

С.М. Мишустин

Донецкий национальный технический университет, г. Донецк
кафедра автоматизированных систем управления

***ИССЛЕДОВАНИЕ АЛГОРИТМА ВFOА В РАМКАХ ЗАДАЧИ
РАСПОЗНАВАНИЯ ПРОСТЫХ ФИГУР НА ИЗОБРАЖЕНИИ***

Аннотация

Мишустин С.М. Исследование алгоритма ВFOА в рамках задачи распознавания простых фигур на изображении. *Выполнен обзор методов, которые могут быть применены для задачи распознавания простых фигур. Рассмотрена работа алгоритма ВFOА на примере распознавания окружности. Рассмотрена модификация алгоритма ВFOА для распознавания нескольких простых фигур на одном изображении. Предложена альтернативная модификация для распознавания нескольких простых фигур на одном изображении.*

Ключевые слова: *методы распознавания, распознавание простых фигур, алгоритм ВFOА.*

Постановка проблемы. Распознавание простых геометрических фигур на изображениях очень распространено в ряде задач, потому что такие фигуры часто присутствуют в среде, созданной человеком. Для этой задачи в последнее время открылась еще одна область – мобильные сенсорные устройства. В некоторых приложениях и играх требуется распознавать примитивные фигуры, считанные с экрана в результате проведения по нему манипулятором.

Так как скорость выполнения алгоритма распознавания крайне важна, то для его реализации следует выполнить следующие этапы:

- произвести обзор существующих методов поиска простых фигур;
- провести анализ найденных методов;
- выбрать наиболее подходящий метод и реализовать его.

Цель статьи – провести обзор методов, подходящих для решения задачи распознавания простых фигур и разобрать наиболее подходящий из них. Предложить улучшенную модификацию метода для распознавания нескольких простых фигур на одном изображении.

Обзор существующих методов распознавания. Одним из самых распространенных среди детерминированных методов является метод

преобразований Хафа (*HT*), который зачастую требуют объемных вычислений и затрат памяти. Чтобы преодолеть эти ограничения, исследователи предложили новые подходы к *HT*, например, вероятностный *HT*, рандомизированный *HT* и нечеткий *HT*. Эффективность использования преобразований Хафа резко падает при увеличении размерности фазового пространства. Преобразование Хафа можно применить для любой фигуры, форма которой полностью определяется некоторым набором параметров, например прямоугольника, треугольника, и т.д [1].

Как альтернатива основанным на *HT* методам, задача распознавания формы решалась методами стохастического поиска. Стохастические алгоритмы, подобно генетическим алгоритмам (*GA*), показывают хорошие результаты поиска оптимальных решений из-за обладания большим количеством неявного параллелизма.

В 2002 году был предложен мощный алгоритм оптимизации, теперь известный как алгоритм оптимизации, основанный на передвижении бактерий (*BFOA*). Он представляет собой многоагентную систему. *BFOA* имитирует отдельное и сгруппированное поведение бактерий, живущих в кишечнике большинства млекопитающих. С помощью этого алгоритма существует возможность обнаруживать формы, если адаптировать его для этой задачи.

Для задачи распознавания простых фигур выбран алгоритм *BFOA*, так как для нахождения им различных фигур достаточно заменить функцию определения оптимальности особи и в зависимости от фигуры будет изменено пространство поиска.

Описание работы алгоритма *BFOA* для распознавания окружности. Каждый предполагаемый круг представлен позицией бактерии, где первые два компонента вектора, x и y – координаты центра этой окружности, а третий компонент r – радиус.

Пусть (x_k, y_k, r_k) будет k -м тестовым кругом популяции, где $k = 1, \dots, S$, где S – это размер популяции, т.е. обозначает общее количество тестовых кругов. В фазе инициализации случайные значения в подходящем диапазоне присвоены каждой из трех координат векторов. Давайте рассматривать $2N$ равномерно распределенных демонстрационных точек на круге. На рис. 1 показан случай, когда $N=4$.

Предположим, что искомая окружность располагается на некотором изображении. Функция $P(x,y)$ равна 1, если пиксель (x,y) является граничным, 0 – иначе. Пусть A будет граничной матрицей. Целевую функцию обозначим как J . Пусть для (x_0, y_0, r_0) тестового круга ее значение будет $J_0 = J(A, x_0, y_0, r_0)$.

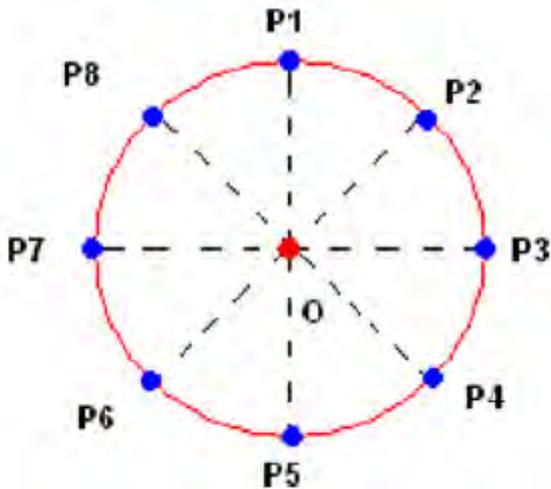


Рисунок 1 – Окружность с демонстрационными точками, изображенными синим

Для определения является ли кандидат кругом можно рассмотреть множество кругов с центром в (x_0, y_0) и радиусом, варьирующимся в диапазоне от $r_0 - \delta$ до $r_0 + \delta$. Назовем это множество тестовой полосой. Значение δ можно взять как процент от радиуса, предлагается $\delta = r_0/8$. После этого N тестовых точек берутся для каждого круга в тестовом диапазоне. Тестовые точки расположены на круге на равном расстоянии друг от друга. Пусть i -я тестовая точка на тестовом круге имеет радиус $r_0 + j$ ($-\delta \leq j \leq \delta$) и обозначается (x_i^j, y_i^j) .

$$x_i^j = (r_0 + j) \cos \frac{2\pi}{N} i, \quad (1)$$

$$y_i^j = (r_0 + j) \sin \frac{2\pi}{N} i, \quad (2)$$

где $i = 1, 2, \dots, N; j = -\delta, -\delta + 1, \dots, 0, \dots, \delta - 1, \delta$.

Чтобы измерить степень принадлежности тестовой точки к окружности центрального круга определим функцию расстояния μ . Значение функции расстояния μ_i^j для тестовой точки (x_i^j, y_i^j) определяется как:

$$\mu_i^j = P(x_i^j, y_i^j) \exp\left(-\frac{1}{2\sigma^2} (r + j + r_0)^2\right) \quad (3)$$

Значение μ может изменяться в диапазоне от 0 до 1. Когда $\mu_i^j = 1$, это означает, что выбранная точка граничная и она находится на центральном круге. Если (x_i^j, y_i^j) не граничная точка то $\mu_i^j = 0$.

Целевая функция, соответствующая (x_0, y_0, r_0) для граничной матрицы A определяется по следующей формуле

$$J(A, x_0, y_0, r_0) = 1 - \frac{1}{(2\delta+1)^N} \sum_{i=1}^N \sum_{j=-\delta}^{\delta} \mu_i^j \quad (4)$$

Целевая функция взвешивает каждую граничную точку согласно радиусу круга кандидата [3].

Таким образом, становится возможным с помощью алгоритма *BFOA* распознать округлость на изображении. Аналогичным образом можно подобрать параметры, которые идентифицируют любую другую фигуру, определить соответствующую этой фигуре целевую функцию и приведенный алгоритм можно будет использовать для поиска этой фигуры на изображении.

Существующая модификация для нахождения нескольких фигур на одном изображении. Предложенный алгоритм с некоторыми модификациями может обнаруживать множество фигур на одном изображении. Этого можно достичь, если задать верхний предел числа фигур. Метод выполняется на исходной граничной карте. Когда фигура найдена – она маскируется на граничной карте. Затем опять выполняем поиск. Процедура повторяется до тех пор, пока заданное количество фигур не будет найдено. В этом подходе есть возможность ложного обнаружения в случае, если фактическое число фигур меньше, чем определенное максимальное количество [3].

Альтернативная модификация для распознавания нескольких фигур на одном изображении. Предлагается альтернативный вариант нахождения нескольких фигур на одном изображении.

В качестве критерия останова алгоритма предлагается некоторое количество итераций алгоритма без улучшения суммарной целевой функции всех особей. Количество таких итераций необходимо определить экспериментально.

Размер популяции является одним из ключевых факторов данной модификации. Недостаточный размер популяции может привести к тому, что некоторые фигуры могут быть не найдены. С другой стороны, большой размер популяции сильно скажется на производительности алгоритма и скорости обнаружения фигур. Размер популяции должен быть определен экспериментально.

После отработки алгоритма необходимо применить любой из алгоритмов кластеризации для неизвестного количества кластеров к особям. Каждая особь с наилучшим значением целевой функции в кластере будет искомой фигурой, если значение ее целевой функции превышает некоторый порог, определенный экспериментально.

Сравнительная характеристика существующей и альтернативной модификаций. Существующий подход имеет следующие проблемы:

- скорость поиска;
- необходимость знать количество искомых фигур заранее;
- возможность ложного обнаружения.

Скорость поиска при работе альтернативной модификации выше, так как в существующей модификации алгоритм выполняется многократно, где количество итераций поиска равняется количеству искомых фигур. В альтернативной модификации выполняется только одна итерация поиска.

Альтернативная модификация решает проблему нахождения ложных фигур благодаря заданию порога.

В альтернативной модификации нет необходимости заранее знать максимальное количество искомых фигур.

Выводы. Проведен обзор методов, которые могут быть применены для задачи распознавания простых фигур. Выбран метод распознавания для этой задачи. Обоснованы причины выбора. Рассмотрена работа алгоритма *BFOA* на примере распознавания окружности. Рассмотрена существующая модификация алгоритма *BFOA* для поиска нескольких фигур на изображении. Предложена альтернативная модификация. Проведена сравнительная характеристика существующей и альтернативной модификаций. Обнаружено, что модификация, разработанная в данной работе, является более производительной и решает ряд других проблем существующей модификации. В дальнейшем планируется исследование работы существующей и предложенной модификаций на различных изображениях для определения числовых характеристик скорости работы, а также планируется анализ различных целевых функций.

Список литературы

1. Преобразование Хафа (Hough transform) [Электронный ресурс]. – Режим доступа : <http://www.cgm.computergraphics.ru/content/view/36>.
2. Kim D.H., Abraham A., Cho J.H. A Hybrid Genetic Algorithm and Bacterial Foraging Approach for Global Optimization // Information Sciences. – 2007. – № 18 (177). – P. 3918-3937.
3. S. Dasgupta, S. Das, A. Biswas, A. Abraham Automatic circle detection on digital images with an adaptive bacterial foraging algorithm // Soft Comput. – 2010. – P. 1151–1164.