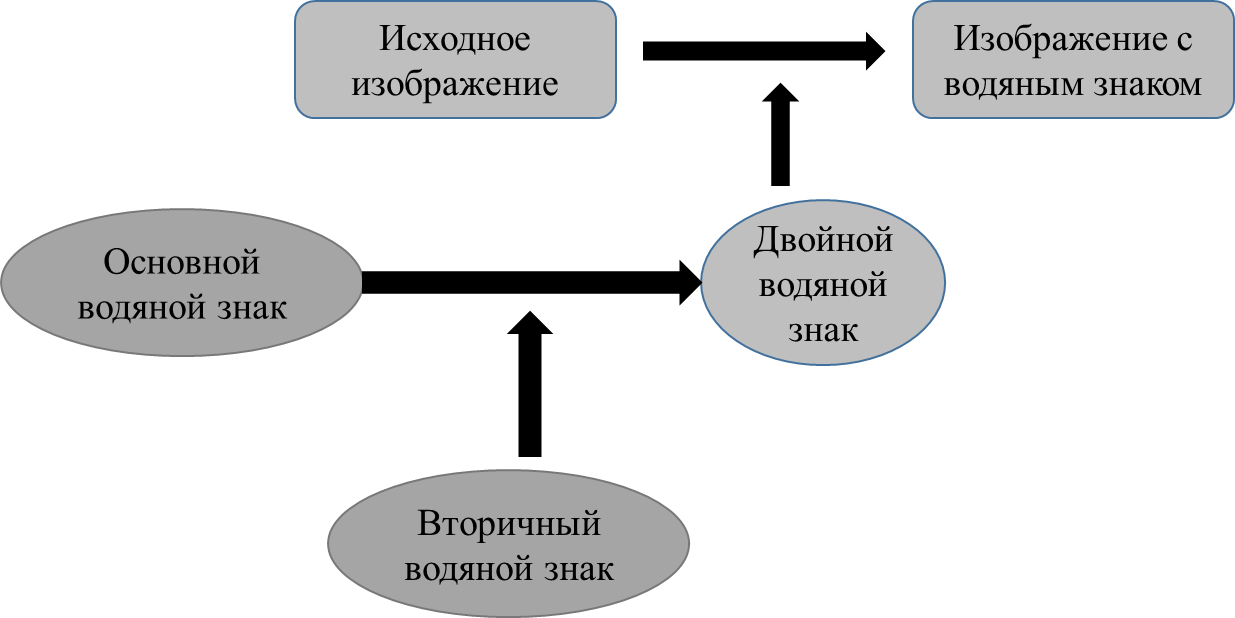


Рисунок 1.10 - Схема встраивания двойного водяного знака [94]

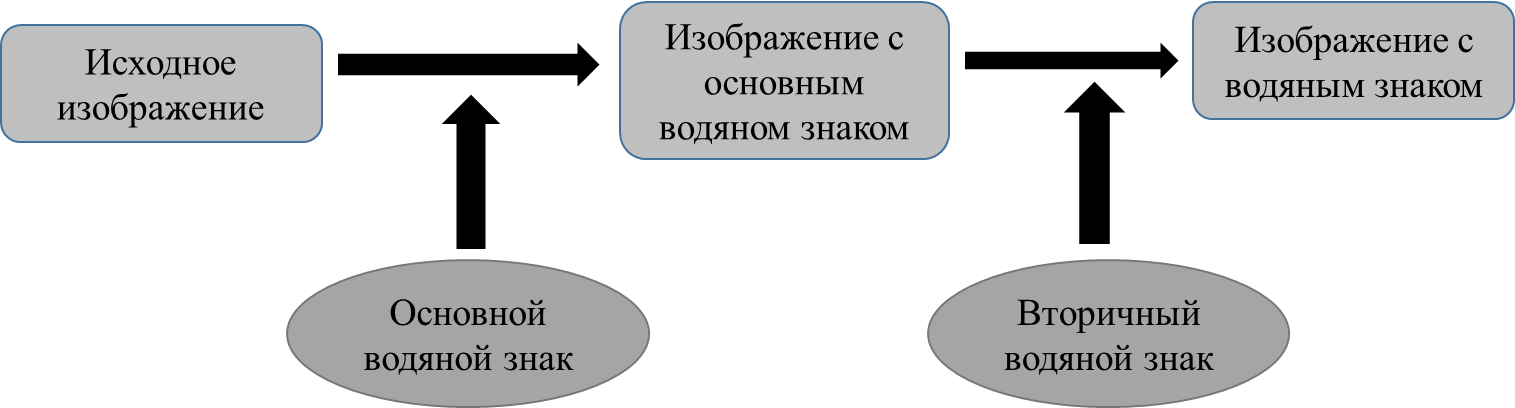


Рисунок 1.11 – Вторая схема встраивания двойного водяного знака [94]

Предлагаемые в данных работах схемы внедрения ЦВЗ не конфликтуют и дополняют друг друга.

Таблица 1.3- Сравнение PSNR с алгоритмом Chemak (Чемак) и новым алгоритмом [94]

|  |  |  |
| --- | --- | --- |
| Тип искажения | PSNR (Chemak) | PSNR (новый) |
| Среднее смещение | 24.6090 | 28.5274 |
| Контрастное растяжение | 24.6003 | 28.5127 |
| Импульсивный шум соли и бумаги | 24.6499 | 28.8416 |
| Мультипликативный спекл-шум | 24.6186 | 28.6741 |
| Aддитивный гауссовский шум | 24.5906 | 28.4598 |
| Cжатие в формате jpeg | 24.7849 | 30.1737 |

Экспериментальный результаты (табл.઎1.3) показывают, что результирующие алгоритмы двойного нанесения водяных знаков имеют значительно более высокую PSNR по сравнению с существующими алгоритмами создания двойных водяных знаков, а также сохраняют ту же надежность и более высокую чувствительность, чем исходные алгоритмы создания одиночных водяных знаков, на которых они основаны.

Эти работы показывают, что цифровые водяные знаки остаются активной областью исследований, и что различные методы могут быть использованы для создания более эффективных и устойчивых водяных знаков.

# 1.4 Обзор известных методов стегоанализа

Стегоанализ — это процесс обнаружения наличия стеганографического сообщения в контейнере, то есть анализа изображений, аудио или видео файлов с целью обнаружения скрытой информации [95,96]. Строго говоря, задача обнаружения эквивалентна задачи классификации. Имеется два класса: пустых и заполненных контейнеров. Известны статистические свойства данного классов. Необходимо определить принадлежность произвольного контейнера к одному из этих классов. Отличие методов стегоанализа между собой заключается только в наборе анализируемой статистики или характеристик контейнера. Некоторые известные методы стегоанализа включают в себя:

* Анализ статистических свойств: этот метод основывается на анализе статистики контейнера, таких как распределение цветов пикселей, их взаимная корреляция и энтропия. Изменения в этих свойствах могут указывать на наличие скрытой информации.
* Алгоритмы машинного обучения могут использоваться для обнаружения изменений в паттернах или свойствах носителя, которые указывают на наличие стеганографического сообщения.
* Анализ на основе спектральных свойств: этот метод основывается на анализе спектральных свойств контейнер, таких как частоты и фазы, чтобы обнаружить изменения, которые могут указывать на наличие стеганографического сообщения.
* Анализ на основе артефактов: этот метод основывается на обнаружении артефактов, которые обычно появляются при внедрении стеганографического сообщения, таких как изменения насыщенности цветов, блоковый эффект и т.д.
* Анализ на основе сравнения: этот метод сравнивает оригинальный носитель с возможными стеганографическими версиями, чтобы найти отличия, которые могут указывать на наличие скрытого сообщения.

Это лишь некоторые известные методы стегоанализа, и исследователи продолжают разрабатывать новые методы и улучшать существующие для более эффективного обнаружения стеганографических сообщений.

Некоторые из новых работ по известным методам стегоанализа включают в себя [97-99]:

1. Анализ на основе графов:Идея подхода заключается в использовании теории графов для анализа стеганографических сообщений в целях повышения точности их обнаружения. Данная методика позволяет отображать взаимосвязи между различными элементами стеганографического сообщения в виде графа, что позволяет выявлять скрытые связи между этими элементами и обнаруживать скрытые сообщения. Этот подход может быть использован для различных целей, например, для обнаружения наличия стеганографического сообщения в файле, для определения его типа, для обнаружения и анализа сокрытой информации. В целом, использование этого подхода может помочь в повышении эффективности и точности анализа стеганографических сообщений.
2. Машинное обучение с глубокими нейронными сетями: Идея подхода заключается в использовании глубоких нейронных сетей для анализа изображений и видео с целью обнаружения скрытой информации. С помощью алгоритмов обработки большого объема данных модели глубоких нейронных сетей позволяют корректно определять изменения в пикселях и фреймах изображений и видео, которые могут указывать на наличие стеганографической информации. Такой подход может быть использован для обеспечения безопасности данных и защиты от нежелательных утечек информации.
3. Анализ на основе многомерных статистических свойств: Идея данного подхода заключается в использовании многомерных статистических свойств для обнаружения стеганографических сообщений, которые были внедрены с помощью более сложных алгоритмов стеганографии. Данный подход использует совместную энтропию для анализа изображений и обнаружения скрытых сообщений. Такой подход может быть эффективен для обнаружения стеганографических сообщений, которые меньше зависят от изменений окружающей среды.
4. Анализ на основе моделей глубокого обучения: Идея данного подхода заключается в использовании моделей глубокого обучения для повышения точности обнаружения стеганографических сообщений. В рамках подхода используется анализ носителя и знания о том, как сообщение было внедрено, что позволяет улучшить результаты анализа. Такой подход может быть эффективным в обнаружении скрытой информации и защите от кибератак.
5. Обнаружение сообщений с использованием машинного зрения: Идея подхода заключается в использовании алгоритмов машинного зрения для обнаружения скрытых сообщений в изображениях. Эти методы основаны на обнаружении изменений в текстуре, цветовых свойствах и других характеристиках изображения, которые могут указывать на наличие стеганографического сообщения. Это позволяет эффективно обнаруживать скрытые сообщения, которые могут быть использованы в криминальной или шпионской деятельности.

# 1.4.1 RS – метод

RS – анализ основан на применении двойной статистики, полученной из пространственных корреляций в изображениях. На данный момент не имеется каких-либо исследований по анализу скрытности внедрения данных алгоритмов при помощи методов стегоанализа. Одним из основных методов статистического стегоанализа является метод RS, который был разработан Фридрич и др. [100-102] в 2001 году. Сокращение в названии расшифровывается как Regular–Singular, то есть «регулярно-сингулярный».

RS — это метод стегоанализа, который основан на анализе остаточной информации, оставленной при встраивании сообщения в изображение или другой мультимедийный контент. Основная идея RS состоит в том, чтобы определить изменения, произведенные в остаточной информации после встраивания сообщения, и использовать эти изменения для обнаружения скрытой информации. Остаточная информация — это разница между оригинальным и измененным контентом после встраивания сообщения.

RS использует статистические методы для анализа остаточной информации и определения ее характеристик, таких как среднее значение, дисперсия и ковариация. Затем эти характеристики используются для построения модели, которая может определять, содержит ли контент скрытую информацию. Одним из преимуществ RS является его способность обнаруживать скрытую информацию, встроенную с помощью различных методов стеганографии, в том числе и более сложных методов. Однако этот метод также может давать ложноположительные результаты, если остаточная информация изменена, например, в результате сжатия изображения или другого контента.

Суть метода состоит в следующем: все изображение разбивается на группы по пикселов где четно (1.1). Для групп пикселов определяется функция регулярности . Под значением пикселов понимаем число от 0 до 255 .

 (1.1)

Функция называется флиппингом и имеет свойство . Определяют две функции флиппинга – , соответствующую инверсии младшего бита пиксела, и  представляющую собой инверсию с переносом в старший бит (1.2-1.3):

 (1.2)

 (1.3)

При применении флиппинга к группе получают преобразованную группу пикселов. Далее делят все группы пикселов на классы следующим образом:

если  то ;

если  то ;

если  то .

Метод RS может указать маленькую ненулевую длину сообщения из-за случайных отклонений даже для пустого контейнера. Это начальное ненулевое отклонение может быть, как положительным, так и отрицательным и накладывает ограничение на достижимую точность RS анализа.

Для каждой группы флиппинг производится два раза: с прямой и с инвертированной маской. После проведения операций классификации для всех групп выполняется подсчет ряда количественных характеристик:

* количество обычных групп для маски ;
* количество необычных групп для маски ;
* количество обычных групп для инверсной маски ;
* количество необычных групп для инверсной маски .

RS анализ показывает более точные результаты для сообщений, которые рассеянно распределены по картинке, по сравнению с анализом сообщений, которые сконцентрированы в определенной области изображения.

На рис.1.12 представлен типичный вид RS-диаграммы – графиков значений  в зависимости от количества пикселей с инвертированными LSB в изображении [92]:

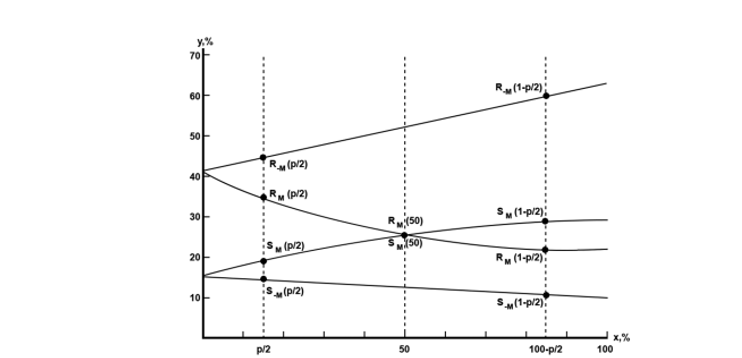


Рисунок 1.12 – RS диаграмма

Ключевая особенность RS – метод состоит с том, что он анализирует количественные характеристики небольших групп пикселей. В связи с чем он, хотя и не способен декодировать область потенциального встраивания, может обнаружить скрытие, произведенное в случайные биты, а не последовательно.

# 1.3.2 SPAM метод

SPAM (Statistic-based Pixel Adjacency Model) - это метод стегоанализа, который использует статистическую модель соседних пикселей для обнаружения скрытых сообщений в мультимедийных данных. Однако, по сравнению с другими методами, такими как паритетная проверка или анализ используемого алгоритма сжатия, SPAM предлагает более точное и эффективное обнаружение скрытой информации. Если в данных присутствует скрытая информация, то статистические свойства контейнера может измениться в тех областях, где скрытая информация была внедрена. Кроме того, SPAM может обнаружить небольшие изменения в данных, которые могут оставаться незамеченными другими методами [103, 104].

При использовании SPAM для стегоанализа следует иметь в виду, что этот метод может быть ограничен в случае, если скрытая информация была внедрена в необычные (шумные) области данных, нарушения в которых не входит за границы известной дисперсии статистики. Также, метод может быть более требовательным к вычислительным мощностям, нежели другие методы стегоанализа, особенно при анализе больших объемов данных.

В работах авторов [105-108] использованы метод SPAM в сочетании с алгоритмами усиления (boosting) для улучшения обнаружения скрытых сообщений в мультимедийных данных, которые были встроены с использованием различных методов стеганографии и с более сложными моделями взаимосвязи соседних пикселов, например, групповое понижение яркости цвета соседних пикселов.

В целом, метод SPAM продолжает оставаться важным инструментом в стегоанализе, и его применение может быть расширено и улучшено с помощью дальнейших исследований в этой области.

# 1.3.3 Метод Хи-квадрат

Хи-квадрат (χ²) [109] - это статистический метод, который используется для анализа распределения данных. В контексте стегоанализа, Хи-квадрат может быть использован для обнаружения скрытых сообщений в изображениях или других мультимедийных данных. Метод Хи-квадрат основывается на сравнении ожидаемого распределения данных с наблюдаемым распределением. Если наблюдаемое распределение отличается от ожидаемого, это может указывать на наличие скрытой информации. В контексте стегоанализа, Хи-квадрат может быть использован для анализа частоты появления пикселей в изображении или других мультимедийных данных. Используя этот метод, можно определить, являются ли эти частоты случайными или сформированы определенным образом. Если частоты появления пикселей не соответствуют ожидаемому распределению, это может указывать на наличие скрытой информации.

Одним из преимуществ метода Хи-квадрат является его универсальность, то есть возможность использовать его для обнаружения скрытых сообщений, встроенных с помощью различных методов стеганографии. Однако, этот метод также может давать ошибочные результаты:

* ошибка первого рода – случай, когда заполненный контейнер признается пустым.
* ошибка второго рода – случай, когда пустой контейнер признается заполненным.

Метод Хи-квадрат широко используется в области стегоанализа для обнаружения скрытых сообщений в различных типах мультимедийных данных, включая изображения, аудио и видео.

Существует множество исследований, которые используют этот метод для анализа качества различных методов стеганографии, а также для разработки новых методов обнаружения скрытых сообщений. Одной из работ в этой области является статья [110]. В этой статье авторы использовали метод Хи-квадрат (рис.1.13) для обнаружения скрытых сообщений, встроенных с использованием метода младшего значащего бита (LSB, НЗБ):

а) при последовательной записи в НЗБ дискретных коэффициентов;

б) при псевдослучайной записи в НЗБ дискретных коэффициентов.

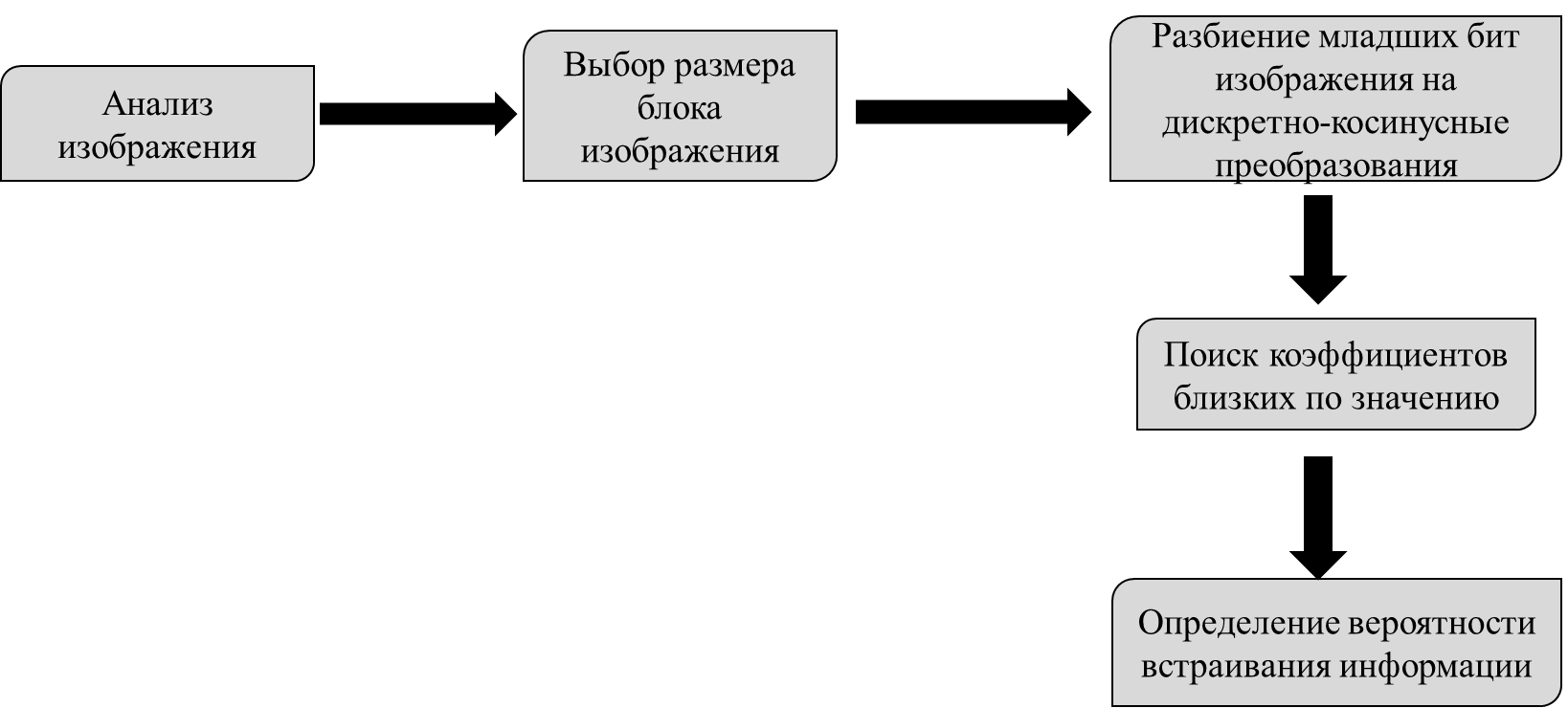


Рисунок 1.13- Алгоритм работы метода Хи-квадрат [110]

Для сравнительной характеристики вероятности обнаружения внедренных данных авторы использовали следующие методы стегоанализа: RS-анализ, метод анализа гистограмм, хи-квадрат, сигнатурный метод (Рис.1.14):



Рисунок 1.14 –Сравнительная статистика вероятности обнаружения скрытой информации: а) при последовательной записи в НЗБ дискретных коэффициентов; б) при псевдослучайной записи в НЗБ дискретных коэффициентов [110]

Согласно результатам эксперимента и утверждениям авторов с точки зрения защищенности информации предложенная методика обеспечивает стойкость и быстродействие выявления и предотвращения стеганографии.

В разработке [111] исследован принцип работы методов стегоанализа сообщений, внедренных с помощью метода замены LSB. В качестве контейнера применяли изображение. Показаны реализации следующих методов стегонализа: хи-квадрат и метод RS. Авторами проанализированы математические реализации данных методов и наглядно представлены результаты их работы в виде графиков (рис.1.15).

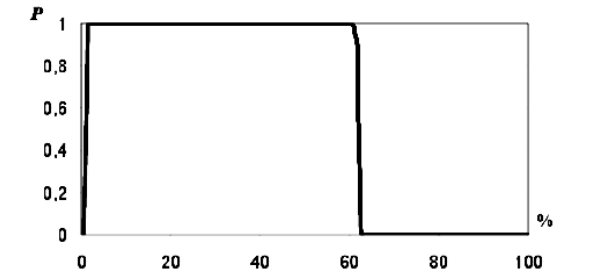


Рисунок 1.15- Вероятность встраивания по критерию Хи –квадрат при анализе стегоконтейнера, полученного методом последовательной замены [111]

Согласно результатам эксперимента, можно увидеть, что метод RS является достаточно эффективным методом стегоанализа, основанным на анализе соотношения между группами в цифровом изображении. Данный метод позволяет избежать недостатков, присущих методу анализа на основе Хи–квадрат, так как он не зависит от метода встраивания информации в пространственные области изображений.

Например, некоторые исследования [112-114] используют метод Хи-квадрат для обнаружения скрытых сообщений, встроенных в различные типы мультимедийных данных, такие как изображения, аудио и видео. Другие работы используют метод Хи-квадрат для оценки качества стеганографических методов, путем сравнения их эффективности в обходе методов стегоанализа.

Кроме того, существует также исследования, которые расширяют метод Хи-квадрат для обнаружения скрытых сообщений в более сложных сценариях, таких как мультимедийные данные с шумом или изменениями размера. В целом, метод Хи-квадрат является важным инструментом в стегоанализе, и его применение может быть расширено и улучшено с помощью последних исследований в этой области.

1.5 Выводы по разделу

В этой главе был приведен обзор существующих и известных методов стеганографии и стегоанализа. Наиболее популярными методами стеганографии являются:

* LSB метод;
* DCT;
* Вейвлет - преобразование.

В стегоанализе изображений наиболее эффективными методами считаются:

* RS анализ;
* Хи-квадрат;
* SPAM.

Было показано, что методы стеганографии имеют широкое применение в области защиты авторских прав и отслеживания траектории распространения цифрового объекта. Оценка эффективности алгоритма внедрения определяется объемом данных, которое встраивается в контейнер, при котором вероятность успешного стегоанализа мала.

Применение стегоанализа может быть найдено в области обеспечения безопасности объектов критической телекоммуникационной инфраструктуры в части контроля за передачей конфиденциальных данных.

# 2 ИНТЕРПОЛЯЦИЯ ИЗОБРАЖЕНИЯ В СТЕГАНОГРАФИИ

# 2.1 Анализ задачи

В сети Интернет передается огромное количество медиа контента. Большая часть этих данных является источником дохода его создателя и рассматривается как объект защиты авторского права. Учитывая легкость и нулевую стоимость воспроизводства (создания копии) любого файла возникает потребность отслеживать траекторию его пути (от создателя до конечного потребителя, в том числе нелицензионного). Одним из самых эффективных решений данной проблемы является применение методов стеганографии, которые используют секретные сообщения, встраиваемые в файл. Такие сообщения могут либо идентифицировать автора (цифровые водяные знаки), либо конечного потребителя (цифровой отпечаток пальца).

В научных публикациях встречаются работы, направленные на создание новых методов внедрения и на создание новых методов обнаружения (стегоанализа). Последние используются для выявления фактов утечки информации, например, через служебную переписку.

Таким образом, возникает острая необходимость анализа существующих методов внедрения и создания новых и эффективных методов внедрения скрытых сообщений. Так один из современных подходов стеганографии базируется на методах интерполяции. В данном исследовании изучается применение методов интерполяции для внедрения сообщения, которые, в некотором смысле, является дискретным аналогом голограммы, и обычно применяется для восстановления сигналов и изображений, подвергшихся воздействиям и приведшим к большой потере информации.

В данной главе проведен анализ эффективности методов внедрения скрытых сообщений в изображение: INMI и NMI [115].

В ходе работы было выполнено:

* Подготовка тестовых наборов изображений различного типа и разрешения для использования в качестве контейнеров для скрытой информации.
* Разработано программное обеспечение для внедрения скрытой информации в изображения с использованием известных ранее алгоритмов стеганографии.
* Произведена оценка качества внедрения информации в изображения.
* Оценена устойчивость рассматриваемых алгоритмов к атакам, использующим изменение цветовой палитры.
* Выполнено сравнение полученных результатов с другими существующими методами стеганографии.

# 2.2 Понятие интерполяции

Интерполяция изображений [116] – это процесс восстановления пикселей изображения, которые были потеряны или не были предоставлены, в целях улучшения качества изображения или изменения его размера.

Одним из самых распространенных методов интерполяции изображений является метод бикубической интерполяции, который использует окрестности 16 ближайших пикселей, чтобы определить значение пропущенного пикселя.

Интерполяция также может быть использована для увеличения размера изображения без потери качества. Этот процесс называется масштабированием. Наиболее распространенные методы масштабирования включают билинейную и бикубическую интерполяцию.

Однако, важно понимать, что использование интерполяции может привести к некоторой потере качества и ухудшению четкости изображения, особенно если она применяется несколько раз или в условиях низкой разрешающей способности изображения.

Бикубическая интерполяция [117] – это метод интерполяции, который используется для увеличения размера изображений, чтобы получить более гладкий и качественный результат, чем при использовании других методов интерполяции, таких как билинейная интерполяция. Бикубическая интерполяция использует более сложную математическую модель, чем билинейная интерполяция, для определения значений новых пикселей. Она основывается на кривых сплайнах, которые проходят через четыре ближайших пикселя в исходном изображении. Эти кривые позволяют определить значения новых пикселей, которые более точно соответствуют исходным данным.

Бикубическая интерполяция [118-120] использует 16 ближайших пикселей для вычисления значения нового пикселя, что позволяет получить более гладкое и качественное изображение, чем билинейная интерполяция. Однако, из-за более сложной математической модели, бикубическая интерполяция может быть более ресурсоемкой, чем другие методы интерполяции, и может потребовать больше времени для выполнения. В целом, бикубическая интерполяция является более точным и качественным методом интерполяции, чем билинейная интерполяция, и может быть полезной для увеличения размера изображений в задачах обработки изображений, таких как фотографии, видео и медицинские изображения.

Один из распространенных методов интерполяции для масштабирования изображений - это билинейная интерполяция [121]. Этот метод использует линейные интерполяционные функции для вычисления значений пикселей, которые не были предоставлены при изменении размера изображения. Билинейная интерполяция определяет значения пропущенных пикселей на основе четырех ближайших пикселей [122-124].

Для достижения наилучшего качества при масштабировании изображений, может потребоваться использование более сложных методов обработки, таких как алгоритмы супер-разрешения.

Ниже приведен краткий обзор нескольких новых работ, связанных с интерполяцией изображений:

1. Работа [125] представляет новый метод интерполяции, который использует глубокое обучение для генерации дополнительных кадров между существующими кадрами в видео. Метод основан на модели глубокого обучения, которая была обучена на большом наборе данных видео и позволяет генерировать более плавные и качественные видеокадры.
2. Статья [126] представляет новый метод интерполяции, который использует адаптивную свертку для комбинирования процессов интерполяции и восстановления изображения. Метод позволяет увеличить размер изображения и снизить шум, используя общую модель свертки, которая может адаптироваться к различным типам изображений.
3. Исследование [127] представляет новый метод интерполяции, который использует глубокое обучение для генерации интерполированных изображений. Метод основан на модели сверточных нейронных сетей, которая была обучена на большом наборе данных изображений и позволяет генерировать более качественные и плавные изображения при увеличении их размера.

Эти работы и многие другие [128-131] связанные с интерполяцией изображений продолжают исследовать новые методы и подходы для улучшения качества интерполированных изображений в различных областях, таких как обработка видео, компьютерное зрение, медицинская обработка изображений и т.д.

Одним из основных преимуществ стеганографии с применением методов интерполяции является относительно высокая устойчивость к атакам. Это связано с тем, что замена некоторых пикселей на значения, содержащие скрытую информацию, не изменяет значительно статистические свойства изображения и не приводит к заметным артефактам на изображении. Однако, методы интерполяции могут приводить к потере части информации в изображении, особенно при использовании более сложных методов интерполяции.

# 2.3. Общие сведения об интерполяции Лагранжа

Интерполяция Лагранжа [132] - это метод аппроксимации функции с помощью полинома, проходящего через заданные точки. Этот метод был разработан именно Жозефом Лагранжем в XVIII веке.

Интерполяция Лагранжа является распространенным методом обработки сигналов и изображения [133-136]. Пусть в отрезке  в заданных точках  известны значения функции . Вид многочлена порядка , , значение которого в точках узла равно значению функции  (Eq.2.1):

 (2.1)

Многочлен  (Eq.2.2):

 (2.2)

называется интерполяционным многочленом, построенным по точкам . С помощью интерполяционной формулы Лагранжа можно записывать в виде линейной комбинации значений интерполяции функции  в точках узлов (Eq 2.3):

 (2.3)

Учитывая все соответствующие условия [136], можно увидеть, что определяется как в формуле (Eq.2.4):

 (2.4)

Таким образом, интерполяционный многочлен Лагранжа имеет следующий вид (Eq.2.5):

 (2.5)

Интерполяция Лагранжа имеет некоторые ограничения, включая проблему погрешности интерполяции на концах интервала и проблему слишком высокой степени полинома, что может привести к неустойчивости при аппроксимации. Однако, метод Лагранжа все еще широко используется в научных и инженерных приложениях, благодаря своей простоте и интуитивной понятности.

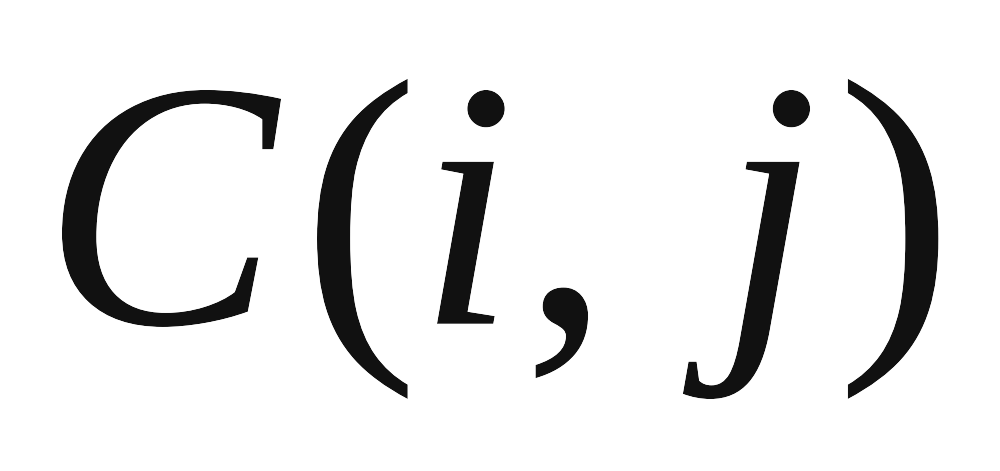
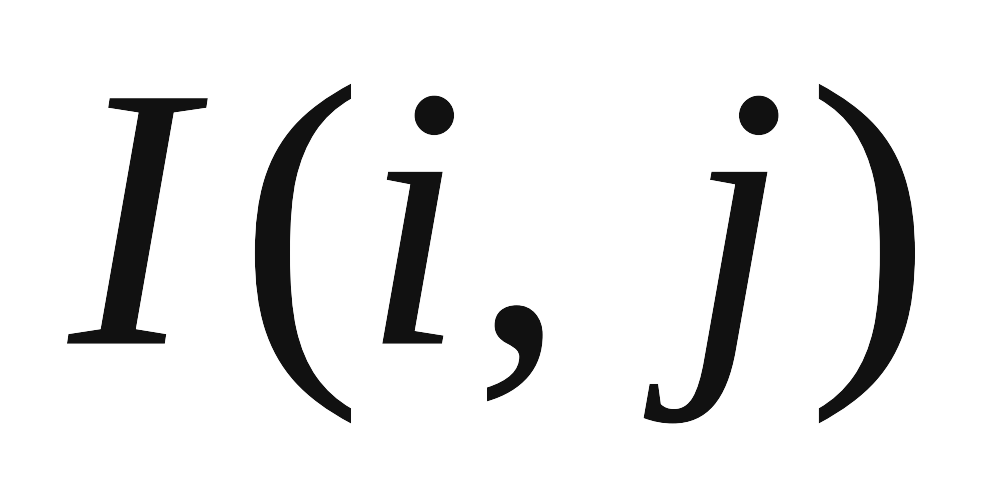
Интерполяция Лагранжа может использоваться в стеганографии для встраивания секретных сообщений в цифровые изображения. Один из подходов заключается в том, чтобы изменять значения пикселей изображения таким образом, чтобы они соответствовали значениям интерполяционного полинома Лагранжа, проходящего через секретное сообщение. Чтобы восстановить секретное сообщение, можно применить обратную операцию интерполяции Лагранжа, используя координаты выбранных пикселей и их измененные значения, чтобы вычислить значения секретного сообщения.

Однако, использование интерполяции Лагранжа для стеганографии может иметь некоторые ограничения, такие как возможность обнаружения внедрения секретного сообщения с помощью стегоанализа. Кроме того, этот метод может быть уязвим к атакам типа сжатия изображений, которые могут изменять значения пикселей и повредить секретное сообщение.

# 2.4 Метод NMI

Neighbor mean interpolation (NMI) [137]- это метод интерполяции, который используется для восстановления изображений с низким разрешением. Этот метод основан на идее усреднения значений яркости пикселей соседних пикселей. При использовании метода NMI пропущенный пиксель восстанавливается путем усреднения значений яркости соседних пикселей. Для каждого пропущенного пикселя вычисляется среднее значение яркости соседних пикселей, которое затем используется для восстановления пропущенного пикселя.

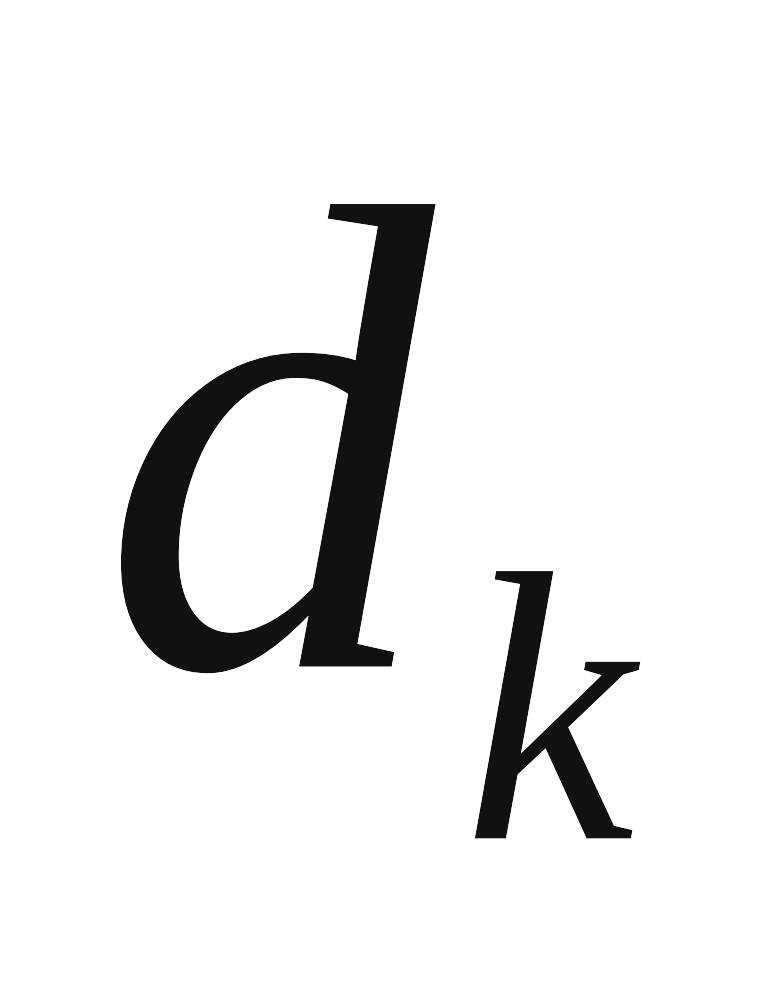
*Рассмотрим подробнее метод NMI*. В данном алгоритме интерполяция изображения осуществляется в следующем виде: пусть – значение пикселя оригинального изображения, тогда значение пикселя изображения-контейнера будет вычисляться по следующей форме (2.6):



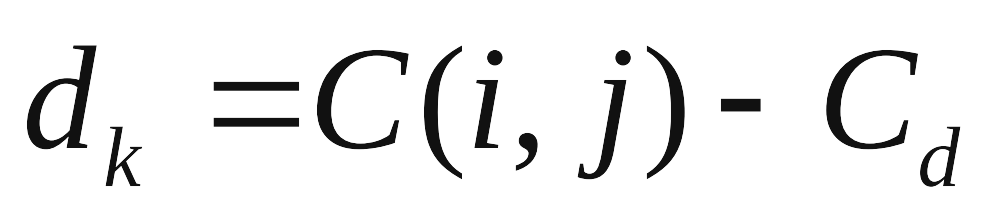
 (2.6)

где .

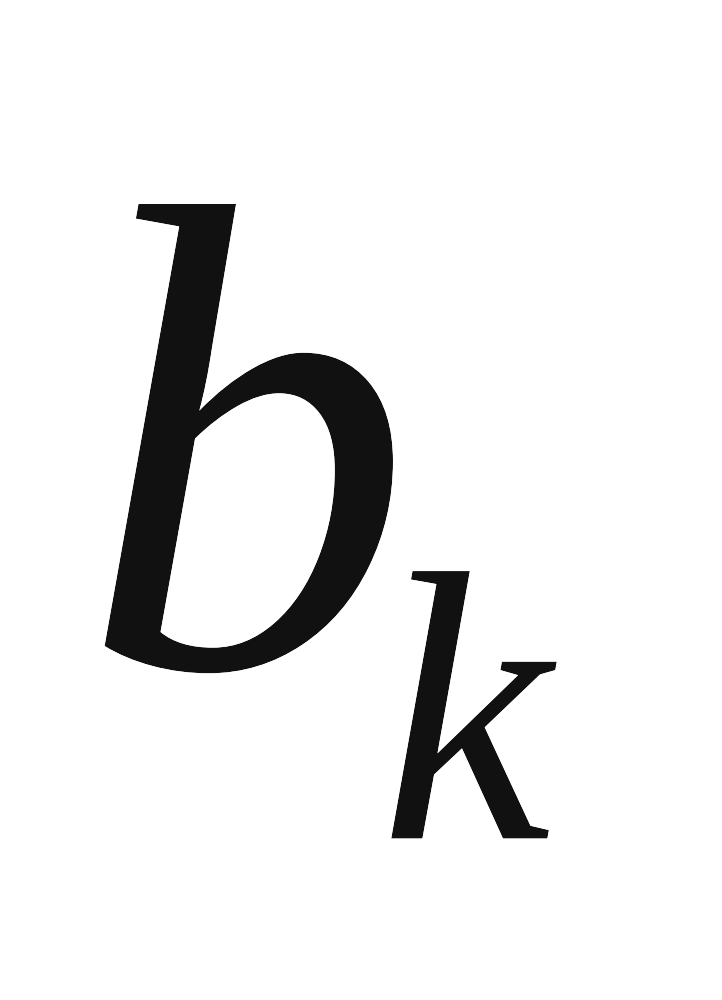
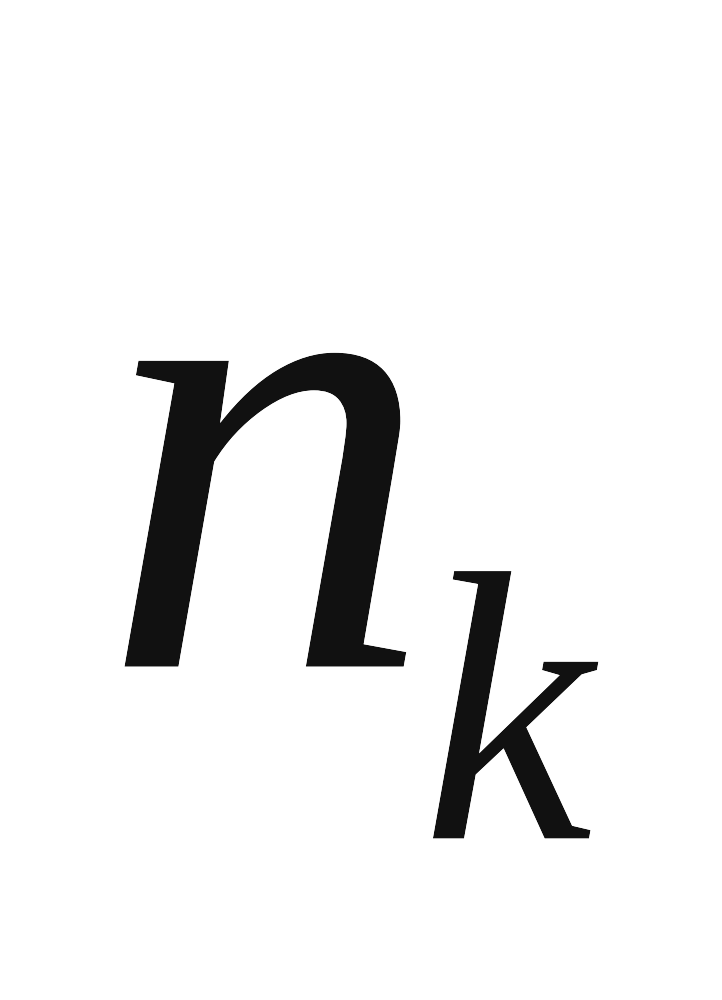
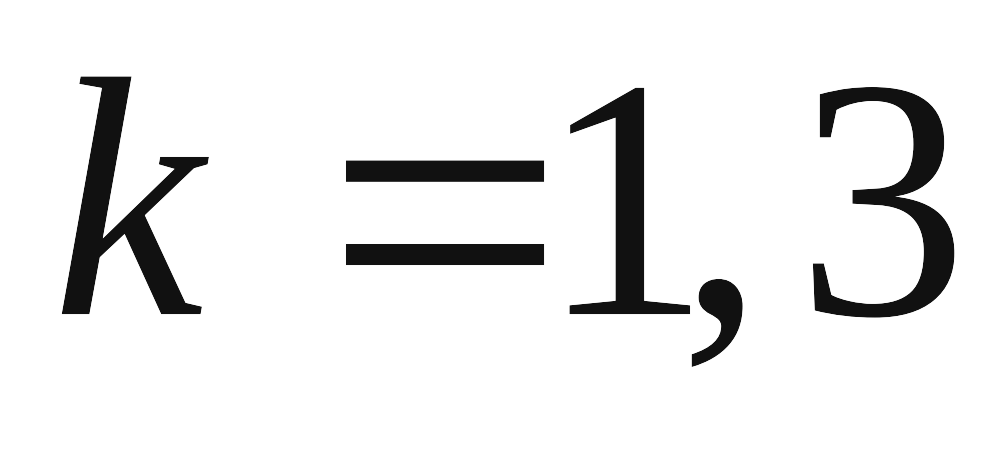
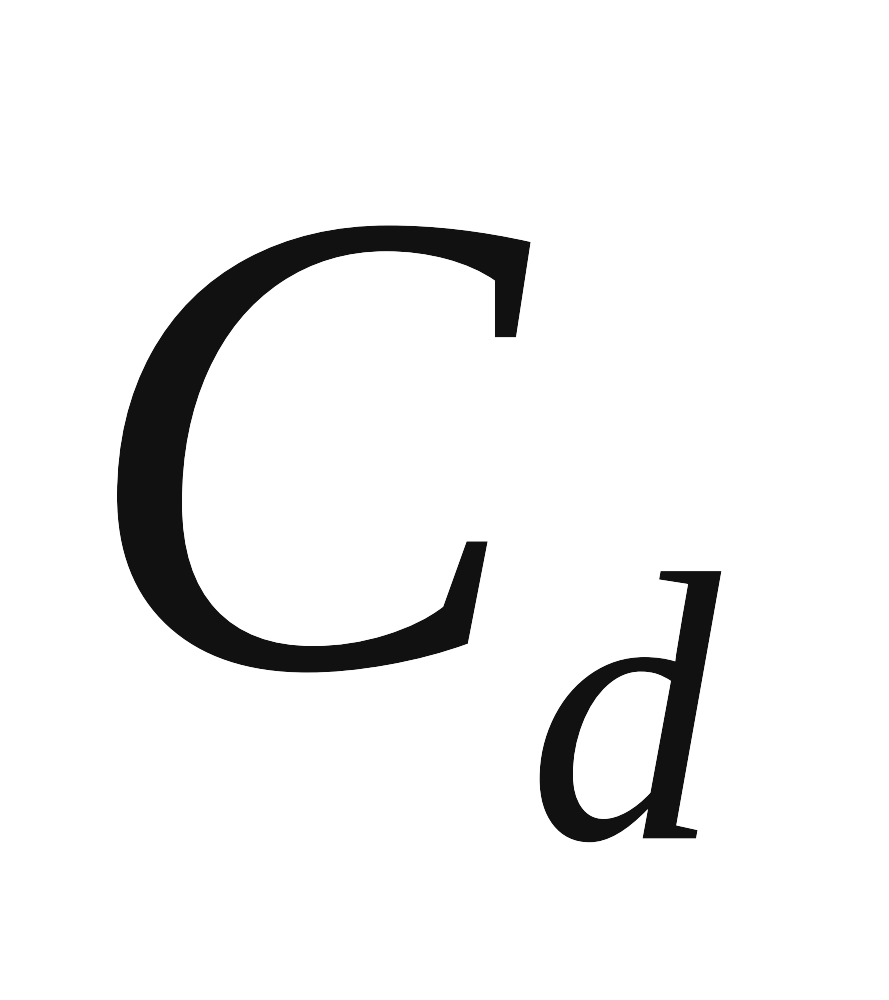
Значение пикселей стегоизображения рассчитываются следующим образом. Для каждого из непересекающихся блоков размером 2х2 пикселя требуется найти величину по формуле [2.7]



, (2.7)



где – левый верхний пиксель блока, , после чего рассчитывается количество битов , которое может быть встроено в блок и его целочисленные представление , и вычисляются значения соответствующих пикселей стегоизображения.



Емкость встраивания по данному методу оказались меньше, при этом возросло качество полученного стегоизображения.

Метод NMI является простым и эффективным методом интерполяции, который может быть использован для восстановления изображений с низким разрешением. Однако, как и любой другой метод интерполяции, он имеет свои ограничения, включая возможность искажения текстур и деталей изображения, особенно при восстановлении изображений с высокой степенью разрешения. Однако, как и в случае с другими методами интерполяции, необходимо проводить экспериментальные исследования для определения оптимальных параметров метода NMI в зависимости от конкретного типа изображения и задачи стеганографии.

Существует множество работ, посвященных методу интерполяции NMI. Некоторые из них представлены ниже:

В статье [138] авторы рассмотрели известный метод NMI, на основе которого они впервые построили новую схему скрытия данных на основе интерполяции изображений (IIDH). В своих исследованиях они использовали интерполяцию среднего значения соседей (NMI) для создания контейнера, с применением LSB внедрения и процесс оптимальной настройки пикселей вместо простого добавления для улучшения визуального качества стегоизображения. По результатам эксперимента: авторы разработали новых схем IIDH, основанные на подходах NMI и DH (LSB и APAP); PSNR всех схем IIDH были экспериментально и теоретически оценены. По мнению авторов статьи, их предложенная схема имеет более высокий PSNR, чем исходная схема NMI.

В статье [139] авторы предложили такой метод скрытие данных, который позволяет восстановить исходное изображение и реализуется с использованием метода интерполяции. Такой метод называется обратимым и реализуется путем корректировки интерполированных пикселей после применения интерполяции NMI. По два бита секретных данных выстраиваются в каждый интерполированный пиксель изображения контейнер, что приводит к среднему значению PSNR 26,94 дБ и полезной нагрузке 1,5 бит/с. Экспериментальные результаты доказывают, что предложенная схема авторами превосходит самые современные схемы с точки зрения качества изображения и скрытности.

В статье [140] разработан обратимый метод сокрытия данных, который использует интерполяцию и метод LSB. Под обратимостью понимается возможность восстановить исходное изображение без внедрения. Здесь, во-первых, используются методы интерполяции для увеличения и уменьшения масштаба изображения контейнера с целью повышения емкости и качества. Во-вторых, метод подстановки LSB используется для встраивания секретные данные. В рисунках 2.1 – 2.2 Приведены процесс встраивания замены LSB и пример метода интерполяции по соседним пикселям:

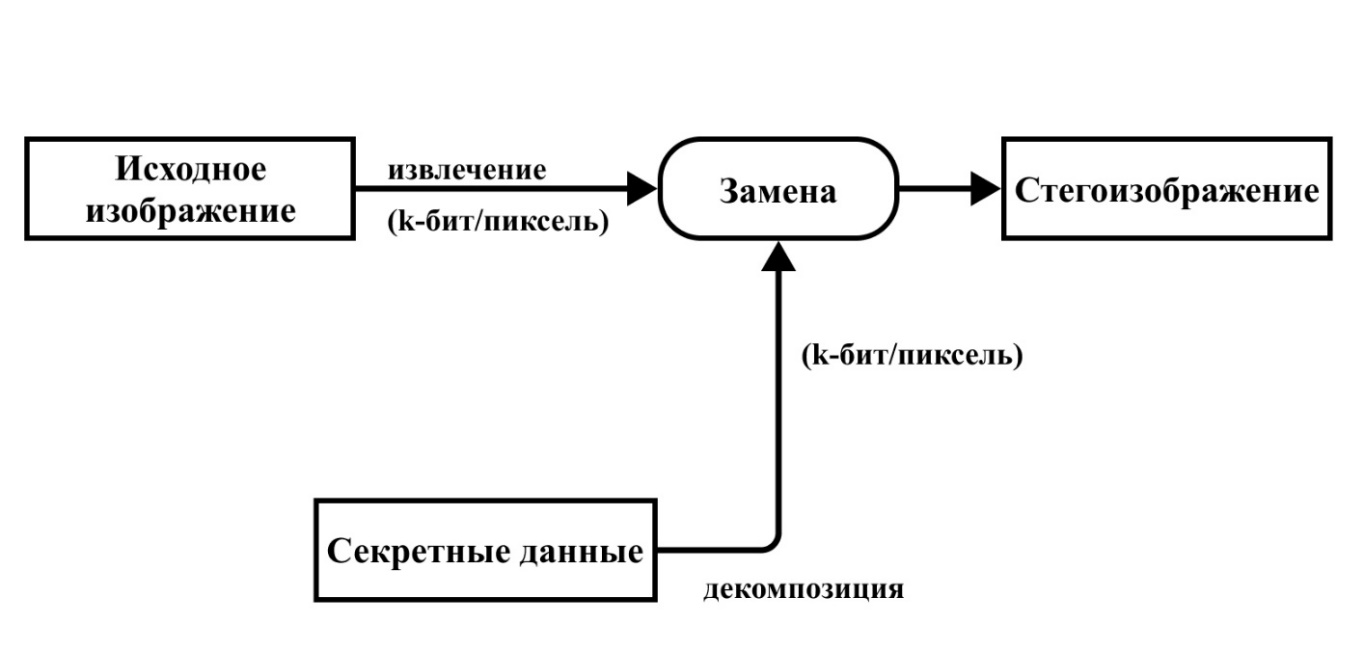


Рисунок 2.1 - Процесс встраивания замены LSB [140]

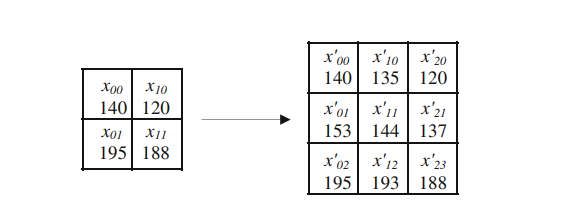


Рисунок 2.2 –Пример метода интерполяции по соседним пикселям [140]

Также согласно графику (рис 2.3) можно увидеть сравнение различных методов по емкости встраивания (бит):

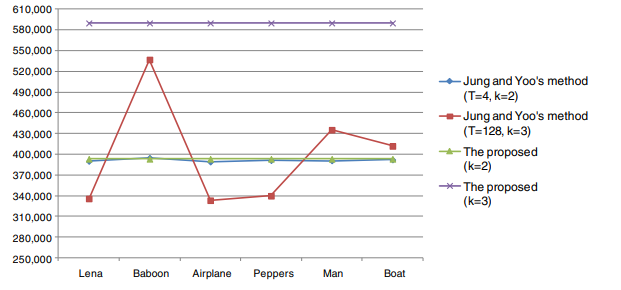


Рисунок 2.3 - Сравнение емкости встраивания (бит) [140]

В данной статье пиковое отношение сигнал/шум (PSNR) используется для измерения скрытности и емкости для объема встроенных данных. Экспериментальные результаты показывают, что предлагаемый метод позволяет внедрять большое количество секретных данных при сохранении высокого визуального качества. Результаты показали, что предложенный метод дает хорошие результаты (PSNR 37,54 дБ (k=3) и 43,94 дБ (k=2)) и работает быстрее, чем другие методы интерполяции.

Таким образом, метод NMI является популярным методом интерполяции в области восстановления изображений с низким разрешением. Различные улучшения и модификации метода NMI позволяют достигать лучших результатов в зависимости от конкретной задачи восстановления изображения.

# 2.5 Метод INMI

Improved neighbor mean interpolation (INMI) [137] - это метод интерполяции, который основан на улучшенной версии метода neighbor mean interpolation (NMI). Этот метод используется для восстановления изображений с низким разрешением путем усреднения значений яркости пикселей соседних пикселей.

В методе INMI соседние пиксели, расположенные вокруг пропущенного пикселя, разделяются на две группы: ближние и дальние пиксели. Ближние пиксели находятся в непосредственной близости от пропущенного пикселя, а дальние пиксели находятся на большем расстоянии. Для каждой группы пикселей вычисляется среднее значение яркости, которое затем усредняется для получения итогового значения яркости пропущенного пикселя.

Одним из преимуществ метода INMI является то, что он позволяет учитывать пиксели, лежащие в некоторой окрестности контекст изображения, при интерполяции, что может улучшить качество восстановленного изображения. Однако, как и любой другой метод интерполяции, INMI также имеет свои ограничения, включая возможность искажения текстур и деталей изображения, особенно при восстановлении изображений с высокой степенью разрешения.

Метод INMI может использоваться в стеганографии для восстановления изображений с низким разрешением, которые могут быть использованы для внедрения скрытой информации. Однако, как и в случае с другими методами интерполяции, необходимо проводить экспериментальные исследования для определения оптимальных параметров метода INMI в зависимости от конкретного типа изображения и задачи стеганографии.

Метод INMI демонстрирует более высокую точность интерполяции, чем классический метод ближайшего соседа, а также является более быстрым и эффективным, чем другие методы интерполяции, такие как билинейная и бикубическая интерполяция. Он может быть использован в различных задачах обработки изображений, таких как масштабирование, поворот, искажение и др. Метод интерполяции INMI был предложен относительно недавно, в 2018 году, поэтому исследований, посвященных этому методу, пока не так много. Однако уже было опубликовано несколько работ, в которых исследовалась эффективность метода INMI в различных задачах обработки изображений.

Например, в работе [141] было проведено сравнение различных методов интерполяции изображений, включая INMI, на основе качественной модели оценки изображений. Результаты исследования показали, что метод INMI демонстрирует высокую точность интерполяции и превосходит другие методы, такие как бикубическая интерполяция.

Например, в статье [142], авторы сравнивают различные методы интерполяции изображений, включая INMI, и описывают его высокую точность и быстродействие.

Также, в статье [143], авторы рассматривают применение метода INMI для увеличения размера изображений и показывают его преимущества в сравнении с другими методами интерполяции.

В статье [144], авторы предложили метод INMI для интерполяции изображений с низким разрешением. Ниже представлена схема извлечения данных (рис. 2.4):



Рисунок 2.4- Пример схемы извлечения [144]

Результаты экспериментов показали, что INMI, являющийся расширением NMI, имеет гораздо меньшие вычислительные затраты. Согласно результатам эксперимента, можно увидеть, что производительность нового предложенного метода интерполяции изображений с точки зрения качества изображения увеличена с 4,87% до 28,77% по сравнению с ENMI. Кроме того, экспериментальные результаты предложенной схемы сокрытия данных показывают, что полезная нагрузка увеличивается с 0,87% до 73,82% без увеличения искажения изображения по сравнению с другими методами.

В статье [145], представлены обратимые методы сокрытия данных с использованием методов интерполяции и проанализированы возможности встраивания и качество визуального изображения. Авторы представили базовую схему обратимого сокрытия данных на основе интерполяции (Рис.2.5)

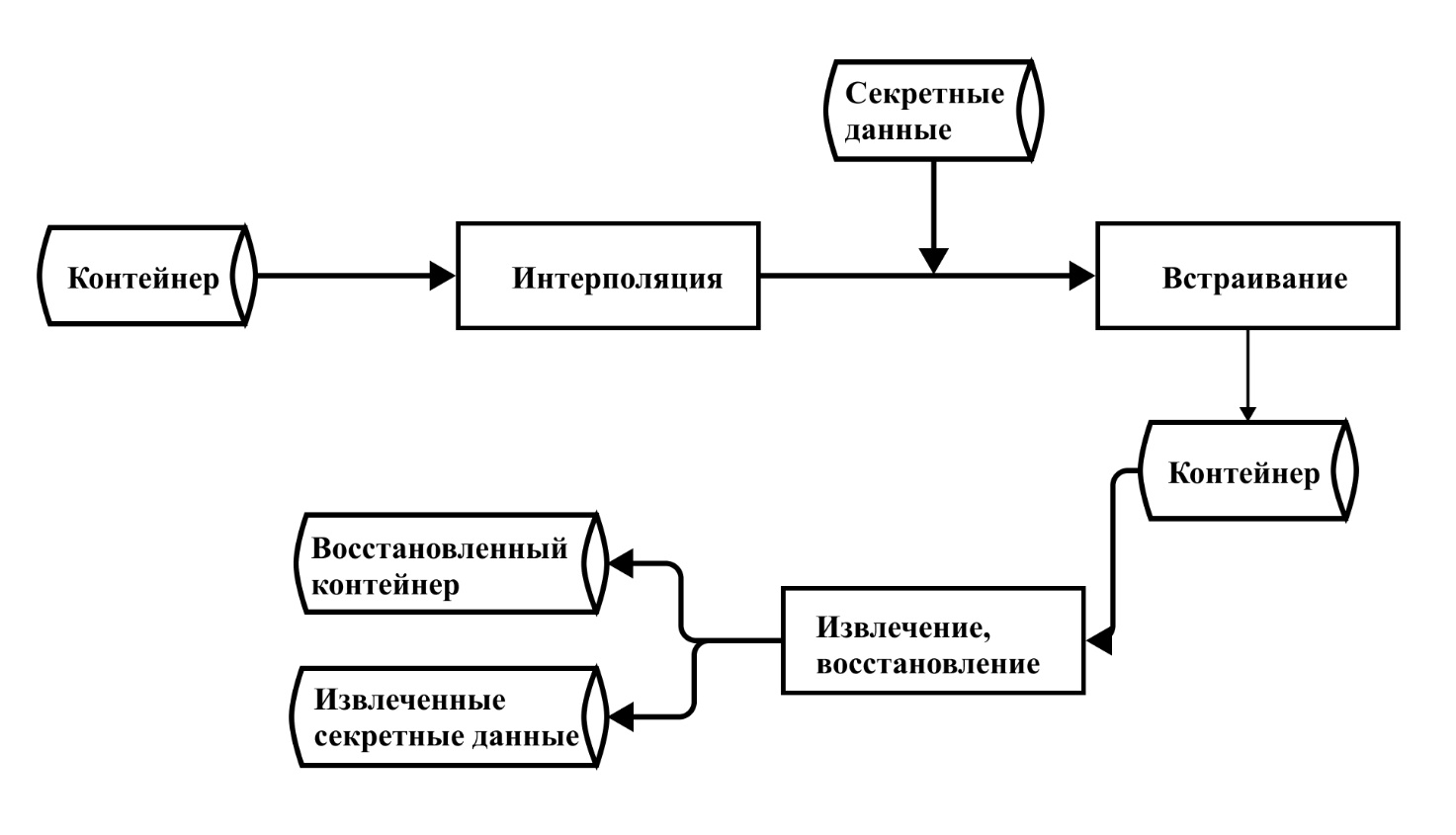
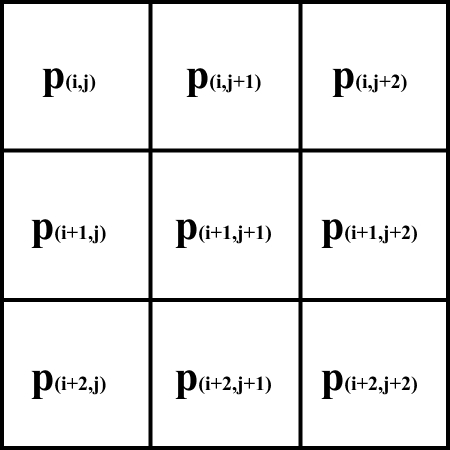
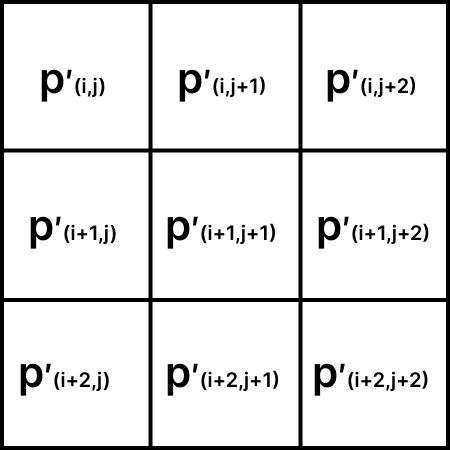


Рисунок 2.5 - Базовая схема обратимого сокрытия данных на основе интерполяции [145]

Также авторы наглядно показали значения пикселей изображения контейнера и стего-изображения (рис. 2.6):



а) пикселы контейнера



б) пикселы стегоконтейнера

Рисунок 2.6- Значения пикселей изображения контейнера и стего-изображения [145]

Авторами были рассмотрены ниже представленные методы (рис.2.7-рис.2.9):



Рисунок 2.7 – Пример метода Jung и Yoo [145]

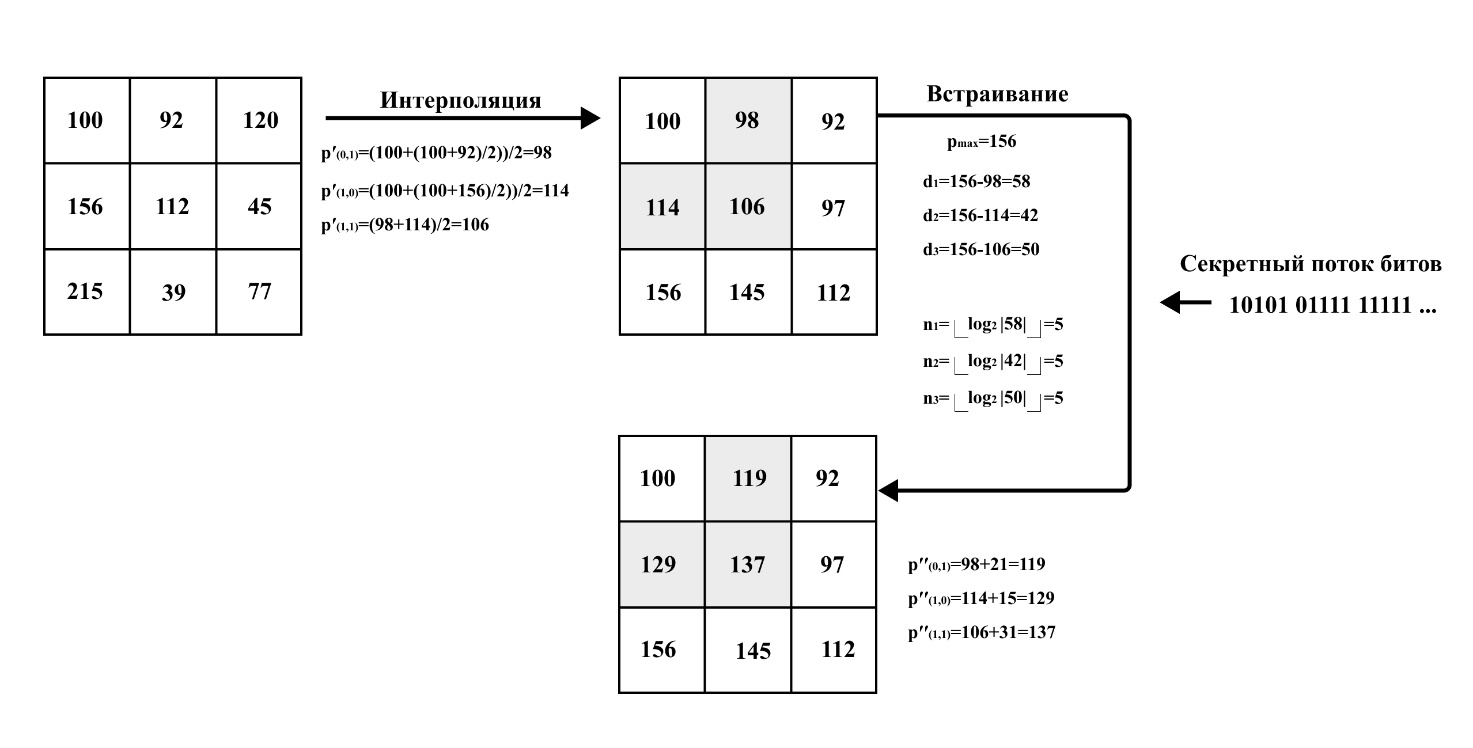


Рисунок 2.8 – Пример метода Lee и Huang [145]

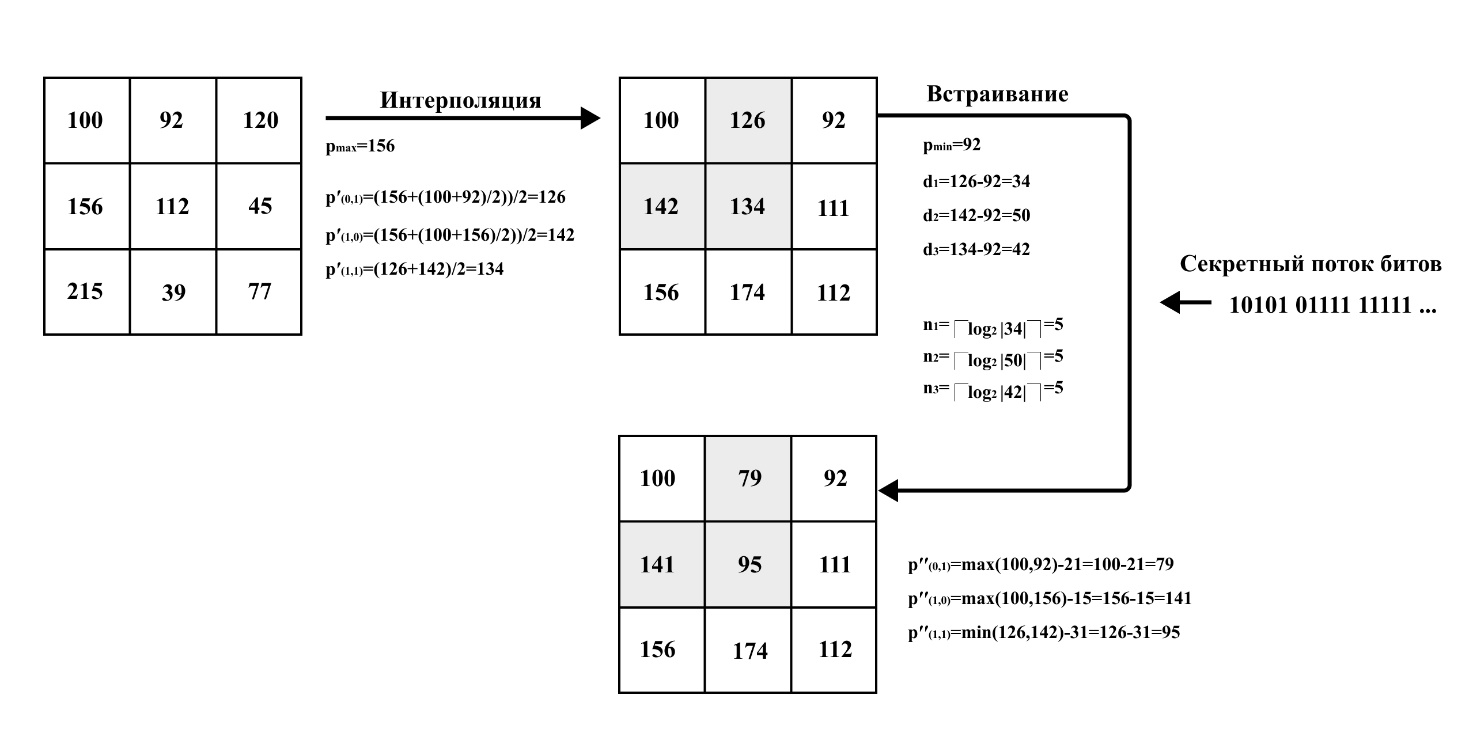


Рисунок 2.9 – Пример метода Hu и Li [145]

В качестве метрик авторами применены PSNR, универсальный индекс Q (для измерения качества визуального изображения) и общее количество встроенных битов или количество встроенных битов на пиксель (bpp) секретных данных на изображении.

Сравнительны результаты по методам показаны в таблице 2.1:

Таблица 2.1 Сравнение средней емкости встраивания и качества визуального изображения [145]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Метод | Усредненные значения | | | | |
| Емкость  (в битах) | Контейнер и интерполированное изображение | | Контейнер и стегоизображение | |
| PSNR (dB) | Q | PSNR (dB) | Q |
| Метод Jung и Yoo | 183,353 | 28.50 | 0.7488 | 28.16 | 0.7413 |
| Метод Lee и Huang | 253,572 | 27.13 | 0.7163 | 27.13 | 0.7278 |
| Метод Hu и LI | 155,488 | 28.35 | 0.7913 | 28.86 | 0.8356 |

В статье [146] представлен обзор основных алгоритмов встраивания информации в цифровые изображения с применением интерполяции, а также предложена собственная модификация INMI, основанного на использовании интерполяционного полинома Лагранжа второй степени для получения изображения – контейнера.

Изображение, полученное путем добавления дополнительных строк и столбцов пикселей к исходному изображению, рассматривали фрагментами по 5 пикселей, пронумерованных от 0 до 4.

Значения пикселей изображения-контейнера получают по следующей формуле (2.8):

 (2.8)

где  – номер пикселя во фрагменте из пяти пикселей.

Выполнено сравнение рассмотренных алгоритмов по таким характеристикам, как пиковое отношение сигнал/шум и максимальная ёмкость (таблица 2.2):

Таблица 2.2 Результаты вычислительных экспериментов с рассмотренными алгоритмами [146]

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Изображение | NMI | | INMI | | INP | | CRS | | Модификация | |
| Ёмкость | PSNR | Ёмкость | PSNR | Ёмкость | PSNR | Ёмкость | PSNR | Ёмкость | PSNR |
| Aerial | 1,17 | 25,24 | 1,46 | 24,15 | 1,88 | 25,32 | 2,35 | 23,59 | 1,44 | 24,65 |
| Airplane | 0,63 | 28,94 | 0,85 | 27,94 | 1,25 | 28,76 | 1,55 | 26,99 | 0,83 | 28,37 |
| Baboon | 1,26 | 22,57 | 1,56 | 22,13 | 2,04 | 22,67 | 2,49 | 21,58 | 1,52 | 22,24 |
| Barbara | 0,88 | 24,42 | 1,13 | 23,90 | 1,51 | 24,50 | 1,90 | 22,98 | 1,10 | 24,07 |
| Boat | 0,78 | 27,98 | 1,06 | 26,93 | 1,43 | 27,89 | 1,83 | 25,53 | 1,04 | 27,34 |
| Goldhill | 0,77 | 29,72 | 1,05 | 28,78 | 1,45 | 29,69 | 1,86 | 26,19 | 1,03 | 29,19 |
| House | 0,84 | 27,52 | 1,11 | 26,64 | 1,45 | 27,46 | 1,83 | 25,15 | 1,07 | 27,05 |
| Lena | 0,57 | 31,64 | 0,80 | 30,27 | 1,20 | 31,61 | 1,50 | 27,68 | 0,78 | 30,92 |
| Peppers | 0,55 | 30,40 | 0,77 | 29,42 | 1,21 | 30,18 | 1,51 | 27,64 | 0,76 | 29,69 |
| Stream | 1,21 | 25,16 | 1,53 | 24,35 | 1,99 | 24,99 | 2,47 | 22,96 | 1,51 | 23,89 |
| СРЗНАЧ | 0,87 | 27,36 | 1,13 | 26,45 | 1,54 | 27,31 | 1,93 | 25,03 | 1,11 | 26,74 |

В целом, метод INMI применяется в различных областях компьютерного зрения и обработки изображений, например, для увеличения разрешения изображений в медицинских и научных приложениях, в системах видеонаблюдения, а также в алгоритмах распознавания образов и идентификации лиц.

Таким образом, метод интерполяции INMI продемонстрировал высокую эффективность и точность в различных задачах обработки изображений и является перспективным направлением исследований в области интерполяции изображений.

# 2.6 Стегоанализ INMI метода

*Проведение RS анализа.* В статье [146] была рассмотрена модификация INMI метода, которая показала улучшение характеристики в виде увеличения значения PSNR при незначительном уменьшении ёмкости встраивания. Максимальное количество информации, которое можно записать в контейнер при использовании метода внедрения называется эмпирической емкостью контейнера. Анализирующая программа RS – анализа выдает количество встроенной информации () в процентах от эмпирической емкости контейнера, которая высчитывается как при LSB-встраивании (2.9):

 бит (2.9)

По значению  можно судить о том, заполнен был контейнер или пуст: при RS классифицирует контейнер как заполненный. Также упомянем о двух гипотезах:означающая, что контейнер содержит стегосообщение, и альтернативная ей гипотеза  означающая, что контейнер не содержит встроенной информации. Правило решения заключается в том, что каждому контейнеру сопоставляется одна из двух гипотез. Здесь возможны два типа ошибок: ошибка первого рода (type I error), которая заключается в установлении гипотезы , когда контейнер пуст, и ошибка второго рода (type II error), когда принято решение  при наличии встроенной информации в контейнере.

В настоящем исследовании [115] проведен RS анализ данной модификации INMI. Схема эксперимента представлена на рисунке 2.10. Исходный контейнер масштабировали при помощи метода интерполяции. Затем, в новые (интерполированные) пикселы встраивалось секретное сообщение. Далее стегоконтейнер подавался на вход RS классификатору.

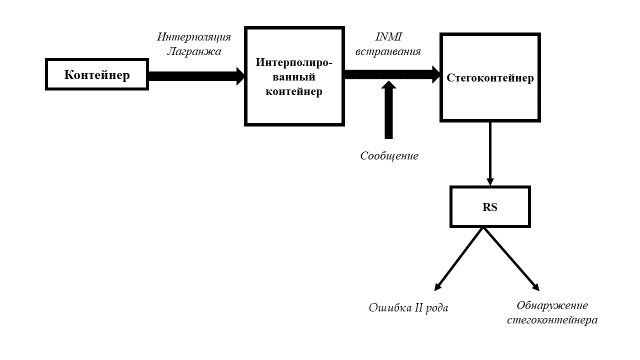


Рисунок 2.10 - Схема экспериментальных исследований

Требовалось определить емкость контейнеров с точки зрения процента вовлечения младших бит матрицы изображения при низкой вероятности обнаружения. Также это необходимо для сравнения полученных результатов стегоанализа с результатами других методов встраивания, рассмотренных в [147].

Исходя из рассмотренного алгоритма INMI, мы определили, что максимальная емкость контейнера составляет 21% и зависит от изображения.

Проведя исследование на наборе из 800 изображений размера 225х225, мы определили, что 60 из них имеют максимальную емкость. Результаты подсчета ошибки первого рода приведены в таблице 2.3, где показано, что ошибка составляет 0%. Результаты RS анализа показаны в таблице 2.4, по которой можно видеть, что метод является устойчивым к RS анализу.

Соответственно результаты исследования для полного набора из 800 изображений, при среднем проценте заполнения 12%, приведены в таблицах 2.3-2.4.

Таблица 2.3. RS анализ на наборе пустых контейнеров размером 225х225 (60 картинок)

|  |  |  |  |
| --- | --- | --- | --- |
| Результаты классификации | | | |
| Определенный объем внедрения | 0% | 1-4% | 5% и более |
| Доля файлов | 54 | 46 | - |

Таблица 2.4 RS анализ на наборе контейнеров, заполненных с помощью метода интерполяции на 21%.

|  |  |  |  |
| --- | --- | --- | --- |
| Результаты классификации | | | |
| Определенный объем внедрения | 0% | 1-4% | 5% и более |
| Доля файлов | 53 | 47 | - |

Результаты стегоанализа, приведенные в табл. 2.5 и 2.6 свидетельствуют о том, что процент обнаружения встроенной информации по предложенному методу приблизительно равен проценту файлов, в которых была ошибка первого рода.

Таблица 2.5. RS анализ на наборе пустых контейнеров (800 картинок)

|  |  |  |  |
| --- | --- | --- | --- |
| Результаты классификации | | | |
| Определенный объем внедрения | 0% | 1-4% | 5% и более |
| Доля файлов | 61 | 39 | - |

Таблица 2.6. RS анализ на наборе контейнеров, заполненных с помощью метода интерполяции на 12%.

|  |  |  |  |
| --- | --- | --- | --- |
| Результаты классификации | | | |
| Определенный объем внедрения | 0% | 1-4% | 5% и более |
| Доля файлов | 51,5 | 48 | 0,5 |

Очевидно, что чем меньше информации мы встраиваем в изображение, тем меньше вероятность появления обнаружимых признаков в результате процесса внедрения [148]. Как мы можем видеть из полученных результатов, метод INMI по стойкости к RS атаке сопоставим со стегосистемой способа перестановок для растровых изображений, рассмотренной в [147]. Процент изменения младших бит в сравниваемых методах примерно одинаковый и отличается в различных реализациях стегосистемы, рассмотренной там же. Однако, фактически в методе интерполяции задействованы не только младшие биты, поэтому имеет смысл провести также специфический метод анализа степени сжатия пустых и заполненных контейнеров.

*Энтропийный анализ.* В работах [149] был показан подход, позволяющий выявлять факт внедрения сообщения при помощи архиватора. Допустим имеется контейнер, который мы получили самостоятельно и знаем, что это стегоконтейнер. Добавим к нему подозрительный контейнер, который нужно проанализировать и сожмем его. Архиватор сработает так: будет собрана статистика и часто встречающиеся последовательности получают короткие кодовые слова, а редко встречающиеся – длинные. Чем лучше собрана статистика, тем лучше будет сжатие. Если исходный и добавленный контейнеры имеют одно происхождение (например, стего), то статистика у них будет совпадать и добавленный контейнер сожмется также хорошо, как и исходный. Если же они имеют разное происхождение, то статистические свойства у них отличаются, из-за чего сжатие добавленной части будет значительно хуже. Таким образом архиватор применяется для установления статистической взаимосвязи.

В работе [150] была доказана эффективность такого метода для абстрактных контейнеров, которые являются порождением стационарного и эргодического временного источника. Однако, когда мы рассматриваем реальный контейнер заранее не знаем, является ли он эргодическим или стационарным, т.к. не существует адекватных моделей ни одного из типов данных. Данный подход был успешно развит в ряде работ для текстовых контейнеров [151] и для изображений [147, 151]. Тем не менее оставался открытым вопрос о границах применимости. В настоящем исследовании данная граница была найдена.

Известно, что исходный контейнер и добавляемая в него информация статистически независимы, поэтому при добавлении скрытых данных в контейнер, размер его при сжатии вырастает по сравнению с размером сжатого исходного пустого контейнера [147]. Изменения степени сжатия в обратную сторону также свидетельствует о признаках изменения. Было проведено исследование, в котором определим коэффициенты сжатия пустых и заполненных контейнеров. Коэффициент сжатия — основная характеристика алгоритма сжатия. Она определяется как отношение объёма исходных несжатых данных к объёму сжатых данных.

В таблицах 2.7-2.8 показаны результаты сравнения коэффициентов сжатия для пустых и заполненных контейнеров. Здесь- последовательность байтов данных пустого контейнера,- коэффициент сжатия последовательности архиватором ZIP. — последовательность байтов данных заполненного контейнера, - коэффициент сжатия последовательностиархиватором RAR,  - разность между коэффициентами сжатия отрезков последовательностейи*.* Для удобства в таблицах представлены только часть результатов, так как они однообразны для всей выборки файлов.

Таблица 2.7. Сравнение разностей коэффициентов сжатия для пустых и заполненных контейнеров

|  |  |  |  |
| --- | --- | --- | --- |
| Название файла | ,  МБ | ,  МБ |  |
| 001.BMP | 0,133 | 0,134 | 0,02 |
| 002.BMP | 0,137 | 0,138 | 0,01 |
| 003.BMP | 0,086 | 0,086 | 0,00 |
| 004.BMP | 0,105 | 0,106 | 0,01 |
| 005.BMP | 0,151 | 0,153 | 0,00 |
| 006.BMP | 0,033 | 0,033 | 0,00 |
| 007.BMP | 0,146 | 0,148 | 0,02 |
| 008.BMP | 0,125 | 0,126 | 0,01 |
| 009.BMP | 0,117 | 0,118 | 0,01 |
| 010.BMP | 0,072 | 0,072 | 0,00 |
| 117.BMP | 0,065 | 0,065 | 0,00 |
| 140.BMP | 0,059 | 0,059 | 0,00 |
| 180.BMP | 0,129 | 0,131 | 0,02 |
| 222.BMP | 0,134 | 0,135 | 0,01 |
| 244.BMP | 0,102 | 0,103 | 0,01 |
| 25.BMP | 0,135 | 0,136 | 0,01 |
| 250.BMP | 0,072 | 0,072 | 0,00 |
| 258.BMP | 0,129 | 0,131 | 0,02 |
| 317.BMP | 0,071 | 0,072 | 0,01 |
| 37.BMP | 0,133 | 0,134 | 0,01 |
| 388.BMP | 0,108 | 0,109 | 0,01 |
| 405.BMP | 0,086 | 0,086 | 0,00 |
| 465.BMP | 0,132 | 0,134 | 0,02 |
| 521.BMP | 0,098 | 0,099 | 0,01 |
| 528.BMP | 0,117 | 0,118 | 0,01 |
| 610.BMP | 0,100 | 0,100 | 0,00 |
| 68.BMP | 0,150 | 0,152 | 0,02 |
| 72.BMP | 0,105 | 0,107 | 0,02 |
| 752.BMP | 0,147 | 0,149 | 0,02 |
| 785.BMP | 0,114 | 0,115 | 0,01 |
| 796.BMP | 0,069 | 0,069 | 0,00 |
|  |  |  |  |
| *Среднее значение* | *0,108* | *0,109* | *0,01* |

Таблица 2.8. Сравнение разностей коэффициентов сжатия для пустых и заполненных контейнеров (по архиватору RAR)

|  |  |  |  |
| --- | --- | --- | --- |
| Название файла | ,  МБ | ,  МБ | , |
| 001.BMP | 0,099 | 0,101 | 0,02 |
| 002.BMP | 0,104 | 0,107 | 0,03 |
| 003.BMP | 0,068 | 0,069 | 0,01 |
| 004.BMP | 0,088 | 0,090 | 0,02 |
| 005.BMP | 0,114 | 0,117 | 0,03 |
| 006.BMP | 0,029 | 0,029 | 0,00 |
| 007.BMP | 0,110 | 0,114 | 0,04 |
| 008.BMP | 0,091 | 0,094 | 0,03 |
| 009.BMP | 0,089 | 0,091 | 0,02 |
| 010.BMP | 0,058 | 0,058 | 0,00 |
| 117.BMP | 0,049 | 0,049 | 0,00 |
| 140.BMP | 0,041 | 0,041 | 0,00 |
| 180.BMP | 0,093 | 0,096 | 0,03 |
| 222.BMP | 0,091 | 0,093 | 0,02 |
| 244.BMP | 0,078 | 0,080 | 0,02 |
| 25.BMP | 0,089 | 0,091 | 0,02 |
| 250.BMP | 0,060 | 0,060 | 0,00 |
| 258.BMP | 0,099 | 0,102 | 0,03 |
| 317.BMP | 0,056 | 0,057 | 0,01 |
| 37.BMP | 0,101 | 0,104 | 0,03 |
| 388.BMP | 0,074 | 0,076 | 0,02 |
| 405.BMP | 0,066 | 0,067 | 0,01 |
| 465.BMP | 0,100 | 0,103 | 0,03 |
| 521.BMP | 0,071 | 0,073 | 0,02 |
| 528.BMP | 0,087 | 0,089 | 0,02 |
| 610.BMP | 0,085 | 0,087 | 0,02 |
| 68.BMP | 0,112 | 0,115 | 0,03 |
| 72.BMP | 0,076 | 0,079 | 0,03 |
| 752.BMP | 0,108 | 0,112 | 0,04 |
| 785.BMP | 0,079 | 0,080 | 0,01 |
| 796.BMP | 0,057 | 0,058 | 0,01 |
|  |  |  |  |
| *Среднее значение* | *0,082* | *0,083* | *0,01* |

Значение , полученное в результате данного исследования, близко к нулю, что означает сходство статистических структур пустых и заполненных контейнеров. Следовательно, их трудно отличить, применяя такой подход стегоанализа.

# 2.7 Выводы по разделу

В рамках диссертации реализован и исследована наилучшая модификация известного алгоритма INMI. Был проведен стегоанализ данного метода встраивания, получены результаты, которые можно сравнить со стегоанализом методов, рассмотренных в [147].

На основании рассмотренного алгоритма INMI, мы определили, что максимальная емкость контейнера составляет 21% и зависит от изображения. По результату исследования на наборе из 800 изображений размера 225х225, мы определили, что 60 из них имеют максимальную емкость. Полученные результаты подсчета ошибки первого рода показали, что ошибка составляет 0%.

Итоговые результаты RS анализа показаны в таблице 2, по которой можно видеть, что метод INMI является устойчивым к стегоанализу RS и сопоставим по стойкости и ёмкости со стегосистемой, предложенной Мерзляковой Е.Ю. в [147].

Был проведен также специфический метод анализа степени сжатия пустых и заполненных контейнеров. Также в исследовании были определены коэффициенты сжатия пустых и заполненных контейнеров. Значение разностей коэффициентов  полученное в результате данного исследования, близко к нулю, что означает сходство статистических структур пустых и заполненных контейнеров.

# 3 ПРИМЕНЕНИЕ КРИВЫХ БЕЗЬЕ В СТЕГАНОГРАФИИ

# 3.1 Кривая Безье

Кривая Безье - это математическая кривая, которая определяется точками начала и конца отрезка и контрольными точками, которые определяют форму кривой. Кривая Безье [152] названа в честь французского инженера Пьера Безье, который разработал этот метод в 1962 году во время работы над проектированием автомобильных кузовов. Кривые были названы именем Безье, а именем де Кастельжо назван разработанный им рекурсивный способ определения кривых (De Casteljau's algorithm). Кривая Безье может быть использована для создания плавных и сложных кривых, которые могут использоваться в различных областях, таких как графический дизайн, анимация и компьютерное моделирование.

Основная идея метода Безье заключается в том, чтобы определить форму кривой, используя контрольные точки. На сегодняшний день формулы кривой Безье используются во многих отраслях науки, также многие ученые ведут исследования в этой области [153-158].

Кривая Безье является частным случаем многочленов Бернштейна, представляет собой параметрическую кривую и определяется следующим выражением (Eq.3.1):

  (3.1)

где

– количество опорных точек;

– номер опорной точки;

 – шаг;

 – координата опорной точки;

 – базисная функция кривой Безье.

Координаты кривой описываются в зависимости от параметра :

Для двух точек (3.2): При n = 1 кривая представляет собой отрезок прямой линии, опорные точки P0 и P1 определяют его начало и конец (рис.3.1).

 (3.2)

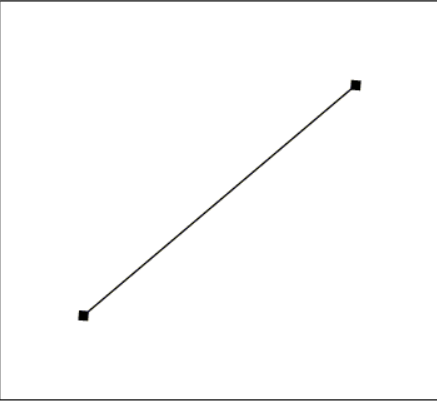


Рисунок 3.1 –Линейная кривая Безье [159]

Для трёх точек (3.3): квадратичная кривая Безье (n = 2) задаётся тремя опорными точками: P0, P1 и P2 (рис.3.2)

 (3.3)

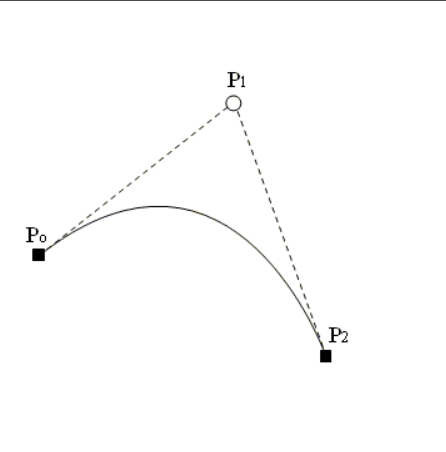


Рисунок 3.2 -Квадратичная кривая Безье (n = 2) задаётся тремя опорными точками: P0, P1 и P2 [159]

Для четырёх точек (3.4): Четыре опорные точки P0, P1, P2 и P3, заданные в 2- или 3-мерном пространстве, определяют форму кривой (рис.3.3)

 (3.4)

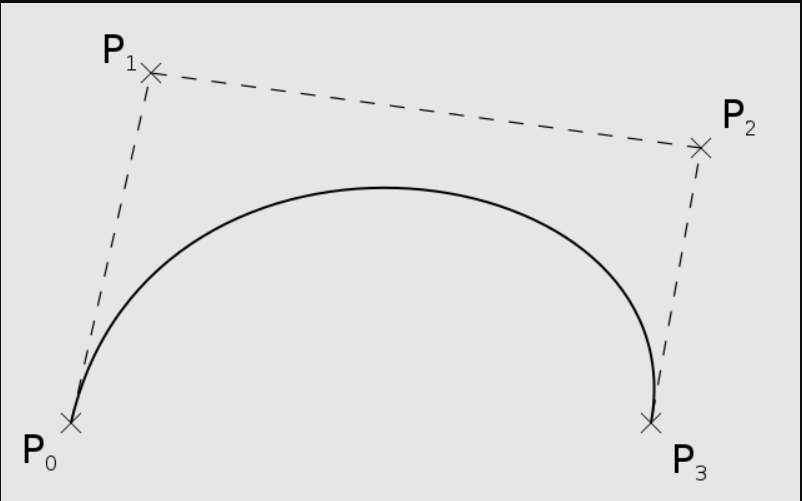


Рисунок 3.3 –Кубическая кривая Безье [159]

Вместо  нужно подставить координаты  -й опорной точки (). Эти уравнения векторные, то есть на для каждой из координат (например, для 3 точек):

 (3.4)

Вместо  подставляются координаты трёх опорных точек, и в то время как t пробегает множество от 0 до 1, соответствующие значения () как раз и образуют кривую.

Можно заметить, что: а) точки не всегда лежат на кривой; б) степень кривой равна числу точек минус один: для двух точек — это линейная кривая (т.е. прямая), для трёх точек — квадратичная кривая (парабола), для четырёх — кубическая; в) кривая всегда находится внутри выпуклой оболочки, образованной опорными точками.

Чем больше контрольных точек используется при определении кривой Безье, тем сложнее и точнее форма кривой может быть определена. Поэтому поступают очень просто — разбивают точки на группы по 4 штуки, строят для каждой из них кривую Безье и соединяют полученные сегменты в одну кривую. Это гораздо проще с точки зрения поддержки и расчётов. Единственная проблема — полученная кривая будет «не очень гладкой» на границах сегментов [159].

Кривая Безье имеет ряд преимуществ перед другими способами определения кривых, такими как возможность создавать плавные и сложные кривые, которые могут быть легко изменены при необходимости. Кривые Безье также могут быть просто аппроксимированы с помощью компьютерных алгоритмов, что делает их удобными для использования в компьютерной графике.

Кривые Безье могут быть использованы в сокрытии данных для создания стегоконтейнера, который может содержать секретные данные. Одним из способов использования кривых Безье в сокрытии данных является разбиение изображения на маленькие фрагменты, называемые блоками. Каждый блок может быть аппроксимирован кривой Безье, которая может быть использована для кодирования секретных данных. Для сокрытия данных можно изменять параметры кривой, такие как контрольные точки, чтобы создать изменения в блоках, которые невидимы невооруженным глазом, но могут быть использованы для хранения секретных данных.

Сокрытие данных с использованием кривых Безье может быть выполнено с помощью различных методов. Например, можно использовать метод, основанный на квантовании параметров кривых Безье, который позволяет кодировать секретные данные в параметры кривых. Другой метод, основанный на использовании множественных кривых Безье, позволяет скрыть секретные данные в нескольких кривых. Кривые Безье также могут использоваться для сокрытия данных в звуковых и видеофайлах. Например, можно использовать кривые Безье для создания стегоконтейнера, который может хранить секретные данные в формате звуковых или видеоданных [160].

Однако, как и в любой другой технике сокрытия данных, использование кривых Безье для сокрытия данных имеет свои ограничения и уязвимости, и может быть обнаружено с помощью методов стегоанализа. Исследования использования кривой Безье в сокрытии данных являются довольно распространенным направлением в области криптографии. Ниже приведен обзор нескольких статей на эту тему:

1. В статье [161] исследователи представили новый метод сокрытия информации в изображениях с использованием кривой Безье. Их подход основан на замене пикселей изображения на точки на кривой Безье, которая строится в соответствии с ключом шифрования. Результаты экспериментов показали эффективность метода в сокрытии информации и низкую вероятность обнаружения.
2. В работе [162] авторы использовали кривую Безье первого порядка. Они разработали уравнение кривой Безье, подходящее для рассеяния значений цвета в процессе обработки изображения. Этот алгоритм (имитирующий кривую Безье первого порядка при кодировании изображений) полностью отличается от предыдущих алгоритмов, использовавших то же самое (кривая Безье), с точки зрения метода, используемого при шифровании. Результаты экспериментов показали эффективность метода в сокрытии информации
3. В статье [163] исследователи использовали кривую Безье в сочетании с усовершенствованным алгоритмом криволинейного преобразования для сокрытия данных в медицинских изображениях. Они предложили новый метод, который основан на замене частей изображения на соответствующие значения на кривой Безье, построенной с использованием криволинейного преобразования. Результаты экспериментов показали эффективность метода и его способность сохранять качество изображения.
4. В работе [164] статье предлагается эффективная и настраиваемая схема представления визуальных изображений путем объединения гиперхаотической системы 6D, измерения сжатия и встраивания кривой Безье. Экспериментальные результаты моделирования и всесторонний анализ производительности показывают, что схема, предложенная авторами, обладает высоким качеством дешифрования, визуальной безопасностью, надежностью и операционной эффективностью.

# 3.2 Анализ существующих методов, базирующихся на кривой Безье

Ранее идея применения интерполяции уже встречалась во многих работах. Использовались различные алгоритмы интерполяции. В статье [165] авторы исследуют применение кривой Безье в качестве инструмента для построения цифрового водяного знака. Авторы предлагают новый метод, который использует кривую Безье для создания ЦВЗ с повышенной устойчивостью к атакам.

Параметрическое уравнение кубической кривой Безье выглядит следующим образом: . Чтобы однозначно определить эту кривую, требуются четыре точки:  и . На рисунке 3.4 показана кубическая кривая Безье. Кривая состоит из трех сегментов, первый из которых отмечен дополнительными красными линиями, второй сегмент отмечен оранжевыми линиями, а третий отмечен зелеными линиями. Линии нанесены на контрольные точки для наглядности, чтобы продемонстрировать, как расположение контрольных точек влияет на внешний вид кривой.

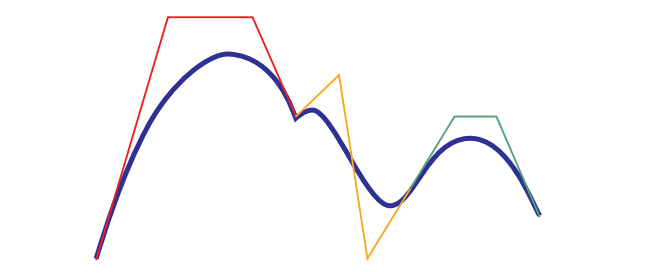


Рисунок 3.4 - Кубическая кривая Безье, состоящая из трех сегментов [165]

Авторы на рисунках 3.5 – 3.6 предложили структурную схем алгоритма встраивания и извлечения сообщения из стегоконтейнера.

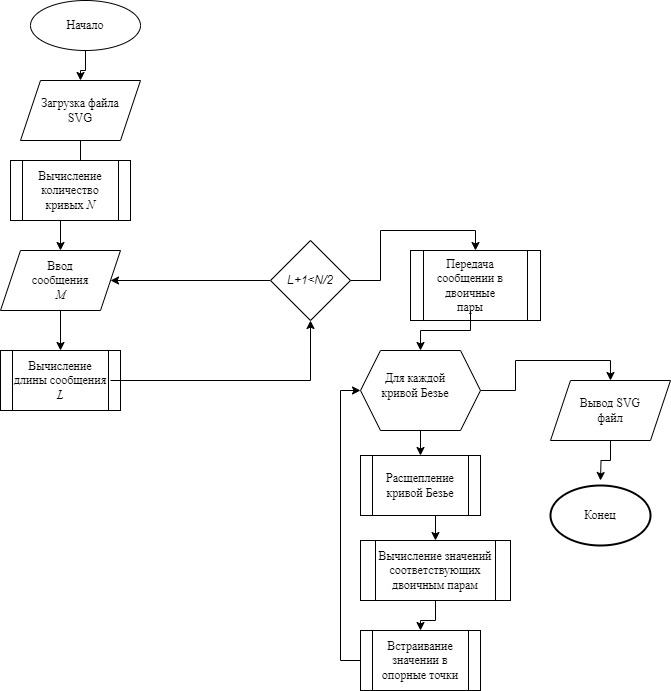


Рисунок 3.5 - Структурная схема алгоритма встраивания скрытого сообщения [165]

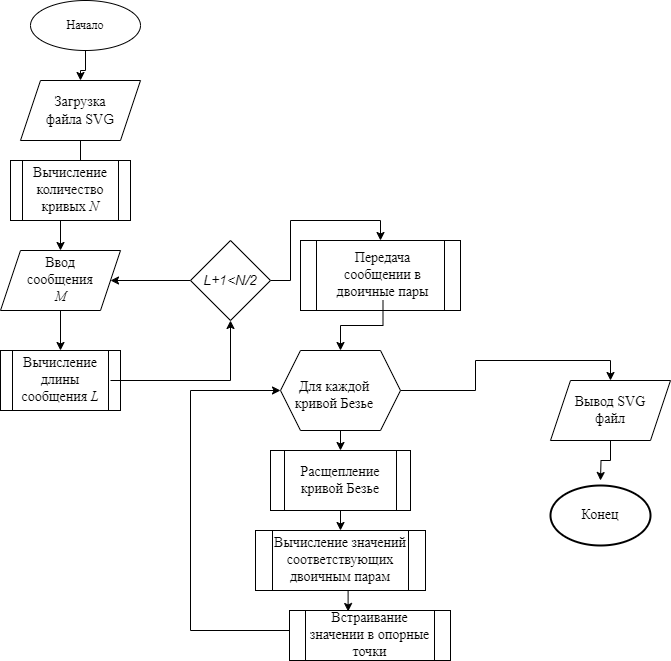


Рисунок 3.6 – Структурная схема алгоритма извлечения сообщения из контейнера [165]

В результате был рассмотрен новый стеганографический метод встраивания и извлечения скрытых сообщений при использовании SVG-файлы в качестве контейнеров для хранения. Метод основан на изменении параметров кубических кривых Безье. Для реализации метода авторами [165] была представлена библиотека StegoSVG, которая использовалась при разработке приложения. Экспериментальные результаты показали эффективность метода в сокрытии информации и низкую вероятность обнаружения. Предлагаемый метод рекомендуется использовать для защиты сайтов от подделок.

Обычно внедряемое сообщение представляет собой случайную последовательность. В ряде случаев допускается внедрять текст. Однако в таком случае требуется, чтобы стегоаналитик не смог извлечь само сообщение без знания секретного ключа. В противном случае наличие осмысленного текста в контейнере вызовет подозрение.

В разработке [166] авторы предлагают новый метод сокрытия текста на основе кривой Безье и описывают новый метод встраивания сообщения в структурированные формы. Алгоритм встраивания сообщений представляет контуры фигуры в виде набора кубических кривых Безье и отрезков прямых линий. Результаты обширного субъективного теста подтверждают, что изменения формы действительно незаметны. Далее, чтобы протестировать восстановление битов сообщения в зашумленных физических средах, текстовый документ проходит процедуру печати-ксерокопирования-сканирования. Обнаружено, что восстановление сообщений происходит стабильно даже после нескольких циклов копирования. Они показывают, что этот метод имеет высокую устойчивость к атакам и может быть эффективно использован для защиты конфиденциальной информации.

Задача восстановления изображения рассматривается в теории электросвязи как одна из главных задач. В статье [167] авторы предлагают метод восстановления изображения с использованием функции кривой Безье для интерполяции данных между каждой точкой выборки. Кривая Безье представляет собой набор независимых переменных, которые соответствуют координатам нескольких точек на изогнутой линии между двумя точками. Следовательно, данные между точками выборки могут быть спрогнозированы и восстановлены с помощью этого метода.

Одной из причин ошибок при восстановлении изображения является недостаточная выборка (в соответствии с теоремой Котельникова). Чтобы решить эту проблему, применяется метод с использованием кривой Безье для интерполяции данных между точками выборки для уменьшения погрешности. Предлагаемый метод сравнивается с предыдущими работами; компрессионной выборкой (CS) и дискретным преобразованием Фурье (DFT).

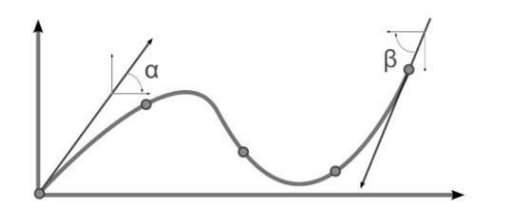


Рисунок 3.10 - Кубическая кривая Безье [168]

Авторы использовали кубическую кривую Безье, где кривая представлена 4 точками. На рисунке 3.10 показаны возможности охвата данных, состоящих из 5 чисел, данных, лежащих на кривой в пределах набора, который может быть представлен 4 числами кубической Безье. Это означает, что, отправив 4 данных из функции Безье, можно создать 5 данных в получателе.

Когда отправитель хочет передать изображение, то оно должно быть преобразовано в матрицу с размером, согласованным между участниками обмена. Пусть матрица A имеет размер B×B, где B делится на 5. Значение матрицы A - это глубина цвета каждого пикселя изображения от 0 до 255 для черно-белого изображения, как показано на рис.3.11.



Рисунок 3.11 - Значение изображения в оттенках серого [167]

Результат показывает, что реконструкция изображения с использованием кривой Безье имеет более близкое качество к исходному изображению по сравнению с компрессионной выборкой и DFT, при этом она может поддерживать небольшой размер восстановленных изображений.

Ниже представлена таблица, в которой указывается низкое сходство с оригинальным изображением. Искаженное изображение имеет меньший размер и может быть с легкостью передано по каналу связи. В последующем изображение будет восстановлено с некоторой погрешностью.

Таблица 3.1 – Сходства пикселей оригинального и сжатого изображений [167]

|  |  |  |  |
| --- | --- | --- | --- |
| Изображение | DFT | CS | Кривая Безье |
| 1 | 1,38% | 0,85% | 12,34% |
| 2 | 1,40% | 0,29% | 33,78% |
| 3 | 3,31% | 1,16% | 12,52% |

Для того чтобы убедиться в том, что предложенный метод обеспечивает лучшую реконструкцию изображения, вычисляется сходство значений пикселей. В таблице 3.1 показано точное сходство каждого пикселя в каждом методе реконструкции по сравнению с исходными изображениями. Метод реконструкции кривой Безье хорошие результаты, по сравнению DFT (ниже среднего), и CS ( очень низкое сходство пикселей - всего около 1%).

Таким образом, кривая Безье имеет более чем 10%-е пиксельное сходство каждого изображения. Поскольку изображение в оттенках серого имеет 256 значений, вычисляем сходство изображений. Вычисляется уровень разницы каждого пикселя по сравнению с исходным изображением. 4%, 10% и 20% означают, что значение разницы между восстановленными изображениями и исходными изображениями составляет менее 10, менее 25 и менее 50 соответственно.

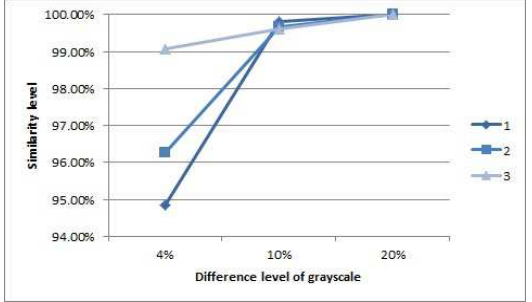


Рисунок 3.12 -Уровень сходства восстановленных изображений по кривой Безье [167]

По результатам экспериментальных исследовании показали, что использования этого метода восстановленное изображение имеет лучшее качество и более высокий уровень сходства по сравнению с другими методами реконструкции изображений.

На основе проведенного анализа можно сделать следующие выводы:

* кривая Безье первого порядка является простой линией, которая может быть использована для сокрытия серий секретных сообщений. Однако, она не обладает достаточным уровнем безопасности, так как легко поддаётся атакам, направленным на вычисление её координат;
* кривая Безье второго порядка описывается полиномом второй степени, которая может быть использована для скрытия более сложных серий секретных сообщений. Она обладает повышенным уровнем безопасности, чем кривая первого порядка, и её координаты вычислить сложнее;
* кривая Безье третьего порядка является кубической кривой и обеспечивает еще более высокий уровень безопасности, чем кривая второго порядка.

Применение кривых Безье разного порядка в стеганографии зависит от уровня безопасности, который требуется для конкретной задачи. Исходя из вышеперечисленных выводов, которые были получены путем проведения анализа, справедливо следующие. Кривая Безье позволяет эффективно скрывать информацию и имеет низкую вероятность обнаружения. Использование кривой Безье более высокого порядка обладают следующими преимуществами:

1. Большая емкость: кривые Безье более высокого порядка имеют большее количество точек, что позволяет сохранить больше информации в стегосообщении.
2. Более сложная структура: кривые Безье более высокого порядка имеют более сложную структуру, что делает их более сложными для обнаружения и анализа.
3. Более гладкое и естественное представление: кривые Безье более высокого порядка позволяют более гладкое и естественное представление изображения, что делает его более приятным для восприятия.
4. Более эффективное кодирование: кривые Безье более высокого порядка позволяют более эффективное кодирование информации, что уменьшает размер стегосообщения.
5. Устойчивость к атакам: кривые Безье более высокого порядка более устойчивы к атакам, таким как вырезание или удаление части стегосообщения. Также они могут быть использованы для защиты от атак, основанных на изменении формы изображения.

Поэтому представленном новом методе была использована кривая Безье с пятью точками, для эффективности внедрения секретного сообщения в изображения и для устойчивости к известным атакам различного типа.

# 3.3 Описание предлагаемого метода встраивания LIBC5

В настоящем разделе главы представлен новый метод встраивания скрытых сообщений в изображения – LIBC5 (Lagrange interpolation Bezier curve for 5 points). Подход заключается в следующем. Исходное изображение увеличивают в размере (масштабируют). При масштабировании применяется методы интерполяции, базирующиеся на кривой Безье (по 5 точкам). Дополнительное параметры кривой изменяются в зависимости от внедряемого сообщения.

Рассмотрим подробнее алгоритм LIBC5. Возьмем группу из пяти значений яркости пикселей интерполированного растрового изображения:. Значения , , являются значениями пикселей оригинального изображения по выбранной составляющей R, G либо B, а значения и являются добавленными путем интерполяции по соответствующей составляющей и взяты последовательно. Встраивание бита информации происходит в пикселях и путем взятия округленного до целого значения ближайшей точки с кривой Безье, такой, что младший бит ее значения совпадает с битом информации, которую мы хотим встроить. Чтобы подобрать нужное значение точки кривой Безье, необходимо задать такой шаг t, который обеспечит достаточный выбор значений точек кривой.

Рассмотрим график усредненных яркостей пикселей и построенной кривой Безье по их значениям c шагом t=0.1 (Рис.3.13):

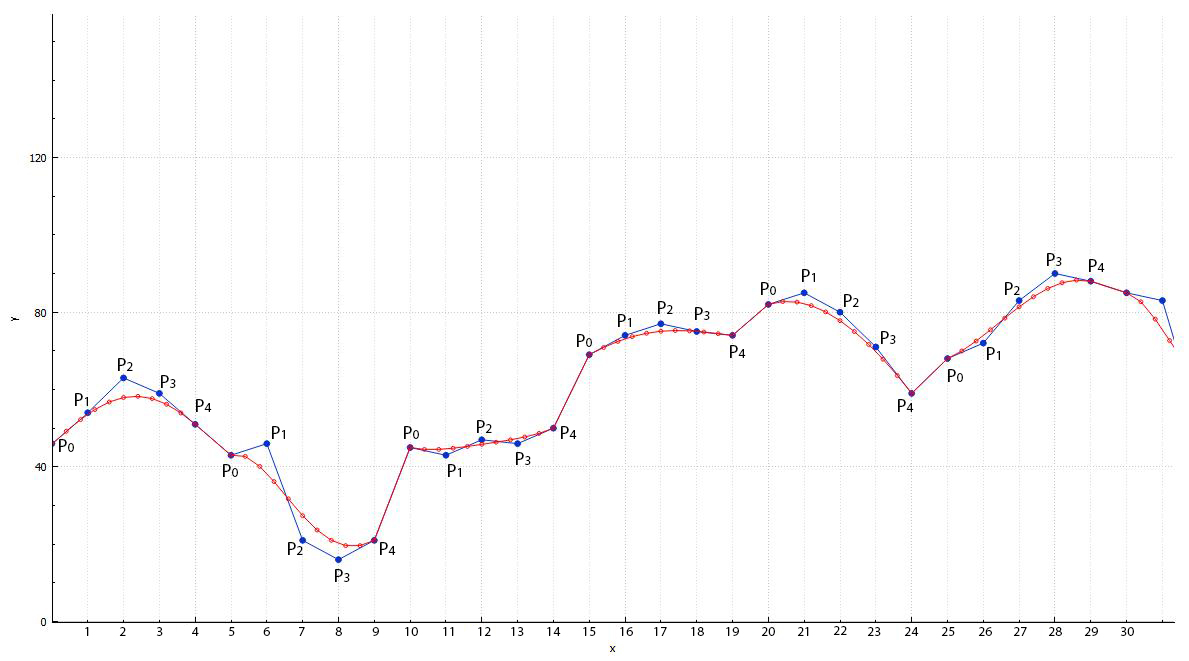


Рисунок 3.13 - Построение кривой Безье для файла 029.bmp с шагом t=0.1

По оси Y находятся значения яркостей изображения одной из составляющих пикселей, взятых из первой строки матрицы пикселей по порядку. Ось Х демонстрирует линейный порядок пикселей и показывает их порядковый номер. Для наглядности в данном примере был взят шаг t=0.1. По графику видно, что на каждые 5 точек яркостей интерполированной картинки (синяя линия), мы имеем 11 точек кривой Безье (красная линия).

Биты информации записываются в и в путем взятия такого значения с кривой Безье, у которого младший бит будет равен встраиваемому биту сообщения. На рисунке 3.14 показано как определяются соответствующие значения для замены пикселей и :

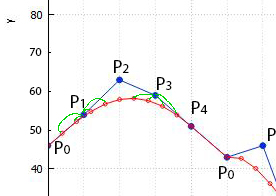


Рисунок 3. 14 - Точки кривой Безье для замены пикселей и для растрового файла с шагом t=0.1

По графику на рисунке 3.14 видно, что для точки имеется выбор из 4 значений точек кривой Безье, которые можно использовать, округлив их до целого значения и используя их младший бит. Необходимо взять с кривой Безье наиболее близкое значение с точкой такое, что младший бит этого значения будет совпадать с битом встраиваемого сообщения. Затем для точки также нужно подобрать подходящее значение с кривой Безье из четырех возможных, после чего строится отрезок кривой Безье по следующим пяти точкам пикселей и действия повторяются.

Обозначим округленные значения кривой Безье, которые можно использовать для замены текущей точки или множествами и соответственно, где k – это количество значений при данном шаге t. Значение k фактически представляет собой количество отрезков, на которые делится кривая, проходящая от точки до точки за исключением первого и последнего отрезков, а также за исключением тех двух отрезков, которые окружают значение кривой, соответствующей по порядку. Таким образом, чтобы вычислить k, нужно разделить единичный отрезок на длину интервала t, отнять 4 отрезка и разделить полученное значение пополам, для замены двух значений и . Затем нужно прибавить единицу, так как нам необходимо получить в итоге не количество отрезков, а количество полученных точек (Eq.3.5):

(3.5)

Упростив данное выражение, получаем формулу для вычисления k (Eq.10):

(3.6)

Таким образом, мы видим, что при t=0.1 создается выбор из четырех возможных значений. Краевые значения на кривой, соответствующие , и не используются для замены значений и , так как они соотносятся с теми пикселями изображения, которые изначально составляли матрицу пикселей до процесса интерполяции.

Так как значения точек кривой Безье должны быть округлены до целого, то мы можем получать одинаковые значения, которые сократят выбор для замены и . Поэтому далее рассмотрим случай при t=0.05. На рисунке 3.15 красной линией показана кривая Безье, а синей линией построен график яркостей текущего растрового изображения:

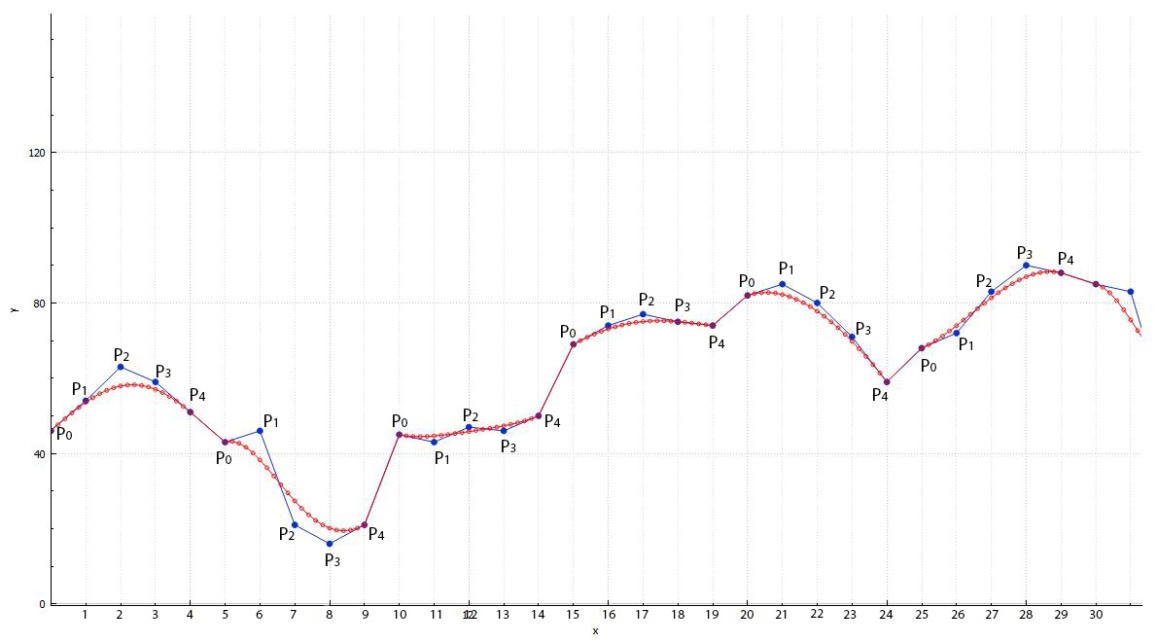


Рисунок 3.15 - Построение кривой Безье для файла 029.bmp с шагом t=0.05

При построении кривой Безье c шагом t=0.05 мы имеет выбор из k=9 значений для замены каждого из и . На рисунке 3.16 показаны возможные точки для замены и .

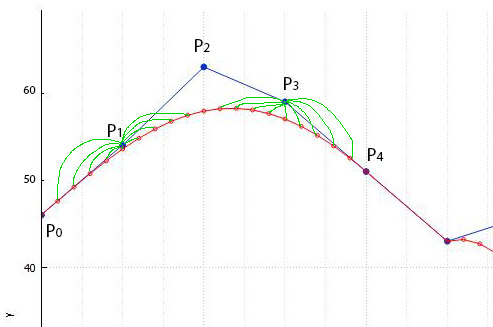


Рисунок 3.16 - Возможные точки для замены и на кривой Безье для файла 029.bmp с шагом t=0.05.

Рассмотрим пример встраивания сообщения с авторской информацией в растровый файл 029.bmp с помощью рассмотренного метода LIBC5. В таблице 3.2 показаны округленные значения R при соответствующих значениях t для файла 029.bmp, а также соответствие этих значений точкам , , , и :

Таблица 3.2. Пошаговые значения  для группы из пяти пикселей

|  |  |  |
| --- | --- | --- |
| (шаг) |  | P  (яркость пикселей) |
| 0 | 46 |  |
| 0.05 | 48 |  |
| 0.1 | 49 |  |
| 0.15 | 51 |  |
| 0.2 | 52 |  |
| 0.25 | 54 |  |
| 0.3 | 55 |  |
| 0.35 | 56 |  |
| 0.4 | 57 |  |
| 0.45 | 57 |  |
| 0.5 | 58 |  |
| 0.55 | 58 |  |
| 0.6 | 58 |  |
| 0.65 | 58 |  |
| 0.7 | 58 |  |
| 0.75 | 57 |  |
| 0.8 | 56 |  |
| 0.85 | 55 |  |
| 0.9 | 54 |  |
| 0.95 | 53 |  |
| 1 | 51 |  |

По таблице 3.2 видно, что для замены значения имеются следующие варианты: 48, 49, 51, 52, 54, 55, 56, 57. А для замены : 58, 57, 56, 55, 54, 53. Также следует учитывать, что из возможных значений для замены и мы должны выбрать одно подходящее значение, младший бит которого совпадет с битом сообщения для встраивания. При этом, если младший бит либо уже совпал с битом сообщения, то замены не потребуется.

Предположим, что последовательность битов сообщения: m={1,0}. Переведем в двоичный вид значения множества R1, соответствующего:

=4810=1100002

=4910=1100012

=5110=1100112

=5210=1101002

=5410=1101102

=5510=1101112

=5610=1110002

=5710=1110012

Для встраивания первого бита по порядку из последовательности m, нам необходимы такие значения из R1, у которых самый младший бит равен 1, то есть , , и . Заменив значения пикселя на любое из них, мы таким образом запишем бит 1 в матрицу пикселей растрового файла и перейдем к .

Для переведем в двоичный вид значения соответствующего множества R3, исключая повторяющиеся значения:

=5810=1110102

=5710=1110012

=5610=1110002

=5510=1101112

=5410=1101102

=5310=1101012

Для встраивания второго бита по порядку из последовательности m, нам необходимы такие значения из R3, у которых самый младший бит равен 0, то есть , , и . Заменив значения пикселя на любое из них, запишем бит 0 в матрицу пикселей растрового файла. Таким образом, блок из пяти пикселей пройден, в него записано 2 бита сообщения m.

Аналогично происходит встраивание в последующие блоки изображения, взятые линейно из матрицы пикселей. Изображение 029.bmp размером 450 по ширине и 450 по высоте позволяет использовать по 90 блоков в каждой строке для каждой из составляющих RGB. Таким образом, общее количество блоков для данного изображения составит 40500 при использовании одной из составляющих пикселей либо 121500 при использовании трех составляющих. При встраивании в каждый блок по 2 бита, мы получаем формулу (3.7) для вычисления максимального количества встроенных бит в растровое изображение по одной из составляющих размером W по ширине и H по высоте предложенным методом встраивания(Eq.3.7):

(3.7)

В рассмотренном примере N=81000 бит или 10125 байт. В случае использования всех составляющих яркость пикселей, значение N увеличится в три раза. Следует отметить, что при необходимости заполнения контейнера на половину его ёмкости, можно использовать блоки пикселей через один, так как кривая Безье по пяти точкам строится в отдельности для каждого блока из пяти пикселей.

Если мы вернемся к рассмотренному примеру, то закономерно возникает вопрос: По какому алгоритму выбирается значение ri для замены пикселя изображения, если мы имеем выборку из нескольких подходящих значений? Данный алгоритм был реализован таким образом, что из подходящей выборки значений ri выбирается первое по порядку и используется для замены пикселя. Таким образом, при замене используется значение точки кривой Безье, наиболее близкое к из набора подходящих значений.

Обозначим множество подходящих значений ri для встраивания каждого очередного бита mj как и , то есть содержит значения из , а содержит значения из такие, что их самый младший бит совпадает с текущим битом встраиваемого сообщения. Значения e и f переменны и зависят от пикселей используемого изображения. Чем меньше шаг t, тем меньше вероятность того, что k окажется равным e или f. При возникновении такой ситуации, что либо , ни одно из значений множества R1 либо R3 не подойдет для встраивания очередного бита сообщения. При декодировании невозможно будет определить наличие такого случая, что создаст ошибку декодирования в связи с потерей бита сообщения. В программной реализации мы взяли шаг t=0.01, который снизит вероятность данной проблемы и позволит получить достаточную выборку S1 и S3. Но каким бы маленьким ни был шаг t, количество значений множеств R1 и R3 всегда будет ограничено, так как они являются округленными значениями, высчитанными по формуле (1) и даже при максимально возможном количестве построенных точек кривой Безье мы будем получать несколько равных значений. Поэтому здесь стоит подумать над качеством исходных контейнеров. Если мы рассмотрим растровое изображение, в котором встречаются блоки только с одинаковой яркостью, то неизбежно столкнемся с потерей встраиваемого бита сообщения. Таким образом, для однозначного декодирования, необходимо предусмотреть случаи заведомо неподходящих блоков пикселей, которые будут пропущены. Алгоритм LIBC5 был запущен для набора из 800 растровых изображений grayscale размером 450x450 точек, произведено успешное декодирование встроенных сообщений, представляющих собой псевдослучайную последовательность. Кривая Безье строилась с шагом t=0.01.

Заменяя в матрице пикселей значения, полученные с помощью интерполяции Лагранжа, мы не нарушаем общую статистику пикселей изображения. В то же время, используя кривую Безье, мы пользуемся плавным переходом значений пикселей, минимизируя возможные искажения. Таким образом, можно заполнить встроенной информацией до 50% пикселей картинки. Но так как в исследованных ранее методах [115, 148] процент встраивания оставался на уровне 21%, то для сравнительного анализа мы встроили примерно такое же количество информации.

# 3.4 Стегоанализ нового метода LIBC5

В данном разделе главы описаны результаты проведения стегоанализа разработанного метода LIBC5. Для эксперимента был использован набор из 800 изображений [168] размером 450х450 пикселей, заполненных предложенным методом на 21%. Так как сообщение перед внедрением шифруют, то оно выглядит как случайная последовательность. Соответственно, в нашем эксперименте сообщение имитировалось при помощи генератора случайных чисел.

Анализ полученных стегоконтейнеров проводился методом RS (Regular–Singular) [100]. RS анализ использует чувствительный метод двойной статистики, полученной из пространственных корреляций в изображениях. В RS методе существуют три главных фактора, которые влияют на точность оцененной длины сообщения: начальное отклонение, уровень шума изображения-контейнера и размещение бит сообщения в изображении. Этот метод показывает достаточно точный результат даже на шумных изображениях. Проведенные эксперименты методом RS показали стегоустойчивость алгоритма LIBC5. В таблице 3.3. приведен результат RS анализа на наборе пустых контейнеров, а в таблице 3.4 - на заполненных контейнерах с помощью метода LIBC5. Полученные результаты подсчета ошибки первого рода показали, что ошибка составляет менее 0,125%. Процент обнаружения встроенной информации методом RS показал высокую устойчивость разработанного нами метода LIBC5 в отношении данного вида стагоанализа.

Таблица 3.3. RS анализ на наборе пустых контейнеров 450x450 (800 картинок)

|  |  |  |  |
| --- | --- | --- | --- |
| Результаты классификации | | | |
| Определенный объем внедрения | 0% | 1-4% | 5% и более |
| Доля файлов | 61,25% | 38% | 0,5% |

Таблица 3.4. RS анализ на наборе заполненных контейнеров (800 картинок)

|  |  |  |  |
| --- | --- | --- | --- |
| Результаты классификации | | | |
| Определенный объем внедрения | 0% | 1-4% | 5% и более |
| Доля файлов | 84,75 | 15,25 | 0 |

Проведем сравнение результатов разработанного метода со стеганографическим методом INMI который мы исследовали в статье [115]. В работе [147] ранее была получена модификация метода INMI, основанная на использовании интерполяционного полинома Лагранжа второй степени для получения изображения-контейнера. Интерполяционная формула Лагранжа имеет следующий вид (Eq.3.8):

 (3.8)

где  - многочлен степени , принимающий значение равное единице, в узле . равное нулю – в остальных узлах , .

В модифицированном алгоритме метода INMI изображение, полученное путём добавления дополнительных строк и столбцов пикселей к исходному изображению, рассматривались фрагментами по 5 пикселей, пронумерованных от 0 до 4. Известные пиксели (0, 2, 4) считаются узлами интерполяции. Поэтому с помощью интерполяционного полинома Лагранжа второй степени были найдены неизвестные значения пикселей (1 и 3). Изображение, полученное путем добавления дополнительных строк и столбцов пикселей к исходному изображению, рассматривали фрагментами по 5 пикселей, пронумерованных от 0 до 4.

Значения пикселей изображения-контейнера получают по следующей формуле:

 (3.9)

где  – номер пикселя во фрагменте из пяти пикселей.

На основании рассмотренного алгоритма INMI, в работе [115, 146] мы определили, что максимальная емкость контейнера составляет 21% и зависит от изображения. Мы проводили исследования на наборе из 800 изображений размера 450 на 450, то есть используя тот же набор стегоконтейнеров, что и в данном исследовании метода LIBC5. Анализирующая программа «RS – анализа» выдает количество встроенной информации () в процентах от эмпирической емкости контейнера. По значению  можно судить о том, заполнен был контейнер или пуст: при RS классифицирует контейнер как заполненный. Приведем результирующие таблицы стегоанализа метода INMI:

Таблица 3.5. RS анализ на наборе пустых контейнеров размером 450х450 (800 картинок)

|  |  |  |  |
| --- | --- | --- | --- |
| Результаты классификации | | | |
| Определенный объем внедрения | 0% | 1-4% | 5% и более |
| Доля файлов | 61% | 39% | - |

Таблица 3.6. RS анализ на наборе контейнеров размером 450х450, заполненных с помощью метода интерполяции на 12%.

|  |  |  |  |
| --- | --- | --- | --- |
| Результаты классификации | | | |
| Определенный объем внедрения | 0% | 1-4% | 5% и более |
| Доля файлов | 51,5% | 48% | 0,5% |

Сравнив полученные результаты по разработанному нами методу LIBC5 в таблицах 2 и 3, и результаты метода INMI в таблицах 3.5 и 3.6, можно сделать вывод, что LIBC5 более устойчив к стегоанализу и может успешно применяться на растровых изображениях в задачах встраивания ЦВЗ.

# 3.5 Выводы по разделу

Рассмотрены существующие методы, базирующиеся на кривой Безье, такие как: а) стеганографический метод встраивания и извлечения скрытых сообщений при использовании SVG-файлы в качестве контейнеров для хранения; б) метод, который основан на замене частей изображения на соответствующие значения на кривой Безье, построенной с использованием криволинейного преобразования; в) метод сокрытия текста на основе кривой Безье; г) метод с использованием квадратичного уравнения кривой Безье, основанного на поведении PGP. Были проанализированы преимущества и недостатки кривых Безье по их порядкам.

Предложен новый метод LIBC5, который был использован в сокрытии данных для создания стегоконтейнера, который содержал секретные данные. Были приведены итоговые сравнительные результаты RS анализа методов INMI и LIBC5.

В ходе экспериментального исследования было показано, что имеется улучшение показателей стегоустойчивости метода LIBC5 по сравнению с INMI. Оба метода используют алгоритм интерполяции Лагранжа, и подходят для применения в задачах встраивания ЦВЗ в растровые изображения. При использовании этого метода восстановленное изображение имеет лучшее качество и более высокий уровень сходства по сравнению с другими методами реконструкции изображений.

Метод LIBC5 не только является устойчивым к методу RS, но и сопоставим по стойкости и ёмкости со стегосистемой способа перестановок для растровых изображений, рассмотренной в [147], что является надежной схемой встраивания сообщений об авторских правах в изображения.

# 4 ОПИСАНИЕ ПРОГРАММНЫХ СРЕДСТВ

Для проведения экспериментальных исследований в рамках диссертации было разработано программное средство. Назначение программы: встраивать секретное сообщение в контейнер используя предложенный подход. Данная программа написана на языке программирования Python. Python является популярным и простой в понимании, а также охватывает многие области программирования. Было реализовано 2 основных функций, 9 вспомогательных функций и 1 класс, собственно, для обработки изображения. Ниже описывается каждая часть кода.

Начинаем описание с вспомогательных функций. Они используются в некоторых частях кода для упрощения работы внутри основных функциях.

*Вспомогательные функций*. Функция рекурсивного высчитывания факториала от числа:

**def** f**(**x**):**

**if** x **<=** 1**:** **return** 1

**return** f**(**x **-** 1**)** **\*** x

Функция для генерации строки с заданной длиной:

**def** generateKey**(**x**):**

**return** ''**.**join**(**random**.**choice**(**string**.**ascii\_uppercase**)** **for** \_ **in** **range(**x**))**

Функция преобразования двоичной строки в строку байт и из него в символ из ASCII-таблицы:

**def** bitstring\_to\_bytes**(**s**):**

**return** **bytes(int(**s**[**i **:** i **+** 8**],** 2**)** **for** i **in** **range(**0**,** **len(**s**),** 8**))**

Функция преобразования двоичное значения байта в десятичное значение:

**def** bytes\_to\_decimical**(**s**):**

deg **=** **[int(**math**.pow(**2**,** i**))** **for** i **in** **range(**0**,** 8**)][::-**1**]**

num **=** 0

**for** i **in** **range(len(**s**)):**

num **+=** deg**[**i**]** **\*** **int(**s**[**i**])**

**return** num

Функция преобразования десятичного значения в двоичное значения байта:

**def** decimical\_to\_bytes**(**s**):**

b **=** ""

**while** s**:**

b **+=** **str(**s **%** 2**)**

s **//=** 2

ans **=** ""

**for** i **in** **range(**8**-len(**b**)):**

ans **+=** '0'

ans **+=** b**[::-**1**]**

**return** ans

Функция преобразования символов строки в двоичное значение байт:

**def** toBinChar**(**x**):**

tmp **=** **[**decimical\_to\_bytes**(ord(**x**[**i**]))** **for** i **in** **range(**0**,** **len(**x**))]**

ans **=** ""

**for** i **in** tmp**:**

ans **+=** i

**return** ans

Функция вычисления Безье для 5-ти точек c шагом 0.01:

**def** bezje**(**x**):**

n **=** 4

t **=** np**.**linspace**(**0**,** 1**,** 100**)**

Pt **=** **[]**

**for** step **in** t**:**

i **=** 0

cnt **=** 0

**while** i **<** n **+** 1**:**

cnt **+=** x**[**i**]** **\*** **(**f**(**n**)** **/** **(**f**(**i**)** **\*** f**(**n **-** i**)))** **\*** **((**1 **-** step**)** **\*\*** **(**n **-** i**))** **\*** **(**step **\*\*** i**)**

i **+=** 1

Pt**.**append**(**math**.**ceil**(**cnt**))**

**return** Pt

Функция шифрования исходного сообщения по методу Вернам:

**def** encodeVernam**(**s**,** k**):**

alpha **=** **[chr(**64**+**i**)** **for** i **in** **range(**1**,** 27**)]**

x**,** y **=** **[],** **[]**

**for** i **in** **range(len(**s**)):**

**for** j **in** **range(**0**,** 26**):**

**if** alpha**[**j**]** **==** s**[**i**]:**

x**.**append**(**j**)**

**break**

**for** j **in** **range(**0**,** 26**):**

**if** alpha**[**j**]** **==** k**[**i**]:**

y**.**append**(**j**)**

**break**

**for** i **in** **range(len(**x**)):**

x**[**i**]** **=** **(**x**[**i**]+**y**[**i**])%**26

x **=** **[**alpha**[**x**[**i**]]** **for** i **in** **range(len(**x**))]**

y **=** ""

**for** i **in** x**:**

y **+=** i

**return** y

Функция декодирования зашифрованного сообщения по методу Вернам:

**def** decodeVernam**(**s**,** k**):**

alpha **=** **[chr(**64**+**i**)** **for** i **in** **range(**1**,** 27**)]**

x**,** y **=** **[],** **[]**

**for** i **in** **range(len(**s**)):**

**for** j **in** **range(**0**,** 26**):**

**if** alpha**[**j**]** **==** s**[**i**]:**

x**.**append**(**j**)**

**break**

**for** j **in** **range(**0**,** 26**):**

**if** alpha**[**j**]** **==** k**[**i**]:**

y**.**append**(**j**)**

**break**

**for** i **in** **range(len(**x**)):**

x**[**i**]** **=** **(**x**[**i**]-**y**[**i**])%**26

x **=** **[**alpha**[**x**[**i**]]** **for** i **in** **range(len(**x**))]**

y **=** ""

**for** i **in** x**:**

y **+=** i

**return** y

Дальше идут основные функций, к которым будут использованы вспомогательные.

*Основные функций.* Функция шифрование заданного сообщения в заданную картинку методу «кривой Безье». Кодирование в методе происходит линейно, т.е. блоки из пяти пикселей берутся последовательно. Сложностью в методе при реализации заключалась в блоках с одинаковыми значениями пикселей, т. к. кривая Безье давала значения похожие, что разрушает цикл и работу программы. Собственно, такие блоки из пикселей пропускались. По количеству пикселей которую мы можем затрагивать можно вывести формулу в зависимости от размера картинок, причем, после интерполяций. В качестве примера берем картинку с размером 225x225. После интерполяций размер будет иметь в два раза больше 450x450. В каждом пикселе есть значения RGB, т.е. три значения что дает 450 \* 3 = 1350 пикселей и этих пикселей разбиваем из пяти блоков, а также не забудем что будем затрагивать только два пикселя в качестве точек P1 и P3 ((1350 / 5) \* 2) = 540 пикселей. Далее, понимаем, что это только первая строка и таких есть еще 450 – 1 = 449 что дается 540 \* 449 = 242.460 пикселей. В итоге 242.460 (~2,4 \* 105) пикселей в количестве можем использовать в хорошем случае без каких-либо ограничений. Код описывается комментариями.

# - является комментарием!

**def** encode**(**img**,** msgBin**):**

# переводим сообщения в двоичной формат

msgBin **=** toBinChar**(**msgBin**)**

# разбивание пикселей из пяти блоков

fiveBlock **=** **[]**

x **=** 0

**while** x **<** **len(**img**):**

px **=** **[]**

**for** i **in** **range(len(**img**[**x**])):**

**for** j **in** **range(**3**):**

px**.**append**(**img**[**x**][**i**][**j**])**

l **=** 0

r **=** 5

c **=** 0

n **=** **len(**px**)**

tmp **=** **[]**

**while** r **<=** n**:**

cpy **=** **[]**

**for** i **in** **range(**l**,** r**):**

cpy**.**append**(**px**[**i**])**

tmp**.**append**(**cpy**)**

l **=** r

r **+=** 5

fiveBlock**.**append**(**tmp**)**

x **+=** 1

bit **=** 0 # указатель на текущий бит строки

x **=** 0 # граница строки массива пикселей из пяти блоков

y **=** 0 # граница столбца массива пикселей из пяти блоков

**while** **True:**

# print(x, ' ', y, ' ', bit, end='\n')

**if** x **==** **len(**fiveBlock**):** # если дошли до строки границы строки массива, выводим сообщение достигли края массива

**print(**"out of range!"**)**

**break**

**if** y **==** **len(**fiveBlock**[**x**]):** # если дошли до границы столбца, начинаем с новой строки

y **=** 0

x **+=** 1

**if** bit **==** **len(**msgBin**):** # если указатель дошел до края бита строки, биты сообщения встроено!

fiveBlock**[**x**][**y**][**0**]** **=** fiveBlock**[**x**][**y**][**2**]** **=** fiveBlock**[**x**][**y**][**4**]** **=** 0 # значения соседей точек P1, P3 зануляем как останочные значения для дешифровывание

**print(**"all message encoded!"**)**

**break**

cnt **=** 0

**for** i **in** **range(**1**,** **len(**fiveBlock**[**x**][**y**])):**

**if** fiveBlock**[**x**][**y**][**i**]** **==** fiveBlock**[**x**][**y**][**0**]:**

cnt **+=** 1

**if** cnt **==** **len(**fiveBlock**[**x**][**y**])** **-** 1**:** # если все точки одинаковые, то пропускаем этот блок

y **+=** 1

**continue**

b **=** bezje**(**fiveBlock**[**x**][**y**])** # вычисляем Безье из пяти точек (пикселей)

**if** decimical\_to\_bytes**(**fiveBlock**[**x**][**y**][**1**])[-**1**]** **==** msgBin**[**bit**]:** # если младший бит точки P1 блока совпал с текущим битов строки

bit **+=** 1 # то пропускаем точку P1

**if** decimical\_to\_bytes**(**fiveBlock**[**x**][**y**][**3**])[-**1**]** **==** msgBin**[**bit**]:** # если младший бит точки P3 блока совпал с текущим битов строки

bit **+=** 1 # то пропускаем точку P3

y **+=** 1 # пропускаем блок

**else:** # иначе

**for** e **in** b**:** # из Безье находим такое значение, что младший бит точки P3 совпал с текущим битом строки

**if** decimical\_to\_bytes**(**e**)[-**1**]** **==** msgBin**[**bit**]:**

fiveBlock**[**x**][**y**][**3**]** **=** e

bit **+=** 1

y **+=** 1

**break**

**else:**

**for** e **in** b**:** # из Безье находим такое значение, что младший бит точки P1 совпал с текущим битом строки

**if** decimical\_to\_bytes**(**e**)[-**1**]** **==** msgBin**[**bit**]:**

fiveBlock**[**x**][**y**][**1**]** **=** e

bit **+=** 1

**break**

**if** decimical\_to\_bytes**(**fiveBlock**[**x**][**y**][**3**])[-**1**]** **==** msgBin**[**bit**]:** # если младший бит точки P3 блока совпал с текущим битов строки

bit **+=** 1 # то пропускаем точку P3

y **+=** 1 # пропускаем блок

**else:**

**for** e **in** b**:** # из Безье находим такое значение, что младший бит точки P3 совпал с текущим битом строки

**if** decimical\_to\_bytes**(**e**)[-**1**]** **==** msgBin**[**bit**]:**

fiveBlock**[**x**][**y**][**3**]** **=** e

bit **+=** 1

y **+=** 1

**break**

# массив пикселей из пяти блоков обратно переводим в формат массива картинки

endImg **=** **[]**

z **=** 0

**while** z **<** **len(**fiveBlock**):**

px **=** **[]**

**for** i **in** **range(len(**fiveBlock**[**z**])):**

**for** j **in** **range(**5**):**

px**.**append**(**fiveBlock**[**z**][**i**][**j**])**

l **=** 0

r **=** 3

c **=** 0

n **=** **len(**px**)**

#this is Pointers

tmp **=** **[]**

**while** r **<=** n**:**

cpy **=** **[]**

**for** i **in** **range(**l**,** r**):**

cpy**.**append**(**px**[**i**])**

tmp**.**append**(**cpy**)**

l **=** r

r **+=** 3

endImg**.**append**(**tmp**)**

z **+=** 1

# массив из пикселей обернем в массив

endImg **=** np**.**array**(**endImg**)**

# возращаем измененный массив пикселей картинки

**return** endImg

*Функция декодировния сообщения из заданной картинки*. Декодирование делается обратным ходом кодированию. Собираем младшие биты, чтобы перевести их в байты и из байтов в символы, собственно. Остановочными пикселями являются соседние пиксели точек P1 и P3, что они должны быть равными нулю. Такое ограничение ставится при кодировании. В коде подробно описано комментариями.

# - является комментарием!

**def** decode**(**img**):**

# пустая строка для сбора битов

msgBin **=** ""

# разбиение на пять блоков пикселей

fiveBlock **=** **[]**

x **=** 0

**while** x **<** **len(**img**):**

px **=** **[]**

**for** i **in** **range(len(**img**[**x**])):**

**for** j **in** **range(**3**):**

px**.**append**(**img**[**x**][**i**][**j**])**

l **=** 0

r **=** 5

c **=** 0

n **=** **len(**px**)**

tmp **=** **[]**

**while** r **<=** n**:**

cpy **=** **[]**

**for** i **in** **range(**l**,** r**):**

cpy**.**append**(**px**[**i**])**

tmp**.**append**(**cpy**)**

l **=** r

r **+=** 5

fiveBlock**.**append**(**tmp**)**

x **+=** 1

# собственно границы строки и столбцы массива из пяти блоков

xx **=** 0 # граница строки массива из пяти блоков

yy **=** 0 # граница столбца массива из пяти блоков

**while** **True:** # работаем пока не достигнем резльутатов

**if** xx **==** **len(**fiveBlock**):** # если граница строки достиг предела, останавливаем цикл

**break**

**if** yy **==** **len(**fiveBlock**[**xx**]):** # если граница столбца достиг предела, переходим в новую строку

yy **=** 0

xx **+=** 1

**if** fiveBlock**[**xx**][**yy**][**0**]** **==** 0 **and** fiveBlock**[**xx**][**yy**][**2**]** **==** 0 **and** fiveBlock**[**xx**][**yy**][**4**]** **==** 0**:** # если соседние значения точки P1, P3 нули, то значить мы собрали все биты

**print(**"all message decoded!"**)**

**break**

cnt **=** 0

**for** i **in** **range(**1**,** **len(**fiveBlock**[**xx**][**yy**])):**

**if** fiveBlock**[**xx**][**yy**][**i**]** **==** fiveBlock**[**xx**][**yy**][**0**]:**

cnt **+=** 1

**if** cnt **==** **len(**fiveBlock**[**xx**][**yy**])** **-** 1**:** # если значения пикселей одинаковы то, пропускаем текущий блок

**continue**

msgBin **+=** decimical\_to\_bytes**(**fiveBlock**[**xx**][**yy**][**1**])[-**1**]** # кладем младший бит точки P1 в строку ответа

msgBin **+=** decimical\_to\_bytes**(**fiveBlock**[**xx**][**yy**][**3**])[-**1**]** # кладем младший бит точки P3 в строку ответа

yy **+=** 1

# строку из двоичного формата переводим в байты символы

**return** bitstring\_to\_bytes**(**msgBin**)**

Разобрались с функциями. Теперь, разберемся с алгоритмом работы с программой. Алгоритм очень простой. На вход подается картинка и сообщение, которую мы хотим встраивать в картинку. В ходе эксперимента мы случайно сгенерировали сообщение с заданной длиной (условно, равной 5000).

F1 **=** **open(**"message.txt"**,** "w"**)**

msg **=** generateKey**(**5000**)**

F1**.**write**(**msg**)**

F2 **=** **open(**"key.txt"**,** "w"**)**

key **=** generateKey**(len(**msg**))**

F2**.**write**(**key**)**

encodedMsg **=** encodeVernam**(**msg**,** key**)**

F3 **=** **open(**"vernamEncode.txt"**,** "w"**)**

F3**.**write**(**encodedMsg**)**

Cоздается отдельные 3 файла для сообщения, ключа и шифрования по методу Вернам:

F1 **=** **open(**"message.txt"**,** "r+"**)**

msg **=** F1**.**readline**()**

F2 **=** **open(**"key.txt"**,** "r+"**)**

key **=** F2**.**readline**()**

F3 **=** **open(**"vernamEncode.txt"**,** "r+"**)**

encodedMsg **=** F3**.**readline**()**

переменные считывает готовые данные из файлов.

Мы изначально договаривались, что встраивание сообщение в картинку будет происходить после интерполяции, поэтому следующее по очереди будет интерполяция картинки. При интерполяции картинка будет иметь размер в 2 раза больше и черно-белый формат.

image **=** cv2**.**imread**(**"C:/Users/User01/Desktop/JOB/Stegonagraphy/225x225(input)/test17.png"**)**

gray **=** cv2**.**cvtColor**(**image**,** cv2**.**COLOR\_BGR2GRAY**)**

grayImage **=** np**.**array**(**gray**)**

inter **=** np**.**array**(**interpolation**(**grayImage**))**

cv2**.**imwrite**(**"inter-img/inter764.BMP"**,** inter**)**

**print(**cv2**.**imread**(**"res-img/res1.png"**))**

Пример интерполяции одной картинки.

collection **=** "test-img/"

**for** i**,** filename **in** **enumerate(**os**.**listdir**(**collection**)):**

**print(**"image[{}]"**.format(**i **+** 1**),** end**=**'\n'**)**

img **=** cv2**.**imread**(**collection **+** filename**)**

gray **=** cv2**.**cvtColor**(**img**,** cv2**.**COLOR\_BGR2GRAY**)**

grayImg **=** np**.**array**(**gray**)**

inter **=** np**.**array**(**interpolation**(**grayImg**))**

cv2**.**imwrite**(**"inter-img/inter" **+** **str(**i **+** 1**)** **+** ".BMP"**,** inter**)**

Пример интерполяции несколько картинок. Как только интерполировали картинку можем встраивать в него зашифрованное исходное сообщение по методу Вернам. Упоминается, что встраивание происходит по методу кривой Безье.

img **=** cv2**.**imread**(**"inter-img/inter238.BMP"**)**

ans **=** encode**(**img**,** encodedMsg**)**

cv2**.**imwrite**(**"res-img/res238.BMP"**,** ans**)**

Пример для одной картинки

collection **=** "inter-img/"

**for** i**,** filename **in** **enumerate(**os**.**listdir**(**collection**)):**

**if** **(**i **+** 1**)** **>** 238**:**

**print(**"image[{}]"**.format(**i **+** 1**),** end**=**'\n'**)**

img **=** cv2**.**imread**(**"inter-img/" **+** filename**)**

ans **=** encode**(**img**,** encodedMsg**)**

cv2**.**imwrite**(**"res-img/res" **+** **str(**i **+** 1**)** **+** ".BMP"**,** ans**)**

Пример для несколько картинок. Таким образом, сообщение скрывается в картинке. Дешифрование обратный процесс шифрования.

img **=** cv2**.**imread**(**"res-img/res1.BMP"**)**

decodedMsg **=** decode**(**img**)**

**print(**msg **==** decodeVernam**(**decodedMsg**.**decode**(**"utf-8"**),** key**))**

Пример декодирования исходного сообщения, зашифрованного по методу Вернам. Заметим, что используется дешифрования по методу Вернам полученного сообщения

collection **=** "res-img/"

**for** i**,** filename **in** **enumerate(**os**.**listdir**(**collection**)):**

**if** **(**i **+** 1**)** **>** 238**:**

**print(**"image[{}]"**.format(**i **+** 1**),** end**=**'\n'**)**

img **=** cv2**.**imread**(**"res-img/" **+** filename**)**

ans **=** decode**(**img**)**

**print(**msg **==** decodeVernam**(**decodedMsg**.**decode**(**"utf-8"**),** key**))**

*Пример декодирования для нескольких картинок*. Обобщая шаги использования программы, возникает вопрос «как запустить программу?». Программа запускается целиком в терминале простой и не сложной командой: python **./**test**.**py, где «./test.py» - название файла, а «python» - собственно, является компилятором языка программирования python. Желательно, устанавливать последнюю версию компилятора.

*Библиотеки.* Использование встроенных функций языка программирования Python для обработки изображения не хватает. Поэтому в работе использованы следующие библиотеки для работы с изображениями (табл.4.1):

Таблица 4.1 – Использованные библиотеки

|  |  |  |
| --- | --- | --- |
| № | Название библиотеки | Функции |
| 1 | **from** scipy **import** ndimage | Scipy использовался для интерполяции |
| 2 | **import** numpy **as** np  **import** math | Numpy, math использовались для работы с многомерными массивами и арифметических вычислений |
| 3 | **import** cv2 | Cv2 использовался для манипуляций пикселями изображения |
| 4 | **import** random**,** string | Random, string использовались для генерации случайных символов |
| 5 | **import** os | Os использовался для получения пути к папкам, где находились изображения |

# **ЗАКЛЮЧЕНИЕ**

Проведенные исследования посвящены разработке эффективных методов внедрения и обнаружения скрытых данных в изображениях.

В рамках работы был создан научно-технический задел в области информационно-коммуникационных технологий и получены новые знания, позволяющие создавать тайные каналы связи с применением стеганографических методов интерполяции.

Цель диссертационной работы: создание новых и эффективных методов внедрения и обнаружения скрытых данных в изображении была достигнута. В диссертации были успешно решены следующие задачи:

Изучены существующие методы стеганографии и проведен анализ их преимуществ и недостатков.

Разработаны новые стеганографические алгоритмы, ориентированные на внедрение секретной информации в изображения, с учетом современных требований к безопасности и эффективности. Для решения этой задачи необходимо провести теоретический анализ и выбрать оптимальный подход к разработке новых стеганографических алгоритмов, затем реализовать эти алгоритмы и провести их тестирование.

Проведено тестирование разработанных алгоритмов на различных наборах данных и оценена их устойчивость к атакам, таким как изменение размера изображения, сжатие, фильтрация и другие. Проведен анализ скрытности внедрения информации предложенных алгоритмов.

Оценена эффективность разработанных алгоритмов по сравнению с существующими методами стеганографии по таким критериям, как скорость внедрения, скрытность информации, устойчивость к атакам.

Разработана программная реализация разработанных алгоритмов с интуитивно понятным пользовательским интерфейсом.

По результатам исследований были сформулированы следующие выводы:

* Метод на основе кривой Безье является устойчивым к анализу RS и сопоставим по стойкости и ёмкости со стегосистемой способа перестановок для растровых изображений;
* Метод INMI и базирующийся на кривой Безье являются уязвимыми к RS и SPAM анализу.
* Метода внедрения LIBC5 является устойчивым к RS анализу.
* Для рассматриваемых методов INMI и базирующийся на кривой Безье увеличение объема встроенной информации повышает вероятность успешного стегоанализа.
* Методы NMI и INMI уязвимы к RS анализу.

Перспектива для дальнейшей разработки темы диссертации состоит в следующем:

* Необходимо рассмотреть возможность использовать иные (отличные от Безье) полиномы для аппроксимации кривых.
* Необходимо рассмотреть возможность предварительного кодирования данных, добавляющего избыточность, которые вероятно могут увеличить объем полезного внедрения и сохранить прежний уровень скрытности внедрения.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Рябко Б.Я., Фионов А.Н., Шокин Ю.И. Криптография и стеганография в информационных технологиях. –Новосибирск: Наука, 2015. –240 с.
2. Дайырбаева Э.Н., Мурзин Ф.А., Липская М.А. Скрытие информации в изображениях //Вестник КазАТК им. М. Тынышпаева. – Алматы, 2020. –№1 (112). –С.283-290.
3. Кори Б., Буррес С. Методы и системы сокрытия информации: основы теории и применения. – М.: Техносфера, 2013. – 400 с.
4. Дайырбаева Э.Н., Липская М.А., Тойгожинова А.Ж. Суреттерді өңдеуде стрип-әдісті пайданалу жолдары мен нәтижелері//Вестник КазНИТУ. –Алматы, 2020. –№5. –Б. 279-284.
5. Дайырбаева Э.Н., Липская М.А., Тойгожинова А.Ж, Нугуманов Ш.Е. Сандық және компьютерлік стеганографиялардың сипаттамалары мен мүмкіншіліктеріне шолу//Вестник КазАТК. – Алматы, 2020. –№3 (114). -Б. 246-252.
6. Фридрихс Дж., Свенсон А. Методы стеганографии // Наука и жизнь. – 2001. – №6. – С. 42-47.
7. Рябко Б.Я., Фионов А.Н. Основы современной криптографии и стеганографии. - М.: Горячая линия – Телеком, (грант РФФИ 10-07-07010-д). - 2010. – 232 с.
8. Иванников А. Д. и др. Цифровая стеганография: шифрование, зашита //Информационные технологии. – 2004. – №8. – С. 1-32.
9. Нечта И. В. Новый метод стеганографии в социальных сетях //Современные проблемы телекоммуникаций. – 2017. – С. 86-90.
10. Абазина Е. С., Ерунов А. А. Цифровая стеганография: состояние и перспективы //Системы управления, связи и безопасности. – 2016. – №2. – С. 182-201.
11. Gustavus J. Simmons. The Prisoners' Problem and the Subliminal Channel. Problems of Control and Information Theory - 1984. Vol. 13, №4. - P. 297-304.
12. Колчин А.А. Методы и алгоритмы стеганографии// Информационно-измерительные и управляющие системы. – 2005. – №4. – С. 20-25.
13. Китани М. Методы и системы сокрытия информации. – М.: Техносфера, 2003. – 336 с.
14. Нечта И. В. Метод внедрения скрытых сообщений в исполняемые файлы //Вестник СибГУТИ. – 2011. – №2. – С. 3-10.
15. Галан Ф., Кабатянский Г. А. Покрытия, центрированные коды и комбинаторная стеганография //Проблемы передачи информации. – 2009. – Т. 45. – №3. – С. 106-111.
16. Рябко Б. Я., Фионов А. Н. Идеальные стенографические системы //Доклады Томского государственного университета систем управления и радиоэлектроники. – 2008. – №2-1 (18). – С. 61-62.
17. Рябко Б. Я., Фионов А. Н. Алгоритмы кодирования для идеальных стеганографических систем //Вестник Новосибирского государственного университета. Серия: Информационные технологии. – 2008. – Т. 6. – № 2. – С. 88-93.
18. Дэвид Н., Форсайт Дж. Стеганография и цифровые водяные знаки. – М.: Издательский дом «Вильямс», 2002. – 344 с.
19. Кори Б., Буррес С. Методы и системы сокрытия информации: основы теории и применения. – М.: Техносфера, 2013. – 400 с.
20. Лэй Х., Уайт М. Information Hiding - A Survey // Proceedings of the IEEE. – 1998. – Vol. 86. – №6. – P. 1062-1078.
21. Нечта И. В. Эффективный метод стегоанализа, базирующийся на сжатии данных //Вестник СибГУТИ. – 2010. – № 1. – С. 50-55.
22. Евсютин О.О. Модификация стеганографического метода LSB, основанная на использовании //Информатика и системы управления. – 2014. – №1. – С. 15-22.
23. Ryabko B., Ryabko D. Constructing perfect steganographic systems// Information and Computation. –2011. –Vol.209. –P. 1223 – 1230
24. Китани М. Техника сокрытия данных на основе LSB метода // Информационные технологии. – 2004. – №10. – С. 56-61.
25. Орлов В. В. Методы скрытой передачи информации в телекоммуникационных сетях //Самара, 2012. –166 с.
26. Hariri M., Karimi R., Nosrati M. An introduction to steganography methods //World Applied Programming. – 2011. – Т. 1. – №. 3. – С. 191-195.
27. Stepanov P. I. Development and testing of the decision-making unit for the control system of the technical condition of electromechanical equipment //Южно-Сибирский научный вестник. – 2018. – С. 113 -122.
28. Пастушков А. В., Калайда В. Т. Методы и алгоритмы поиска объекта на видеопотоке //Сборник научных трудов SWorld. – 2013. – Т. 6. – №3. – С. 38-42.
29. Чучкалова А. С. и др. Алгоритм стеганографии на основе дискретного косинусного преобразования. – 2023.
30. Грибунин В. Г. Вейвлеты в стеганографии //Статья. СПб. – 2009. –P. 1-10
31. Кэмпбелл М. Стеганография: техника сокрытия информации // Журнал «Хакер». – 2003. – №56. – С. 30-37.
32. Фридрихс Дж., Свенсон А. Методы стеганографии // Наука и жизнь. – 2001. – №6. – С. 42-47.
33. Колчин А.А. Методы и алгоритмы стеганографии // Информационно-измерительные и управляющие системы. – 2005. – №4. – С. 20-25.
34. Назаренко Ю. Л. Стегоанализ метода сокрытия информации в изображении замены наименьшего значащего бита (LSB) //European science. – 2018. – №. 3 (35). – С. 23-27.
35. Neeta D., Snehal K., Jacobs D. Implementation of LSB steganography and its evaluation for various bits //2006 1st international conference on digital information management. – IEEE, 2006. – С. 173-178.
36. Fridrich J., Goljan M., Du R. Reliable detection of LSB steganography in color and grayscale images //Proceedings of the 2001 workshop on Multimedia and security: new challenges. – 2001. – С. 27-30.
37. Fridrich J., Goljan M., Du R. Detecting LSB steganography in color, and gray-scale images //IEEE multimedia. – 2001. – Т. 8. – №. 4. – С. 22-28.
38. Juneja M., Sandhu P. S. Improved LSB based Steganography Techniques for Color Images in Spatial Domain //Int. J. Netw. Secur. – 2014. – Т. 16. – №. 6. – С. 452-462.
39. Chen C. C., Chang C. C. LSB-based steganography using reflected gray code //IEICE transactions on information and systems. – 2008. – Т. 91. – №. 4. – С. 1110-1116.
40. Chandramouli R., Memon N. Analysis of LSB based image steganography techniques //Proceedings 2001 international conference on image processing (Cat. No. 01CH37205). – IEEE, 2001. – Т. 3. – С. 1019-1022.
41. Li B. et al. A survey on image steganography and steganalysis //J. Inf. Hiding Multim. Signal Process. – 2011. – Т. 2. – №. 2. – С. 142-172.
42. Fridrich J. Steganography in digital media: principles, algorithms, and applications. – Cambridge University Press, 2009. – 436p.
43. Cheddad A. et al. Digital image steganography: Survey and analysis of current methods //Signal processing. – 2010. – Т. 90. – №. 3. – С. 727-752.
44. Янг У. Реализация алгоритма стеганографии на основе LSB метода // Труды Международной конференции по компьютерной безопасности и криптографии. – 2012. – С. 143-147.
45. Daiyrbayeva E., Yerimbetova A., Toigozhinova A., Maratov Z., Sambetbayeva M. // Learning steganography with a strip transform. 6th International Conference on Computer Science and Engineering (UBMK). – Ankara, 2021. – P.209-212.
46. Al-Afandy K. A. et al. High security data hiding using image cropping and LSB least significant bit steganography //2016 4th IEEE International Colloquium on Information Science and Technology (CiSt). – IEEE, 2016. – С. 400-404.
47. Al-Taani A. T., Al-Issa A. M. A novel steganographic method for gray-level images //International Journal of Computer, Information, and Systems Science, and Engineering. – 2009. – Т. 3. – №. 3. – С. 574-579.
48. Lou D. C., Hu C. H. LSB steganographic method based on reversible histogram transformation function for resisting statistical steganalysis //Information Sciences. – 2012. – Т. 188. – С. 346-358.
49. Deshmukh P. U., Pattewar T. M. A novel approach for edge adaptive steganography on LSB insertion technique //International Conference on Information Communication and Embedded Systems (ICICES2014). – IEEE, 2014. – С. 1-5.
50. Arora H., Bansal C., Dagar S. Comparative study of image steganography techniques //2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN). – IEEE, 2018. – С. 982-985.
51. Vinodhini R. E., Malathi P., Kumar T. G. A survey on DNA and image steganography //2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS). – IEEE, 2017. – С. 1-7.
52. Charan G. S. et al. A novel LSB based image steganography with multi-level encryption //2015 international conference on innovations in information, embedded and communication systems (ICIIECS). – IEEE, 2015. – С. 1-5.
53. Rahman S. et al. A novel approach of image steganography for secure communication based on LSB substitution technique //Computers, Materials & Continua. – 2020. – Т. 64. – №. 1. – С. 31-61.
54. Najih M.N.M. et al. An improved secure image hiding technique using PN-sequence based on DCT-OTP //2017 1st International Conference on Informatics and Computational Sciences (ICICoS). – IEEE, 2017. – С. 47-52.
55. El\_Rahman S.A. A comparative analysis of image steganography based on DCT algorithm and steganography tool to hide nuclear reactors confidential information //Computers & Electrical Engineering. – 2018. – Т. 70. – С. 380-399.
56. Walia E., Jain P., Navdeep N. An analysis of LSB & DCT based steganography //Global Journal of Computer Science and Technology. – 2010. – Т. 10. – № 1. – С. 4-8.
57. Emam A. M., Ouf M. M. Performance evaluation of different universal steganalysis techniques in jpg files //Annales Universitatis Mariae Curie-Skłodowska. Sectio AI, Informatica. – 2012. – Т. 12. – № 3.
58. Goel S., Rana A., Kaur M. A review of comparison techniques of image steganography //Global Journal of Computer Science and Technology. – 2013.
59. Walia E., Jain P., Navdeep N. An analysis of LSB & DCT based steganography //Global Journal of Computer Science and Technology. – 2010. – Т. 10. – № 1. – С. 4-8.
60. Poljicak A. et al. Portable real-time DCT-based steganography using OpenCL //Journal of Real-Time Image Processing. – 2018. – Т. 14. – С. 87-99.
61. Singla D., Syal R. Data security using LSB & DCT steganography in images //Int. J. Of Computational Engineering Research. – 2012. – Т. 2. – № 2. – С. 359-364.
62. Fateh M., Rezvani M., Irani Y. A new method of coding for steganography based on LSB matching revisited //Security and Communication Networks. – 2021. – Т. 2021. – С. 1-15.
63. Maheswari S. U., Hemanth D. J. Different methodology for image steganography-based data hiding //International Journal of Information and Communication Technology. – 2015. – Vol. 7. – № 4-5. – P. 521-536.
64. Thai T. H., Cogranne R., Retraint F. Steganalysis of Jsteg algorithm based on a novel statistical model of quantized DCT coefficients //2013 IEEE International Conference on Image Processing. – IEEE, 2013. – P. 4427-4431.
65. Goel S., Rana A., Kaur M. A review of comparison techniques of image steganography //Global Journal of Computer Science and Technology. – 2013.
66. Abdel-Aziz M. M., Hosny K. M., Lashin N. A. Improved data hiding method for securing color images //Multimedia Tools and Applications. – 2021. – Vol. 80. – P. 12641-12670.
67. Muhuri P. K., Ashraf Z., Goel S. A novel image steganographic method based on integer wavelet transformation and particle swarm optimization //Applied Soft Computing. – 2020. – Vol. 92. – P. 106257.
68. Chu R. et al. A DCT-based image steganographic method resisting statistical attacks //2004 IEEE International Conference on Acoustics, Speech, and Signal Processing. – IEEE, 2004. – Vol. 5. – P. V-953.
69. Hou D. et al. Reversible data hiding in JPEG image based on DCT frequency and block selection //Signal Processing. – 2018. – Vol. 148. – P. 41-47
70. Muhuri P. K., Ashraf Z., Goel S. A novel image steganographic method based on integer wavelet transformation and particle swarm optimization //Applied Soft Computing. – 2020. – Vol. 92. – P. 106257.
71. Zhang T., Ping X. A new approach to reliable detection of LSB steganography in natural images //Signal processing. – 2003. – V. 83. – № 10. – P. 2085-2093.
72. Abd EL-Latif A.A., Abd-El-Atty B., Venegas-Andraca S.E. A novel image steganography technique based on quantum substitution boxes //Optics & Laser Technology. – 2019. – Vol. 116. – P. 92-102.
73. Li Y., Guan Z., Xu C. Digital image self restoration based on information hiding //2018 37th Chinese Control Conference (CCC). – IEEE, 2018. – P. 4368-4372.
74. Xiao J. et al. Research of multi-direction transition probability matrices algorithm for JPEG steganalysis //2012 24th Chinese Control and Decision Conference (CCDC). – IEEE, 2012. – P. 3907-3912.
75. Evsutin O., Melman A., Meshcheryakov R. Algorithm of error-free information embedding into the DCT domain of digital images based on the QIM method using adaptive masking of distortions //Signal Processing. – 2021. – Vol. 179. – P. 107811.
76. Daiyrbayeva, E., Murzin, F., Yerimbetova, A., Toigozhinovа A. Using wavelet transform in image processing//Advanced technologies and computer science. –2020. –№2. –P 15–20.
77. D.J. Evans, N.F. Johnson, and S. Roe, "Wavelet-based steganography," IEEE Transactions on Information Forensics and Security, vol. 1, no. 1, pp. 142-154, 2006.
78. El Safy R. O., Zayed H. H., El Dessouki A. An adaptive steganographic technique based on integer wavelet transform //2009 International Conference on Networking and Media Convergence. – IEEE, 2009. – P. 111-117.
79. Miri A., Faez K. An image steganography method based on integer wavelet transform //Multimedia Tools and Applications. – 2018. – Vol. 77. – P. 13133-13144.
80. Al-Ataby A., Al-Naima F. A modified high capacity image steganography technique based on wavelet transform //changes. – 2008. – Vol. 4. – P. 6.
81. Baby D. et al. A novel DWT based image securing method using steganography //Procedia Computer Science. – 2015. – Т. 46. – С. 612-618.
82. Kadhim I. J., Premaratne P., Vial P. J. High capacity adaptive image steganography with cover region selection using dual-tree complex wavelet transform //Cognitive Systems Research. – 2020. – Т. 60. – С. 20-32.
83. Hemalatha S., Acharya U. D., Renuka A. Wavelet transform based steganography technique to hide audio signals in image //Procedia Computer Science. – 2015. – Т. 47. – С. 272-281.
84. Kumar S., Singh B. K. DWT based color image watermarking using maximum entropy //Multimedia Tools and Applications. – 2021. – Т. 80. – С. 15487-15510.
85. Дайырбаева Э.Н., Еримбетова А.С. Роль метрики в обработке изображений // МНСК-2021 Информационные технологии Материалы 59-й Международной научной студенческой конференции. –Новосибирск, 2021. - С.53
86. Anand A., Singh A. K. Watermarking techniques for medical data authentication: a survey //Multimedia Tools and Applications. – 2021. – Vol. 80. – P. 30165-30197.
87. Cox I. et al. Digital watermarking and steganography. – Morgan kaufmann, 2007.
88. Begum M., Ferdush J., Uddin M. S. A Hybrid robust watermarking system based on discrete cosine transform, discrete wavelet transform, and singular value decomposition //Journal of King Saud University-Computer and Information Sciences. – 2022. – Vol. 34. – №8. – P. 5856-5867.
89. Tao H. et al. Robust image watermarking theories and techniques: A review //Journal of applied research and technology. – 2014. – Vol. 12. – №1. – P. 122-138.
90. Nyeem H., Boles W., Boyd C. Digital image watermarking: its formal model, fundamental properties and possible attacks //EURASIP Journal on Advances in Signal Processing. – 2014. – Vol. 2014. – №1. – P. 1-22.
91. Seo J. S., Yoo C. D. Image watermarking based on invariant regions of scale-space representation //IEEE Transactions on Signal Processing. – 2006. – Vol. 54. – № 4. – P. 1537-1549.
92. Cox I.J. et al. Secure spread spectrum watermarking for multimedia //IEEE transactions on image processing. – 1997. – Vol. 6. – №12. – P. 1673-1687.
93. Seitz J. Digital watermarking for digital media. – IGI Global, 2005. eBook ISBN: 1-59140-520-3
94. Shen H., Chen B. From single watermark to dual watermark: a new approach for image watermarking //Computers & Electrical Engineering. – 2012. – Vol. 38. – № 5. – P. 1310-1324.
95. Fridrich J., Goljan M., Du R. Reliable detection of LSB steganography in color and grayscale images //Proceedings of the 2001 workshop on Multimedia and security: new challenges. – 2001. – P. 27-30.
96. Fridrich J., Goljan M. Practical steganalysis of digital images: state of the art //security and Watermarking of Multimedia Contents IV. – 2002. – Vol. 4675. – P. 1-13.
97. Thanikaiselvan V. et al. A graph theory practice on transformed image: A random image steganography //The Scientific World Journal. – 2013. – Т. 2013.
98. Li F., Tang H., ZouY., Huang Y., Feng Y., Peng L. Research on information security in text emotional steganography based on machine learning. Enterprise Information Systems. –2021. –Vol.15. –№7. – P. 984-1001
99. Chaumont M. Deep learning in steganography and steganalysis // Digital Media Steganography, Academic Press. –2020. –P. 321–349
100. Fridrich J., Golja M., Du R. Reliable Detection of LSB Steganography in Color and Grayscale Images//Proceedings of the 2001 workshop on Multimedia and security: new challenges. – P.27–30.
101. Xu G., Wu H.Z., Shi Y.Q. Structural design of convolutional neural networks for steganalysis // IEEE Signal Process. Lett. 2016. Vol. 23. – № 5. – P. 708–712. 39.
102. Дайырбаева Э.Н. Использование RS анализа в стеганографии // МНСК-2022 Информационные технологии. Материалы 60-ой Международной научной студенческой конференции. – Новосибирск, 2022. – С.9
103. Ye J., Ni J., Yi Y. Deep learning hierarchical representations for image steganalysis // IEEE Trans Inf Forensics Secur. – 2017. Vol. 12, №11. P. 2545-–255
104. Chhikara S., Kumar R. Information theoretic steganalysis of processed image LSB steganography //Multimedia Tools and Applications. – 2023. – Vol. 82. – №9. – P. 13595-13615.
105. Chiesa V. An approach for detecting global image manipulations. Milano, 2014. –121 p.
106. Zhang H. et al. Steganalysis by subtractive pixel adjacency matrix and dimensionality reduction //Science China Information Sciences. – 2014. – Vol. 57. – С. 1-7.
107. Farsi H., Shahi A. Steganalysis of images based on spatial domain and two-dimensional JPEG array //Journal of the Chinese Institute of Engineers. – 2014. – Vol. 37. – № 8. – P. 1055-1063.
108. Chutani S., Goyal A. Improved universal quantitative steganalysis in spatial domain using ELM ensemble //Multimedia Tools and Applications. – 2018. – Т. 77. – С. 7447-7468.
109. Грачев Я. Л., Сидоренко В. Г. Стегоанализ методов скрытия информации в графических контейнерах //Надежность. – 2021. – Т. 21. – №. 3. – С. 39-46.
110. Частикова В. А., Аббасов Т. О., Аббасова С. С. Методика распознавания скрытой информации в изображениях на основе алгоритмов стеганографии //Вестник Адыгейского государственного университета. Серия 4: Естественно-математические и технические науки. – 2020. – № 3 (266). – С. 40-45.
111. Назаренко Ю.Л. Стегоанализ метода сокрытия информации в изображении замены наименьшего значащего бита (LSB) //European science. – 2018. – № 3 (35). – С. 23-27.
112. Киселева А.В., Кудрина М.А. Стеганография и методы стегоанализа //Перспективные информационные технологии (ПИТ 2017). – 2017. – С. 226-229.
113. Курс К.С., Сабирзянова Э.И., Кротова Е. Л. LSB-стеганография //Автоматизированные системы управления и информационные технологии. – 2020. – С. 465-472.
114. Кириченко А.А., Боровых Н.Е. Оценка ёмкости контейнеров для графематического метода текстовой стеганографии //Актуальные проблемы технических наук в России и за рубежом. – 2017. – С. 46-48.
115. Daiyrbayeva E., Yerimbetova A., Nechta I., Merzlyakova E., Toigozhinova A., Turganbayev A. A Study of the Information Embedding Method into Raster Image Based on Interpolation//J. Imaging, 8, 288. -2022.
116. Kameneva A.A. Some methods of image interpolation//Scientific community of students: Collection of materials X International Student Scientific and Practical Conference, Cheboksary. – Cheboksary: Limited Liability Company «Center for Scientific Cooperation «Interactive Plus», 2016. -Р. 121–123.
117. Li Y., Qi F., Wan Y. Improvements on bicubic image interpolation //2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC). – IEEE, 2019. – Vol. 1. – P. 1316-1320.
118. Дайырбаева Э.Н., Еримбетова А.С., Тұрғанбаев А.Ж., Тойгожинова А.Ж., Нурланбек А.Д. Интерполяция арқылы ақпараттарды жасыру жолдарына талдау. //ҚазККА хабаршысы. –Алматы, 2022. – №3. – Б. 376-383.
119. Parsania P.S., Virparia P.V. A comparative analysis of image interpolation algorithms //International Journal of Advanced Research in Computer and Communication Engineering. – 2016. – Vol. 5. – №1. – P. 29-34.
120. Yerimbetova A., Daiyrbayeva E., Cherikbayeva L. Embedding hidden information in images based on bicubic interpolation // Известия НАН РК. Серия «Информатика». – Алматы, 2023. –№1. –P. 50-63
121. Han D. Comparison of commonly used image interpolation methods //Conference of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013). – Atlantis Press, 2013. – P. 1556-1559.
122. Sa Y. Improved bilinear interpolation method for image fast processing //2014 7th International Conference on Intelligent Computation Technology and Automation. – IEEE, 2014. – P. 308-311.
123. Arif F., Akbar M. Resampling air borne sensed data using bilinear interpolation algorithm //IEEE International Conference on Mechatronics, 2005. ICM'05. – IEEE, 2005. – P. 62-65.
124. Li J., Su J., Zeng X. A solution method for image distortion correction model based on bilinear interpolation //Компьютерная оптика. – 2019. – Vol. 43. – № 1. – P. 99-104.
125. Choi J., Kweon I. S. Deep iterative frame interpolation for full-frame video stabilization //ACM Transactions on Graphics (TOG). – 2020. – Vol. 39. – № 1. – P. 1-9.
126. Li D. et al. Deep image compression based on multi-scale deformable convolution //Journal of Visual Communication and Image Representation. – 2022. – Vol. 87. – P. 103573.
127. Hung K. W., Wang K., Jiang J. Image interpolation using convolutional neural networks with deep recursive residual learning //Multimedia Tools and Applications. – 2019. – Vol. 78. – P. 22813-22831.
128. Roy A.M. An efficient multi-scale CNN model with intrinsic feature integration for motor imagery EEG subject classification in brain-machine interfaces //Biomedical Signal Processing and Control. – 2022. – Vol. 74. – P. 103496.
129. Tai Y., Yang J., Liu X. Image super-resolution via deep recursive residual network //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2017. – P. 3147-3155.
130. Siu W.C., Hung K. W. Review of image interpolation and super-resolution //Proceedings of The 2012 Asia Pacific Signal and Information Processing Association Annual Summit and Conference. – IEEE, 2012. – P. 1-10.
131. Дайырбаева Э.Н., Еримбетова А.С. Исследование и внедрение встраивания скрытой информации в изображения на основе интерполяции // Российская конференция с международным участием Распределенные информационно-вычислительные ресурсы (DICR-2022) – Новосибирск, 2022. – С. 45-48.
132. Phung V. M., Nguyen V. T., Dinh H. L. Combining Interpolation Schemes and Lagrange Interpolation on the Unit Sphere in ℝN+ 1 //Ukrainian Mathematical Journal. – 2022. – P. 1-21.
133. Zadorin A. I., Zadorin N. A. Lagrange interpolation and the Newton–Cotes formulas on a Bakhvalov mesh in the presence of a boundary layer //Computational Mathematics and Mathematical Physics. – 2022. – Vol. 62. – № 3. – P. 347-358.
134. Georgiev S.G., Erhan İ.M. Lagrange interpolation on time scales //Journal of Applied Analysis & Computation. – 2022. – Vol. 12. – №4. – P. 1294-1307.
135. Calvi J. P., Manh P. Lagrange interpolation at real projections of Leja sequences for the unit disk //Proceedings of the American Mathematical Society. – 2012. – Vol. 140. – №12. – P. 4271-4284.
136. Исмоилова М.Н., Имомова Ш.М. Интерполяция функции. Вестник Науки и Образования. – 2020. – № 3(81). –С. 5-8.
137. Jung K.H., Yoo K.Y. Data hiding method using image interpolation //Computer Standards & Interfaces. – 2009. – Vol. 31. – №2. – P. 465-470.
138. Yang C. N., Hsu S. C., Kim C. Improving stego image quality in image interpolation based data hiding //Computer Standards & Interfaces. – 2017. – Vol. 50. – P. 209-215.
139. Jana M., Joardar S., Jana B. A New Reversible Data Hiding Scheme by Altering Interpolated Pixels Exploiting Neighbor Mean Interpolation (NMI) //Computational Intelligence in Pattern Recognition: Proceedings of CIPR 2022. – Singapore: Springer Nature Singapore, 2022. – P. 393-402.
140. Jung K. H., Yoo K. Y. Steganographic method based on interpolation and LSB substitution of digital images //Multimedia Tools and Applications. – 2015. – Vol. 74. – P. 2143-2155.
141. Parsania P. S., Virparia P. V. A comparative analysis of image interpolation algorithms //International Journal of Advanced Research in Computer and Communication Engineering. – 2016. – Vol. 5. – №1. – P. 29-34.
142. Amanatiadis A., Andreadis I. A survey on evaluation methods for image interpolation //Measurement Science and Technology. – 2009. – Vol. 20. – №10. – P. 104015.
143. Wang Y. et al. An improved interpolation algorithm using nearest neighbor from VTK //2010 International Conference on Audio, Language and Image Processing. – IEEE, 2010. – P. 1062-1065.
144. Malik A., Sikka G., Verma H. K. Image interpolation based high capacity reversible data hiding scheme //Multimedia Tools and Applications. – 2017. – Vol. 76. – P. 24107-24123.
145. Jung K. H. A survey of interpolation-based reversible data hiding methods //Multimedia Tools and Applications. – 2018. – Vol. 77. – P. 7795-7810.
146. Евсютин О.О., Кокурина А.С., Мещеряков Р.В. Алгоритмы встраивания информации в цифровые изображения с применением интерполяции //Доклады Томского государственного университета систем управления и радиоэлектроники. – 2015. – № 4 (38). – С. 108-112.
147. Мерзлякова Е.Ю. Построение стеганографических систем для растровых изображений, базирующихся на теоретико-информационных принципах:дис….канд.тех.наук:05.13.19 / Сибирский государственный университет телекоммуникаций и информатики. – Новосибирск, 2011. – С.161. Инв. №04201166092.
148. Елтышева (Мерзлякова) Е.Ю., Фионов А.Н. Построение стегосистемы на базе растровых изображений с учетом статистики младших бит // Вестник ФГОБУ ВПО «СибГУТИ». – Новосибирск, 2009. –№ 1. С. 67-84.
149. Ryabko B. Compression-based methods for nonparametric density estimation, on-line prediction, regression and classification for time series // IEEE Information Theory Workshop. – Porto, Portugal, 2008.
150. Нечта И.В. Разработка методов обеспечения безопасности использования информационных технологий, базирующихся на идеях стеганографии: дис.…канд. тех.наук:05.13.17/Сибирский государственный университет телекоммуникаций и информатики. – Новосибирск, 2012. – С.130.
151. Жилкин М.Ю. Теоретико-информационные методы стегоанализа графических данных: дис….канд.тех.наук:05.12.13/ Сибирский государственный университет телекоммуникаций и информатики. Новосибирск, 2009. – С.158.
152. Bezier, P.E. Numerical Control-Mathematics and applications. (1972) John Wileyand Sons, London.
153. Ntoko N-M. A formulation for Bézier-type curves. [Computers in Industry](https://www.scopus.com/sourceid/19080). – 1990. – Vol.15. – №4. – P.363 - 368.
154. Kajla A., Acar T. Bézier–Bernstein–Durrmeyer type operators //Revista de la Real Academia de Ciencias Exactas, Físicas y Naturales. Serie A. Matemáticas. – 2020. – Т. 114. – С. 1-11.
155. Wu Y., Li X. Curve intersection based on cubic hybrid clipping //Visual Computing for Industry, Biomedicine, and Art. – 2022. – Т. 5. – №. 1. – С. 17.
156. Блинова Е. А., Голик А. А. Модификация стеганографического метода на основе встраивания дополнительных значений координат в изображениях формата SVG. – 2018. – C.130-133
157. Анисимова Э. С. Сжатие изображений с помощью квадратичных кривых Безье //Естественные и математические науки в современном мире. – 2014. – №14. – С. 42-46.
158. Ковтун В. Ю., Кинзерявый А. Н. Экспериментальное исследование метода побитового сокрытия информации в векторные изображения //Безпека iнформацii. – 2014. – Т. 20. – №1. – С. 66-70.
159. Bartels R. H., Beatty J. C., Barsky B. A. Bézier Curves//Ch. 10 in [An Introduction to Splines for Use in Computer Graphics and Geometric Modelling.](http://www.amazon.com/exec/obidos/ASIN/1558604006/ref=nosim/ericstreasuretro) San Francisco, CA: Morgan Kaufmann.- 1998. P. 211-245.
160. Менщиков А.А., Шниперов А.Н. Метод скрытого встраивания информации в векторные изображения //Доклады Томского государственного университета систем управления и радиоэлектроники. – 2015. – № 1 (35). – С. 100-106.
161. Karim A.A., Hussein A.M. J. A., Alwan H. M. Image Steganography System Using Bezier Curve //Al-Mansour Journal. – 2019. – Vol. 31. – № 1. – P. 111-133.
162. Al-kufi M. A. H. J., Kadhim O. N., Razaq E. S. Simulate a first-order Bézier curve in image encoding //Journal of Physics: Conference Series. – IOP Publishing, 2020. – . Vol. 1530. – № 1. – P. 012080.
163. Dumka A. et al. Advanced digital image processing and its applications in Big Data. – CRC Press, 2020.
164. Wang X., Ren Q., Jiang D. An adjustable visual image cryptosystem based on 6D hyperchaotic system and compressive sensing //Nonlinear Dynamics. – 2021. – Vol. 104. – №4. – P. 4543-4567.
165. Блинова Е. А., Урбанович П. П. Steganographic method based on hidden messages embedding into Bezier curves of SVG images //Журнал Белорусского государственного университета. Математика. Информатика. – 2021. – №3. – С. 68-83.
166. Das S., Rane S., Vetro A. Hiding information inside structured shapes //2010 IEEE International Conference on Acoustics, Speech and Signal Processing. – IEEE, 2010. – P.1782-1785.
167. S. Rizal, Dong-Seong Kim. Image Transmission in Military Network Using Bézier Curve. Journal of Advances in Computer Networks. –2015. –Vol. 3. –№2, 2015. -P.141-145.
168. Digital Image Interpolation [URL]. Access Mode. Available online: https://disk.yandex.ru/d/ZHx4FV3t1t\_T5g (10.10.2022).

# ПРИЛОЖЕНИЕ А АВТОРСКОЕ СВИДЕТЕЛЬСТВО (3 шт.)



# ПРИЛОЖЕНИЕ В АКТ ВНЕДРЕНИЯ