

Министерство образования и науки Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»

*Кафедра компьютерной физики и метаматериалов
на базе Саратовского филиала Института
радиотехники и электроники
имени В. А. Котельникова РАН*

Методы сжатия изображений

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ
студента 4 курса 431 группы
направления 03.03.02 «Физика» физического факультета

Мамедова Сархана Муса-оглы

Научный руководитель

доцент. к.ф.- м.н.,

должность, уч. степень, уч. звание

А.С. Ремизов

инициалы, фамилия

Заведующий кафедрой

д.ф.- м.н., профессор

должность, уч. степень, уч. звание

В.М. Аникин

инициалы, фамилия

Саратов, 2017

Введение

Актуальность работы. Алгоритмы сжатия изображений - бурно развивающаяся область машинной графики. Основной объект приложения усилий в ней - изображения - своеобразный тип данных, характеризуемый рядом особенностей. Разобраться в методах сжатия изображений, произвести сравнение алгоритмов, овладеть необходимой теоретической базой по данной тематике представляется актуальной задачей для выпускника кафедры компьютерной физики. Работа носит методический характер.

Целью данной квалификационной работы является изучение возможностей алгоритмов сжатия изображений и методическое изложение материала. В **задачи** квалификационной работы входит: изложение теоретической базы, необходимой в работе; описание принципов работы и возможностей канонических алгоритмов сжатия без потерь; описание принципов работы и возможностей алгоритмов сжатия изображений.

Структура и объем работы. Выпускная квалификационная работа изложена на 53 страницах, состоит из введения, 3 глав и заключения. Библиографический список включает 9 наименований. Текст содержит 12 таблиц и иллюстрирован 12-ю рисунками.

В **первой главе** приведены сведения из теории информации и кодирования, основные понятия, аксиоматика, дискретные источники, энтропия, теорема кодирования.

Вторая глава посвящена методам сжатия информации без потерь. Подробно и с примерами рассматриваются такие известные алгоритмы как канонический алгоритм Хаффмана, арифметическое сжатие, а также идеи словарных методов и алгоритмы LZ77, LZSS, LZ78.

В **третьей главе** собственно приводится ряд алгоритмов сжатия изображений. Дана классификация изображений и сравнение алгоритмов по различным критериям.

Содержание работы

Алгоритм RLE (Run-length encoding). Идея алгоритма довольно проста. Исходное изображение вытягивается в цепочку байт по строкам раstra. Сжатие в RLE происходит за счет того, что в исходном изображении встречаются цепочки одинаковых байт. Замена их на пары <счетчик повторений, значение> уменьшает избыточность данных. Такой подход хорошо работает с длинными сериями повторяющихся величин. Алгоритм рассчитан на деловую графику — изображения с большими областями повторяющегося цвета.



Рисунок 1. Пример работы алгоритма RLE

На рисунке 1 приведен пример кодирования строки (для наглядности) длиной 21 символ, результат получился длиной 10 символов - сжатие примерно в два раза.

Рекурсивный (волновой) алгоритм. Английское название рекурсивного сжатия - wavelet. На русский язык оно переводится как волновое сжатие, как сжатие с использованием всплесков, а в последнее время и калькой вэйвлет-сжатие. Этот вид архивации известен довольно давно и напрямую исходит из идеи использования когерентности областей. Ориентирован алгоритм на цветные и черно-белые изображения с плавными переходами. Идеален для картинок типа рентгеновских снимков.

Идея алгоритма заключается в том, что мы рекурсивно усредняем значения соседних блоков в изображении, а затем сжимаем полученные коэффициенты, например методом Хаффмана.

По четырем точкам в исходном изображении $a_{2i,2j}, a_{2i+1,2j}, a_{2i,2j+1}, a_{2i+1,2j+1}$ мы получаем другие четыре точки, из которых у последних трех значения заметно меньше исходных:

$$b_{i,j}^1 = (a_{2i,2j} + a_{2i+1,2j} + a_{2i,2j+1} + a_{2i+1,2j+1})/4$$

$$b_{i,j}^2 = (a_{2i,2j} + a_{2i+1,2j} - a_{2i,2j+1} - a_{2i+1,2j+1})/4$$

$$b_{i,j}^3 = (a_{2i,2j} - a_{2i+1,2j} + a_{2i,2j+1} - a_{2i+1,2j+1})/4$$

$$b_{i,j}^4 = (a_{2i,2j} - a_{2i+1,2j} - a_{2i,2j+1} + a_{2i+1,2j+1})/4$$

Затем операция повторяется рассматривая $b_{i,j}^1$ как $a_{i,j}$, в результате с картинкой получается примерно следующее (рисунок 2):

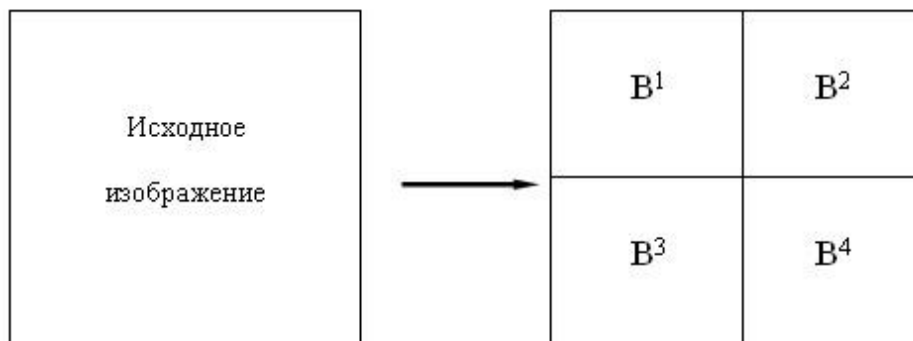


Рисунок 2. Иллюстрация работы волнового алгоритма



Рисунок 3. Иллюстрация работы волнового алгоритма на примере фотографии

К достоинствам этого алгоритма можно отнести то, что он очень легко позволяет реализовать возможность постепенного “проявления” изображения при

передаче изображения по сети. Кроме того, поскольку в начале изображения мы фактически храним его уменьшенную копию, упрощается показ “огрубленного” изображения по заголовку.

Алгоритм JPEG-2000. Разработан той же группой экспертов в области фотографии, что и *JPEG*. основные отличия заключаются в следующем:

1. Лучшее качество изображения при сильной степени сжатия. Или, что то же самое, большая степень сжатия при том же качестве. Фактически это означает заметное уменьшение размеров графики "Web-качества", используемой большинством сайтов.

2. Поддержка кодирования отдельных областей с лучшим качеством. Известно, что отдельные области изображения критичны для восприятия человеком (например, глаза на фотографии), в то время как качеством других можно пожертвовать (например, задний план).

3. Основной алгоритм сжатия заменен на wavelet. Помимо указанного повышения степени сжатия это позволило избавиться от 8-пиксельной блочности, возникающей при повышении степени сжатия. Кроме того, плавное проявление изображения теперь изначально заложено в стандарт (*Progressive JPEG*, активно применяемый в Интернет, появился много позднее *JPEG*).

4. Для повышения степени сжатия в алгоритме используется арифметическое сжатие. Изначально в стандарте *JPEG* также было заложено арифметическое сжатие, однако позднее оно было заменено менее эффективным сжатием по Хаффману, поскольку арифметическое сжатие было защищено патентами. Но сейчас срок действия основного патента истек.

5. Поддержка сжатия без потерь. Помимо привычного сжатия с потерями новый *JPEG* поддерживает и сжатие без потерь. Таким образом, стало возможным использование *JPEG* для сжатия медицинских изображений, в *полиграфии*, при сохранении текста под распознавание *OCR* системами и т.д.

6. Поддержка сжатия однобитных (2-цветных) изображений.

7. На уровне формата поддерживается прозрачность. Кроме того, поддерживается не только 1 бит прозрачности (пиксел прозрачен/непрозрачен), а

отдельный канал, что позволяет задавать плавный переход от непрозрачного изображения к прозрачному фону.

Кроме того, на уровне формата поддерживаются включение в изображение информации о копирайте, поддержка устойчивости к битовым ошибкам при передаче и ширококовещании, можно запрашивать для декомпрессии или обработки внешние средства (plug-ins), можно включать в изображение его описание, информацию для поиска и т.д.

Идея алгоритма

Базовая схема *JPEG-2000* очень похожа на базовую схему *JPEG*. Отличия заключаются в следующем:

1. Вместо дискретного косинусного преобразования (*DCT*) используется дискретное вэйвлет-преобразование (*DWT*).
2. Вместо кодирования по Хаффману используется арифметическое сжатие.
3. В алгоритм изначально заложено *управление качеством* областей изображения.
4. Не используется явно *дискретизация* компонент *U* и *V* после преобразования цветовых пространств, поскольку при *DWT* можно достичь того же результата, но более аккуратно.

Заключение

В заключение, сведем в таблицу параметры различных алгоритмов сжатия изображений.

Алгоритм	Коэффициенты сжатия	Симметрич. по времени	На что ориентирован	Потери	Размерность
RLE	32, 2, 0.5	1	3,4-х битные	Нет	1D
LZW	1000, 4, 5/7	1.2-3	1-8 битные	Нет	1D
Хаффмана	8, 1.5, 1	1-1.5	8 битные	Нет	1D
JBIG	2-30 раз	~1	1-битные	Нет	2D

Lossless JPEG	2 раза	~1	24-битные, серые	Нет	2D
JPEG	2-20 раз	~1	24-битные, серые	Да	2D
Рекурсивное сжатие	2-200 раз	1.5	24-битные, серые	Да	2D
Фрактальный	2-2000 раз	1000-10000	24-битные, серые	Да	2.5D

С одной стороны, приведенные выше алгоритмы достаточно универсальны и покрывают все типы изображений, с другой — у них, по сегодняшним меркам, слишком маленький коэффициент архивации. Используя один из алгоритмов сжатия без потерь, можно обеспечить архивацию изображения примерно в два раза. В то же время алгоритмы сжатия с потерями оперируют с коэффициентами 10-200 раз. Помимо возможности модификации изображения, одна из основных причин подобной разницы заключается в том, что традиционные алгоритмы ориентированы на работу с цепочкой. Они не учитывают, так называемую, “когерентность областей” в изображениях. Идея когерентности областей заключается в малом изменении цвета и структуры на небольшом участке изображения. Справедливости ради следует отметить, что и в классических алгоритмах можно использовать идею когерентности. Существуют алгоритмы обхода изображения по “фрактальной” кривой, при работе которых оно также вытягивается в цепочку; но за счет того, что кривая обегает области изображения по сложной траектории, участки близких цветов в получающейся цепочке удлиняются.

Список использованных источников

1. *Журкин И.Г. Шавенько Н.К.* Сигналы: учеб. пособие по курсу «Автоматизированная обработки аэрокосмической информации. М.: Изд-во МИИГАиК, 2007.
2. *Моциль В.И., Шавенько Н.К.* Основы теории информации: учеб. пособие. М.: Изд-во МИИГАиК, 2006.
3. *Новиков Ф.А.* Дискретная математика для программистов. СПб: Питер, 2008.
4. *Потапов В.* Введение в теорию информации: учеб. пособие. Изд-во НИЦ "Регулярная и хаотическая динамика", 2014. 152 с.
5. *Кудряшов Б.* Основы теории кодирования. учеб. пособие. СПб :БХВ-Петербург, 2016. 400 с.
6. *Сэломон Д.* Сжатие данных, изображений и звука. М. : Техносфера, 2006. 368 с.
7. *Евсютин О., Шелупанов А., Росошек С., Мещеряков Р.* Сжатие цифровых изображений. М. : Горячая Линия - Телеком, 2013. 124 с.
8. *Ватолин Д.* Методы сжатия изображений. Онлайн-учебник. [Электронный ресурс] URL: <http://algotlist.manual.ru/compress/image/fractal/>
9. Алгоритмы сжатия и кодирования изображений. [Электронный ресурс] URL: <http://algotlist.manual.ru/compress/image/>