Библиографический список

1. Шевченко А.С. Численные методы: учеб. пособие. – Барнаул; Рубцовск: Изд-во АлтГУ, 2016. – 388 с.
2. Рихтер Джеффри. Программирование на платформе Microsoft. NET Framework 4.5 на языке C#. – М.: Питер, 2016. – 365 с.

УДК 004.056.5

Методы встраивания и обнаружения сокрытых сообщений, основанные на особенностях GIF-изображений

Д.И. Строкин, И.В. Пономарев
Алтайский государственный университет, Барнаул

В статье рассматриваются метод обеспечения конфиденциальности данных средствами Цифровой Стеганографии, использующий в качестве файлов-контейнеров изображения формата GIF, а также метод, оценивающий вероятность сокрытия информации в блоках файла-контейнера. Как результат, была создана компьютерная программа на базе среды программирования Microsoft Visual Studio 2019 Community и языка программирования C# (платформа .NET Framework).

Ключевые слова: *контейнер, сообщение, палитра, индекс цвета, наименее значащий бит*.

Определение. *Контейнером (носителем)* называют несекретные данные, которые используют для сокрытия сообщений. *Пустой контейнер* – контейнер без встроенного сообщения; *заполненный контейнер или стего-контейнер*, содержащий встроенную информацию [1].

В качестве носителей информации были выбраны изображения формата .GIF, использующие LZW-сжатие без потерь. Отличительными чертами данного формата являются [4]:

1) Использование блочной структуры данных;
2) Использование *палитры цветов* – фиксированного набора (диапазона) цветов и оттенков, имеющего физическую или цифровую реализацию в том или ином виде.

При использовании палитры, каждая точка изображения содержит лишь номер цвета из палитры, а не информацию о ее цвете в цветовом пространстве.

Разработка метода сокрытия информации

Разрабатываемый алгоритм базируется на достаточно распространенном методе *замены Наименее Значащего Бита* (Least Significant Bit), основной принцип которого заключается в том, что передаваемая информация встраивается в значения младших битов изображения. Модификация именно таких битов не способна восприниматься человеческим зрением, так как они несут в себе меньше всего информации [1].

Рассмотрим пример. Допустим, имеется 8-битное изображение в градациях серого. 00 (00000000) обозначает чёрный цвет, FF (11111111) – белый. Всего имеется 256 градаций. Также предположим, что сообщение состоит из 1 байта – например, 01101011.

При использовании 2 младших бит в описаниях пикселей, нам потребуется 4 пикселя. Допустим, они чёрного цвета. Тогда пиксели, содержащие скрытое сообщение, будут выглядеть следующим образом: 00000001 00000010 00000010 00000011.

Однако, такой способ внедрения сообщений справедлив лишь для растровых изображений. Как отмечалось выше, пиксели GIF-изображений – это поток индексов из цветовой палитры, и, если элементы, близкие по индексу, будут иметь совершенно разные представления в цветовом пространстве, изменения младшего бита могут привести к заметным изменениям самого изображения.

Наилучшем решением в подобной ситуации будет использование «подобных» элементов палитры [1]. Под *подобными*, в данном случае, понимаются пары элементов, цветовая интенсивность которых отличается на незначительное число d. Например, значения яркости для цветов (255, 255, 255) и (255, 254, 253) будут некритично отличаться, а значит, индекс одного элемента можно легко заменить индексом другого.

Алгоритм встраивания сообщения

Шаг 1. Сортировка палитры цветов по возрастанию веса W, где:

$$W = R \cdot 65536 + G \cdot 256 + B.$$

Шаг 2. Поиск пар элементов в отсортированной палитре, для которых разность весов W меньше заданной пороговой величины d. Обозначим одну такую пару за (j_i, j_k), где i и k – это индексы элементов в неотсортированной палитре, причем в отсортированной таблице j_i от j_k отличается на 1.

Шаг 3. Сокрытие сообщения. Последовательно просматриваются все точки изображения, по значению точки k определяется соответствующий номер j_k. Если элемент отсортированной палитры j_k пригоден для сокрытия, то его Наименее Значащий Бит заменяется на очередной бит сообщения. Затем по получившемуся номеру $j_{k'}$ опре-

деляется связанный с ним элемент исходной таблицы k', который и присваивается текущей точке.

Шаг 4. Извлечение сообщения происходит аналогичным способом. Для текущей точки k ищется номер j_k в отсортированной по весу W палитре цветов и, если:

- младший бит индекса j_k равен нулю, смотрим, удовлетворяет ли пара $(j_k, j_k + 1)$ условию: $W_{j_k+1} - W_{j_k} < d$. Если удовлетворяет, значит, из индекса j_k извлекаем младший бит и записываем его в сообщение;
- младший бит индекса j_k равен единице, смотрим, удовлетворяет ли пара $(j_k - 1, j_k)$ условию: $W_{j_k} - W_{j_k-1} < d$. Если удовлетворяет, значит, из индекса j_k извлекаем младший бит и записываем его в сообщение.

Алгоритм обнаружения факта скрытия сообщения

Выявление факта сокрытия информации внутри файла-контейнера – отдельный вид стеганографических атак, часто основывающихся на различных статистических закономерностях контейнеров. Модификацию одной из таких атак мы и рассмотрим.

Гистограммный метод или *метод, основанный на критерии* χ^2 [1, 2, 5] предполагает, что вероятность одновременного появления соседних (то есть отличных на наименее значащий бит) цветов в незаполненном контейнере крайне мала. А при последовательном встраивании равномерного сообщения, пиксели изображения, напротив, приобретают равномерное распределение. Поэтому степень различия между вероятностными распределениями элементов естественных контейнеров и полученных из них стего может быть использована для оценки вероятности существования стегоканала.

Шаг 1. Сортировка палитры цветов по возрастанию веса W, где:

$$W = R \cdot 65536 + G \cdot 256 + B.$$

Шаг 2. Разбиение изображения на отдельные блоки. Гораздо удобнее оценивать вероятность внедрения секретной информации в отдельный блок, чем во все изображение целиком.

Шаг 3. Для текущего блока подсчитывается, сколько раз n_i^* её элемент x_i принял рассматриваемые значения, где всего k-элементов. Иными словами, мы строим эмпирическую гистограмму по количеству вхождений для каждого элемента палитры.

Шаг 4. Далее происходит построение теоретической гистограммы на основе эмпирической, путём нахождения среднего арифмети-ческого количества пикселей элементов с соседними номерами:

$$n_0 = n_1 = \frac{n_0^* + n_1^*}{2}.$$

Шаг 5. Величина χ^2 для сравниваемых распределений последовательности и ожидаемого распределения стего равна:

$$\chi^2 = \sum_{i=1}^{v} \frac{(n_i - n_i^*)^2}{n_i^*}.$$

Шаг 6. Таким образом, вероятность p того, что два распределения одинаковы [5], определяется:

$$p = 1 - \frac{1}{2^{\frac{k-1}{2}}\Gamma\left(\frac{k-1}{2}\right)} \int_0^{\chi^2_{k-1}} e^{-\frac{x}{2}} x^{\frac{k-1}{2}-1} dx,$$

где Г – гамма-функция Эйлера, k – количество цветов в палитре.

Библиографический список:

1. Аргановский А.В., Балакин А.В., Грибунин В.Г., Сапожников С.А. Стеганография, цифровые водяные знаки и стеганоанализ. – М: Вузовская книга. 2009 – 220 с.

2. Грибунин В.Г., Оков И.Н., Туринцев И.В. Цифровая стеганография. – М.: СОЛОН-ПРЕСС, 2009 – 272 с.

3. Конахович Г.Ф., Пузыренко А.Ю. Компьютерная стегано-графия. Теория и практика. – К: «МК-Пресс». 2006 – 288 с.

4. Сэломон Д. Сжатие данных, изображений и звука – М.: Техносфера. 2004. – 368 с.

5. Westfeld A., Pfitzmann A. 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.

УДК 004

Технологии дополненной реальности для обучения детей устному счету

В.В. Ширяев, О.Н. Половикова
АлтГУ, г. Барнаул

В данной работе рассматривается проблема использования технологий дополненной реальности для обучения детей школьного и дошкольного возраста навыкам устного счета. Изучены основные особенности разработки обучающих систем с дополненной реальностью. Разработанная мобильная система позволяет получить и закрепить базовые навыки устного счета, используя набор игровых уровней в виде