

УДК 004.932.2

DOI: [10.26102/2310-6018/2020.29.2.006](https://doi.org/10.26102/2310-6018/2020.29.2.006)

Алгоритм выявления стеганографических вставок типа LSB-замещения на основе метода анализа иерархий

А.К. Гуц, Д.Э. Вильховский

*Омский государственный университет им. Ф.М. Достоевского
Омск, Россия*

Резюме: В статье предлагается алгоритм выявления стеганографических вставок, осуществляемых в виде замены наименее значащих битов. Предлагаемый алгоритм базируется на методе анализа иерархий. Рассматриваются слои младших битов синей компоненты. Области встраивания определяются при помощи алгоритма таксономии. Применяются алгоритм предварительной обработки для того чтобы повысить эффективности на участках, которые содержат градиентную заливку. Научная новизна заключается в разработке алгоритма стеганографического анализа метода LSB-замены при низком заполнении стегаконтейнера, основанного на сравнительном анализе нескольких слоев изображения с помощью метода анализа иерархий, отличающегося тем, что выделенные критерии принятия решения представляют возможность учитывать структуру исходного изображения-контейнера, которая хранится в более высоких битовых слоях и за счет этого представляется возможным сформировать карту подозрительных пикселей, повышающую эффективность обнаружения встроенного сообщения. Был выполнен эксперимент на ЭВМ. Для искусственных изображений с градиентной и равномерной заливкой предлагаемый алгоритм дает возможность определять в среднем 91% подмененных битов, тогда как ложные срабатывания составляют не больше 1%. Положение встроенных битов может быть определено на основе сопоставления матрицы решений с начальным изображением. Предлагаемый алгоритм эффективен при малых размерах встроенного сообщения в отличие от созданных ранее алгоритмов.

Ключевые слова: выявление стеговставок, анализ изображений со вставками, анализ стегаконтейнера, поиск LSB-вставок, LSB встраивание

Для цитирования: Гуц А.К., Вильховский Д.Э. Алгоритм выявления стеганографических вставок типа LSB-замещения на основе метода анализа иерархий. *Моделирование, оптимизация и информационные технологии*. 2020;8(2). Доступно по: https://moit.vivt.ru/wp-content/uploads/2020/05/GutsVilkhovskiy_2_20_1.pdf DOI: 10.26102/2310-6018/2020.29.2.006

An algorithm for identifying steganographic inserts of the LSB-replacement type based on the hierarchy analysis

A.K. Guts, D.E. Vilkhovskiy

*Omsk State University F.M. Dostoevsky
Omsk, Russia*

Abstract: The article proposes an algorithm for identifying steganographic inserts implemented as a replacement of the least significant bits. The proposed algorithm is based on the hierarchy analysis method. The layers of the least significant bits of the blue component are considered. The embedding areas are determined using the taxonomy algorithm. A preprocessing algorithm is applied in order to increase efficiency in areas that contain gradient fill. The scientific novelty lies in the development of an algorithm for steganographic analysis of the LSB replacement method with low filling of the container, based on a comparative analysis of several image layers using the hierarchy analysis method, characterized in that the selected decision criteria provide the opportunity to take into account the structure of the original container image that is stored in higher bit layers and due to this it is possible to

form a map of suspicious pixels, increasing efficiency embedded messages. A computer experiment was performed. For artificial images with gradient and uniform fill, the proposed algorithm makes it possible to determine on average 91% of the replaced bits, while false positives are no more than 1%. The position of the embedded bits can be determined by matching the decision matrix with the initial image. The proposed algorithm is effective for the small size of the embedded message, in contrast to the previously created algorithms.

Keywords: identification of stego inserts, analysis of images with inserts, analysis of stegocontainer, search for LSB inserts, embedding LSB.

For citation: Guts A.K., Vilkhovskiy D.E. An algorithm for identifying steganographic inserts of the LSB-replacement type based on the hierarchy analysis *Modeling, Optimization and Information Technology*. 2020;8(2). Available from: https://moit.vivt.ru/wp-content/uploads/2020/05/GutsVilkhovskiy_2_20_1.pdf DOI: 10.26102/2310-6018/2020.29.2.006 (In Russ).

Введение

Замена наименее значимых бит (LSB-замещение) на сегодняшний день является самым известным методом встраивания стеганографических вставок (далее по тексту LSB-вставки) [1]. Этот метод базируется на том, что замена от одного до четырёх младших бит в пикселях синей компоненты изображения для человеческих глаз в целом будет незаметной. Выбор синей компоненты обусловлен строением сетчатки глаза. В настоящее время создано большое количество алгоритмов, которые встраивают информацию в изображения, но метод LSB-замещения стал первым в истории и получил широкое распространение.

На данный момент времени все имеющиеся алгоритмы направлены на выявление факта наличия или отсутствия LSB-вставки в изображении.

Работы [2,3] приводят метод Хи-квадрат относящийся к статистическому анализу. Данный метод основан на предположении, что распределение младших битов цветовых байтов изображения носит случайный характер. Метод Хи-квадрат показал весьма неплохие результаты в случае равномерного заполнения контейнера, однако он достаточно слабо себя показал для замены младших битов при случайном выборе пикселей. Работа [4] приводит метод визуального выявления наличия LSB-вставки посредством сопоставления цветовых срезов изображений. Указанный метод выдал относительно неплохие результаты в случае наличия больших полей сплошной заливки на изображении.

Основой стеганографического анализа в работах [5-7] является сопоставление в соседних байтах младших битов с применением формализма цепей Маркова. В работе [8] предлагается метод выявления LSB-вставки на базе искусственной нейронной сети. Нейронная сеть в случае большого объема обучающей выборки может выявить наличие вставки с ошибкой не менее 15%.

Самой сложной является задача определения LSB вставок при случайном распределении встраиваемых битов в изображении. Существующие в настоящее время методы показывают свою эффективность, если стеганографический контейнер заполнен хотя бы на половину [3]. В работе [9] предлагается метод выявления встроеной информации, основанный на алгоритмах сжатия. Суть метода заключается в том, что случайные данные будут слабее сжиматься, в сравнении с упорядоченными. Такой подход дает возможность с высокой точностью выявлять наличие LSB-вставки, в случае если контейнер заполнен от 40%. В работе [10] предложен метод, содержащий модуль предварительной обработки изображения, что дало возможность применять его, когда контейнер имеет существенно меньшее заполнение.

Нужно подчеркнуть, что в настоящее время не существует алгоритмов, которые отвечают на вопрос о том, была ли выполнена подмена младшего бита пикселя. Такая задача по своей постановке приближается к проблеме определения пикселей, которые были повреждены импульсным шумом.

Импульсный шум можно охарактеризовать изменением цвета случайно выбранного байта на произвольно выбранную величину. Но задача, связанная с поиском встроенного сообщения осложняется тем, что величина изменения равна одному биту. В настоящее время есть несколько методов выявления поврежденных пикселей. Метод SD-ROM [11] нужно отметить в первую очередь, так как он был первым и стал фундаментом для ряда алгоритмов [12,13].

В статьях [14,15] предлагается алгоритм поиска битов нулевого слоя изображения, которые изменяются в процессе LSB-вставки, на базе анализа ближайших соседей. Авторы применяют двухмерную модель Изинга, массово используемую для моделирования спиновых систем в статистической физике. На базе анализа статистической суммы, устанавливается вероятность текущего значения каждого бита. Когда количество битов, у которых низкая вероятность состояния превышает определенное критическое значение, то можно сделать вывод о том, что присутствует LSB-вставка. Авторами, несмотря на попиксельный анализ изображения, предлагается только метод анализа изображения в целом на наличие LSB-вставки. Нужно выделить ограничение данного алгоритма на объем встраиваемого сообщения – он не должен быть слишком маленьким и не должен быть слишком большим. Данный алгоритм имеет большой процент ложных срабатываний на изображениях без LSB-вставок.

В работах [16-18] LSB-вставки рассматриваются как высокочастотный аддитивный шум. Для определения замещенных битов применяются скользящие окна с размером 7x7, 5x5 и 3x3 пикселей нулевого слоя. Выполняется анализ пикселей, которые окружают тестируемый пиксель. На их основе выстраивается прогнозируемое значение тестируемого бита. Изменённые биты распознаются, основываясь на разнице между прогнозным образом изображения и его исходным образом. Для прогнозирования и распознавания заменённых битов применяются случайные Марковские поля или медианный фильтр.

В данной статье предложен алгоритм определения пикселей изображения, в которых была осуществлена замена наименее значащего бита при стеганографическом встраивании сообщения, на базе метода анализа иерархий [19].

Постановка задачи

Проанализируем изображения, в которых в виде LSB-вставок может быть встроена информация в младший бит синей компоненты. Выбор синей компоненты обусловлен тем, что встраивание в неё визуально малозаметно, из-за чего для сокрытия сообщений рекомендуется использовать именно её. Аналогично можно проанализировать и иные компоненты без утраты общности используемого метода.

Сформируем 2-а предположения. Первое сформулируем так: нам доподлинно неизвестно присутствует ли LSB-вставка или нет. Второе предположение: нам предварительно неизвестно ни геометрическое положение встроенных битов на изображении, ни их количество. Задачей является выявление присутствия LSB-вставки и установление наибольшего количества пикселей, в которых заменен младший бит синей компоненты.

Наличие второго предположения значительно усложняет задачу, потому что может возникнуть ситуация, при которой все младшие пиксели синей компоненты будут заменены. Анализ нулевого слоя изображения в таком случае не даст полезной

информации. Из-за чего нужно проанализировать более высокие слои. Нужно исходить из того, что ключевые закономерности изображения плавно изменяются от одного слоя к другому. В связи с чем, закономерности, установленные в одном слое должны повториться с высокой вероятностью в близлежащих слоях.

Будем искать пиксели, в которых осуществлена замена нулевого бита, анализируя отдельно нулевой слой и ближайшие к нему три слоя. В последующем создадим схему сравнения результатов этих двух алгоритмов и схемы для принятия общего решения.

Зададим k -ый слой синей компоненты исследуемого изображения в виде бинарной матрицы цветов $B_{ij}^{(k)}$. Пусть координаты встраиваемой информации определены в виде матрицы R_{ij} . Если происходит замена младшего бита синей компоненты соответствующего пикселя, то $R_{ij} = 1$. Если не происходит замены, то $R_{ij} = 0$. В результате встраивания LSB-вставки сформируется матрица $A_{ij}^{(0)}$ взамен нулевого слоя $B_{ij}^{(0)}$. Задача сводится к максимально точному восстановлению матрицы R_{ij} из анализа матриц $A_{ij}^{(0)}$, $B_{ij}^{(1)}$, $B_{ij}^{(2)}$, $B_{ij}^{(3)}$.

Использование метода анализа иерархий для поиска LSB-вставок

Для того, чтобы принять решение о замене бита, будем использовать метод анализа иерархий [14]. Для этого необходимо сформулировать альтернативные решения, из которых осуществляется выбор, в том числе критерии для проведения анализа альтернатив. Как уже было сказано в постановке задачи, требуется определить пиксели, в которых произошла замена младшего бита. В связи с этим, существует лишь одно из 2-х решений, которое обозначим в последующем Y , если в данном пикселе выполнена замена младшего бита, или N , если пиксель не менялся.

Сформируем систему определения подмененных битов на основе анализа нулевого слоя. Для этого необходимо выполнить последовательно проход по всем битам нулевого слоя и проанализировать ближайших соседей каждого из них.

Предложим критерии:

K_1 – соседние по сторонам биты имеют такое же значение, что и анализируемый бит или оно отличается от него.

K_2 – соседние по углам биты имеют такое же значение, что и анализируемый бит или отличаются от него.

K_3 – отклонение значения бита от среднего значения окружающих восьми битов.

K_1 , K_2 дают возможность определять протяженные области изображения одного цвета. K_3 предназначен для определения областей с градиентной заливкой. В результате получается иерархическое двухуровневое дерево альтернатив, Рисунок 1. Окончательное решение обозначим – R .

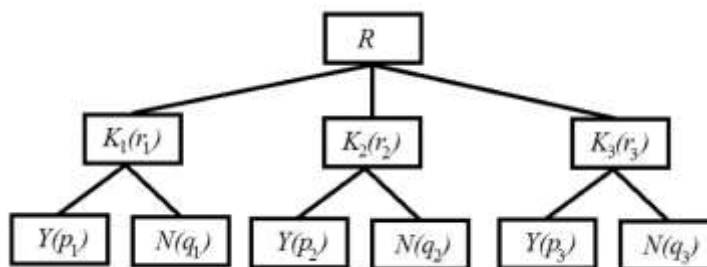


Рисунок 1 – Иерархия критериев для выявления замены бита из анализа нулевого слоя.
 Figure 1 – The hierarchy of criteria for identifying a replacement of the analysis of the zero bits layer.

Для того чтобы применить схемы анализа иерархий нужно установить относительные веса критериев r_i ($i = 1,2,3$), в том числе веса решений в рамках одного критерия p_i и q_i ($i = 1,2,3$). Примем, что критерий K_1 значимее критерия K_2 в n раз, а критерий K_2 значимее критерия K_3 в k раз. Предположим также о наличии транзитивности, т.е. критерий K_1 значимее критерия K_3 в nk раз. В таком случае согласованная матрица парных сравнений:

	K_1	K_2	K_3
K_1	1	n	kn
K_2	$1/n$	1	k
K_3	$1/(kn)$	$1/k$	1

Из данной матрицы с помощью стандартных способов [12] можно получить весовые коэффициенты:

$$r_1 = \frac{nk}{nk+k+1}, r_2 = \frac{k}{nk+k+1}, r_3 = \frac{1}{nk+k+1} \quad (1)$$

Парные сравнения в случае классического использования метода анализа иерархий устанавливаются на основе экспертных оценок. Применим вместо экспертных оценок определенные объективные показатели, которые можно определить численно. Например, ограничения на значения n и k в последующем мы установим из рассмотрения тривиальных примеров. Самые подходящие значения таких характеристик, как и l , будут найдены из эксперимента на ЭВМ.

Теперь следует перейти к определению весовых коэффициентов в рамках каждого из критериев. Начнём с критерия K_1 . Пусть из четырех битов, которые соприкасаются с рассматриваемым x имеют такое же значение, получается, что решение N более весомо в сравнении с Y (т.е. рассматриваемый бит не заменён) в $x/(4-x)$ раз. Записывая матрицу парных сравнений и выполняя соответствующие преобразования получим значения коэффициенты $p_1 = (4-x)/4$, $q_1 = x/4$. Для критерия K_2 всё будет аналогично. Пусть из четырех битов, соприкасающихся с данным только по вершинам, u обладают таким же значением. В таком случае весовые коэффициенты будут иметь значения $p_2 = (4-u)/4$, $q_2 = u/4$.

Для того, чтобы вычислить весовые коэффициенты по критерию K_3 , будем исходить из предположения, что значение изучаемого бита c , а окружающие его биты имеют среднее значение c_0 . Для определения весовых коэффициентов будем рассуждать следующим образом. Пусть решение N более весомо, чем Y в a раз, при этом величина a находится в зависимости от абсолютного значения отклонения значения бита c от среднего значения окружающих битов c_0 ($dc = |c - c_0|$). Весовые коэффициенты:

$$p_3 = \frac{1}{a+1}, q_3 = \frac{a}{a+1} \quad (2)$$

Теперь перейдём к рассмотрению предельных случаев. В случае, когда значение рассматриваемого бита равно среднему значению окружающих битов ($dc = 0$) будем

считать, что он не заменён, коэффициенты при этом будут иметь значение $p_3 = 0, q_3 = 1$. Когда бит от окружающих отличается максимально ($dc = 1$), то будем считать его однозначно заменённым, т.е. $p_3 = 1, q_3 = 0$. Значит при $dc = 0$ должно быть $a \rightarrow \infty$. Если значение $dc = 1$ то $a = 0$.

Указанным условиям будет удовлетворять выражение: $a = \frac{1}{dc} - 1$

Значения для весовых коэффициентов в таком случае: $p_3 = dc, q_3 = 1 - dc$

Чтобы окончательно принять решение нужно рассчитать величины:

$$P(Y) = r_1p_1 + r_2p_2 + r_3p_3 \quad (3)$$

$$P(N) = r_1q_1 + r_2q_2 + r_3q_3 \quad (4)$$

Когда $P(Y) > P(N)$, то принимается решение Y – бит подменённый, в ином случае $P(Y) \leq P(N)$, принимается решение N – бит не подменённый.

Расширим предлагаемый метод для анализ бита на основе сопоставления с 3-я вышележащими слоями. В каждом слое будем рассматривать бит, который лежит над данным и 8 его соседних битов. Такой набор битов назовём окном в соответствующем слое. Предлагаемые критерии принятия решений на основе анализа k -го слоя ($k = 1,2,3$):

$K_1^{(k)}$ – соседние по сторонам биты в окне k -го слоя обладают таким же значением, что и рассматриваемый бит нулевого слоя или значение которое отличается от него.

$K_2^{(k)}$ – соседние по углам биты в окне k -го слоя обладают таким же значением, что и рассматриваемый бит нулевого слоя или значение которое отличается от него.

$K_3^{(k)}$ – отклонение значения бита в нулевом слое от среднего значения битов окна в k -ом слое.

На Рисунке 2 отображено трёхуровневое иерархическое дерево альтернатив. Через R обозначим окончательное решение.

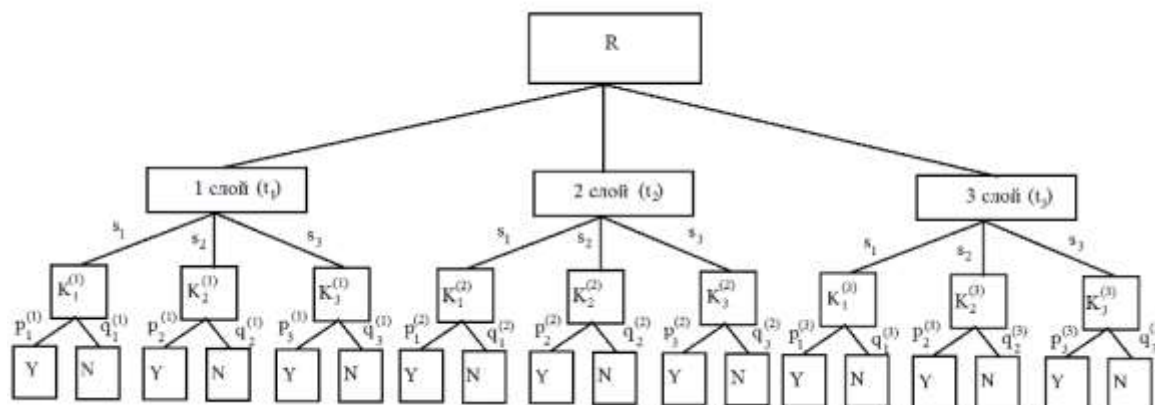


Рисунок 2 – Иерархия критериев для выявления замены бита из анализа вышележащих слоев.
 Figure 2 – A hierarchy of criteria for identifying bit replacements from an analysis of overlying layers.

Примем, что результаты анализа первого слоя значимее результатов второго в 2 раза, и второй слой значимее третьего также в 2 раза. Значения весовых коэффициентов: $t_1 = 4/7, t_2 = 2/7, t_3 = 1/7$.

Сложно в пределах одного слоя выделить какой-либо из критериев, поэтому предположим, что все они равнозначны: $s_1 = s_2 = s_3 = 1/3$.

В рамках одного критерия для весовых коэффициентов в случае 2-х решений воспользуемся таким же подходом, как и при анализе нулевого слоя. Для первого критерия:

$$p_1^{(k)} = (4 - x^{(k)})/4, q_1^{(k)} = x^{(k)}/4. \quad (5)$$

где $x^{(k)}$ – число соседей по бокам, обладающих таким же значением в окне k -слоя. Для 2-го критерия:

$$p_2^{(k)} = (4 - y^{(k)})/4, q_2^{(k)} = y^{(k)}/4. \quad (6)$$

где $y^{(k)}$ – число соседей по диагонали, обладающих таким же значением в окне k -слоя. Весовые коэффициенты 3-го критерия:

$$p_3^{(k)} = dc^{(k)}, q_3^{(k)} = 1 - dc^{(k)}, \quad (7)$$

где $dc^{(k)}$ – отличие значения бита от среднего значения битов окна в k -ом слое.

Формально алгоритм, в предлагаемом методе, заключается в проходе по всем пикселям изображения и в принятии решения для каждого пикселя о том, заменён ли его младший бит. Решение должно приниматься на основе анализа битов данного пикселя и его ближайших соседей. На выходе алгоритма формируется перечень пикселей, в которых была произведена замена младших битов на биты встраиваемого сообщения. В связи с тем, что в алгоритме реализуется один проход по всем пикселям, то и для каждого пикселя производится определённое число шагов, поэтому трудоёмкость алгоритма будет линейной. Нужно подчеркнуть, что локализация данных, требуемая для того чтобы принять решение, в малой области вокруг рассматриваемого пикселя, даёт возможность относительно просто выполнить распараллеливание алгоритма простым разделением изображения на области.

Алгоритм предварительной обработки изображений

Эксперимент на ЭВМ показал, что непосредственное использование предлагаемого алгоритма на основе метода анализа иерархий сталкивается с проблемами уже на изображениях, которые представляют собой область градиентной заливки. Это объясняется тем, что даже самые небольшие перемены цвета существенно меняют младшие биты изображения. К примеру, для градиентной заливки слой нулевых битов (нулевой слой) – полосы из единиц и нулей, а это отрицательно влияет на работу алгоритма принятия решений. Поэтому, необходимо выполнить предварительную обработку изображения. Для этого воспользуемся линейным преобразованием:

$$d(x, y) = ax + by - e, \quad (8)$$

где $a = c(x + 1, y) - c(x, y)$, $b = c(x, y + 1) - c(x, y)$, $c(x, y)$ – значения цвета пикселя, который расположен в точке с координатами (x, y) . Определим e как наименьшее значение $c(x, y)$ на множестве всех пикселей изображения. К множеству $d(x, y)$ применим алгоритм поиска пикселей с подменёнными младшими битами. Эксперимент на ЭВМ для искусственных изображений, которые представляют собой прямоугольник с градиентной заливкой, дал возможность определить заменённые младшие биты,

эффективность при этом составила $eff = 99,6\%$ с процентом ложных срабатываний $err = 0,1\%$ независимо от цвета заливки. Получается, что такое линейное преобразование убирает градиент цвета.

Перед тем, как применить предлагаемый алгоритм на фотографических изображениях, необходимо выполнить предварительную обработку этих изображений. Это позволит выявить области градиентной заливки и равномерной заливки. Эксперименты показали положительные результаты лишь на указанных областях изображения. Данный факт можно легко объяснить. Алгоритм поиска LSB-вставок основывается на анализе младших 3-х битов пикселей, которые входят в область малых размеров. В случае если на таком множестве, состоящем лишь из единиц и нулей присутствуют большие перепады цвета, то определить закономерности и соответствующие отклонения от них будет трудно.

Для использования данного линейного преобразования к фотографическим изображениям нужно установить области градиентной заливки. Определим вторые производные от функции $c(x, y)$ и области, в которых они имеют нулевое значение. Из-за того, что области идеальной градиентной заливки на фотографических изображениях встречаются достаточно редко, упростим выполнение 3-х условий:

$$\left| \frac{\partial^2 c(x,y)}{\partial x^2} \right| \leq 2, \quad \left| \frac{\partial^2 c(x,y)}{\partial y^2} \right| \leq 2, \quad \left| \frac{\partial^2 c(x,y)}{\partial x \partial y} \right| \leq 2. \quad (9)$$

Нестрогое неравенство взамен равенства нулю введено для учёта небольших отклонений от градиентной заливки и для того, чтобы не потерять встроенные биты.

После определения связанных областей, которые удовлетворяют условиям на вторые производные, нужно установить коэффициенты функции $d(x, y)$. Для их поиска воспользовались методом наименьших квадратов. Затем для каждой из выделенных областей определялось значение функции $d(x, y)$. После чего для поиска области встраивания $map(x, y)$ выполнялось построение карты:

1. Когда точка (x, y) находится в области градиентной заливки – $map(x, y) = d(x, y)$.
2. Для всех прочих точек $map(x, y) = 0$.

К карте $map(x, y)$ был применён алгоритм поиска пикселей с заменёнными младшими битами на основе метода анализа иерархий. Присваивание в точках, которые не принадлежат к областям градиентной заливки нулевых значений $map(x, y)$, даёт возможность уменьшить воздействие границ областей изображения на результаты работы алгоритма.

Компьютерный эксперимент и результаты

Эксперимент на ЭВМ выполнялся для разных цветных изображений, как фотографических, так и искусственных. Встраиваемое сообщение было представлено в виде текстовой строки, представленной в виде последовательности битов. Встраивание осуществлялось в синюю компоненту, потому что человеческий глаз может меньше всего её заметить. Встраивание выполнялось в область прямоугольной формы. В качестве задачи стеганографического анализа ставилось установление области, в которой располагаются заменённые биты. После того как алгоритм принятия решений по изображению сделает проход получится некоторая матрица R из единиц и нулей, размеры этой матрицы будут совпадать с размерами изображения. Нулевое значение элемента матрицы будет соответствовать отрицательному решению алгоритма о замене пикселя, положительному – единичное значение. Данную матрицу можно представить в

виде изображения с палитрой, которая состоит из 2-х цветов. Единичному значению будут соответствовать черные точки, нулевому значению – белые.

Изначально алгоритм проходил тестирование на искусственных изображениях, у которых равномерная заливка выполнена одним цветом. На Рисунке 3 представлены результаты работы алгоритма для искусственного изображения.

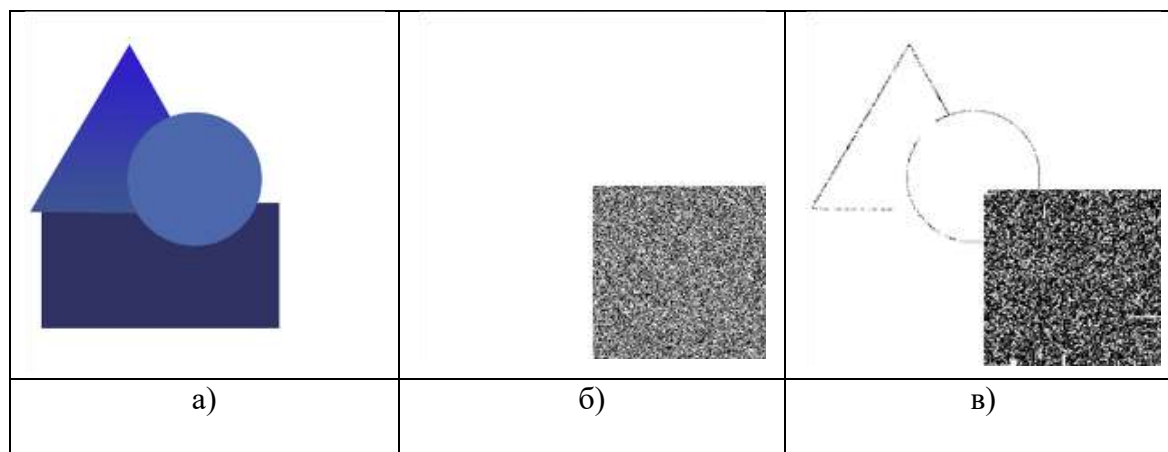


Рисунок 3 – Визуализация матрицы решений R для искусственного изображения: а) изображение со встроенной вставкой, б) карта измененных пикселей (пиксели, значение которых было изменено при встраивании, выделены черным цветом), в) визуализированная матрица решений R .

Figure 3 – Visualization of the decision matrix R for the artificial image: a) an image with an embedded insertion, b) a map of changed pixels (pixels whose value was changed during embedding are highlighted in black), c) a visualized decision matrix R .

Из Рисунка 3 видно, что в результате работы алгоритма на искусственном изображении область встраивания может быть визуально установлена довольно точно. На Рисунке 4 представлены результаты работы алгоритма для фотографического изображения, у которого резкие границы.

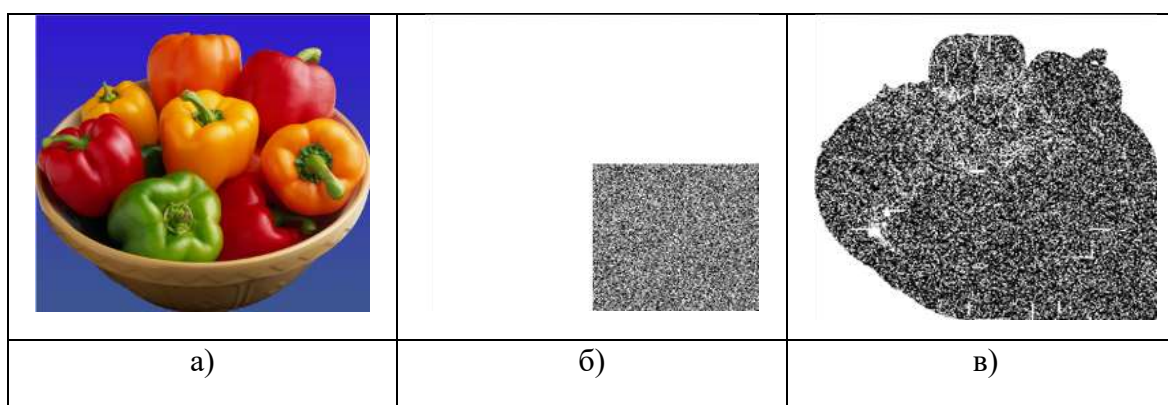


Рисунок 4 – Визуализация матрицы решений R для фотографического изображения: а) изображение со встроенной вставкой, б) карта изменённых пикселей (пиксели, значение которых было изменено при встраивании, выделены черным цветом), в) визуализированная матрица решений R .

Figure 4 – Visualization of the decision matrix R for a photographic image: a) an image with a built-in insert, b) a map of changed pixels (pixels whose value was changed during embedding are highlighted in black), c) a visualized decision matrix R .

Из Рисунка 4 видно, что предлагаемый алгоритм даёт возможность определять изменённые пиксели в области градиентной или равномерной заливки, но для областей, которые содержат мелкие детали, будут происходить ложные срабатывания. Такое ошибочное срабатывание алгоритма заключается в том, что алгоритм предварительной обработки не может устранить быстрые изменения цвета и резкие границы. Но сопоставление Рисунков 4а и 4в даёт возможность сделать выводы о наличии в изображении встроенного сообщения и о том, что оно расположено в правом нижнем углу.

Обсуждение результатов и выводы

Предлагаемый алгоритм определения LSB-вставок даёт возможность с высокой вероятностью выявить наличие сообщения, которое встроено методом замены наименее значащих битов. В случае искусственных изображений с градиентной и равномерной заливкой предлагаемый алгоритм даёт возможность в среднем 91% заменённых битов, а ложных срабатываний будет не больше 1%. Визуализация матрицы решений при этом даёт возможность с высокой точностью установить размеры и положение области встраивания сообщения.

Предлагаемый алгоритм тестировался на коллекции фотографических изображений «The Berkeley Segmentation Dataset and Benchmark» предложенной в работе [23]. Коллекция пополняется и в настоящее время содержит несколько сотен изображений. Первоначально коллекция создавалась для тестирования алгоритмов сегментации изображений. Целесообразность применения её для предлагаемого алгоритма заключается в том, что она содержит в себя изображения, имеющие сегменты разного размера и различных типов заливки.

Для фотографических изображений, предлагаемый алгоритм, правильно выделяет в среднем 89%, пикселей с заменённым младшим битом, а ложных срабатываний при этом в среднем будет 37%. Положение встроенных битов может быть установлено на основе сопоставления матрицы решений с начальным изображением.

Предлагаемый алгоритм, в отличие от созданных ранее, проявляет свою эффективность при малых размерах встроенного сообщения. Если прочие методы, которые были описаны во введении, более эффективны при заполнении контейнера более чем на 50%, то предлагаемый алгоритм выдаёт наилучшие результаты при заполнении стеганографического контейнера в 10-30%.

Предлагаемый алгоритм, в отличие от работ [18-19], проводит анализ не только нулевого слоя битов, но и 2-х ближайших к нему слоя, что даёт возможность установить дополнительные закономерности на изображении. С привлечением 2-х дополнительных слоёв понижается процент ложных срабатываний и увеличивается эффективность выявления LSB-вставок.

Эффективность алгоритма на основе прогнозных значений [20] доходит до 84%. Как было указано ранее, предлагаемый алгоритм имеет эффективность до 89%.

ЛИТЕРАТУРА

1. Adelson E. *Digital Signal Encoding and Decoding Apparatus*. U.S. Patent. 1990. N. 4,939515.
2. Westfeld A., Pfitzmann A. *Attacks on Steganographic Systems: Breaking the Steganographic Utilities EzStego, Jsteg, Steganos and Stools and Some Lessons Learned*. 3rd International Workshop on Information Hiding. 2000: 61-76.
3. Provos N., Honeyman P. *Detecting steganographic content on the internet*. Technical Report CITI 01-1a, University of Michigan, 2001.

4. Алиев А.Т. *О применении стеганографического метода LSB к графическим файлам с большими областями монотонной заливки*. Вестник ДГТУ. 2004;4(22):454-460.
5. Барсуков В.С. Романцов А.П. *Оценка уровня скрытности мультимедийных стеганографических каналов хранения и передачи информации.*: Специальная Техника. 2000:1.
6. Кустов В.Н., Параскевопуло А.Ю. *Простые тайны стегоанализа*. Защита информации, INSIDE. 2005;4:72-78.
7. Голуб В.А., Дрюченко М.А. *Комплексный подход для выявления стеганографического скрывтия в JPEG-файлах*. Инфокоммуникационные технологии. 2009;7(1):44-50.
8. Абденюв А.Ж., Леонов Л.С. *Использование нейронных сетей в слепых методах обнаружения встроенной стеганографической информации в цифровых изображениях*. Ползуновский Вестник. 2010;2:221-225.
9. Жилкин М.Ю. *Стегоанализ графических данных в различных форматах*. Доклады ТУСУРа. 2008;2(18):63-64.
10. Монарев В. А. *Сдвиговой метод обнаружения скрытой информации*. Вестник СибГУТИ. 2012;4:62-68.
11. Abreu E., Lightstone M., Mitra S.K., Arakawa S.K. *A new efficient approach for the removal of impulse noise from highly corrupted images*. IEEE Transactions on Image Processing. 1996;5:1012-1025.
12. Garnett R., Huegerich T., Chui C., He W. *A Universal Noise Removal Algorithm with an Impulse Detector*. IEEE Trans Image Process, 2005;14(11):1747-1754.
13. Сорокин С.В., Щербаков М.А. *Реализация SD-ROM фильтра на основе концепции нечеткой логики*. Известия высших учебных заведений. Поволжский регион. 2007;3:56-65.
14. Amritha P.P., Sreedivya Muraleedharan M., Rajeev K. and Sethumadhavan M. *Steganalysis of LSB Using Energy Function*. Advances in Intelligent Systems and Computing. 2016;384:549-558
15. Debasis Mazumdar, Apurba Das, and Sankar K. Pal *MRF Based LSB Steganalysis: A New Measure of Steganography Capacity*. S. Chaudhury et al. (Eds.): PReMI 2009, LNCS 5909. 2009:420–425.
16. Yun Q. Shi, Patchara Sutthiwan, and Licong Chen *Textural Features for Steganalysis*. M. Kirchner and D. Ghosal (Eds.): IH 2012, LNCS 7692. 2013:63–77.
17. Ojala T., Pietikainen M., Maenpaa T. *Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local binary Patterns*. IEEE Transactions on Pattern Analysis and Machine Intelligence 24. 2002:971–987.
18. Ojala T., Pietikainen M., Harwood D *A Comparative Study of Texture Measures with Classification Based on Feature Distributions.*: Pattern Recognition 29. 1996:51–59.
19. Saaty T.L. *Relative Measurement and its Generalization in Decision Making: Why Pairwise Comparisons are Central in Mathematics for the Measurement of Intangible Factors - The Analytic Hierarchy/Network Process*. Review of the Royal Spanish Academy of Sciences, Series A, Mathematics. 2008;102(2):251–318.

REFERENCES

1. Adelson E. *Digital Signal Encoding and Decoding Apparatus*. U.S. Patent. 1990. N. 4,939515.
2. Westfeld A., Pfitzmann A. *Attacks on Steganographic Systems: Breaking the*

- Steganographic Utilities EzStego, Jsteg, Steganos and Stools and Some Lessons Learned*. 3rd International Workshop on Information Hiding. 2000:61-76.
3. Provos N., Honeyman P. *Detecting steganographic content on the internet*. Technical Report CITI 01-1a, University of Michigan, 2001.
 4. Aliev A.T. *On application LSB steganographics method to digital images with the greater monochrome areas*. Vestnik DGTU. 2004;4(22):454-460.
 5. Barsukov V.S., Romancov A.P. *Assessment of the stealth level of multimedia steganographic channels for storing and transmitting information*. Specialnaya Tekhnika. 2000:1.
 6. Kustov V.N., Paraskevopulo A.Ju.. *Simple secrets of steganalysis*. Zashhita informacii, INSIDE. 2005;4:72-78.
 7. Golub V.A., Drjuchenko M.A. *Comprehensive approach for revealing steganographic concealment in JPEG files Comprehensive approach for revealing steganographic concealment in JPEG files*. Infokommunikacionnye tehnologii. 2009;7(1):44-50.
 8. Abdenov A.Zh., Leonov L.S. *The use of neural networks in blind methods for detecting embedded steganographic information in digital images*. Polzunovskij Vestnik. 2010;2:221-225.
 9. Zhilkin M.Ju. *Stegoanalysis of graphic data in various formats*. Doklady TUSURa. 2008;2(18):63-64.
 10. Monarev V. A. *Shift detection of hidden information*. Vestnik SibGUTI. 2012;4:62-68.
 11. Abreu E., Lightstone M., Mitra S.K., Arakawa S.K. *A new efficient approach for the removal of impulse noise from highly corrupted images*. IEEE Transactions on Image Processing. 1996;5:1012-1025.
 12. Garnett R., Huegerich T., Chui C., He W. *A Universal Noise Removal Algorithm with an Impulse Detector*. IEEE Trans Image Process, 2005;14(11):1747-1754.
 13. Sorokin S.V., Shherbakov M.A. *SD-ROM filter implementation based on fuzzy logic concept*. Izvestija vysshih uchebnyh zavedenij. Povolzhskij region. 2007;3:56-65.
 14. Amritha P.P., Sreedivya Muraleedharan M., Rajeev K. and Sethumadhavan M. *Steganalysis of LSB Using Energy Function*. Advances in Intelligent Systems and Computing. 2016;384:549-558.
 15. Debasis Mazumdar, Apurba Das, and Sankar K. Pal *MRF Based LSB Steganalysis: A New Measure of Steganography Capacity*. S. Chaudhury et al. (Eds.): PReMI 2009, LNCS 5909. 2009:420–425.
 16. Yun Q. Shi, Patchara Sutthiwan, and Licong Chen *Textural Features for Steganalysis*. M. Kirchner and D. Ghosal (Eds.): IH 2012, LNCS 7692. 2013:63–77.
 17. Ojala, T., Pietikainen, M., Maenpaa, T. *Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local binary Patterns*. IEEE Transactions on Pattern Analysis and Machine Intelligence 24. 2002:971–987.
 18. Ojala, T., Pietikainen, M., Harwood, D. *A Comparative Study of Texture Measures with Classification Based on Feature Distributions*. Pattern Recognition 29. 1996:51–59.
 19. Saaty T.L. *Relative Measurement and its Generalization in Decision Making: Why Pairwise Comparisons are Central in Mathematics for the Measurement of Intangible Factors - The Analytic Hierarchy/Network Process*. Review of the Royal Spanish Academy of Sciences, Series A, Mathematics. 2008;102(2):251–318.

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Гуц Александр Константинович, д-р физ.-мат.наук, профессор, декан факультета компьютерной безопасности (ФКН) в Омском государственном университете им. Ф.М. Достоевского (ОмГУ)
Омск, Российская Федерация
e-mail: aguts@mail.ru

Alexander K. Guts, Doctor of Physical and Mathematical Sciences, Professor, Dean of the Faculty of Computer Security (FCN) at Omsk State University. F.M. Dostoevsky (OmSU), Omsk, Russian Federation

Вильховский Данил Эдуардович, ассистент кафедры информационной безопасности ФКН в Омском государственном университете им. Ф.М. Достоевского (ОмГУ), Омск, Российская Федерация
e-mail: vilkhovskiy@gmail.com

Danil E. Vilkhovskiy, Assistant at the Department of Information Security of the FCS of Omsk State University, Omsk, Russian Federation