УДК 003.26

Д.М. Валеев, А.А. Смагин

ОПРЕДЕЛЕНИЕ КОЛИЧЕСТВА ИНФОРМАЦИИ В КОНТЕЙНЕРЕ, ПОЛУЧЕННОМ СТЕГАНОГРАФИЧЕСКИМ АЛГОРИТМОМ, НЕ ВЫЗЫВАЮЩИМ ИСКАЖЕНИЙ

Валеев Дамир Маратович, *окончил Ульяновский государственный университет, аспирант кафедры «Телекоммуникационные технологии и сети» УлГУ. Имеет статьи в области криптографии и стеганографии. [e-mail: damirivaleev@gmail.com].*

Смагин Алексей Аркадьевич, *доктор технических наук, профессор, окончил радиотехнический факультет Ульяновского политехнического института. Заведующий кафедрой «Телекоммуникационные технологии и сети» УлГУ. Имеет статьи, изобретения, монографии в области разработки информационных систем различного назначения. [e-mail: smaginaa1@mail.ru].*

Аннотация

В данной статье рассматриваются параметры и характеристики стеганографического алгоритма, не вносящего искажения в контейнер, такие как: универсальность применения к различным видам информации, простота реализации, а также защищенность и количество передаваемой информации.

Цель данной работы состоит в определении зависимости количества информации, передаваемой по скрытому каналу связи, от размеров встраиваемого сообщения и контейнера. Для достижения цели ставились задачи определить количество информации при различных размерах контейнера и сообщения, а также удалось ли встроить сообщение полностью. Для решения задач проведено два эксперимента с различными соотношениями размеров встраиваемого секрета и контейнера. Определено соотношение размеров сообщения и контейнера для эффективного встраивания.

Описано теоретико-множественное представление модели стегосистемы, в котором применяется данный алгоритм, а также показана схема работы этого алгоритма.

Проведено сравнение передаваемого количества информации контейнером, полученным с помощью предложенного алгоритма, с существующими стеганографическими методами на тестовых изображениях Lenna и Baboon. Показано, что предложенный алгоритм эффективнее большинства существующих методов по показателю количества информации.

Ключевые слова: стеганография, встраивание, ключ, количество информации, алгоритм, секрет, контейнер.

doi: 10.35752/1991-2927-2020-1-5-30-37

DETERMINATION OF PAYLOAD CAPACITY OF CONTAINER OBTAINED BY STEGANOGRAPHIC TECHNIQUE, NOT CAUSING DISTORTION

Damir Maratovich Valeev, *graduated from Ulyanovsk State University; Postgraduate Student at the Department of Telecommunication Technologies and Networks of Ulyanovsk State University; an author of articles in the field of cryptography and steganography. e-mail: damirivaleev@gmail.com.*
Aleksei Arkadevich Smagin, *Doctor of Sciences in Engineering, Professor; graduated from the Radioengineering Faculty of Ulyanovsk Polytechnic Institute; Head of the Department of Telecommunication Technologies and Networks of Ulyanovsk State University; an author of articles, inventions, and manuals in the field of the development of information systems of different purposes. e-mail: smaginaa1@mail.ru.*

Abstract
This article discusses the parameters and characteristics of a steganographic algorithm that does not distort the container, such as the versatility of application to different types of information, ease of implementation, as well as the security and payload capacity.
The purpose of this work is to determine the dependence of payload in hidden communication channel on the size of the embedded data and container. To achieve this goal, the tasks were set to determine payload at different sizes of the container and the data, as well as whether it was possible to embed secret data completely. To solve these problems, two experiments were conducted with different ratios of the size of the embedded secret and the container. The ratio of sizes of data and container for efficient embedding was determined.
Described the set-theoretic representation of model of the stegosystem for algorithm, and shown the scheme of operation of this algorithm.
Compared payload of container obtained using the proposed algorithm and the existing steganographic methods on the test images of Lenna and Baboon. The experimental results show that the proposed algorithm is more effective than most existing methods in terms of embedding capacity.

Key words: steganography, embedding, key, payload, algorithm, secret, container.

Введение

В настоящее время в стеганографии используются различные методы сокрытия информации. Главная задача в стеганографическом сокрытии информации – скрыть сам факт передачи информации по каналу. Общая модель стеганографических систем состоит из передаваемого сообщения, контейнера и секретного ключа. Если сообщение и ключ должны храниться в тайне, то контейнер передается по открытому каналу связи и может быть представлен в любом виде информации. Существуют алгоритмы, применимые только к отдельным видам информации, то есть к тексту, графическим изображениям, аудио- и видеофайлам. Применение таких специализированных алгоритмов приводит к тому, что для организации скрытого канала абоненты вынуждены использовать один вид информации, что значительно ослабляет надежность скрытой передачи данных.

Зачастую такие методы являются также достаточно сложными в реализации, поскольку используют механизмы, относящиеся к конкретному виду информации. Так, например, для встраивания сообщений в изображения, вычисляются коэффициенты дискретного косинусного преобразования (ДКП), для аудио используются эхо-методы и частотное маскирование [1].

Не менее важное значение имеют решения задач повышения устойчивости стеганографических алгоритмов к атакам на контейнер [2].

Еще одной особенностью существующих методов является встраивание битов в контейнер, заменяя исходные, что приводит к его искажению, в той или иной степени, в зависимости от подхода [3]. Это приводит к тому, что полученный стегоконтейнер отличается от исходного, из-за чего стеганографическая система становится обнаружимой, что снижает надежность канала передачи.

В работе [4] был предложен стеганографический алгоритм сокрытия информации с использованием ключа в виде однострочной таблицы, который не вызывает искажений контейнера, что повышает его надежность по сравнению с текущими методами. Алгоритм использует в качестве входных параметров сообщение и контейнер в форме двоичных последовательностей. В результате работы алгоритма формируется ключевая таблица, необходимая для извлечения сообщения. В ней записываются координаты элементов сообщения, находящегося в контейнере. При этом считается, что ключевую таблицу необходимо передать по защищенному каналу для извлечения сообщения.

Основными этапами работы этого алгоритма являются:

1. Выполняется преобразование контейнера и секретного сообщения в двоичный вид и производится считывание первого бита секрета и старшего бита первого байта контейнера.

Далее этапы повторяются, пока не считан последний бит секрета или байт контейнера.

2. Проверяется равенство считанного бита секретного сообщения считанному биту контейнера.

3. Если условие проверки выполнено, формируется ключевая таблица из координат нахождения бит секретного сообщения в контейнере и считываются следующий бит секрета и байт контейнера.

4. Если условие проверки не выполнено, считывается следующий байт контейнера.

5. Когда достигнут конец контейнера или секрета, ключевая таблица сформирована и готова для передачи по защищенному каналу связи.

Рис. 1. Работа алгоритма

У алгоритма есть сходство с книжным шифром, который использует замены блоков исходного текста на адреса или координаты блоков ключевых файлов [5]. Подобно этому шифру, в ключевой таблице помещаются координаты битов секрета, совпадающих со старшими битами контейнера. Затем по координатам восстанавливается исходная секретная информация.

1 Модель стегосистемы

Введем теоретико-множественное представление модели системы, в которой может применяться такой алгоритм встраивания. Пусть C, S, K, R – множества контейнеров, сообщений, ключевых таблиц и сформированных стеганограмм. Тогда стеганограммы выглядят следующим образом:

$$R:\left(C^*, K\right),$$

где C^* – множество полученных контейнеров после встраивания, но, поскольку с помощью предложенного алгоритма встраивание сообщения как таковое не происходит, исходный контейнер C равен заполненному контейнеру C^*, а ключевая таблица K формируется вместе с алгоритмом встраивания

$$E: C \times S \to K.$$

Множество правил встраивания E состоит из элементов, удовлетворяющих равенству $C_{8j} = S_i$, где C_{8j} – старший бит j-го байта в контейнере, S_i – i-й бит сообщения. То есть встраивание происходит, если очередной бит сообщения совпадает со старшим битом свободного байта контейнера. Подробный алгоритм получения ключевой таблицы описан в работе [4].

Стеганограмма состоит из контейнера со встроенным секретом и ключевой таблицы, которую необходимо передать по защищенному каналу связи. Получив стеганограмму, приемник должен восстановить секрет. Алгоритм извлечения секрета может быть представлен как

$D: C^* \times K \to S$, в котором по ключевой таблице происходит поиск байтов контейнера и извлекаются их старшие биты [4].

Совокупность (C, C^*, S, K, R, E, D) является теоретико-множественным представлением модели стегосистемы.

2 Некоторые преимущества предложенного алгоритма

Поскольку этот алгоритм оперирует с битовыми представлениями файлов, он может быть применен к любому виду информации: тексту, изображениям, аудио- и видеофайлам. Поэтому контейнер и встраиваемое сообщение может быть в любой форме. Это дает преимущество универсальности по сравнению с другими методами, которые применяются к конкретным видам информации, к примеру, эхо-методы для аудиофайлов или метод ДКП для изображений [1].

Преимущество универсальности выражается в том, что чаще всего в качестве контейнеров для встраивания используют графические файлы, которые априори представляются аналитику подозрительными, когда как аудио- или видеофайл реже будет объектом для анализа на наличие секретного сообщения. Если передавать по скрытому каналу связи информацию в разных формах, это приведет аналитика в замешательство, поскольку такой способ коммуникации не будет отличаться от обычного канала связи.

Кроме того, данный метод достаточно легко реализуем, поскольку оперирует битовыми представлениями данных и не требует каких-либо сложных математических операций, когда как другим методам необходимо применение таких вычислений, как преобразование Фурье, косинусное преобразование и т. д.

3 Свойства предложенного алгоритма

Рассмотрим надежность обнаружения сообщения в контейнере. Исходя из особенности данного метода, встраивание как таковое не происходит, и контейнер не отличается от стегоконтейнера, то обнаружить сообщение в нем не представляется возможным. Поэтому такие методы обнаружения, как гистограммный анализ или RS-анализ, не дадут результата. Однако существует возможность активного воздействия аналитика на стегоконтейнер путем его модификации. Любое воздействие, приводящее к изменению старших бит в байтах контейнера, приведет к нарушению координат нахождения сообщения в контейнере по ключевой таблице. Известно, что изменение старших бит значительно искажает исходный контейнер так, что эти изменения можно увидеть на изображении или услышать в аудиофайле. Можно заметить также, что поскольку данный метод использует секретный ключ в виде таблицы, который необходимо передавать по защищенному каналу, он подвержен атакам, характерным для криптографических алгоритмов с ключом.

Еще одной характеристикой стеганографической системы является размер передаваемого сообщения в контейнере. В работе [4] было показано, что длина сообщения не может быть более 1/8 части размера контейнера, однако стоит сделать уточнение: равенство длины сообщения и 1/8 части размера контейнера достигается, только если каждый бит сообщения «встраивается» в каждый старший бит очередного байта контейнера без пропуска. Таким образом, если найдется i-й бит сообщения, который не равен старшему биту j-го байта, то, следуя работе алгоритма, будет произведен переход к следующему байту контейнера. В результате окажется, что j-й байт контейнера не заполнен и необходимо увеличить размер контейнера, в противном случае сообщение будет не полностью «встроено» в контейнер.

В настоящей статье предлагается использовать модификацию алгоритма, в которой для каждого бита сообщения производится поиск совпадения со старшим битом байта контейнера не в порядке прямого следования, а по всем байтам, которые не были помечены, как встроенные. При этом, если совпадение найдено, позиция байта контейнера помечается и в дальнейшем поиске не участвует. Такая модификация не позволяет оставлять свободными байты контейнера, которые были пропущены из-за несовпадения их старших бит с битами сообщения.

4 Эксперименты с различными параметрами встраивания

Как указывалось ранее, целью данной работы является получение значений зависимости количества встроенной стеганографической информации в контейнере от размера секретного сообщения и размера контейнера. Для реализации этой цели ставилась задача проведения экспериментов по встраиванию с различных параметров работы алгоритма. Необходимо было проверить, будет ли теоретическое значение встроенного количества информации, полученное на ранней стадии исследований [4], подтверждено в практических условиях. Для этого были проведены 2 эксперимента, в ходе которых решались следующие задачи:

1. Определить, какую часть контейнера займет встроенный секрет, т. е. количество встроенной секретной информации.

2. Определить, удалось ли встроить сообщение полностью, используя разные размеры секрета и контейнера.

В качестве исходных данных использовалась следующая информация:

1. В стеганографическом алгоритме в качестве контейнеров использовались аудиофайлы формата mp3, как наиболее часто встречаемого при обмене информацией. При этом файлы были представлены в форме двоичных последовательностей и разделены на части по 1, 2, … 1000 байт для удобства вычислений.

2. В качестве секрета использовались случайные последовательности, полученные с помощью криптографического генератора случайных последовательностей, размер которых выбирался исходя из следующего правила: длина сообщения не может превышать 1/8 размера контейнера, так например, для контейнера размером 800 байт использовалась случайная последовательность длиной 100 байт.

Заметим, что можно использовать и другие форматы файлов, поскольку в рассматриваемом случае в качестве контейнера применяется двоичная последовательность.

4.1 Эксперимент № 1

Встраивание происходит следующим образом: в контейнер-файл размера N_i байт встраивается случайная последовательность длиной M_j байт, определяется сколько бит удалось встроить, затем вычисляется отношение количества бит к размеру файла, т. е. количество информации $K_{ij} = \frac{M_j}{N_i}$, передаваемое встроенным секретом. Для большей точности для каждого i-го размера контейнера и j-го размера секрета проводится 100 итераций, при этом на каждой итерации используется новая случайная последовательность.

Для статистической точности было использовано по 10000 файлов размера 1, 2, …, 1000 байт, то есть для

каждого набора файлов одного размера получилось по 1000000 значений K.

Затем вычисляется среднее значение количества передаваемой информации для каждого размера контейнера как математическое ожидание, при этом в эксперименте встраивания значение количества информации рассматривалось как случайная величина. Событием k_i в таком эксперименте будет встраивание k бит, статистической вероятностью p_i является отношение количества событий «встраивание k бит» ко всем возможным встраиваниям.

Отсюда количество встроенной информации блока файлов можно оценить как $S = \sum_{S}^{1000000} p_i \cdot k_i$. Полученные экспериментальные результаты отражены на рисунке 2. Здесь на горизонтальной оси отображены размеры контейнера, на вертикальной – доля контейнера, использованная для секрета.

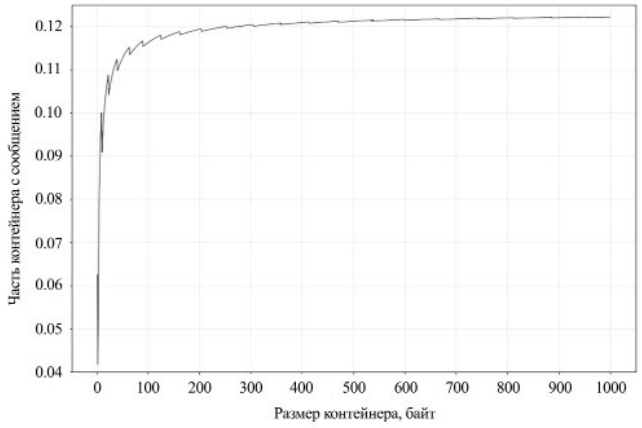

Рис. 2. Отношение количества встроенной информации к размеру контейнера

Из результатов видно, что показатель количества информации с ростом размера контейнера приближается к 1/8 части от размера контейнера. Другими словами, при больших значениях размера контейнера отношение длины сообщения к размеру контейнера будет стремиться к теоретическому показателю:

$\frac{S}{C} = \frac{1}{8}$, где S – размер сообщения, а C – размер контейнера.

В результате эксперимента были найдены такие соотношения размеров секрета и контейнера, при которых секрет занимал максимально возможную часть контейнера, т. е. контейнер использовался эффективно. Результат представлен на рисунке 3. На горизонтальной оси расположены значения размеров секрета, на вертикальной – значения размеров контейнера.

Как видно из графика, с ростом размера секрета размер контейнера, необходимый для эффективного встраивания, растет линейно, из чего следует важный вывод: в среднем для встраивания одного байта сообщения требуется восемь байт контейнера, независимо от соотношений размеров сообщения и контейнера.

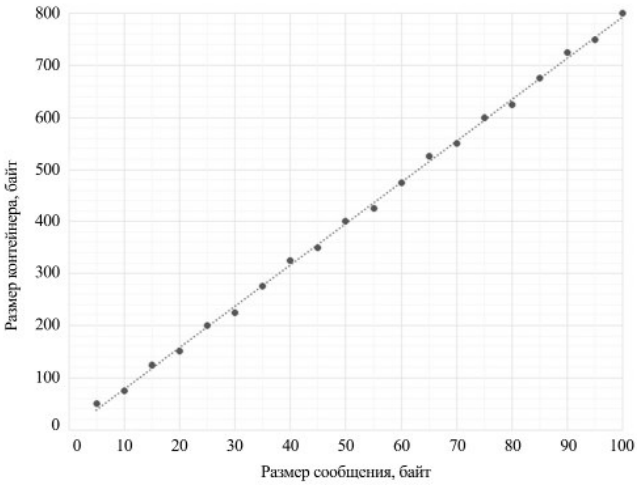

Рис. 3. Соотношения размеров для эффективного встраивания

4.2 Эксперимент № 2

В ходе следующего эксперимента были определены значения той части секрета, которую удалось встроить в контейнер. Эксперимент базировался на следующих действиях: при работе основного алгоритма запоминалось количество k встроенных битов секрета, затем вычислялось отношение k/n, где n – исходная длина секрета. Полученное значение этого отношения и определяет размер встроенной части секрета. Для каждого дискретного размера секрета (1,2,… 1000 бит) и размеров контейнера (1,2,… 1000 байт) происходили встраивание и вычисление значения встроенной части секрета.

Результаты эксперимента отражены на рисунке 4. Здесь по горизонтальной оси отмечены значения размеров контейнера, по вертикальной оси – часть секрета, которую удалось встроить (значение, равное 1, означает, что секрет полностью встроен), графики представ-

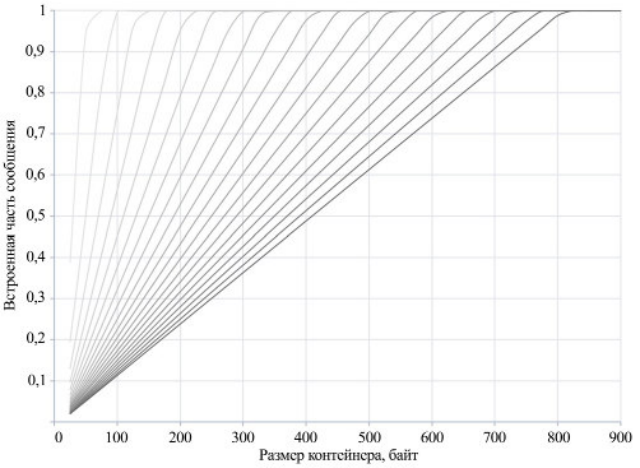

Рис. 4. Рост значения встроенной части для разных секретов и контейнеров

ляют собой сообщения разной длины от 5 до 100 байт слева направо.

По результатам эксперимента можно утверждать, что при разных размерах секретов и контейнеров рост значений встроенной части секрета происходит линейно до точки перегиба, в которой секрет практически полностью встроен и свободного места в контейнере не осталось, то есть в этой точке достигнута наибольшая эффективность встраивания. Дальнейшее увеличение размера контейнера приводит к медленному приближению к 1, а значит, к полному встраиванию сообщения.

Важно отметить, что только лишь при размере контейнера 1000 байт удалось полностью встроить все биты секретов каждого размера. Хотя и при меньших размерах, встроенная часть также приближается к значению 1, это говорит о том, что при малых размерах контейнера вероятность полного встраивания секрета уменьшается.

Если же использовать контейнеры, превышающие секреты не в 8 раз, а меньше, тогда при встраивании будет утеряна значительная часть секрета. Такой подход может быть использован, если найдется ситуация, в которой будет не так важно потерять часть информации в конце сообщения. В таком случае соотношение размеров секрета и контейнера может отличаться от 1/8.

5 Сравнение с существующими методами

Основными параметрами любого стеганографического метода являются количество передаваемой информации в контейнере и вносимые в контейнер искажения. Поскольку предложенный алгоритм не вносит искажения, в этом его преимущество перед существующими. Поэтому для сравнения используем значение количества информации. Во всех рассмотренных методах для вычисления тестовых значений используют общепринятые изображения Lenna и Baboon с разрешением 512x512 и глубиной цвета 8 бит. Результаты взяты из работы [6] и отражены в таблице 1 для изображения Lenna и в таблице 2 – для Baboon.

Исходя из экспериментальных результатов и сравнения существующих методов видно, что предложенный алгоритм позволяет встраивать больше количества информации, чем большинство других методов за счет того, что в предложенном алгоритме не используются встраивания в пространственные формы контейнера и сложные математические преобразования, как в других. При этом в предложенном алгоритме не происходит искажение исходного контейнера, что увеличивает надежность скрытой передачи информации.

Заключение

Можно сделать вывод, что на практике данный алгоритм обеспечивает значение количества встроенной информации, близкое к теоретическому. По результатам экспериментов видно, что зачастую не удавалось встроить полную строку секрета, а лишь 0,9999 ее части. Это означает, что существуют такие параметры битов секрета и контейнера, при которых хотя бы 1 бит не будет встроен. Еще один способ решить эту проблему – это использовать помехоустойчивое кодирование перед встраиванием, считая, что биты сообщения, которые не удалось встроить, были искажены в результате помех. При извлечении секрета придется восстанавливать необходимые биты секрета, которые не удалось встроить.

Для того чтобы секрет был полностью встроен, следует использовать контейнеры, размер которых более чем в 8 раз превышает размер секретного сообщения.

Таблица 1

Сравнение количества информации по изображению Lenna

Методы	Количество информации (в битах)
[7]	837332
[8]	5460
[9]	5336
[10]	59900
[11]	60241
[12]	50960
[13]	24108
[14]	1024
[15]	85507
[16]	74600
[17]	71674
[6]	262144
Предложенный	259381

Таблица 2

Сравнение количества информации по изображению Baboon

Методы	Количество информации (в битах)
[7]	916010
[8]	5421
[9]	5208
[10]	19130
[11]	21411
[12]	22696
[13]	2905
[14]	1024
[15]	14916
[16]	15176
[17]	56291
[6]	262144
Предложенный	249225

Однако можно предположить возможность увеличения передаваемого количества информации, например, встраивая сразу несколько бит секретного сообщения в один байт контейнера.

Для повышения количества информации, передаваемой по каналу связи, целесообразно провести эксперименты со встраиванием не одного бита сообщения в байт контейнера, а нескольких. Также можно встраивать биты сообщения не в порядке очереди, при котором возможны пропуски байтов контейнера, а в случайном – используя дополнительный стеганографический ключ, что также может привести к увеличению размера передаваемого сообщения.

Также предлагается использовать не биты секрета, а целиком байты, и искать совпадения с байтами контейнера. Однако использование битов представляется более эффективным для повышения количества передаваемой секретной информации.

СПИСОК ЛИТЕРАТУРЫ

1. Fabien A.P. Petitcolas, Ross J. Anderson, Markus G. Kuhn. Information Hiding: A Survey // Proceedings of the IEEE (special issue). 1999. Vol. 87, no. 7. pp. 1062–1078.

2. Смагин А.А., Валишин М.Ф. Устойчивые к атакам на контейнер стеганографические алгоритмы // Инфокоммуникационные технологии. 2015. Т. 13, № 1. С. 82–88.

3. Khurana, Anil & Mohit Mehta, B. (2012). Comparison of LSB and MSB based Image Steganography. URL: http://ijcst.com/vol33/4/anil2.pdf (дата обращения: 18.12.2019).

4. Валеев Д.М. Стеганографический метод передачи информации, не вызывающий искажений контейнера // Сб. ст. X междунар. науч.-практ. конф. М.: «Научно-издательский центр «Актуальность.РФ», 2017. С. 41-43.

5. Changda Wang, Shiguang Ju. A Novel Method to Implement Book Cipher // Journal of Computers. 2010. 5(11). pp. 1621–1628.

6. An improved image steganography technique based on MSB using bit differencing / A.U. Islam et al. // 2016 Sixth International Conference on Innovative Computing Technology (INTECH). Dublin, 2016. pp. 265–269.

7. Adaptive data hiding in edge areas of images with spatial LSB domain systems / C.-H. Yang, C.-Y. Weng, S.-J. Wang, H.-M. Sun // IEEE Transactions on Information Forensics and Security. 2008. Vol. 3. pp. 488–497.

8. Reversible data hiding / Z. Ni, Y.-Q. Shi, N. Ansari, W. Su // IEEE Transactions on Circuits and Systems for Video Technology. 2006. Vol. 16. pp. 354–362.

9. Hwang J., Kim J., Choi J. A reversible watermarking based on histogram shifting // Digital Watermarking. Springer. 2006. pp. 348–361.

10. Lin C.-C., Hsueh N.-L. A lossless data hiding scheme based on three-pixel block differences // Pattern Recognition. 2008. Vol. 41. pp. 1415–1425.

11. Hu Y., Lee H.-K., Li J. DE-based reversible data hiding with improved overflow location map // IEEE Transactions on Circuits and Systems for Video Technology. 2009. Vol. 19. pp. 250–260.

12. Wu D.-C., Tsai W.-H. A steganographic method for images by pixel-value differencing // Pattern Recognition Letters. 2003. Vol. 24. pp. 1613–1626.

13. Goljan M., Fridrich J.J., Du R. Distortion-free data embedding for images // Proceedings of the 4th Information Hiding Workshop. Pittsburgh, PA, 2001. pp. 27–41.

14. Distortionless data hiding based on integer wavelet transform / G. Xuan, J. Zhu, J. Chen, Y.Q. Shi, Z. Ni, W. Su // Electronics Letters. 2002. Vol. 38. pp. 1646–1648.

15. Celik M.U., Sharma G., Saber E. Reversible data hiding // Proceedings of IEEE International Conference on Image Processing. 2002. Vol. 2. pp. 157–160.

16. Yalman Y., Akar F., Erturk I. An image interpolation based reversible data hiding method using R-weighted coding // IEEE 13th International Conference on Computational Science and Engineering. 2010. pp. 346–350.

17. Reversible image watermarking using interpolation technique / L. Luo, Z. Chen, M. Chen, X. Zeng, Z. Xiong // IEEE Transactions on Information Forensics and Security. 2010. Vol. 5. pp. 187–193.

REFERENCES

1. Fabien A.P. Petitcolas, Ross J. Anderson, Markus G. Kuhn. Information Hiding: A Survey. *Proceedings of the IEEE (special issue)*, 1999, vol. 87, no. 7, pp. 1062–1078.

2. Smagin A.A., Valishin M.F. Ustoichivye k atakam na konteiner steganograficheskie algoritmy [Steganography Algorithms Robust to Active Attacs]. *Infokommunikatsionnye tekhnologii* [Infocommunication Technologies], 2015, vol. 13, no. 1., pp. 82–88.

3. Khurana, Anil, B. Mohit Mehta. Comparison of LSB and MSB based Image Steganography. *IJCST*, 2012, vol. 3, iss. 3. Available at: http://ijcst.com/vol33/4/anil2.pdf (accessed: 18.12.2019).

4. Valeev D.M. Steganograficheskii metod peredachi informatsii, ne vyzyvaiushchii iskazhenii konteinera [Steganographic Method of Information Transmission without Distortion of the Cover Object]. *Sb. st. X mezhdunar. nauch.-prakt. konf.* [Proc. of the 10th Int. Workshop], Moscow, Aktualnost.RF Sci. Publ., 2017, p. 41-43.

5. Changda Wang, Shiguang Ju. A Novel Method to Implement Book Cipher. *Journal of Computers*, 2010, 5(11), pp. 1621–1628.

6. Islam, A.U., et al. An Improved Image Steganography Technique based on MSB using Bit Differencing. 2016 *Sixth International Conference on Innovative Computing Technology (INTECH)*. Dublin, 2016, pp. 265–269.

7. Yang, C.-H., C.-Y. Weng, S.-J. Wang, H.-M. Sun. Adaptive Data Hiding in Edge Areas of Images with Spatial LSB Domain Systems. *IEEE Transactions on Information Forensics and Security*, 2008, vol. 3, pp. 488–497.

8. Ni, Z., Y.-Q. Shi, N. Ansari, W. Su. Reversible Data Hiding. *IEEE Transactions on Circuits and Systems for Video Technology*, 2006, vol. 16, pp. 354–362.

9. Hwang J., Kim J., Choi J. A Reversible Watermarking based on Histogram Shifting. *Digital Watermarking*, Springer, 2006, pp. 348–361.

10. Lin C.-C., Hsueh N.-L. A Lossless Data Hiding Scheme based on Three-Pixel Block Differences. *Pattern Recognition*, 2008, vol. 41, pp. 1415–1425.

11. Hu Y., Lee H.-K., Li J. DE-based Reversible Data Hiding with Improved Overflow Location Map. *IEEE Transactions on Circuits and Systems for Video Technology*, 2009, vol. 19, pp. 250–260.

12. Wu D.-C., Tsai W.-H. A Steganographic Method for Images by Pixel-Value Differencing. *Pattern Recognition Letters*, 2003, vol. 24, pp. 1613–1626.

13. Goljan M., Fridrich J.J., Du R. Distortion-Free Data Embedding for Images. *Proceedings of the 4th Information Hiding Workshop*. Pittsburgh, PA, 2001, pp. 27–41.

14. G. Xuan, J. Zhu, J. Chen, Y.Q. Shi, Z. Ni, W. Su. Distortionless Data Hiding based on Integer Wavelet Transform. *Electronics Letters*, 2002, vol. 38, pp. 1646–1648.

15. Celik M.U., Sharma G., Saber E. Reversible Data Hiding. *Proceedings of IEEE International Conference on Image Processing*. 2002, vol. 2, pp. 157–160.

16. Yalman Y., Akar F., Erturk I. An Image Interpolation based Reversible Data Hiding Method using R-Weighted Coding. *IEEE 13th International Conference on Computational Science and Engineering*. 2010, pp. 346–350.

17. L. Luo, Z. Chen, M. Chen, X. Zeng, Z. Xiong Reversible Image Watermarking using Interpolation Technique. *IEEE Transactions on Information Forensics and Security*, 2010, vol. 5, pp. 187–193.