

УДК 004.932.2 + 004.622 + 004.627

## АДАПТИВНЫЙ ВАРИАНТ ФРАКТАЛЬНОГО АЛГОРИТМА СЖАТИЯ И ЕГО ИСПОЛЬЗОВАНИЕ К ОБРАБОТКЕ МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ

*Анастасова Е.А., Беловодский В.Н.**Донецкий национальный технический университет*

*Рассматриваются существующие алгоритмы и системы маскирования изображений. Сделаны выводы по критериям оценки погрешностей работы алгоритмов, обоснован выбор модификаций для дальнейшего применения к оцифрованным медицинским изображениям. Предлагается адаптивный вариант фрактального алгоритма сжатия и его использование к обработке медицинских изображений.*

### Введение

Медицинские изображения относятся к классу информативных. В тоже время такие изображения зачастую имеют большие размеры, что может вызывать сложности при работе с ними. При обработке изображений такого рода следует аккуратно варьировать соотношение степени сжатия и потерь качества изображения. Целью работы является разработка адаптивного варианта фрактального алгоритма для обработки медицинских изображений. Алгоритма, который обеспечивал бы, при незначительных потерях информации, небольшой размер файла на выходе, достаточный для дальнейшей его оценки медицинским работником.

### 1 Выделение значимой области

Для реализации этапа выделения значимой области было создано приложение на языке MatLab. Область интереса задается двумя противоположными точками предполагаемой области прямоугольной формы (верхняя левая и правая нижняя). После этого выбранная область выделяется на исходном изображении линиями синего цвета (по контуру). При нажатии на кнопку “PolyMask” происходит отсечение той части изображения, которая лежит вне выбранной области. В результате этого, получаем новое изображение, которое и подлежит последующему сжатию (рис. 1)



Рисунок 1. Этап выделения значимой области

### 2 Разбиение изображения на домены и ранги

Для данного этапа был выбран алгоритм quadro-дерева. Выбор был основан на анализе литературы по данной теме. Опишем основные этапы предложенного алгоритма.

В изображении  $f$  выделяют множество доменных блоков. Для этого используют такие параметры как минимальный  $K_{Dmin}$  и максимальный  $K_{Dmax}$  показатели уровня разбиения доменов. Размер доменов  $S_D$  на первом уровне разбиения  $K_D$  определяется формулой (1):

$$S_D = \frac{S}{2^{K_D}}, \quad (1)$$

где  $S$  – размер изображения. Домены могут перекрываться, а плотность перекрытия задается коэффициентом перекрытия доменов  $V_D$ . Изображение разбивают на ранговые блоки  $\{R_k\}$ , которые перекрываются. Они могут не быть одинакового размера, потому что используется адаптивное разбиение методом квадродерева со сменным размером блоков. Более глубокое разбиение каждого отдельного ранга на четыре ранга одинакового размера следующего уровня происходит по мере необходимости в процессе работы алгоритма.

Этот этап является одинаковым для всех модификаций алгоритма, принимающих участие в анализе, так как особенность их работы проявляется на этапе поиска пары домен-ранг.

### 3 Подбор пары домен – ранг

На данном этапе предусматривается возможность применения 4 модификаций. Работа каждой из них предполагает, что разбиение исходного изображения на ранги и домены уже произведено.

#### 3.1 FE – алгоритм

Особенностью работы данного алгоритма является вычисление вектора характеристик для каждого домена: стандартное отклонение, асимметрия, межпиксельная контрастность, максимальный градиент, коэффициент бета, который определяется по формуле (ниже), он характеризует отличие значений пикселей от значения центрального пикселя.

Происходит перебор всех рангов: формируется вектор характеристик ранга, вычисляется разность между вектором характеристик текущего ранга и векторами каждого домена (2):

$$d = \sum_{j=1}^5 |f_j^R - f_j^D|, \quad (2)$$

где  $f_j^R$  и  $f_j^D$  – это  $j$ -ые характеристики ранга и домена соответственно.

Задается  $q$  (процент доменов, обеспечивающих минимальное значение расстояний между векторами характеристик). Формируется новый набор доменов путем отбора  $\%q$  (например,  $q=2\%$ ) доменов обеспечивающих минимальные значения расстояний между векторами характеристик. Алгоритм может работать в одном из двух режимах, выбранном пользователем, - с поиском и без поиска наилучшего домена [1].

#### 3.2 Корреляция Пирсона

Особенностью работы алгоритма является вычисление среднеквадратичного отклонения яркостей пикселей всех рангов и всех доменов ( $\{\sigma D\}, \{\sigma R\}$ ). Если выполняется условие  $\sigma D > \sigma R$ , где  $\sigma D$  – среднеквадратичное отклонение яркостей пикселей доменов,  $\sigma R$  – среднеквадратичное отклонение яркостей пикселей рангов, то формируется новый набор доменов. Для текущего ранга вычисляется  $r' = \rho(R, D)$ . Поиск самого близкого к 1 значения по модулю коэффициента корреляции  $r'$ . Затем происходит сжатие домена до размера ранга. Вычисление коэффициентов контрастности и яркости пикселей происходит по формулам (3,4) [2]:

$$s = r' \frac{\sigma R}{\sigma D}, \quad (3)$$

$$o = \bar{R} - s\bar{D}, \quad (4)$$

#### 3.3 Показатель энтропии

Данный алгоритм несколько отличается от работы других модификаций. Формируется наборы доменов для каждого уровня квадродерева (depth). Для начального уровня  $depth=0$ . Задается начальное значение порога отбора доменов  $\lambda_{depth}$ . В анализируемой работе  $\lambda_{depth}=0,7$ . Вычисляются энтропии рангов и доменов ( $H(p_0, p_1, \dots, p_k) = -\sum_{i=0}^k p_i \cdot \log_2 p_i$ , где  $k$  – максимальное количество уровней серого в данном изображении). Для текущего ранга перебираются все домены. Проверяется

условие:  $|\text{Entropy}(R_i) - \text{Entropy}(D_j)| \leq \lambda \text{depth}$ . Если выполняется условие выше, то происходит вычисление коэффициентов (5, 6) [3].

$$s = n \frac{\left[ \sum_{i=1}^n d_i r_i - \sum_{i=1}^n d_i \sum_{i=1}^n r_i \right]}{\left[ n \sum_{i=1}^n d_i^2 - \left( \sum_{i=1}^n d_i \right)^2 \right]}, \quad (5)$$

$$o = \frac{1}{n} \left[ \sum_{i=1}^n r_i - s \sum_{i=1}^n d_i \right]. \quad (6)$$

### 3.4 Алгоритм, использующий нелинейное отображение

Работа алгоритма, использующего нелинейное отображение, аналогична работе базового алгоритма за исключением вычисления коэффициентов.

Для анализа динамики поведения показателей качества в этом случае используется квадратичное отображение вида (7):

$$r_{xy} = w(d_{xy}) = a \cdot d_{xy}^2 + b \cdot d_{xy} + c, \quad (7)$$

где  $r_{xy}$  – пиксель ранга,  $d_{xy}$  – пиксель домена, сжатого до размеров ранга.

Коэффициенты  $a$ ,  $b$ ,  $c$  находятся методом наименьших квадратов [4], так чтобы выполнялось требование (8):

$$F(a, b, c) = \sum_{i, j=1}^n (r_{xy} - (a \cdot d_{xy}^2 + b \cdot d_{xy} + c))^2 \rightarrow \min, \quad (8)$$

где  $n$  – размер ранга в пикселях.

Функция  $F$  принимает свои минимальные значения в точках экстремума. Для их нахождения необходимо приравнять частные производные функции к нулю. В развернутом виде получаем систему линейных уравнений. Решая ее по правилу Крамера [5], находят определители, а значение коэффициентов находят по формуле (9):

$$a = \frac{\Delta a}{\Delta}; b = \frac{\Delta b}{\Delta}; c = \frac{\Delta c}{\Delta}. \quad (9)$$

Вместо набора используется непосредственно основной критерий, но применять его к уменьшенным копиям сравниваемых пар домен-ранг [1].

## 4 Оценка погрешности

Для оценки погрешности используются следующие характеристики:

- средняя пиксельная ошибка (10):

$$e = 100 \cdot \frac{\sum_{x=1}^{I_W} \sum_{y=1}^{I_H} |p_{xy} - p'_{xy}|}{256 \cdot I_W \cdot I_H}, \quad (10)$$

где  $p_{xy}$ ,  $p'_{xy}$  – значения пикселя в точке  $(x, y)$  выходного и декодированного изображения соответственно,  $I_W$ ,  $I_H$  – ширина и высота (в пикселях) изображения соответственно.

- Коэффициент сжатия  $C$  – это отношение размера выходной информации об изображении к размеру исходной.
- Время кодирования  $T_{код}$  – это промежуток времени от начала исполнения первого шага алгоритма до окончания последнего.
- Время декодирования  $T_{дек}$  – это промежуток времени декодирования изображения с принятым количеством итераций декодирования.

Характеристики рассмотренных алгоритмов представлены в таблице 1.

Таблица 1. Сравнительная характеристика алгоритмов

Название	Время кодирования	Средняя пиксельная ошибка	Коэффициент сжатия
FE-алгоритм	ускорение в 8-10 раз	уменьшение на 9%	уменьшается на 12-37%
Использование коэффициента корреляции Пирсона	ускорение в 9,35 раза	повышение качества	увеличение
Энтропия	существенное ускорение	практически совпадает с оригинальным при декодировании	практически такой же
Нелинейное отображение	уменьшение до 58%	малы	может увеличиваться не больше чем на 11%

### Выводы

Предложен один из возможных вариантов адаптивного алгоритма сжатия изображений, включающий в себя этап выделения значимых областей, и предусматривающий использование нескольких методов подбора пары домен-ранг. Алгоритм реализован программно, на конкретном образце медицинского изображения выполнены вычислительные эксперименты, проведен сравнительный анализ результатов.

### Литература

- [1] Bublichenko, A.V. Algorithms for image compression: a comparative analysis and modification / A.V. Bublichenko, V.N. Belovodsky / Qualification Masters work. - 2008.
- [2] Илюшин С.В. Фрактальное сжатие телемедицинских изображений [Текст] / С.В. Илюшин, С.Д. Свет //, – Режим доступа : [www/URL: http://www.elsv.ru/files/actual/130.pdf](http://www.elsv.ru/files/actual/130.pdf) – Загл. с экрана.
- [3] Venkata Rama Prasad VADDELLA, Ramesh Babu INAMPUDI, Journal of Applied Computer Science & Mathematics, no. 9 (4) /2010, Suceava. Режим доступа: [http://jacs.usv.ro/getpdf.php?paperid=9\\_3](http://jacs.usv.ro/getpdf.php?paperid=9_3)
- [4] Р.М. Кроновер Фракталы и хаос в динамических системах. Основы теории. – М.: Постмаркет, 2000
- [5] В.Е. Гмурман. Теория вероятностей и математическая статистика: Учеб. пособие для ВУЗов. – 10-е изд. стер. – М.: Высшая школа, 2004. – 479с.