МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Самарский национальный исследовательский университет

имени академика С.П. Королева»

(Самарский университет)

Институт информатики, математики и электроники

Факультет информатики

Кафедра информационных систем и технологий

На правах рукописи

УДК 004.832.3

Сахибназарова Виктория Бахтиёровна

ИССЛЕДОВАНИЕ алгоритмов фрактального сжатия изображений

Автореферат

выпускной квалификационной работы магистра

по направлению подготовки 09.04.01 «Информатика и вычислительная техника»

профиль «Программное обеспечение мобильных устройств»

Самара – 2018 год

Работа выполнена в Самарском национальном исследовательском университете имени академика С. П. Королева.

Научный руководитель: к.т.н., доцент, Кудрина М.А.

Научные консультанты:

Рецензент:

Общая характеристика работы

**Актуальность темы**

Одной из главных характеристик любого канала связи является скорость передачи информации. Необходимо передавать как можно больше информации в сообщении наименьшего размера. В случае передачи графической информации для уменьшения объема передаваемых данных используются различные методы сжатия изображений. Все алгоритмы сжатия изображений обобщенно можно разделить на две группы: сжатие без потерь информации и сжатие с частичной потерей информации. Как правило, алгоритмы сжатия с потерей информации имеют более высокие коэффициенты сжатия по сравнению с алгоритмами сжатия без потерь. При этом качество изображения ухудшается не сильно, порой изменения даже незаметны для человеческого глаза. Кроме того, при сжатии изображений для передачи их по каналам связи повышается надежность передачи [1]. Надежность растет с уменьшением трафика, поскольку время передачи сокращается, и тем самым увеличивается объем передаваемой информации за единицу времени без увеличения ширины канала.

Одним из алгоритмов сжатия изображений с частичной потерей информации является фрактальное сжатие. Образно процесс фрактального сжатия можно описать как поиск самоподобных областей на изображении. Данный метод основан на использовании системы итерируемых функций Iterated Function System (IFS) [2]. Так как метод базируется на самоподобии элементов изображения, он заключается в моделировании рисунка несколькими меньшими фрагментами его самого. Специальные уравнения позволяют переносить, поворачивать и изменять масштаб участков изображения; таким образом, эти участки служат компоновочными блоками остальной части картины[3].

Сама система итерируемых функций представляет собой набор трехмерных аффинных преобразований, переводящих одно изображение в другое. Преобразованию подвергаются точки в трехмерном пространстве (х\_координата, у\_координата, яркость). С учетом использования IFS для осуществления фрактального сжатия (или фрактальной компрессии) исходное изображение делится на подобласти квадратной формы, называемые *ранговыми блоками*. Ранговые блоки пересекаться не могут. Также на исходном изображении выделяют *доменные блоки* (домены), являющиеся совокупностью 4-х соседних ранговых блоков. Домены могут пересекаться. Все ранговые блоки и домены – это квадраты со сторонами, параллельными сторонам исходного изображения. В ходе работы алгоритма компрессии для каждого рангового блока производится поиск домена, который после аффинных преобразований (сжатие доменного блока в 4 раза, поворот и/или отражение) и изменения яркости наименее всего отличается от рангового блока.

Достоинствами фрактальной компрессии являются степень сжатия на уровне JPEG при сравнительно одинаковом качестве, быстрый процесс декодирования и независимость восстанавливаемого изображения от разрешения (хранится структура изображения, а не данные о пикселях). Недостатками являются большие временные затраты сжатия и невозможность гарантировать ту или иную степень потерь (качество декодированного изображения зависит от самоподобия сжимаемого).

Таким образом, выбранная тема является актуальной, так как объем графической информации, которым мы оперируем ежедневно, чрезвычайно велик и поэтому изучение методов сжатия изображений имеет важное практическое значение. Также представляется интересным исследовать возможность ускорения выполнения фрактального сжатия изображений.

**Целью работы** является исследование вариантов реализации алгоритма фрактального сжатия изображений, а также изучение подходов, позволяющих ускорить выполнение данного алгоритма.

В соответствии с поставленной целью в выпускной квалификационной работе магистра решаются следующие **задачи**

**исследования:**

1. Изучение основного алгоритма фрактального сжатия и вариантов его реализации;
2. Изучение методов повышения скорости выполнения фрактального сжатия изображений;
3. Разработка и реализация программного и информационного обеспечения, позволяющего применять изученные алгоритмы и методы;
4. Исследование зависимости скорости сжатия исходного изображения и качества декодируемого изображения от использованного алгоритма.

**Методы исследования**, используемые в выпускной квалификационной работе магистра, основаны на положениях теории вероятностей и математической статистики.

**Научная новизна** работы заключается в исследовании времени выполнения фрактального сжатия изображения и качества декодируемого изображения, в зависимости от выбранного варианта реализации, используемого метода ускорения и параметров сжатия.

**Практическая ценность** работы заключается в разработке программного обеспечения автоматизированной системы, позволяющего решать следующие задачи:

1. Выбор изображения для проведения исследования.
2. Выбор варианта реализации алгоритма, метода ускорения и параметров сжатия.
3. Получение данных о времени сжатия, времени декомпрессии, предоставление изображения, полученного в результате декодирования и оценки его качества.

**Положения, выносимые на защиту:**

1. Исследование вариантов реализации алгоритма фрактального сжатия и методов его ускорения.
2. Автоматизированная система фрактального сжатия изображения, позволяющая выбрать сжимаемого изображение, вариант реализации алгоритма и метод ускорения и предоставляющая информацию о времени декомпрессии и качестве декодированного изображения.

**Апробация работы**

Основные положения и результаты работы представлялись, докладывались и обсуждались на международной конференции «Перспективные информационные технологии - 2017» (Самара, 2017), всероссийской заочной научно-технической конференции «Инновационные технологии в проектировании и производстве» (Пенза, 2017).

**Публикации**

Работы по теме выпускной работы опубликованы в 2 сборниках трудов конференции, индексируемых в РИНЦ, и 1 статья опубликована в сборнике материалов студенческой международной научно-практической конференции.

сОдержание работы

**Во введении** приведены основные определения и понятия, показана актуальность темы выпускной квалификационной работы.

**В первой главе** приведена постановка задачи, математическое обоснование фрактального сжатия изображений и приведены формулы, используемые алгоритмом для преобразования блоков изображений, приведены классический алгоритм сжатия и алгоритм декомпрессии.

Фрактальное сжатие осуществляется за счет использования так называемых *сжимающих* отображений, для которых справедливо следующее утверждение:

Если к какому-то изображению *F0* мы начнём многократно применять отображение *W* таким образом, что

то в пределе, при *i*, стремящемся к бесконечности, мы получим одно и то же изображение вне зависимости от того, какое изображение мы взяли в качестве *F0*:

Это конечное изображение *F* называют *аттрактором*, или *неподвижной точкой отображения W*.

Как будет описано далее, процесс декодирования изображения будет осуществляться именно путем многократно применения отображения *W* к базовому изображению. Именно наличие аттрактора у отображения *W* позволяет при декомпрессии в качестве базового использовать любой изображение.

Отображение *W* является объединением отображений *wi*, применяемых к отдельным блокам изображения. Такими отображениями являются аффинные и яркостные преобразования.

Аффинное преобразование − отображение плоскости или пространства в себя, при котором параллельные прямые переходят в параллельные прямые, пересекающиеся − в пересекающиеся, скрещивающиеся − в скрещивающиеся [4].

В общем виде аффинные преобразования на плоскости описываются следующими формулами:

где A, B, C, D, E, F – некие константы, (x,y) - координаты точки на плоскости до преобразования, (X,Y) – координаты точки на плоскости после преобразования.

Аффинным преобразованием, применяемым мной в данной работе является поворот на градусов. Такое преобразование выражается формулами:

Каждый из приведенных далее алгоритмов фрактального сжатия включает применение 8 аффинных преобразований к доменными блоками.

Примеры этих преобразований можно увидеть в таблице 1.

Таблица 1 – Аффинные преобразования доменного блока

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| № | Название | Формулы | Пример | № | Название | Формулы | Пример |
| 1 | Поворот на 0о |  |  | 5 | Отражение относительно оси Y |  |  |
| 2 | Поворот на 90о |  |  | 6 | Отражение относительно оси Х |  |  |
| 3 | Поворот на 180о |  |  | 7 | Поворот на 90о и отражение относительно оси Y |  |  |
| 4 | Поворот на 270о |  |  | 8 | Поворот на 90о и отражение относительно оси Х |  |  |

При поиске подходящего доменного блока для достижения максимального соответствия блоков часто нам необходимо не только поворачивать доменный блок, но также и изменять его яркость и контрастность.

Для таких изменений блока используют *контрастность s* и *яркость o* – яркостные характеристики преобразования доменного блока к ранговому блоку.

Оптимальные контрастность и яркость минимизируют выражение

в котором и это соответственно значения цветовых компонент пискелей ранговой и доменной областей, а – длина стороны рангового и доменного блоков. Сами яркость и контрастность вычисляются по формулам

где

и – размер стороны рангового (доменного) блока.

По своей сути, фрактальное сжатие (или фрактальная компрессия) − это процесс поиска самоподобных областей изображения и определения для них параметров аффинных и яркостых преобразований.

Для реализации алгоритма компрессии необходимо выполнить следующие шаги:

1) исходное изображение разбивается на подобласти, которые представляют из себя квадраты, называемые *ранговыми блоками*. Ранговые блоки пересекаться не могут;

2) на исходном изображении выделяются *домены* – совокупности четырех ранговых блоков. Домены могут пересекаться. Все ранговые блоки и домены – это квадраты со сторонами, параллельными изображению;

3) для каждого рангового блока производится попытка найти на изображении домен, такой чтобы этот домен можно было преобразовать в ранговый блок при помощи аффинных преобразований;

4) перевод домена в ранговый блок производится с помощью поворота домена на 0°, 90°, 180°, 270° и с помощью вертикального и горизонтального зеркальных преобразований;

5) при переводе доменной области в ранговую, ее линейный размер уменьшается в 2 раза;

6) изменение яркости производится кратно некоторому коэффициенту;

7) совпадение преобразованного домена с ранговым блоком может производиться при помощи среднеквадратичного отклонения:

где – точка в домене; – точка в блоке; – пороговое значение «похожести»;

8) если же для некоторого рангового блока не было найдено ни одного удовлетворяющего среднеквадратичному отклонению домена, то ранговый блок разбивается на 4 подобласти, и для каждой из них ищутся домены.

Для осуществления декомпрессии необходимо задать базовое изображение, k – количество итераций декодирования, фрактальный код.

Алгоритм фрактальной декомпрессии включает следующие шаги:

1. задаем значение исходных данных;
2. из фрактального кода выделяем параметры преобразований для рангового блока;
3. на базовом изображении выделяем соответствующий параметрам доменный блок;
4. уменьшаем его;
5. применяем аффинное преобразование;
6. применяем яркостное преобразование;
7. копируем преобразованный доменный блок на место текущего рангового блока в базовом изображении;
8. если декодировали все ранговые блоки, то переходим к п.9, иначе – к п.2;
9. повторяем данный выполнение данного алгоритма k раз.

**Во второй главе** приведен логический проект разработанной системы, описанный с использованием нотации UML.

Методология UML является мощным средством проектирования, устранившим недостатки более ранних методологий, в том числе и основной недостаток SADT-методологии – отсутствие объектно-ориентированного представления моделей сложных систем. Основная задача, которая стояла при создании проекта – отобразить функциональность системы.

Разработка проекта системы выполнена в бесплатной среде UML-моделирования StarUML 5.0. На рисунке 1 приведена диаграмма вариантов использования разработанной системы.

**В третьей главе** описан выбор операционной системы, языка программирования и средств разработки, а так же приведены схемы алгоритмов обучения нейронной сети.

Важной составляющей разработки любого программного продукта является выбор операционной системы (ОС) – платформы, на которой он будет разработан. Операционная система определит в дальнейшем не только варианты пользовательского интерфейса, дружественность по отношению к пользователю, критичность к ошибкам, но и задаст, уже на этапе проектирования системы, ограничения в выборе программных средств, с помощью которых будет создаваться система. Для разработки была выбрана Microsoft Windows 7. Для реализации системы была выбрана среда Microsoft Visual Studio 2012 и язык C# 4.0. Интерфейс разработанной системы представлен на рисунке 2.

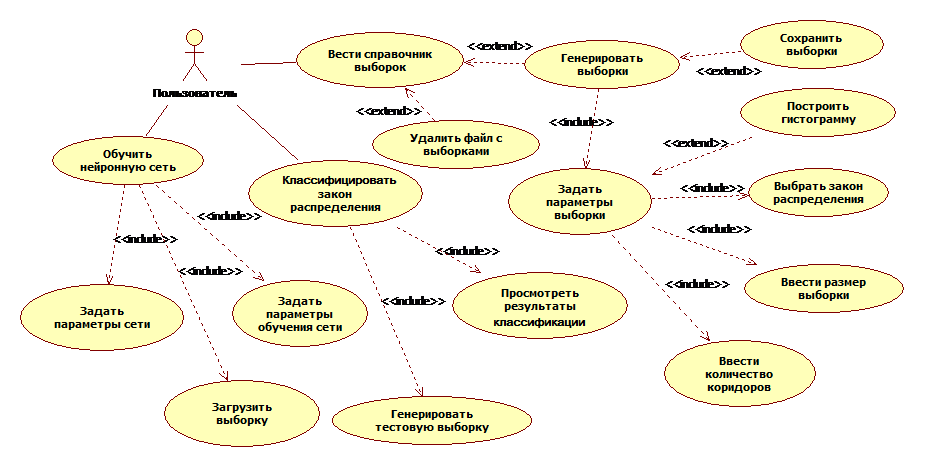
****

Рисунок 1 – Диаграмма вариантов использования системы

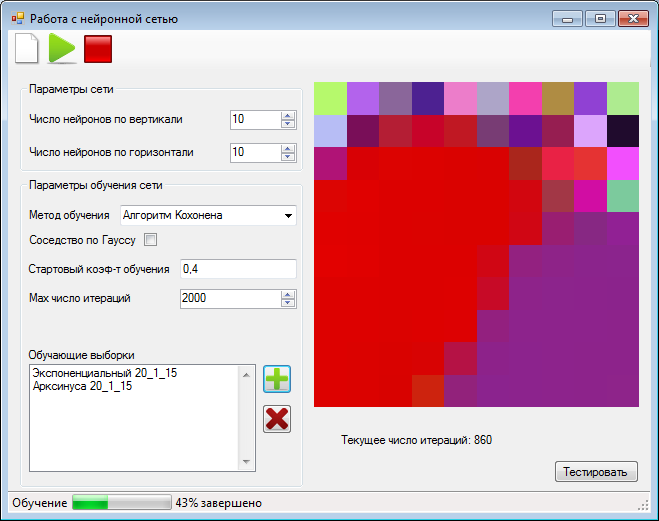


Рисунок 2 – Интерфейс разработанной системы

**В четвертой главе** исследованы классификационные возможности нейронной сети Кохонена при различных параметрах обучения и входных данных, а также произведено сравнение результатов классификации при одинаковых условиях при обучении различными алгоритмами.

Для сравнения классификационных возможностей сети с разными параметрами использовался в качестве характеристики ошибки процент верно распознанных законов распределения.

В таблице 2 представлены результаты исследования зависимости процента верно распознанных законов распределения от числа нейронов в сети.

Таблица 2 *–* Зависимость процента верно распознанных законов распределения от числа нейронов в сети

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Число нейронов | | | Процент верно распознанных законов | | | |
| a | b | Всего | WTA | Кохонена (прямоугольное соседство) | Кохонена (соседство по Гауссу) | Нейронного газа |
| 5 | 5 | 25 | 90 | 86 | 82 | 92 |
| 10 | 5 | 50 | 92 | 94 | 94 | 95 |
| 10 | 10 | 100 | 96 | 97 | 96 | 96 |
| 20 | 5 | 100 | 96 | 96 | 97 | 97 |
| 15 | 10 | 150 | 98 | 97 | 97 | 97 |
| **15** | **15** | **225** | **98** | **98** | **99** | **99** |
| **20** | **20** | **400** | **99** | **99** | **99** | **99** |

Из таблицы видно, что не имеет значения, какого типа карта – квадратная или прямоугольная, основным фактором является количество нейронов в сети. Наилучшие результаты показаны при количестве нейронов не меньше 225 и размерах решетки 15 на 15.

В разработанной системе осуществляется генерация последовательностей случайных чисел, распределенных по следующим законам: арксинуса, нормальному, Симпсона, Рэлея, экспоненциальному, Вейбулла, sech2 ax, Лапласа, Коши и равномерному.

Обучающие и тестовые данные для нейронной сети могут различаться по числу отсчетов (размеру) выборки.

В таблице 3 представлены результаты исследования зависимости процента верно распознанных законов от числа дифференциальных коридоров.

По данным таблицы 3 сеть одинаково хорошо настраивается на обучающие выборки с различным количеством коридоров.

Таблица 3 *–* Зависимость процента верно распознанных законов от числа дифференциальных коридоров

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Число диф. коридоров M | Процент верно распознанных законов | | | |
| WTA | Кохонена (прямоугольное соседство) | Кохонена (соседство по Гауссу) | Нейронного газа |
| 10 | 97 | 96 | 97 | 96 |
| **15** | **98** | **98** | **99** | **99** |
| 20 | 92 | 92 | 98 | 99 |

В таблице 4 представлены результаты исследования зависимости процента верно распознанных законов в режиме тестирования от размера выборки.

Таблица 4 *–* Зависимость процента верно распознанных законов от размера выборки

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Объем выборки N | Процент верно распознанных законов | | | |
| WTA | Кохонена (прямоугольное соседство) | Кохонена (соседство по Гауссу) | Нейронного газа |
| 250 | 90 | 90 | 92 | 92 |
| 500 | 92 | 92 | 94 | 94 |
| 1000 | 94 | 94 | 94 | 96 |
| 5000 | 96 | 92 | 94 | 96 |
| 10000 | 98 | 94 | 98 | 99 |
| **20000** | **98** | **98** | **99** | **99** |

Чем большее число отсчетов учитывается при построении гистограммы, тем она более точно построена и приближена к реальной функции плотности вероятности. Следовательно, сеть будет учиться на более верных и приближенных к тестовым и контрольным выборкам результатах. Однако стоит отметить, что нейронная сеть Кохонена, благодаря таким своим свойствам, как адаптивность, нечувствительность к ошибкам, способность к обобщению полученных знаний, достаточно успешно классифицирует объекты и данные, будучи обученной на выборках данных, построенных при малом количестве отсчетов от 250 до 500.

В таблице 5 представлены результаты исследования зависимости процента верно распознанных законов от количества обучающих выборок для каждого закона.

Таблица 5 *–* Зависимость процента верно распознанных законов от числа обучающих выборок для каждого закона

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Число обучающих выборок | Процент верно распознанных законов | | | |
| WTA | Кохонена (прямоугольное соседство) | Кохонена (соседство по Гауссу) | Нейронного газа |
| 100 | 74 | 75 | 80 | 88 |
| **1000** | **98** | **98** | **99** | **99** |
| 5000 | 86 | 88 | 89 | 94 |

По данным таблицы 5 наилучшие результаты дает сеть, обученная на 1000 выборках для каждого закона распределения.

В таблице 6 представлены результаты исследования зависимости процента верно распознанных законов от стартового коэффициента обучения.

Таблица 6 *–* Зависимость процента верно распознанных законов от стартового коэффициента обучения

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Стартовый коэффициент обучения | Процент верно распознанных векторов | | | |
| WTA | Кохонена (прямоугольное соседство) | Кохонена (соседство по Гауссу) | Нейронного газа |
| 0,05 | 91 | 92 | 94 | 96 |
| **0,2** | **98** | **98** | **99** | **98** |
| **0,4** | **97** | **98** | **99** | **98** |
| **0,8** | **95** | **96** | **97** | **99** |

Несмотря на сравнимые показатели для малых стартовых значений коэффициентов обучения по алгоритму WTA и реализациям алгоритма Кохонена и нейронного газа, вектора весов, которые были найдены нейронной сетью, обученной по алгоритму WTA, в процессе классификации имели гораздо более сильные отклонения весов от значений исходной выборки. Рекомендуется использовать значение коэффициента обучения из диапазона [0.2, 0.8].

По результатам проведенных исследований можно сделать вывод о достаточно хороших классификационных возможностях двумерной нейронной сети Кохонена. Процент верно распознанных законов при различных конфигурациях и параметрах сети, обучающих данных и параметрах обучения составляет 74*–*99%. Однако стоит отметить, что обучение по алгоритму WTA не обеспечивает достаточно полного охвата всей карты Кохонена, некоторые нейроны так и остаются «мертвыми» в процессе обучения, даже несмотря на учет активности каждого нейрона.

Важно отметить, что карты Кохонена обладают существенным достоинством – это удобная визуализация процесса обучения сети. Веса нейронов определяют их цвет на карте. Таким образом, пользователь может сделать вывод о принадлежности входной выборки к какому-либо закону распределения, не руководствуясь выводами системы, а просто взглянув на обученную карту Кохонена.

Также стоит отметить, что обучение сети по алгоритму Кохонена позволяет визуально разделить карту на четко очерченные области, отвечающие за классификацию определенного закона, что не позволяют осуществить алгоритмы WTA и нейронного газа, где нейроны, отвечающие за идентификацию одного и того же закона более разрознены.

**В заключении** сформулированы основные выводы, перечислены полученные в работе результаты.

основные выводы и результаты

1. Разработана автоматизированная система, в которой реализованы следующие алгоритмы обучения нейросети: алгоритм WTA, алгоритм Кохонена с двумя типами соседства нейронов, алгоритм нейронного газа. В системе реализована генерация псевдослучайных последовательностей, распределенных по заданному закону распределения. Также визуализирован процесс обучения нейросети.
2. Определены оптимальная конфигурация сети, оптимальные параметры обучения сети и обучающих данных для решения задачи классификации.
3. Произведен сравнительный анализ результатов решения задачи классификации при обучения нейросети тремя различными алгоритмами.

Основные публикации по теме диссертации

**Работы, опубликованные в ведущих рецензируемых научных журналах и изданиях, определенных Высшей аттестационной комиссией:**

* 1. Лёзина, И.В. Автоматизированная система идентификации законов распределения двумерной нейронной сетью Кохонена [Текст]/И.В. Лёзина, Н.Ю. Яшин//Известия Самарского научного центра Российской академии наук. Том 16 №4(2), 2014 – С. 344-350

**Тезисы докладов:**

* 1. Лёзина, И.В. Автоматизированная система идентификации законов распределения нейронной сетью Кохонена [Текст]/И.В. Лёзина, Н.Ю. Яшин//Современное общество, образование и наука: сб. науч. тр. по мат-лам Междунар. заоч. науч-практ. конф. 25 июня 2012 г.: в 3 частях. Часть 2/Тамбов: Изд-во ТРОО «Бизнес-Наука-Общестов», 2012. – 163 с.
  2. Лёзина, И.В. Идентификация законов распределения нейронной сетью Кохонена [Текст]/И.В. Лёзина, Н.Ю. Яшин//Перспективные информационные технологии (ПИТ 2013): труды Международной научно-технической конференции / под ред. С.А. Прохорова. – Самара: Издательство Самарского научного центра РАН, 2013. – С. 128-132

**Патент:**

* 1. Свидетельство о государственной регистрации программы для ЭВМ № 2014613802 «Автоматизированная система идентификации законов распределения двумерной нейронной сетью Кохонена», дата государственной регистрации в Реестре программ для ЭВМ 08 апреля 2014 г.