

УДК 004.931

## АЛГОРИТМИЧЕСКОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ РАСПОЗНАВАНИЯ ФИГУР С ПОМОЩЬЮ ФУРЬЕ-ДЕСКРИПТОРОВ И НЕЙРОННОЙ СЕТИ

Тоан Тханг Нгуен

Томский политехнический университет

Email: thangngt.cntt@gmail.com

Приведен обзор простых сигнатур фигур на основе контура. Предложены алгоритмы и создано приложение для распознавания фигур с использованием Фурье-дескрипторов и многоуровневой нейронной сети. Сделан вывод о возможности использования Фурье-дескрипторов в качестве входных данных для нейронных сетей при распознавании сложных фигур.

### Ключевые слова:

Фурье-дескрипторы, распознавание фигур, нейронные сети, многоуровневый персептрон.

### Key words:

Fourier descriptors, shape recognition, neural networks, multilayer perceptron.

### Введение

Распознавание образов является центральной задачей в области машинного зрения. Распознавание фигур является частным случаем распознавания образов. Многие задачи распознавания объектов на изображениях могут сводиться к распознаванию фигур. Среди различных способов представления фигур Фурье-дескрипторы доказали свою эффективность и простоту в реализации [1]. В данной работе Фурье-дескрипторы применены совместно с нейронной сетью для решения задачи распознавания фигур.

### Обзор простых методов представления фигур на основе контура

Существующие подходы для представления фигур можно охарактеризовать следующим образом: это методы, основанные на контур[0]е и методы, основанные на области, пространственном домене и домене преобразования; информационно-сохраняющие (IP) и информационно-несохраняющие методы (NIP). Однако подходы к выделению и представлению фигур обычно разделяются, в зависимости от способов обработки, на одномерные функции представления фигуры (*One-dimensional function*), аппроксимацию полигонов (*Polygonal approximation*), взаимосвязь пространственных признаков (*Spatial interrelation feature*), моменты (*Moments*), методы деления шкалы (*Scale-space methods*), домены преобразования фигуры (*Shape transform domains*) [2].

Для представления несложных фигур на основе контуров часто используют: комплексные координаты, функцию расстояния, касательный угол, кривизну контура и Фурье-дескрипторы. Все эти методы (кроме Фурье-дескрипторов) входят в класс «одномерные функции представления фигуры».

#### Комплексные координаты

Допустим, что изображение представляется в виде функции  $f(x, y)$ , и  $P_n = (x_n, y_n)$ ,  $n = [1, N]$  является множеством точек на границе (контуре) фигуры.

В этом случае  $z_n = x_n + iy_n$  называется комплексной координатой. Эти комплексные числа можно использовать в качестве характеристики или дескриптора фигуры или в качестве входных данных для Фурье-преобразования. Тогда контура обозначается в виде функции  $P_n = z_n$ ,  $n = [1, N]$ .

Этот способ очень простой в реализации, но он имеет ряд недостатков: получаемый результат является неинвариантным к перемещению, масштабированию и вращению. Чтобы комплексные координаты были инвариантными к перемещению,  $z_n$  считаются с учетом центра тяжести (центроид):  $z_n = (x_n - x_g) + i(y_n - y_g)$ , где  $g = (x_g, y_g)$  – центр тяжести фигуры.

#### Функция расстояния

Функция расстояния  $R_n$  для контура  $P_n = (x_n, y_n)$ ,  $n = [1, N]$  вычисляется как расстояние каждой точки  $(x_n, y_n)$  от неподвижной точки  $C(x_0, y_0)$ . В качестве  $C$  часто выбирается центроид фигуры [2].

$$R_n = \sqrt{(x_n - x_0)^2 + (y_n - y_0)^2}.$$

Функция расстояния имеет те же преимущества и недостатки, что и комплексные координаты.

#### Касательный угол

Каждый контур считается кривой линией, поэтому можно рассчитать угол прямой касательной к каждой его точки [2]:

$$\theta_n = \arctan \left( \frac{y_n - y_{n-w}}{x_n - x_{n-w}} \right).$$

Здесь  $w$  – окно небольшого размера.

Несмотря на простоту реализации, метод имеет два больших недостатка: чувствительность к шуму и прерывность. Чтобы избежать прерывности определяется кумулятивная угловая функция  $\varphi_n = \theta_n - \theta_0$ , где  $\theta_0$  – касательный угол к случайной выбранной точке на контуре. Перед тем, как рассчитать эту функцию, часто применяется фильтр нижних частот. В настоящей работе используется кумулятивная угловая функция как исходная функция для Фурье-преобразования.

Фурье-дескрипторы

Фурье-дескрипторы получаются путем применения Фурье-преобразования к вышеуказанным одномерным функциям представления фигуры [3, 4]. Фурье-дескрипторами называются нормированные коэффициенты Фурье-разложения. Предположим, что контур объект обозначается непрерывной и периодичной функцией  $c(t)$ , и что

$$a_k = \frac{2}{T} \int_0^T c(t) \cos(kt) dt,$$

$$b_k = \frac{2}{T} \int_0^T c(t) \sin(kt) dt,$$

$$c_k = \sqrt{a_k^2 + b_k^2}.$$

Здесь  $a_k$  – реальная часть;  $b_k$  – мнимая часть;  $c_k$  – Фурье-дескриптор.

Фурье-дескрипторы устойчивы к перемещению, масштабированию и вращению объекта [2, 3] и, следовательно, идеально подходят для представления фигуры.

#### Алгоритм и его реализация

Процесс реализация системы состоит из двух этапов: обучения и тестирования. Общая схема алгоритма показана на рис. 1.

База данных для обучения содержит 20 «чистых» изображений с одним объектом на черном фоне. Объекты принадлежат классам: окружности, треугольники, прямоугольники, и полигоны (рис. 2).

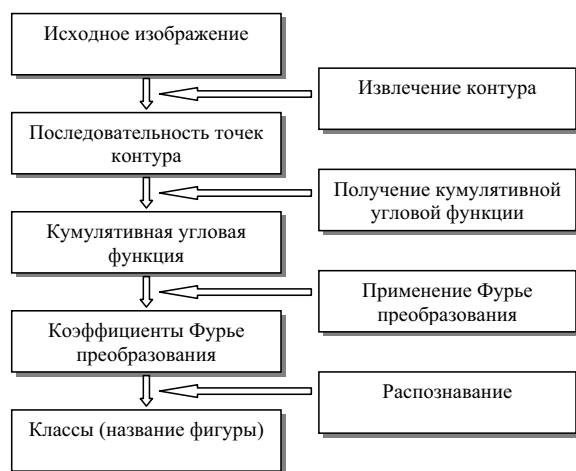


Рис. 1. Общая схема алгоритма

Для выделения внешних граничных точек используется алгоритм «соседние точки Мора» (*Moré's neighbors*) [5]. Результат выделения границы объекта алгоритмом Мора показан на рис. 3.

Выделенный контур сохраняется в виде массива точек  $P_n=(x_n, y_n)$ ,  $n=[1, N]$ , где  $N$  – количество граничных точек. Точки упорядочены по часовой стрелке. В каждой точке подсчитывается угол наклона касательной линии к горизонтальной оси (угловая функция). Угловая функция меняется в диапазоне  $[0, 2\pi)$ . Таким образом, она прерывна (резкий переход из  $2\pi$  в 0) и не может служить ис-

ходной функцией для Фурье-преобразования. Для устранения этой проблемы используется кумулятивная угловая функция. Однако кумулятивная функция не лишена недостатков: она прерывна в последней точке контура, и ее значения зависят от длины контура. Чтобы применять Фурье-преобразование кумулятивная функция должна быть нормирована [3]:

$$\phi^*(t) = \phi\left(\frac{L}{2\pi}t\right) + t.$$

Здесь  $\phi$  – кумулятивная функция;  $\phi^*$  – нормированная функция;  $L$  – длина контура.

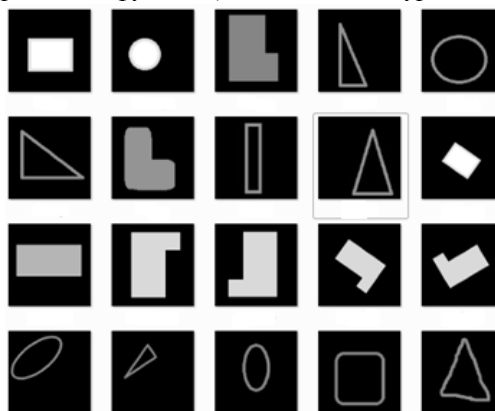


Рис. 2. База данных для обучения нейронной сети

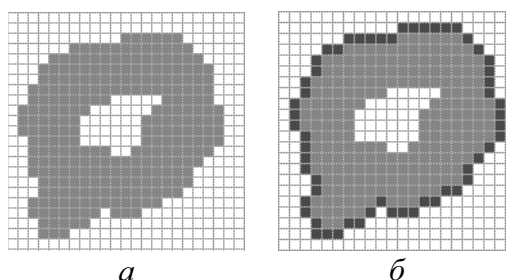


Рис. 3. Выделение контура алгоритмом Мора: а) исходное изображение; б) объект с выделенным контуром

На рис. 4, б–г, показаны касательная функция, кумулятивная угловая функция и нормированная кумулятивная угловая функция для окружности (рис. 4, д–ж).

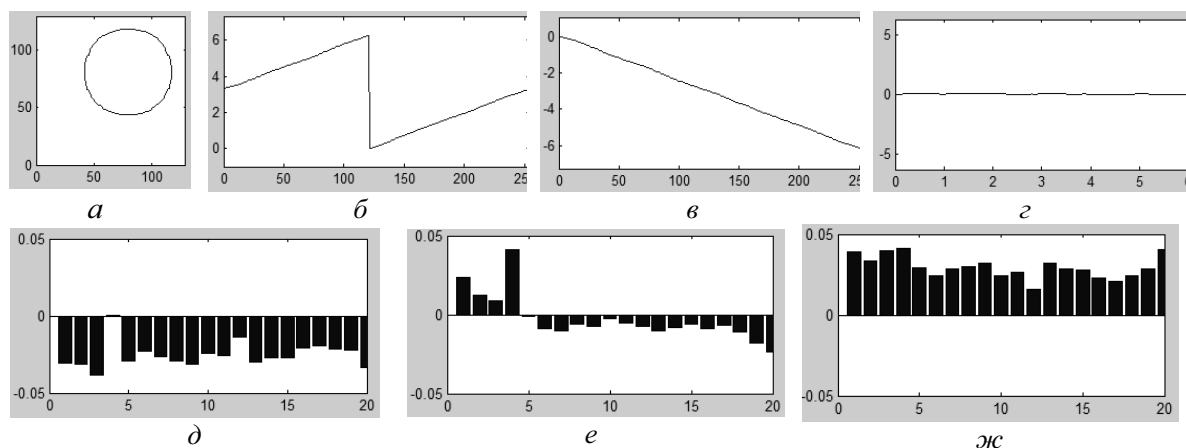
При применении Фурье-преобразования к нормированной кумулятивной угловой функции получается (рис. 4, д–ж):

$$a_k = \frac{1}{\pi} \int_0^{2\pi} \phi^*(t) \cos(kt) dt,$$

$$b_k = \frac{1}{\pi} \int_0^{2\pi} \phi^*(t) \sin(kt) dt,$$

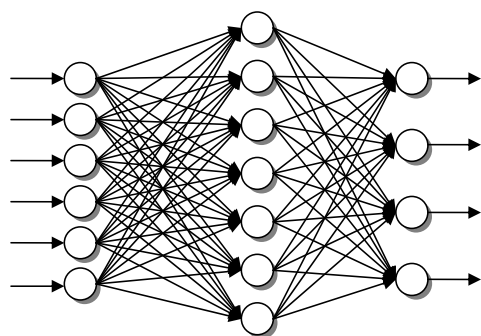
$$c_k = \sqrt{a_k^2 + b_k^2}.$$

Полученные таким образом Фурье-дескрипторы инвариантны к перемещению, масштабированию и вращению и будут использованы как входные данные для нейронной сети. Количество коэффициентов Фурье-преобразования для нейрон-



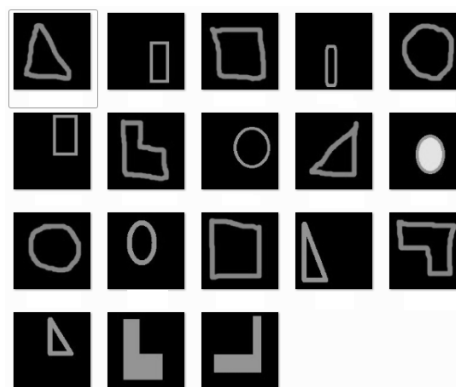
**Рис. 4.** Результаты преобразования для окружности: а) исходный контур; б) касательная функция; в) кумулятивная угловая функция; г) нормированная кумулятивная угловая функция; д) реальная часть Фурье-преобразования; е) мнимая часть Фурье-преобразования; ж) Фурье-дескриптор

ной сети будет зависеть от «сложности» фигуры. Эксперимент показывает, что 15–20 коэффициентов достаточно для распознавания несложных тригонометрических фигур. В этой работе используются 20 коэффициентов (дескрипторов).



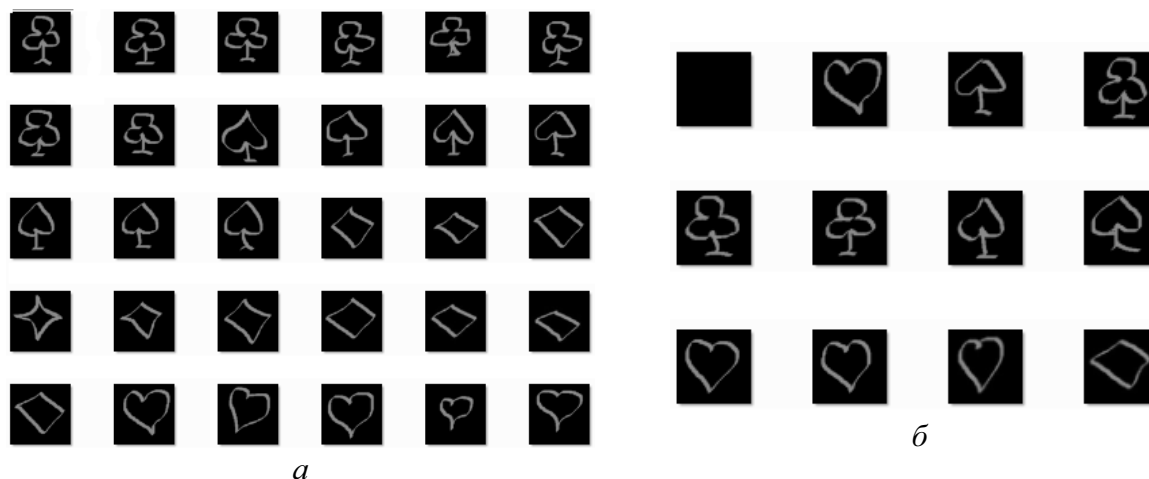
Входной слой – 20 нейронов      Скрытый слой – 50 нейронов      Выходной слой – 4 нейрона

**Рис. 5.** Схема нейронной сети



**Рис. 6.** Тестовые изображения

Для распознавания фигур применяется традиционная многослойная нейронная сеть с обратным распространением ошибки, структура которой показана на рис. 5. В качестве функции активации используется обычная биполярная сигмоидальная функция. Для повышения скорости сходимости сети применяются модификации: *Nguyen-Widrow*



**Рис. 7.** Распознавание более сложных фигур. Данные для: а) обучения; б) тестирования

инициация, моментум, групповое обновление [6]. Эксперимент показывает, что 40–60 нейронов в скрытом слое дают лучший результат в соотношении «времени обучения – сходимости». В данной работе используются 50 скрытых нейронов.

#### Обсуждение результатов

Программа была реализована на языке C# 2008. Программа предоставляет возможность формирования базы данных для обучения, создания и обучения нейронной сети (многослойный перцептрон), а также имеет отдельный интерфейс для проверки и тестирования работоспособности.

На этапе обучения сеть сходится после 10000 эпох со среднеквадратической ошибкой 0,001. Программа была протестирована 50 раз с тестовой базой данных, состоящей из 18 изображений (рис. 6). Частота появления ошибок составляет 0,1 %.

#### СПИСОК ЛИТЕРАТУРЫ

1. Folders A., Samet H. Content-based Image Retrieval Using Fourier Descriptors on a Logo Database // Proc. of the 16<sup>th</sup> Intern. Conf. on Pattern Recognition. – Quebec, 2002. – V. 3. – P. 521–524.
2. Zhang D., Lu G. Review of shape representation and description techniques // Pattern Recognition. – Oxford: Elsevier, 2004. – V. 37. – P. 1–19.
3. Nixon M., Aguado A. Feature Extraction & Image Processing. – Oxford: Elsevier, 2008. – 406 p.
4. Pattern Recognition Techniques, Technology and Applications / Ed. Peng-Yeng Yin. – Croatia: InTech, 2008. – 626 p.

Для проверки работоспособности алгоритма были созданы другие базы данных тренировки и тестирования (рис. 7). Полученные результаты показали, что алгоритм позволяет распознавать достаточно сложные фигуры, состоящие из простых элементов (круг, эллипс, треугольники...) с высокой точностью: для данных на рис. 7 после 30 тестов были 2 ошибки (частота ошибок – 0,15 %).

#### Выводы

1. Создана программа для распознавания фигур на основе анализа контура с применением Фурье-дескрипторов и нейронной сети.
2. Показано, что Фурье-дескрипторы и нейронные сети являются эффективным механизмом для решения задачи распознавания объектов.
3. Разработанная программа способна распознавать сложные фигуры с высокой точностью.
5. Ghuneim A.G. Moore-Neighbor Tracing // Contour Tracing, 2010. URL: [http://www.imageprocessingplace.com/downloads\\_V3/root\\_downloads/tutorials/contour\\_tracing\\_Abeer\\_Ghuneim/moore.html](http://www.imageprocessingplace.com/downloads_V3/root_downloads/tutorials/contour_tracing_Abeer_Ghuneim/moore.html) (дата обращения: 20.07.2010).
6. Fausett L.V. Fundamentals of Neural Networks – Architectures, Algorithms, and Applications. – Upper Saddle River: Prentice Hall, 1993. – 461 p.

Поступила 15.09.2010 г.

УДК 519.688:53.083.98

## АЛГОРИТМ СОВМЕСТНОЙ ФИЛЬТРАЦИИ РЯДОВ КЛЕТОЧНЫХ СТРУКТУР ГОДИЧНЫХ КОЛЕЦ ДЕРЕВЬЕВ

Ю.В. Волков, В.А. Тартаковский

Институт мониторинга климатических и экологических систем СО РАН, г. Томск  
E-mail: tv@imces.ru

*На примере рядов ежегодного прироста трахеид четырех климатипов сосны рассмотрен алгоритм совместной фильтрации, приемлемый для повышения качества восстановления биоиндикационной информации.*

#### Ключевые слова:

*Биоиндикация, годовые кольца, сезонный рост, математическое моделирование, анализ сигналов.*

#### Key words:

*Bioindication, tree-ring, vegetative growth, mathematical simulation, signal analysis.*

#### Введение

Годичные кольца, сформированные в стволе дерева рядами трахеид, являются биологическими индикаторами климатоэкологического состояния окружающей среды. Знание о биоиндикационных свойствах годовых колец может быть расширено и дополнено в результате изучения процессов роста дерева на клеточном уровне.

На поперечном сечении ствола дерева годовые слои отображаются в виде ряда клеток определенного количества (рис. 1), в которых каждая отдельная клетка формировалась в течение некоторого промежутка времени в пределах вегетационного периода под воздействием внешних и внутренних факторов, оказывавших определяющее воздействие на форму и размер клетки, а также на толщину клеточной стенки [1].