УДК 004.932

ОПТИМИЗАЦИЯ КОДИРОВАНИЯ ЦИФРОВЫХ ИЗОБРАЖЕНИЙ ПО МЕТОДУ JPEG

Умняшкин С.В., Космач М.В.

(Московский государственный институт электронной техники)

Аннотация. Рассматривается способ повышения характеристик кодирования алгоритма JPEG, позволяющий понизить уровень вносимой ошибки. Оптимизация не затрагивает спецификации (формата) сжатых данных и не требует изменения алгоритма декодирования.

Алгоритм сжатия цифровых изображений JPEG [1] на сегодняшний день является фактическим стандартом, поддерживаемым многочисленными программными и аппаратными системами цифровой обработки, хранения и передачи изображений. Ограничимся случаем обработки черно-белых полутоновых изображений и рассмотрим кратко основную схему кодирования (sequential encoding) спецификации JPEG, которая наиболее широко применяется на практике.

В самом общем виде оптимизацию произвольного алгоритма кодирования данных с потерями можно свести к нахождению минимума функции

$$J(\mathbf{u}) = D(\mathbf{u}) + \lambda R(\mathbf{u}), \qquad (1)$$

где D – мера вносимой ошибки, R – длина битового кода, параметр $\lambda \geq 0$ определяет необходимое соотношение между качеством обработки и длиной кода, а вектор **u** представляет собой набор параметров, которые позволяет варьировать конкретный алгоритм [2]. В качестве меры вносимых искажений обычно используется среднеквадратичная ошибка (PSNR) [2]. Предлагаемый нами метод оптимизации кодера JPEG может быть представлен как процесс минимизации функции (1), где в качестве варьируемых параметров **u** используются:

- набор из 100 рекомендуемых JPEG матриц квантования, $\{\mathbf{Q}_Q\}_{Q=1}^{100}$ (чем больше индекс матрицы, тем меньше ошибка квантования),
- все возможные таблицы кодов Хаффмена $\{\mathbf{h}\}$,
- все возможные способы *дополнительного усечения* коэффициентов спектров, $L = \bigcup L_i$ (i – индекс изображения-фрагмента 8×8).

Таким образом, $\mathbf{u} = \{\mathbf{Q}, \mathbf{h}, L\}$. Под термином «дополнительное усечение» будем понимать возможность уменьшения абсолютного значения целочисленной величины проквантованных коэффициентов $\tilde{y}_{k,l} = \mathrm{Round}(y_{k,l}/q_{k,l})$. Частный случай подобной процедуры, а именно, принудительное *обнуление* $T = \bigcup T_i$ (т.е. усечение до нуля, для каждого спектра ДКП производится отдельно), предложен в работе [2]. Поскольку JPEG специальным образом кодирует группы нулевых коэффициентов, можно ожидать (и это подтверждают эксперименты), что возможность увеличения числа нулевых коэффициентов спектра может повысить характеристики кодирования. Мы предлагаем обобщить эту идею следующим образом: помимо возможности обнуления коэффициентов анализировать также возможность их дополнительного «огрубления» в сторону уменьшения абсолютного значения, *не обязательно до нуля*. Действительно, кодируя ненулевые коэффициенты ДКП, кодер JPEG должен записать код Хаффмана для ключевого байта и необходимое количество бит, определяемое разрядностью проквантованного значения $\tilde{y}$ этого коэффициента [1]: 1 бит – для $\tilde{y} \in \{-1,1\}$, 2 бита для $\tilde{y} \in \{-3,-2\} \cup \{2,3\}$, 3 бита – для $\tilde{y} \in \{-7,-6,-5,-4\} \cup \{4,5,6,7\}$ и т.д. Чем больше разрядность числа $\tilde{y}$, тем реже он встречается в спектрах ДКП и тем больше также длина соответствующего кода Хаффмана для ключевого байта. Например, если коэффициент $\tilde{y} = -2$, то вполне возможно, что более эффективным с точки зрения минимизации (1) может оказаться его кодирование как -1.

Предлагаемый алгоритм оптимизации кодера JPEG, т.е. алгоритм минимизации функции (1) при заданном λ, можно разбить на три следующих шага.

1. Проинициализировать таблицу кодов Хаффмана по таблице, рекомендуемой группой JPEG, $\mathbf{h} \leftarrow \mathbf{h}_{\mathbf{JPEG}}$.

2. Минимизировать по $\mathbf{Q}$ функцию

$$J(\mathbf{Q},\mathbf{h}) = \min_{L}\left[J(\mathbf{Q},L,\mathbf{h}) = D(\mathbf{Q},L) + \lambda R(\mathbf{Q},L,\mathbf{h})\right], \qquad (2)$$

т.е. определить $\mathbf{Q}^* = \arg\min_{\mathbf{Q}\in\{\mathbf{Q}_1,\ldots,\mathbf{Q}_{100}\}} J(\mathbf{Q},\mathbf{h})$. При этом находим также L^*.

3. Выполнить минимизацию функции (2) (или (1)) по $\mathbf{h}$, т.е. определить $\mathbf{h}^* = \arg\min_{\mathbf{h}}\left[J(\mathbf{Q}^*,L^*,\mathbf{h}) = D(\mathbf{Q}^*,L^*) + \lambda R(\mathbf{Q}^*,L^*,\mathbf{h})\right]$. Считаем оптимальные параметры $(\mathbf{Q}^*,L^*,\mathbf{h}^*)$ найденными.

Так же, как и в работе [2], процедура поиска оптимальных кодов Хаффмана $\mathbf{h}^*$ не является итерационной (шаг 3), что практически не влияет на характеристики оптимизации, однако позволяет использовать быстрые алгоритмы поиска на шаге 2. Принципиальные отличия нашей схемы от предложенной в работе [2], как уже упомянуто выше, заключены в процедуре поиска оптимального способа усечения L^*:

$$L^* = \arg\min_L \left[J(\mathbf{Q}, L, \mathbf{h}) = D(\mathbf{Q}, L) + \lambda R(\mathbf{Q}, L, \mathbf{h}) \right], \qquad (3)$$

что необходимо для вычисления значений функции (2). Процедуру усечения коэффициентов спектров представим как результат двух воздействий: обнуления T и уменьшения C абсолютных величин (до ненулевого значения), $L=T\times C$. Если способ усечения L_i не изменяет коэффициентов i-ого спектра, будем обозначать это как $L_i = \mathbf{1}$ (при этом, очевидно, $T_i = C_i = \mathbf{1}$). Поиск $L^*=T^*\times C^*$ (3) производится в два этапа.

1. Минимизировать функцию (1) по T, считая $C = \mathbf{1}$, т.е. определить $T^* = \arg\min_T \left[J(\mathbf{Q}, T\times\mathbf{1}, \mathbf{h}) = D(\mathbf{Q}, T\times\mathbf{1}) + \lambda R(\mathbf{Q}, T\times\mathbf{1}, \mathbf{h}) \right]$.
2. Минимизировать функцию (1) по C, т.е. определить $C^* = \arg\min_C \left[J(\mathbf{Q}, T^*\times C, \mathbf{h}) = D(\mathbf{Q}, T^*\times C) + \lambda R(\mathbf{Q}, T^*\times C, \mathbf{h}) \right]$.

Поиск оптимального усечения проводится для каждого фрагмента по отдельности, т.е. $L^* = \cup L_i^*$ ($T^* = \cup T_i^*$, $C^* = \cup C_i^*$). Особенности выполнения этапа 1, способы построения быстрых процедур поиска T^* - подробно рассмотрены в работе [2]. Для оставшихся после первого этапа ненулевых коэффициентов

$\tilde{y} \neq 0$ далее на втором этапе необходимо проверять возможность их перевода в каждый из диапазонов меньшей разрядности, при этом ясно, что возможность полного обнуления коэффициентов исключена предшествующим этапом оптимизации по T (поэтому рассматривать значения $\tilde{y} = \pm 1$ не имеет смысла). Однако перенос проквантованного коэффициента дальше, чем в ближайший диапазон – это уже «почти обнуление» по величине вносимой ошибки, и в силу того, что возможность обнуления была проанализирована ранее, можно ограничиться анализом только одного варианта дополнительного усечения – на общую эффективность оптимизации такое упрощение не оказывает заметного влияния, однако сокращает вычислительные затраты. В итоге C-оптимизация (этап 2) выступает в роли надстройки к схеме [2]. Дополнительный выигрыш в PSNR невелик и на тестовых изображениях *Barbara* и *Lena* составляет величину около 0.1 дБ, однако соответствующее усложнение оптимизации не влечет за собой практически никаких вычислительных затрат.

ЛИТЕРАТУРА

1. *Wallace G.K.* The JPEG algorithm for image compression standard // Communications of the ACM. - 1991. -Vol. 34. -№4. - P. 30-44.

2. *M.Crouse and K.Ramchandrn.* Joint thresholding and quantizer selection for transform image-coding: entropy-constrained analysis and applications to baseline JPEG // IEEE Trans. on Image Processing. – 1997. - Vol. 6. -№2 - P. 285-297.