

УДК 62-11

А.В. Дагаев, Ю.М. Бородянский, М.М. Бойченко

**АНАЛИЗ ГРУППЫ ОБЪЕКТОВ ОДНОГО КЛАССА С ПРИМЕНЕНИЕМ
МАТЕМАТИЧЕСКИХ МЕТОДОВ**

*Представлена применяющая фильтры методика определения границ торцов труб, дается алгоритм определения объектов и подсчет их площади.
Алгоритм, объект.*

A.V. Dagaev, Y.M. Borodyansky, M.M. Boichenko

**ANALYSIS OF OBJECTS OF ONE CLASS WITH THE USE OF
MATHEMATICAL METHODS**

*The technique of delimitation of end faces of pipes applying filters is presented, the algorithm of definition of objects and calculation of their area is given.
Algorithm; object.*

Введение. В последние десятилетия развитие компьютерной техники и информационных технологий позволило сделать большой шаг во всех областях науки. Так, например, высококачественные фотографии можно просматривать и редактировать, используя обычные компьютеры и современные графические пакеты. Но анализировать графические объекты даже с использованием стандартных пакетов является трудоемкой задачей. Поэтому сегодня развиваются методы анализа графических объектов, позволяющие распознать заданные объекты и сократить время анализа. Подобные методы используются во многих областях, таких как: дактилоскопия, распознавание текста, лиц людей, номеров машин и др. Без компьютерной автоматизации процесса распознавания это было бы невозможно. Анализ объектов востребован не только в представленных выше областях, но и на производстве. У производителей и потребителей трубопрокатной продукции, леса, пиломатериалов и др. есть острая необходимость автоматизации процесса подсчета имеющейся в наличии продукции по ее фотографии. Для этого приходится разрабатывать новые и применять существующие методы анализа объектов. Разработке методики анализа подобных объектов посвящена данная статья.

Постановка задачи. Перед авторами стояла цель – разработать методику анализа образов определенных объектов. Для ее решения были поставлены следующие задачи:

- ◆ разработать методику анализа;
- ◆ разработать программу анализа объектов, на основе разработанной методики.

Выделение контуров объектов. На сегодняшний день известно много алгоритмов и методов распознавания образов, например нейросетевые методы, кластерный анализ, Марковские модели и др. [1,2]. Все они направлены на поиск и определение объектов, представленных на изображении. В данной работе анализировались цветные фотографии труб, которые далее преобразовывались в серые. Для анализа графических объектов необходимо было реализовать решение двух задач: выделение границ объектов, определение выделенного объекта к определенному классу геометрических фигур. В основном интересовали объекты, которые удовлетворяют признакам окружности.

Итак, решение первой задачи достигается путем применения различных фильтров и преобразований исходного изображения. На диаграмме, расположенной на рис. 1, изображен алгоритм выделения контуров объектов.

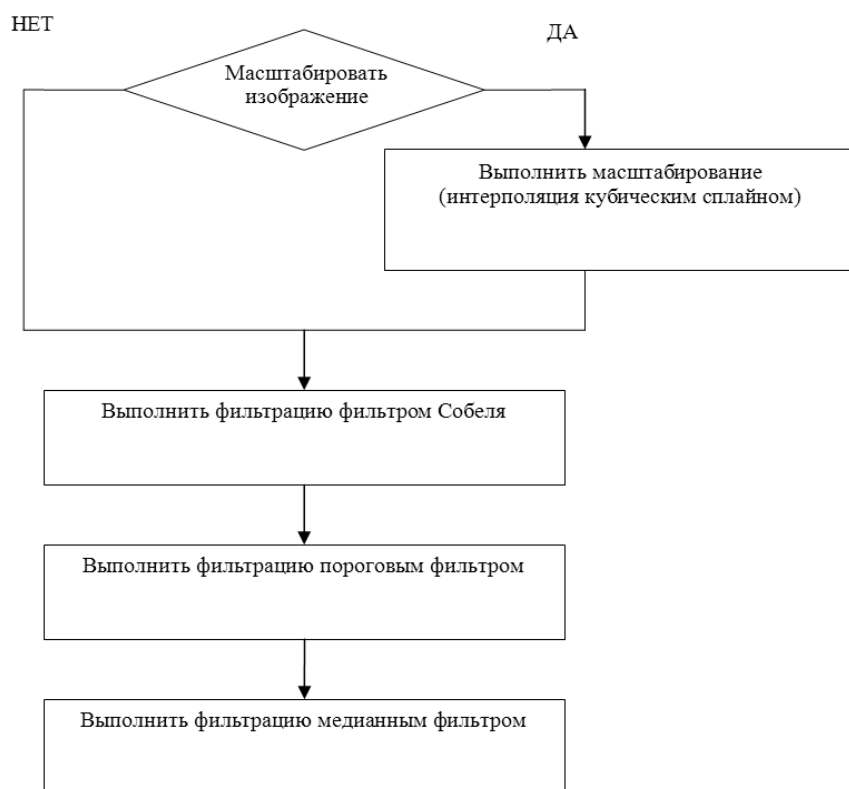


Рис. 1. Алгоритм выделения контуров объектов

Рассмотрим подробнее все этапы алгоритма:

1. При получении изображения необходимо проверить его размеры, и при необходимости выполнить масштабирование, в нашем случае уменьшение исходного изображения (рис. 2).

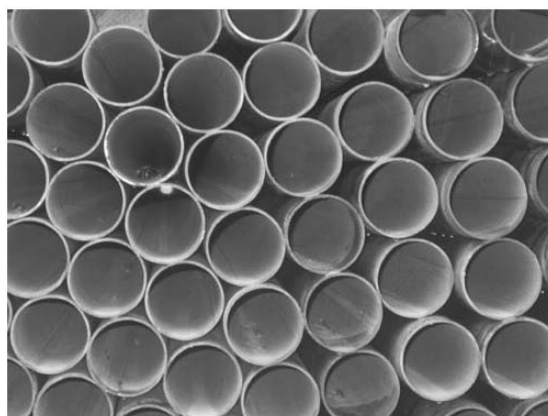


Рис. 2. Исходное изображение

2. Масштабирование выполняется в том случае, если необходимо повысить скорость работы алгоритма: чем меньше точек в матрице изображения, тем меньше нужно выполнять вычислений при фильтрации изображения (в нашем случае необходимо выполнить три фильтра). Масштабирование (одна из составляющих частей ресамплинга) выполняется при помощи интерполяции кубическим сплайном. Также уменьшение изображения влияет и на качество обнаружения границ: при кубической интерполяции учитываются несколько соседних пикселей и после уменьшения изображения пиксели «уплотняются», т.е. вместо группы из N пикселей, разница цветов между которыми небольшая, получается группа из m ($m < N$) пикселей, причем разность цветов между ними увеличивается. Это обстоятельство особенно важно для фильтра Собеля. То есть масштабирование решает две подзадачи: увеличение скорости работы алгоритма (за счет уменьшения количества элементов в матрице изображения), увеличение разности цветов между пикселями.
3. Фильтр, или преобразование, Собеля – это самый основной шаг алгоритма. Данный фильтр относится к классу разностных фильтров, служащих для решения задач выявления границ объектов на изображении (рис. 3).

Фильтр Собеля имеет следующий вид: $M = \frac{1}{4} \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$. При при-

менении данного фильтра мы выделим вертикальные перепады цветов, т.е. границы, объектов изображения. Если же транспонировать матрицу и провести фильтрацию, то выделяются горизонтальные перепады цветов изображения. Этот факт связан с тем, что средний столбец (или строка для транспонированной матрицы) состоит из нулевых элементов, а столбцы слева и справа имеют элементы разных знаков. При таком фильтре как раз и получаем сумму разностей цветов вокруг заданного пикселя. Для того чтобы учитывать перепады цветов горизонтальных и вертикальных, необходимо дважды профильтровать изображение фильтрами M и M^T , после чего можно получить разность перепадов цветов вокруг выбранного пикселя, которая равна

$$R_{x,y} = \sqrt{\Delta X^2 + \Delta Y^2},$$

$$\Delta X^2 = \sum_{i,j} M_{i,j} \text{ЦветПикселя}_{x,y} \quad \Delta Y^2 = \sum_{i,j} M_{i,j}^T \text{ЦветПикселя}_{x,y}.$$

После вычисления $R_{x,y}$ для всех пикселей изображения, получим матрицу, где цвет каждого пикселя будет заменен на показатель его отличия от других соседних пикселей изображения. Чем больше этот показатель, тем больше вероятность того, что рассматриваемый пиксель является граничным, т.е. цвета соседних от него пикселей отличаются достаточно сильно.

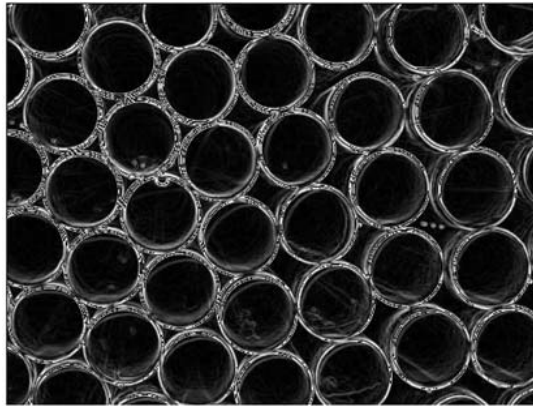


Рис. 3. Изображение после применения фильтра Собеля

4. После преобразования Собеля мы получаем матрицу, элементы которой показывают насколько пиксель изображения соответствует граничному пикселю объекта. Теперь необходимо проанализировать получившуюся матрицу и устранить возникшие шумы. Это можно сделать при помощи порогового фильтра. К классу таких фильтров в общем случае можно отнести фильтр следующего вида:

$$L_{x,y} = \begin{cases} 1 & \text{если } A_{x,z} > \alpha \\ 0 & \text{иначе} \end{cases},$$

где α – некоторое пороговое значение. Если значение элемента полученной матрицы превышает значение порога, то мы считаем, что этот элемент является граничным, иначе нет. Величина порога может быть выбрана весьма разнообразно: константа, среднее арифметическое значение определенной области матрицы изображения, среднее арифметическое всего изображения и так далее. Также числа 0 и 1 можно заменить на любые произвольные, главное знать какое число соответствует границы объекта. После применения порогового фильтра получаем матрицу, элементы которой будут иметь всего два значения (рис. 4).

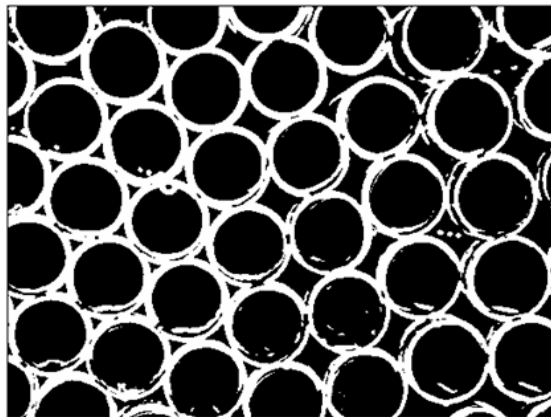


Рис. 4. Результат работы пороговой фильтрации

5. На последнем этапе можно применить медианную фильтрацию, которая позволяет убрать определенное количество шума. Медианным фильтром называют фильтр следующего вида: $B_{x,y} = \text{median}\{N_{i,j}\}$. Операция взятие медианы выполняется над массивом элементов и в результате выполнения выдает средний элемент данного массива. Если массив состоит из пяти элементов, то для вычисления медианы необходимо отсортировать массив и взять средний элемент, т.е. третий по порядку. Медианная фильтрация позволяет убрать шум, который напоминает снег на изображении. То есть единичные светлые пиксели на темном фоне будут заменены на темные пиксели, и наоборот – темные на светлом фоне на светлые.

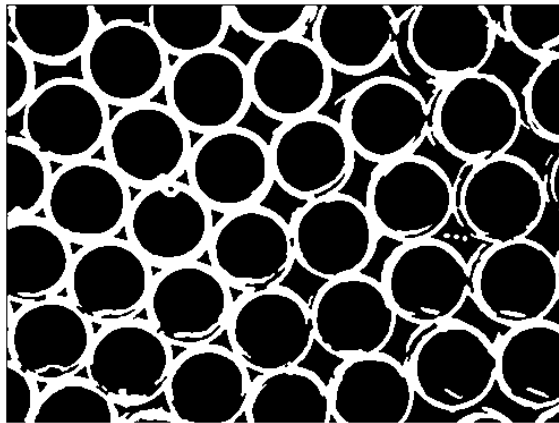


Рис. 5. Применение медианной фильтрации (результат работы алгоритма)

Анализ графических образов. Выделение признаков позволяет упростить реализацию распознавания объектов. При выборе наиболее информативных признаков необходимо учитывать как свойства самих объектов, так и возможности решающей способности первичных формирователей сигнала изображения.

При обработке предпочтительными обычно являются следующие геометрические признаки объектов:

- ◆ площадь и периметр изображения объекта;
- ◆ размеры вписанных простейших геометрических фигур (окружностей, прямоугольников, треугольников и др.);
- ◆ число и взаимное расположение углов;

Определение площади и периметра. Площадь изображения объекта вычисляется путём подсчёта числа элементов, относящихся к объекту:

$$A = \sum_x \sum_y S(x, y), \quad S(x, y) = \begin{cases} 1, & S(x, y) \in L \\ 0, & S(x, y) \notin L \end{cases}$$

где L – множество координат массива $S(x, y)$, принадлежащих выделенному объекту.

Периметр изображения объекта P вычисляется после выделения границ объекта путем суммирования множество контурных точек изображения объекта.

На основе выделенных признаков нормированный признак, инвариантный к масштабу изображения $U = A/P^2$ или $V = P/\sqrt{A}$.

Определение радиусов вписанных и описанных окружностей складывается из двух этапов:

1. Определение координат геометрического центра изображения объекта:

$$X_{\text{ц}} = \sum_x \sum_y xS(x, y) / \sum_x \sum_y S(x, y), \quad Y_{\text{ц}} = \sum_x \sum_y yS(x, y) / \sum_x \sum_y S(x, y),$$

где x и y – номера строк и столбцов всех пикселей $S(x, y)$, входящих в объект.

2. Вычисление минимального и максимального расстояний от центра до границ изображения объекта.

$$r(x, y) = \sqrt{(x - X_{\text{ц}})^2 + (y - Y_{\text{ц}})^2}.$$

$$R_{\text{max}} = r(x, y_{\text{max}}); \quad R_{\text{min}} = r(x, y_{\text{min}}),$$

где $x, y \in P$ (точки периметра).

Определение сторон описанного прямоугольника выполняется следующим образом. Определяются максимальные и минимальные значения абсцисс и ординат изображения объекта x_{max} и x_{min} , y_{max} и y_{min} , после чего определяются высота и основание прямоугольника:

$$L = x_{\text{max}} - x_{\text{min}}, \quad H = y_{\text{max}} - y_{\text{min}}.$$

Определение числа и взаимного положения углов. Классический способ определения угловых точек изображения объекта заключается в анализе небольшого фрагмента контура в окрестностях данной точки и в определении радиуса её кривизны. Если радиус окажется меньше установленного порога – это угловой элемент, в противном случае – нет.

С практической точки зрения предпочтительным представляется более простой алгоритм. Он заключается в оценке расстояний между начальной и конечной точками фрагмента контура, например, между элементами контура с порядковыми номерами $k-2$ и $k+2$, что представлено на рис. 6.

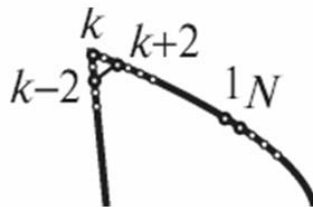


Рис. 6. Точки фрагмента контура

Пусть $x(k)$ и $y(k)$ – абсцисса и ордината контурных элементов соответственно. Тогда решающее правило может выглядеть следующим образом:

$$\{|x(k-2) - x(k+2)| + |y(k-2) - y(k+2)| \leq H\}.$$

Если условие выполняется, тогда данная точка контура принадлежит множеству угловых точек L . Здесь H – пороговое значение, выбираемое с учётом свойств объектов [1].

Блок-схема алгоритма, используемая при решении поставленной задачи изображена на диаграмме, представлена на рис. 7.

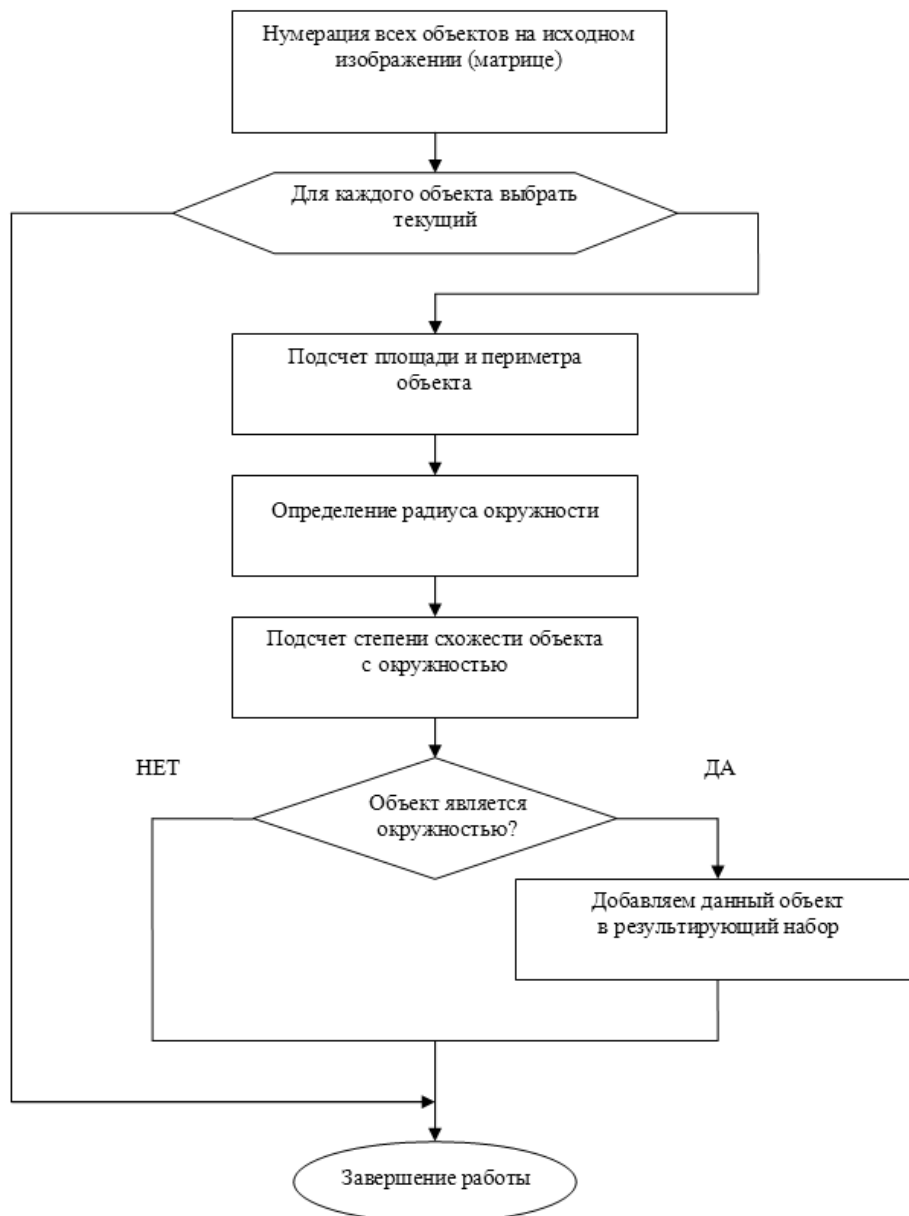


Рис. 7. Блок-схема алгоритма, используемая при решении поставленной задачи

Рассмотрим подробнее все этапы алгоритма:

1. Сразу же следует отметить, что алгоритм работает не с пикселями изображения, а с матрицей цветов, построенной на данном изображении. Это вызвано тем, что доступ к пикселям очень долгая операция. С целью ускорения работы вся работа ведется с матрицей, тем более что после определения границ матрица состоит только из элементов двух цветов, а именно черного (цвет фона) и белого (цвет границ). При такой организации достаточно легко пронумеровать все объекты: просто выбираем эле-

мент с цветом фона и начинаем заливать область каким-нибудь значением цвета. Заливку производим при помощи алгоритма заливки четырехсвязной области. Заливать будем замкнутую область заранее известным цветом. При этом границами заливаемой области являются элементы белого цвета, т.е. границы самих объектов. Каждую новую заливку делаем при помощи цвета, который ранее не встречался. После того как в матрице не останется черных элементов, прекращаем заливку и получим уже не черно-белую матрицу, а цветную. При этом у каждого объекта будет свой цвет, а границей по-прежнему будет белый цвет. По данному цвету заливки можно будет определить нумерацию всех объектов.

2. Далее по циклу выбираем все пронумерованные объекты (цвет используется в качестве номера объекта). Определив текущий объект, начинаем рассчитывать его геометрические параметры: периметр, площадь, радиус окружности.
3. Зная цвет объекта и цвет его границ (белый), достаточно просто рассчитываются площадь и периметр объекта. Данный расчет описан выше.
4. Для расчета радиуса предполагаемой окружности необходимо вычислить радиус вписанной и описанной окружностей для данного объекта. После этого берем среднее арифметическое двух этих радиусов и считаем, что именно это значение – радиус данного объекта.
5. После вычисления геометрических показателей необходимо рассчитать степень того, насколько выбранный объект соответствует окружности – для этого по ранее определенному радиусу вычисляем геометрическую площадь и периметр окружности и сравниваем данное значение с физическим подсчитанным значениями площади и периметра.
6. На последнем этапе определяем, является ли объект окружностью: просто смотрим, насколько значения рассчитанных и геометрических значений совпадают, удовлетворяет ли нас степень погрешности. Если степень погрешности нас удовлетворяет, тогда считаем, что текущий объект является окружностью, добавляем данный объект в результирующий набор.

Следует заметить, что для определения и анализа объектов произвольной формы требуется использовать ряд других методов, например градиентных [4].

Выводы. В ходе проделанной работы был выполнен ряд поставленных задач:

1. Проведен анализ существующих методов распознавания образов.
2. Выбраны методы, позволяющие анализировать информацию об объекте.
3. Разработана методика анализа графических объектов, подобраны рабочие коэффициенты фильтров.
4. Проведены тестовые испытания по распознаванию объектов.

В дальнейшем предполагается развить методику таким образом, чтобы она позволяла распознавать объекты заранее заданной геометрической формы.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *А.С. Потапов.* Распознавание образов и машинное восприятие. – М.: Политехника, 2007. – 552 с.
2. Компьютеры. Распознавание образов. Электронный справочник. Руссобит-М, 2006. CD-ROM.
3. *Андреев А.Л.* Автоматизированные телевизионные системы наблюдения. Часть II. Арифметико-логические основы и алгоритмы: Учебное пособие. – СПб.: СПб, ГУИТМО, 2005. – 88 с.
4. *Гашиков М.В., Глумов Н.И., Сойфер В.А.* Методы компьютерной обработки изображений. – М.: Физматлит, 2003. – 784 с.

Дагаев Александр Владимирович

Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

E-mail: adagaev@list.ru.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 88634371980.

Бойченко Михаил Михайлович

E-mail: m.boichenko@tti.sfedu.ru.

Тел.: 88634371980.

Бородянский Юрий Михайлович

E-mail: borodyansky@yandex.ru.

Тел.: 88634371787.

Кафедра системного анализа и телекоммуникаций; доцент.

Dagaev Aleksander Vladimirovich

Taganrog Institute of Technology – Federal State-Owned Educational Establishment of Higher Vocational Education “Southern Federal University”.

E-mail: adagaev@list.ru.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Phone: 88634371980.

Boichenko Mixail Mixailovich

E-mail: m.boichenko@tti.sfedu.ru.

Phone: 88634371980.

Borodyanskiy Yuriy Mixaylovich

E-mail: E-mail: borodyansky@yandex.ru.

Phone: 88634371787.

The Department of System Analysis and Telecommunications; associate professor.

УДК 519.1

Р.А. Кочкаров, И.Х. Утакаева

**АЛГОРИТМЫ РАСПОЗНАВАНИЯ ПРЕДФРАКТАЛЬНЫХ ГРАФОВ
С РАЗЛИЧНЫМИ ЗАТРАВКАМИ**

Рассматриваются задачи распознавания различных предфрактальных графов. Произведена математическая постановка, разработаны эффективные алгоритмы распознавания исследуемых предфрактальных графов.

Граф; алгоритм; распознавание.

R.A. Kochkarov, I.H. Utakaeva

**ALGORITHMS OF RECOGNITION OF PREFRACTAL GRAPHS WITH
VARIOUS PRIMING**

Problems of recognition of various prefractal graphs are considered. Mathematical statement is made, effective algorithms of recognition of investigated prefractal graphs are developed.

Graph; algorithm; pattern recognition.

Распознавание образов – едва ли не самая распространенная задача, которую человеку приходится решать практически ежесекундно. Для этого человек использует огромные ресурсы своего мозга, включая одновременно около 10 млрд нейронов. Именно это дает возможность людям мгновенно узнавать друг друга, читать