

УДК 621.397

ФРАКТАЛЬНАЯ ОБРАБОТКА ИЗОБРАЖЕНИЙ

Перегуда Е.С.

Введение.

Постановка задачи

Задача фрактальной обработки изображений представляет большой интерес как с научной, так и с практической точек зрения. Теория фрактального преобразования и построения аттрактора хорошо исследована. Данная теория обширна. Она затрагивает множество областей математики, физики, химии, биологии.

Важной перспективой приложения фрактального преобразования к задаче сжатия с потерями является возможность расширения теории фрактального множества на другие типы медиа данных, допускающих сжатие с потерями. К таким типам данных можно отнести аудио и видео информацию. Расширение фрактального множества на данные типы информации позволяет говорить о возможности создания универсального алгоритма сжатия данных с потерями. Создание такого универсального алгоритма представляет большой практический интерес.

Алгоритмическое решение обратной задачи фрактального преобразования можно представить в следующем виде: кодируемое множество разбивается на два класса – рангов R и доменов D , и каждому рангу ищется соответствующий домен, который после применения нужного преобразования в точности соответствует рангу. Сложность расчета заключается в переборе всех комбинаций рангов с доменами и применения всех выбранных преобразований. В качестве преобразований часто выбирают самые простые изометрические аффинные преобразования. Большинство исследователей выбирают в качестве базового алгоритм Фишера [1]. Выражение расчета метрики отклонения домена и ранга $\|E(D, R)\|$ имеет следующий вид:

$$\|E(D, R)\| = \sqrt{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (\varphi \circ (s \cdot d_{ij} + o) - r_{ij})^2}, \quad (1)$$

где d_{ij} - значение точки домена D , r_{ij} - значение точки ранга R , φ - оператор преобразования, связанный с перестановкой точек в пространстве (чаще всего это аффинные изометрические преобразования), s и o - масштабирование и смещение яркости точки. Параметры s и o рассчитываются, исходя из условия экстремума метрики отклонения. Выражение фрактальной аппроксимации $s \cdot d_{ij} + o$ является линейной функцией, поэтому выражение метрики отклонения (1) обладает только одной точкой экстремума-минимума. Исходя из этого, выражение (1) дифференцируют относительно переменных s и o . В итоге, исходя из условия экстремума-минимума, получают следующие выражения оптимума для переменных s и o :

Решается задача сокращения вычислительной сложности фрактального анализа. Разработаны три алгоритма сокращения вычислительной сложности и приведено их обоснование. Представлены результаты исследований, показывающие эффективность предложенных алгоритмов.

$$s = \frac{n \cdot m \left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (\varphi \circ d_{ij}) \cdot r_{ij} \right) - \left(\sum_{i=0}^{n-1} d_{ij} \right) \left(\sum_{i=0}^{m-1} r_{ij} \right)}{n \cdot m \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} d_{ij}^2 - \left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} d_{ij} \right)^2}; \quad (2)$$

$$o = \frac{1}{n \cdot m} \left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} r_{ij} - s \cdot \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} d_{ij} \right). \quad (3)$$

Данные выражения позволяют выполнить фрактальный анализ с наименьшим отклонением аттрактора от исходного изображения. Данный факт определяет высокое качество аппроксимации фракталом любого произвольного изображения, что обуславливает выбор алгоритма Фишера в качестве базового при разработке ускоренного фрактального анализа.

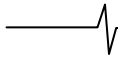
Сложность алгоритма заключается в необходимости выполнения множества операций умножения, деления, сложения, вычитания и извлечения корня. Наиболее затратными по времени выполнения являются операции умножения и деления, в то время как операцию извлечения квадратного корня можно исключить. Операции деления в выражениях (2) и (3) производятся только на переменные

$$n \cdot m \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} d_{ij}^2 - \left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} d_{ij} \right)^2 \text{ и } n \cdot m,$$

которые носят постоянный характер и зависят только от текущего анализируемого домена. Такая постановка вопроса позволяет произвести предварительные расчеты данных значений и преобразовать значение данной переменной из знаменателя в числитель. Таким образом, появляется возможность исключить из расчета операции деления. Следующими по сложности являются операции умножения. Среди всех операций умножения в выражениях (1), (2), (3) следует выделить три наиболее часто повторяемые:

1. Операция умножения при расчете свертки текущего домена и ранга

$$\left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} (\varphi \circ d_{ij}) r_{ij} \right)$$



на каждый пиксель в выражении (2). Важно отметить, что для каждого нового преобразования $\varphi \circ$ необходимо заново пересчитывать свертку домена и ранга.

2. Две операции умножения на каждый пиксель из выражения (1).

Данные операции невозможно заменить на более простые операции сложения и вычитания. Исключение операций умножения позволило бы значительно сократить вычислительную сложность фрактальных алгоритмов.

Полный перебор всех комбинаций можно представить в виде следующего графа состояний (рис. 1). В этом графе узлы представляют из себя множество рангов $\{R_j\}$, посещение которых возможно, если найдены наиболее подходящие для них домены из множества $\{D_i\}$, и метрика отклонения $\|E(D_i, R_j)\|$ минимальна, при этом количество доменов и рангов равны соответственно N и M . Таким образом, ребро графа характеризует домен, а его вес характеризует метрику отклонения домена от ранга, на который указывает ребро. Очевидно, что качество решения обратной задачи фрактального преобразования будет тем выше, чем короче путь прохождения через все узлы графа. Важно отметить, что каждому домену можно применить несколько преобразований φ . На рис. 1 это выражается в восьми дополнительных ребрах $\varphi = \{1, 2, 3, 4, 5, 6, 7, 8\}$ для каждого домена. Число восемь определяется восемью аффинными изометрическими преобразованиями, применяемыми к прямоугольному сегменту изображения.

Важно отметить сходство решения данной задачи с известной задачей о коммивояжере. Но в отличие от нее граф задачи на рис. 1 располагается в линейном одномерном пространстве, а это значит, что отсутствуют промежуточные узлы. Так же в задаче о коммивояжере решалась задача нахождения наикротчайшего пути по заданному графу узлов и ребер, в то время как в рассмотренном графе длина ребер априори неизвестна, и необходимо для каждого ранга рассчитать вес ребра отклонения домена для всех выбранных преобразований. Именно необходимость многократного пересчета комбинаций доменов и рангов и выбора ребра наименьшего веса и определяет вычислительную сложность решения обратной задачи фрактального преобразования. С увеличением изображения увеличивается количество рангов, доменов, и количество требуемых математических операций увеличивается в прогрессии степени натурального числа.

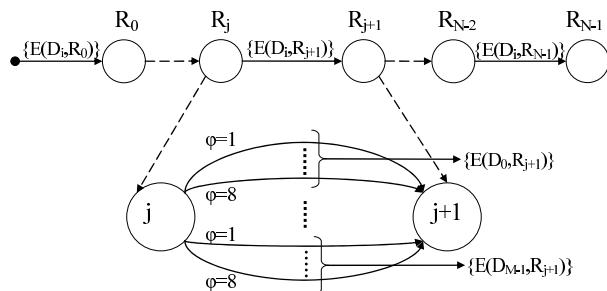


Рис. 1. Граф состояний алгоритма решения обратной задачи фрактального преобразования.

Автором было предложено три алгоритма сокращения вычислительной сложности алгоритма решения об-

ратной задачи фрактального преобразования [5-13]. В совокупности сокращение времени работы алгоритма составляет 30 – 40 раз по сравнению с алгоритмом без включения данных алгоритмов.

В качестве отправной точки в поиске методов сокращения сложности рассматривают каждый ранговый и доменный блок как вектор \vec{R} и \vec{D} соответственно, в линейном векторном пространстве \mathcal{R}^n где $n = N \times N$ - количество пикселей в квадратном ранговом блоке со сторонами N . Преобразование из квадратного блока со стороной N в вектор длиной $n = N^2$ может быть проведено путем построчного сканирования блока.

Алгоритмы сокращения вычислительной сложности

Первый алгоритм. Первоначально при подстановке выражения (3) в выражение (1) получим следующее выражение:

$$\|E(D, R)\| = \sqrt{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \left(\varphi \circ \left(s \cdot d_{ij} + \frac{1}{n \cdot m} \left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} r_{ij} - s \cdot \sum_{i=0}^{n-1} \sum_{j=0}^{m-1} d_{ij} \right) \right) - r_{ij} \right)^2} \quad (4)$$

Обозначим среднюю яркость домена и ранга следующим образом:

$$\vec{R} = \frac{1}{n \cdot m} \left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} r_{ij} \right), \quad (5)$$

$$\vec{D} = \frac{1}{n \cdot m} \left(\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} d_{ij} \right). \quad (6)$$

Средняя яркость независима от пространственного преобразования. Результатом подстановки соотношений (5) и (6) в (4) является:

$$\|E(D, R)\| = \sqrt{\sum_{i=0}^{n-1} \sum_{j=0}^{m-1} \left(\varphi \circ (s \cdot (d_{ij} - \vec{D})) - (r_{ij} - \vec{R}) \right)^2}. \quad (7)$$

Представление блоков рангов и доменов в векторной форме позволяет ввести обозначения соответствующих векторных операций в выражение расчета метрики отклонения между доменом и рангом. Если для упрощения расчетов операцию пространственного преобразования φ исключить, то

$$\|E(D, R)\| = \|s \cdot \vec{D} - \vec{R}\|,$$

Где $\vec{D} = \{d_{ij} - \vec{D}\}$ - вектор доменного блока с нулевой средней яркостью, $\vec{R} = \{r_{ij} - \vec{R}\}$ - вектор рангового блока с нулевой средней яркостью.

Как видно при векторном представлении преобразований происходит переход к векторным операциям. В таком линейном пространстве операции сложения и вычитания векторов, операции умножения и сложения скаляров с векторами линейны и расчет метрики имеет следующий вид:

$$\|E(D, R)\| = \|s \cdot \vec{D} - \vec{R}\| = \|s \cdot \vec{D}^{n-1} - \vec{R}^{n-1}\|, \quad (8)$$

где \vec{D}^{n-1} и \vec{R}^{n-1} - векторы доменного и рангового блоков в ортогональном пространстве S^n с Евклидовой метрикой без постоянной составляющей.

Ортогональное дополнение пространств домена \vec{D}^{n-1} и ранга \vec{R}^{n-1} в пространстве S^n также является векторным пространством, состоящим из одного вектора постоянной составляющей.

Выражение (8) описывает простое условие отклонения с одним неизвестным. Частным случаем является условие полного подобия:

$$\|E(\vec{D}^{n-1}, \vec{R}^{n-1})\| = 0, \quad (9)$$

классическое решение которого имеет вид:

$$s = \vec{R}^{n-1} \cdot (\vec{D}^{n-1})^+ \quad (10)$$

Решение с псевдообратной матрицей с вычислительной точки зрения представляет большую сложность. Но учитывая тот факт, что эта матрица имеет смысл наименьшего квадратичного приближения, возможно, следует выбрать более простую операцию приближения, с меньшей точностью, но более быструю.

Решение можно получить при рассмотрении среднеквадратичной метрики:

$$\sqrt{r_1^2 + r_2^2 + \dots + r_{n-1}^2} = \sqrt{(s \cdot d_1)^2 + (s \cdot d_2)^2 + \dots + (s \cdot d_{n-1})^2};$$

$$s = \frac{\sqrt{r_1^2 + r_2^2 + \dots + r_{n-1}^2}}{\sqrt{(d_1)^2 + (d_2)^2 + \dots + (d_{n-1})^2}} = \frac{\|\vec{R}^{n-1}\|}{\|\vec{D}^{n-1}\|} \quad (11)$$

Подставив выражение (11) в (8), получим:

$$\|E(\vec{D}^{n-1}, \vec{R}^{n-1})\| = \left\| \frac{\|\vec{R}^{n-1}\|}{\|\vec{D}^{n-1}\|} \cdot \vec{D}^{n-1} - \vec{R}^{n-1} \right\| = \left\| \frac{\vec{D}^{n-1}}{\|\vec{D}^{n-1}\|} - \frac{\vec{R}^{n-1}}{\|\vec{R}^{n-1}\|} \right\| = 0, \quad (12)$$

если принять во внимание что множитель метрики рангового блока $\|\vec{R}^{n-1}\|$ постоянен при переборе доменов \vec{D}^{n-1} из множества $\{D_i\}$. Дроби вида

$$\frac{\vec{D}^{n-1}}{\|\vec{D}^{n-1}\|} = \phi(\vec{D}^{n-1}) \text{ и } \frac{\vec{R}^{n-1}}{\|\vec{R}^{n-1}\|} = \phi(\vec{R}^{n-1})$$

являются выражениями нормализации векторов домена и ранга в нормированном метрическом пространстве.

Решение (12) соответствует условию полного подобия. Во всех других случаях решения носят характер интерполяции и стремятся к минимизации метрики отклонения домена и ранга

$$\|E(\vec{D}^{n-1}, \vec{R}^{n-1})\|.$$

Операция нормализации применяется к векторам доменных и ранговых блоков только один раз и в дальнейшем может использоваться только их нормированные значения. Таким образом, исключается одна операция умножения. Вторая операция умножения исключается путем замены операции свертки на операцию деления одного скаляра на другой. Третью операцию умножения исключают путем перехода от метрики в Евклидовом пространстве к метрике по абсолютному значению:

$$\|E(\vec{D}^{n-1}, \vec{R}^{n-1})\| = \left(\left| \frac{d_1}{\|\vec{D}^{n-1}\|} - \frac{r_1}{\|\vec{R}^{n-1}\|} \right| \right) + \dots + \left(\left| \frac{d_{n-1}}{\|\vec{D}^{n-1}\|} - \frac{r_{n-1}}{\|\vec{R}^{n-1}\|} \right| \right) \quad (13)$$

Результатом реализации является сокращение времени анализа в 4 раза. При этом исключаются наиболее

часто повторяющиеся операции умножения, и алгоритм требует только одну операцию сложения и вычитания.

Второй алгоритм. Далее приводится обоснование исключения цикла перебора восьми изометрических аффинных преобразований. Данное исключение основывается на том факте, что при выборе в качестве ортогонального дискретного косинусного преобразования ДКП (DCT) появляется возможность использовать свойство четности ядра данного преобразования. Данное свойство выражается в тождестве:

$$DCT(Y(x, y)) = DCT(Y(x, (N-1) - y)),$$

где $Y(x, y)$ - прямое изображение, а $Y(x, (N-1) - y)$ - его зеркальное отражение. Данное тождество имеет смысл только для четных коэффициентов ДКП. Таким образом, для изометрических аффинных преобразований одного изображения четные коэффициенты ДКП идентичны. Это значит, что в произведении над изображением операции пространственного преобразования ϕ , четные коэффициенты ДКП остаются неизменными. Это позволяет представить метрику отклонения домена и ранга $\|E(\vec{D}^{n-1}, \vec{R}^{n-1})\|$ в следующем виде:

$$\|E(\vec{D}^{n-1}, \vec{R}^{n-1})\| = \|E(\vec{D}^{n-1}, \vec{R}^{n-1})_{ЧЕТ}\| + \|E(\vec{D}^{n-1}, \vec{R}^{n-1})_{НЕЧЕТ}\| \quad (14)$$

Задавшись параметром качества метрики отклонения домена и ранга ε , получим:

$$\|E(\vec{D}^{n-1}, \vec{R}^{n-1})\| = \|E(\vec{D}^{n-1}, \vec{R}^{n-1})_{ЧЕТ}\| + \|E(\vec{D}^{n-1}, \vec{R}^{n-1})_{НЕЧЕТ}\| \leq \varepsilon \quad (15)$$

Если неравенство (15) тождественно, то домен аппроксимирует ранг с искажением не больше чем заданное ε . С учетом оператора преобразования ϕ получим:

$$\|E(\phi \circ \vec{D}^{n-1}, \vec{R}^{n-1})\| = \|E(\vec{D}^{n-1}, \vec{R}^{n-1})_{ЧЕТ}\| + \|E(\phi \circ \vec{D}^{n-1}, \vec{R}^{n-1})_{НЕЧЕТ}\| \leq \varepsilon$$

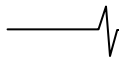
и условие расчета метрики отклонения $\|E(\phi \circ \vec{D}^{n-1}, \vec{R}^{n-1})\|$ можно представить в виде:

$$\|E(\phi \circ \vec{D}^{n-1}, \vec{R}^{n-1})\| = \begin{cases} \|E(\vec{D}^{n-1}, \vec{R}^{n-1})_{ЧЕТ}\| + \|E(\phi \circ \vec{D}^{n-1}, \vec{R}^{n-1})_{НЕЧЕТ}\|; \text{ если } \|E(\vec{D}^{n-1}, \vec{R}^{n-1})_{ЧЕТ}\| \leq \varepsilon \\ \text{Следующий элемент множества } \{D_i\}; \text{ если } \|E(\vec{D}^{n-1}, \vec{R}^{n-1})_{ЧЕТ}\| > \varepsilon \end{cases} \quad (16)$$

Это условие является условием ветвления алгоритма динамического программирования.

Результатом реализации является сокращение времени анализа от 1 до 32 раза, в зависимости от состава изображения. Данный элемент динамического программирования позволяет сравнивать домен с рангом на основе только четверти всех коэффициентов.

Третий алгоритм. Последним алгоритмом сокращения сложности является алгоритм выбора оптимального изометрического аффинного преобразования с использованием неравномерного распределение энергии по коэффициентам ДКП. Данное свойство применяется при зигзагообразном представлении коэффициентов ДКП, хорошо известном из спецификации JPEG. Исходя из этого, достаточно, чтобы знаковое распределение младших индексов рангового и доменного блоков



совпадало. Таким образом зачастую появляется возможность быстрого установления наиболее подходящего аффинного преобразования на основе сравнения знаков только младших индексов (например, трех) коэффициентов. Исходя из этого, действие разработанного алгоритма заключается в использовании прямого метода адресации – поиска номера аффинного преобразования в ячейке памяти, адресуемой трехразрядным двоичным кодом.

Результатом реализации является сокращение времени анализа в 8 раз.

Работу разработанных алгоритмов лучше всего представить в виде графа состояний. На рис. 2 дан граф состояний улучшенного алгоритма решения обратной задачи фрактального преобразования.

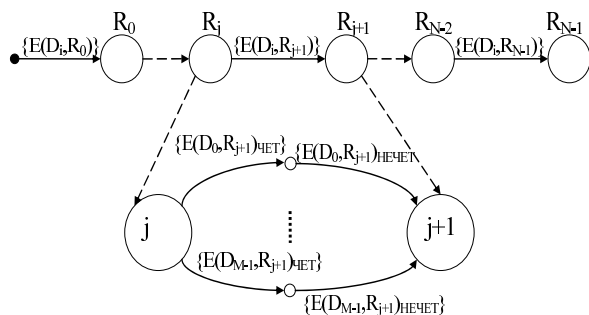


Рис. 2. Граф состояний улучшенного алгоритма решения обратной задачи фрактального преобразования.

При сравнении рисунков 1 и 2 видна значительная разница, между базовым алгоритмом и новым разработанным алгоритмом. Наличие в новом алгоритме промежуточной вершины между двумя соседними вершинами – рангами позволяет ввести элемент динамического программирования.

Динамическое программирование реализуется в виде разработанного второго алгоритма сокращения вычислительной сложности. Третий алгоритм позволяет выбрать оптимальные нечетные коэффициенты домена. Таким образом, между соседними вершинами-рангами каждому домену соответствует только одно ребро. Время перебора всех доменов сокращается примерно в восемь раз. Первый алгоритм исключает три операции умножения и две операции сложения, что соответствует сокращению времени работы в три – четыре раза. Суммарное сокращение времени работы составляет 30 – 40 раз.

Статистические исследования

Ниже приводятся статистические исследования улучшенного алгоритма решения обратной задачи фрактального преобразования. В качестве тестового было выбрано изображение «Lena».

Для оценки качества преобразования использована величина пикового отношения сигнал/шум (PSNR).

При реализации разработанных алгоритмов сокращения вычислительной сложности на базе алгоритма Фишера возникает проблема с применением ДКП. На практике данное преобразование не может быть применимо к изображениям произвольного размера, поэтому

первоначально следует отказаться от разбиения изображения по схеме квадродерева и установить фиксированный размер рангов. Для выбора оптимального размера рангов необходимо сравнить результаты анализа при нескольких фиксированных размерах.

Определим следующую серию размеров рангов: 16×16, 8×8, 4×4. Зависимость качества от рангов представлена на рис. 3. Количество доменов при этом является фиксированным и равно 62000.

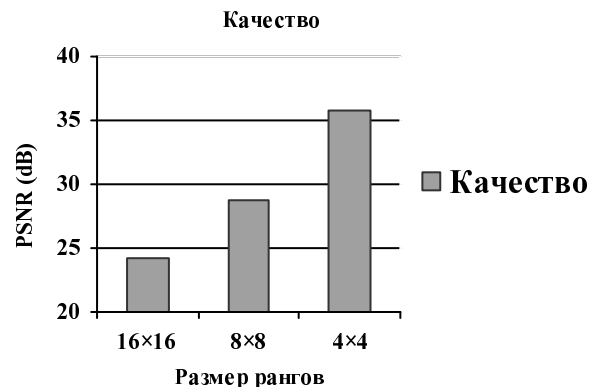


Рис. 3. График PSNR для размеров рангов 16×16, 8×8, 4×4

Как видно из рис. 3 оптимальный размер ранга составляет 4×4.

Далее приводятся результаты статистических исследований. Данные исследования представлены в виде графика на рис. 4. На данном графике представлено две кривые: время выполнения программы на базе алгоритма Фишера (1) и время выполнения программы на базе разработанных алгоритмов (2).

Как видно из графика на рис. 4 эффективность разработанных алгоритмов значительно превосходит алгоритм Фишера (30 – 40 раз). Причем эта эффективность растет с увеличением числа доменов.

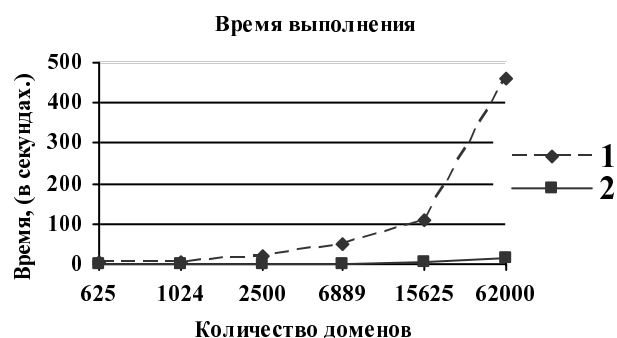


Рис. 4. График времени выполнения программы на базе алгоритма Фишера (1) и времени выполнения программы на базе разработанных алгоритмов (2)

Кроме исследований сокращения времени работы приведены исследования качества. Исследования представлены в виде графика качества (рис. 5). На данном графике представлено две кривые: качество работы программы на базе алгоритма Фишера (1) и качество работы программы на базе разработанных алгоритмов (2). Как видно из графика на рис. 6 каче-

ство работы разработанных алгоритмов хуже, чем качество работы алгоритма Фишера. Это связано с тем, что в разработанных алгоритмах обработка данных производится в нормированном пространстве. При этом динамический диапазон ограничивается, что и приводит к небольшой неточности. Как видно из графика, данная неточность невелика (не более 1 – 1,5 дБ), и визуально разница незаметна.

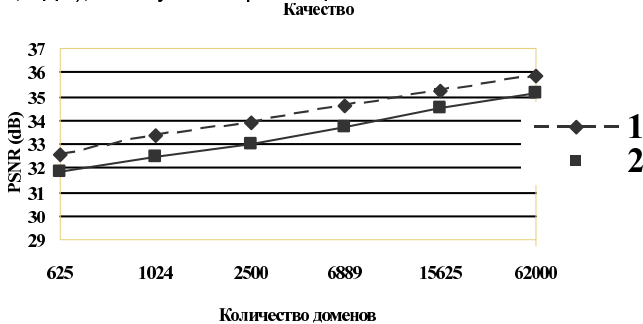


Рис. 5. График качества работы программы на базе алгоритма Фишера (1) и качества работы программы на базе разработанных алгоритмов (2).

Кроме исследований времени работы алгоритмов и качества анализа приведены исследования степени сжатия нового алгоритма фрактального анализа. Результаты сравнения качества сжатия от степени сжатия фрактального алгоритма и алгоритмов JPEG и JPEG2000 представлены в виде графиков на рис. 6.

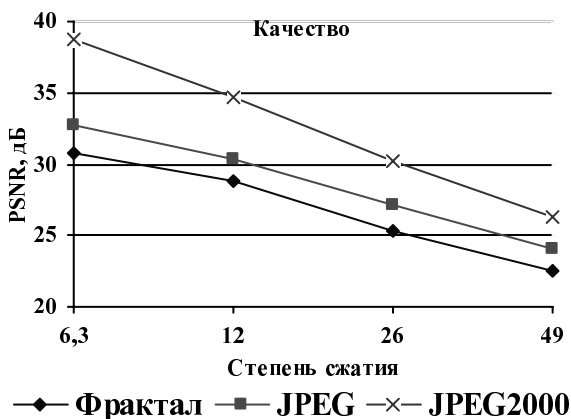


Рис. 6. Сжатие изображения «Lena»

Как видно из графиков, качество фрактального сжатия изображений уступает аналогам стандарта JPEG.

Заключение

В заключении важно отметить, что разработанные алгоритмы сокращения вычислительной сложности показали значительную эффективность, позволив уменьшить количество выполняемых математических операций и выбрать оптимальные операции преобразования, применяемые к изображениям. При этом, как очевидно,

возможно дополнительно применить уже существующие алгоритмы классификации, такие как нейронные сети, для достижения еще большего сокращения времени решения обратной задачи фрактального преобразования.

Литература

1. Уэлстид С. Фракталы в вейвлеты для сжатия изображений в действии. Учебное пособие. / С. Уэлстид – М.: Издательство Триумф, 2003 – 320 с.: ил.
2. Ватолин Д. С. Алгоритмы сжатия изображений / Д. С. Ватолин – М.: Диалог-МГУ, – 1999. – 273 с.
3. Шабаршин А. А. Метод фрактального сжатия изображений / А. А. Шабаршин // Научные школы УПИ-УГТУ. – 1997. – №1. – С. 70-82.
4. Zhang N., Yan B. Hybrid image compression method based on fractal geometry / N. Zhang, B. Yan // Electronics Letters 30 (1994), P. 406-408.
5. Перегуда Е.С. О применимости фрактальной обработки в технике передачи изображения// Телевидение: передача и обработка изображений. Материалы 4-й международной конференции, Санкт-Петербург, 24-26 мая 2005. с. 42 – 43.
6. Pereguda E.S. Acceleration of Algorithm of Fractal Image Compression// IEEE International Siberian Conference on Control and Communication (SIBCON-2005). Proceedings, Tomsk, Russia, October 21-22, 2005. p. 159 – 162.
7. Pereguda E.S. Acceleration of Algorithm of Fractal Compression of Images// Proceeding of The KOREA – RUSSIA Joint – Workshop 2006 on Signal Transmission, Processing, Sensor and Monitoring Systems, Khabarovsk, Russia, October 26 – 28, 2006, p. 46 – 48.
8. Сай С.В. и Перегуда Е.С. Методы сокращения объема вычислений в алгоритмах фрактального сжатия изображений// Вестник Тихоокеанского Государственного Университета. №1, июнь, 2006. с. 9 – 14.
9. Перегуда Е.С. Методы ускорения фрактальной обработки изображений// Теоретические и прикладные вопросы современных информационных технологий. Материалы Седьмой Всероссийской научно-технической конференции, Улан-Удэ, 24-30 июля 2006. Часть 1 с. 98 – 103.
10. Перегуда Е.С. Методы ускорения фрактального сжатия изображений// НАУКА. ТЕХНОЛОГИИ. ИННОВАЦИИ, Материалы всероссийской научной конференции молодых ученых, Новосибирск, 07-10 декабря 2006 г., Часть 1. с. 186 – 187.
11. Перегуда Е.С. Ускорение фрактального алгоритма в системах сжатия и передачи изображений// Телекоммуникации №6 2007, с. 2 – 7.
11. Перегуда Е.С. Архиватор цветного изображения на основе фрактального преобразования// Свидетельство об официальной регистрации программы для ЭВМ № 50200600375, ВНИИЦ, 2006.
12. Сай С.В., Перегуда Е.С. Алгоритм поиска мелких структур фото и видео изображений// Телевидение: передача и обработка изображений. Материалы 6-й международной конференции, Санкт-Петербург, 19-21 июня 2007. с. 81 – 82.