УДК 004:627
ББК 32.97

МЕТОДЫ КОМПАКТНОГО ПРЕДСТАВЛЕНИЯ ОЦИФРОВАННОГО ТЕКСТА

Иванов В. Г.,

Ломоносов Ю. В.,

Любарский М. Г.

Национальный юридический
университет имени Ярослава Мудрого,
Украина, г. Харьков

Анотація. *У роботі розглядаються методи класифікації, що використовуються при стисненні файла із зображенням тексту, отриманим скануванням або цифровим фотографуванням. Для існуючих на сьогоднішній день алгоритмів класифікації, включаючи добре відомий алгоритм JB2, приведені кількісні характеристики класифікації – кількість класів, що отримуються цими алгоритмами для зображення стандартної сторінки тексту. Чим менша ця кількість, тим якість класифікації вважається вищою, оскільки дає краще стиснення файла із зображенням тексту.*

Ключові слова: *зображення тексту, методи класифікації, стиснення даних.*

Бесценные сокровища литературной, научной, философской мысли, которые накопило человечество за несколько тысяч лет своей истории, хранятся в многочисленных библиотеках в печатном или рукописном виде и нуждаются в переводе в электронную форму. Для этого достаточно много причин. Например, физическое старение многих уникальных экземпляров, дороговизна содержания библиотек и библиотечных хранилищ, малая доступность редких книг, документов, периодики для читателя и многое-многое другое.

Процесс перевода бумажных книг в электронный (цифровой) вид *называется оцифровкой*. Электронные копии книг могут создавать электронные библиотеки, а распространяются они в Интернете. Примером

может служить библиотека «Либрусек» (http://lib.rus.ec), насчитывающая более 100000 экземпляров, и многие другие платные и бесплатные библиотеки. В результате оцифровки получаются *электронные книги*, то есть хранимый в файле текст, оформленный в виде привычной книги. Электронная книга обычно разделена на страницы, которые пронумерованы, имеют поля, иллюстрации и тому подобное.

Еще в недавнем прошлом создание электронной книги происходило только с помощью ручного набора текста, что является крайне трудоемкой и, следовательно, дорогой операцией. В настоящее время оцифровка печатных документов осуществляется с помощью сканера или цифрового фотоаппарата с последующей программной обработкой и сохранением в одном из форматов графических файлов. Этот этап обязателен. На втором, необязательном, этапе производится оптическое распознавание текста (технология OCR), превращающее изображение текста в собственно текст.

Таким образом, важно различать *сканированные* и *вёрстанные* электронные книги.

Вёрстанные книги – это либо материал, подготовленный авторами в каком-либо редакторе, например, во всем доступном MS Word, либо распознанная и вручную вычитанная и отформатированная печатная книга. Конечным результатом является электронная книга в формате PDF (Adobe Portable Document Format), e-Book (Electronic Publication), FB2 (Fiction Book) и многих других. Такие файлы обычно содержат векторные шрифты и иллюстрации высокого качества, поэтому они пригодны для печати в любом разрешении, для просмотра на экране и для поиска по тексту книги, включая возможность выделять и копировать фрагменты текста и иллюстрации. Файлы этого вида обычно называют *векторными*. Типичные векторные PDF-файлы имеют размеры около 10-15 килобайт на страницу, в зависимости от числа формул и иллюстраций. В этом случае становится возможен полнотекстовый поиск по книге и индексация больших массивов электронных книг, однако затрудняется воспроизведение оригинальной

вёрстки, изображений, схем и формул, практически неизбежными становятся ошибки распознавания. Нынешнее состояние программ оптического распознавания заставляет форматировать всё это вручную и исправлять многочисленные ошибки в распознанном тексте. Поэтому для большинства печатных книг гораздо легче делать *растровые*, а не векторные электронные версии.

Растровая версия печатной книги представляет собой набор изображений каждой ее страницы. Даже в чисто текстовых книгах – без иллюстраций, таблиц или формул – оптическое распознавание порой даёт трудно выявляемые ошибки. А в растровых книгах полностью сохраняется оригинальная вёрстка и исключаются какие-либо ошибки. Изготовление растровой электронной книги очень дешево, так как основные трудозатраты приходятся на сканирование страниц исходной бумажной книги. Однако невозможен контекстный поиск или извлечение фрагментов текста, например, для цитирования. Еще один недостаток – без специального сжатия растровая книга занимает очень много места. Поэтому в последнее время усиленно ищутся специальные алгоритмы сжатия изображений страниц, которые в основном содержат текст, но могут включать иллюстрации, схемы, формулы. В этом направлении уже достигнуты результаты. Например, средний размер растровых книг в формате DjVu [4; 7; 10] – 13 КБ на страницу, то есть примерно столько же, сколько и в векторном варианте.

Есть промежуточный путь. Некоторые программы позволяют делать файлы формата PDF [8; 9], в которых весь плохо распознанный материал содержится в растровом виде, а остальная часть – в векторном. Такие PDF файлы однако сильно проигрывают чисто растровым книгам и по внешнему виду (нестыковка векторных шрифтов и фрагментов изображения страницы), и по размеру файлов. Так что истина не всегда посередине.

Из сказанного можно сделать вывод, что массовый перевод печатной продукции прошлых лет в векторную электронную форму – это слишком дорогой путь, по крайней мере, пока программы оптического распознавания

не будут существенно усовершенствованы. Однако ждать этого также перспективно, как и появления хороших электронных переводчиков с одного языка на другой. Остается единственный путь – улучшение сжатия растровых изображений текста.

В этом направлении сделаны определенные шаги, начиная от уже показавших свою практическую ценность форматов PDF и DjVu и заканчивая алгоритмами [1–3; 5; 6], находящимися еще в стадии разработки.

Сжатие изображения текста на основе выделения символов и их классификации

Высокие результаты, демонстрируемые алгоритмом JB2 (формат DjVu), объясняются тем, что он использует классификацию символов. Вообще идея сжатия информации с помощью классификации очень проста и идеально подходит для сжатия изображений текста.

Пусть необходимо сжать некую информацию, которую можно разбить каким-то образом на элементы. Если эти элементы информации объединить в классы так, чтобы в каждом классе находились тождественные (pattern matching) или почти тождественные (soft pattern matching) элементы, то нет нужды хранить все элементы информации – достаточно хранить только по одному элементу из каждого класса. Совокупность таких элементов – представителей классов – называется *словарем*. Кроме того, для восстановления информации нужно еще иметь таблицу, называемую «картой размещения классов», которая для каждого класса указывает, в каком месте исходной информации находятся его элементы.

Ясно, что степень сжатия информации с помощью классификации тем выше, чем меньше классов образуется при классификации и чем больше элементов в каждом классе. В случае сжатия изображения бинарного (далее черно-белого) текста естественным элементом информации является изображение отдельного символа (буквы, цифры, знака препинания и т. п.). Выделение символов не представляет собой особо трудную задачу. Во всех известных алгоритмах, включая алгоритм JB2, символы выделяются как

связанные области, состоящие из черных точек.

Следует заметить, что при этом некоторые грамматические символы распадаются на части (например, буква «ё» дает три символа), а некоторые (например, сочетания вида «fh») объединяются в один. Кроме того, метод непригоден для текстов с псевдорукописным шрифтом. Сжатие таких текстов алгоритмом JB2 и другими катастрофически низкое. Однако не это представляет собой главную трудность при классификации уже разделенных символов.

На рис. 1, взятом из работы [2], представлены три случайно выбранные изображения буквы «n» из различных 257, входящих в изображение страницы текста формата A4, при разрешении сканирования 300 dpi.

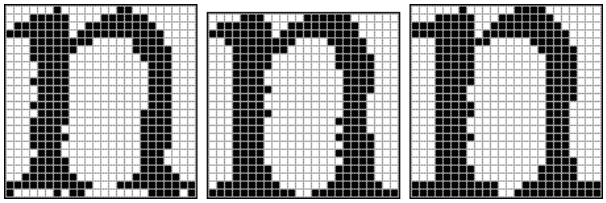

Рис. 1. Влияние шумов печати и сканирования на изображения символа «n»

Легко верится, и это действительно так, что на странице не найдется ни одной пары символов «n», полностью совпадающих друг с другом. То же, за редким исключением, относится и к другим символам, даже точкам. Причиной этого явления есть шумы (то есть случайные искажения), возникающие при печати страницы и ее последующем сканировании. Шумы печати, в основном, вызваны диффузией краски, жидкой или твердой, вдоль хаотически расположенных капилляров бумаги, а шумы сканирования – несовпадением контуров символа с матрицей сканера, подобно тому, как прямая наклонная линия на экране монитора отображается «ступеньками».

Человеку легко заметить, что все три изображения, приведенные на рис. 1, представляют собой букву «n». Однако пока не существует алгоритма, который мог бы установить тождественность этих символов с той же достоверностью, что и человек. Это и есть главная трудность, не позволяющая разбить изображения символов на классы, так чтобы

одновременно выполнялись два условия:

Условие 1. В каждом классе находятся изображения только одного и того же символа.

Условие 2. Все изображения какого-либо символа находятся в одном классе.

Все алгоритмы классификации являются тем или иным компромиссом между этими условиями, причем условие 1 должно выполняться достаточно жестко, иначе в восстановленном тексте будут перепутаны символы, чем иногда грешит алгоритм JB2 (например, иногда путает между собой буквы «b» и «h»). Соблюдение условия 1 влечет за собой ужесточение алгоритма сравнения изображений символов, так что условие 2, практически, невыполнимо. Это приводит к появлению значительно большего числа классов, чем количество символов, изображенных на странице, так как практически все символы дают по нескольку классов своих изображений. Чем больше при классификации образуется классов, тем больше словарь и (логарифмически) больше карта расположения классов. Как следствие, понижается степень сжатия. И хотя алгоритм JB2 и другие используют те или иные методы дополнительного сжатия словаря и карты, эффективность алгоритма в целом определяется *качеством классификации*, то есть количеством получившихся классов, которое в идеале (условие 2) должно совпадать с количеством символов, присутствующих в тексте, чье изображение сжимается. В таблице 1 из работы [2] для различных разрешений сканирования показано количество классов, полученных предлагаемым в этой работе алгоритмом классификации (предпоследний столбец) и алгоритмом JB2 (последний столбец).

Первый столбец показывает разрешение, использованное при сканировании одной и той же страницы формата A4 с черно-белым текстом, набранным шрифтом Times New Roman, 12 pt. Второй столбец – количество классов при тождественной классификации, то есть классов, состоящих из полностью совпадающих изображений символов.

Таблица 1

Количество классов при классификации

Разрешение сканирования (dpi)	Количество классов в исходном изображении	Количество классов классификации	Количество классов после классификации алгоритмом JB2
600 dpi	3558	72	314
500 dpi	3557	72	259
400 dpi	3557	71	199
300 dpi	3545	95	235
200 dpi	3890	148	451

Из таблицы следует несомненное превосходство алгоритма ИЛЛ – будем для краткости называть так алгоритм, предложенный в упомянутой работе [2], – над алгоритмом JB2. Словарь ИЛЛ получается почти в три раза короче, чем словарь JB2. Отсюда вытекает и превосходство в степени сжатия той же страницы, что показывает следующая таблица, относящаяся к той же странице, что и таблица 1.

Таблица 2

Сравнительная степень сжатия изображения текста рассматриваемыми методами

Разрешение сканирования (dpi)	200	300	400	500	600
Исходный размер файла (kb)	505,3	1080,2	2003,9	3111,2	4498,0
Методы	Размер файла после сжатия (kb) / Коэффициент сжатия				
JPEG 2000	132,8/3,8	288,6/3,74	532,4/3,76	830,0/3,75	1200,3/3,7
JBIG2	61,4/8,2	96,1/11,2	119,6/16,7	148,9/20,9	178,9/25,1
JB2	9,6/52,6	8,7/124,1	9,9/202,4	11,4/272,9	13,6/330,7
ИЛЛ	8,1/62,3	8,0/135,0	8,0/250,4	8,8/353,5	10,3/436,7

Не слишком существенное различие между коэффициентами сжатия, продемонстрированное благодаря алгоритмам ИЛЛ и JB2, объясняется тем, что авторы алгоритма ИЛЛ интересовались только классификацией выделенных изображений символов и не оптимизировали алгоритм

дополнительного сжатия словаря и карты размещения классов (использовался универсальный алгоритм без потерь 7z).

Кроме того, таблица 2 показывает, что применение лучшего для сжатия размытых изображений алгоритма JPEG 2000 мало что дает при сжатии изображения черно-белого текста без иллюстраций.

Исходя из сказанного можно сделать такие выводы. В настоящее время самыми мощными алгоритмами сжатия двуцветного изображения текста являются те, которые используют классификацию выделенных символов. При этом конечный результат – степень сжатия изображения – больше всего зависит от качества классификации, то есть количества полученных классов при непременном условии, что в каждый класс входят изображения только одного символа. Основным препятствием к достижению идеальной классификации, в которой классов ровно столько, сколько различных символов встречается в тексте, являются шумы печати и сканирования, искажающие изображения символов. И хотя человек легко справляется с такой задачей (интересно, как мы это делаем?), пока не найдена мера отличия двух сравниваемых изображений символов, позволяющая сделать то же самое.

Еще один аспект при обсуждении любого метода сжатия – качество восстановленного изображения по сравнению с исходным сжимаемым оригиналом. Обычно, чем выше степень сжатия, тем это качество хуже. Методы сжатия сканированных изображений текста, основанные на классификации выделенных символов, позволяют получать восстановленные изображения символов более высокого качества, чем оригинальные. Причем, чем лучше проведена классификация, тем больше сжатие и тем лучше качество изображения символов. Этот парадоксальный факт объясняется очень просто. При классификации выделенных символов, если она обладает высоким качеством, получаются классы, состоящие из большого числа изображений одного и того же символа. Подходящая статистическая

обработка этого класса позволяет избавиться от искажений, привнесенных при печати и сканировании и имеющих случайный характер.

Описанные выше меры отличия, учитывающие контурный характер шумов печати и сканирования, являются достаточно сложными. Но все же есть уверенность, что они будут существенно улучшены, так чтобы проводимая с их помощью классификация выделенных изображений была близка к идеальной.

Список использованных источников

1. Автоматический анализ сложных изображений : сб. переводов / под ред. Э. М. Бравермана. – М. : Мир, 1969. – 310 с.

2. Иванов В. Г. Сжатие изображения текста на основе выделения символов и их классификации / В. Г. Иванов, Ю. В. Ломоносов, М. Г. Любарский // Проблемы управления и информатики. – 2010. – № 6. – С. 111–122.

3. Иванов В. Г. Сжатие изображения текста на основе формирования и классификации вертикальных элементов строки в графическом словаре символьных данных / В. Г. Иванов, М. Г. Любарский, Ю. В. Ломоносов // Проблемы управления и информатики. – 2011. – № 5. – С. 98–109.

4. Книгосканирование и формат PDF [Электронный ресурс] // Создание книг в электронном виде из бумажных книг (в формате DjVu). – Режим доступа : http://www.djvu-soft.narod.ru/scan/bookscan_pdf.htm (дата обращения: 03.03.2014). – Загл. с экрана.

5. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео / Д. Ватолин, А. Ратушняк, М. Смирнов, В. Юкин. – М. : ДИАЛОГ-МИФИ, 2002. – 384 с.

6. Прикладная статистика : классификация и снижение размерности : справ. изд. / С. А. Айвазян, В. М. Бухштабер, И. С. Енюков, Л. Д. Мешалкин. – М. : Финансы и статистика, 1989. – 607 с.

7. DjVu [Электронный ресурс] // Википедия : свобод. энцикл. – Режим доступа : http://ru.wikipedia.org/wiki/DjVu (дата обращения: 03.03.2014). – Загл. с экрана.

8. JBIG2 [Электронный ресурс] // Википедия : свобод. энцикл. – http://ru.wikipedia.org/wiki/JBIG2 (дата обращения: 03.03.2014). – Загл. с экрана.

9. Portable Document Format [Электронный ресурс] // Википедия : свобод. энцикл. – Режим доступа : http://ru.wikipedia.org/wiki/Portable_Document_Format (дата обращения: 03.03.2014). – Загл. с экрана.

10. Specification of DjVu image compression format [Electronic resource] : version of 1999-04-29 15:46 EDT. – Way of access : http://djvu.cvs.sourceforge.net/djvu/djvulibre-3.5/doc/djvu2spec.djvu (date of appeal: 03.03.2014). – Title from the screen.

Аннотация. *В работе рассматриваются методы классификации, применяемые при сжатии файла с изображением текста, полученным сканированием или цифровым фотографированием. Для существующих на сегодняшний день алгоритмов классификации, включая хорошо известный алгоритм JB2, приведены количественные характеристики классификации – число классов, получаемых этими алгоритмами для изображения стандартной страницы текста. Чем меньше это число, тем качество классификации считается выше, так как дает лучшее сжатие файла с изображением текста.*

Ключевые слова: *изображение текста, методы классификации, сжатие данных.*

Annotation. *In paper the methods of classification applied at compression of a file with the image of the text, are considered by the received scanning or digital photographing. For algorithms of classification known for today, including well-known algorithm JB2, quantitative characteristics of classification - number of the classes received by these algorithms for the image of standard page of the text are resulted. The less this number, the quality of classification is considered above as gives the best compression of a file with the text image.*

Key words: *the text image, classification methods, compression of data.*