

Е.К. Мазайшвили, Е.Ю. Авксентьева

## УСТОЙЧИВОЕ КОДИРОВАНИЕ ДАННЫХ ВНУТРИ ИЗОБРАЖЕНИЯ ФОРМАТА JPEG

**Аннотация.** Передача машиночитаемых данных в изображениях формата JPEG является нетривиальной задачей. Существует два подхода к ее решению: метаданные файла и изменение самого изображения. Однако невозможно использовать метаданные, например, если изображение является частью анимации или произошло удаление метаданных при загрузке в Интернет. В этом случае единственным решением является кодирование данных в пикселях изображения. В статье рассмотрен метод кодирования данных в пиксели изображения с высокой устойчивостью к JPEG-сжатию.

**Ключевые слова:** JPEG, кодирование данных, изображение, дискретное косинусное преобразование.

E.K. Mazaishvili, E.Yu. Avksentieva

## RELIABLE DATA ENCODING INSIDE JPEG IMAGES

**Abstract.** Transferring machine-readable data inside jpeg images is a non-trivial task. There are two approaches to solving it – file metadata and changing the image itself. However, it is often not possible to use metadata. For example, if the image is part of an animation or metadata was deleted when uploading to the Internet. In this case, the only solution is to encode the data in pixels of the image. The article describes a method for encoding data into pixels of an image with high resistance to JPEG compression.

**Keywords:** JPEG, data encoding, image, discrete cosine transform.

### *Введение*

Для хранения и передачи произвольных бинарных данных внутри изображения существует два подхода. Первый заключается в использовании метаданных файла. Однако не всегда этот подход является возможным: изображение может находиться в контейнере, не поддерживающем метаданные, или может быть составной частью видео или анимации, также метаданные могут быть уничтожены при передаче через социальные сети. Второй подход – кодирование данных напрямую в пиксели изображения как без применения стеганографии, так и с ней [1]. В этом случае возникает проблема потери данных, если изображение будет сжато.

Самый популярный метод сжатия цифровых изображений с потерями – JPEG. Алгоритм использует дискретное косинусное преобразование для уменьшения количества высокочастотных деталей на изображении. Хотя существуют, возможно, лучшие способы сжатия изображений с потерями [2], JPEG значительно обходит их по популярности.

По причине того, что JPEG-сжатие влияет на мелкие детали рисунка, невозможно кодировать полезные данные напрямую в пикселях изображения – их значения могут сильно поменяться даже при незначительном сжатии, как можно видеть на Рисунке 1.

Несмотря на то что существуют методы, направленные на использование стеганографии в JPEG-изображениях [3], плотность данных можно повысить, если не ставить целью скрыть передаваемые данные за другим изображением.

**Цель** данной работы – разработать новый способ кодирования полезных данных непосредственно в пикселях изображения, который будет устойчив к JPEG-сжатию.

**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



**Рисунок 1.** Изменения значений пикселей при JPEG-сжатии с качеством 80 % до (слева) и после (справа)

*Источник:* здесь и далее рисунки выполнены авторами

Предлагаемый метод заключается в кодировании данных не в индивидуальные пиксели изображения, а в блоки  $8 \times 8$  пикселей, которые состояются из базисных функций дискретного косинусного преобразования.

*Кодирование с помощью базисных функций дискретного косинусного преобразования*

Дискретное косинусное преобразование (далее – ДКП), используемое при JPEG-сжатии, преобразует черно-белые значения пикселей изображения (от 0 до 255) в значения амплитуды базисных функций (от  $-1023$  до  $1023$ ). Каждый блок  $8 \times 8$  пикселей преобразуется в блок значений амплитуды 64 функций<sup>1</sup>. При последовательном сжатии значение амплитуды постепенно теряет в точности, пока не станет равным  $0^2$ .

На Рисунке 2 приведен пример работы сжатия.

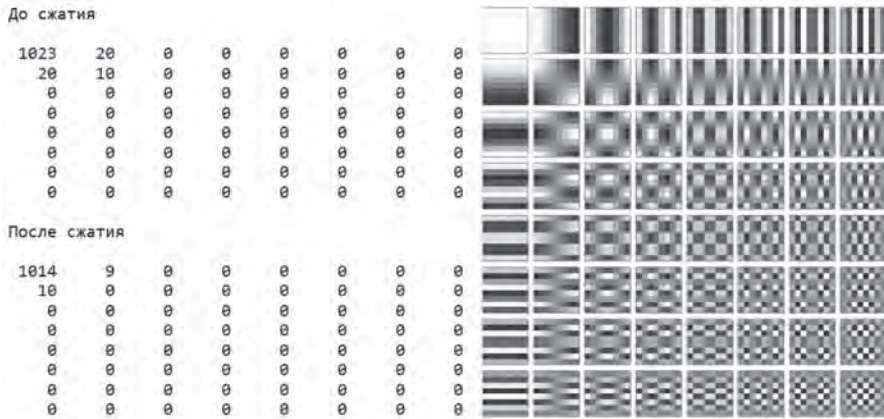
Идея предлагаемого метода состоит в кодировании полезной информации не в значениях пикселей, а в значениях амплитуд базисных функций. На первый взгляд решение

<sup>1</sup> Buchanan W.J. DCT – Discrete Cosine Transform // ASecuritySite.com. 2024. URL: <https://asecuritysite.com/comms/dct2> (дата обращения: 24.05.2024).

<sup>2</sup> The Ultimate Guide to JPEG Including JPEG Compression & Encoding // The Webmaster blog. URL: <https://www.thewebmaster.com/jpeg-definitive-guide> (дата обращения: 30.05.2024).

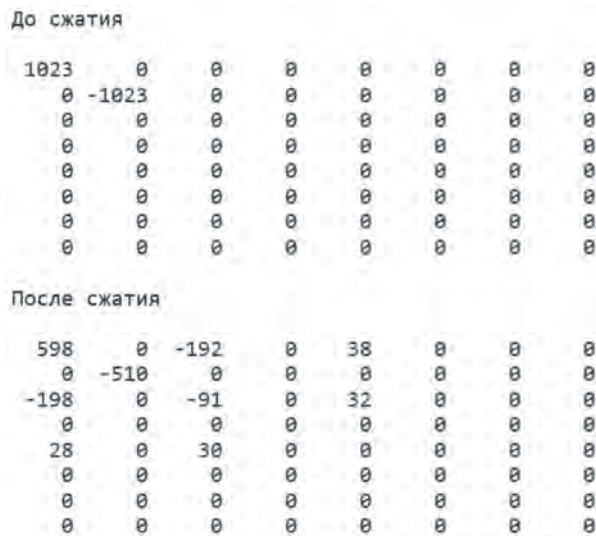
Устойчивое кодирование данных внутри изображения формата JPEG

кажется тривиальным: если для каждой из 64 функций установить два фиксированных значения амплитуды:  $-1023$  или  $1023$ , то можно будет закодировать 64 бита информации, трактуя отрицательное значение как 0, а положительное – как 1. Уровень сжатия до определенного предела не будет иметь значения, так как даже если значение потеряет в точности настолько, что значение  $-1023$  превратится в  $-5$ , по его знаку всё равно можно однозначно трактовать его как 0.



**Рисунок 2.** Дискретное косинусное преобразование при JPEG-сжатии. Значения амплитуд соответствующих базисных функций ДКП до (слева сверху) и после (слева снизу) сжатия на 60 %. Все 64 базисные функции изображены справа

Однако при преобразовании в пиксели и обратно функции влияют друг на друга. На Рисунке 3 можно видеть, как всего два установленных значения до сжатия превратились в 9 различных значений после сжатия и преобразования в пиксели.

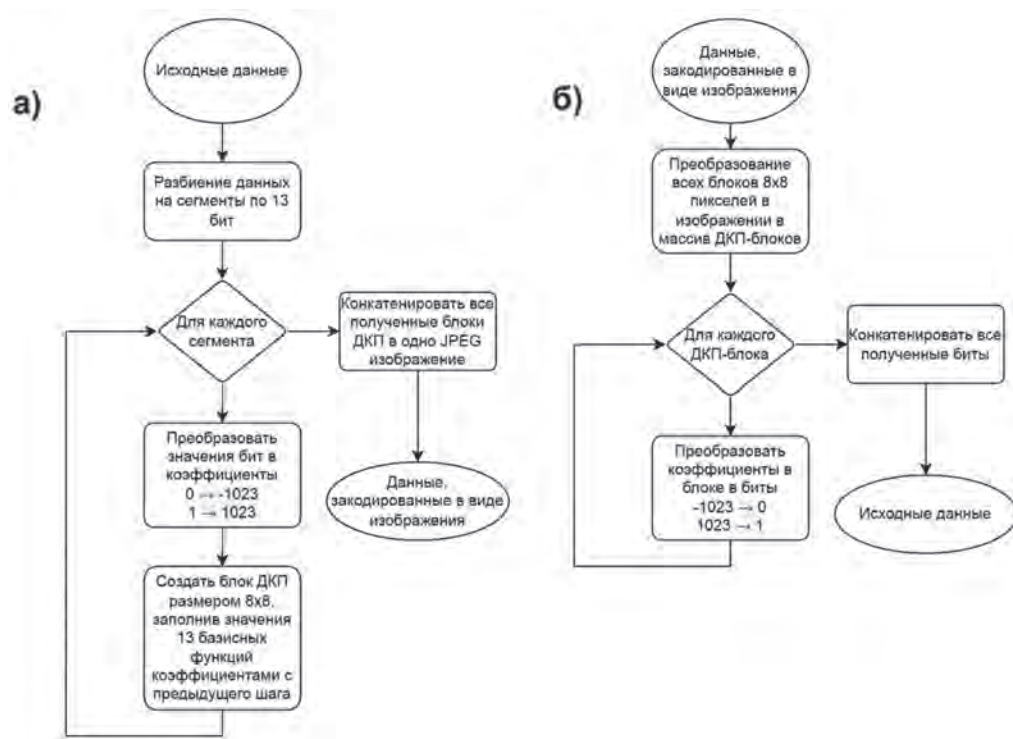


**Рисунок 3.** Влияние базисных функций друг на друга

После сжатия ненулевые значения не только остались у «нужных» функций, но и появились у других.

### Предлагаемый метод

Изображение составляется из базисных функций дискретного косинусного преобразования. Каждая такая функция используется только с двумя значениями амплитуды:  $-1023$  и  $+1023$ . При этом используется определенный набор из 13 функций, не создающих помех друг другу. С помощью 13 значений, каждое из которых может быть трактовано либо как 0, либо как 1, можно закодировать 13 бит информации в каждом блоке  $8 \times 8$  пикселей. Блок-схема метода приведена на Рисунке 4.



**Рисунок 4.** Предлагаемый метод: кодирование (а) и декодирование (б) изображения

То есть при расшифровке каждый коэффициент из 13 переводится в 1 бит согласно формуле

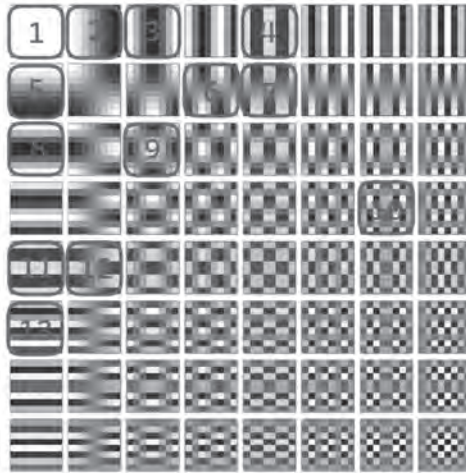
$$\begin{cases} 0 & c_i < 0, \\ 1 & c_i \geq 0, \end{cases}$$

где  $c_i$  – коэффициент при одной из 13 базисных функций.

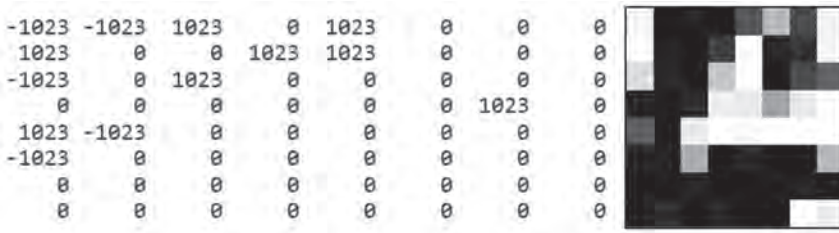
Метод использует 13 базисных функций, изображенных на Рисунке 5. Эти функции были экспериментально найдены как не конфликтующие друг с другом при сжатии, преобразовании в пиксели и обратно.

Пример сообщения, закодированного таким образом, приведен на Рисунках 6 и 7.

Устойчивое кодирование данных внутри изображения формата JPEG



**Рисунок 5.** Используемые 13 из 64 базисных функций для каждого из 13 бит исходных данных (выделены красным и пронумерованы)



**Рисунок 6.** Пример одного блока с 13 битами закодированной информации: значения ДКП (слева) и соответствующее ему изображение (справа)



**Рисунок 7.** Изображение, состоящее из 9 блоков в высоту и 10 в длину. Итоговый размер: 72x80 пикселей. Вместимость: 1170 бит

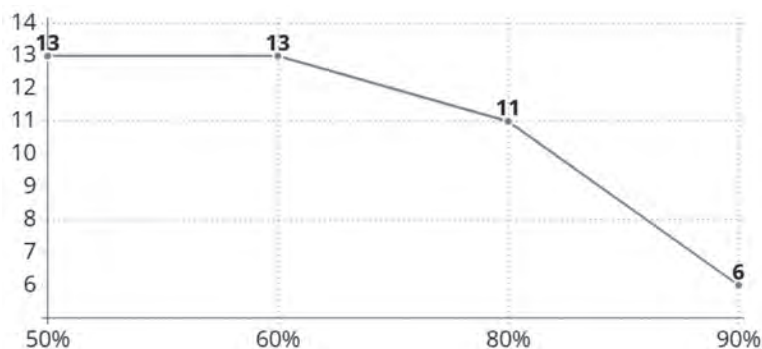
Изображение, представленное на Рисунке 7, может быть сжато алгоритмом JPEG до 60 % и будет корректно расшифровано.

### Эксперимент

Для постановки эксперимента было выбрано 4 степени сжатия изображения – 50, 60, 80 и 90 %. Степенью сжатия названа величина, обратно пропорциональная Quality Factor (QF) в настройках JPEG-энкодера. Степени сжатия 50 % соответствует QF50, 60 % – QF40, 80 % – QF20, 90 % – QF10.

Для каждой степени сжатия создавался блок, заполненный случайными битами. То есть нужные базисные функции в блоке (см. Рисунок 5) устанавливались в случайное значение –1023 или 1023, после чего блок кодировался в изображение, которое сжималось с соответствующей степенью. После проведения декодирования биты не должны были поменяться. Операция повторялась 10000 раз для каждой степени сжатия. Если обнаруживалась потеря бита, количество хранимых бит уменьшалось, и цикл начинался заново, пока не были успешно пройдены все 10000 кодирований и декодирований.

Максимальное стабильное количество бит для каждой степени сжатия приведено на Рисунке 8.



**Рисунок 8.** Количество бит, надежно кодируемых в одном блоке при разных уровнях сжатия. Биты представлены базисными функциями в количестве и порядке, как на Рисунке 5.

### Заключение

Предложенный метод позволяет кодировать 13 бит информации в каждые 64 пикселя изображения, которые могут пережить JPEG-сжатие до 60 %. Степень сжатия можно повысить, уменьшив количество хранимых бит.

Метод обладает рядом других преимуществ перед обычным кодированием информации в пиксели изображения; например, область изображения с закодированной информацией, даже если она была сжата несколько раз, можно легко восстановить, установив амплитуду базисных функций обратно в 1023 или –1023 в зависимости от знака.

Предложенный метод можно использовать, например, для хранения внутри изображения данных о подлинности самого изображения с цифровой подписью [4]. Это позволит создать изображение, устойчивое к сжатию и содержащее цифровую подпись в самих пикселях.

### Литература

1. Частикова В.А., Аббасов Т.О., Аббасова С.С. Методика распознавания скрытой информации в изображениях на основе алгоритмов стеганографии // Вестник Адыгейского государственного

## Устойчивое кодирование данных внутри изображения формата JPEG

университета. Серия 4: Естественно-математические и технические науки. 2020. № 3 (266). С. 40–45. EDN QJLQNO

2. Сай С.В., Зинкевич А.В., Фомина Е.С. Сравнение дискретного косинус и вейвлет-преобразований в системах сжатия raw-изображений // Компьютерная оптика. 2022. Т. 46. № 6. С. 929–938. EDN TUNQAC. DOI: 10.18287/2412-6179-CO-1094

3. Куркина М.В., Пономарев И.В., Строкин Д.И. Стеганографические методы, устойчивые к jpeg сжатию // Известия АлтГУ. 2021. № 1 (117). С. 102–105. EDN UMSLOI. DOI: 10.14258/izvasu(2021)1-17

4. Erfurth S. Digital Signatures for Authenticating Compressed JPEG Images // SCID' 24: Proceedings of the 1<sup>st</sup> Workshop on Security-Centric Strategies for Combating Information Disorder. 2024. Article no. 4. P. 1–12. DOI: 10.1145/3660512.3665522

## References

1. Chastikova V.A., Abbasov T.O., Abbasova S.S. (2020) Methodology for detecting hidden information in images based on steganography algorithms. *Bulletin of Adyghe State University. Series 4: Natural-Mathematical and Technical Sciences*. No. 3 (266). Pp. 40–45. (In Russian).

2. Sai S.V., Zinkevich A.V., Fomina E.S. (2022) Comparison of Discrete Cosine and Wavelet Transforms in Raw Image Compression Systems. *Computer Optics*. Vol. 46. No. 6. Pp. 929–938. DOI: 10.18287/2412-6179-CO-1094 (In Russian).

3. Kurkina M.V., Ponomarev I.V., Strokin D.I. (2021) Steganographic Methods Resistant to JPEG Compression. *Izvestiya of Altai State University*. No. 1 (117). P. 102–105. DOI: 10.14258/izvasu(2021)1-17 (In Russian).

4. Erfurth S. (2024). Digital Signatures for Authenticating Compressed JPEG Images. *SCID' 24: Proceedings of the 1<sup>st</sup> Workshop on Security-Centric Strategies for Combating Information Disorder*. Article no. 4. Pp. 1–12. DOI: 10.1145/3660512.3665522