Наводиться опис розробленого програмного комплексу для практичної оцінки стійкості стеганографічних методів приховування інформації в цифрових зображеннях. Програма може використовуватися для вибору найбільш ефективного стеганографічного алгоритму, якщо відомі характеристики зображень такі як формат, розмір, палітра та ін., а також атаки, яким може бути піддданий контейнер з вкрапленою інформацією

Ключові слова: стеганографія, приховування інформації, цифрове зображення, захист авторського права, аналіз стійкості

Приводится описание разработанного программного комплекса для практической оценки устойчивости стеганографических методов сокрытия информации в цифровых изображениях. Программа может использоваться для выбора наиболее эффективного стеганографического алгоритма, если известны характеристики изображений такие как формат, размер, палитра и др., а также атаки, которым может быть подвергнут контейнер со встроенной информацией

Ключевые слова: стеганография, сокрытие информации, цифровое изображение, защита авторского права, анализ стойкости

УДК 004.056

СИСТЕМА ДЛЯ ИССЛЕДОВАНИЯ СТОЙКОСТИ РОБАСТНЫХ СТЕГАНОГРАФИЧЕСКИХ АЛГОРИТМОВ

Д. М. Андрущенко
Младший научный сотрудник, ассистент
Кафедра защиты информации
Запорожский национальный
технический университет
ул. Жуковского, 64, г. Запорожье,
Украина, 69063
E-mail: andrush85@mail.ru

1. Введение

В последнее время предложено большое количество стеганографических алгоритмов встраивания данных в цифровые изображения [1 – 7]. Одни из них используются для встраивания цифровых водяных знаков с целью подтверждения авторских прав на изображения. Другие – для встраивания цифровых отпечатков пальцев с целью идентификации нелегальных копий и выявления правонарушителя, нарушившего лицензионное соглашение. Исследованию стойкости таких алгоритмов посвящено значительно меньше публикаций. К тому же до сих пор нет единой объективной методики сравнения стойкости различных алгоритмов между собой.

При построении стеганосистем для защиты авторских прав к алгоритмам предъявляется два требования.

Во-первых, необходимый уровень стойкости (робастности) встроенных в изображение данных к преднамеренным либо непреднамеренным искажениям контейнера [1]. Во-вторых, требование минимального видимого изменения контейнера при встраивании данных в изображение. Эти требования являются противоречивыми: чем меньше будет величина вносимых искажений в изображение при встраивании информации, тем больше будут разрушаться встроенные данные при изменении контейнера. Практически все предлагаемые алгоритмы позволяют варьировать параметры стойкости и величины вносимых искажений путем изменения, так называемого, коэффициента силы встраивания P. Кроме того, стойкость встроенных данных зависит как от формата, размера и палитры изображения, так и от видов и интенсивности искажений (атак), которым может быть подвержено изображение.

Поэтому для построения надежных и эффективных стеганосистем при заданных условиях, таких как формат и размеры изображений, а также известных видах и интенсивности атак, которым может быть подвержен контейнер со встроенной информацией, необходимо достигнуть компромисса между этими двумя параметрами. Для этого потребуется исследовать известные алгоритмы, найти оптимальные значения коэффициента силы встраивания P для каждого из них и сравнить алгоритмы между собой, после чего выбрать наиболее подходящий метод для данного случая.

2. Постановка проблемы

В работах [1 – 3] представлены теоретические оценки стойкости стеганосистем, однако, как известно, наиболее достоверным методом оценки стойкости является непосредственное определение параметров искажений, при которых встроенная информация будет разрушена.

Поэтому возникает необходимость в построении системы, представляющей собой программный комплекс, реализующий методы встраивания скрытых данных в цифровые изображения с использованием различных стеганографических алгоритмов и такого параметра, как коэффициент силы встраивания.

В литературе [8, 9] встречаются описания подобных программных комплексов. Однако, обычно они предназначены для исследования определенного алгоритма, например, предложенного самим автором такой программы либо для выполнения сравнения

алгоритмов в рамках определенного исследования. Например, в работе [8] исследуется стойкость алгоритмов к сжатию формата JPEG.

Полностью реализованных и опубликованных кодов универсальных систем такого типа в открытом доступе не выявлено.

Стеганографические системы, к которым предъявляется требование робастности и которые используются для защиты авторского права на изображения, должны быть стойкими как минимум к следующим типам искажения контейнера: сжатие с потерями, масштабирование, поворот, кадрирование, цветовая коррекция, редактирование, ретуширование, повышение резкости, добавление шумов, фильтрация от шума.

В данной работе была поставлена задача создания программного комплекса, позволяющего проводить анализ известных методов встраивания скрытых данных в цифровые изображения при различных значениях такого параметра, как коэффициент силы встраивания. Изображения со встроенной информацией должны подвергаться различным атакам с различной степенью искажения, после чего программа должна вычислять степень разрушения встроенной информации.

3. Практическая оценка стойкости стеганографических систем

В основном, стеганографические алгоритмы, предназначенные для защиты авторского права, основаны на встраивании данных в области преобразования [1], например, дискретно-косинусной, вейвлет либо фрактальной. Такие алгоритмы наиболее стойки к компрессии изображений с потерями. Суть их заключается в том, что изображение разбивается на блоки 8x8 пикселей и в каждый такой блок встраивается 1 бит информации (0 либо 1) путем изменения одного, двух или нескольких коэффициентов в соответствующей матрице коэффициентов выбранного преобразования. Значение коэффициента силы встраивания P при этом влияет на величину изменения выбранных коэффициентов. Таким образом, коэффициент силы встраивания P и координаты выбранных коэффициентов влияют на величину вносимых искажений в изображения и, соответственно, на стойкость алгоритма. Однако, эти параметры являются сугубо индивидуальными для каждого метода, поэтому при сравнении различных алгоритмов между собой их использовать нельзя.

Для измерения величины вносимых искажений обычно используют максимальное соотношение «сигнал/шум» [3]:

$$PSNR = 10\log_2 \frac{N \cdot 255^2}{\sum_{i=1}^{N} \left(x_i - \underline{x}_i \right)^2}, \quad (1)$$

где $\underline{x}_i$ – значения пикселей исходного изображения;

x_i – значения пикселей изображения со встроенной информацией; N – количество пикселей в изображении.

Для количественной оценки величины разрушения встроенных данных в изображение, подвергнутого различным атакам, используют коэффициент корреляции [3]:

$$\rho = \frac{\sum_{i=1}^{n} w_i \widehat{w}_i}{\sqrt{\sum_{i=1}^{n} w_i^2} \sqrt{\sum_{i=1}^{n} \widehat{w}_i^2}}, \quad (2)$$

где w_i, $\widehat{w}_i$ – соответственно биты исходных и извлеченных данных из изображения подверженного атакам; n – количество бит встроенных данных.

4. Программная реализация

Для проведения исследований реализована программа на языке программирования C#. Для анализа пользователь выбирает тестовые изображения, обладающие необходимыми свойствами, такими как формат, размер, палитра, способ получения картинки и др. Их количество должно быть не менее 100. Каждое изображение разбивается на блоки 8 на 8 пикселей. В каждый блок встраивается один бит информации, вначале 0, затем 1, при этом используются различные методы встраивания и различные параметры (коэффициент силы встраивания и выбранные коэффициенты) для каждого алгоритма. Полученный блок подвергается трансформации, фильтрации, зашумлению, сжатию, используя различные алгоритмы и коэффициенты искажения. После каждой очередной операции из каждого искаженного блока извлекается записанный бит. Если извлеченный бит совпадает с оригиналом, то тест считается пройденным успешно. Кроме того, для количественной оценки величины искажения каждого блока используется пиковое отношение сигнал/шум, вычисляемое в децибелах по формуле (1). А для количественной оценки величины разрушения встроенных данных используется формула (2).

В программе были реализованы следующие функции:

Conditions() – функция задания пользователем тестовых изображений, граничных условий и шага изменения исследуемых параметров;

AlgorithmSelection() – выбор стеганографического алгоритма встраивания данных;

ParamSelection() – выбор параметров алгоритма встраивания;

Embedding() – функция встраивания информации в очередной блок изображения;

!ConEmpty() – условие полного заполнения контейнера;

IsAttack() – условие наличия в списке еще не проведенных атак на изображение;

Attack() – проведение атаки на изображение и определение степени разрушения встроенных данных;

IsParam() – условие наличия в списке еще не использованных параметров алгоритма;

IsAlgorithm() – условие наличия в списке еще не протестированных алгоритмов;

Result() – Формирование результатов.

Блок схема разработанного алгоритма представлена на рис. 1. Фрагмент программы на языке программирования C# для проведения атаки масштабирования изображения представлен в листинге 1.

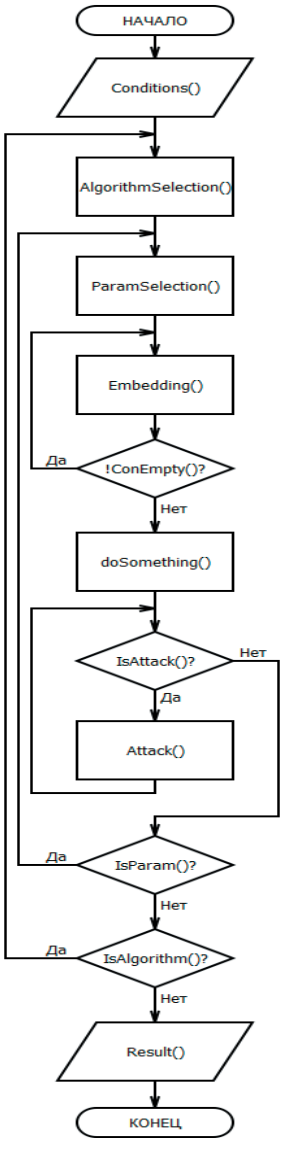

Рис. 1. Блок схема алгоритма для исследования стойкости стеганографических методов скрытия данных

Листинг. Фрагмент программы для проведения атаки масштабирования изображения на языке программирования C#.

```
private void ScaleImage(string image)
{
Image source = Image.FromFile("C:\\images\\" +
+ image);
int width = source.Width * 200 / source.Height;
int height = 200;
Image dest = new Bitmap(width, height);
using (Graphics gr = Graphics.FromImage(dest))
{
gr.FillRectangle(Brushes.White, 0, 0, width, height); //
Очищаем экран
gr.InterpolationMode = System.Drawing.Drawing2D.
InterpolationMode.HighQualityBicubic;
float srcwidth = source.Width;
float srcheight = source.Height;
float dstwidth = width;
float dstheight = height;
if (srcwidth <= dstwidth && srcheight <= dstheight) //
Исходное изображение меньше целевого
{
int left = (width - source.Width) / 2;
int top = (height - source.Height) / 2;
gr.DrawImage(source, left, top, source.Width, source.
Height);
}
else if (srcwidth / srcheight > dstwidth / dstheight) //
Пропорции исходного изображения более широкие
{
float cy = srcheight / srcwidth * dstwidth;
float top = ((float)dstheight - cy) / 2.0f;
if (top < 1.0f) top = 0;
gr.DrawImage(source, 0, top, dstwidth, cy);
}
else // Пропорции исходного изображения более
узкие
{
float cx = srcwidth / srcheight * dstheight;
float left = ((float)dstwidth - cx) / 2.0f;
if (left < 1.0f) left = 0;
gr.DrawImage(source, left, 0, cx, dstheight);
}
dest.Save("C:\\image_res\\" + image);
source.Dispose();
dest.Dispose();
}
}
```

5. Результаты тестирования стеганографических алгоритмов

При помощи реализованной программы были получены результаты анализа для следующих стеганографических алгоритмов: алгоритм Коха-Жао [5], стеганоалгоритм Бенхам [6], однокоэффициентный алгоритм, предложенный в работе [9], и метод, предложенный в патенте [10].

В табл. 1 представлены результаты анализа для стойкости перечисленных алгоритмов к преобразованиям изображений обозначенных, соответственно:

A_1 – стойкость к сжатию с коэффициентом компрессии 30 из 100;

A_2 – стойкость к масштабированию (уменьшению изображения в 10 раз);

A_3 – стойкость к низкочастотной гауссовской фильтрации;

A_4 – стойкость к усредняющей фильтрации с шагом 3 пикселя;

A_5 – стойкость к повышению резкости с шагом 3 пикселя.

Таблица 1

Результаты анализа стойкости стеганоалгоритмов

Алгоритм	A_1	A_2	A_3	A_4	A_5
Коха-Жао	-	+	+	+	-
Бенхам	-	+	+	+	+
Однокоэффициентный	+	+	+	+	-
Метод, предложенный в патенте [10]	+	+	+	+	+

6. Выводы

Полученные результаты позволяют при создании систем стеганографического встраивания данных в цифровые изображения обоснованно выбирать наиболее подходящие методы и параметры алгоритмов, обеспечивающие необходимый уровень стойкости к определенным искажениям контейнера одновременно с максимально возможной «незаметностью» встроенного сообщения. Например, при предъявлении требования стойкости алгоритма к сжатию с коэффициентом компрессии 30, можно использовать одно-коэффициентный алгоритм, предложенный Михайличенко О. В. либо метод, предложенный в патенте [10]. А если требуется стойкость алгоритма к повышению резкости с шагом 3 пикселя, то можно использовать алгоритм Бенхам, либо метод, предложенный в патенте [10]. В этом случае данные, встроенные в цифровое изображение, полностью сохранятся при выполнении указанных преобразований с контейнером.

Внедрение наиболее подходящих методов и параметров алгоритма, полученных при помощи разработанной системы, позволит повысить эффективность существующих стеганографических систем.

Литература

1. Грибунин, В. Г. Цифровая стеганография [Текст] / В. Г. Грибунин, И. Н. Оков, И. В. Туринцев // М.: Солон-Пресс, 2002. – 272 с.
2. Аграновский, А. В. Основы компьютерной стеганографии [Текст] / А.В. Аграновский, П.Н. Девянин, Р.А. Хади и др. // М.: Радио и связь, 2003. – 151 с.
3. Аграновский, А. В. Стеганография, цифровые водяные знаки и стеганоанализ [Текст] / А. В. Аграновский, А. В. Балакин, В. Г. Грибунин, С. А. Сапожников // М.: Вузовская книга, 2009. – 220 с.
4. Коханович, Г. Ф. Компьютерная стеганография. Теория и практика. [Текст] / Г. Ф. Коханович, А. Ю. Пузыренко // К.: МК-Пресс, 2006. – 288 с.
5. Koch, E. Towards robust and hidden image, copyright labeling [Текст] / E. Koch, J. Zhao // In Proceedings of the IEEE Workshop on Nonlinear Signal and Image Processing, pages 452-455, Halkidiki, Greece, 1995.
6. Benham, D. Fast watermarking of DCT-based compressed images [Текст] / Dave Benham, Nasir Memon, Boon-Lock Yeo, Minerva M. Yeung // Proceedings of the International Conference on Image Science, Systems, and Technology (CISST '97), Las Vegas USA. – 1997. – P. 243-252.
7. Eyadat, M. Performance evaluation of an incorporated DCT Block-Based Watermarking algorithm with Human Visual system Model [Текст]/ M. Eyadat, S. Vasikarla // Pattern Recognition Journal. – 2005. – V. 26. – P. 1405-1411.
8. Балакин, А. В. Разработка архитектуры программного комплекса и методов информационной защиты мультимедиа-информации с использованием цифровых водяных знаков [Текст]: дис. канд. техн. наук: 05.13.11, 05.13.19 / А. В. Балакин. – Ростов-на-Дону, 2005. – 195 с.
9. Михайличенко, О. В. Методы и алгоритмы защиты цифровых водяных знаков при JPEG сжатии [Текст]: дис. канд. техн. наук: 05.13.19. / О. В. Михайличенко. – Санкт-Петербург, 2009. – 115 с.
10. Пат. 81168 Україна, МПК H04L 9/8 (2006.01). Спосіб захисту авторського права на цифрові зображення [Текст] / Д. М. Андрущенко, Г. Л. Козіна, Л. М. Карпуков. – № u2012 14519, заявл. 18.12.2012, опубл. 25.06.2013, Бюл. № 12. – 3 с.