Вестник Ивановского государственного университета

Серия «Биология, Химия, Физика, Математика»

Вып. 3 / 2002 С. 140 – 143

С. И. Хашин

ИНТЕРПОЛЯЦИОННЫЙ МЕТОД СЖАТИЯ ГРАФИКИ

Предлагается метод сжатия графических файлов (с потерями), основанный на интерполяции. По силе сжатия он не уступает, а часто превосходит JPEG, но значительно быстрее работает. Эту особенность возможно применить при сжатии видеоинформации.

It is offered the method of compression of graphics files (with losses), based on interpolation. On power of compression it does not concede and frequently exceeds JPEG, but much faster works. This feature can be applied for compression of videoinformation.

УДК 513.6.

1. Введение

Целью предлагаемого метода является разработка методов сжатия графической и, в перспективе, видеоинформации.

Самый известный и, по всей видимости, самый эффективный на сегодняшний день метод "сжатия с потерями"— стандарт JPEG. Он был разработан еще в 1991 году. Главное его достоинство — большой коэффициент сжатия. Применение формата JPEG с максимальным качеством уменьшает объем файла до 10-15% от оригинала, а стандартное качество (т. е. достаточное практически всегда) доводит размер до 5-7% от оригинала.

Стандарт является одним из наиболее распространенных, поэтому логично было бы сравнивать все новые (см., напр., [1, 2]) разработки именно с ним.

2. Предлагаемый новый метод сжатия графики

Основная идея предлагаемого метода сжатия графической информации (с потерями) — интерполяция. Рассмотрим его поэтапно.

2.1. Дискретизация

Пусть $f[0\dots M_y-1, 0\dots M_x-1]$ — табличная функция от двух переменных, заданная на прямоугольнике размером $M_y \times M_x$ (массив). Разобьем исходный прямоугольник на меньшие прямоугольники размером $dy \times dx$. Во всем дальнейшем изложении некоторые (непринципиальные) тонкости, связанные с тем, что M_y не кратно dy или M_x не кратно

Работа выполнена при финансовой поддержке РФФИ (проект 02-01-00079).

dx, сознательно опущены. Параметры M_y, M_x, dy, dx предполагаются зафиксированными. Количество прямоугольников равно $ry \cdot rx$, где $ry = [(M_y + dy - 1)/dy]$, $rx = [(M_x + dx - 1)/rx]$.

Через $g = D(f)$ обозначим массив $g[0 \dots ry - 1, 0 \dots rx - 1]$ размером $ry \times rx$, где $g[y, x]$ равно среднему значению функции f на (y, x)-прямоугольнике. Массив $D(f)$ назовем **дискретизацией** массива f с уровнем (dy, dx). Если требуется явно указать параметры dy, dx, то будем писать $D(f, dy, dx)$.

2.2. Простая интерполяция

Пусть даны значения функции в вершинах прямоугольника

$$\begin{array}{cc} f(y_0, x_0), & f(y_0, x_0 + dx), \\ f(y_0 + dy, x_0), & f(y_0 + dy, x_0 + dx). \end{array}$$

Билинейная 4-точечная интерполяционная формула для функции f приближает значение функции f в произвольной точке $(y_0 + a_y, x_0 + a_x)$ через ее значения в 4 соседних.

Пусть дан массив $g[0 \dots ry-1, 0 \dots rx-1]$ размером $ry \times rx$. Попробуем построить, интерполируя эти данные, массив $f[0 \dots M_y-1, 0 \dots M_x-1]$ размером $M_y \times M_x$. Используя интерполяционную формулу, вычислим значения $f[y, x]$ для всех пар (y, x). Эту операцию назовем **интерполяцией** и обозначим

$$f = I(g, dy, dx).$$

2.3. Усиленная интерполяция

Разумеется, операции I, D не являются взаимно обратными. Нельзя надеяться получить равенство

$$I(D(f)) = f,$$

так как массив f задается $M_y \cdot M_x$ числами, а массив $D(f)$ — в $dy \cdot dx$ раз меньшим количеством. Но и обратное равенство

$$D(I(g)) = g$$

тоже не выполняется. Если проинтерполировать некоторый массив g, а потом взять его дискретизацию, то полученный массив g' будет некоторым усреднением от исходного. Однако оказывается возможным так подправить операцию интерполирования (I'), что $D(I'(g)) = g$ для произвольного массива $g[0 \dots ry - 1, 0 \dots rx - 1]$. Операцию I' назовем **усиленной интерполяцией**.

2.4. Предварительный этап интерполяционного сжатия

Пусть $f[0 \dots M_y-1, 0 \dots M_x-1]$ — массив размером $M_y \times M_x$. Через f_8 обозначим массив $D(f, 8, 8)$. Его размер — $1/64$ размера массива f. Из свойств операции I' следует, что

$$D(f - I'(f_8, 8, 8), 8, 8) = 0,$$

т. е. среднее значение массива $h_8 = f - I'(f_8, 8, 8)$ на каждом 8×8-квадрате равно 0. Имеем:

$$f = h_8 + I'(f_8, 8, 8),$$

т. е. массив f восстанавливается через массивы f_8 и h_8.

Через f_4 обозначим массив $D(h_8), 4, 4)$. Его размер — 1/16 размера массива f. Из свойств операции I' следует, что

$$D(h_8 - I'(f_4, 4, 4), 4, 4) = 0,$$

т. е. среднее значение массива $h_8 - I'(f_4, 4, 4)$ на каждом 4×4-квадрате равно 0. Далее поступаем аналогично. Выпишем для удобства все формулы в одном месте:

$$\begin{array}{llll} f_8 & = D(f, 8, 8), & h_8 = f - I'(f_8, 8, 8), \\ f_4 & = D(h_8, 4, 4), & h_4 = h_8 - I'(f_4, 4, 4), \\ f_2 & = D(h_4, 2, 2), & \\ f_1 & = h_4 - I'(f_2, 2, 2). & \end{array}$$

Отсюда получаем:

$$f = f_1 + I'(f_2, 2, 2) + I'(f_4, 4, 4) + I'(f_8, 8, 8),$$

т. е. исходный массив f однозначно восстанавливается по округленным данным (разного уровня) f_8, f_4, f_2, f_1.

Таким образом, на первом этапе из массива f получаются массивы f_8, f_4, f_2 и f_1. Исходный массив f по ним однозначно восстанавливается.

2.5. Основной этап: квантование

Квантованием числа x с уровнем l назовем число $\mathrm{round}(x/l)$. Если число x лежало в пределах $0 \ldots 256$, для его хранения требовалось 8 бит. После квантования с уровнем 32 оно будет лежать в пределах $0 \ldots 8$, и для его хранения будет достаточно 3 бит.

Вместо того, чтобы полностью хранить массивы f_8, f_4, f_2 и f_1, будем запоминать только результаты их квантования с уровнями l_8, l_4, l_2 и l_1 соответственно. Числа l_8, l_4, l_2, l_1 являются параметрами алгоритма и определяют уровень сжатия. Естественно предположить, что для высокочастотных компонент следует использовать больший уровень квантования. Типичным мог бы служить выбор: $l_8 = 2$, $l_4 = 8$, $l_2 = 32$, $l_1 = 128$.

2.6. Завершение: энтропийное сжатие

Завершающий этап должен выглядеть примерно так же, как и для всех остальных методов: обычное сжатие (уже без потерь) информации, полученной на основном этапе.

2.7. Достоинства и недостатки

Было проведено большое количество экспериментов, реализующих предложенный метод сжатия. Помимо рассмотренной выше билинейной интерполяции (по 4 точкам), рассматривалась аналогичная 6-точечная интерполяция. К сожалению, в отличие от сжатия без потерь, оценка качества сжатия с потерями весьма субъективна. К тому же эффекты потери качества в интерполяционном методе совершенно отличны от аналогичных эффектов в JPEG-методе. Поэтому бессмысленно говорить, например, что один метод на 5 – 10% лучше другого.

Тем не менее общие выводы можно сформулировать следующим образом.

1. Предлагаемый метод не хуже, а чаще всего на 15 – 20% лучше jpeg-сжатия.

2. Предлагаемый метод требует для своей реализации в несколько раз меньше вычислений, чем jpeg-сжатие.

Достигнутое небольшое улучшение сжатия не позволяет надеяться на замену общепринятого стандарта JPEG. Преимущества нового метода могут проявиться при сжатии видеоинформации.

Библиографический список

1. *Ватолин Д. С.* Алгоритмы сжатия изображений: Метод. пособие. М.: МГУ, ВМиК, 1999. 78 с.
2. *Ватолин Д., Ратушняк А., Смирнов М., Юкин В.* Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео. М.: Диалог — МИФИ, 2002. 384 с.