



О ПРЕДВАРИТЕЛЬНОЙ ОБРАБОТКЕ ИЗОБРАЖЕНИЙ В ЗАДАЧЕ ИХ СЖАТИЯ

В. А. ГОЛОЩАПОВА

*Белгородский
государственный
национальный
исследовательский
университет*

*e-mail:
VGoloschapova
@bsu.edu.ru*

В работе описаны предварительные преобразования изображения, которые позволяют увеличить коэффициент его сжатия, при условии дальнейшего применения алгоритмов сжатия и представлены результаты вычислительных экспериментов

Ключевые слова: изображение, предварительные преобразования, сжатия без потерь, коэффициент сжатия

Развитие современных информационных систем и сетей привело к широкому использованию цифровых изображений. В настоящее время многие отрасли техники, имеющие отношение к получению, обработке, хранению и передаче информации, в значительной степени ориентируются на развитие систем, в которых информация представлена в виде изображений. В связи с этим растет интерес к улучшению алгоритмов сжатия изображений. Существующие алгоритмы сжатия с потерями не всегда приемлемы, для некоторых изображений они могут привести к потере мелких деталей и сделать изображения вообще бесполезными (например, для медицинских или космических изображений), поэтому в таких случаях применяются алгоритмы сжатия изображений без потерь, такие как GIF, PNG, Lossless JPEG, Хаффман. Можно легко заметить, что изображение обладает избыточностью в двух измерениях, т.е. как правило, соседние точки, как по горизонтали, так и по вертикали, в изображении близки по значению. В данной работе приведен способ предварительной обработки изображений, использующий корреляционные связи, как вертикальные (по столбцу), так и горизонтальные (по строке) изображения, который при дальнейшем применении алгоритмов сжатия, позволит достичь более высоких коэффициентов сжатия для изображений по сравнению с известными методами.

В данной работе в качестве исходных анализируемых изображений были взяты изображения размером $m \times n$ в градациях серого с 256 уровнями серого. Предлагаемые предварительные преобразования состоят в следующем.

На первом этапе значения яркости пикселей второй и последующих строк заменяются их разностями

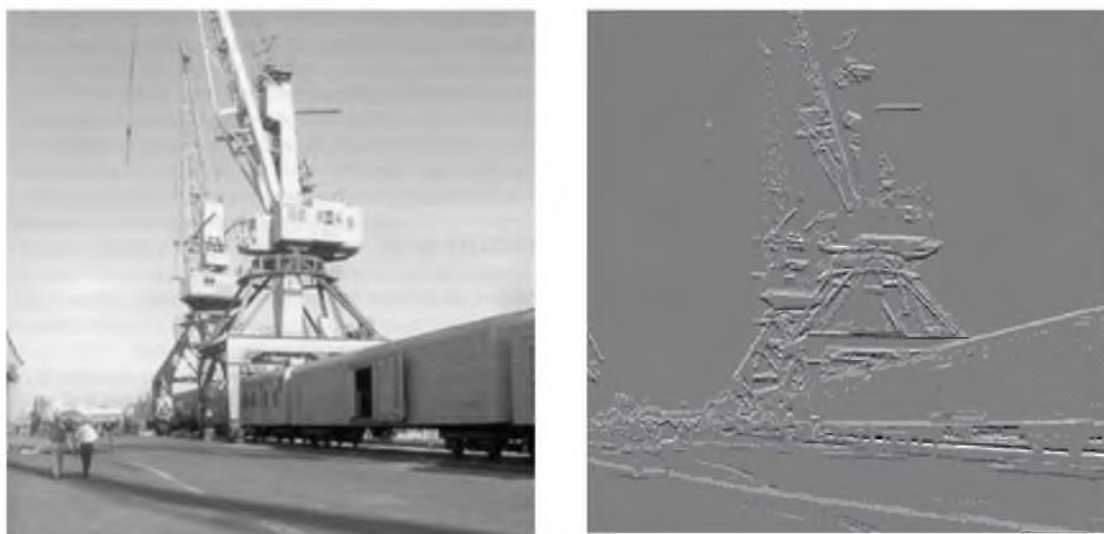
$$f(i, j) = f(i - 1, j) - f(i, j), \text{ где } i=2, \dots, m, j=1, \dots, n. \quad (1)$$

Далее значения яркости пикселей первой строки, за исключением первого элемента заменяются разностями между ними

$$f(i, j) = f(i, j - 1) - f(i, j), \text{ где } i=1, j=n, \dots, 2. \quad (2)$$

На последнем этапе преобразований сохраняется $a = f(1, 1)$, значение которого не учитывается в процессе сжатия.

В результате указанных преобразований изображение приобретает следующий вид, приведенный на рисунке 1а.



а

б

Рис. 1. Изображение 1: а – исходное изображение;
б – изображение, полученное после предварительной обработки

На рисунке 1б видно, что практически все изображение 1 имеет приблизительно одинаковую яркость пикселей, за исключением тех областей, где наблюдались резкие перепады яркости, что позволило выделить контуры на рисунке. Таким образом, используя несложные математические преобразования (1, 2), было сокращено количество различных значений яркости пикселей, что при дальнейшем применении алгоритмов сжатия позволит увеличить коэффициент сжатия.

Рассмотрим обратные преобразования.

На первом этапе первому элементу изображения присваиваем ранее сохраненное значение

$$\tilde{f}(1,1) = a. \quad (3)$$

Далее восстанавливаем остальные значения элементов первой строки

$$\tilde{f}(i, j) = \tilde{f}(i, j-1) - \tilde{f}(i, j), \text{ где } i=1, j=2, \dots, n. \quad (4)$$

Затем восстанавливаем элементы оставшихся строк изображения

$$\tilde{f}(i, j) = \tilde{f}(i-1, j) - \tilde{f}(i, j), \text{ где } i=2, \dots, m, j=1, \dots, n. \quad (5)$$

Результат восстановления полностью совпадает с исходным изображением.

На основе использования предложенной предварительной обработки (1, 2) и дальнейшего применения метода сжатия Хаффмана (алгоритм сжатия без потерь), коэффициент сжатия изображения 1 с предварительной обработкой увеличивается в 4 по отношению к коэффициенту сжатия изображения 1 методом Хаффмана без предварительной обработки.

Для множества изображений размером $m \times n$ в градациях серого с разными яркостными характеристиками были проведены вычислительные эксперименты по оцениванию коэффициента сжатия предложенного алгоритма и существующих алгоритмов сжатия без потерь, таких как GIF, PNG, Lossless JPEG, Хаффман [1, 2].

GIF использует формат сжатия LZW. Таким образом, хорошо сжимаются изображения, строки которых имеют повторяющиеся участки. В особенности изображения, в которых много пикселей одного цвета по горизонтали. Основным недостатком метода LZW (и, естественно, форматом GIF) является то, что на основе данного метода можно сжимать только полутоновые и индексированные изображения с палитрой, в которых значения пикселей могут принимать не более 256 значений. Поэтому, прежде чем сохранять полноцветное изображение в формате GIF, необходимо перевести его в индексированное изображение. При этом часто происходит существенная потеря качества.



PNG использует сжатие без потерь по алгоритму Deflate. Формат обладает более высокой степенью сжатия для файлов с большим количеством цветов, чем GIF, но разница составляет около 5-25 %, что недостаточно для абсолютного преобладания формата, так как небольшие 2-16-цветные файлы формат GIF сжимает с не меньшей эффективностью.

Lossless JPEG представляет собой дополнение к JPEG. В отличие от «обычного» JPEG, построенного на основе дискретного косинусного преобразования, Lossless JPEG для энтропийного кодирования ошибки предсказания Lossless JPEG использует код Хаффмана. В качестве альтернативного стандарт допускает использование арифметического кодирования.

Метод кодирования Хаффмана. Метод кодирования Хаффмана относится к группе методов сжатия данных без потерь информации. Этот метод используется для поддержки факсимильной связи и представления документов. Применяется также при записи графических изображений в файлы и является компонентом алгоритма сжатия данных JPEG. Особенностью метода является использование кодов переменной длины, при этом наиболее вероятным символам присваиваются наиболее короткие кодовые слова, а менее вероятным – длинные. Благодаря такой стратегии, код Хаффмана дает минимальную среднюю длину кодовой последовательности, приближающуюся к энтропии источника сообщения.

Коэффициент сжатия $\delta_{сж}$ изображений рассчитывался по формуле (6)

$$\delta_{сж} = \frac{V_{исх}}{V_c}, \tag{6}$$

где $V_{исх}$ – первоначальный объем (в битах) изображения, V_c – объем сжатого изображения.

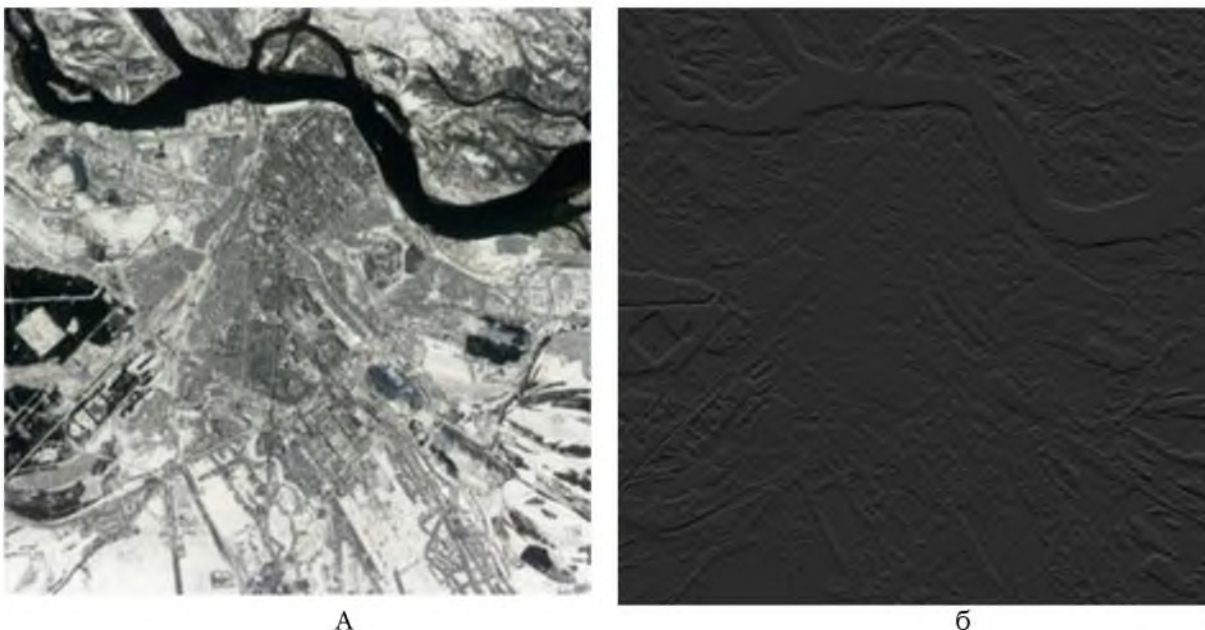
В табл. приведены коэффициенты сжатия изображений известными алгоритмами сжатия без потерь (GIF, PNG, Lossless JPEG, Хаффман) и предложенным алгоритмом.

Таблица

Коэффициент сжатия $\delta_{сж}$ изображений

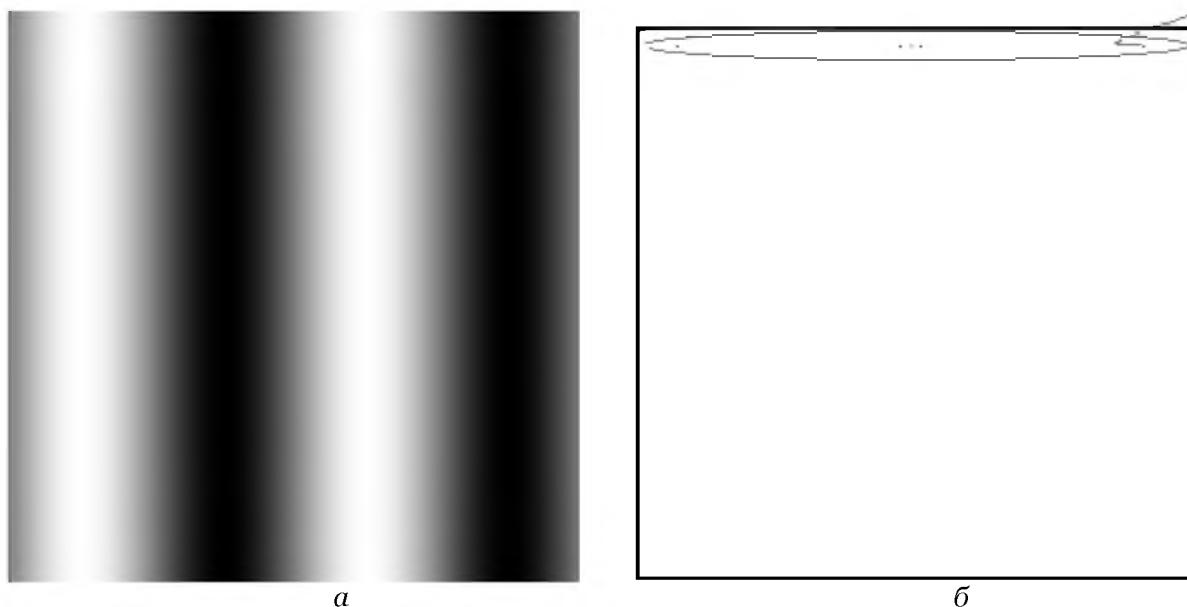
Из.	mхn	$V_{исх}$	GIF		PNG		Lossless JPEG		Хаффман		Преобразованное и сжатое Хаффманом	
			V_c	$\delta_{сж}$	V_c	$\delta_{сж}$	V_c	$\delta_{сж}$	V_c	$\delta_{сж}$	V_c	$\delta_{сж}$
И1	256x256	67	32	2,1	64	1,05	44	1,5	64	1,05	17	3,9
И2	256x256	197	64	3,1	152	1,3	86	2,3	197	1	43	4,6
И3	250x243	62	32	1,9	52	1,2	32	1,9	61	1,01	22	2,8
И4	450x337	154	76	2	124	1,24	76	2	152	1,01	48	3,2
И5	800x731	1754	352	4,98	548	3,2	276	6,35	1440	1,21	255	6,87
И6	900x506	1366	152	8,98	272	5,02	156	8,75	1290	1,05	138	9,89
И7	510x415	636	88	7,22	156	4,07	68	9,35	530	1,2	61	10,42
И8	624x212	397	68	5,83	148	2,68	64	6,2	341	1,16	43	9,23
И9	624x395	740	52	14,23	96	7,7	52	14,23	396	1,86	47	15,74
И10	256x256	196	8	24,5	4	49	16	12,25	149	1,31	0,76	256

Результаты приведенных вычислительных экспериментов, представленные в таблице 1 показали, что коэффициент сжатия изображений для предложенного алгоритма сжатия существенно превышает коэффициент сжатия изображений для известных алгоритмов сжатия без потерь качества. При этом предложенный алгоритм лучше всего сжимает изображения отличающиеся высокой избыточностью информации, в смысле наличия на изображении множества различных мелких деталей, по сравнению с известными алгоритмами сжатия без потерь (рис. 2).



А Б
Рис. 2. Изображение 2: а – исходное изображение;
б – изображение, полученное после предварительной обработки

Применив указанные преобразования (1, 2) для модельных изображений, например, синусоида (рис. 3) можно достичь значение коэффициента сжатия изображений в 100 и более раз без потерь. При этом в зависимости от размера изображения, значение коэффициента сжатия может достигнуть 1000 и более.



а б
Рис. 3. Изображение 10: а – исходное изображение;
б – изображение, полученное после предварительной обработки

Из рисунка 3б можно увидеть, что за исключением первой строки, пиксели всех строк изображения 10 после предварительной обработки имеют одинаковые значения, в данном случае 255. Следовательно, применив любой из методов сжатия изображений без потерь, можно сократить объем изображения в 256 раз (исходное изображение на рис. 3 имеет размерность 256x256 пикселей). В данном случае ал-



горитм предварительной обработки изображения применялся для преобразования строк, так как в изображении 10 (рис. 3) наибольшая корреляция наблюдается по строкам.

Результаты приведенного вычислительного эксперимента, представленные в таблице 1, показали, что использование предложенного алгоритма предварительных преобразований для сжатия изображений без потерь, позволяет получить коэффициент сжатия, превосходящий коэффициент сжатия для этих изображений, сжатых другими известными методами.

Работа выполнена в рамках дополнительного внутривузовского конкурса грантов «Инициатива», проект № ВКГИ 036-2013.

Литература

1. Гонсалес, Р. Цифровая обработка изображений в среде Matlab [Текст] / Р. Гонсалес, Р. Вудс, С. Эддинс. – М.: Техносфера, 2006. – 616 с.
2. Яне, Б. Цифровая обработка изображений [Текст] / Б. Яне. – М.: Техносфера, 2007. – 584 с.

ABOUT THE IMAGE PRE-PROCESSING IN THE PROBLEM OF THEIR COMPRESSION

V. A. GOLOSCHAPOVA

*Belgorod
National Research University*

*e-mail:
VGoloschapova@bsu.edu.ru*

This paper describes preliminary conversions of image which increase its compression ratio, while using compression algorithms. Results of computational experiments are given

Key words: image, preliminary conversion, lossless, compression ratio.