



ся на вход сети. Параметры модели подбираются таким образом, чтобы уменьшить погрешность прогнозирования [2, 3].

Для проведения вычислительных экспериментов были использованы показатели цен на нефть марки Brent и WTA за период с 31.12.2005 по 31.12.2016 [4].

По результатам эксперимента можно сделать вывод, что случайная инициализация весовых коэффициентов дает результаты хуже, чем эволюционные алгоритмы и алгоритм имитации отжига. Сравнивая два последних алгоритма можно сказать, что при определенной настройке сети, они дают приблизительно равные результаты.

Литература

1. Осовский, С. Нейронные сети для обработки информации // Пер. с польского И.Д. Рудинского, М.: Финансы и статистика, 2002. – 79 с., 82 с., 87 с., 344 с.: ил.
2. Лихачева, Д.В. Автоматизированная система прогнозирования изменения курса валют на основе сети Вольтерри // XII Королёвские чтения: Международная молодёжная научная конференция: тезисы докладов, Самара: Издательство СГАУ, 2013. – С. 209 - ISBN 978-5-7883-0952-1
3. Лёзина И.В., Сараева К.В. Автоматизированная система прогнозирования нефтяных котировок нейронной сетью Вольтерри // Вестник научных конференций. 2016. №9(13). Ч.2.С. 68-69
4. Интернет-трейдинг "ФИНАМ" [Электронный ресурс]. - <https://www.finam.ru/>

В.Б. Сахибазарова, М.А. Кудрина

ИССЛЕДОВАНИЕ ЗАВИСИМОСТИ СКОРОСТИ ФРАКТАЛЬНОГО СЖАТИЯ ИЗОБРАЖЕНИЯ ОТ ПАРАМЕТРОВ СЖАТИЯ

(Самарский университет)

В настоящее время сложно представить себе область деятельности человека, не включающую в себя, хоть в малой степени, необходимость обмена информацией по сети Интернет. При использовании сети важно учитывать два критерия: скорость передачи информации и объем передаваемых данных. Необходимо передать как можно больше информации в сообщении наименьшего размера. В случае передачи графической информации используются различные методы сжатия изображений для уменьшения объема передаваемых данных.

В данной работе рассматривается алгоритм фрактального сжатия изображений, основанный на том, что мы представляем изображение в более компактной форме – с помощью коэффициентов системы итерированных функций Iterated Function System (IFS). IFS представляет собой набор трехмерных аффинных преобразований, переводящих одно изображение в другое. Преобразо-



ванию подвергаются точки в трехмерном пространстве (x_координата, y_координата, яркость) [1].

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

Для реализации алгоритма компрессии необходимо учитывать следующие правила:

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 зеркального преобразования;

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

б) изменение яркости производится кратно некоторому коэффициенту;

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

$$\sum (x_{\text{дом}} - x_{\text{блк}})^2 < \varepsilon_{\text{доп}},$$

где $x_{\text{дом}}$ – точка в домене; $x_{\text{блк}}$ – точка в блоке; $\varepsilon_{\text{доп}}$ – 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;

9) координаты, которые будут сохраняться в файл:

- координата 2 числа;
- 3 бита (номер аффинного преобразования);
- изменение яркости.

При фрактальном сжатии может использоваться несколько различных алгоритмов разбиения на ранговые блоки; также программная реализация сжатия отличается для цветных изображений и изображений в градациях серого.

Для проведения данного исследования была разработана программа, производящая фрактальное сжатие изображения в градациях серого с использованием алгоритма разбиения изображения на множество блоков фиксированного размера.

Исследование проводилось над изображением, сохраненным в размерах 64×64 , 160×160 и 312×312 . Исследовалась зависимость времени сжатия от раз-



мера рангового блока и коэффициента компрессии ϵ (порогового значения «похожести»).

Результаты исследований можно видеть на следующих рисунках.

Как видно из рисунка 1 уменьшение или увеличение рангового блока не может гарантированно уменьшать или увеличивать время компрессии. Из сравнения рисунков 1 и 2 можно сделать вывод, что размер рангового блока влияет на общую тенденцию изменения времени компрессии (которая для каждого изображения индивидуальна и может зависеть от степени его «детальности»), а коэффициент компрессии влияет на качество сжатия и, как следствие на время компрессии (чем ниже коэффициент, тем чаще ранговый блок разбивается на подобласти и тем больше доменных блоков приходится искать). Из рисунка 3 следует что увеличение коэффициента компрессии для любого размера изображения приводит к уменьшению времени компрессии.

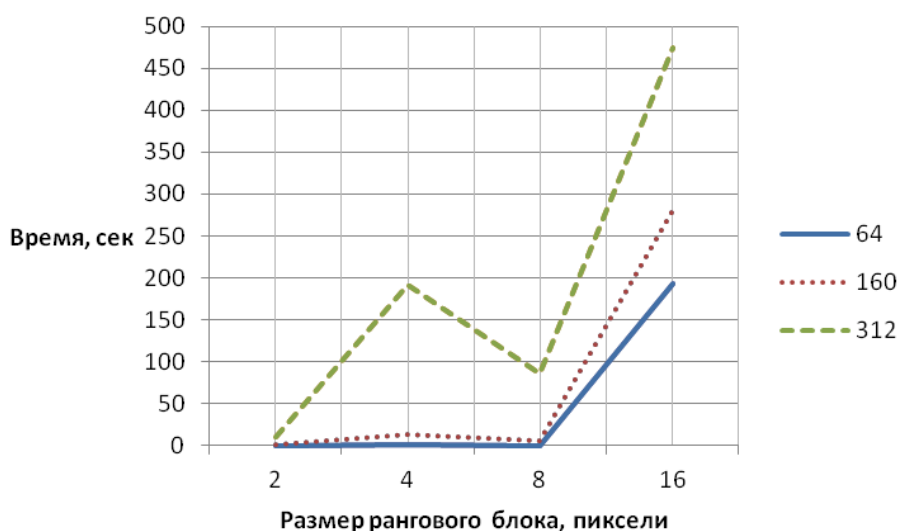


Рисунок 1 – Зависимость времени компрессии изображений разных размеров от размера рангового блока

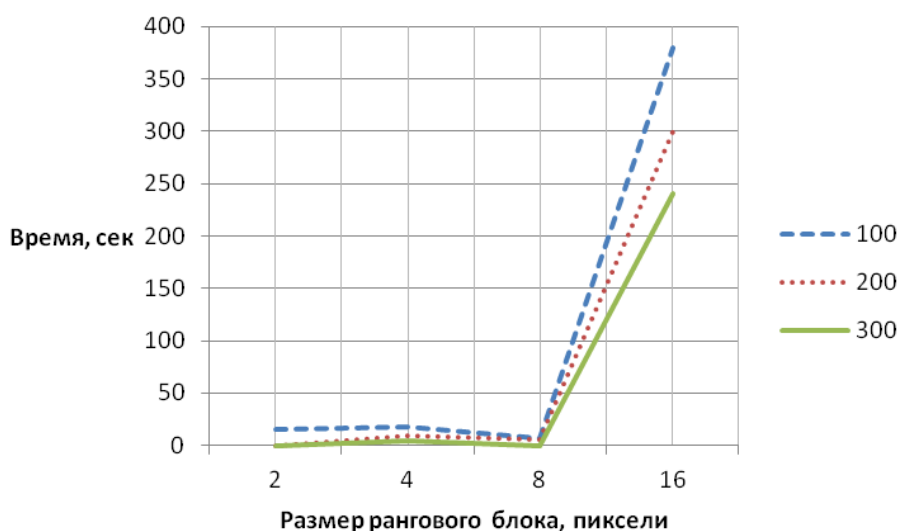


Рисунок 2 – Зависимость времени компрессии изображения размером 160×160 от размера рангового блока и коэффициента компрессии

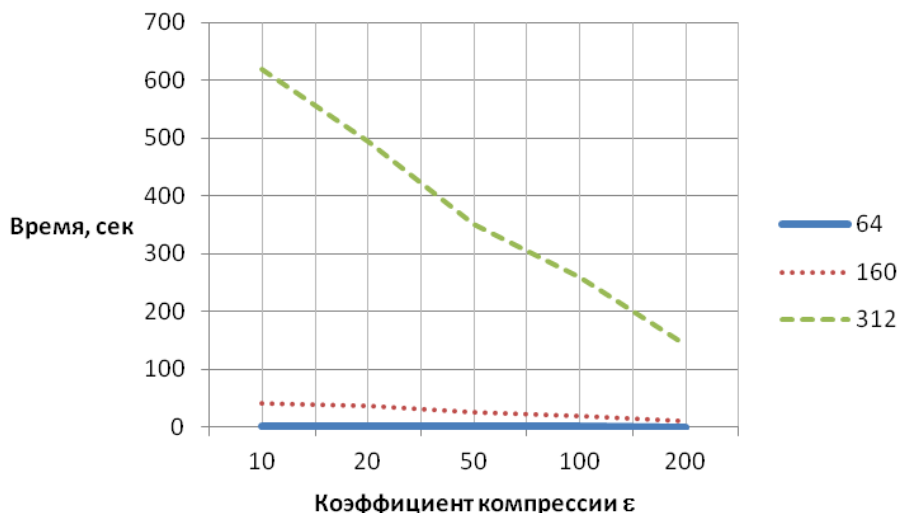


Рисунок 3 – Зависимость времени компрессии изображений разного размера от коэффициента компрессии.

Литература

1 Кудрина М.А., Климентьев К.Е. Компьютерная графика. – Издательство СГАУ, 2013. – 140 с.

В.А. Семенова

ПОСТРОЕНИЕ ФОРМАЛЬНЫХ ОНТОЛОГИЙ НА ОСНОВЕ ПРОТИВОРЕЧИВЫХ ЭМПИРИЧЕСКИХ ДАННЫХ

(Самарский государственный аэрокосмический университет)

Сложные информационные системы результативны лишь при надежном и согласованном представлении их предмета. Систематизация, разработка и использование соответствующих информационных моделей составляют современное содержание *онтологического подхода* в вычислительных науках.

В отличие от философии онтология в информатике описывает некоторую ограниченную сферу знания, *предметную область* (ПрО). Различают лингвистические и *формальные онтологии*, где последние наследуют парадигму моделей и методов представления знаний, разработанных в искусственном интеллекте [1]. В простом случае онтология описывает ПрО с помощью совокупности формальных понятий и неоднородных связей между ними.

Сегодня можно указать три основных пути разработки онтологий [2]:

- прямая формализация опыта и знаний экспертов [3];
- синтезирование актуальных онтологий в результате человеко-машинных процедур конкретизации, композиции/декомпозиции апробированных формальных онтологий, их прототипов разного уровня и направленности [4];
- автоматический «вывод» формальных онтологий из доступных эмпирических данных. Эти данные рассматриваются как результат измерений