БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Факультет радиофизики и компьютерных технологий
Кафедра интеллектуальных систем

Аннотация к магистерской диссертации

«Стеганографическое скрытие данных в полях битовых плоскостей изображений»

специальность 1-31 80 08 «Физическая электроника»

Гололобов Андрей Васильевич

Научный руководитель: кандидат технических наук, профессор кафедры интеллектуальных систем, В. С. Садов

2015

РЕФЕРАТ

Магистерская диссертация: 75 страниц, 33 рисунка, 15 источников, 1 приложение.

СТЕГАНОГРАФИЧЕСКАЯ СИСТЕМА, BMP, LSB, КОНТЕЙНЕР, ЦВЕТНОСТЬ, ДИСПЕРСИЯ, ЭНТРОПИЯ, ХИ-КВАДРАТ, RS-мера

Объект исследования – цифровая стеганографическая система на основе цифровых статических изображений в качестве контейнера.

Предмет исследования – статистические свойства цифровых изображений BMP формата.

Цель работы: исследование статистических свойств цифровых изображений, обеспечивающих наилучшую стойкость стеганографической системы, использующей цифровые изображения в качестве контейнеров для встраивания секретного сообщения.

Методы исследования: встраивание данных в пространственные области изображений, замена наименее значащих бит, структурный анализ изображений, стеганографический анализ, экспериментальные исследования.

В ходе работы исследованы свойства цифровых изображений, на которые следует опираться при выборе цифровых изображений в качестве контейнера при разработке стеганостойких систем. В качестве метода встраивания информации в пространственные области изображений выбран метод замены наименее значащих бит.

Научно-практическое значение: результаты диссертации могут быть использованы при практической разработке стеганографических систем, обладающих высокой стеганографической стойкостью.

РЭФЕРАТ

Магістарская дысертацыя: 75 старонак, 33 малюнка, 15 крыніц, 1 дадатак.

СЦЕГАНАГРАФІЧНАЯ СІСТЭМА, BMP, LSB, КАНТЭЙНЕР, КАЛЯРОВАСЦЬ, ДЫСПЕРСІЯ, ЭНТРАПІЯ, XI-КВАДРАТ, RS.

Аб'ект даследавання – лічбовая сцеганаграфічная сістэма на аснове лічбовых статычных малюнкау у якасці кантэйнера.

Прадмет даследавання – статыстычныя ўласцівасці лічбавых малюнкаў BMP фармату.

Мэта работы: даследаванне статыстычных уласцівасцяу лічбовых малюнкау, якія забяспечваюць найлепшую стойкасць стэганаграфічная сістэмы, якая выкарыстоувае лічбовыя малюнкі у якасці кантэйнерай для ўбудавання сакрэтнага паведамлення.

Метады даследавання: ўбудаванне данных у прасторавыя вобласці малюнкаў, замена найменьш значных біт, структурны аналіз малюнкаў, сцеганаграфічны аналіз, эксперыментальныя даследаванні.

Падчас работы даследаваны уласцівасці лічбовых малюнкау, на якія трэба абапірацца пры выбары лічбовых малюнкау у якасці кантэйнера пры распрацоўцы стеганостойких сістэм. У якасці метада будавання інфармацыі ў прасторавыя вобласці малюнкаў абраны метад замены найменш значных біт.

Навукова-практычнае значэнне: вынікі дысертацыі могуць быць выкарыстаны пры практычнай распрацоўцы сцеганаграфічных сістэм, якія валодаюць высокай сцеганаграфічнай стойкасцю.

ABSTRACT

Master thesis: 75 pages, 33 figures, 15 sources, 1 application.

STEGANOGRAPHIC SYSTEM, BMP, LSB, CONTAINER, COLOUR, DISPERSION, ENTROPY, CHI-SQUARE, RS.

Object of research – digital steganographic system based on digital images used as a container.

Subject of research – statistical properties of the digital images of BMP format.

Objective: the study of the statistical properties of digital images which provide the best stability of steganography system which uses the digital images as a container for secret message embedding.

Methods: the data embedding into the images spatial domain, least significant bits replacement, structural analysis of the images, steganographic analysis, experimental studies.

The properties of digital images which should be taken in account while develop of steganographic systems were investigated. The method of least significant bits replacement was selected as the method of data embedding into spatial domain of digital images.

Scientific and practical importance: the results of the thesis can be used in practical design of steganographic systems, which have high steganographic stability.

СОДЕРЖАНИЕ

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ

LSB – Least Significant Bit
RGB – Red Green Blue
JPEG – Joint Photographic Experts Group
BMP – Bitmap Picture
RS – Regular-Singular, то есть «регулярно-сингулярный».
TIFF – Tagged Image File Format
PNG – Portable Network Graphics (формат хранения растровых графических изображений)
TGA – Truevision TGA (TGA – растровый графический формат)
ЦВЗ – цифровой водяной знак
БЧХ – коды Боуза-Чоудхури-Хоквингема (оставляют один из больших классов линейных кодов, исправляющих ошибки)
НЗБ – наименее значащие биты

ВВЕДЕНИЕ

Во все времена на протяжении истории человечества задача защиты информации от несанкционированного доступа оставалась актуальной и в настоящее время остается не решенной до конца. Уже в древнем мире выделилось два основных направления решения этой задачи, существующие и по сегодняшний день: криптография и стеганография. Целью криптографии является скрытие содержимого сообщений за счет их шифрования. В отличие от этого, при стеганографии скрывается сам факт существования тайного сообщения.

Сильным толчком к развитию стеганографии послужило то, что в большинстве стран на криптографию накладываются определенные ограничения: так, например, требуется передача ключей от используемых систем шифрования государству. Обязательна так же регистрация и лицензирование криптографических систем независимо от того, являются они аппаратными или программными средствами. Стеганография не попадает под действие указанных ограничений и является при этом эффективным способом сокрытия данных.

Методы стеганографии применяются не только для скрытной передачи сообщений, но и используют для защиты авторских или имущественных прав на цифровое изображение, фотографии или другие оцифрованные произведения искусства. Преимущества, которые дают представление и передача сообщений в цифровом виде, могут оказаться перечеркнутыми легкостью, с которой возможно их воровство или модификация. Поэтому разрабатываются различные меры защиты информации, организационного и технического характера. Один из наиболее эффективных технических средств защиты мультимедийной информации заключается во встраивании в защищаемый объект невидимых меток – цифровых водяных знаков. Они могут содержать много полезной информации: когда создан файл, кто владеет авторскими правами, как вступить в контакт с автором и т.д. Все внесенные данные могут рассматриваться как веские доказательства при рассмотрении вопросов и судебных разбирательств об авторстве или для доказательства факта нелегального копирования и часто имеют решающее значение.

Наиболее распространенным на сегодняшний день методом цифровой стеганографии является метод, заключающейся во вложении скрываемого сообщения в изображение путем модификации наименее значимых бит (LSB). Цифровые изображения представляют собой матрицу пикселов. Пиксел – это единичный элемент изображения. Он имеет фиксированную разрядность двоичного представления.

Например, в простейшей черно-белой картинке каждый пиксель описывается одним байтом, который кодирует яркость пикселя: ноль — черный, 255 — белый, все остальное — градации серого. Если изменить любой байт такого файла или, что тоже самое, отдельные биты этого байта, то соответствующий ему пиксель изменит яркость. При этом изменение разных битов влияет на яркость пикселя по-разному: первый очень сильно, второй слабее, а последний, восьмой бит может добавить байту (а значит, и пикселю) только единицу. Нормальный человек не заметит изменение яркости точки на одну (1/255) градацию серого. А значит, абсолютно не важно, каковы последние биты каждого байта. И их можно обнулять, переставлять, заменять, — картинка при этом будет казаться одинаковой.

Достоинства метода заключаются в его простоте и сравнительно большом объеме встраиваемых данных. Однако он имеет два серьёзных недостатка:

Скрытое сообщение легко разрушить. Для этого необходимо просто записать в один или два младших бита каждого байта графического изображения нули или единицы, тогда, если картинка не содержала скрытого сообщения, то видимых искажений не появится, а если в картинке было скрыто сообщение, то оно будет испорчено. То есть, те же достоинства, которые используются для сокрытия информации, могут быть использованы и для незаметной борьбы со стеганографией.

Не обеспечена секретность встраивания информации. Нарушителю точно известно местоположение всего сообщения. В случае перехвата информации, если у перехватчика возникнет подозрение, на то, что в изображении скрыто какое-то сообщение, ему относительно нетрудно будет извлечь эту информацию, так как количество возможных способов извлечения невелико.

Целью данной магистерской диссертации является исследование свойств естественных изображений-контейнеров, обеспечивающих наилучшую секретность встраивания информации в результате применения статистических методов стеганографического анализа.

ГЛАВА 1 ОСНОВНЫЕ ПОЛОЖЕНИЯ СТЕГАНОГРАФИИ

1.1 Основные определения

Стеганографию можно разделить на 3 раздела:

- классическая стеганография – включает в себя все “некомпьютерные методы”.
- компьютерная стеганография – направление классической стеганографии, основанное на особенностях компьютерной платформы.
- цифровая стеганография – направление классической стеганографии, основанное на сокрытии или внедрении дополнительной информации в цифровые объекты, вызывая при этом некоторые искажения этих объектов. Используется избыточность аудио- и визуальной информации [1].

Именно цифровая стеганография представляет собой наибольший интерес, с точки зрения защиты информации, как наиболее перспективное направление. Ее мы рассмотрим подробнее.

Основные положения стеганографии:

- Методы скрытия должны обеспечивать аутентичность и целостность файла.
- Предполагается, что криптографу полностью известны возможные стеганографические методы.
- Безопасность методов основывается на сохранении стеганографическим преобразованием основных свойств открыто передаваемого файла при внесении в него секретного сообщения и некоторой неизвестной противнику информации - ключа.
- Даже если факт скрытия сообщения стал известен противнику через сообщника, извлечение самого секретного сообщения представляет сложную вычислительную задачу [1].

Несмотря на многочисленные открытые публикации и ежегодные конференции, длительное время стеганография не имела сложившейся терминологии. Основные понятия стеганографии были согласованы в 1996 г. на 1-й Международной конференции по скрытию данных – Information Workshop on Information Hiding ‘96. Тем не менее, даже такое основополагающее понятие как «стеганография» разными специалистами трактуется неодинаково [2].

Приведем определения наиболее важных, с точки зрения стеганографии, терминов.

Стеганосистема – система, осуществляющая встраивание и выделение одной битовой последовательности из другой. Последовательность, подлежащая скрытию, называется сообщением. Последовательность, в которую осуществляется встраивание, называется контейнером. Если в контейнер не встраивалось сообщение, то он называется пустым, иначе – заполненным. Как правило, в составе стеганосистемы дополнительно выделяют подсистемы, такие как прекодер, стеганокодер, стеганодетектор, декодер [3]. Сравнительно недавно была разработана математическая модель стеганосистемы.

В любой стеганосистеме важную роль играет стеганографический протокол – порядок действий, к которым прибегают две или более сторон, с целью решения определенных задач [4].

Цифровой водяной знак (ЦВЗ) – внедренная в мультимедийный сигнал информация, назначение которой – аутентификация содержимого, охрана прав собственника, защита от копирования и т.п.

Стеганосистема образует стеганоканал, по которому передается заполненный контейнер. Этот канал считается подверженным воздействиям со стороны нарушителей. Следуя [5], в стеганографии обычно рассматривается постановка задачи в виде «проблема заключенных», желающих тайно обмениваться сообщениями посредством передачи их в скрытом. Пассивный нарушитель может лишь обнаружить факт наличия стеганоканала и (возможно) читать сообщения. Диапазон действий активного нарушителя значительно шире. Скрытое сообщение может быть им удалено или разрушено. В этом случае передающая и, возможно, принимающая сторона узнают о факте вмешательства. Действия злоумышленного нарушителя наиболее опасны. Он способен не только разрушать, но и создавать ложные сообщения.

При построении стеганосистемы должны учитываться следующие положения, многие из которых лежат в основе критериев эффективности стеганографических алгоритмов изображений:

- стеганосистема должна иметь приемлемую вычислительную сложность реализации;
- заполненный контейнер должен быть визуально неотличим от незаполненного;
- должна обеспечиваться необходимая пропускная способность (что особенно актуально для стеганосистем скрытой передачи данных);
- методы скрытия должны обеспечивать аутентичность и целостность секретной информации для авторизованного лица;

- потенциальный нарушитель имеет полное представление о стеганосистеме и детали её реализации, единственное, что ему неизвестно, – это ключ, с помощью которого только его обладатель может установить факт наличия и содержание скрытого сообщения;
- если факт существования скрытого сообщения становится известным нарушителю, это не должно позволить последнему извлечь его до тех пор, пока ключ сохраняется в тайне;
- нарушитель должен быть лишен любых технических и других преимуществ в распознании или, по крайней мере, раскрытии содержания секретных сообщений [3].

1.2 Методы встраивания в пространственные области изображений

Алгоритмы, осуществляющие скрытие данных в пространственной области, внедряют ЦВЗ в области исходного изображения. Их преимуществом является то, что для внедрения ЦВЗ нет необходимости выполнять вычислительно громоздкие линейные преобразования изображений. ЦВЗ внедряется за счет манипуляций яркостью или цветовыми составляющими$\big(r(x,y), g(x,y), b(x,y)\big)$. Рассмотрим некоторые из этих алгоритмов [8].

1.2.1 Встраивание в незначащие элементы контейнера

Цифровые изображения представляют собой матрицу пикселов. Младший значащий бит изображения несет в себе меньше всего информации. Известно, что человек обычно не способен заметить изменение в этом бите. Фактически, он является шумом. Поэтому его можно использовать для встраивания информации. Достоинства рассматриваемого метода заключаются в его простоте и сравнительно большом объеме встраиваемых данных [6].

1.2.2 Метод Kutter

Пусть изображение имеет RGB-кодировку. Встраивание выполняется в канал синего цвета, так как к синему цвету система человеческого зрения наименее чувствительна. Пусть s_i – встраиваемый бит, $I = \{R, G.B\}$ – контейнер, $p = (x, y)$ – псевдослучайная позиция, в которой выполняется вложение. Секретный бит встраивается в канал синего цвета путем модификации яркости $l(p) = 0.299r(p) + 0587g(p) + 0.114b(p)$ (1)

$$b`(p) = \begin{cases} b(p) + ql(p), \text{ если } s_i = 0 \\ b(p) - ql(p), \text{ если } s_i = 1, \end{cases} \quad (2)$$

где q – константа, определяющая энергию встраиваемого сигнала. Ее величина зависит от предназначения схемы. Чем больше q, тем выше робастность вложения, но тем сильнее его заметность. Максимальное отклонение синей цветовой составляющей при условии неизменности двух других цветов составляет 9–26%. Цветовая компонента каждого пикселя описывается одним байтом. Изменение происходит по маске 11100011, то есть модификации подлежат 4, 5 или 6 биты. Отклонение интенсивности цвета в данном случае не превышает 6,3%, а общее изменение яркости пикселя не превышает 1% [6].

1.2.3 Метод Bruyndonckx

ЦВЗ представляет собой строку бит. Для повышения помехоустойчивости применяется код БЧХ. Внедрение осуществляется за счет модификации яркости блока 8×8 пикселов. Процесс встраивания осуществляется в три этапа:

- классификация, или разделение пикселов внутри блока на две группы с примерно однородными яркостями.
- разбиение каждой группы на категории. Для этого на блоки накладываются маски, разные для каждой группы и каждого блока. Назначение масок состоит в обеспечении секретности внедрения
- модификация средних значений яркости каждой категории в каждой группе [6].

1.2.4 Метод Langelaar

Алгоритм работает с блоками 8×8. Вначале создается псевдослучайная маска нулей и единиц такого же размера $pat(x, y) \in \{0, 1\}$. Далее каждый блок B делится на два субблока B_0 и B_1, в зависимости от значения маски. Для каждого субблока вычисляется среднее значение яркости l_0 и l_1. Далее выбирается некоторый порог α, и бит ЦВЗ встраивается следующим образом:

$$s = \begin{cases} 1, \ l_0 - l_1 > +\alpha \\ 0, \ l_0 - l_1 < -\alpha, \end{cases} \quad (3)$$

Если это условие не выполняется, мы изменяем значение яркости пикселов субблока B_1. Для извлечения бита ЦВЗ вычисляются средние значения яркости субблоков – $l\hat{}_0$ и $l\hat{}_1$. Разница между ними позволяет определить искомый бит: [6]

$$s = \begin{cases} 1, & \hat{l}_0 - \hat{l}_1 > 0, \\ 0, & \hat{l}_0 - \hat{l}_1 < 0. \end{cases} \quad (4)$$

1.2.5 Метод Pitas

ЦВЗ представляет собой двумерный массив бит размером с изображение, причем число единиц в нем равно числу нулей. Существует несколько версий алгоритма, предложенного Питасом. Вначале предлагалось встраивать бит ЦВЗ в каждый пиксел изображения, но позже было решено использовать для этой цели блоки размером 2×2 или 3×3 пиксела, что делает алгоритм более робастным к сжатию или фильтрации. ЦВЗ складывается с изображением: $\hat{l}(x, y) = l(x, y) + \alpha s(x, y)$. В случае использования для внедрения блоков детектор ЦВЗ вычисляет среднее значение яркости этого блока. Отсюда появляется возможность неравномерного внедрения ЦВЗ в пиксели, то есть величина $\alpha \neq \text{const}$. Таким образом можно получить ЦВЗ, оптимизированный по критерию робастности к процедуре сжатия алгоритмом JPEG. Для этого в блоке 8×8 элементов заранее вычисляют «емкость» каждого пикселя (с учетом ДКП и матрицы квантования JPEG). Затем ЦВЗ внедряют в соответствии с вычисленной емкостью. Эта оптимизация производится раз и навсегда, и найденная маска применяется для любого изображения [6].

1.2.6 Метод Rongen

Также, как и в предыдущем алгоритме, ЦВЗ представляет собой двумерную матрицу единиц и нулей с примерно равным их количеством. Пиксели, в которые можно внедрять единицы (то есть робастные к искажениям), определяются на основе некоторой характеристической функции (характеристические пиксели). Эта функция вычисляется локально, на основе анализа соседних пикселов. Характеристические пиксели составляют примерно 1/100 от общего числа, так что не все единицы ЦВЗ встраиваются именно в эти позиции. Для повышения количества характеристических пикселов в случае необходимости предлагается осуществлять небольшое предыскажение изображения. Детектор находит значения характеристических пикселов и сравнивает с имеющимся у него ЦВЗ. Если в изображении ЦВЗ не содержится, то в характеристических пикселах количество единиц и нулей будет примерно поровну [6].

1.2.7 Метод Patchwork

В основе алгоритма Patchwork лежит статистический подход. Вначале псевдослучайным образом на основе ключа выбираются два пикселя изображения.

Затем значение яркости одного из них увеличивается на некоторое значение (от 1 до 5), значение яркости другого уменьшается на то же значение. Далее этот процесс повторяется большое число раз (~10000) и находится сумма значений всех разностей. По значению этой суммы судят о наличии или отсутствии ЦВЗ в изображении. Авторами также предложены улучшения основного алгоритма для повышения его робастности. Вместо отдельных пикселов предлагается использовать блоки, или patches. Отсюда и название алгоритма. Алгоритм Patchwork является достаточно стойким к операциям сжатия изображения, его усечения, изменения контрастности. Основным недостатком алгоритма является его неустойчивость к аффинным преобразованиям, то есть поворотам, сдвигу, масштабированию. Другой недостаток заключается в малой пропускной способности. Так, в базовой версии алгоритма для передачи 1 бита скрытого сообщения требуется 20000 пикселов [6].

1.2.8 Метод Bender

Алгоритм, основанный на копировании блоков из случайно выбранной текстурной области в другую, имеющую сходные статистические характеристики. Это приводит к появлению в изображении полностью одинаковых блоков. Эти блоки могут быть обнаружены следующим образом:

- анализ функции автокорреляции стегоизображения и нахождение ее пиков;
- сдвиг изображения в соответствии с этими пиками и вычитание изображения из его сдвинутой копии;
- разница в местоположениях копированных блоков должна быть близка к нулю. Поэтому можно выбрать некоторый порог и значения, меньшие этого порога по абсолютной величине, считать искомыми блоками.

Так как копии блоков идентичны, то они изменяются одинаково при преобразованиях всего изображения.

Если сделать размер блоков достаточно большим, то алгоритм будет устойчивым по отношению к большинству из негеометрических искажений. В проведенных экспериментах показана робастность алгоритма к фильтрации, сжатию, поворотам изображения. Основным недостатком алгоритма является исключительная сложность нахождения областей, блоки из которых могут быть заменены без заметного ухудшения качества изображения. Кроме того, в данном алгоритме в качестве контейнера могут использоваться только достаточно текстурные изображения [6].

1.3 Классификация стеганографических атак

Субъективная атака. Аналитик внимательно рассматривает изображение (слушает аудиозапись), пытаясь определить “на глаз”, имеется ли в нем скрытое сообщение. Ясно, что подобная атака может быть проведена лишь против совершенно незащищенных стеганосистем. Тем не менее, она, наверное, наиболее распространена на практике, по крайней мере, на начальном этапе вскрытия стегосистемы**.**

Атака на основе известного заполненного контейнера. В этом случае у нарушителя есть одно или несколько стего. В последнем случае предполагается, что встраивание скрытой информации осуществлялось отправителем одним и тем же способом. Задача аналитика может состоять в обнаружении факта наличия стеганоканала (основная), а также в его извлечении или определения ключа. Зная ключ, нарушитель получит возможность анализа других стеганосообщений.

Атака на основе известного встроенного сообщения. Этот тип атаки в большей степени характерен для систем защиты интеллектуальной собственности, когда в качестве водяного знака используется известный логотип фирмы. Задачей анализа является получение ключа. Если соответствующий скрытому сообщению заполненный контейнер неизвестен, то задача крайне трудно решаема.

Атака на основе выбранного пустого контейнера. В этом случае аналитик способен заставить отправителя пользоваться предложенным ему контейнером. Например, предложенный контейнер может иметь большие однородные области (однотонные изображения), и тогда будет трудно обеспечить секретность внедрения.

Атака на основе известной математической модели контейнера или его части. При этом атакующий пытается определить отличие подозрительного сообщения от известной ему модели. Например, допустим, что биты внутри отсчета изображения коррелированы. Тогда отсутствие такой корреляции может служить сигналом об имеющемся скрытом сообщении. Задача внедряющего сообщение заключается в том, чтобы не нарушить статистики контейнера. Внедряющий и атакующий могут располагать различными моделями сигналов, тогда в информационно - скрывающем противоборстве победит имеющий лучшую модель[5].

Атака на основе известной математической модели естественного контейнера представляет наибольший интерес для данной научной работы и может быть в дальнейшем применена для исследования статистических свойств контейнера, позволяющих обеспечить наибольшую стойкость стеганографической системы.

1.4 Анализ метода замены наименее значащих бит

Суть метода замены наименее значащего бита (Least Significant Bits - LSB) заключается в сокрытии информации путем изменения последних битов изображения, кодирующих цвет на биты скрываемого сообщения. Разница между пустым и заполненным контейнерами должна быть не ощутима для органов восприятия человека.

В BMP изображение хранится как матрица значений оттенков цвета для каждой точки хранимого изображения. Если каждая из компонент пространства RGB (их еще называют каналами цвета) хранится в одном байте, она может принимать значения от 0 до 255 включительно, что соответствует 24-х битной глубине цвета. Особенность зрения человека заключается в том, что оно слабо различает незначительные колебания цвета. Для 24-х битного цвета изменение в каждом из трех каналов одного наименее значимого бита (то есть крайнего правого) приводит к изменению менее чем на 1% интенсивности данной точки, что позволяет изменять их незаметно для глаза по своему усмотрению.

Принцип работы стеганографического метода заключается в следующем. Пусть, имеется 24-х битное изображение в градациях серого. Пиксел

кодируется 3 байтами, и в них расположены значения каналов RGB. Изменяя наименее значащий бит, мы меняем значение байта на единицу. Такие градации, мало того, что незаметны для человека, могут вообще не отобразиться при использовании низкокачественных устройств вывода.

Приведенный ниже пример показывает, как сообщение может быть скрыто в первых восьми байтах, относящихся к трем пикселям в 24-битного изображения:

Pixels: (00100111 11101001 11001000)
(00100111 11001000 11101001)
(11001000 00100111 11101001)
A: 01000001

Result: (00100110 11101001 11001000)
(00100110 11001000 11101000)
(11001000 00100111 11101001) .

В примере подчеркнуты только биты только те три бита, которые были фактически изменены. Применение стеганографического метода LSB в среднем требует, что только половина бит изображения-контейнера были изменены.

Небольшая модификация этой стеганографической техники позволяет использовать для встраивания сообщения два или более младших битов на байт. Это увеличивает объем скрытой информации в объекте-контейнере, но скрытность сильно снижается, что облегчает обнаружение стеганографии. Другие вариации этого метода включают в себя нивелирование статистических изменений в изображении. Некоторые интеллектуальное программное обеспечение для выявления стеганографии проверяет области, которые состоят из одного сплошного цвета. Для повышения скрытности следует избежать записи изменений в эти пиксели.

Преимущества метода:

- размер файла-контейнера остается неизменным;
- при замене одного бита в канале синего цвета внедрение невозможно заметить визуально;
- возможность варьировать пропускную способность, изменяя количество заменяемых бит.

Недостатки метода:

- Скрытое сообщение легко разрушить, например, при сжатии или отображении.
- Не обеспечена секретность встраивания информации. Точно известно местоположение зашифрованной информации. Для преодоления этого недостатка можно встраивать информацию не во все пикселы изображения, а лишь в некоторые из них, определяемые по псевдослучайному закону в соответствии с ключом, известному только законному пользователю. Пропускная способность при этом уменьшается [7].

Выводы

В главе рассмотрены основные понятия и положения стеганографии, исследованы принципы построения цифровых стеганосистем. В качестве ключевых принципов при построении стеганографических систем можно выделить визуальную неразличимость заполненного и незаполненного контейнеров, а также аутентичность и целостность секретной информации.

Произведен краткий анализ существующих методов встраивания информации в пространственные области цифровых изображений, основанных на манипуляции яркостью либо цветовыми составляющими изображений. В качестве достоинств приведенных методов можно выделить отсутствие необходимости выполнять вычислительно громоздкие линейные преобразования изображений, что удовлетворяет одному из положений построения стеганосистем. Одним из наиболее часто используемых методов встраивания информации в пространственные области изображений является метод замены наименее значащих бит цифровых изображений, который в дальнейшем может быть использован в работе в качестве используемого метода встраивания информации. Изменение количества заменяемых бит позволяет варьировать пропускную способность стеганографической системы, однако,

существует необходимость исследования возможности использования старших бит для встраивания информации.

Дана классификация применяемых методов стеганографических атак. Субъективная атака, основанная на визуальной неразличимости пустого контейнера и контейнера, содержащего стего, является одной из наиболее распространенных видов атак, используемых для обнаружения факта наличия скрытого сообщения. Однако, в данной работе наибольший интерес представляет атака на основе известной математической модели контейнера или его части. Наличие скрытой информации не должно нарушать статистических свойств исходного цифрового изображения, используемого в качестве контейнера. Именно атака на основе известной математической модели может быть далее применена для оценки влияния использования старших бит изображений на стойкость стеганографической системы.

ГЛАВА 2 АНАЛИЗ КРИТЕРИЕВ ВЫБОРА КОНТЕЙНЕРА

2.1 Общие критерии выбора контейнеров

Существенное влияние на надежность стегосистемы и возможность обнаружения факта передачи скрытого сообщения оказывает выбор контейнера.

По протяженности контейнеры можно подразделить на два типа: непрерывные (потоковые) и ограниченной (фиксированной) длины. Особенностью потокового контейнера является то, что невозможно определить его начало или конец. Более того, нет возможности узнать заранее, какими будут последующие шумовые биты, что приводит к необходимости включать скрывающие сообщение биты в поток в реальном масштабе времени, а сами скрывающие биты выбираются с помощью специального генератора, задающего расстояние между последовательными битами в потоке.

В непрерывном потоке данных самая большая трудность для получателя - определить, когда начинается скрытое сообщение. При наличии в потоковом контейнере сигналов синхронизации или границ пакета, скрытое сообщение начинается сразу после одного из них. В свою очередь, для отправителя возможны проблемы, если он не уверен в том, что поток контейнера будет достаточно долгим для размещения целого тайного сообщения.

При использовании контейнеров фиксированной длины отправитель заранее знает размер файла и может выбрать скрывающие биты в подходящей псевдослучайной последовательности. С другой стороны, контейнеры фиксированной длины, как это уже отмечалось выше, имеют ограниченный объем и иногда встраиваемое сообщение может не поместиться в файл-контейнер.

Другой недостаток заключается в том, что расстояния между скрывающими битами равномерно распределены между наиболее коротким и наиболее длинным заданными расстояниями, в то время как истинный случайный шум будет иметь экспоненциальное распределение длин интервала. Конечно, можно породить псевдослучайные экспоненциально распределенные числа, но этот путь обычно слишком трудоемок. Однако на практике чаще всего используются именно контейнеры фиксированной длины, как наиболее распространенные и доступные.

Возможны следующие *варианты контейнеров*:

- Контейнер генерируется самой стегосистемой. Такой подход можно назвать конструирующей стеганографией.
- Контейнер выбирается из некоторого множества контейнеров. В этом случае генерируется большое число альтернативных контейнеров, чтобы затем выбрать наиболее подходящий контейнер, используемый для сокрытия сообщения. Такой подход можно назвать селектирующей стеганографией. В данном случае, при выборе оптимального контейнера из множества сгенерированных, важнейшим требованием является естественность контейнера. Единственной же проблемой остается то, что даже оптимально организованный контейнер позволяет спрятать незначительное количество данных при очень большом объеме самого контейнера.
- Контейнер поступает извне. В данном случае отсутствует возможность выбора контейнера и для сокрытия сообщения берется первый попавшийся контейнер, не всегда подходящий к встраиваемому сообщению. Назовем это безальтернативной стеганографией [5].

В настоящее время большинство исследований в области стеганографии посвящено использованию в качестве стеганоконтейнеров цифровых изображений. Это обусловлено следующими причинами:

- существованием практически значимой задачи защиты фотографий, картин, видео от незаконного тиражирования и распространения;
- относительно большим объемом цифрового представления изображений, что позволяет внедрять сообщение большого объема либо повышать скрытность внедрения;
- заранее известным размером контейнера, отсутствием ограничений, накладываемых требованиями реального времени;
- наличием в большинстве реальных изображений текстурных областей, имеющих шумовую структуру и хорошо подходящих для встраивания информации;
- слабой чувствительностью человеческого глаза к незначительным изменениям цветов изображения, его яркости, контрастности, содержанию в нем шума, искажениям вблизи контуров;
- хорошо разработанными в последнее время методами цифровой обработки изображений.

Надо отметить, что последняя причина вызывает и значительные трудности в обеспечении скрытности секретных сообщений: чем более

совершенными становятся методы сжатия, тем меньше остается возможностей для встраивания посторонней информации[3].

2.2 Классификация критериев выбора контейнера для LSB-метода

От выбора контейнера зависит объем секретного сообщения, а также устойчивость стегоконтейнера к различным видам анализа: визуального или статистического. Способов сокрытия данных много, однако, проблема выбора подходящего контейнера до сих пор не решена. При исследовании было найдено всего несколько источников, в которых затрагивалась данная проблема.

Выбор контейнера должен рассматриваться с точки зрения метода внедрения данных, так как именно он определяет биты, которые будут модифицированы на биты сообщения. Также должен учитываться тот факт, что существуют методы анализа, позволяющие обнаружить секретное сообщение.

На данном этапе исследований выбор контейнера сделан для метода замены младших бит (LSB-метода), на основе которого сделано большинство программ внедрения сообщений. Учитывалось влияние визуального стеганоанализа, как начального этапа анализа контейнера на наличие сообщения [8].

Классификация критериев выбора контейнера:

- отказ от общеизвестных изображений в качестве контейнера, как, например, картины «Джоконда»;
- отказ от использования в качестве контейнера изображений, конвертированных из JPEG-формата в формат BMP;
- получение изображения при помощи фотоаппарата или сканера, а не при помощи графических редакторов;
- большой размер контейнера;
- отсутствие полезной составляющей на младших битовых плоскостях изображения;
- зашумленность;
- отсутствие плавных переходов и монотонных областей;
- «пестрость»;
- большое число перепадов яркости;
- наличие большого числа пикселей, оттенки цветов которых плохо различаются глазом человека (зеленый, желтый).

Эти критерии в достаточной мере учитывают все особенности контейнера, необходимые для получения стеганоустойчивого контейнера к визуальному стеганоанализу для метода замены младших бит [9].

2.3 Цветность изображения как критерий выбора контейнера

На визуальную скрытность данных влияет цветность изображения, то есть наличие цветовых областей того или иного цвета. Это объясняется неравномерной чувствительностью человеческого глаза к малым изменениям различных длин волн видимого диапазона. Человеческий глаз обладает свойством порога цветоразличения при небольших цветовых отличиях, то есть он воспринимает цвет и его «соседний» цвет как один. Величина этого порога неодинакова для разных цветов. Этот эффект представлен на рисунке 2.1:

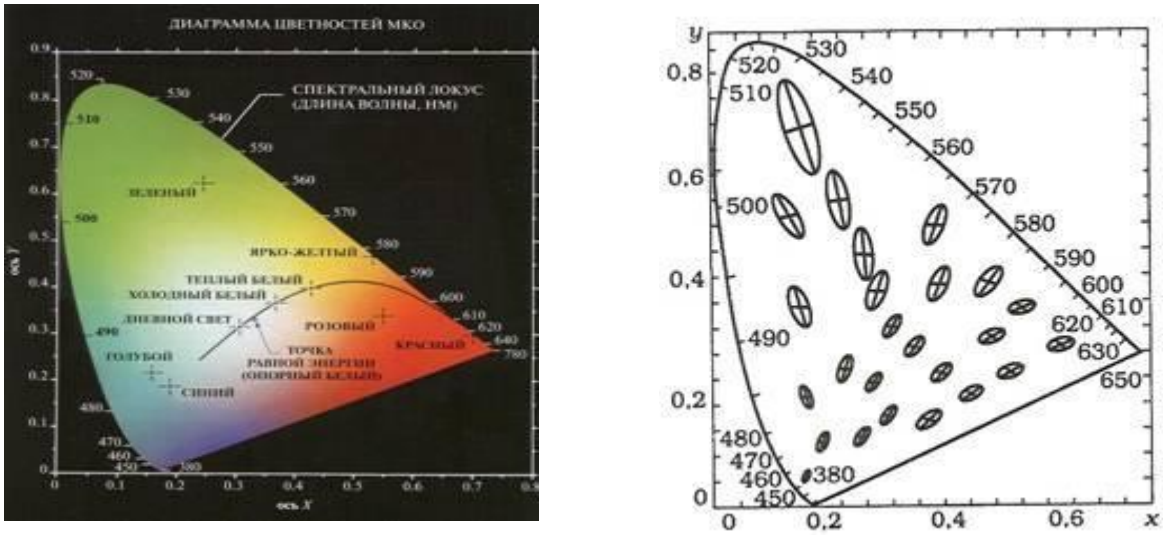

Рисунок 2.1 – Диаграмма цветностей и пороговые эллипсы

Таким образом, замена одинакового количества младших бит красной, синей области будет более опасной для обнаружения произведенной замены глазом, чем младших бит желтой или зеленой области за счет разного порога различимости этих цветов. Выбор контейнера, который содержит наибольшие области зеленого, желтого и их смесей с белым цветом обеспечит наилучшую скрытность данных с точки зрения визуального стеганоанализа [4].

2.4 Критерий эффективности в стеганографии изображений

Под термином «эффективность» в стеганографии будем понимать возможность решения с помощью цифровых изображений основных задач стеганографии: быстро и скрытно передавать большие объемы информации. Существует очень большое количество факторов, влияющих на эффективность стеганографии цифровых изображений.

Среди этих факторов можно выделить группу технических критериев эффективности, которые поддаются строгому математическому описанию и имеют некоторый набор численных характеристик. В качестве примера такого критерия можно привести отношение максимального размера встраиваемого сообщения, не приводящего к искажению изображения, к размеру самого контейнера.

С другой стороны, существуют критерии эффективности, не поддающиеся техническому описанию, но по-прежнему играющие исключительную роль в формировании понятия «эффективность». Рассматривая несколько графических форматов, можно утверждать, что применять один из них эффективнее, чем другой. Причиной для этого может являться то, что один из форматов имеет гораздо большее распространение (в том числе, в сети Интернет), чем остальные. Более того, использование некоторых форматов для нетипичных для них целей само по себе может быть подозрительным и провоцировать атаки. Например, выложенные на сайт в сети Интернет фотографии друзей в формате BMP (имеющие размер порядка нескольких мегабайт) определенно вызовут подозрение у посетителей (ведь современные алгоритмы сжатия позволяют сжимать фотографии в 20-30 раз с приемлемой потерей качества). К тому же, для некоторых форматов (например, упомянутый выше формат BMP) разработан широчайший спектр методов и инструментов стеганоанализа, и эти форматы являются более уязвимыми, а значит и менее эффективными с точки зрения стеганографии.

Проанализируем наиболее важные критерии эффективности применения цифровых изображений в стеганографии.

Скрытность или стеганографическая стойкость.

Удовлетворение требованию скрытности является обязательным для абсолютно любой стеганосистемы. В применении к графической стеганосистеме, стойкость связана с изменениями (искажениями), вносимыми в исходное изображение при встраивании сообщения. Требование стойкости считается невыполненным, если изображение поддается атаке посредством простого визуального анализа. Данная стеганосистема обладает крайне низкой эффективностью и не может найти практического применения, так как не соответствует минимальному уровню безопасности (рисунок 2.2).

1 2

Рисунок 2.2 – Результат работы алгоритма, не отвечающего требованиям стойкости: 1 – исходное изображение, 2 – изображение со встроенным сообщением

Как правило, при создании стеганографических алгоритмов, наибольший объем исследований связан именно с обеспечением скрытности. Производятся эксперименты, позволяющие установить, как изменение той или иной части файла-контейнера влияет на результирующее изображение. Стойкость стеганографического алгоритма в значительной степени определяется размерами встраиваемого сообщения.

Размер встраиваемого сообщения

Эффективность использования цифрового изображения для хранения секретной информации в значительной мере определяется максимальным возможным размером секретного сообщения. Как правило, численно этот критерий характеризуется процентным соотношением между объемом встраиваемого сообщения и исходным объемом контейнера. В отношении изображений, данная величина варьируется в зависимости от используемого графического формата.

Главным «ограничителем» максимального размера сообщения для конкретного графического файла выступает описанное выше требование скрытности. В стеганографии имеется фундаментальная зависимость между стойкостью встраивания и размером встраиваемого сообщения. Эта зависимость имеет обратно пропорциональный характер: чем больше объем встраиваемого в заранее заданный контейнер сообщения, тем ниже надежность сокрытия этой информации в контейнере (рисунок 2.3).

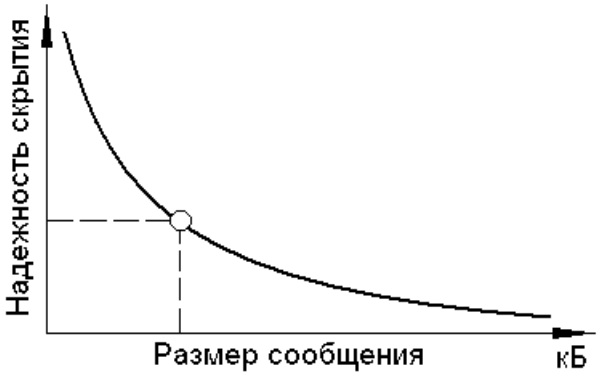

Рисунок 2.3 – Зависимость надежности сокрытия информации от объема сообщения

Казалось бы, приведенная закономерность не позволяет увеличивать эффективность стеганографического встраивания информации путем наращивания размера сообщения. Но это не так. Существует несколько методов повышения размеров сообщения без ущерба стойкости, о которых речь пойдет дальше.

Устойчивость к модификации заполненного контейнера (сжатию)

Устойчивость к модификации характеризует вероятность восстановления сообщения, при условии некоторой модификации заполненного контейнера. Частным случаем модификации является сжатие с потерями. Особое значение этот фактор эффективности имеет для технологий внедрения цифровых водяных знаков.

Модификация заполненного контейнера может осуществляться как непреднамеренно (сжатие, ошибки при передаче файла по каналу связи с помехами), так и преднамеренно (попытка нарушить авторские права путем уничтожения ЦВЗ). Повышение устойчивости к сжатию осуществляется путем тщательного исследования алгоритмов компрессии с целью определения областей контейнера, не подвергающихся модификациям. Действенным методом борьбы с преднамеренным разрушением ЦВЗ может считаться встраивание информации в ту область файла-контейнера, изменение которой приводит к деградации изображения. Традиционным и достаточно мощным способом борьбы с «помехами» может служить увеличение избыточности встраиваемого сообщения.

Объем вычислений, необходимый для встраивания сообщения в цифровое изображение

Несмотря на стремительный рост возможностей современных компьютеров, проблема вычислительной сложности алгоритмов встраивания продолжает играть ключевую роль в некоторых областях применения стеганографии. Это, как правило, информационные системы реального времени, где временные рамки выполнения алгоритма сильно ограничены. В качестве примера, можно привести гипотетический скрытый канал голосовой связи, работающий посредством встраивания аудиоинформации в поток графических файлов, передаваемых по сети. Очевидно, что в данном случае, во избежание потери качества передаваемой информации, пакеты данных (цифровые изображения) должны подготавливаться (заполняться сообщениями) и передаваться без задержек.

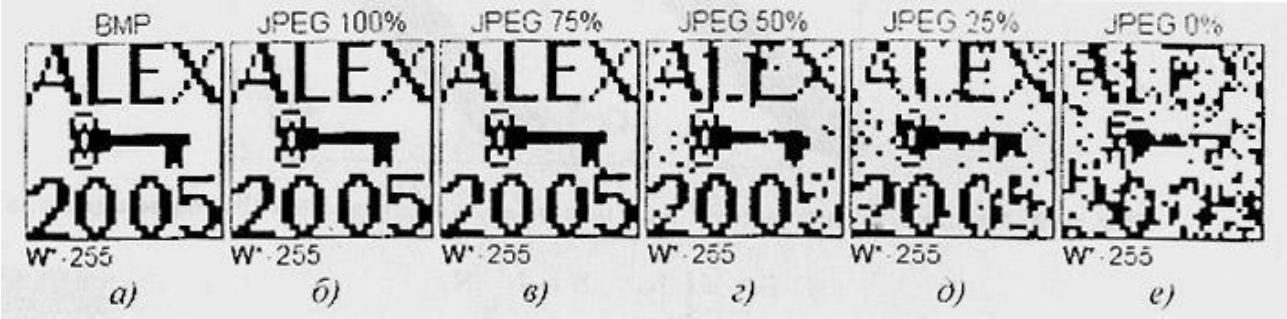

Рисунок 2.4 – Искажение ЦВЗ при сжатии: а) – исходный ЦВЗ; б) - е) – ЦВЗ, извлеченный контейнера, сжатого с различной степенью

Стоит отметить, что большинство стеганографических алгоритмов не обладают большой вычислительной сложностью. Тем не менее, попытки увеличения некоторых параметров эффективности (скрытность, размер сообщения), могут значительно увеличивать объемы вычислений и ограничивать использование алгоритма в системах реального времени.

Используемый графический формат

В значительной степени эффективность применения цифровых изображений в стеганографии зависит от формата их хранения.

В компьютерной стеганографии в качестве контейнера может выступать практически любой файловый формат, однако наиболее распространенным типом носителя являются файлы изображения формата BMP. Это объясняется тем, что для целей стеганографии наиболее предпочтительны файлы форматов, в которых используются методы сжатия без потерь (такие виды сжатия типичны для изображений формата BMP, TIFF, PNG, TGA, и др.). Также положительной стороной в пользу выбора формата BMP выступает высокое качество изображения и простота формата.

Стоит отметить, что при работе с форматами файлов, использующих сжатие с потерями, таким как JPEG, обычно все равно выполняют преобразование потока данных JPEG в поток данных BMP. С позиции стеганографии файлы данного формата позволяют скрывать сравнительно большие объемы информации [10].

В данной работе в качестве контейнера рассматривается 24-битовое растровое изображение в системе цветности RGB формата BMP. Каждая цветовая комбинация тона (пикселя) представляет собой комбинацию значений яркости трех составляющих цветов – красного (R), зеленого (G) и синего (B), которые занимают каждый по 1 байту (итого по 3 байта на точку). Таким образом, яркость каждой составляющей записывается 8 - битным числом и может изменяться в диапазоне от 0 до 255 (комбинация (0, 0, 0) соответствует черному цвету, комбинация (255, 255, 255) – белому).

Использование BMP-файлов обусловлено только лишь простотой их программной обработки, – все полученные результаты с легкостью могут быть перенесены на случай изображений в файлах других форматов.

Выводы

Во второй главе рассмотрены общие типы стеганографических контейнеров, классифицированных по критерию протяженности. Цифровые изображения относятся к типу контейнеров фиксированной длины. Ограниченный объем контейнеров фиксированной длины является существенным недостатком данного типа контейнеров, что, в свою очередь, делает еще более актуальным исследование возможности увеличения объема встраиваемой информации за счет использования старших бит изображений.

Сформулированы основные требования к выбору контейнера для стеганографического скрытия данных методом наименее значащих бит цифрового изображения, основанных на свойствах цифровых изображений. Представленные требования к выбору контейнера являются важными условиям, позволяющими лишить нарушителя заведомых преимуществ в

обнаружении факта сокрытия информации и необходимы для удовлетворения условий эффективности в стеганографии, использующей цифровые изображения в качестве контейнеров. Критерии эффективности, описанные в текущей главе, можно выделить в две условные группы: технические критерии и критерии, не поддающиеся техническому описанию. В качестве технического критерия оценки эффективности, можно привести пример отношения максимального размера встраиваемого сообщения, не приводящего к искажению изображения, к размеру самого контейнера. В свою очередь, используемых графический формат, не поддающийся строгому математическому описанию, является важным условием эффективности в стеганографии. Таким образом, обе группы являются равнозначными условиями оценки эффективности стеганографической системы.

Полученные результаты в дальнейшем могут быть использованы при исследовании возможности встраивания информации в битовые плоскости изображений.

ГЛАВА 3 СТАТИСТИЧЕСКИЕ АНАЛИТИЧЕСКИЕ МЕТОДЫ СТЕГАНОГРАФИЧЕСКОГО АНАЛИЗА ДЛЯ ОБНАРУЖЕНИЯ LSB СТЕГАНОГРАФИИ

Ранее рассмотренные теоретические оценки стойкости стеганосистем, например, теоретико-информационные, предполагают, что скрывающий информацию и нарушитель обладают неограниченными вычислительными ресурсами для построения стеганосистем и, соответственно, стеганоатак на них, придерживаются оптимальных стратегий скрывающего преобразования и стеганоанализа, располагают бесконечным временем для передачи и обнаружения скрываемых сообщений и т.д. Разумеется, такие идеальные модели скрывающего информацию и нарушителя неприменимы для реалий практических стеганосистем. Поэтому рассмотрим известные к настоящему времени практические оценки стойкости некоторых стеганосистем, реально используемых для скрытия информации.

В последние годы появились программно-реализованные стегосистемы, обеспечивающие скрытие информации в цифровых видео- и аудиофайлах. Такие программы свободно распространяются, легко устанавливаются на персональные компьютеры, сопрягаются с современными информационными технологиями и не требуют специальной подготовки при их использовании. Они обеспечивают встраивание текста в изображение, изображение в изображение, текста в аудио- сигнал и т.п. В современных телекоммуникационных сетях типа Интернет передаются очень большие потоки мультимедийных сообщений, которые потенциально могут быть использованы для скрытия информации. Одной из наиболее актуальных и сложных проблем цифровой стеганографии является выявление факта такого скрытия. В реальных условиях наиболее типичным видом атаки нарушителя является атака только со стего, так как истинный контейнер ему обычно неизвестен. В этих условиях обнаружение скрытого сообщения возможно на основе выявления нарушений зависимостей, присущих естественным контейнерам. Практический стеганоанализ цифровых стеганосистем является очень молодой наукой, однако в его арсенале уже имеется ряд методов, позволяющих с высокой вероятностью обнаруживать факт наличия стеганоканала, образованных некоторыми предложенными к настоящему времени стеганосистемами. Среди методов практического стеганоанализа наибольший интерес представляет класс статистических атак.

Нарушение статистических закономерностей естественных контейнеров является одним из наиболее перспективных подходов для выявления факта существования скрытого канала передачи информации является подход, представляющий введение в файл скрываемой информации. При данном подходе анализируются статистические характеристики исследуемой последовательности и устанавливается, похожи ли они на характеристики естественных контейнеров (если да, то скрытой передачи информации нет), или они похожи на характеристики стего (если да, то выявлен факт существования скрытого канала передачи информации). Этот класс стеганоатак является вероятностным, то есть они не дают однозначного ответа, а формируют оценки типа “данная исследуемая последовательность с вероятностью 90% содержит скрываемое сообщение”. Вероятностный характер статистических методов стеганоанализа не является существенным недостатком, так как на практике эти методы часто выдают оценки вероятности существования стеганоканала, отличающиеся от единицы или нуля на бесконечно малые величины.

3.1 Атака на основе анализа статистики Хи – квадрат

В методе используется анализ гистограммы, полученной по элементам изображения и оценка распределения пар значений этой гистограммы. Для BMP- файлов пары значений формируются значениями пикселов изображения, для JPEG-квантуемыми коэффициентами дискретного косинусного преобразования, которые отличаются по младшему биту. Младшие биты изображений не являются случайными. Частоты двух соседних элементов контейнера должны находиться достаточно далеко от значения частоты среднего арифметического этих элементов. В «пустом» изображении ситуация, когда частоты элементов со значениями 2N и 2N + 1 близки по значению, встречается достаточно редко. При встраивании информации данные частоты сближаются или становятся равными.

Идея атаки хи-квадрат заключается в поиске этих близких значений и подсчете вероятности встраивания на основе того, как близко располагаются значения частот четных и нечетных элементов анализируемого контейнера.

Особенностью алгоритма является последовательный анализ всего изображения и, соответственно, накапливание частот элементов.

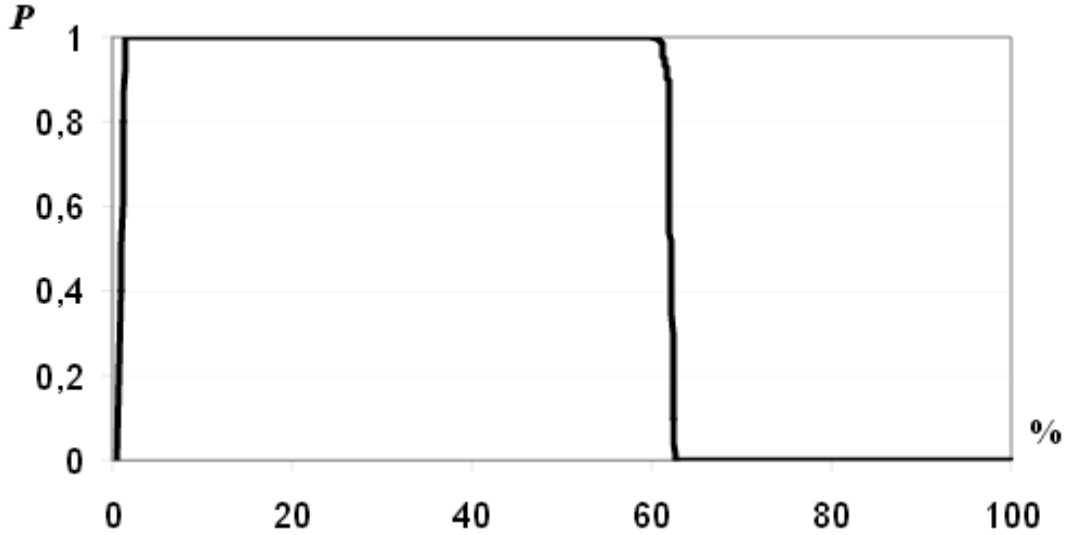

Рисунок 3.1 – Вероятность встраивания по критерию Хи – квадрат при анализе стегоконтейнера, полученного методом последовательной замены.

Метод Хи–квадрат является универсальным, так как подходит для анализа изображений, созданных различными программами скрытия. Однако результаты работы метода по критерию Хи–квадрат в значительной мере зависят от способа скрытия данных.

При последовательной записи в НЗБ элементов контейнера метод обеспечивает хорошие результаты (рисунок 3.1), а при псевдослучайном выборе младших бит и рассеивании сообщения по всей длине контейнера метод не срабатывает [11].

3.2 Разностный стеганоанализ на основе двойной статистики

Одним из оригинальных методов статистического стеганоанализа является метод RS, впервые опубликованным в 2001 г. коллективом ученых под руководством Дж.Фридрих. Сокращение в названии расшифровывается как Regular-Singular, то есть «регулярно-сингулярный».

Суть метода состоит в следующем. Все изображение разбивается на группы по *n* пикселов *G(x,x2,…,xn)*, где n четно, например по 2 пиксела, находящихся рядом по горизонтали. Для группы пикселов определяется функция регулярности или «гладкости» *f(G)*, в качестве такой функции можно выбрать, например, дисперсию значений внутри группы, либо просто сумму перепадов значений смежных пикселов. Под значением пиксела понимаем целое число от 0 до 255:

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{n-1} |x_{i+1} - x_i|. \quad (5)$$

Функция F(x) называется флиппингом и имеет свойство $F(F(x)) = x$. Определим две функции флиппинга – F_1, соответствует инверсии младшего бита пиксела, и F_{-1}, представляющая собой инверсию с переносом в старший бит (прибавление единицы):

$$F_1: \ 0 \leftrightarrow 1,\ 2 \leftrightarrow 3,\ \ldots,\ 254 \leftrightarrow 255, \tag{6}$$

$$F_{-1}: \ 255 \leftrightarrow 0,\ 1 \leftrightarrow 2,\ 3 \leftrightarrow 4,\ \ldots, 253 \leftrightarrow 254,\ 255 \leftrightarrow 0. \tag{7}$$

При применении флиппинга к группе получаем преобразованную группу пикселов. Далее, поделим все группы пикселов на классы следующим образом:

Регулярные группы: $G \in R \Leftrightarrow f(F(G)) > f(G)$; (8)

Сингулярные группы: $G \in S \Leftrightarrow f(F(G)) < f(G)$; (9)

Неиспользуемые группы: $G \in U \Leftrightarrow f(F(G)) = f(G)$. (10)

В дальнейшем нас будет интересовать соотношение между группами в изображении. Определим количество групп попавших в тот или иной класс как R_M, S_M, U_M и R_{-M} , S_{-M}, U_{-M}, где индексы M и –M означают соответственно применение F_1 и F_{-1} для получения распределения. Наша цель – определить каким образом внедрение сообщения методом LSB будет влиять на вышеописанную статистику групп пикселов.

Метод основывается на статистическом предположении, что для естественного изображения, другими словами, незаполненного контейнера, характерно следующее:

$$R_M \cong R_{-M} \text{ и } S_M \cong S_{-M}. \tag{11}$$

Предположение основано на том, что применение F_{-1} даст то же распределение, что и F_1 на изображении, значения пикселов которого сдвинуты на единицу. Для обыкновенного изображения соотношение между группами не должно существенно меняться. Значительное расхождение между значениями свидетельствует о применении LSB-стеганографии для младших бит изображения.

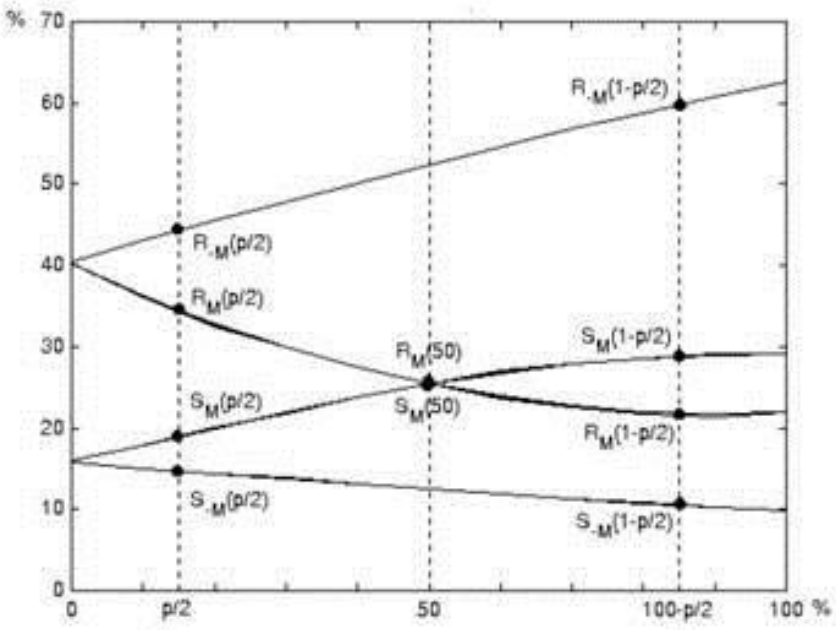

Рисунок 3.2 – RS-диаграмма типичного изображения

Рассмотрим изменения младших бит изображения при 100% перезаписи их битами сообщения. Внедрение случайного сообщения длиной, равной размеру изображения, приведёт к тому, что 50% младших бит будут инвертированы. Это, в свою очередь сведет к нулю разность между значениями R_M и S_M. Однако на R_{-M} и S_{-M} внедрение сообщения будет влиять прямо противоположно, и разность этих величин будет пропорциональна степени заполненности контейнера, иными словами длине сообщения. На рисунке 3.2 приведена RS-диаграмма для типичного изображения. На оси абсцисс расположено количество инвертированных бит x, искомая длина сообщения p, на оси ординат – относительные значения регулярных и сингулярных групп по отношению к общему числу групп изображения [7,12].

Выводы

В третьей главе приведены наиболее точные и широко распространённые методы стеганографического анализа факта сокрытия данных в пространственных областях цифровых изображений, основанные на исследовании статистических свойств контейнеров. Приведенные методы анализа относятся к категории методов на основе известной математической модели цифровых изображений.

Метода анализа на основе Хи–квадрат является универсальным методом стеганографического анализа и основан на сравнении частот соседних элементов изображения. Данный метод показывает хорошие результаты при

использовании последовательного встраивания информации в элементы контейнера. Однако существенным недостатком данного метода является то, что при псевдослучайном встраивании данных метод не может быть применен.

Метод RS является достаточно новым методом стеганоанализа, основанном на анализе соотношения между группами в цифровом изображении. Данный метод позволяет избежать недостатки, присущие методу анализа на основе Хи–квадрат, так как он не зависит от метода встраивания информации в пространственные области изображений.

В дальнейшем данные методы могут быть использованы в ходе научной работы для исследования возможности встраивания информации в битовые области изображений в качестве критерия стеганографической стойкости системы.

ГЛАВА 4 ОЦЕНКА СТЕГАНОГРАФИЧЕСКОЙ ЕМКОСТИ БИТОВЫХ ПЛОСКОСТЕЙ СТЕГАНОКОНТЕЙНЕРОВ

Одной из немаловажных задач стеганографии является выбор подходящего контейнера. Несмотря на большое количество исследований в данной области, выбор контейнера для стеганографического скрытия данных все еще освещен в недостаточной мере.

Целью данной научной работы является исследование статистических свойств естественных изображений-контейнеров, позволяющий обеспечить наилучшую стойкость стеганографической системы.

В качестве исследуемого формата цифровых изображений был выбран формат BMP. Выбор данного формата цифровых изображений был обусловлен тем, что он обеспечивает возможность встраивания большего количества данных и не использует алгоритмов сжатия изображения. Последнее является важным фактом, т.к. встраивание информации в цифровое изображение происходит в наименее значащий бит, что при использовании изображений с различными алгоритмами сжатия может привести к потере или повреждению встроенной информации.

В ходе работы было проанализировано более 100 различных изображений формата BMP. Довольно часто при выборе подходящего контейнера для стеганографии, изображения подразделяют на различные группы. Однако эти разделения являются достаточно условными, поэтому в ходе данной работы изображения не подвергались на разделения по группам. Научный интерес представляют лишь статистические свойства исследуемых изображений. Ко всем исследуемым изображениям предъявлялись следующие требования:

1. изображения должны являться исходным файлом, а не быть полученными путем конвертации других цифровых форматов изображений в формат BMP;
2. изображения не должны быть созданы с использованием каких-либо графических редакторов;
3. все изображения должны иметь одинаковый размер, что исключит влияние размера изображения на полученные результаты.

Встраивание информации в исследуемые изображения происходило по средствам метода LSB, которым является довольно популярным методом встраивания в стеганографии и подразумевает использование наименее значащие биты изображения. Одним из серьезных заблуждений в стеганографии является то, что младшие биты изображений являются ничем иным, как шумом. Однако это вовсе не так, между младшими битами

изображений устанавливаются вполне определенные зависимости, которые изменяются при стеганографическом скрытии информации.

Для определения этих изменений, в работе используется метод исследования статистики распределения Хи–квадрат и RS-метод стеганографического анализа. Таким образом, изображение с наименьшим отклонением данных оценок между естественным и стегаконтейнером будет наилучшим с точки зрения стеганографического скрытия информации.

В ходе исследования были изучены зависимости оценок Хи–квадрат и RS от статистических свойств естественных изображений-контейнеров, таких как монотонность, энтропия и дисперсия изображения.

4.1 Исследование статистических свойств изображения при встраивании информации в младшую битовую плоскость

Внедрение информации будет осуществляться в младшую битовую плоскость цифрового изображения по средствам LSB метода. Рассмотрим зависимость изменения статистики Хи – квадрат от монотонности изображения. За монотонность будем принимать величину, характеризующую относительное процентное соотношение цветовых оттенков изображения.

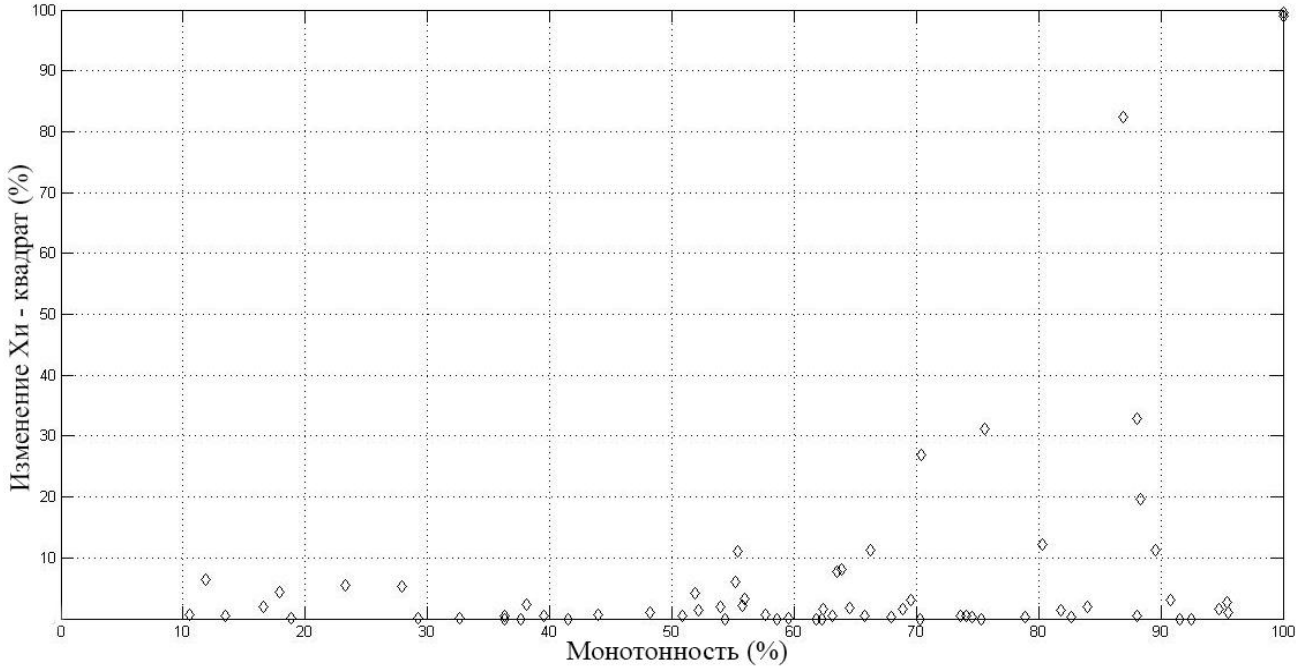

Рисунок 4.1 – Зависимость изменения статистики Хи–квадрат от монотонности изображения

На основе графика (рисунок 4.1) можно сделать вывод, что наиболее подходящими изображениями в качестве стегоконтейнера являются изображения, обладающие наименьшей монотонностью. Максимальное

изменение статистики Хи–квадрат (100%) находится в точке, где монотонность изображения достигает 100%. Таким образом, анализируя такую характеристику как “монотонность” изображения, изображения содержащие большее количество областей монотонной заливки, являются менее предпочтительными в качестве стегоконтейнера и обеспечивают плохую скрытность встроенного сообщения.

Однако, одной характеристики изображения недостаточно, чтобы определить наиболее подходящие при выборе изображений в качестве контейнера. Немаловажными статистическим характеристиками цифрового изображения являются энтропия и дисперсия.

Зависимость изменения Хи–квадрат от энтропии младшей битовой плоскости изображения представлены на рисунке 4.2.

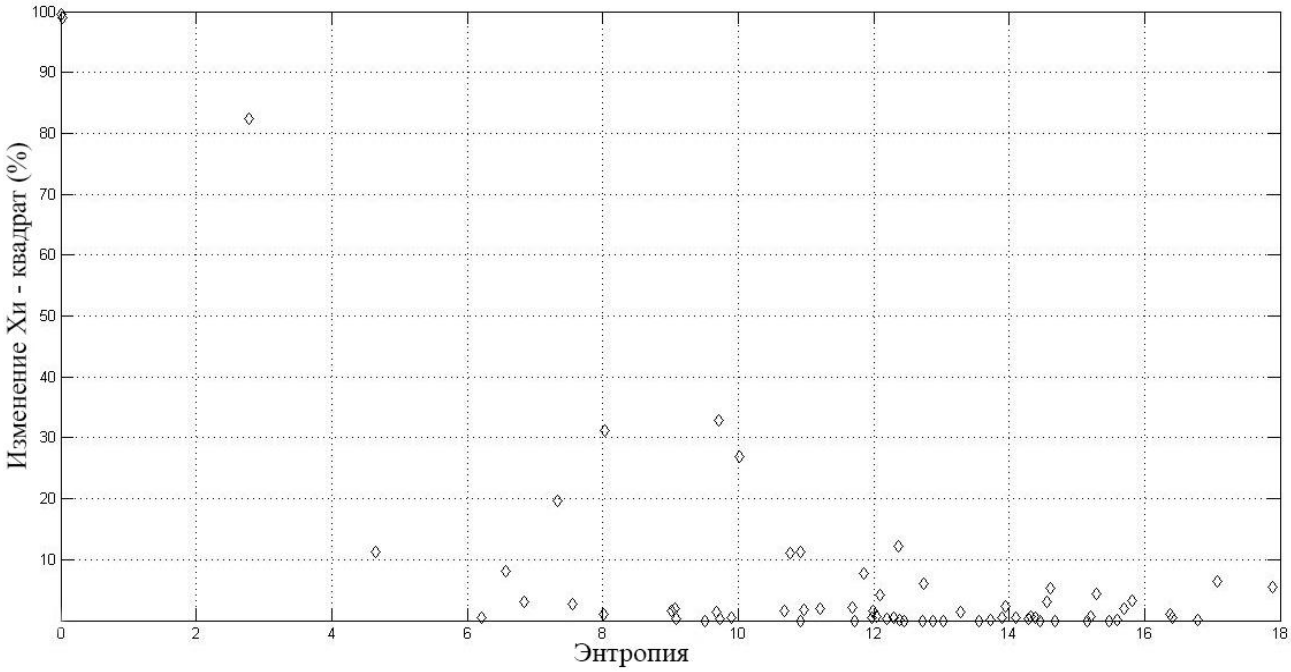

Рисунок 4.2 – Зависимость изменения Хи–квадрат от энтропии младшей битовой плоскости

На графике заметно, что изменение Хи–квадрат уменьшается с увеличением энтропии исследуемых изображений. Исходя из этого можно сделать вывод, что критерий Хи–квадрат естественного изображения и изображения, содержащего стего, ближе у тех изображений, энтропия которых выше. Так как энтропия цифрового изображения характеризует величины яркостных вариаций изображения, то система, использующая в качестве контейнера пестрые изображения с большим количеством мелких деталей, обеспечивает большую надежность.

Для определения разброса значений младших бит используется математическая величина – дисперсия. Недостатком метода определения дисперсии является чувствительность к размеру изображения. Соседние пиксели двух одинаковых изображений с разным разрешением отличаются количеством пикселей, приходящихся на одинаковые фрагменты изображения. Чем меньше размер изображения, тем больше будут отличаться два соседних пикселя, чем у такого же изображения большего размера. Именно поэтому в данной работе было принято использовать изображения одинакового размера. На рисунке 4.3 представлена зависимость изменения Хи-квадрат и дисперсии изображения.

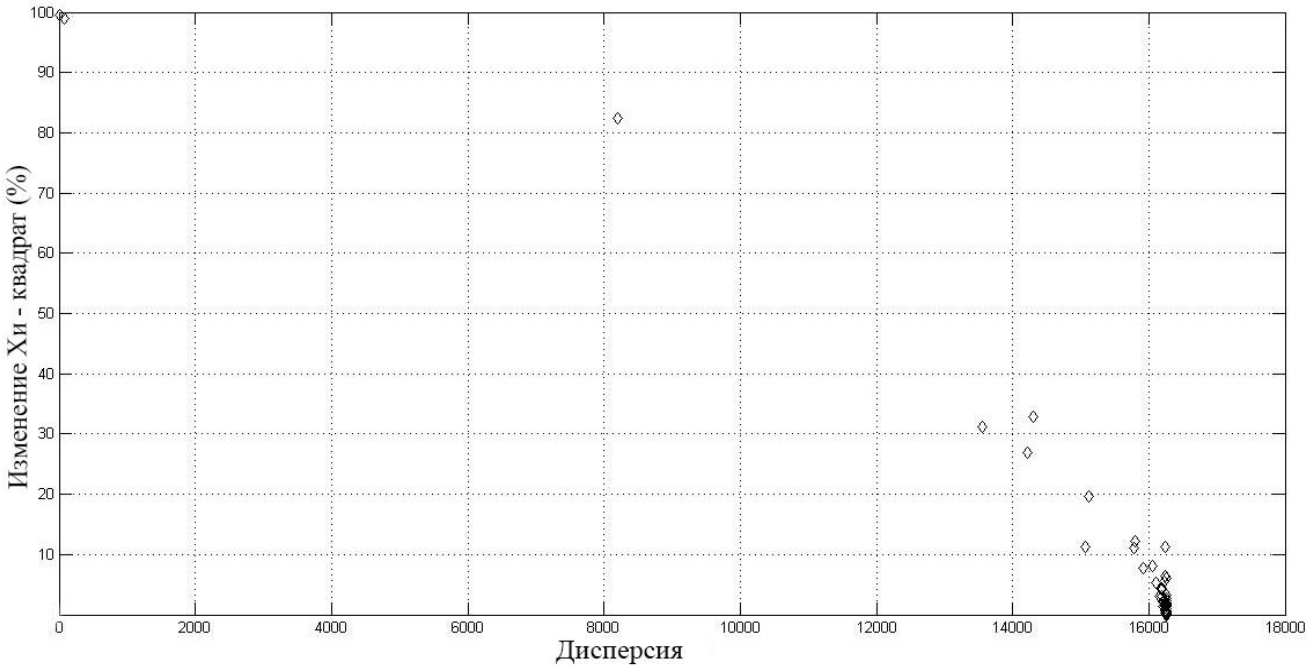

Рисунок 4.3 – Зависимость изменения Хи–квадрат от дисперсии младшей битовой плоскости

Результаты анализа зависимости изменения (рисунок 4.3) Хи–квадрат от дисперсии изображения показывают, что с увеличением дисперсии изображения, разница Хи–квадрат естественного контейнера и контейнера, содержащего стего, уменьшается. Таким образом, изображения, обладающие большей дисперсией в младшей битовой плоскости, являются наиболее подходящими в качестве контейнера для встраивания секретной информации.

Подводя итоги стеганографической атаки на основе критерия Хи-квадрат, можно заметить, что наиболее подходящими в качестве стегоконтейнера являются изображения, обладающие меньшей монотонностью и большими энтропией и дисперсией.

Рассмотрим зависимости RS – стеганоанализа от статистических свойств изображений, представленных выше, т.е. монотонности, энтропии и дисперсии. Как уже было подмечено ранее RS - стеганоанализ является довольно эффективным методом для обнаружения стеганографического скрытия информации в цифровых изображениях.

На рисунке 4.4 и рисунке 4.5 представлены зависимости изменений регулярных и сингулярных групп от монотонности изображений.

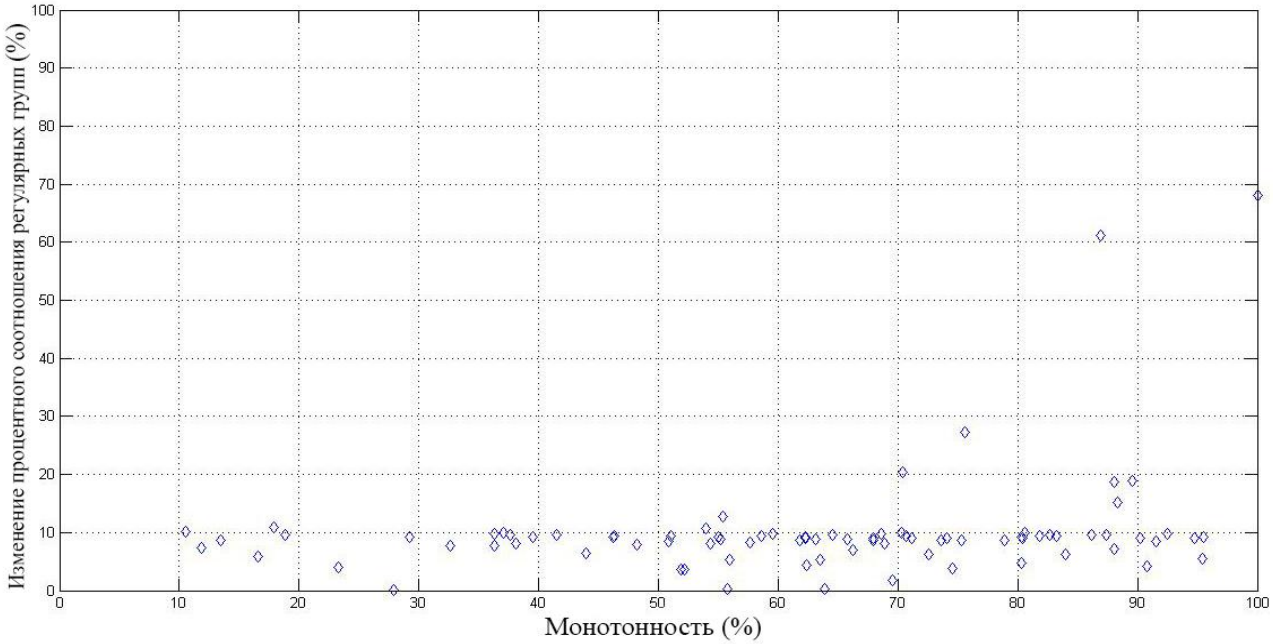

Рисунок 4.4 – Зависимость изменения процентного соотношения регулярных групп от монотонности

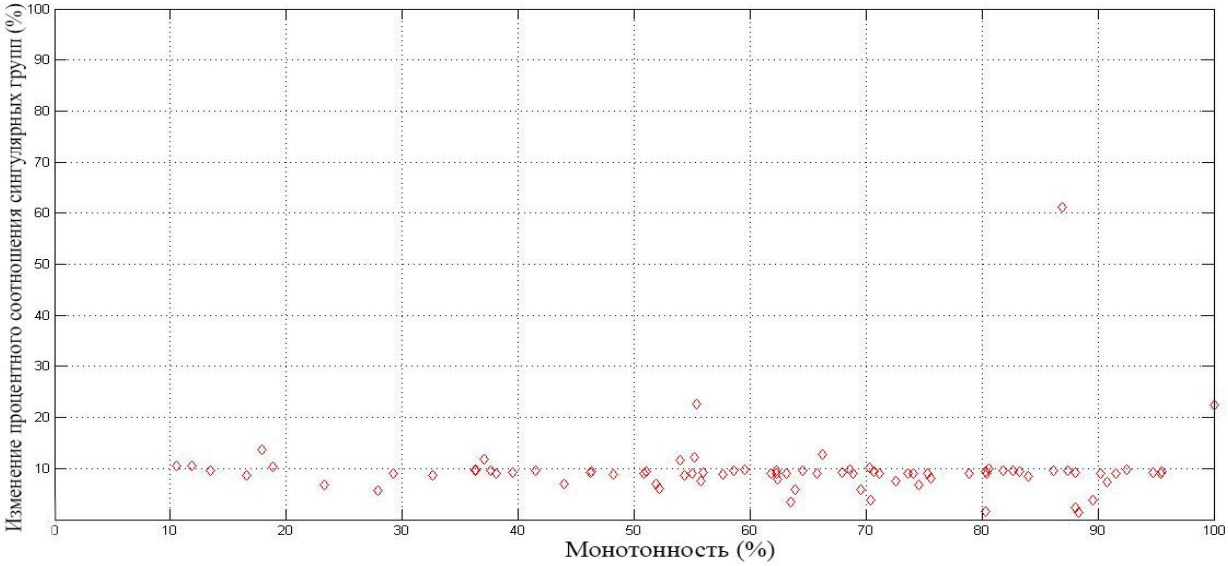

Рисунок 4.5 – Зависимость изменения процентного соотношения сингулярных групп от монотонности

Анализ зависимостей на рисунке 4.4 и рисунке 4.5 показывает, что RS-метод является менее чувствительным к монотонности изображения, поэтому данная характеристика изображения не является объективной в случае применения RS-метода при выборе стегоконтейнера.

Обратимся к энтропии изображения и проанализируем поведение метода в данном случае. Анализ полученных результатов показывает, что, как и в случае с методом анализа, основанного на критерии Хи–квадрат, наиболее подходящими изображениями в качестве стегоконтейнера, являются изображения, обладающие большей энтропией. Изменение регулярных и сингулярных групп естественных и стего- контейнеров меньше у изображений с большей энтропией в младшей битовой плоскости рисунках 4.6 и 4.7.

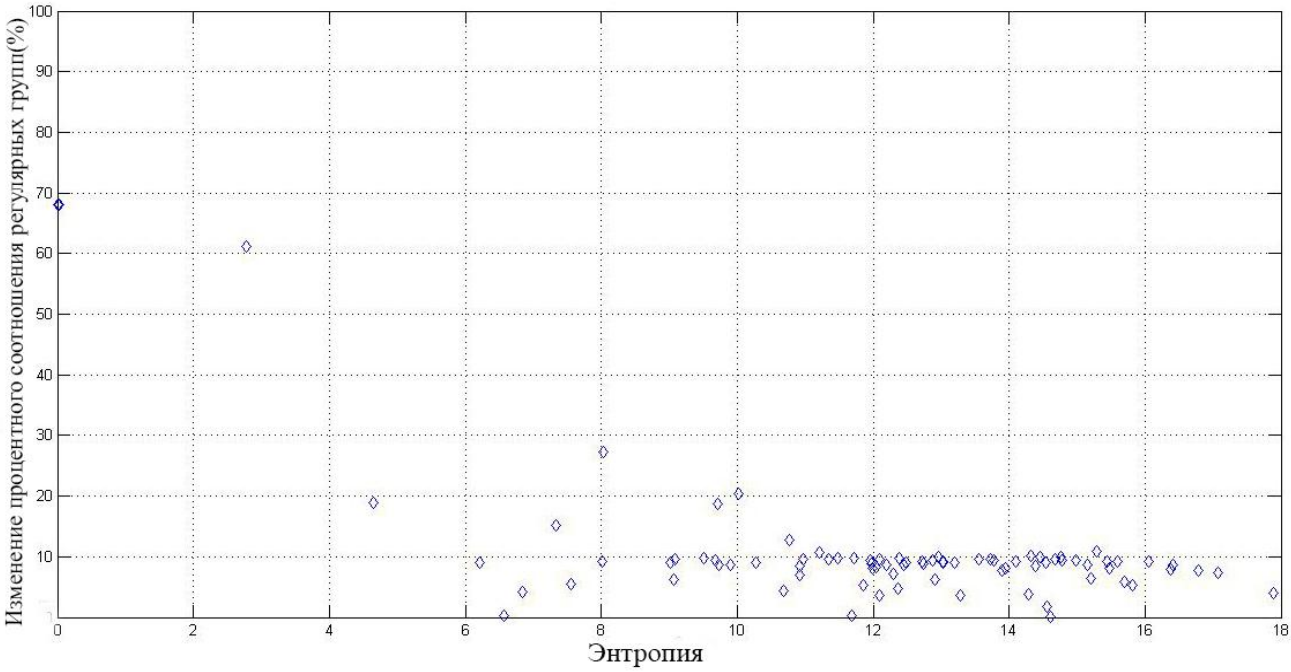

Рисунок 4.6 – Зависимость изменения процентного соотношения регулярных групп от энтропии младшей битовой плоскости

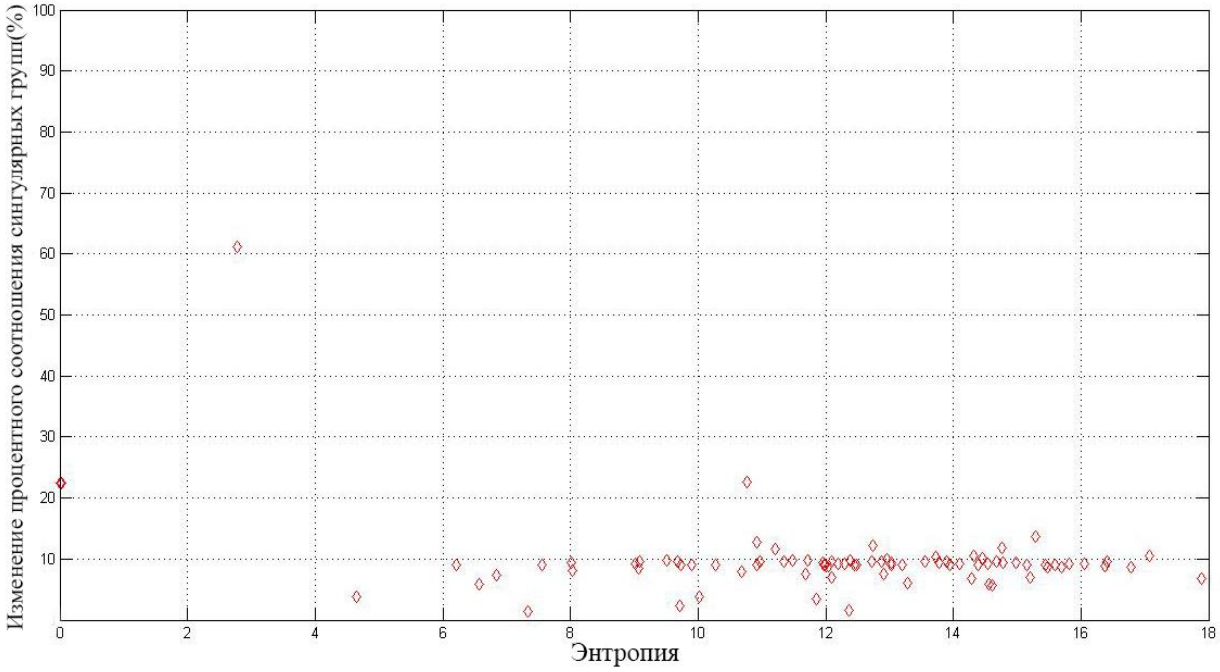

Рисунок 4.7 – Зависимость изменения процентного соотношения сингулярных групп от энтропии младшей битовой плоскости

Анализ зависимости RS – стегоанализа от дисперсии изображения (рисунок 4.8 и рисунок 4.9) также показал, что, как и в методе анализа на основе критерия Хи –квадрат, изображения с большей дисперсией наиболее подходящие в качестве контейнера для метода LSB.

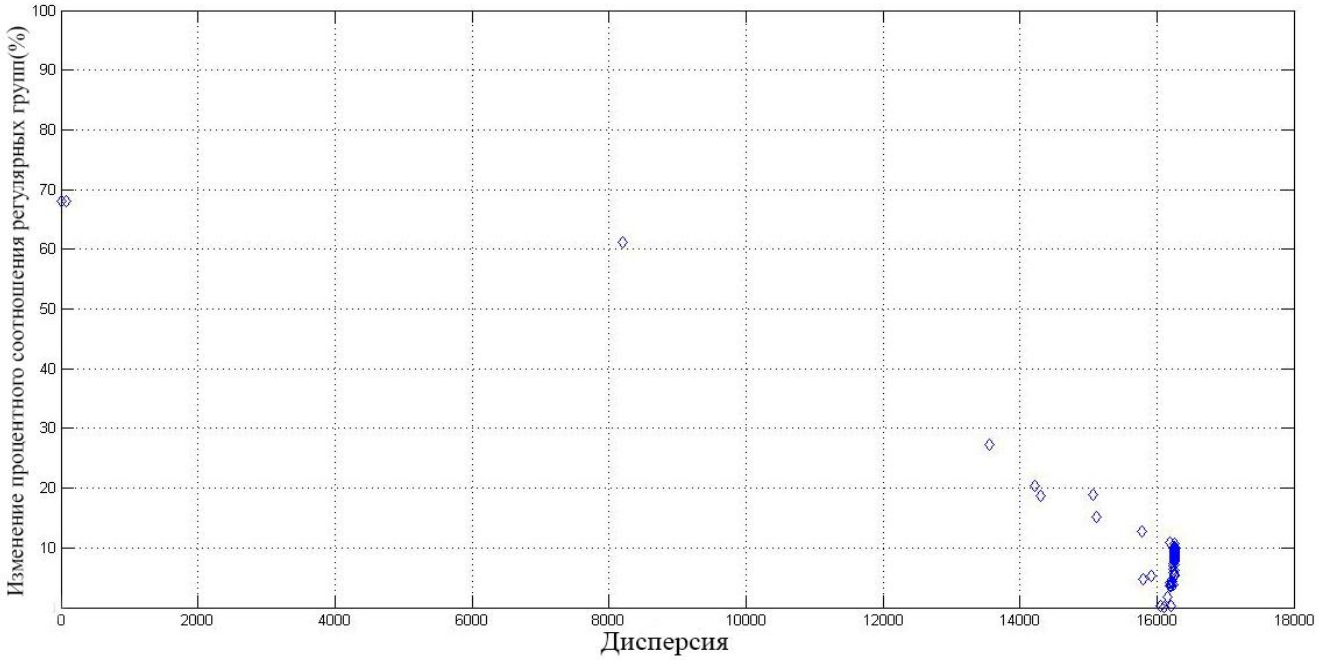

Рисунок 4.8 – Зависимость изменения процентного соотношения регулярных групп от дисперсии младшей битовой плоскости

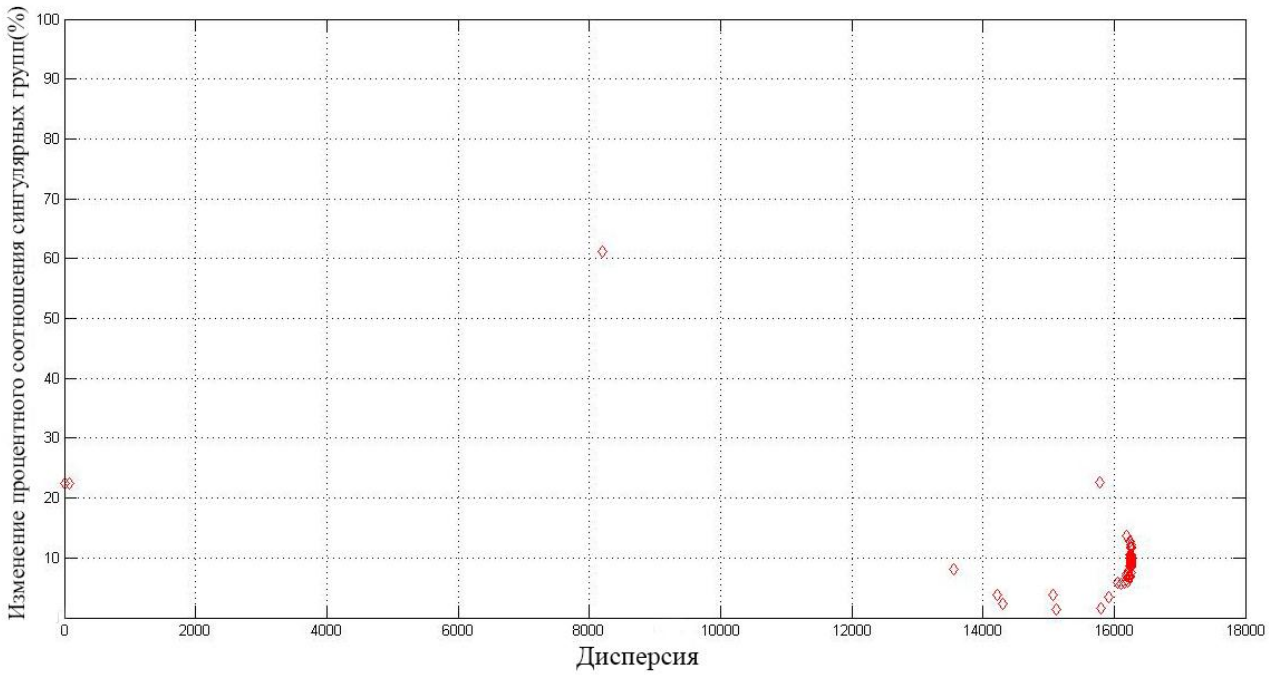

Рисунок 4.9 – Зависимость изменения процентного соотношения сингулярных групп от дисперсии младшей битовой плоскости

4.2 Исследование статистических свойств цифровых изображения при встраивании информации во вторую битовую плоскость

Одним из существенных недостатков метода LSB является небольшое количество информации возможное встроить в случае использования исключительно наименее значащих бит цифрового изображения. Таким образом, встает необходимость изучения возможности задействования наиболее значащих бит изображения для встраивания информации. Рассмотрим результаты встраивания сообщения во вторые биты изображения. На рисунке 4.10 показана зависимость изменения критерия Хи–квадрат от дисперсии изображения.

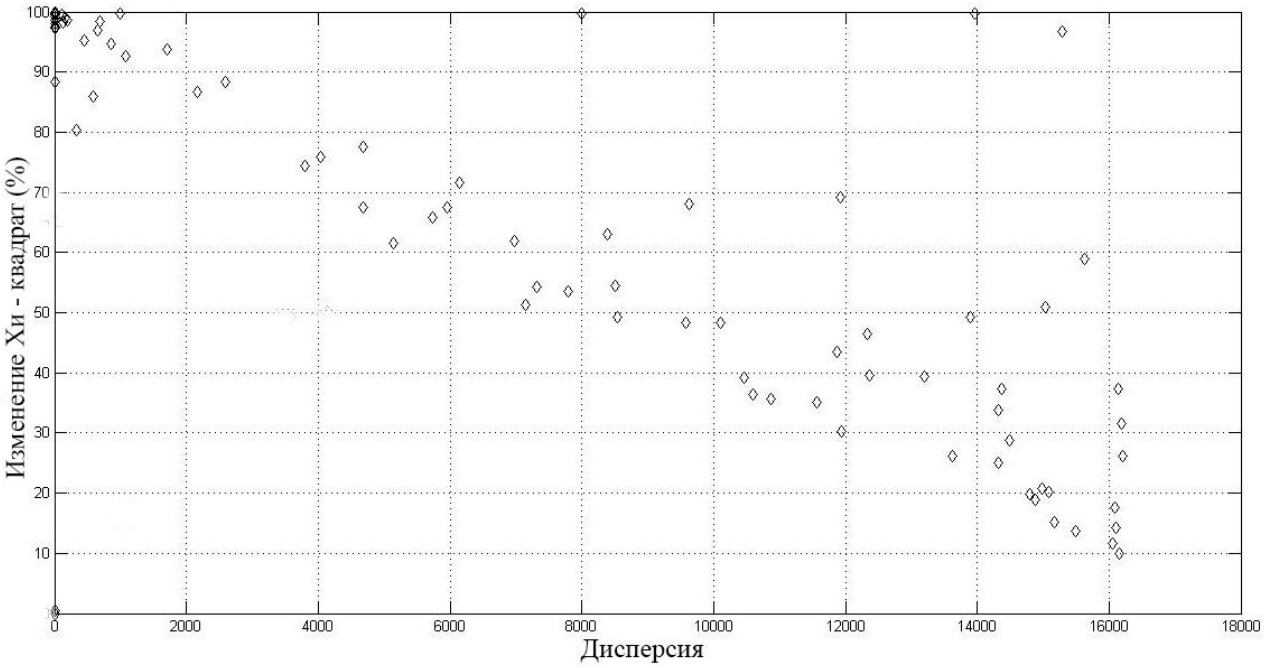

Рисунок 4.10 – Зависимость изменения Хи–квадрат от дисперсии второй битовой плоскости

Исходя из полученных данных, можно сделать вывод, что с увеличением дисперсии второй битовой плоскости изменение показателя Хи–квадрат уменьшается. Однако, анализируя значения дисперсии второй битовой плоскости, можно заметить, что разброс значений дисперсии увеличивается по сравнению с младшей битовой плоскостью. На рисунках 4.11 и 4.12 показаны результаты работы метода регулярных – сингулярных групп. Результаты являются схожими с результатами работы метода Хи–квадрат, и показывают, что наиболее подходящими изображениями в качестве контейнера являются изображения, обладающие наибольшим значением дисперсии второй битовой плоскости изображения.

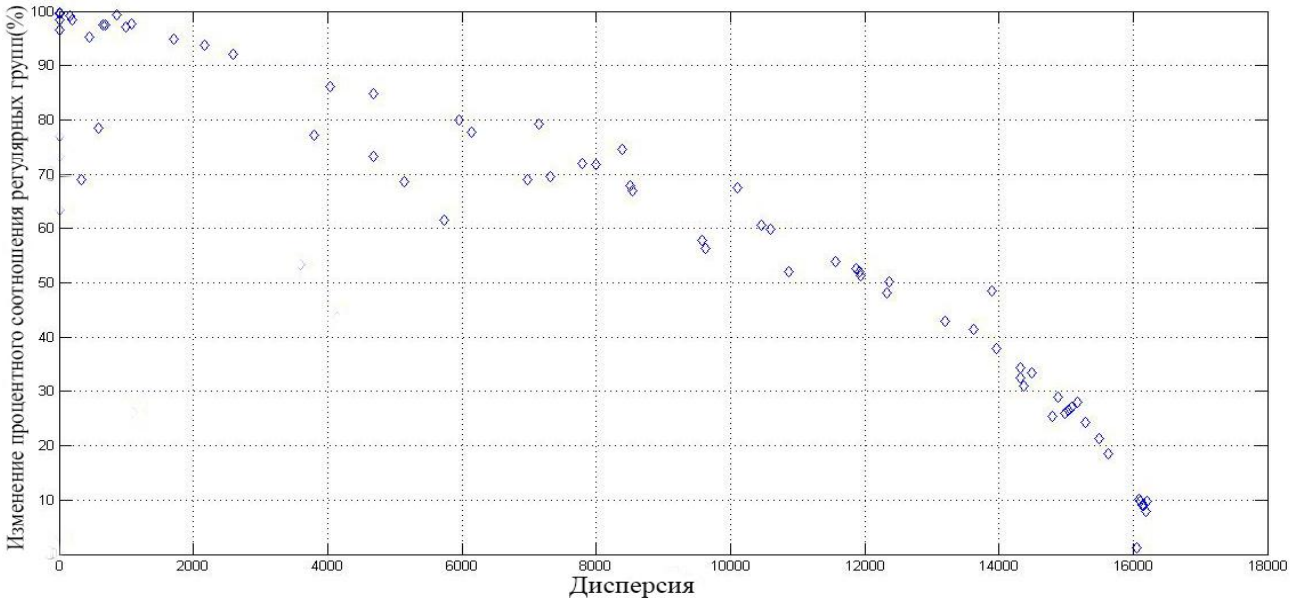

Рисунок 4.11 – Зависимость изменения процентного соотношения регулярных групп от дисперсии второй битовой плоскости

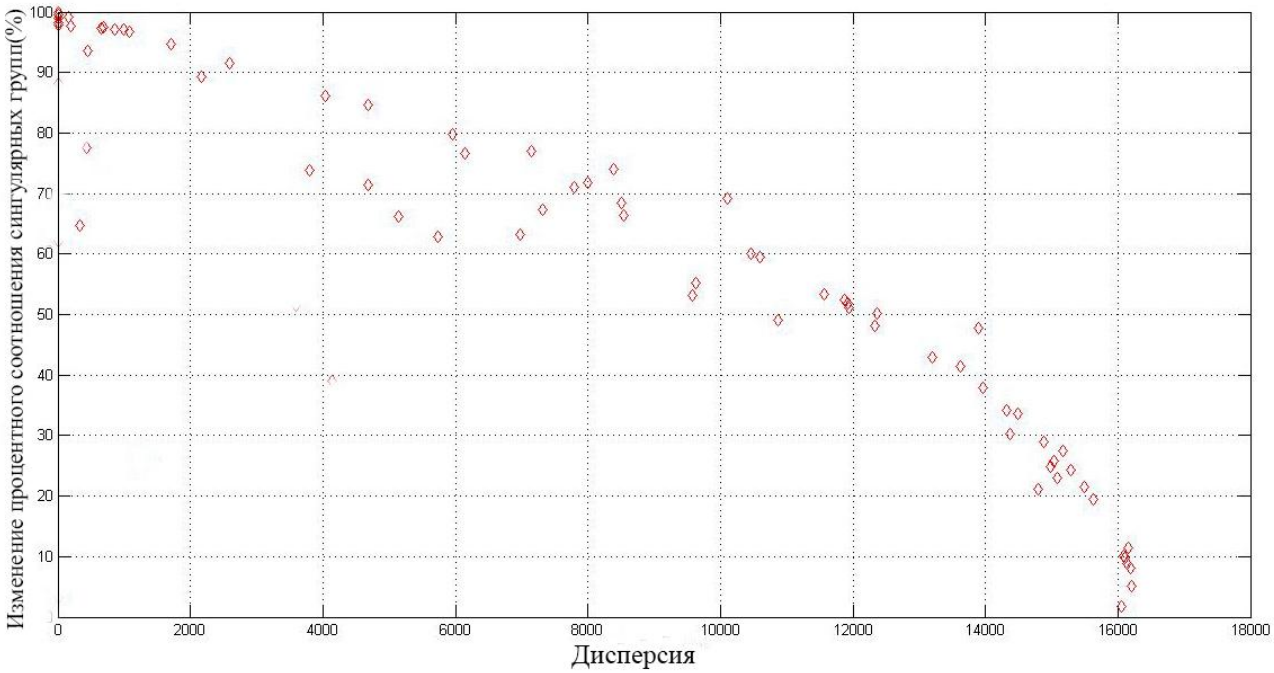

Рисунок 4.12 – Зависимость изменения процентного соотношения сингулярных групп от дисперсии второй битовой плоскости

Рассмотрим результаты работы метода Хи – квадрат и RS метода касательно энтропии изображения. На рисунке 4.13 представлена зависимость изменения Хи – квадрат от энтропии.

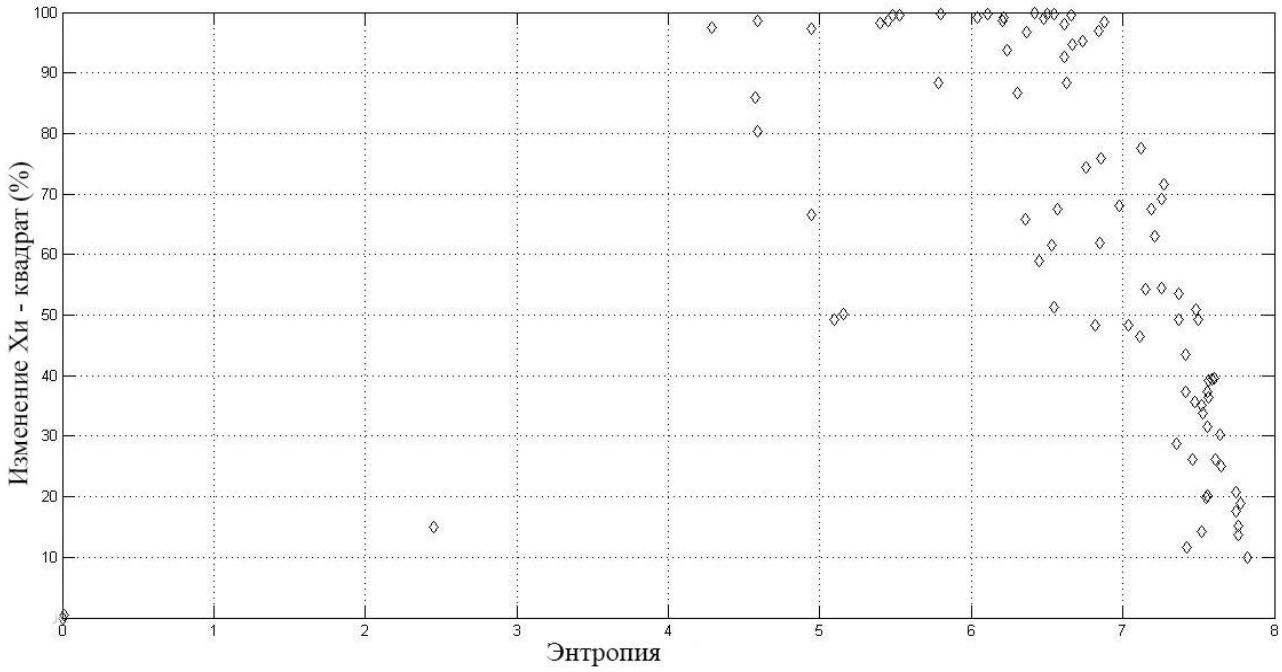

Рисунок 4.13 – Зависимость изменения критерия Хи – квадрат от энтропии второй битовой плоскости

Так же, как и в случае встраивания в наименее значащие биты изображения, при встраивании во вторую битовую плоскость наибольшей стойкостью к статистическому методу анализа Хи – квадрат обладают изображения с большим значением энтропии во второй битовой плоскости. Как можно заметить, численное значение энтропии второй битовой плоскости уменьшается относительно младшей битовой плоскости, что делает вторые биты менее привлекательными для встраивания по сравнению с младшими битами.

RS метода стеганоанализа относительно энтропии цифрового изображения показывает схожие результаты: наибольшей стеганографической стойкостью обладают изображения с большим значением энтропии второй битовой плоскости (рисунок 4.14 и рисунок 4.15).

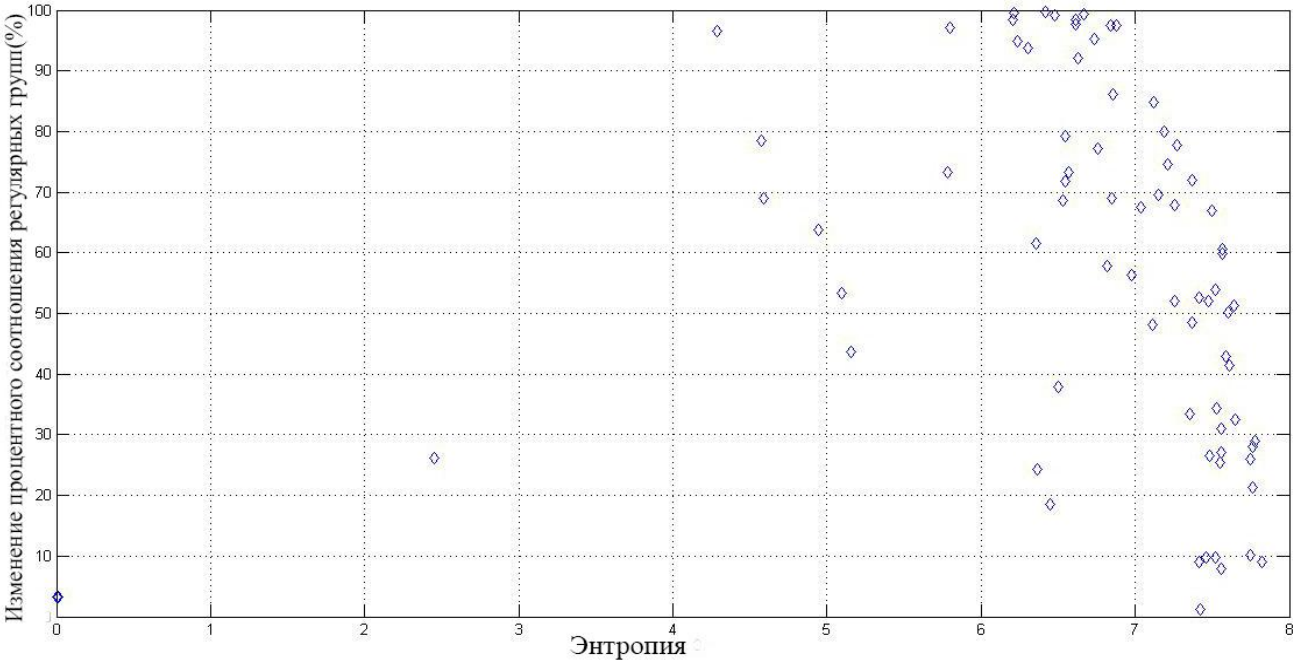

Рисунок 4.14 – Зависимость изменения процентного соотношения регулярных групп от энтропии второй битовой плоскости

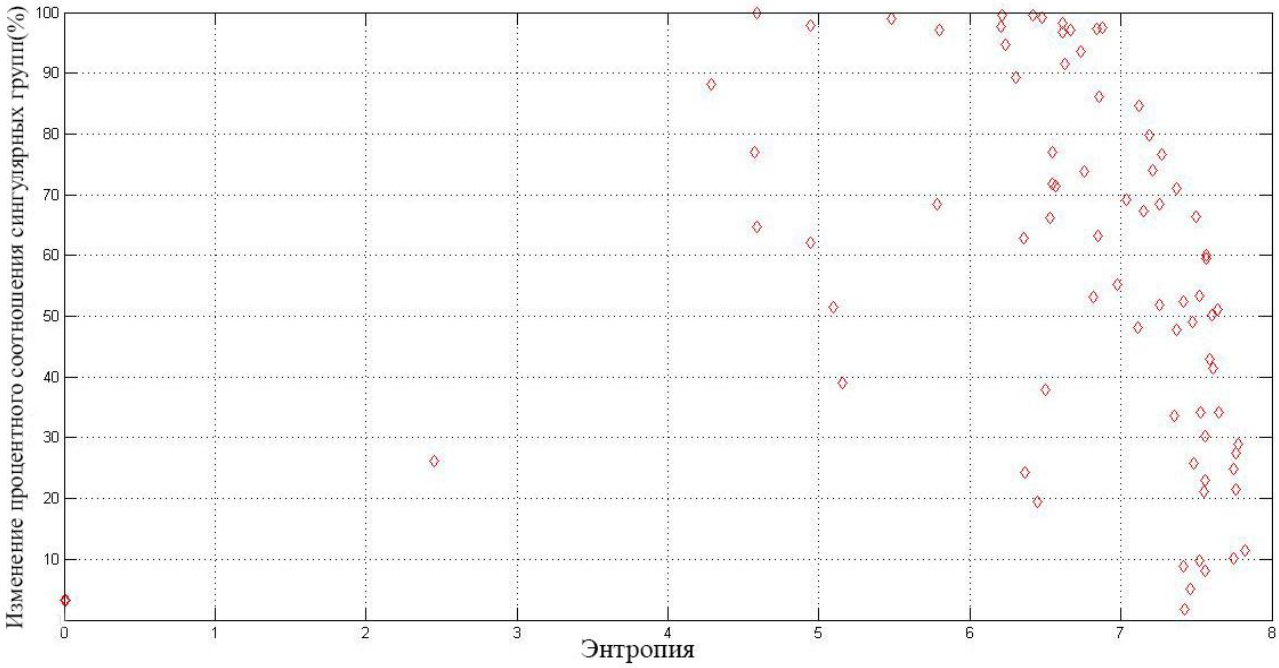

Рисунок 4.15 – Зависимость изменения сингулярной группы от энтропии второй битовой плоскости

Таким образом, в случае необходимости задействования вторых бит цифрового изображения для встраивания стеганографического сообщения, необходимо отдавать предпочтения изображениям, обладающим большей дисперсией и энтропией во второй битовой плоскости. Однако использование вторых бит изображения повышает шанс обнаружения факта внедрения сообщения по средствам анализа битовых срезов изображения.

4.3 Исследование статистических свойств цифровых изображения при встраивании информации в третью битовую плоскость

Проанализируем возможность использования третье битовой плоскости цифровых изображений для стеганографического скрытия данных.

На рисунке 4.16 представлена зависимость изменения критерия Хи – квадрат от дисперсии.

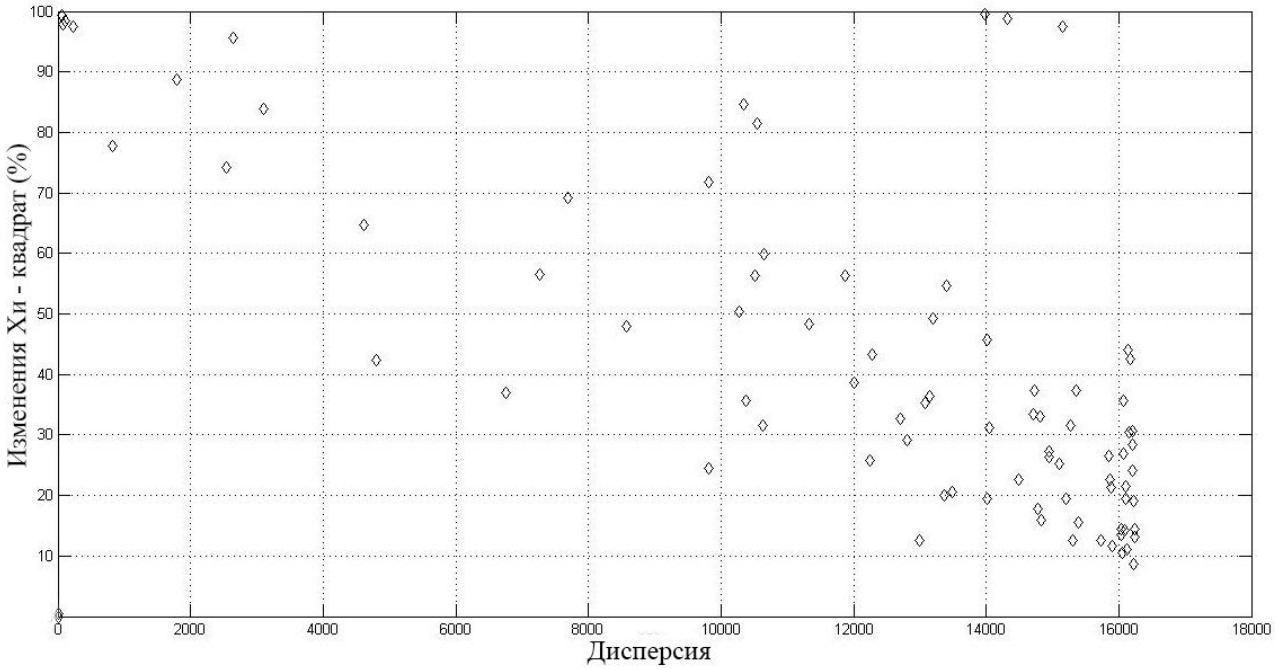

Рисунок 4.16 – Зависимость изменения Хи – квадрат от дисперсии третьей битовой плоскости

Из зависимости на рисунке 4.16 можно увидеть, что, как и в случае с более младшими битовыми плоскостями, наибольшей стеганографической устойчивостью обладают изображения с большим значением дисперсии.

Рассматривая поведение метода RS, можно прийти к аналогичному выводу (рисунок 4.17 и рисунок 4.18).

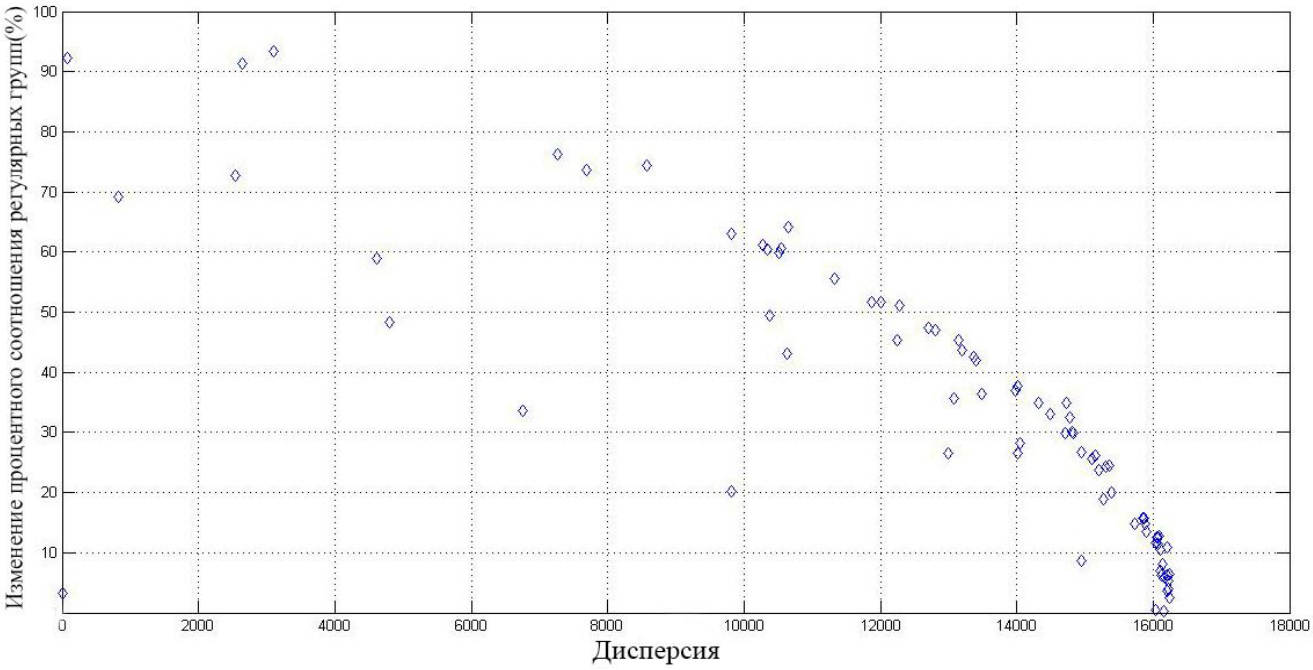

Рисунок 4.17 – Зависимость изменения процентного соотношения регулярных групп от дисперсии третьей битовой плоскости

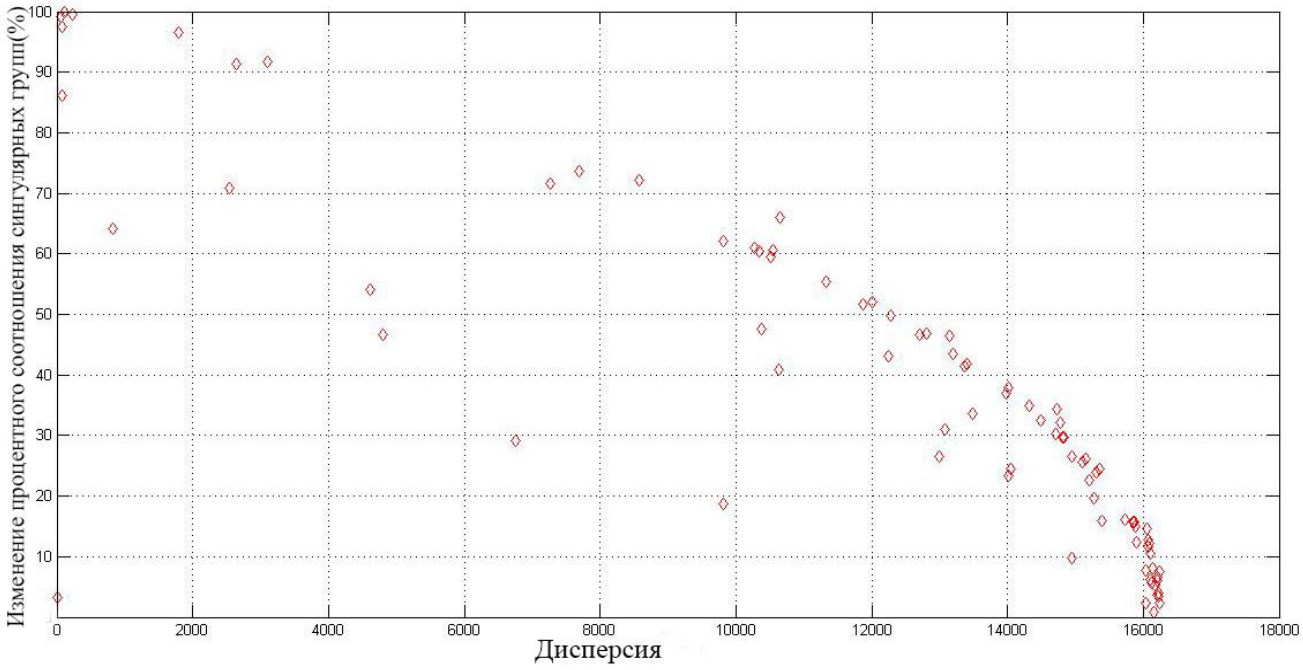

Рисунок 4.18 – Зависимость изменения процентного соотношения сингулярных групп от дисперсии третьей битовой плоскости

Рассмотрим далее зависимость критерия Хи – квадрат и регулярных, сингулярных групп от энтропии. На рисунке 4.19 представлена зависимость Хи – квадрат от энтропии третьей битовой плоскости изображения. Как и в предыдущих случаях внедрения информации в битовые плоскости изображений, наибольшей стойкостью, как стеганографической системы, обладает цифровое изображение, энтропия которой выше.

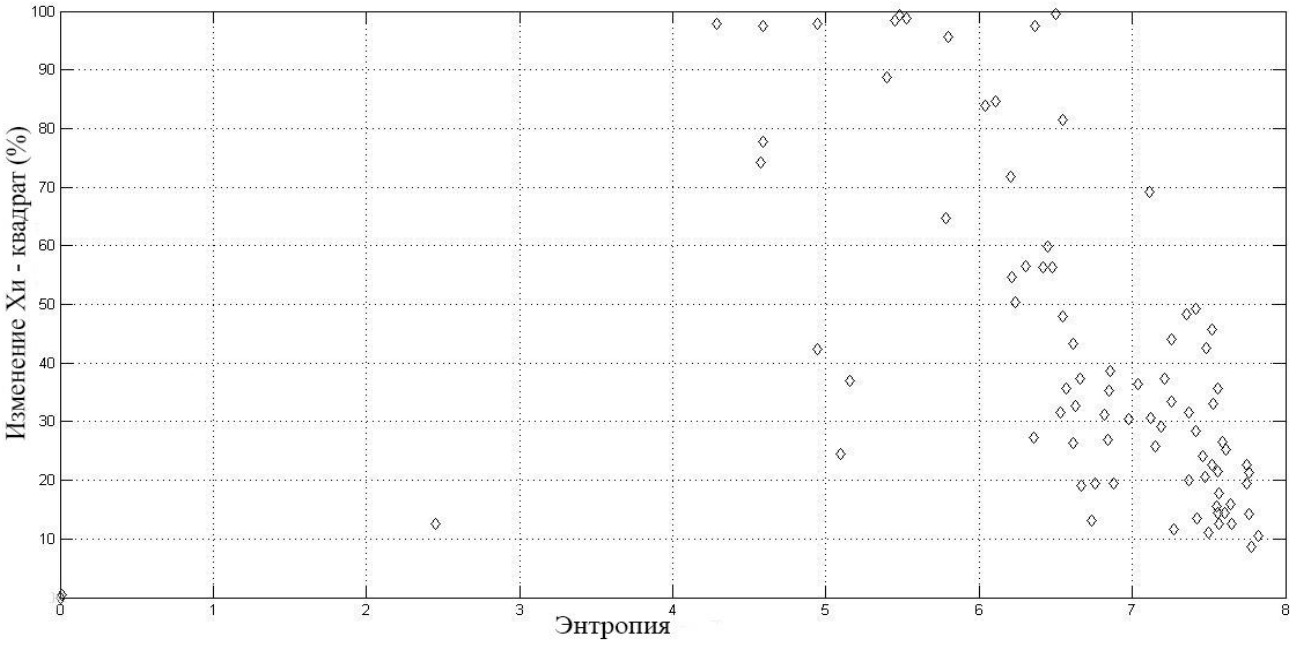

Рисунок 4.19 – Зависимость изменения критерия Хи – квадрат от энтропии третьей битовой плоскости

Анализ результатов работы RS метода (рисунок 4.20 и 4.21), также показывают, что для наибольшей надежности цифрового изображения в качестве контейнера, для внедрения информации в третью битовую плоскость необходимо использовать изображения, обладающие наибольшей энтропией в третьих битах.

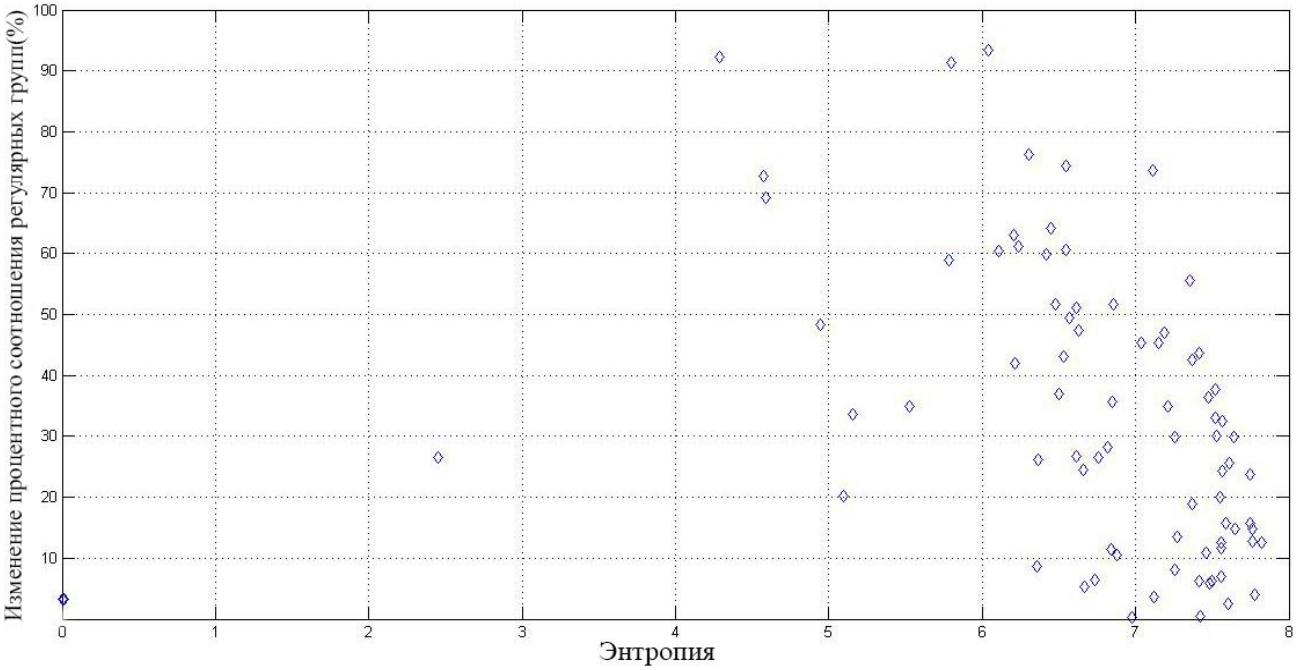

Рисунок 4.20 – Зависимость изменения процентного соотношения регулярных групп от энтропии третьей битовой плоскости

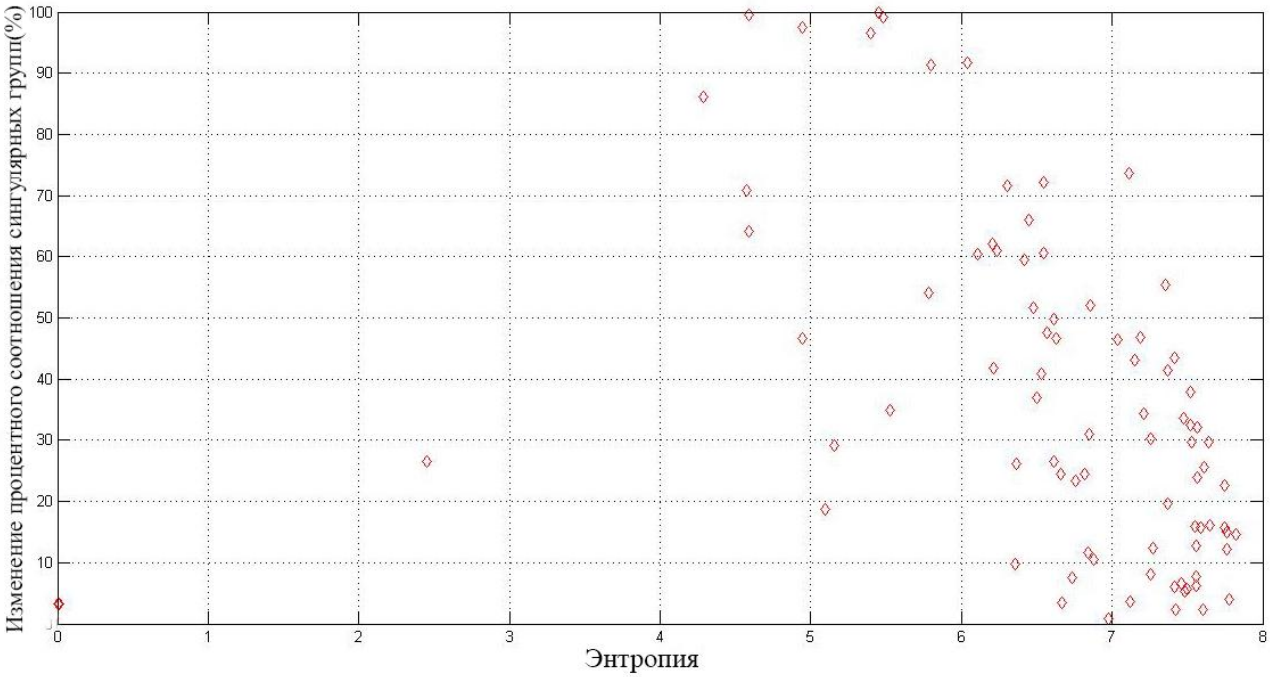

Рисунок 4.21 – Зависимость изменения процентного соотношения сингулярных групп от энтропии третьей битовой плоскости

4.4 Исследование статистических свойств цифровых изображения при встраивании информации в четвертую битовую плоскость

Для получения большего количества статистических данных исследуем статистические свойства цифровых изображений для изучения возможности встраивания информации в четвертые биты. Четвертые биты вносят значительный вклад в формирование изображения, что делает их использование достаточно опасным для обнаружения стеганографического скрытия информации в изображении-контейнере.

Как и в случае с менее значащими битами, анализ проводится по средствам методов стегоанализа Хи–квадрат и RS. На рисунке 4.22 представлена зависимость изменения статистики Хи–квадрат от дисперсии четвертой битовой плоскости.

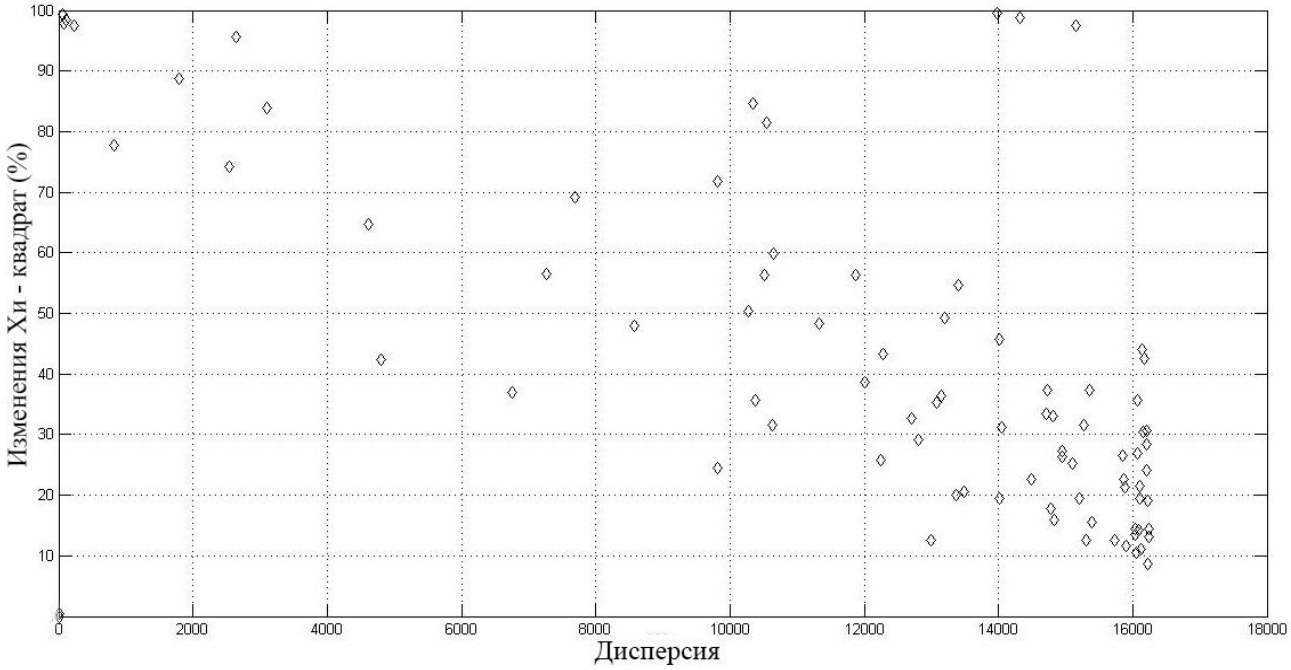

Рисунок 4.22 – Зависимость изменения Хи–квадрат от дисперсии четвертой битовой плоскости

Анализ зависимости показывает, что наименьшее изменение статистических показателей Хи–квадрат показывают цифровые изображения, четвертые биты которых обладают большей дисперсией. Однако следует отметить, что значения дисперсии четвертых битовых плоскостей исследуемых изображений все больше выравниваются относительно друг друга.

Рассмотрим зависимость изменения регулярных и сингулярных групп изображений относительно значений дисперсии (рисунок 4.23 и рисунок 4.24).

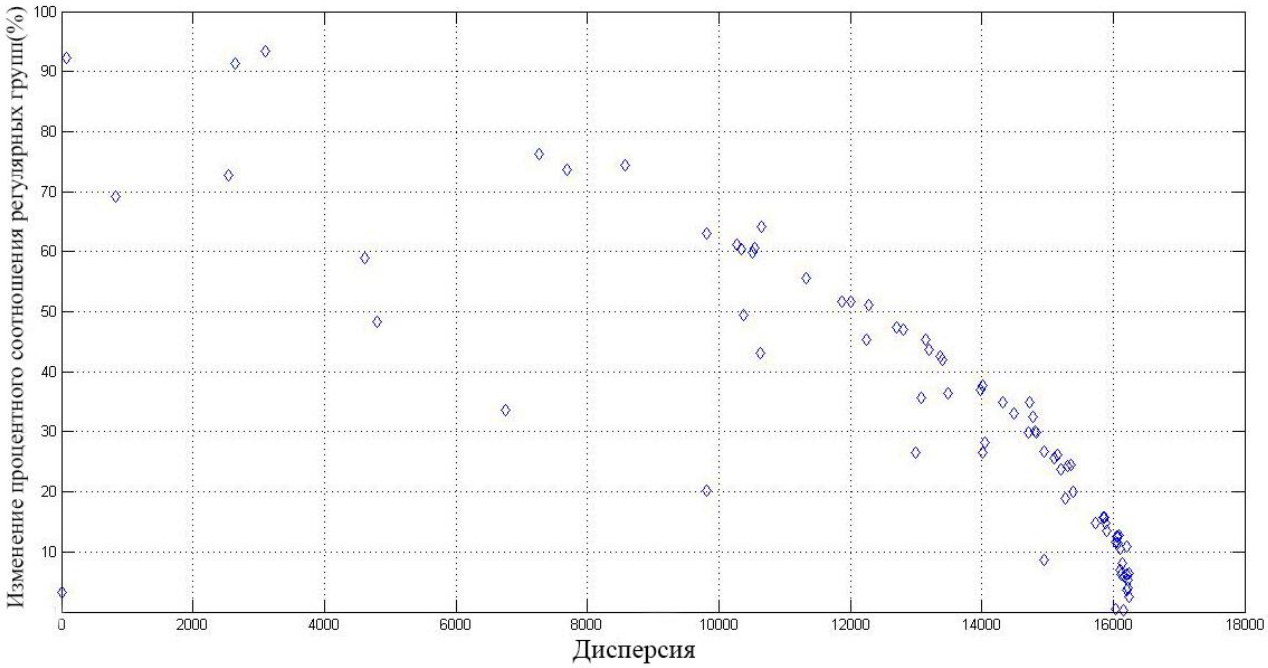

Рисунок 4.23 – Зависимость изменения процентного соотношения регулярных групп от дисперсии четвертой битовой плоскости

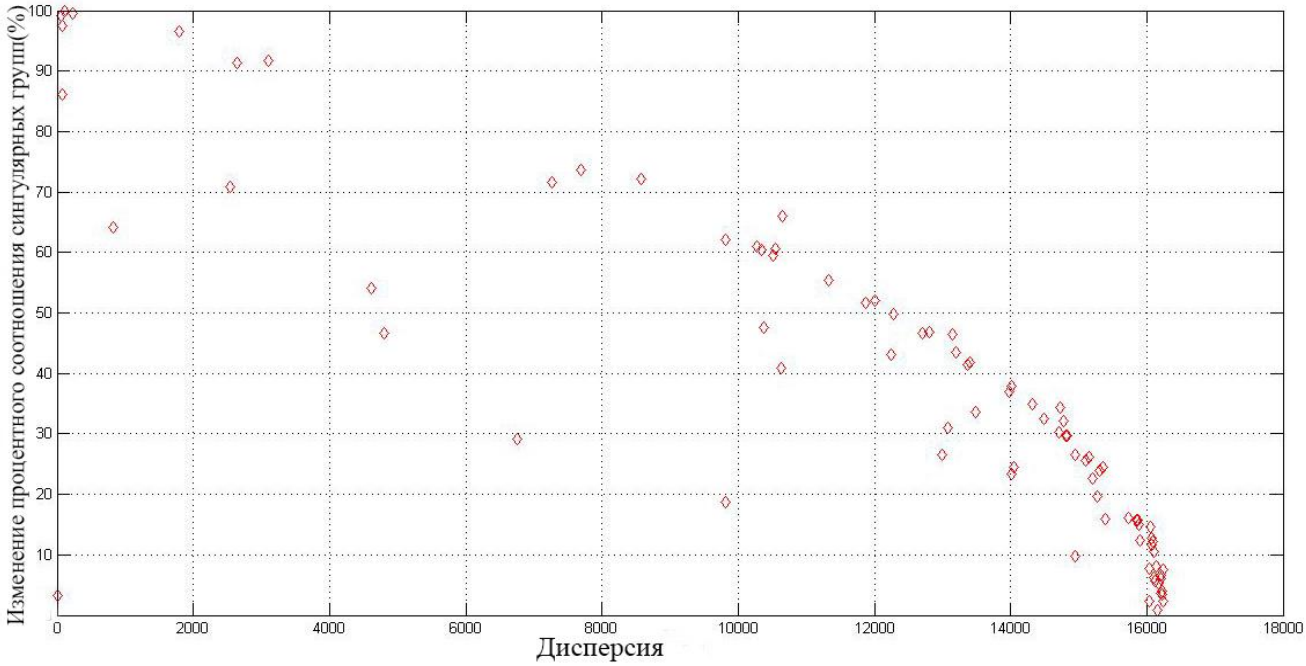

Рисунок 4.24 – Зависимость изменения процентного соотношения сингулярных групп от дисперсии четвертой битовой плоскости

Полученные данные RS анализа показывают, что, как и в случае анализа изменения Хи–квадрат, изображения, обладающие большей дисперсией в четвертой битовой плоскости, являются более устойчивыми к статистическим методам стеганоанализа.

Далее на рисунке 4.25 представлена зависимость изменения Хи–квадрат от энтропии четвертой битовой плоскости изображений.

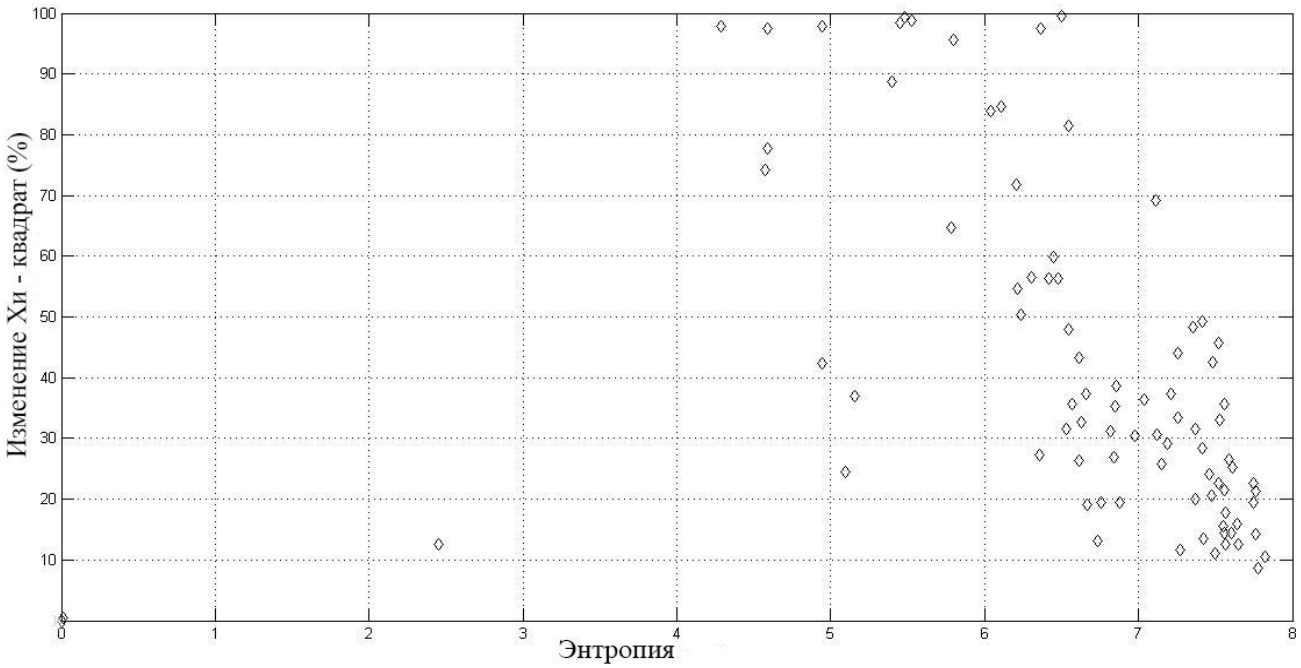

Рисунок 4.25 – Зависимость изменения Хи–квадрат от энтропии четвертой битовой плоскости

Не сложно заметить, что значения энтропии четвертых битовых плоскостей исследуемых изображений еще больше выровнялись друг относительно друга, по сравнению со значениями энтропии третьей или второй битовых плоскостей. Несмотря на это, результат анализа зависимости на рисунке 4.25, что при выборе изображения–контейнера для встраивания в четвертые биты, следует обращать более пристальное внимание на изображения, обладающие большим значением энтропии в этих битах.

Каким образом изменения регулярных и сингулярных групп RS стегоанализа, при встраивании в четверные биты цифровых изображений, зависят от энтпропии показано на рисунках 4.26 и 4.27.

При необходимости задействования четвертых бит изображения для сокрытия секретных данных, следует использовать изображения, обладающие максимальным значением энтропии четвертых бит.

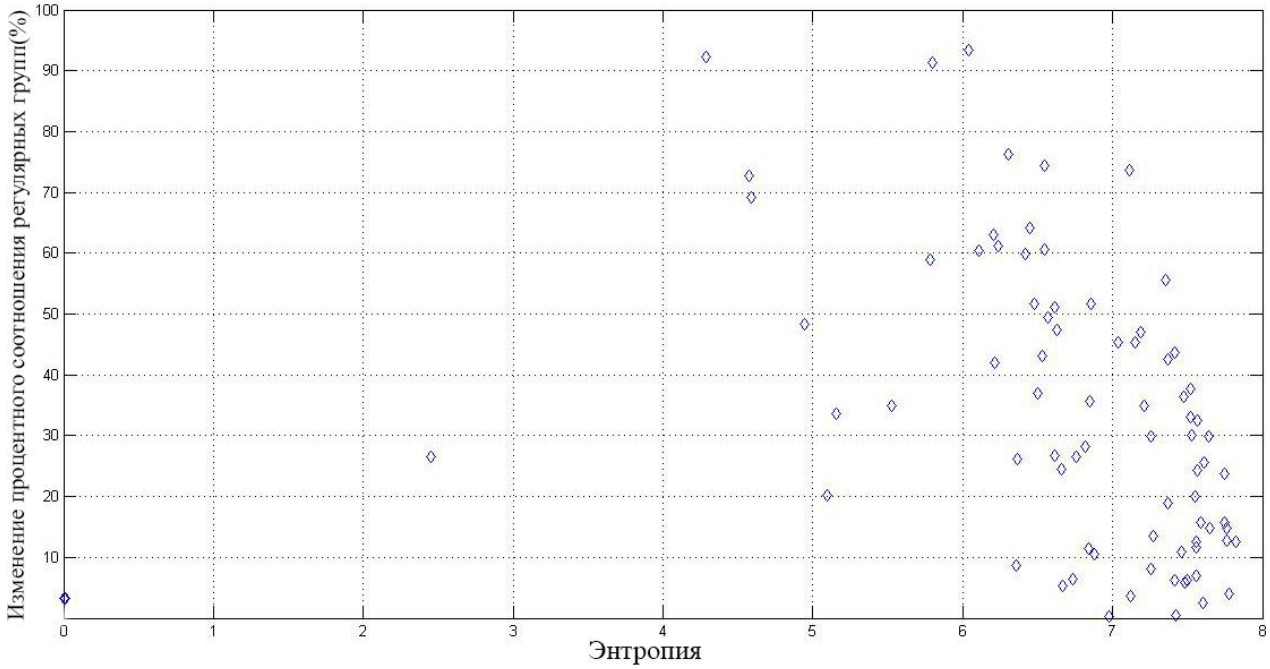

Рисунок 4.26 – Зависимость изменения процентного соотношения регулярных групп от энтропии четвертой битовой плоскости

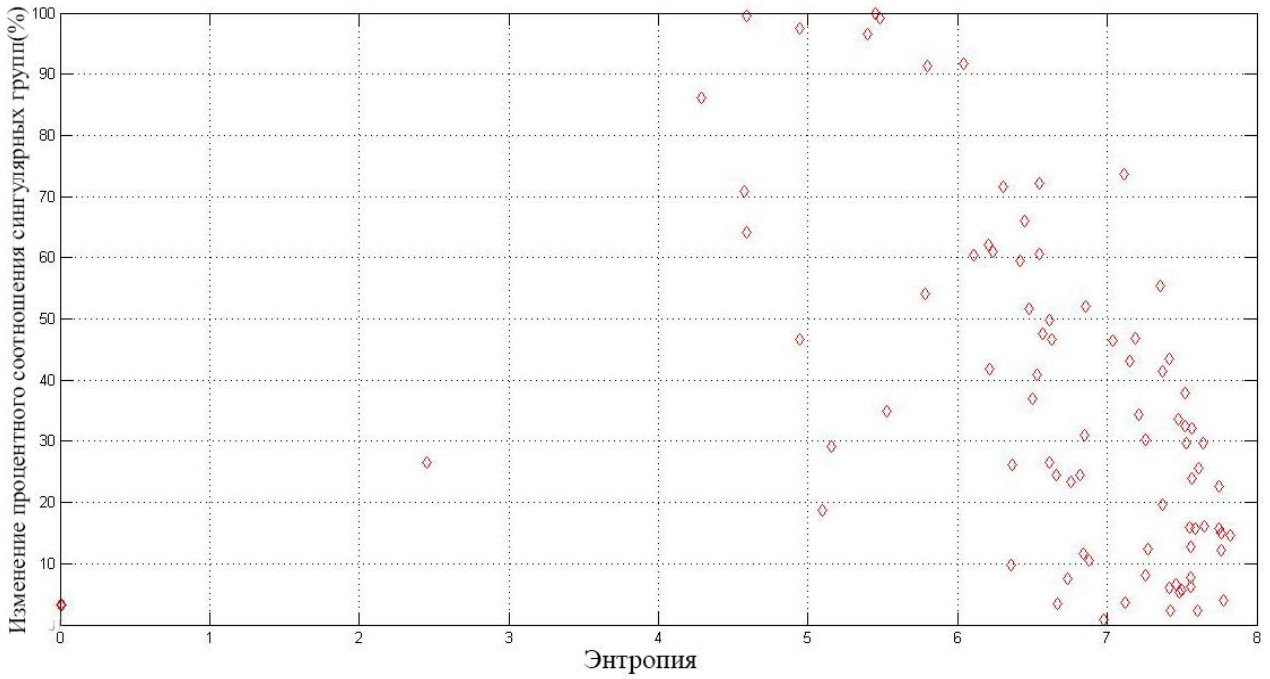

Рисунок 4.27 – Зависимость изменения процентного соотношения сингулярных групп от энтропии четвертой битовой плоскости

Выводы

В данной главе были исследованы статистические характеристики цифровых изображений, позволяющие выбрать наиболее подходящее изображение в качестве стеганографического контейнера для встраивания информации в ее различные битовые плоскости.

Таким образом, исходя из полученных данных, при выборе контейнеров для LSB стеганографии следует отдавать предпочтения изображениям с наибольшими значениями энтропии и дисперсии в битовых плоскостях. При равных значениях энтропии и дисперсии в младших битовых плоскостях, необходимо анализировать более старшие биты изображения, где также следует выбирать изображения с большими значениями энтропии и дисперсии. Анализ необходимо продолжать до того момента, как выбранное изображение будет в достаточной степени удовлетворять требованиям, предъявляемых к стеганографической системе.

Исследование приостановлено на этапе внедрения информации на уровне четвертых бит, так как уже на данном этапе изображения-контейнеры перестали удовлетворять условию визуальной скрытности внедрения сообщения. Однако полученные результаты дают возможность выбора подходящих изображений-контейнеров и обеспечивают стеганографическую стойкость системы при использовании более совершенных алгоритмов встраивания информации.

ЗАКЛЮЧЕНИЕ

Цель настоящей работы заключается в определении характеристик цифровых изображений–контейнеров, обеспечивающих наибольшую стойкость стеганографической системы. Для достижения указанной цели, был решен ряд теоретических и практических задач. В ходе проведенных исследований получены следующие основные результаты.

1. Сделан краткий обзор методов встраивания информации в пространственные области цифровых изображений. Дана краткая характеристика существующим методам.
2. Рассмотрен принцип действия LSB метода для стеганографической системы на основе цифровых изображений. Сформирован общий список требований и критериев выбора изображений-контейнеров для алгоритмов стеганографического скрытия информации на основе метода LSB. На основе сформулированных требований в качестве исследуемых изображений-контейнеров выбраны изображения формата BMP.
3. Проведено исследование статистических характеристик цифровых изображений BMP формата, изображения получены с помощью цифрового фотоаппарата путем конвертации из RAW формата. Конвертация из RAW формата, не предназначенного для непосредственной визуализации, в формат BMP происходит без потери качества изображения. В качестве исследуемых характеристик были выбраны монотонность, энтропия и дисперсия изображений. Исследование изменение статистических характеристик изображений проведено методами оценки критерия Хи-квадрат и RS методом стегоанализа. На основе полученных данных можно сделать вывод, что наибольшей стеганографической стойкостью обладают изображений с наибольшей энтропией и дисперсией младшей битовой плоскости, с повышением монотонности изображения стойкость стеганографической системы падает ухудшается.
4. Анализ дисперсии битовых плоскостей показал, что младшие битовые плоскости тестируемых изображения обладают достаточно большими близкими показателями дисперсии порядка 16000. Минимальное значение дисперсии младших битовых плоскостей составляет около 6000, что соответствует вероятности обнаружения сообщения метода RS анализа в 60 процентов. С увеличением бита встраивания увеличивается разброс значений дисперсии исследуемых изображений. RS метод стегоанализа показывает, что при показателях дисперсии стремящихся к 0, вероятность обнаружения

скрытой информации, содержащейся в изображении, стремится к 100 процентам. Таким образом, приведенные данные о дисперсии подтверждают теорию случайности младших бит и показывают их предпочтительность наиболее старшим битам при встраивании информации методом LSB. При возникновении необходимости задействования старших бит при LSB стеганографии, следует более внимательно относиться к выбору изображения в силу больших разбросов показателей дисперсии различных изображений и отдавать предпочтения изображениям, обладающих наибольшей дисперсией в этих битах.

5. Анализ данных об энтропии показывает, что ее относительная величина младших битовых плоскостей является большей по сравнению со старшими битами. Максимальное относительное значение младших битовых плоскостей составляет порядка 18, в свою очередь максимальное значение энтропии второй битовой плоскости уменьшилось более чем в два раза. Максимальная вероятность обнаружения внедрения сообщения в младший бит изображения составляет около 60 процентов при значении энтропии равном 3. При задействовании наиболее старших бит изображений, вероятность обнаружения в 60 процентов достигается при относительном значении энтропии около 6. Необходимо отметить, что с увеличением бита встраивания, относительное значение энтропии битовых плоскостей тестируемых изображений стремится к максимальному значению. Подводя итог, можно сделать вывод, что, как и в случае с дисперсией изображений, при выборе изображения в качестве контейнера для стеганографического скрытия данных методом LSB, следует выбирать изображения с большими значениями энтропии в битовых плоскостях.

Результаты представлены для цифровых изображений формата BMP, однако могут быть адаптированы и на иные форматы.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Конахович Г. Ф., Пузыренко А. Ю. Компьютерная стеганография. Теория и практика. - К.: "МК-Пресс", 2006.
2. Pfitzmann B. Information Hiding Terminology, in Information Hiding, Springer Lecture Notes in Computer Science, v.1174, 1996
3. Гилен Р. Моя первая книга о Microsoft Office PowerPoint 2003. Эксмо, 2005.
4. Генне, О. В. Основные положения стеганографии. "Защита информации. Конфидент" №3 за 2000 год
5. Гонсалес Р., Вудс Р. Цифровая обработка изображений (перевод с английского), под ред. Чочиа П. А., ТЕХНОСФЕРА, Москва, 2005. - 1072 с.
6. Грибунин В. Г., Оков И. Н., Туринцев И. В. Цифровая Стеганография. - М.:Солон-Пресс, 2002 - 272 с.
7. Замена наименее значащего бита [Электронный ресурс]. – Режим доступа: http://www.nestego.ru/2012/07/lsb.html. – Дата доступа: 17.01.2015
8. J. Fridrich, R. Du, and L. Meng, “Steganalysis of LSB Encoding in Color Images”, *ICME 2000*, New York City.
9. Бородин Г.А., Чиркова С.В., «Классификация критериев выбора контейнера для LSB-метода», «Радиоэлектроника, электротехника и энергетика 13-ая межд. науч.-техн. конф. студ. и асп. Тезисы докладов в 3-ех томах». Т.1. –М.: МЭИ, 2007
10. J. Fridrich, G. Miroslav, R. Du Steganalysis Based on JPEG Compatibility. - SUNY Binghamton, New York, 2001. - 6 с
11. Разинков Е.В. Стойкость стеганографических систем /Е.В.Разинков, Р.Х.Латыпов //Учёные записки Казан.гос.ун-та. –Казань, 2009. –Т. 151,No 2
12. Westfeld A. Attacks on Steganographic Systems: Breaking the Steganographic Utilities EzStego, Jsteg, Steganos and S-Tools-and Some Lessons Learned / A. Westfeld, A. Pfitzmann // 3rd International Workshop on Information Hiding (2000)
13. J. Friedrich, G. Miroslav, R. Du. Reliable Detection of LSB Steganography in Color and Grayscale Images. Binghampton, New York: SUNY, 2001.
14. Фисенко В.Т., Фисенко Т.Ю. Компьютерная обработка и распознавание изображений: учеб. пособие. - СПб.: СПбГУ ИТМО, 2008
15. Кустова В. Н. и Федчука А. А. "Методы встраивания скрытых сообщений" ("Защита информации. Конфидент", №3, 2000

ПРИЛОЖЕНИЕ

Листинг исходного кода приложения.

Исходный код main.m:

```
close all;
clear all;
clc;
pause(0.1);

%% config
bit = 1;

%% import lib
%%path('stego',path);
%%path('analyze',path);

imageFolder = 'images';

saveFile_old = 'RESULT3.mat’;
saveFile = 'RESULT4.mat';

forceBlackWhite = 0; % convert images to gray !

graph3d = 0; %% plot 3d graph instead of 2d

%% console output:
%% OX line:
%% 1-monotone
%% 2-dispersion
%% 3-entropy
%% OY line:
%% x - chi-square
%% r - regular group (RS analysis)
%% s - singular group (RS analysis)
graph = '123xrs';

%%
analyzeResult = [];
path = [pwd,'\',imageFolder,'\'];

%% scan folder
if (exist(saveFile,'file'))
    %% load calculated data
    disp('Previous calculation loaded’);
    load(saveFile,'analyzeResult');

else if(exist(saveFile_old,'file'))
    %% load calculated data
```

```
    disp(['Previous calculation loaded (', saveFile_old, ' now
deprecated. New calculations will be saved as ',saveFile, ')'] );
    loadedData = load(saveFile_old,'analyzeResult');
    analyzeResult_old = loadedData.analyzeResult;

    analyzeResult = zeros(6, size(analyzeResult_old, 2));
    analyzeResult(1,:) = analyzeResult_old(1,:)*100; %% immonot
    analyzeResult(2,:) = analyzeResult_old(2,:); %% d
    analyzeResult(3,:) = analyzeResult_old(3,:); %% ent

    analyzeResult(4,:) = 100*(analyzeResult_old(5,:)-
analyzeResult_old(4,:)); %% chi2 (take the difference)
    analyzeResult(5,:) =
100*(analyzeResult_old(6,:)./analyzeResult_old(8,:)); %% ps_reg
(the difference)
    analyzeResult(6,:) =
100*(analyzeResult_old(7,:)./analyzeResult_old(8,:)); %% ps_sing
(the difference)

    %% v3: analyzeResult = [analyzeResult,[immonot; d; ent;
chi2_orig; chi2_enc; dR; dS; N]]; %#ok<AGROW>
    %% v4: analyzeResult = [analyzeResult,[immonot*100; d; ent;
100*(chi2_enc-chi2_orig); 100*dR./N; 100*dS./N ]]; %#ok<AGROW>

else

if(forceBlackWhite)
    disp(['Colored images will be converted to gray. It can save
time']);
else
    disp(['Working with colored images. Take a quite, it will take
much more time...']);
end;

%% calculate data
filename_arr = dir(path);
n = length(filename_arr);
for i=1:n
    filedata = filename_arr(i);
    if filedata.isdir>0 continue; end;

%% skip folder
    [pathstr,name,ext] = fileparts([filedata.name]);

    if(strcmp(ext,'.png') && strcmp(ext,'.bmp') ) continue; end;

%% skip non-images
    disp('-----------------------------');
    disp(['Loading image: ', name,ext, ' (',num2str(i),' of
',num2str(n),')']);

    image = imread([path,'\',name,ext]);
```

```
    if(length(size(image))==3 && forceBlackWhite)
        %%make it black & white
        image = rgb2gray(image);
    end;

    %% resize image to make size equivalent
    %% image = imresize(image,[480 640]);

    imshow(image);
    disp(['Analyze image...']);
    pause(0.1);

    [immonot, d, ent, chi2_orig, chi2_enc, dR, dS, N] =
image_analyze_complex(image, bit);

    %% chi2_enc-chi2_orig    (take the difference)
    %% dR, dS    (the difference)
    analyzeResult = [analyzeResult,[immonot*100; d; ent;
100*(chi2_enc-chi2_orig); 100*dR./N; 100*dS./N ]]; %#ok<AGROW>

    disp(['monotone: ',num2str(immonot), ' (', num2str(
ceil(immonot*100)), '%)']);
    disp(['dispersion: ', num2str(d)]);
    disp(['entropy: ', num2str(ent) ]);
    disp(['’and: ', num2str(chi2_orig), ' -> ', num2str(chi2_enc),
' (', num2str( ceil(chi2_orig*100)), '% -> ', num2str(
ceil(chi2_enc*100)), '%)']);
    disp(['regular group changes: ',num2str( dR ), ' (',num2str(
round(N\dR*10000)/100),'%)']);
    disp(['singular group changes: ',num2str( dS ), ' (',num2str(
round(N\dS*10000)/100),'%)']);
    pause(0.1);
end; end;
close all;

%% save result
save(saveFile,'analyzeResult');
end;

%% plot graphics

disp('-----------------------------');
disp(' construct graphs...');

if (graph3d)

    error('not implemented');

else

    param_map = struct('f1', 1, 'f2', 2, 'f3', 3, 'x', 4, 'r', 5,
's', 6);
```

```
    param_label = struct('f1', 'Monotone (%)',...
                         'f2', 'Dispersion',...
                          'f3','Entropy',...
                          'x', 'Chi-square changes (%)',...
                          'r', 'regular group changes(%)',...
                          's', ' regular group changes (%)');

    param_lim = struct('f1', [0, 100],...
                       'x', [0, 100],...
                       'r', [0, 100],...
                       's', [0, 100]);


    graph_style = 'd';
    %% only for OY
    param_color = struct( 'x', 'k',...
                          's', 'r');

    pairs = l_pairs(graph);
    for i=1:length(pairs)

        %% parse config
        graph_config = pairs(i,:);
        ox_p = ['f', graph_config(1)]; %% digit
        oy_p = graph_config(2); %% letter

        %%  verify reqirements
        if(~isfield(param_map, ox_p))
            error(['Invalid graph param: ',ox_p]);
        end;
        if(~isfield(param_map, oy_p))
            error(['Invalid graph param: ',oy_p]);
        end;

        %%  color
        color = graph_style;
        if(isfield(param_color, oy_p))
            color = [color, getfield(param_color,oy_p)];
%#ok<AGROW,GFLD>
        end;

        %%  get values
        ox_values = analyzeResult( getfield(param_map,ox_p) ,:);
%#ok<GFLD>
        oy_values = analyzeResult( getfield(param_map,oy_p) ,:);
%#ok<GFLD>

        %%  PLOT
        figure();
        plot(ox_values,oy_values, color);
        grid on;
```

```
        %%  labels
        if(isfield(param_label, ox_p))
            xlabel( getfield(param_label,ox_p) ); %% #ok<GFLD>
        end;
        if(isfield(param_label, oy_p))
            ylabel( getfield(param_label,oy_p) ); %% #ok<GFLD>
        end;

        % limits
        if(isfield(param_lim, ox_p))
            xlim( getfield(param_lim,ox_p) ); %% #ok<GFLD>
        end;
        if(isfield(param_lim, oy_p))
            ylim( getfield(param_lim,oy_p) ); %% #ok<GFLD>
        end;
    end;
end

disp('Done');

%%
disp(['[', datestr(clock), ']',' Done!']);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код lorem_ipsum.m:

```
function new_str = lorem_ipsum(str, charlength)
%% new_str = lorem_ipsum(str, charlength)
%% more description will be later...
%% more http://www.lipsum.com/

    new_str = '';
    repeat = ceil(charlength/length(str));
    for i=1:repeat
        new_str = [new_str,str];
    end;
    %%cut unnecessary symbols
    new_str = new_str(1:charlength);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код l_pairs.m:

```
function pairs = l_pairs(str)
%% pairs = l_pairs(str)
%% split string into pairs of letter+digit
   pairs = [];
   char_offset = 1;

   while(1==1)
```

```
     char_pos = lookup_char(str, char_offset);
     if (char_pos>0)
        digit_offset = 1;
        while(1==1)
            digit_pos = lookup_digit(str, digit_offset);
            if(digit_pos>0)
                %% char and digit found
                pairs = [pairs; [str(digit_pos), str(char_pos)]];
                digit_offset = digit_pos+1;
            else
                break;
            end;
        end;
        char_offset = char_pos+1;
     else
        break;
     end;
   end;
end

function pos = lookup_char(str, offset)
    for pos=offset:length(str)
        letter = str(pos)*1-'0';
        if(letter>=0 && letter<=9)
            %% digit
        else
            %% char
            return;
        end;
    end;
    pos = -1;
end

%% induss code :)
function pos = lookup_digit(str, offset)
    for pos=offset:length(str)
        letter = str(pos)*1-'0';
        if(letter>=0 && letter<=9)
            %% digit
            return;
        else
            %% char
        end;
    end;
    pos = -1;
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код imhist3.m:

```
function [H] = imhist3(image)
%% H = imhist3(image)
%% imhist for colour images. return array of hist

%% s = size(image);
%% H = [];
%% for i=1:s(3)
%% H = [H, imhist(image(:,:,i))];
%% end;
      if(length(size(image))~=3)
        error('not implemented');
      end;


      H = int32(zeros(256,256,256));
      for i=1:size(image,1)
          for j=1:size(image,2)
            pixel = image(i,j,:);
            H(pixel(1)+1, pixel(2)+1, pixel(3)+1 )  = H(pixel(1)+1,
pixel(2)+1, pixel(3)+1 ) + 1;
          end;
      end;


%% if (nargout == 0)
%% plot_result(x, y, map, isScaled, class(a), range);
%% else
%% yout = y;
%% end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код image_ps.m:

```
function [dR, dS, N]  = image_ps(image, blocksize)
%%% function [R_plus, R_minus, S_plus, S_minus,  U_plus, U_minus,
N_group]  = image_ps(image, blocksize)
%% [R_plus, R_minus, S_plus, S_minus,  U_plus, U_minus] =
image_ps(image, blocksize)
%% image - image
%% R_plus, R_minus – regular groups
%% S_plus, S_minus - – singular groups
%% U_plus, U_minus – not used groups
%% N_group – common number of groups

%% blocksize - size of block. must divide on 2
if(blocksize<=0 || mod(blocksize, 2))
```

```
   error('Block size must divide on 2 (like 2*n)');
end;

if(mod( size(image, 1), blocksize) ~=0)
   error(['You must block size aliquot to image width
(n*blocksize=width). Cannot use blocksize ', num2str(blocksize), '
for image with width ', num2str(size(image, 1))] );
end;


%% theoretical values (average)
   if(length(size(image))==2)
[dR, dS, N] = image_ps_layer(image, blocksize);
   else
     res_r = zeros(1,3);
     res_s = zeros(1,3);

[res_r(1), res_s(1), N] = image_ps_layer(image(:,:,1), blocksize);
% R
[res_r(2), res_s(2), N] = image_ps_layer(image(:,:,2), blocksize);
% G
[res_r(3), res_s(3), N] = image_ps_layer(image(:,:,3), blocksize);
% B
        dR = sum(res_r);
        dS = sum(res_s);
        N =  N*3;
        %p = p/3; % because 3 layer
        %histval = imhist3(image);
    end;
end


function [dR, dS, N]  = image_ps_layer(image, blocksize)

    if(length(size(image))~=2)
        error(['You must pass only one layer (R, G or B) for
colored images']);
    end;

%% invert last bit
    image_one = invert_bit(image);
%% invert +1
    image_minus_one = invert_bit_plus_one(image);

%% select sum of changes
%% more: http://network-journal.mpei.ac.ru/cgi-
bin/main.pl?l=ru&n=11&pa=13&ar=4
    s = size(image);
    L = floor( s(1)/blocksize); % count block in one row
    N = L*s(2); %result block count

    f_original = zeros(L, s(2) );
```

```
    f_one = zeros(L, s(2) );
    f_m_one = zeros(L, s(2) );

    for i=1:L
       diapason = (i-1)*blocksize+1:i*blocksize;

       %% SUM HERE. can be dispersion
       f_original(i,:) = sum( image(diapason, :),1);
       f_one(i,:) = sum( image_one(diapason, :),1);
       f_m_one(i,:) = sum( image_minus_one(diapason, :),1);
    end;

%% count groups
    R_plus = sum(sum(uint8( f_one>f_original ))); %% regular
groups
    S_plus = sum(sum(uint8( f_one<f_original ))); %% singular
groups
    %% U_plus = sum(sum(uint8( f_one==f_original ))); %% not used
groups
    R_minus = sum(sum(uint8( f_m_one>f_original ))); %% regular
groups
    S_minus = sum(sum(uint8( f_m_one<f_original ))); %% singular
groups
    %% U_minus = sum(sum(uint8( f_one==f_original ))); %% not used

%% difference on level of direct and indirect groups

dR = -(R_plus-R_minus);
    dS = S_plus-S_minus;
    %dU = U_plus-U_minus;

%% disp([num2str(dR), ' / ', num2str(dS), ' / ', num2str(N)]);
end

function image_one = invert_bit(image)
    %% invert last bit
    image_one = bitset(image, 1, 1-bitget(image, 1));
end

function image_minus_one = invert_bit_plus_one(image)
    %% invert +1

    image = uint8(image); %% VERY important

    %% e bit plus one
    ones_layer = image.*bitget(image, 1);
    %% convert 1->2, 3->4, ..., 253->254, 255->255 (! we cannot
exceed upper 255!!!)
    ones_layer = ones_layer+1;
    %% convert 255->0,
    ones_layer = ones_layer.*(1-bitget(ones_layer, 1));
```

```
    %% convert 0->255,
    zeros_layer = 255.*uint8(image==0);

    %% convert 2->1, 4->3, ..., 254->253, 0->0 (! we already
convert it, so - just ignore)
    odd_layer = image.*(1-bitget(image, 1));
    odd_layer = odd_layer-1;

    %%combine layers
    image_minus_one = ones_layer+zeros_layer+odd_layer;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код image_monotonn.m:

```
function [monoton] = image_monotonn(image, histval)
%% calculate image monotonnost'
%% histval - calculated image histogramm. can save a bit processor
time. You can simply pass [] (empty array) to calculate it
%% [monoton] = image_monotonn(image)

    if(numel(histval)==0)
        %% recalc histogramm
        if(length(size(image))==2)
            %% grayscale image is easier
            histval = imhist(image);
        else
            %% colour image is more difficult
            histval = imhist3(image);
        end;
    end;

    if ( length(size(image))~=length(size(histval)) )
        error('histval must be calculated from image via hist or
hist3');
    end;

    if (length(size(image))==2)
%% grayscale image is easier
%% histval = imhist(image);
%% perfomance trick: value passed here

%% monoton = max(histval)/numel(image);

        %% get index of max element
        maxval = max(max(max(histval)));
        imax = find(maxval==histval, 1); % much faster than for-
end loop

        scale = ones(256, 1);
```

```
        for i=1:256
            scale(i) = 1 - exp( abs(imax-i)/256 )/exp(1); %
Gaussian attenuation
        end;

        monoton = sum(histval .* scale)/numel(image);


        %% disp(['  Monochrome image']);
    else
        %% colour image is more difficult
        %% histval = imhist3(image);
        %% perfomance trick: value passed here

        %% get index of max element
        maxval = max(max(max(histval)));
        indexmax = find(maxval==histval, 1); %% much faster than
for-end loop

        kMax = floor(indexmax/(256*256))+1;
        jMax = floor( (indexmax-(kMax-1)*256*256) /256)+1;
        iMax = indexmax-(kMax-1)*256*256 - (jMax-1)*256;

[iMax, jMax, kMax]
%% too hard to calculate. load preset

%% on recalc - it will recalculate about 1-3 minute
%% scale = ones(256, 256, 256);
%% for i=1:256
%% for j=1:256
%% for k=1:256
%% scale(i,j,k) = ( 1- sqrt( (i-1)^2+(j-1)^2+(k-
1)^2)/(255*sqrt(3)) )^3 ; %%  parabolic attenuation
%% scale(i,j,k) = exp(-( sqrt( (i-1)^2+(j-1)^2+(k-
1)^2)/(255*sqrt(3)) )^2 ) ; %% gaussian attenuation
%% end;
%% end;
%% end;
%% figure; surf(1:256, 1:256, scale(:,:,1))
%% figure; surf(1:256, 1:256, scale(:,:,128))
%% scaleInt = int8(scale*127); % save a lot of memory
%% save('gaussi3d.mat', 'scaleInt')

          load('gaussi3d.mat','scaleInt');
          scale = int32(scaleInt);
          histval = int32(histval);

%% cube sum
          monoton = 0;
          monoton = monoton + sum(sum(sum( histval(iMax:256,
jMax:256,  kMax:256).*scale(1:(256-iMax+1), 1:(256-jMax+1),
1:(256-kMax+1)) )));
```

```
          monoton = monoton + sum(sum(sum( histval(iMax:-1:1,
jMax:256,  kMax:256).*scale(1:iMax,         1:(256-jMax+1),
1:(256-kMax+1)) )));
          monoton = monoton + sum(sum(sum( histval(iMax:256,
jMax:-1:1, kMax:256).*scale(1:(256-iMax+1), 1:jMax,
1:(256-kMax+1)) )));
          monoton = monoton + sum(sum(sum( histval(iMax:-1:1,
jMax:-1:1, kMax:256).*scale(1:iMax,         1:jMax,
1:(256-kMax+1)) )));

          monoton = monoton + sum(sum(sum( histval(iMax:256,
jMax:256,  kMax:-1:1).*scale(1:(256-iMax+1), 1:(256-jMax+1),
1:kMax) )));
          monoton = monoton + sum(sum(sum( histval(iMax:-1:1,
jMax:256,  kMax:-1:1).*scale(1:iMax,         1:(256-jMax+1),
1:kMax) )));
          monoton = monoton + sum(sum(sum( histval(iMax:256,
jMax:-1:1, kMax:-1:1).*scale(1:(256-iMax+1), 1:jMax,
1:kMax) )));
          monoton = monoton + sum(sum(sum( histval(iMax:-1:1,
jMax:-1:1, kMax:-1:1).*scale(1:iMax,         1:jMax,
1:kMax) )));


%% remove doubled plains
          monoton = monoton - sum(sum( histval(iMax:256,
jMax:256,  kMax).*scale(1:(256-iMax+1), 1:(256-jMax+1), 1) ));
          monoton = monoton - sum(sum( histval(iMax:-1:1,
jMax:256,  kMax).*scale(1:iMax,         1:(256-jMax+1), 1) ));
          monoton = monoton - sum(sum( histval(iMax:256,  jMax:-
1:1, kMax).*scale(1:(256-iMax+1), 1:jMax,         1) ));
          monoton = monoton - sum(sum( histval(iMax:-1:1, jMax:-
1:1, kMax).*scale(1:iMax,         1:jMax,         1) ));

          monoton = monoton - sum(sum( histval(iMax:256,  jMax,
kMax:256).*scale(1:(256-iMax+1), 1, 1:(256-kMax+1)) ));
          monoton = monoton - sum(sum( histval(iMax:-1:1, jMax,
kMax:256).*scale(1:iMax,         1, 1:(256-kMax+1)) ));
          monoton = monoton - sum(sum( histval(iMax:256,  jMax,
kMax:-1:1).*scale(1:(256-iMax+1), 1, 1:kMax) ));
          monoton = monoton - sum(sum( histval(iMax:-1:1, jMax,
kMax:-1:1).*scale(1:iMax,         1, 1:kMax) ));

          monoton = monoton - sum(sum( histval(iMax,  jMax:256,
kMax:256) .*scale( 1, 1:(256-jMax+1), 1:(256-kMax+1)) ));
          monoton = monoton - sum(sum( histval(iMax,  jMax:-1:1,
kMax:256) .*scale( 1, 1:jMax,         1:(256-kMax+1)) ));
          monoton = monoton - sum(sum( histval(iMax,  jMax:256,
kMax:-1:1).*scale( 1, 1:(256-jMax+1), 1:kMax) ));
          monoton = monoton - sum(sum( histval(iMax,  jMax:-1:1,
kMax:-1:1).*scale( 1, 1:jMax,         1:kMax) ));
```

```
%% add lines summed:
          monoton = monoton + sum( histval(iMax,  jMax,  kMax:256)
.*scale( 1, 1, 1:(256-kMax+1)) );
          monoton = monoton + sum( histval(iMax,  jMax,  kMax:-
1:1).*scale( 1, 1, 1:kMax) );

          monoton = monoton + sum( histval(iMax,  jMax:256,
kMax).*scale(1, 1:(256-jMax+1), 1) );
          monoton = monoton + sum( histval(iMax,  jMax:-1:1,
kMax).*scale(1, 1:jMax,         1) );

          monoton = monoton + sum( histval(iMax:256,  jMax,
kMax).*scale(1:(256-iMax+1), 1, 1) );
          monoton = monoton + sum( histval(iMax:-1:1, jMax,
kMax).*scale(1:iMax,         1, 1) );

%% compensate central point
          monoton = monoton - sum( histval(iMax,  jMax,
kMax).*scale(1, 1, 1) );


%% normalize
          monoton = monoton/(size(image,1)*size(image,2));
          monoton = monoton/127; %% see algoritm of calc scaleInt
matrix


         %% monoton = sum(histval .* scale)/numel(image);

        %% disp(['  Colour image']);
    end;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код image_extract.m:

```
function [str, newoffset] = image_extract(image, offset, bit,
bytecount)
%% [value, newoffset] = image_extract(image, offset, bit)
%% rarray - image data is inserted
%% value - value that is built
%% size - type value for the size (in bytes)
%% offset - offset from beginning
%% bit - number of bit to be embedded
%% wrarray - with built-in image data
%% newoffset
```

```
%% bitget(A,BIT) gets the bit at position BIT
%%    return 0 or 1.
    str = '';
    for i=1:bytecount
        value = 0;
        for b=1:8  %*2 %2-byte character available
            value = bitset(value, b, bitget(image(offset+b-1),
bit) );
        end;
        str = [str, char(value)];
        offset = offset+8;
    end
    newoffset = offset;
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код image_entropiya.m:

```
function [ent] = image_entropiya(image, histval)
%% calculate image entropiya
%% histval - calculated image histogramm. can save a bit processor
time. You can simply pass [] (empty array) to calculate it
%% [ent] = image_entropiya(image)

    image = bitget(image, 1)*255; % make image from bitcut

%% perfomance trick: value passed here
    if(numel(histval)==0)
        %% recalc histogramm
        if(length(size(image))==2)
            %% grayscale image is easier
            histval = imhist(image);
        else
            %% colour image is more difficult
            histval = imhist3(image);
        end;
    end;


%% remove zeros (it will be limits to zero, but matlab cannot
calculate limits)
    nonzero_index = find(histval~=0);
    n = numel(nonzero_index);
    p = zeros(1, n);
    for i=1:n
        p(i) = histval(nonzero_index(i));
    end;

    p = p./(size(image,1)*size(image,2)); % possibilty
    ent = sum(p.*log(1./p)/log(2));
end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код image_embed.m:

```
function [image_embed, newoffset] = image_embed(image, value, bit,
offset)
%% [image_embed, newoffset] = image_embed(image, value, bit,
offset)
%% rarray - image data is inserted
%% value - value that is built
%% size - type value for the size (in bytes)
%% offset - offset from beginning
%% bit - number of bit to be embedded
%% wrarray - with built-in image data
%% newoffset


%% BITSET(A,BIT,V) sets the bit at position BIT to the value V.
%%   V must be either 0 or 1.
    image_embed = image;
    for i=1:length(value)
        for j=1:8  %*2 % 2-byte character available
            image_embed(offset) = bitset(image(offset), bit,
bitget(1*value(i), j) );
            offset=offset+1;
        end;
    end;
    newoffset = offset;
end
```

Исходный код mage_dispersion.m

```
function [D, M] = image_dispersion(image)
%% [D, M] = image_dispersion(image)
%% more description will be later...
%% M - mat.ogidanie (mean)
%% D - dispersion (var)
%% more: http://network-journal.mpei.ac.ru/cgi-
bin/main.pl?l=ru&n=11&pa=13&ar=5
    image = bitget(image, 1)*255; % make image from bitcut


M = mean(double(image(:)));
%% M = mean(mean_blocks);
    D = var(double(image(:)));
%% D = var(mean_blocks);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код image_chi2.m:

```
function p = image_chi2 (image)
%% p = image_chi2(image)
%%image - image
%% histval - calculated image histogramm. can save a bit processor
time. You can simply pass [] (empty array) to calculate it

%% p - probability of having stegocontainer, embedded by LSB
method inside image
%% p - chi2 dispersion [0,1]
%% more: http://users.ece.cmu.edu/~adrian/487-s06/westfeld-
pfitzmann-ihw99.pdf
%% https://ru.wikipedia.org/wiki/Критерий_согласия_Пирсона


%%theoretical values (average)
    if(length(size(image))==2)
%% grauscale image is easier
        p = image_chi2_layer(image);
    else
        p=0;
        p = p + image_chi2_layer(image(:,:,1)); % R
        p = p + image_chi2_layer(image(:,:,2)); % G
        p = p + image_chi2_layer(image(:,:,3)); % B
        p = p/3; % because 3 layer
        %% colour image is more difficult
        %% histval = imhist3(image);
    end;
end


function p = image_chi2_layer(image)

    histval = imhist(image);

    hist_size = numel(histval);
    %% n = sum(h)/2; % correct, but difficult to uderstand

    %% n = numel(image)/2;
    pairs_theory = ( histval(1:2:hist_size)+histval(2:2:hist_size)
)/2;
    pairs_sum = sum(pairs_theory); %%  correct

    %% practical values
    pairs_practice = histval(2:2:hist_size);

    %remove zeros (it will be limits to zero, but matlab cannot
calculate limits)
    nonzero_index = find(pairs_theory~=0);
    pairs_theory_wo_zero = pairs_theory(nonzero_index);
```

```
    pairs_practice_wo_zero = pairs_practice(nonzero_index);

%% x^2
    p = 1 - sum( (pairs_practice_wo_zero-
pairs_theory_wo_zero).^2./pairs_theory_wo_zero )/pairs_sum;

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код image_bitshift_cycle.m:

```
function result = image_bitshift_cycle(image, n, n0)
%% pos = e_strfind(str, find)
%% more description will be later...
    result = image;
    if(n>=0)
        for i=1:n
            result = bitget(result(:),n0) + bitshift(result(:),1);
        end;
    else
        for i=1:-n
            result =  bitshift( bitget(result(:),1), n0-1) +
bitshift(result(:),-1);
        end;
    end;
end


%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Исходный код image_analyze_complex.m:

```
function [immonot, d, ent, chi2_orig, chi2_enc, dR, dS, N] =
image_analyze_complex(image, bit)
%% [] = image_analyze_complex(image)
%% more description will be later...
 encode_str = ['Steganography is the art and science of writing
hidden messages in such a way that no one, apart from the
senderand intended recipient, suspects the existence of the
message, a form of security through obscurity. The word
steganography is of Greek origin and means "concealed writing"
from the Greek words steganos (????????) meaning "covered or
protected", and graphei (?????) meaning "writing". The first
recorded use of the term was in 1499 by Johannes Trithemius in his
Steganographia, a treatise on cryptography and steganography
disguised as a book on magic. Generally, messages will appear to
be something else: images, articles, shopping lists, or some other
covertext and, classically, the hidden message may be in invisible
ink between the visible lines of a private letter.'];
```

```
%% bit = 1;
%% create string to completely fill container
 n = floor( numel(image) / 8);
 fillfile_encode_str = lorem_ipsum(encode_str, n );

%% embed
 image_encoded = image_embed(image, fillfile_encode_str, bit, 1);
%% 1 - offset

imshow(image_encoded);
 pause(0.1);

%% shift img
image = image_bitshift_cycle(image, bit-1, 8);

%% HELPER (for faster calculating
if(size(image,3)==1)
    %% grayscale image is easier
    histval = imhist(image);
else
    %% colour image is more difficult
    histval = imhist3(image);
end;

%% monotonnost'
 immonot = image_monotonn(image, histval);

%% meanr and dispersion
[d, m] = image_dispersion(image);

%% entropiya
ent = image_entropiya(image, histval);

%% chi2
chi2_orig = image_chi2(image);
chi2_enc = image_chi2(image_encoded);

%% ps

[dR_orig, dS_orig, N] = image_ps(image, 4);
[dR_enc, dS_enc] = image_ps(image_encoded, 4);

dR = abs(dR_enc - dR_orig);
dS = abs(dS_enc - dS_orig);

end
```