УДК 004:627

ЭФФЕКТИВНЫЙ МЕТОД ПРЕДСТАВЛЕНИЯ И ТРАНСЛЯЦИИ ЦИФРОВЫХ БИБЛИОТЕЧНЫХ ФОНДОВ В СОВРЕМЕННОМ ИНФОРМАЦИОННОМ ПРОСТРАНСТВЕ

Ломоносов Юрий Вячеславович

кандидат технических наук, доцент,
Національний юридичний університет імені Ярослава Мудрого,
Україна, м. Харків
e-mail: lomonosov@ukr.net
ORCID: 0000-0002-6115-6194

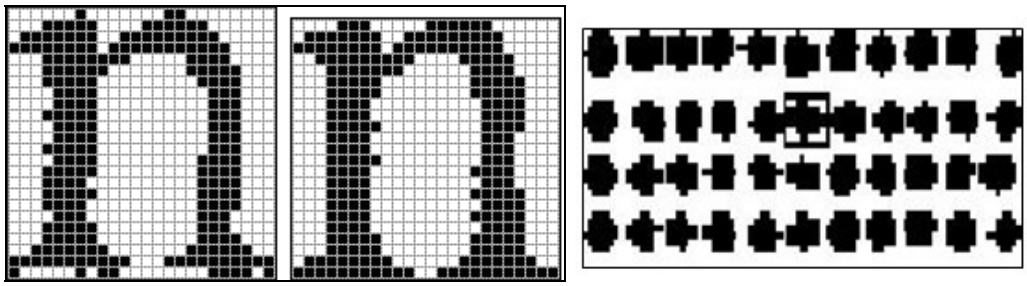
Анотація. *Запропоновано новий метод стиснення бітональних зображень тексту, при якому як основні елементи обробки розглядаються не зв'язані частини символів, що входять до зображення тексту, а вертикальні елементи рядків цього зображення. Пропонований алгоритм дозволяє отримати ступінь стиснення на 30–40 % вище, ніж алгоритм JB2 (формат DjVu) для найбільш популярної роздільної здатності сканування.*

Ключові слова: *стиснення зображення тексту, вертикальні елементи рядки, статистичний аналіз, нечітка класифікація.*

Введение и постановка задачи. В настоящее время лучшие алгоритмы для сжатия битональных изображений текста основаны на выделении изображений символов и их классификации. Это алгоритмы JB2 и JBIG2, используемые, соответственно, в широко распространённых форматах DjVu и PDF. Степень сжатия информации с помощью методов классификации тем выше, чем меньше классов образуется при классификации и чем больше элементов в каждом классе [1, 5]. В этом смысле алгоритм JB2 превосходит алгоритм JBIG2.

В идеале при классификации символов текста изображения каждого символа должны находиться в одном и только одном классе. Однако ни один из известных алгоритмов этому условию не удовлетворяет. Дело в шумах (случайных искажениях), возникающих при печати страницы и ее

последующем сканировании. На рис. 1, а, представлены два случайно выбранные изображения буквы «n» из различных 257, входящих в изображение страницы текста формата А4, при разрешении сканирования 300 dpi. Легко верится, и это действительно так, что на странице не найдется ни одной пары символов «n», полностью совпадающих друг с другом. То же относится и к другим символам, даже точкам, рис. 1, *б*.



а б

Рис. 1. Влияние шумов на изображения символов:
а – искажения символа «n»; б – искажения символа «точка»

И хотя человек без труда может правильно разбить изображения символов на классы, формализовать его действия пока не удалось. Имеющиеся алгоритмы классификации отводят несколько классов для изображений одного и того же символа, что уменьшает степень сжатия изображения. Кроме того, в один класс иногда попадают изображения разных символов. Так, алгоритм JB2 иногда «путает» буквы «b» и «h».

Указанные недостатки алгоритмов, классифицирующих изображения символов, наталкивают на мысль о том, что, хотя выбор изображений символов в качестве элементов изображения страницы является естественным, этот выбор все равно не оптимальный.

В работе [2], в качестве классификации элементов изображения страницы рассматриваются вертикальные элементы ее строк. Результаты этой работы отображают новый подход к сжатию графических текстовых данных на основе статистических методов анализа и классификации совокупности вертикальных элементов строки изображения.

Основной принцип сжатия информации методом классификации состоит в следующем. Пусть информацию можно разбить каким-то образом на элементы. В случае изображения текста естественными элементами являются изображения символов – букв, цифр, знаков препинания. Если эти элементы информации объединить в классы так, чтобы в каждом классе находились тождественные (или почти тождественные) элементы, то нет нужды хранить все элементы информации – достаточно хранить только по одному представителю каждого класса. Совокупность этих представителей называется *словарем*. Кроме того, для восстановления информации нужно еще составить таблицу, называемую *картой размещения классов*, которая для каждого класса указывает, где в исходной информации находятся его элементы.

Новый подход к сжатию графических текстовых данных заключается в следующем. Если представить себе прямоугольник, охватывающий какую-либо строку, то *вертикальным элементом* этой строки будем называть пересечение прямоугольника с любой вертикальной линией шириной в один пиксель. На рис. 2 показано разбиение изображения буквы «е» на вертикальные элементы строки.

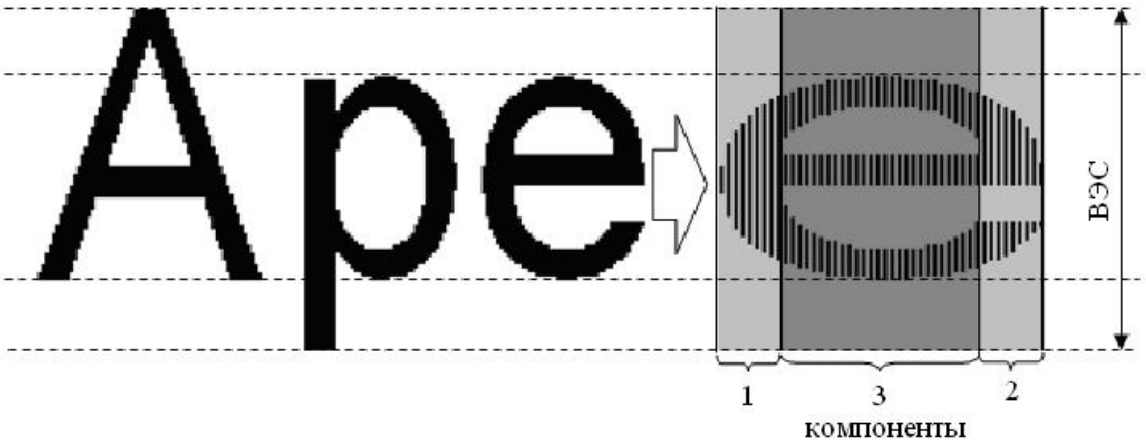


Рис. 2. Изображение буквы «е» и составляющие его вертикальные элементы строки с различным числом компонент

Таким образом, страницу текста можно рассматривать как упорядоченную совокупность вертикальных элементов. Такое разбиение удобно тем, что все вертикальные элементы имеют один и тот же размер и их

можно представлять и как двоичные числа, и как векторы с координатами 0 (черный пиксель) и 1 (белый пиксель).

Шумы печати и сканирования случайным образом искажают вертикальные элементы. Таким образом, среди этих элементов могут быть искаженные и неискаженные. Однако бессмысленно разбивать совокупность вертикальных элементов, составляющих изображение страницы, на классы тождественных или почти тождественных элементов, поскольку многие из них могут быть искажениями сразу нескольких неискаженных элементов. Более того, встречаются пары неискаженных элементов, которые совпадают с искажениями друг друга.

Имеет смысл говорить только о нечеткой классификации вертикальных элементов, то есть о вероятности того, что данный элемент есть искажение того или иного неискаженного элемента. При этом вопрос о том, является ли какой-то элемент неискаженным, тоже имеет лишь вероятностный ответ.

Таким образом, основная задача статистического анализа совокупности вертикальных элементов, представляющих текстовую страницу, ставится так: *по имеющейся на странице совокупности* $\tilde{X}$ *вертикальных элементов указать минимальную наиболее правдоподобную совокупность* $C \subset \tilde{X}$ *неискаженных элементов, а также для каждой пары* $x \in \tilde{X}$ *и* $c \in C$ *найти вероятность того, что данный элемент* x *является искажением элемента* c.

После нахождения этих вероятностей легко получить правильную классификацию изображений символов, представив последние как упорядоченный набор вертикальных элементов. Грубо говоря, изображения двух символов можно отнести к одному классу, если у каждой пары вертикальных элементов, составляющих эти изображения и имеющих один и тот же порядковый номер, достаточно велика вероятность того, что они являются искажениями одного и того же вертикального элемента.

Классификация изображений символов, основанная на нечеткой классификации вертикальных элементов строки

Окончательным этапом обработки изображения текста является классификация изображений символов в изображении текста. Под изображением символа понимается упорядоченная совокупность вертикальных элементов строки, не содержащая пробелов и ограниченная пробелами слева и справа.

Два изображения символов считаются изображениями одного и того же символа, если их можно совместить так, чтобы достаточно много пар совмещенных вертикальных элементов были близкими. Точнее, множество таких пар должно быть достаточно плотным, то есть число подряд идущих пар вертикальных элементов, не являющихся близкими, не должно превосходить некоторого параметра, пропорционального разрешению сканирования.

Описанный критерий позволяет разбить совокупность всех изображений символов на классы, каждый из которых в идеале должен соответствовать одному и только одному символу. Количество полученных классов при различном объеме текста приведено в табл. 1, где для сравнения дополнительно указано количество классов, полученных алгоритмом JB2, а также действительное количество классов в неискаженном тексте.

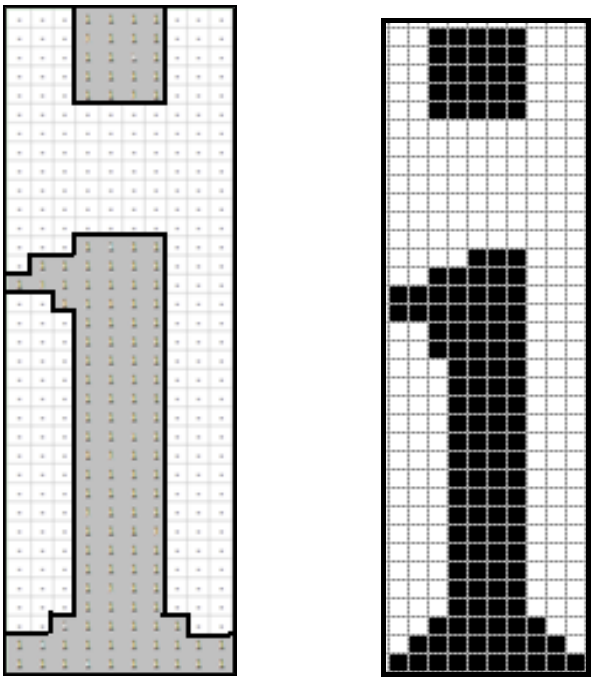
Таблица 1

Количество классов при различном объеме текста

Объем текста в символах	160	240	320	480	960	1920	2880	3840
Предлагаемый алгоритм	31	39	40	44	47	54	66	67
Алгоритм JB2	29	42	46	51	84	121	174	235
Действительное кол-во классов	26	37	38	42	43	49	57	58

После разбиения всех изображений символов на классы в каждом из них находится усредненное изображение, которое в дальнейшем будет представлять этот класс. Процедура усреднения состоит в наложении друг на друга всех изображений класса, совмещающем их «центры тяжести», и

вычислении среднего значения яркости для каждой точки с последующим округлением. На рис. 4,а представлен полученный результат для класса, содержащего изображения символа «i» Для сравнения на рис. 4,б приводится неискаженное изображение этого символа.



а *б*

Рис. 4. Изображения символа «i»: *а* – усредненное в классе изображение символа «i»; *б* – неискаженное изображение символа «i»

Усредненные изображения символов формируют словарь символов, который дополняется картой расположения символов в тексте. Эти два объекта и представляют собой закодированное изображение текста.

Практические результаты кодирования

Для сравнения работы рассматриваемого алгоритма и лучшего на сегодняшний день алгоритма сжатия изображений текста (алгоритма JB2) был выбран битональный текст в электронном виде с параметрами: шрифт – Times New Roman, кегль – 12, число символов – 3840 (страница формата А4). Этот текст был распечатан на черно-белом лазерном принтере, а затем отсканирован с параметрами: формат изображения – *.bmp, глубина цвета – 1 бит, разрешение 300 dpi.

Результаты классификации изображений символов приведены в табл. 1. Количество классов, полученных предлагаемым алгоритмом, близко к истинному и, начиная примерно с объема текста 10^3 символов, существенно меньше, чем при работе алгоритма JB2. С ростом объема текста этот эффект усиливается, что объясняется увеличением статистики вертикальных элементов строки.

Сравнительно большое количества классов, даваемых алгоритмом JB2, приводит к тому, что сформированные им словарь и карта расположения классов занимают значительно больший объем, чем у рассматриваемого алгоритма. Поэтому последний имеет лучший коэффициент сжатия, чем алгоритм JB2, что можно проследить на рис. 5.

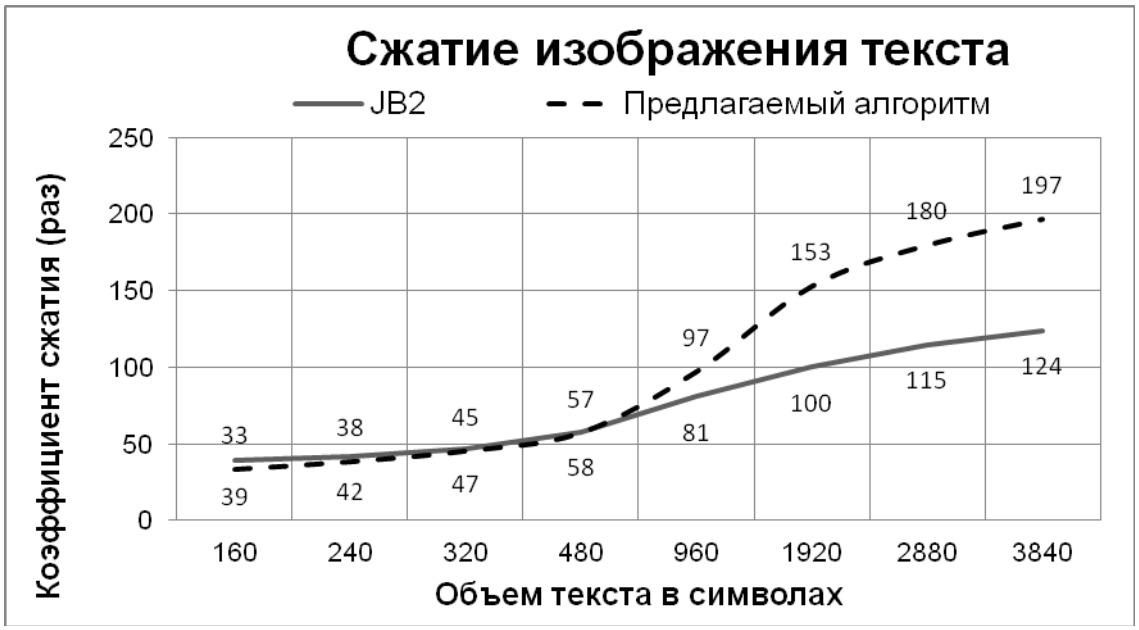
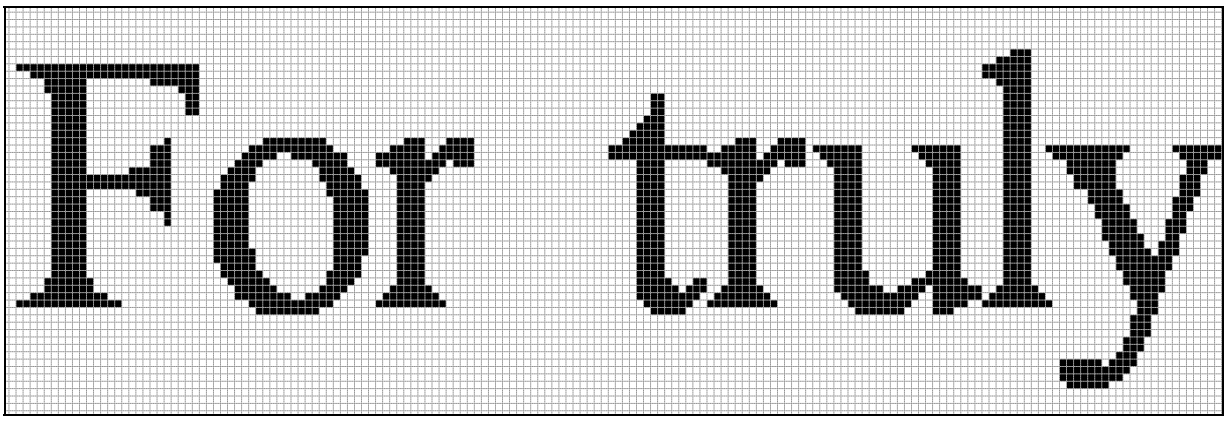


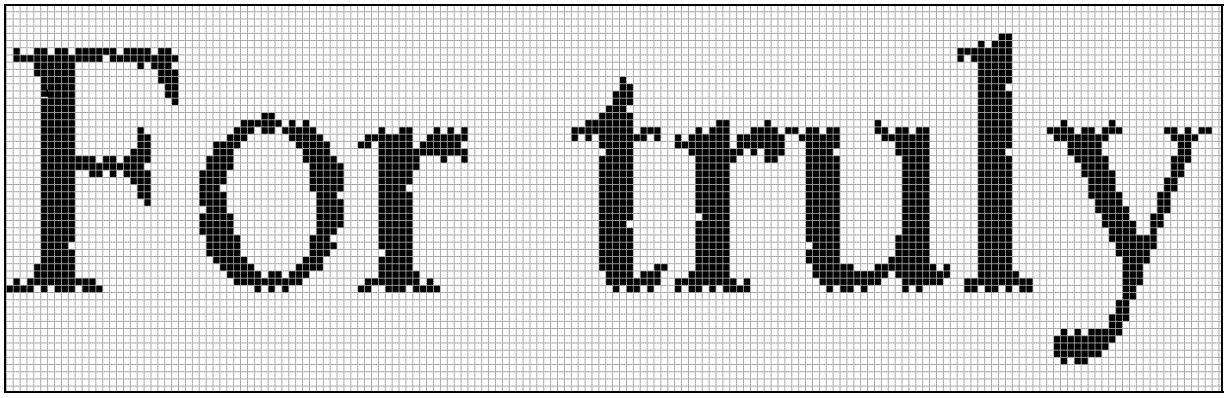
Рис. 5. Коэффициент сжатия изображения текста при разных объемах текста.

Заметное отличие между рассматриваемыми алгоритмами в степени сжатия как раз наблюдается начиная с объема текста в 10^3 символов, когда количество классов, даваемых этими алгоритмами, становится существенно разным.

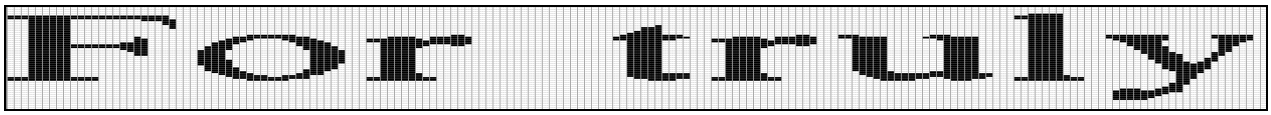
На рис. 6 представлен небольшой фрагмент текста в электронном виде. Объем текста – 3840 символов, стандартная страница А4.



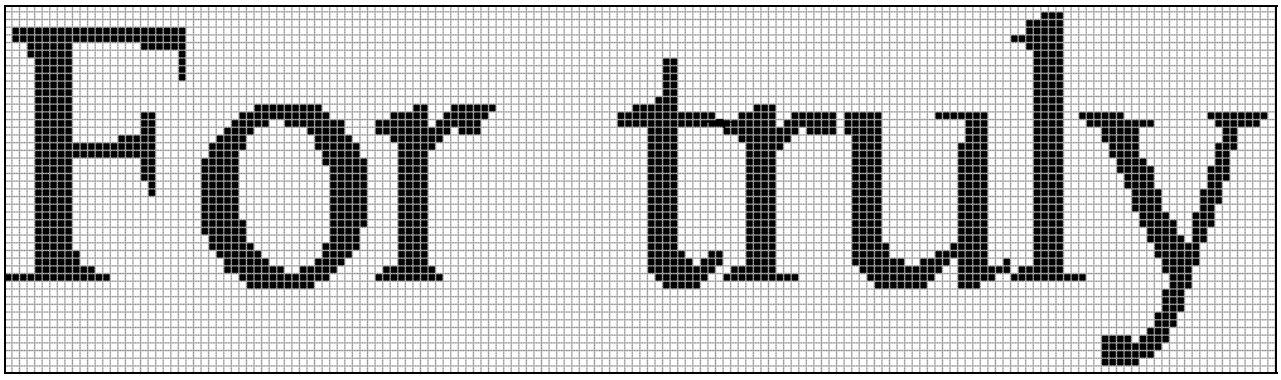
а



б



в



г

Рис. 6. Фрагмент текста: *а* – в электронном виде; *б* – изображение после печати и сканирования; *в* – восстановленное изображение рассматриваемым алгоритмом; *г* – алгоритмом JB2.

Отметим, что сканированное изображение текста заметно искажено контурными шумами. После сжатия этого изображения предлагаемым алгоритмом и последующего восстановления шумы исчезают, и хотя изображения символов несколько отличаются от их электронного варианта,

читаемость восстановленного текста такая же, как и у текста в электронном виде.

Отсутствие шумов в восстановленном изображении текста объясняется тем, что каждый элемент словаря, формируемого предлагаемым алгоритмом, получается усреднением всех изображений одного класса, что подавляет случайные шумы. Эффективность усреднения как фильтра случайных шумов тем выше, чем больше элементов в каждом классе изображений. Благодаря тому, что обсуждаемый алгоритм имеет число классов, близкое к минимально возможному, количество элементов в каждом классе близко к числу, с каким символ встречается в тексте. В рассматриваемом примере (табл. 1) каждый класс в среднем содержит примерно 56 изображений одного и того же символа, классы, образованные алгоритмом JB2, соответственно, – 16. Этим объясняется видимое наличие остаточных случайных шумов на изображения текста, восстановленного алгоритмом JB2 (два изображения буквы «r» неодинаковы, вертикальные линии в изображении символа «u» заметно отличаются друг от друга, засечки на изображении литеры «F» несимметричны и т. д.).

Заключение. Рассматриваемый в этой работе алгоритм сжатия сканированного изображения текста обладает, по сравнению с лучшим на сегодняшний день алгоритмом JB2, двумя преимуществами:

1. имеет ощутимо более высокий коэффициент сжатия: для текста объемом в $4*10^3$ символов (стандартная страница А4) превышение на 37%;

2. восстановленное изображение текста по качеству заметно лучше сканированного изображения и приближается к изначально электронному.

Эти преимущества достигнуты благодаря классификации символов, основанной на статистическом анализе совокупности всех вертикальных элементов строк (нечеткой классификации). Такой подход позволяет получить количество классов, близкое к числу отличных друг от друга символов, встречающихся в тексте. Это обеспечивает и высокую степень

сжатия, и хорошее качество восстановленного текста, заметно превышающее качество исходного сканированного текста.

Для наиболее часто используемого на практике разрешения изображения текста 300 dpi авторами были получены следующие сравнительные количественные показатели сжатия:

– в работе [3] преимущество над JB2 – 8 %;

– в работе [4] преимущество над JB2 – 25 %;

– в работе [2] преимущество над JB2 – 37 %.

Это является основной характеристикой представленного метода и раскрывает новые возможности повышения информативности представления текстовых графических данных в инженерных реализациях.

Список использованной литературы

1. Автоматический анализ сложных изображений : сб. переводов / под ред. Э. М. Бравермана. – Москва : Мир, 1989. – 310 с.

2. Иванов В. Г. Сжатие изображения текста на основе статистического анализа и классификации вертикальных элементов строки / В. Г. Иванов, Ю. В. Ломоносов, М. Г. Любарский // Восточно-Европейский журнал передовых технологий = Східно-Європейський журнал передових технологій = Eastern-European journal of enterprise technologies. – 2014. – № 4/2. – С. 4–15.

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

4. 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.

5. Прикладная статистика: Классификация и снижение размерности / С. А. Айвазян, В. М. Бухштабер, И. С. Енюков [и др.]. – Москва : Финансы и статистика, 1989. – 607с.

Аннотация. *Предложен новый метод сжатия битонального изображения текста, при котором в качестве основных элементов обработки рассматриваются не связные части символов, входящих в изображения текста, а вертикальные элементы строк этого изображения. Предлагаемый алгоритм позволяет получить степень сжатия на 30 – 40% выше, чем алгоритм JB2 (формат DjVu) для наиболее часто используемых разрешений сканирования.*

Ключевые слова: *сжатие изображения текста, вертикальные элементы строки, статистический анализ, нечеткая классификация.*

Summary. *A new method of compression bitonal image text in which as basic processing elements are considered not connected parts of characters included in the text image, and the vertical lines of the elements of the image. The proposed algorithm provides a compression ratio of 30 – 40 % higher than the algorithm JB2 (format DjVu) for the most frequently used scanning resolution.*

Keywords: *text image compression, vertical line items, statistical analysis, fuzzy classification.*