DOI: 10.18137/RNU.V9187.24.02.P.114 УДК 004.63

Е.К. Мазайшвили, Е.Ю. Авксентьева

МЕТОД ЦИФРОВОЙ ПОДПИСИ ИЗОБРАЖЕНИЙ БЕЗ ДОПОЛНИТЕЛЬНЫХ ФАЙЛОВ, С УСТОЙЧИВОСТЬЮ К JPEG-СЖАТИЮ И УДАЛЕНИЮ МЕТАДАННЫХ

Аннотация. Рассмотрен метод цифровой подписи изображений, не использующий метаданные или дополнительные файлы. Метод обладает устойчивостью к jpeg-сжатию в определенных пределах. В качестве объекта цифровой подписи выступают блоки 8×8 пикселей с примененным дискретным косинусным преобразованием, из которых состоит jpeg-файл. Метод достижения устойчивости к jpeg-сжатию заключается в квантизации (уменьшении точности) значений в блоках, чтобы при вычислении подписи они оставались неизменными до определенного уровня сжатия.

Ключевые слова: JPEG, дискретное косинусное преобразование, цифровая подпись, сжатие, изображения.

E.K. Mazaishvili, E.Yu. Avksentieva

DIGITAL SIGNATURE METHOD FOR IMAGES WITHOUT ADDITIONAL FILES, ROBUST TO JPEG COMPRESSION AND METADATA REMOVAL

Abstract. The article discusses a method for digitally signing images that does not use metadata or additional files. The method is resistant to jpeg compression. The object of the digital signature is 8x8 pixel blocks after a discrete cosine transform applied, which make up the jpeg file. A method to achieve resistance to jpeg compression is to quantize (reduce the precision) of the values in the blocks so that when calculating the signature, they remain unchanged up to a certain compression level.

Keywords: JPEG, discrete cosine transform, digital signature, compression, images.

Введение

Электронная цифровая подпись (далее – ЭЦП) представляет собой математическую схему, позволяющую верифицировать подлинность и целостность документа, представленного в цифровом виде [1]. Современный криптографический алгоритм цифровой подписи может быть применен к любому цифровому файлу. Большинство реализаций предполагают создание дополнительного файла таким образом, что подписанным документом будет считаться оригинальный файл + файл цифровой подписи. Многие форматы файлов, как, например, PDF, поддерживают встраивание цифровой подписи внутрь исходного файла [2]. В этом случае подписанным документом будет считаться исходный файл со встроенной цифровой подписью.

Однако для цифровых изображений в двух самых популярных форматах – JPEG и PNG – не существует схемы хранения цифровой подписи внутри исходного файла. Таким образом, имеются лишь два варианта реализации цифровой подписи изображения:

1) встраивание подписи в метаданные изображения, например, dSIG-чанки для изображений в формате PNG и метаданные EXIF для изображений в формате JPEG;

2) сохранение отдельного файла цифровой подписи в дополнение к исходному файлу.

Обе реализации имеют недостаток в виде невозможности передачи подписанных изображений через загрузку в популярные социальные сети, мессенджеры и на популярные хостинги изображений. Причина заключается в том, что вышеперечисленные среды вы-

Мазайшвили Евгений Константинович
аспирант, Национальный исследовательский университет ИТМО, Санкт-Петербург. Сфера научных интересов: программная инженерия, искусственный интеллект. Автор пяти опубликованных научных работ. ORCID: 0000-0001-8592-0751.
Электронный адрес: evgenij997@yandex.ru

Авксентьева Елена Юрьевна
кандидат педагогических наук, доцент, доцент факультета программной инженерии и компьютерной техники, Национальный исследовательский университет ИТМО, Санкт-Петербург. Сфера научных интересов: программная инженерия, искусственный интеллект, электронное обучение. Автор более 80 опубликованных научных работ. ORCID 0000-0001-5000-4868, SPIN-код: 2688-1540, AuthorID: 559672.
Электронный адрес: avksentievaelena@rambler.ru

полняют перекодирование изображений, удаляя метаданные. Многие из них выполняют jpeg-сжатие в дополнение к удалению метаданных.

JPEG (Joint Photographic Experts Group) – это спецификация для сжатия цифровых изображений с потерями, основанная на дискретном косинусном преобразовании – discrete cosine transform (далее – DCT) [3]. Использование DCT позволяет перевести изображение из пространственного представления (массива пикселей) в частотное представление (сумму косинусных функций разной частоты). Удаление самых высокочастотных функций и обратное преобразование в пространственное представление приводит к большому коэффициенту сжатия без видимого ухудшения качества изображения.

Целью данной работы является создание метода встраивания цифровой подписи в изображение формата JPEG для передачи через среды, выполняющие перекодирование изображений (Рисунок 1).

Рисунок 1. Ожидаемая схема работы цифровой подписи
Источник: рисунки 1–7 выполнены авторами.

Предлагаемый метод выполняет модификацию самого изображения в виде добавления на его правую грань дополнительных пикселей, содержащих подпись. Подпись должна быть устойчива к jpeg-сжатию в определенных пределах.

Предлагаемый метод

Главная проблема при реализации ЭЦП для подписи изображений с устойчивостью к сжатию – отсутствие гарантии, что после сжатия цвета пикселей изображения останутся без изменений. Соответственно, в качестве объекта подписи нельзя использовать массив всех пикселей. Также нельзя использовать приблизительные значения пикселей как объект цифровой подписи. В этом случае злоумышленник имеет возможность изменять цвета пикселей в определенных пределах, формируя неправильное изображение, не нарушая подписи.

Подпись изображения. Предлагаемый метод подписи состоит из шести шагов.

1. Вычисление массива DCT-блоков из черно-белого исходного изображения.
2. Квантизация DCT-блоков на фиксированную величину.
3. Вычисление таблицы четности для каждого DCT-блока. Эта таблица позволит цифровой подписи остаться валидной при небольших изменениях в DCT-блоке, например, при сжатии.
4. Цифровая подпись массива DCT-блоков алгоритмом RSA.
5. Кодирование полученной подписи и таблиц четности в отдельное изображение (далее по тексту – изображение-подпись, см. Рисунок 8, справа) методом, дающим изображению-подписи устойчивость к jpeg-сжатию.
6. Присоединение изображения-подписи к исходному изображению.

Верификация подписанного изображения. Метод верификации подписи состоит также из шести шагов.

1. Разделение подписанного изображения на исходное изображение и изображение-подпись.
2. Расшифровка данных из изображения-подписи – получение подписи в готовом для верификации виде и таблиц четности.
3. Вычисление массива DCT-блоков из исходного изображения.
4. Квантизация DCT-блоков на фиксированную величину.
5. Внесение поправок в DCT-блоки из таблиц четности.
6. Проверка: совпадает ли прочитанная на шаге 2 цифровая подпись с хешем массива DCT-блоков.

Предлагаемый метод имеет ограничение в виде возможности верифицировать только черно-белые изображения.

Метод имеет некоторую схожесть со стеганографией, основанной на сокрытии данных в DCT-коэффициентах [4]. Коэффициенты анализируются, и результат этого анализа выявляет скрытую изначально информацию – является подпись валидной или нет. Однако данный метод не предназначен для сокрытия информации внутри jpeg-изображений. При взгляде на итоговое изображение будет видно, что оно состоит из двух частей – само изображение и его подпись.

Второстепенной проблемой в данном методе является то, что цифровая подпись должна сама являться изображением и быть присоединенной к подписываемому изображению. Это сделает ее подверженной jpeg-сжатию, которое нарушит подпись. Соответственно, подпись должна устойчиво читаться при сжатии изображения в определенных пределах.

Создание подписи

Исходное изображение может быть как сжатым без потерь, например, в формате png, так и сжатым с потерями в формате jpeg. Далее будет описан процесс для изображения, сжатого без потерь. Процесс сжатия файла формата jpeg отличается лишь отсутствием необходимости вычислять массив DCT-блоков, так как он уже содержится в файле.

Вычисление массива DCT-блоков из исходного изображения. DCT – алгоритм, преобразующий изображение из пространства пикселей в частотное пространство, где каждому блоку 8×8 пикселей ставится в соответствие блок 8×8 коэффициентов [5]. Пример DCT-блока показан на Рисунке 2.

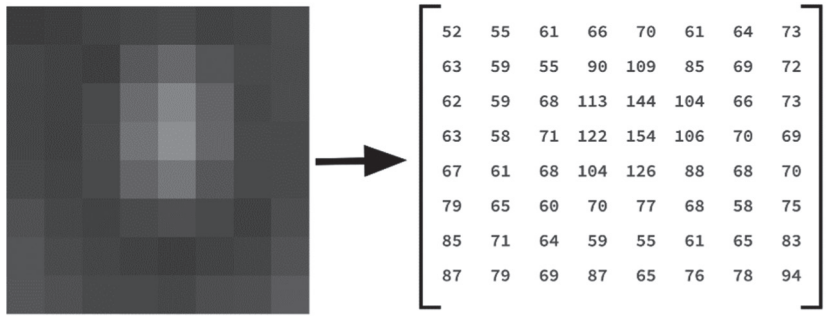

Рисунок 2. Дискретное косинусное преобразование участка 8×8 пикселей

Значения в DCT-блоке имеют диапазон –1023 … 1023 с шагом 1. Большему значению соответствует большая амплитуда соответствующей косинусной функции и бо'льшая яркость определенных пикселей, зависящих от функции.

С точностью до ошибок округления, DCT является алгоритмом преобразования без потерь – таблицу можно преобразовать обратно в исходный массив пикселей.

На этом шаге метода исходное изображение преобразуется в массив DCT-блоков.

Квантизация DCT-блоков на заданную величину. Квантизация является вторым шагом метода. Этот шаг делает значения в DCT-блоках более грубыми, выполняя квантизацию с определенным шагом. Квантизация представляет собой преобразование, где каждое значение внутри DCT-блока преобразуется по формуле

$$value_i = floor\left(\frac{value_i}{Q_i}\right)Q_i\,, \tag{1}$$

где Q_i – значение шага квантизации из таблицы шагов квантизации.

В методе используется переменное значение шага в зависимости от места значения в таблице: шаг увеличивается к нижнему правому углу таблицы. Это сделано по причине того, что алгоритм сжатия jpeg вносит более сильные искажения в значения ближе к нижнему правому углу DCT-блока, где представлены более мелкие детали изображения. Поскольку мы хотим точнее сохранить более явные детали, то для значений ближе к левому верхнему углу таблицы используются меньшие шаги квантизации. Значение в левом верхнем углу имеет больший шаг, так как его изменение приводит лишь к изменению общей яркости блока в итоговом изображении. Используемые значения шага обозначены на Рисунке 3.

$$\begin{bmatrix} 40 & 25 & 25 & 25 & 25 & 40 & 40 & 80 \\ 25 & 25 & 25 & 25 & 40 & 40 & 80 & 80 \\ 25 & 25 & 25 & 40 & 40 & 80 & 80 & 90 \\ 25 & 25 & 40 & 40 & 80 & 80 & 90 & 90 \\ 25 & 40 & 40 & 80 & 80 & 90 & 90 & 90 \\ 40 & 40 & 80 & 80 & 90 & 90 & 90 & 90 \\ 40 & 80 & 80 & 90 & 90 & 90 & 90 & 90 \\ 80 & 80 & 90 & 90 & 90 & 90 & 90 & 90 \end{bmatrix}$$

Рисунок 3. Значение шага квантизации в зависимости от места значения в блоке

Смысл квантизации можно представить как уход от точных значений в блоке и работу с диапазонами значений. Например, значение 529 в блоке при 60-процентном сжатии исказится до 521, но при шаге квантизации 10 они оба будут попадать в интервал 520…530. Аналогично, при шаге 50 оба значения будут попадать в интервал 500…550. Нижнее значение диапазона является результатом квантизации с определенным шагом, как на Рисунке 4.

```
До квантизации:
[[ -98 -227  118  -45   72  -68   48  -24]
 [-197 -165  -42   41   -2   -2   -4   -4]
 [  96   81    3  -24  -15    4   -1    6]
 [ -28  -13   34   37   11    5    4    0]
 [  -8  -19  -26  -13   -1   -3   -6    2]
 [  11   14    4   -7  -12    0    5    0]
 [  -5   -2   10   19   15    4   -1    2]
 [   2   -4  -11  -12   -9    0    3    2]]

После квантизации:
[[-120 -250  100  -50   50  -80   40  -80]
 [-200 -175  -50   25  -40  -40  -80  -80]
 [  75   75    0  -40  -40    0  -80    0]
 [ -50  -25    0    0    0    0    0    0]
 [ -25  -40  -40  -80  -80  -90  -90    0]
 [   0    0    0  -80  -90    0    0    0]
 [ -40  -80    0    0    0    0  -90    0]
 [   0  -80  -90  -90  -90    0    0    0]]
```

Рисунок 4. Значения в одной из DCT-таблиц до и после квантизации (используются шаги квантизации с Рисунка 3)

Квантизованные блоки не переводятся обратно в пиксели и не сохраняются в итоговом подписанном изображении. Они нужны только для этапа вычисления их хеша. Именно хеш от всех квантизованных блоков является объектом цифровой подписи.

Если бы в качестве объекта подписи вместо квантизованных, то есть приблизительных значений DCT-блоков использовались приблизительные значения пикселей с заданной погрешностью, злоумышленник мог бы изменять их в пределах этой погрешности, вырисовывая нужные детали на изображении. В данной работе используются приблизи-

тельные значения, но не пикселей, а DCT-блоков, изменение которых злоумышленником приведет лишь к изменению малозаметных деталей на изображении.

Вычисление таблицы четности для каждого DCT-блока. При квантизации блока может оказаться, что некоторые значения находятся на границе интервала квантизации или близко к границе. Например, для квантизации с шагом 10 значение 50 будет находиться на границе интервалов 40…50 и 50…60. В этом случае даже небольшое искажение при сжатии может сместить значение в неправильный интервал, делая подпись невалидной. Чтобы избежать этой проблемы, каждому из интервалов, определяемых шагом квантизации, присваивается четность, как на Рисунке 5.

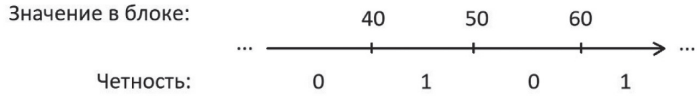

Рисунок 5. Четность для каждого значения в квантизованном блоке

Вместе с цифровой подписью сохраняется четность каждого значения для каждого блока в виде одного бита. Например, при шаге квантизации 10 значение 51 находится в нечетном интервале 50…60, а значение 49 – в четном интервале 40…50.

Таким образом, для каждого квантизованного блока рассчитывается соответствующая ему таблица четности. Эта таблица позволяет на этапе расшифровки делать поправки в значения блока, если значение при сжатии случайно преодолеет границу интервала. Пример таблицы четности показан на Рисунке 6. Таблица занимает 64 бита на каждый участок изображения 8×8 пикселей.

```
[[1 0 0 0 0 0 1 1]
 [0 1 0 1 1 1 1 1]
 [1 1 0 1 1 0 1 0]
 [0 1 0 0 0 0 0 0]
 [1 1 1 1 1 1 1 0]
 [0 0 0 1 1 0 0 0]
 [1 1 0 0 0 0 1 0]
 [0 1 1 1 1 0 0 0]]
```

Рисунок 6. Таблица четности для значений из нижней таблицы на Рисунке 4

Цифровая подпись хеша массива DCT-блоков алгоритмом RSA. Массив всех квантизованных блоков + массив таблиц четности хешируются алгоритмом SHA-256, и получаемый хеш подписывается закрытым ключом согласно алгоритму RSA.

В результате четвертого шага получается всего два артефакта, необходимых для верификации подписи: подписанный хеш размером 2048 бит (при использовании RSA-2048) и массив таблиц четности размером 64 бит × количество блоков 8×8 пикселей в изображении.

Кодирование полученной подписи и таблиц четности в изображение-подпись. Подписанный хеш и таблицы четности должны храниться непосредственно в изображении, и для этого они должны быть устойчивыми к jpeg-сжатию.

Чтобы достичь устойчивости к сжатию, была исследована одна из особенностей работы дискретного косинусного преобразования. DCT-блок представляет собой блок 8×8

значений, от –1023 до 1023. Блок, состоящий только из максимальных и минимальных значений, будет демонстрировать устойчивость к сжатию, так как сжатие такого блока приведет к тому, что в значения будут внесены лишь незначительные искажения. Например, значение –1023 может стать значением –990 после сжатия. Однако близость его к минимальному значению позволит однозначно считать его минимальным значением. Таким образом, изображение, составленное из DCT-блоков с максимальным и минимальным значением, будет устойчиво к очень высокому уровню сжатия.

В данной работе используются три значения: –1023, 0 и 1023. Таким образом, в одном значении DCT-блока в изображении-подписи хранится 2,5 бит полезной информации. Пример такого блока представлен на Рисунке 7.

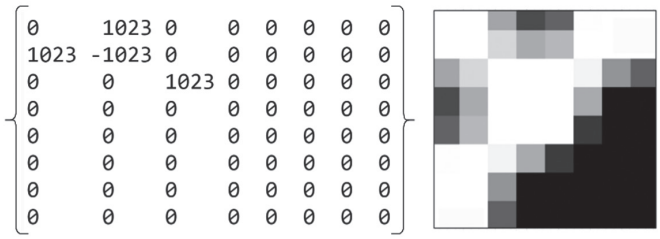

Рисунок 7. Искусственно созданный DCT-блок (слева) и соответствующий ему блок пикселей (справа), который будет включен в финальное изображение-подпись

Таким образом, если кодировать 64 × 2,5 бита в картинку 8×8 пикселей, все таблицы четности уместятся в зону, занимающую не более 1/3 изображения.

Присоединение изображения-подписи к исходному изображению. После кодирования изображение-подпись, содержащее подписанный хеш и таблицы четности, присоединяется к исходному изображению (см. Рисунок 8).

Рисунок 8. Слева направо: исходное изображение, изображение с цифровой подписью
Источник: фотография Marco Bonomo. URL: https://unsplash.com/photos/Sa7787z58VQ/ (дата обращения: 11.01.2024).

Размер области с подписью всегда составляет фиксированную часть изображения, так как количество таблиц четности линейно зависит от размера изображения, а подписанный хеш имеет фиксированную длину.

На этом шаге процесс подписи завершается.

Верификация ЭЦП

Входные данные для этапа верификации – изображение с цифровой подписью. Изображение не содержит метаданных из исходного файла и, вероятно, было сжато алгоритмом jpeg.

Разделение подписанного изображения на исходное и изображение-подпись. Поскольку размер цифровой подписи известен, разделение изображений на исходное и подпись является тривиальным.

Расшифровка данных из изображения-подписи – получение подписи в готовом для верификации виде и таблиц четности. Для расшифровки изображения-подписи каждый участок 8×8 пикселей переводится в DCT-блок, значения которого округляются до ближайшего: 1023, 0 или –1023.

Таким образом, восстанавливаются подписанный хеш и таблицы четности.

Вычисление массива DCT-блоков из исходного изображения. Шаг аналогичен первому шагу при подписывании изображения. Вычисление DCT не требуется, если файл уже находится в формате jpeg, в таком случае достаточно лишь прочитать готовые DCT-блоки из файла.

Квантизация DCT-блоков на фиксированную величину. Шаг квантизации полностью повторяет таковой при подписи изображения. Цель этого шага – получить такие же квантизованные блоки, какие были в изображении на момент подписи, чтобы подпись была валидной.

Внесение поправок в DCT-блоки из таблиц четности. На этом шаге происходит расчет таблиц четности для квантизованных блоков аналогично шагу подписи. Однако в отличие от аналогичного шага при подписи при верификации происходит сравнение таблиц, и уже по результатам сравнения происходит корректировка значений блоков: те значения, которые могли выйти за границу своего интервала, поскольку находились близко к ней, возвращаются в свой интервал.

Если рассчитанная четность значения в блоке не совпадает с соответствующей четностью, записанной в таблице четности в подписи, выполняется коррекция.

Алгоритм данной коррекции описывается формулой

$$v_i = \begin{cases} \left|\hat{v}_i - v_i\right| > \dfrac{Q_i}{2}: & v_i + \dfrac{Q_i}{2}, \\ else: & v_i - \dfrac{Q_i}{2}, \end{cases} \qquad (2)$$

где Q_i – таблица шагов квантизации; v – квантизованное значение в блоке, $\hat{v}$ – значение в блоке до квантизации.

Формула описывает следующее преобразование. Если значение ближе к нижней границе интервала, то оно считается ошибочно попавшим в него из нижестоящего интервала. Интервал в квантизованной таблице меняется на нижестоящий. Если же значение ближе к верхней границе интервала, то оно считается ошибочно попавшим в него из вышестоящего интервала. Интервал в квантизованной таблице меняется на вышестоящий.

Проверка – совпадает ли прочитанная на шаге 2 цифровая подпись с рассчитанным хешем. Последним шагом происходит хеширование массива DCT-таблиц и таблиц

четности из подписи по той же схеме, что и при подписывании изображения, после чего полученный хеш сравнивается с тем, что находится в цифровой подписи. Совпадение хешей указывает на то, что подпись верна. Несовпадение указывает либо на слишком сильное сжатие, либо на намеренное искажение изображения.

Заключение

В работе рассмотрен метод цифровой подписи изображений в формате JPEG, а также в любом формате сжатия без потерь, например PNG. Метод обладает устойчивостью к jpeg-сжатию благодаря использованию в качестве объекта подписи квантизованных DCT-блоков. Устойчивость к сжатию достигается за счет хранения данных цифровой подписи в виде изображения, состоящего из базовых функций DCT максимальной амплитуды.

Метод может использоваться как упрощенная схема верификации изображений, не требующая ничего, кроме самого контента изображения.

В качестве перспективы развития метода можно увеличить плотность данных в изображении-подписи, чтобы оно занимало меньшую часть основного изображения.

Литература

1. *Goldwasser S., Bellare M.* Lecture Notes on Cryptography. Cambridge; Massachusetts, 2008. 289 с. URL: https://pdf4pro.com/view/lecture-notes-on-cryptography-633cfb.html (дата обращения: 11.01.2024).
2. PDF (Portable Document Format), version 1.7, ISO 32000-1, 2008 // Digital Formats. Sustainability of Digital Formats: Planning for Library of Congress Collections. URL: https://www.loc.gov/preservation/digital/formats/fdd/fdd000277.shtml (дата обращения: 11.01.2024).
3. *Buchanan W.J.* DCT (Discrete Cosine Transform) // Asecuritysite.com. URL: https://asecuritysite.com/comms/dct2 (дата обращения: 24.12.2023).
4. *Abdelhamid A.A., Mursi Ahmed M., Alsammak A.K.* Data hiding inside JPEG images with high resistance to steganalysis using a novel technique: DCT-M3 // Ain Shams Engineering journal. 2017. Vol. 9. No. 4. Pp. 1965–1974. DOI: 10.1016/j.asej.2017.02.003
5. *Griffin J.* The Ultimate Guide to JPEG Including JPEG Compression & Encoding // The Webmaster blog. January 04. URL: https://www.thewebmaster.com/jpeg-definitive-guide/ (дата обращения: 10.12.2023).

References

1. Goldwasser S., Bellare M. (2008) Lecture Notes on Cryptography. Cambridge; Massachusetts. 289 c. URL: https://pdf4pro.com/view/lecture-notes-on-cryptography-633cfb.html (accessed 11.01.2024).
2. PDF (Portable Document Format), version 1.7, ISO 32000-1, 2008. *Digital Formats. Sustainability of Digital Formats: Planning for Library of Congress Collections*. URL: https://www.loc.gov/preservation/digital/formats/fdd/fdd000277.shtml (accessed 11.01.2024).
3. Buchanan W.J. (2024) DCT (Discrete Cosine Transform). *Asecuritysite.com*. URL: https://asecuritysite.com/comms/dct2 (accessed 24.12.2023).
4. Abdelhamid A.A., Mursi Ahmed M., Alsammak A.K. (2017) Data hiding inside JPEG images with high resistance to steganalysis using a novel technique: DCT-M3. *Ain Shams Engineering journal*. Vol. 9. No. 4. Pp. 1965–1974. DOI: 10.1016/j.asej.2017.02.003
5. Griffin J. (2023) The Ultimate Guide to JPEG Including JPEG Compression & Encoding. *The Webmaster blog*. January 04. URL: https://www.thewebmaster.com/jpeg-definitive-guide/ (accessed 10.12.2023).