Алгоритм стегоанализа на основе метода анализа иерархий

С.В. Белим
sbelim@mail.ru

Д.Э. Вильховский
vilkhovskiy@gmail.com

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

Аннотация

В статье предложен метод выявления стеганографических вставок, формируемых с помощью подмены наименее значащего бита (LSB). Обнаружение осуществляется с помощью разбиения изображения на слои и анализа окружения каждого бита младшего слоя. Анализируются также биты расположенные над исследуемым битом в двух вышележащих слоях. Для принятия решения о подмене конкретного бита используется метод анализа иерархий. Весовые коэффициенты в рамках метода анализа иерархий формируются на основе значений самих битов. Проведен компьютерный эксперимент с встраиванием сообщения в ограниченную прямоугольную область изображения. Показана высокая эффективность предложенного метода.

Введение

Самым простым и самым распространенным методом встраивания стеганографических вставок является подмена наименее значащих бит (LSB-замещение) [1]. Основная идея метода состоит в замене от одного до четырех младших бит в байтах цветового представления пикселей изображения. Наименее заметной является замена в синей составляющей цвета, что связано с особенностями световосприятия человеческого глаза. Этот метод используется как самостоятельно, так и в качестве составной части более сложных методов. Не смотря на простоту алгоритма формирования стеганографической вставки, задача ее обнаружения без дополнительной информации является достаточно сложной. На сегодняшний день не существует методов, которые с полной достоверностью могут определить наличие и размеры стеганографической вставки в произвольном контейнере. Большинство методов носят статистический характер и основываются на предположении об изменении статистических свойств битов изображения при помещении в него встроенной информации. Известные на сегодняшний день методы эффективны при заполнении стегоконтейнера не менее чем на 50% [2]. В работе [3] обнаружение стеганографических вставок осуществляется из предположения об изменении корреляций между соседними пикселями. Метод, предложенный авторами, состоит в том, что рассматриваются ближайшие соседи каждого пикселя. Из анализа окружающих пикселей делается прогноз о значении центрального пикселя и сравнивается с его текущим значением. В работе [4] предложен алгоритм обнаружения стегановставок с использованием шаблонов для соседних пикселей. Построение шаблонов также основывается на предположении о сильной корреляции между пикселями исходного изображения. Корреляции между пикселями также использованы в статье [5] для построения

In: Sergey V. Belim, Nadezda F. Bogachenko (eds.): Proceedings of the Workshop on Data Analysis and Modelling (DAM 2016), Omsk, Russia, October 2016, published at http://ceur-ws.org

статистического метода обнаружения стеганографических вставок. Аналогичный статистический метод основанный на величине корреляции между соседними пикселями предложен в работе [6]. В работе [8] приводится обобщенный метод определения длины стеганографической вставки на основе объединения нескольких детекторов. Использование авторегрессивной модели для обнаружения скрытых сообщений, а также оценка их относительной длины предложено в [7]. Таким образом, на сегодняшний день основными задачами стегоанализа ставиться принципиальное обнаружение наличия скрытой вставки и, по возможности, определение ее длины. Целью данной статьи ставиться разработка алгоритма для принятия решения, является ли тот или иной бит подмененным. То есть не просто определения наличия стеганографической вставки, а, по возможности, ее определение.

1 Постановка задачи

Будем анализировать изображения, в которых может быть встроена информация в виде стеганографических вставок в младший бит синей компоненты. При этом будем исходить из двух предположений. Во-первых, будем считать, что достоверно неизвестно есть ли стеганографическая вставка или нет. Во-вторых, заранее неизвестно ни количество встроенных битов, ни их геометрическое положение на изображении. Задачей ставиться определение наличия стеганографической вставки и определение максимального количества пикселей, в которых подменен младший бит синей компоненты. Второе предположение существенно осложняет задачу, так как возможна ситуация, при которой заменены все младшие пиксели синей компоненты. В этом случае анализ нулевого слоя изображения не принесет никакой информации. При этом заранее неизвестно позволит ли анализ нулевого слоя сделать какие-либо выводы. В связи с этим необходим анализ более высоких слоев. Будем опираться на предположение о том, что основные закономерности изображения плавно меняются от одного слоя к другому. Поэтому закономерности, выявленные в одном слое должны с высокой вероятностью повторяться в близлежащих слоях.

Будем искать пиксели, в которых произведена подмена нулевого бита отдельно анализируя нулевой слой и ближайшие к нему три слоя. В дальнейшем построим схему сравнения результатов этих двух алгоритмов и принятия общего решения. Пусть k-ый слой синей компоненты исходного изображения задана в виде бинарной матрицы цветов B_{ij}^k, а координаты встраиваемой информации задаются в виде матрицы R_{ij}. При этом $R_{ij} = 1$, если происходит подмена младшего бита синей компоненты соответствующего пикселя и $R_{ij} = 0$, если подмены не происходит. В результате встраивания стеганографической вставки вместо нулевого слоя B_{ij}^0 сформируется матрица A_{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.

2 Применение метода анализа иерархий для выявления подмененных битов

Применим метод анализа иерархий [8] для принятия решения о подмене бита. Для этого необходимо сформулировать альтернативные решения, из которых осуществляется выбор, а также критерии для анализа альтернатив. Как уже было сказано в постановке задачи необходимо выявить пикселы, в которых произошла подмена младшего бита. Поэтому возможно только одно из двух решений, обозначаемых в дальнейшем либо , если в данном пикселе осуществлена подмена младшего бита, либо , если пиксель не изменялся.

Сначала построим систему выявления подмены битов на основе анализа нулевого слоя. Для этого осуществим последовательный проход по всем битам нулевого слоя и осуществим анализ ближайших соседей каждого из них. Выделим три критерия:

K_1 - соседние по сторонам биты имеют то же значение, что и анализируемый или отличное от него.

K_2 - соседние по углам биты имеют то же значение, что и анализируемый или отличное от него.

K_3 - отклонение значения бита от среднего значения окружающих восьми битов. Первые два критерия позволяют выявлять протяженные области изображения одного цвета. Третий критерий необходим для выявления областей с градиентной заливкой. Таким образом, получаем двухуровневое иерархическое дерево альтернатив, изображенное на рисунке 1.

Для применения метода анализа иерархий необходимо определить относительные веса критериев r_i $(i = 1, 2, 3)$, а также веса решений в рамках одного критерия p_i и q_i $(i = 1, 2, 3)$. Будем считать, что критерий K_1 важнее критерия K_2 в n раз, а критерий K_2 важнее критерия K_3 в k раз. Также будем предполагать наличие транзитивности, то есть критерий K_1 важнее критерия K_3 в nk раз. Тогда согласованная матрица парных сравнений будет иметь вид:

Из данной матрицы стандартными способами [8] могут быть получены весовые коэффициенты:

$r_1 = nk/(nk + k + 1), r_2 = k/(nk + k + 1), r_3 = 1/(nk + k + 1)$

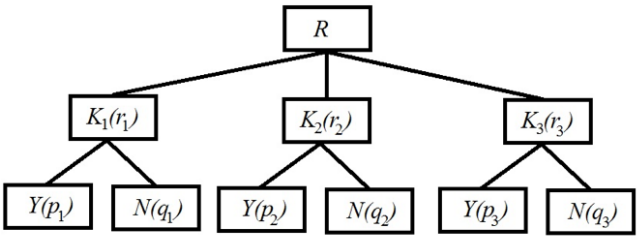


Рис. 1: Иерархия критериев для определения подмены бита из анализа нулевого слоя.

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

При классическом использовании метода анализа иерархий парные сравнения определяются на основе экспертных оценок. В нашем подходе вместо экспертных оценок будем использовать некоторые объективные показатели, определяемые численно. В частности, ограничения на значения n и k мы в дальнейшем определим из рассмотрения тривиальных примеров. Наиболее подходящие значения этих параметров найдено из компьютерного эксперимента. Перейдем к определению весовых коэффициентов в рамках каждого из критериев. Начнем рассмотрение с K_1. Пусть из четырех битов, соприкасающихся с исследуемым x имеют тот же значение, тогда решение N более весомо по сравнению с Y (то есть исследуемый бит не подменен) в $x/(4-x)$ раз. Записывая матрицу парных сравнений и производя необходимые преобразования получаем значения коэффициентов в $p_1 = (4-x)/4$, $q_1 = x/4$. Аналогично для критерия K_2. Пусть из четырех битов, соприкасающихся с данным, только по вершинам, имеют то же значение. Тогда весовые коэффициенты примут значения $p_2 = (4-y)/4$, $q_2 = y/4$. Для вычисления весовых коэффициентов по критерию K_3 предположим, что значение анализируемого бита c, а среднее значение окружающих его битов c_0. Для нахождения весовых коэффициентов применим следующие рассуждения. Пусть решение N более весомо, чем Y в a раз, где величина a зависит от абсолютного значения отклонения значения бита c от среднего значения окружающих битов c_0 ($dc = |c - c_0|$). Тогда весовые коэффициенты будут иметь вид: $p_3 = 1/(a+1)$,$q_3 = a/(a+1)$

Рассмотрим предельные случаи. В случае равенства значения исследуемого бита среднему значению окружающих битов ($dc = 0$) будем считать, что он не подменен, коэффициенты при этом будут иметь значение $p_3 = 0$, $p_3 = 0$. Если бит максимально отличается от окружающих ($dc = 1$), то будем считать его однозначно подмененным, то есть $p_3 = 1$, $q_3 = 0$. Следовательно, при $dc = 0$ должно быть $a \to \infty$. При значении $dc = 1$ должно выполняться $a = 0$. Этим условиям удовлетворяет выражение: $a = 1/dc - 1$.

Откуда следуют значения для весовых коэффициентов: $p_3 = dc$, $q_3 = 1 - dc$.

Для окончательного принятия решения необходимо вычислить величины: $P(Y) = r_1p_1 + r_2p_2 + r_3p_3$, $P(N) = r_1q_1 + r_2q_2 + r_3q_3$

Если $P(Y) > P(N)$, то принимается решение $R = Y$, то есть бит является подмененным, в противном случае, при $P(Y) \leq P(N)$, принимается решение $R = N$, то есть бит не является подмененным. Расширим предложенный метод на анализ бита на основе сравнения с тремя вышележащими слоями. Будем в каждом слое рассматривать бит, лежащий над данным, и восемь его ближайших соседей. В дальнейшем этот набор битов будем называть окном в соответствующем слое. Введем критерии принятия решений на основе анализа k-го слоя ($k = 1, 2, 3$):

${K_1}^k$ - соседние по сторонам биты в окне 1-го слоя имеют то же значение, что и анализируемый бит нулевого слоя или отличное от него.

${K_2}^k$ - соседние по углам биты в окне 2-го слоя имеют то же значение, что и анализируемый бит нулевого слоя или отличное от него.

${K_3}^k$ - отклонение значения бита в нулевом слое от среднего значения битов окна в 3-ом слое. Трехуровневое иерархическое дерево альтернатив, изображено на рисунке 2. Окончательное решение обозначим R_1.

Будем считать, что результаты анализа первого слоя важнее результатов второго в два раза, и второй слой важнее третьего также в два раза. Отсюда получаем значения весовых коэффициентов: $t_1 = 4/7$,

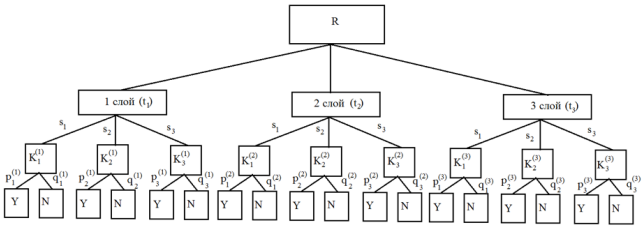


Рис. 2: Иерархия критериев для определения подмены бита из анализа вышележащих слоев

$t_2 = 2/7$, $t_3 = 1/7$.

В рамках одного слоя трудно выделить какой-то из критериев, поэтому положим, что все они равнозначны: $s_1 = s_2 = s_3 = 1/3$.

Для весовых коэффициентов для двух решений в рамках одного критерия применим подход аналогичный использованному при анализе нулевого слоя. Для первого критерия: ${p_1}^k = (4 - x^k)/4$, ${q_1}^k = (x^k)/4$, где x^k - количество соседей по бокам, имеющих то же значение в окне -слоя. Для второго критерия: ${p_2}^k = (4 - y^k)/4$, ${q_2}^k = y^k/4$, где y^k - количество соседей по диагонали, имеющих то же значение в окне -слоя. Весовые коэффициенты третьего критерия: ${p_3}^k = dc^k$, ${q_3}^k = 1 - dc^k$, где dc^k - отличие значения бита от среднего значения битов окна в k-м слое.

Для принятия решения необходимо вычислить величины: $P_1(Y) = t_1(s_1{p_1}^1 + s_2{p_2}^1 + s_3{p_3}^1) + t_2(s_1{p_1}^2 + s_2{p_2}^2 + s_3{p_3}^2) + t_3(s_1{p_1}^3 + s_2{p_2}^3 + s_3{p_3}^3)$, $P_1(N) = t_1(s_1{q_1}^1 + s_2{q_2}^1 + s_3{q_3}^1) + t_2(s_1{q_1}^2 + s_2{q_2}^2 + s_3{q_3}^2) + t_3(s_1{q_1}^3 + s_2{q_2}^3 + s_3{q_3}^3)$

Если $P_1(Y) > P_1(N)$, то принимается решение $R_1 = Y$, то есть бит является подмененным, в противном случае, при $P_1(Y) \leq P_1(N)$, принимается решение $R_1 = N$, то есть бит не является подмененным.

3 Алгоритм выявления подмененных пикселей

Запишем формально алгоритм, реализующий предложенный метод. Осуществляем последовательный проход по всем пикселям изображения. Для каждого пикселя выполняем последовательность шагов:

Шаг 1. Выделяем окна размером 3×3 в нулевом, первом, втором и третьем слоях.

Шаг 2. Вычисляем величины $P(Y)$, $P(N)$, $P_1(Y)$, $P_2(N)$.

Шаг 3. Если выполняется хотя бы одно из двух равенств $R = Y$ или $R_1 = Y$, то бит считается подмененным. Заносим в соответствующий элемент матрицы R_ij единичное значение, в противном случае нулевое.

На выходе алгоритма будет получена матрица подмененных пикселей R_{ij}. Так как в алгоритме осуществляется один проход по всем пикселям и для каждого пикселя выполняется фиксированное количество шагов, то трудоемкость алгоритма будет линейной. Также следует отметить локализацию данных, необходимых для принятия решения, в малой области вокруг исследуемого пикселя, что позволяет легко производить распараллеливание алгоритма простым разбиением изображения на области.

4 Компьютерный эксперимент и результаты

Для исследования эффективности предложенного метода был проведен компьютерный эксперимент по выявлению встроенной информации. Исследования проводились на трех типах изображений: градиентной заливке, искусственном изображении геометрических фигур и широко используемом изображении "Lena". Все изображения имели размер 256×256 пикселей, глубина цвета составляла 256 цветов. Встраивался текст на русском языке в виде побитовой последовательности в прямоугольную область расположенную случайным образом в центре изображения. Поменялось 9% исходного нулевого слоя. На рисунке 3 представлены результаты эксперимента с градиентной заливкой.

Как хорошо видно из сравнения рисунков 3б и 3г явно виден прямоугольник, в который производилось встраивание. Аналогичные результаты для искусственного изображения с геометрическими фигурами приведено на рисунке 4 и для фотографического изображения на рисунке 5.

На обоих рисунках явно видна область, в которую встраивалось скрываемое сообщение.

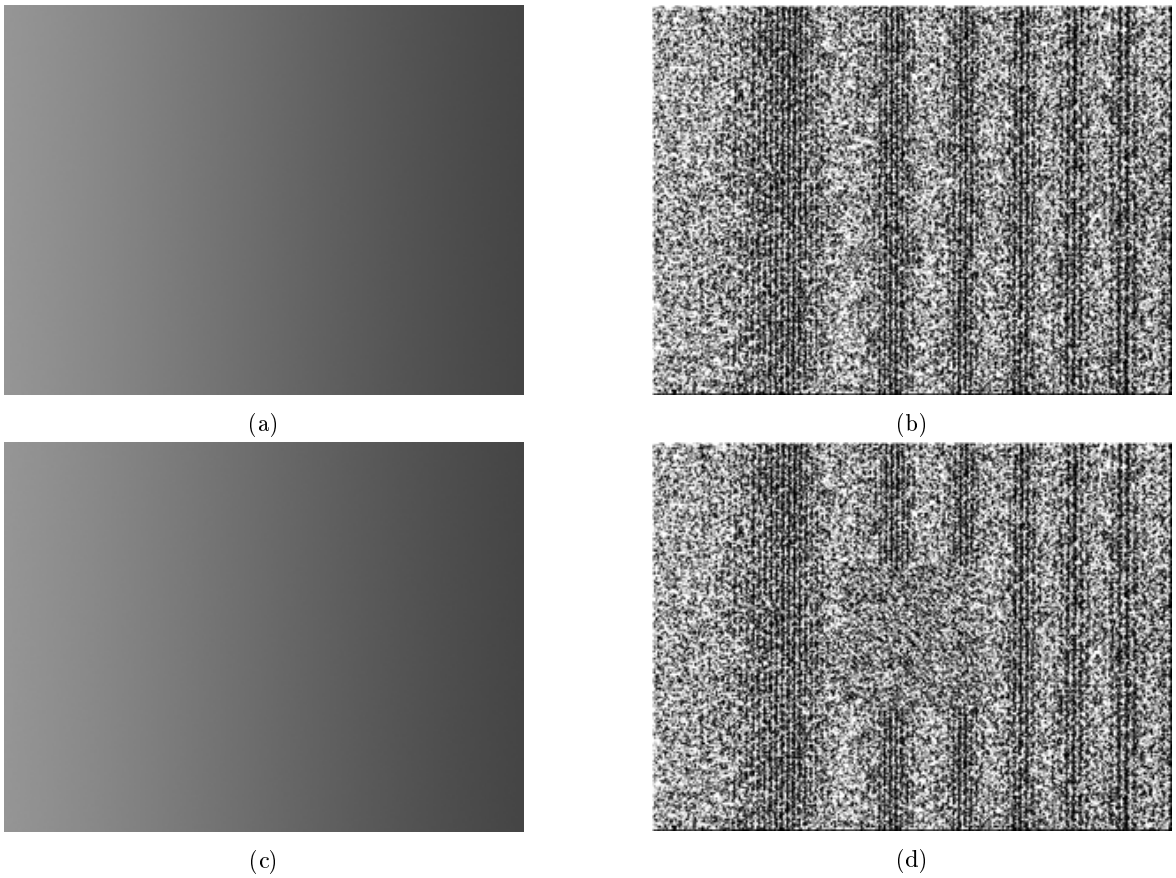


Рис. 3: Результат работы алгоритма по выявлению подмененных пикселей на градиентной заливке: a) исходное изображение, b) матрица для исходного изображения, c) изображение с встроенным сообщением, d) матрица для изображения со стегановставкой.

5 Обсуждение результатов и выводы

Таким образом, предложенный в данной статье алгоритм позволяет не только выявлять наличие стеганографической вставки в изображения, но и с достаточно высокой точностью определять ее местонахождение и объем. В отличие от распространенных на данный момент алгоритмов в предложенном методе не используется статистический подход. Следует отметить, что исследованные в рамках компьютерного эксперимента изображения со стеганографической вставкой легко проходят тест Хи-квадрат, который не обнаруживает в них встроенных сообщений. Данный метод требует дополнительных исследований, для разработки алгоритмов анализа полученной матрицы.

Список литературы

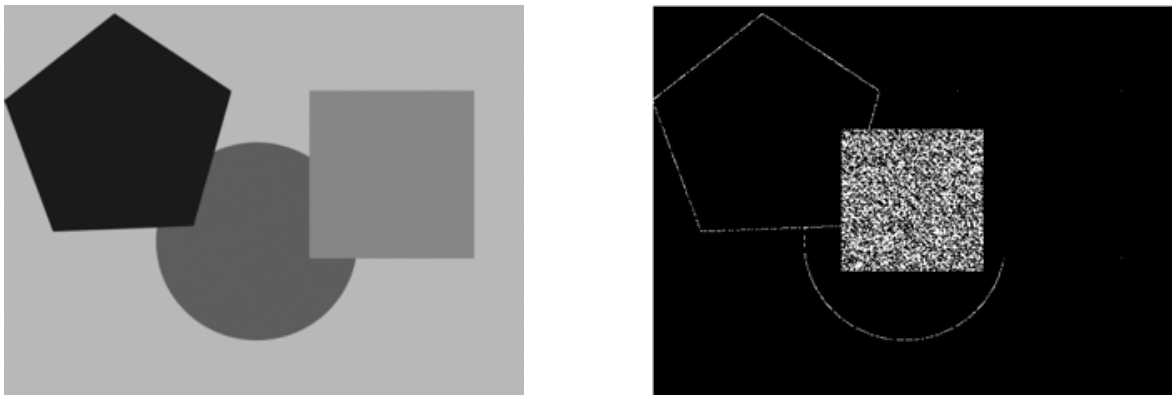
[1] E. Adelson. *Digital Signal Encoding and Decoding Apparatus.* U.S. Patent. 1990, N. 4,939,515.

[2] A. Westfeld,A. Pfitzmann. Attacks on Steganographic Systems. Breaking the Steganographic Utilities EzStego, Jsteg, Steganos and S-Tools and Some Lessons Learned. *Lecture Notes in Computer Science*, 1768:61–75, 2000.

[3] J. Zhang,F. Xiong. Steganalysis for LSB Matching Based on the Dependences Between Neighboring Pixels. *Journal of multimedia*, V.7,N.5:380–385, 2012.

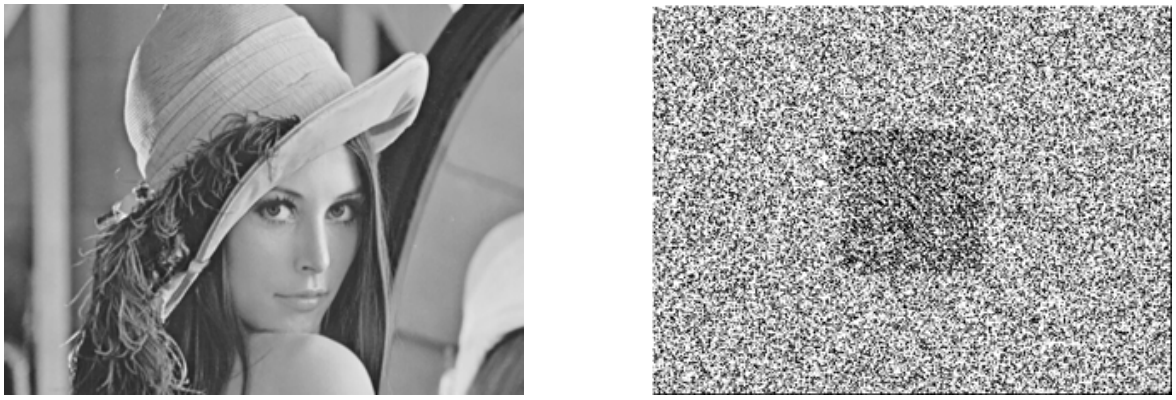
[4] D. Lerch-Hostalot,D. Meg?as. LSB Matching Steganalysis Based on Patterns of Pixel Differences and Random Embedding. *Computers and Security*, V.32:192–206, 2013.

[5] Zh. Xia,X. Wang,X. Sun,B. Wang Steganalysis of least significant bit matching using multi-order differences. *Security and Communication Networks*, V.7,N.8:1283–1291, 2014.



(a) (b)

Рис. 4: Результат работы алгоритма по выявлению стеганографической вставки на искусственном изображении с геометрическими фигурами: a) изображение с встроенным сообщением, b) матрица для изображения со стегановставкой.



(a) (b)

Рис. 5: Результат работы алгоритма по выявлению стеганографической вставки на фотографическом изображении: a) изображение с встроенным сообщением, b) матрица для изображения со стегановставкой..

[6] Q. Guan,J. Dong,T. Tan An effective image steganalysis method based on neighborhood information of pixels. *18th IEEE International Conference on Image Processing*, 2777–2780, 2011.

[7] D. Andrew,A. Ker. General Framework for Structural Steganalysis of LSB Replacement. *M. Barni et al. (Eds.)*, LNCS.3727:296–311, 2005.

[8] S. Bhattacharyya,G. Sanya. Steganalysis of LSB Image Steganography using Multiple Regression and Auto Regressive (AR) Model. *Int. J. Comp. Tech. Appl*, V.2(4):1069–1077, 2011.

[9] T. Saaty,G. Sanya. 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*, V.102 (2):251–318, 2008.

Steganalysis Algorithm Based on Hierarchy Analysis Method

Sergey V. Belim, Danil E. Vilkhovskiy

This article presents a new method of image analysis with steganographic inserts based on hierarchy analysis method. Images with applied algorithm of replacing the least significant bit (LSB) are investigated. Detection

is performed by dividing the image into layers and making an analysis of zero-layer of adjacent bits for every bit. First-layer and second-layer are analyzed too. Hierarchies analysis method is used for making decision if current bit is changed. Weighting coefficients as part of the analytic hierarchy process are formed on the values of bits. Then a matrix of corrupted pixels is generated. Visualization of matrix with corrupted pixels allows to determine size, location and presence of the embedded message. Computer experiment was performed. Message was embedded in a bounded rectangular area of the image. This method demonstrated efficiency even at low filling container, less than 10. Widespread statistical methods are unable to detect this steganographic insert. The location and size of the embedded message can be determined with an error which is not exceeding to five pixels.