

Адаптивный алгоритм сжатия изображения на основе стандарта JPEG

Вихляев Сергей Алексеевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Баженов Руслан Иванович

Приамурский государственный университет имени Шолом-Алейхема

к.п.н., доцент, зав. кафедрой информационных систем, математики и методик обучений

Аннотация

В статье рассматривается применение обобщенной модели адаптивного алгоритма сжатия изображений на основе стандарта JPEG. Данный алгоритм ориентирован на сжатие изображений с глубиной цвета 24 бита, по 8 бит на каждую из 3-х компонент или изображений в градациях серого без резких переходов цветов.

Ключевые слова: стандарт JPEG, графические файлы, сжатие изображения, качество изображения, адаптивный алгоритм.

Adaptive image compression algorithm based on JPEG standard

Vihljaev Sergej Alekseevich

Sholom Aleichem Priamursky State University

student

Bazhenov Ruslan Ivanovich

Sholom Aleichem Priamursky State University

Candidate of pedagogical sciences, associate professor, Head of the Department of Information Systems, Mathematics and teaching methods

Abstract

The article considers the application of the generalized model of adaptive image compression algorithm based on JPEG standard. This algorithm is focused on compressing images with a color depth of 24 bits, 8 bits for each of the 3 components or images in grayscale without sharp color transitions.

Keywords: JPEG standard, graphic files, image compression, image quality, adaptive algorithm.

Все существующие методы адаптивного алгоритма сжатия изображений на основе стандарта JPEG основаны на очень простом предположении, что в потоке данных об изображении всегда содержатся избыточные элементы.

Поэтому сам факт сжатия достигается в процессе поиска и кодирования данных избыточных элементов. Набор данных об изображении всегда имеет n кол-во излишней информации, которую можно устранить практически без заметных визуальных искажений. При этом различают основное два вида избыточности.

Статистическая избыточность связана с взаимосвязью и предсказуемостью данных. Такую избыточность можно устранить без потери информации, исходные данные при этом можно полностью восстановить.

И субъективная или визуальная избыточность с частичной потерей данных. Набор излишней информации, который можно убрать из изображения, не нарушая качество.

Результат сжатия зависит как от формата изображения, так и от конкретной реализации алгоритма. Компрессия в JPEG осуществляется за счет плавного перехода цветов в изображении. Принцип работы адаптивного алгоритма заключается в устранении информации, из-за которой визуального восприятия изображения существенно не изменится. Такой подход позволяет добиться высокой степени сжатия, прежде чем визуальное качество изображения станет заметно хуже.

Данный алгоритм оперирует областями 8×8 точек, где яркость и цвет меняются сравнительно плавно. При применении к матрице такой области дискретного косинусного преобразования, теряется $3/4$ полезной информации и получаем сжатие почти в 2 раза. На изображении RGB, как показали тесты, это сказывается несильно.

Проблема применения алгоритма JPEG рассматривалась многими учеными. С.В. Сай провел сравнительный анализ четкости изображений в стандартах сжатия JPEG и JPEG 2000 [1]. Применение корректирующего фильтра для повышения качества изображений, сжатых методом JPEG, показал Е.И.Тимбай [2]. Использование wavelet-преобразования в алгоритме сжатия изображения JPEG реализовал П.А.Богданов [3]. А.А. Кучумов представил алгоритм восстановления матрицы квантования по зашумленному изображению изначально полученному из изображения формата JPEG [4]. Исследование однородности искажений частотных коэффициентов ДКП матрицы, вносимых jpeg-сжатием и медианной фильтрацией цифровых изображений провел М.П.Коваленко [5].

Далее рассмотрим демонстрационную модель адаптивного алгоритма сжатия изображений на основе стандарта JPEG устраняющий субъективную избыточность, не нарушая визуальное качество.

Данное приложение разработана на языке C#. Программа работает на операционной системе Windows 7 и выше. Для запуска требуется установленный на компьютере. NetFramework 4.0 и выше. Отдельно устанавливать приложение не требуется, достаточно запустить исполняемый файл CompressImg.exe. На скриншоте ниже представлено основное окно приложения (рис .1).

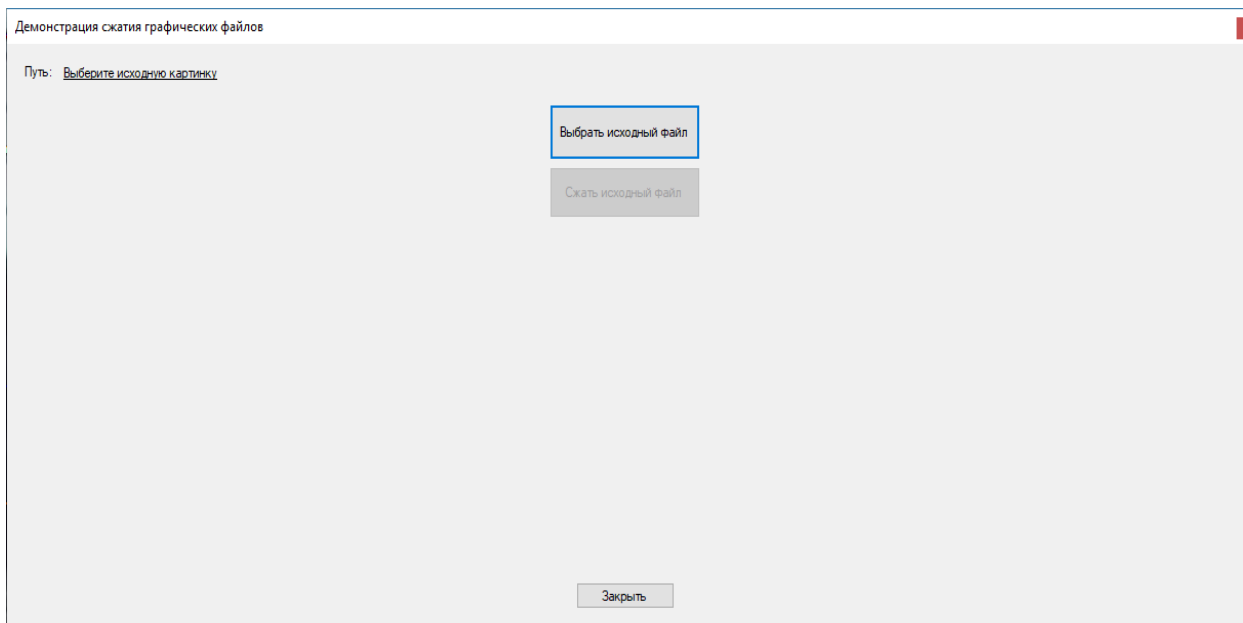


Рисунок 1- Основное окно приложения

В первую очередь необходимо выбрать требуемое изображение для сжатия, в нашем случае это известное тестовое изображение «Lenna.png» разрешением 512x512 и размером 474 Кбайт в формате png (рис. 2).



Рисунок 2 - Тестовое изображение «Lenna.png»

Нажимаем кнопку «Выбрать исходный файл», в диалоговом окне выбираем наше тестовое изображение. В основном окне приложения в левой его области появится выбранное изображение, выше него указаны несколько параметров, а именно путь расположения файла, исходный размер, разрешение картинки в пикселях, а также вычисляемый параметр «Количество информации» для оценки изображения (рис.3).

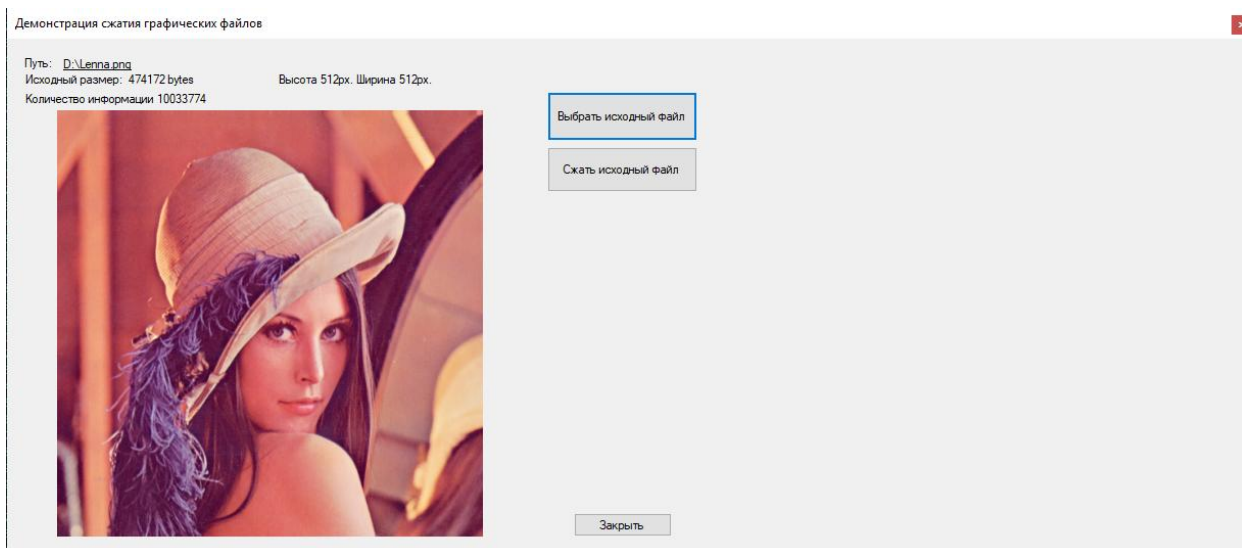


Рисунок 3 - Процесс сжатия исходного изображения «Lenna.png» с помощью использования разработанного демонстрационного модуля

Нажимаем кнопку «Сжать исходный файл», после выполнения операции, картинка в сжатом состоянии появится в левой части окна приложения. Так же будут отображены параметры изображения после его преобразования (рис.4).

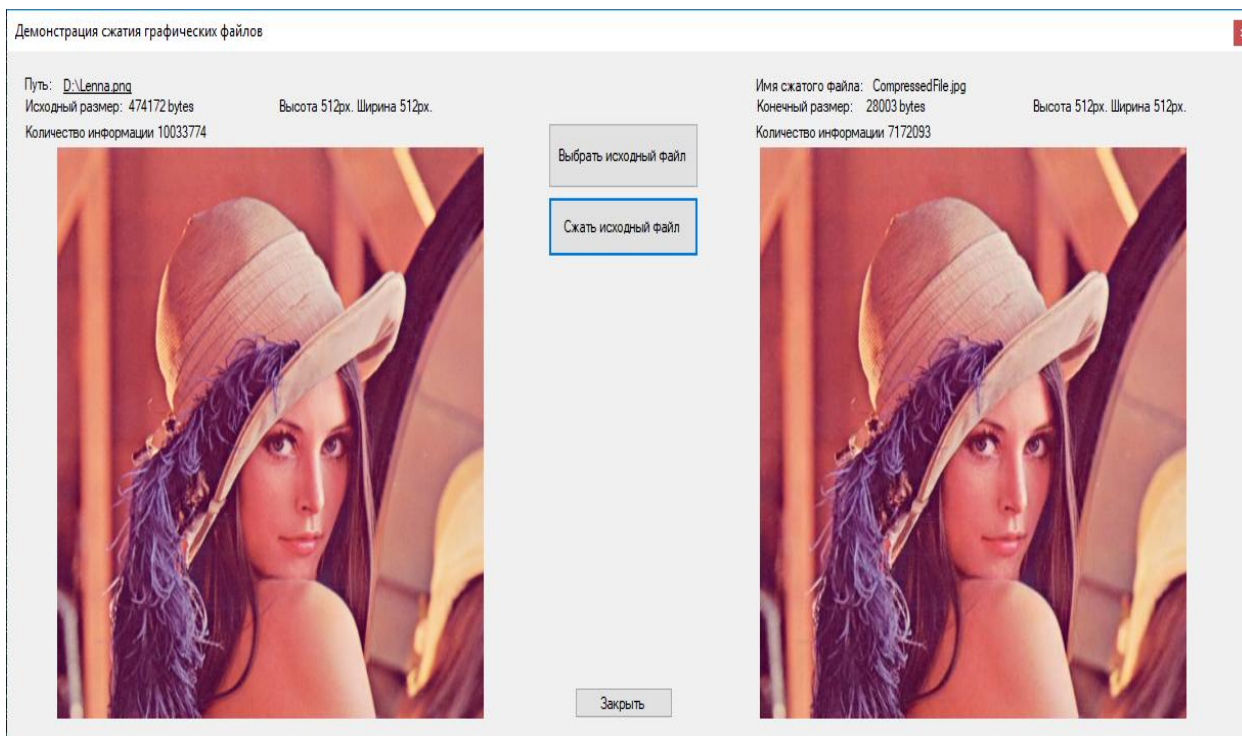


Рисунок 4 - Параметры до и после сжатия исходного изображения «Lenna.png» на экране демонстрационного модуля

Время, затраченное на обработку изображения, составило долю секунды, а размер уменьшился почти в 17 раз. Изображение в результате

сохраняется в формате jpeg, в ту же директорию где и находилось выбранное исходное изображение.

На примере рассмотрим, приближенную часть исходного и сжатого изображения (рис. 5).

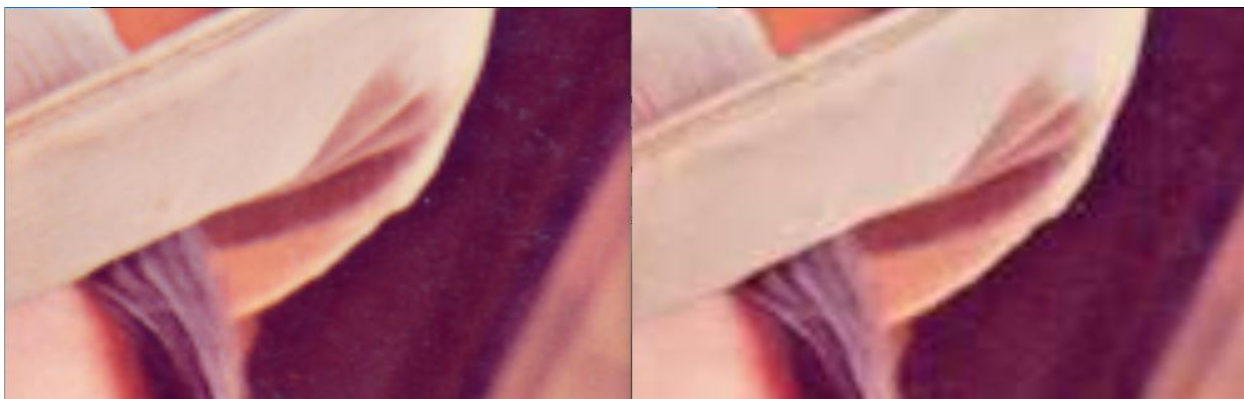


Рисунок 5 - Приближенные части исходного и сжатого изображения

Изображение слева имеет резкие переходы, большую контрастность, соответственно несет больше информации. Изображение справа более замылено, и имеется частичная потеря информации в данном случае не потеряна фактура головного убора.

Таким образом, следует вывод, что адаптивный алгоритм на основе стандарта JPEG является неотъемлемой частью хорошей программы обработки графических файлов. Существует и особый вариант сжатия, без потери качества изображения, подробное описание приводят авторы [6]. Применяют его редко, поскольку его реализация требует значительного времени, а достижимая степень сжатия сильно уступает вариантам с потерями.

Библиографический список

1. Сай С.В. Сравнительный анализ четкости изображений в стандартах сжатия JPEG и JPEG 2000 // Электронные средства и системы управления. 2007. № 1. С. 268-271.
2. Тимбай Е.И. Применение корректирующего фильтра для повышения качества изображений, сжатых методом JPEG // Компьютерная оптика. 2011. Т. 35. № 4. С. 513-518.
3. Богданов П.А. Использование wavelet2преобразования в алгоритме сжатия изображения JPEG // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 9. С. 32-36.
4. Кучумов А.А. Алгоритм восстановления матрицы квантования по зашумленному изображению изначально полученному из изображения формата JPEG // Т-Comm: Телекоммуникации и транспорт. 2011. Т. 5. № 11. С. 51-52.
5. Коваленко М.П. Исследование однородности искажений частотных

коэффициентов ДКП матрицы, вносимых jpeg-сжатием и медианной фильтрацией цифровых изображений // Технические науки - от теории к практике. 2012. № 8. С. 51-57

6. Коробейников А.В., Смирнов В.С., Погудин С.И., Гафаров Р.М., Егоров С.Ф. Итерационное прогнозирование значений пикселей на основе адаптивных линейных нейронов при сжатии изображений без потерь // Интеллектуальные системы в производстве. 2015. №3 (27).