

РАСПОЗНАВАНИЕ ОБРАЗОВ МЕТОДОМ ПОТЕНЦИАЛЬНЫХ ФУНКЦИЙ

Во многих областях техники используются различные автоматы и устройства, более или менее удачно решающие задачу распознавания образов. Это, например, автомат для сортировки почтовых конвертов по индексу, зенитная ракета, захватывающая горячее сопло самолетного двигателя, но игнорирующая солнце, различные системы анализа спутниковых снимков, голосовой вызов мобильного и многое другое.

Алгоритм распознавания образов методом потенциальных функций относится к числу так называемых алгоритмов с обучением. Доказано, что он имеет глубокую аналогию с популярной в настоящее время нейросетевой технологией распознавания образов. Рассмотрим его в применении к задаче распознавания графических изображений.

Для простоты мы будем рассматривать только черно-белые изображения. Пусть рисунок состоит всего из двух пикселей. Тогда множество всех объектов, которое можно будет изобразить (универсальное множество), состоит из четырех объектов: $(0,0)$, $(0,1)$, $(1,0)$, $(1,1)$, где 1 — черный пиксель, 0 — белый.









| X_1 | X_2 | X_1 | X_2 |
|---|---|-------|-------|
|  |  | 0 | 0 |
|  |  | 1 | 0 |
|  |  | 0 | 1 |
|  |  | 1 | 1 |
| А) | | Б) | |

Рисунок 1

Все объекты универсального множества можно разместить в вершинах единичного квадрата. Таким образом, множеству фигур, изображенных на двухпиксельном поле, может быть сопоставлено множество точек в двумерном пространстве. Ребру этого квадрата будет соответствовать переход от одного изображения к другому. Для перехода от $(1,1)$ к $(0,0)$ нужно будет пройти два ребра, для перехода от $(0,1)$ к $(0,0)$ — одно. Отметим, что число ребер в нашем переходе — это количество несовпадающих пикселей двух изображений. Вывод: **расстояние от одного рисунка до другого равно числу несовпадающих пикселей в них.** Это расстояние называется *расстоянием по Хэммингу*.

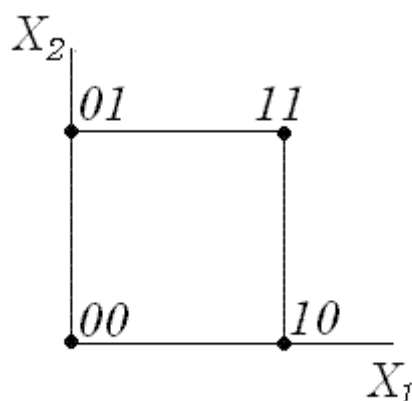


Рисунок 2

Теперь представим себе, что у нас рисунок состоит из трех пикселей. Коды изображений тогда будут состоять из трех значений, универсальное множество — из восьми элементов, которые мы разместим в вершинах единичного **куба**. Но принципиально ничего не изменится, и расстояние по Хэммингу вычисляется так же.

Если в задаче используется рисунок $50 \times 70 = 3500$ пикселей, то в этом случае код любого изображения состоит из 3500 значений, универсальное множество — из 2^{3500} элементов, которые мы будем размещать в вершинах единичного 3500-мерного куба. Основная идея заключается в том, что в этом многомерном кубе изображения, соответствующие какому-то определенному образу, лежат недалеко друг от друга. Эта идея получила название "**Гипотеза о компактности образов**".

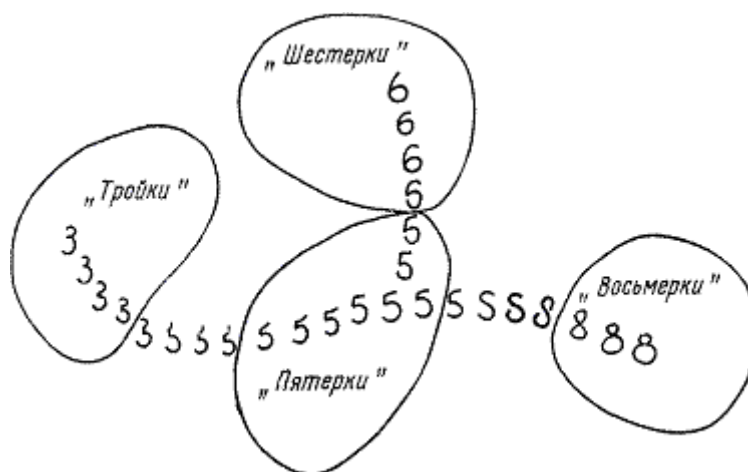


Рисунок 3

Теперь можно сформулировать задачу: нужно универсальное множество разбить на "куски", компактные множества, каждому из которых соответствует образ.

Реализация алгоритма несложная. Программе в процессе обучения сообщаются изображения (точки многомерного куба) и указания, к какому образу каждое изображение относится. При распознавании программа просто смотрит, в какую из известных компактных областей попало входное изображение. Для этого используется некоторая характеристика, которая показывает удаленность одного рисунка (точки в вершине многомерного куба) до группы таких же изображений. В качестве меры удаленности рисунка от группы рисунков используется **потенциал**.

Известно, что электрический заряд создает вокруг себя поле, одной из характеристик которого является потенциал. В любой точке он может быть вычислен по формуле

$$P = a \frac{q}{R^2}$$

Рисунок 4

где a — некоторый постоянный коэффициент, q — величина заряда, R — расстояние от данной точки до заряда. Если электрическое поле образовано двумя или более зарядами, то потенциал в данной точке равен сумме потенциалов каждого заряда. Аналогия очевидна — каждый рисунок, на котором программа обучалась, создает в пространстве универсального множества потенциал. После обучения программе дают распознать какой-либо рисунок (точку в вершине многомерного куба), программа вычисляет потенциал, создаваемый в этой точке всеми объектами образа "а", образа "б"... на которых программу учили и распознаваемый рисунок относится к образу, который создал наибольший потенциал.

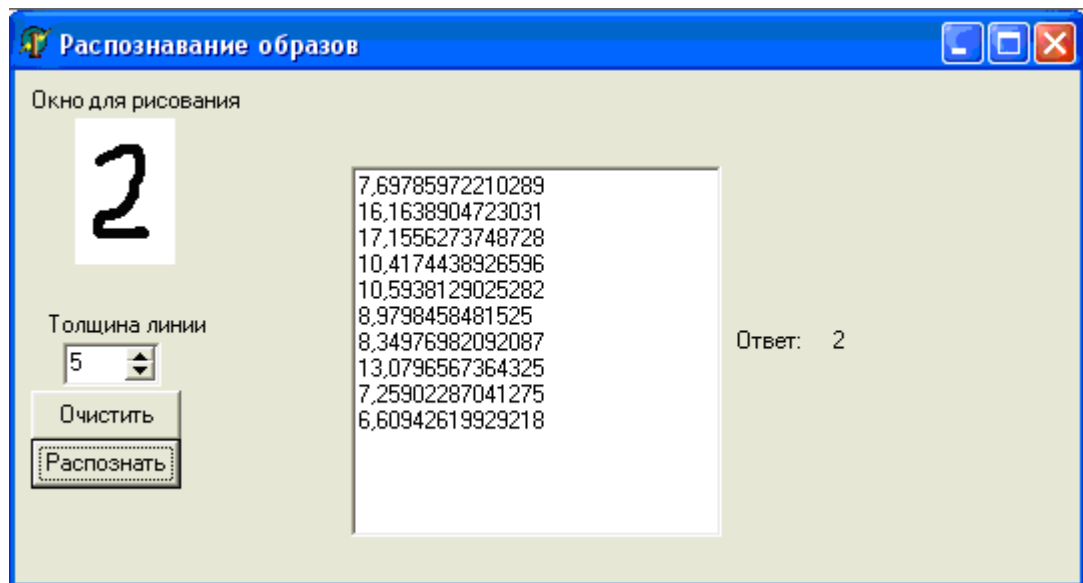
Таким образом, алгоритм представляет собой следующую последовательность операций.

1. Формирование множества распознаваемых образов (например, изображения цифр от 0 до 9) и обучающей выборки для каждого из образов (например, по 10 изображений каждой цифры, нарисованных различными шрифтами).
2. Формирование распознаваемого изображения (например, изображения какой-либо цифры, нарисованной от руки);
3. Вычисление расстояний от распознаваемого изображения до каждого из эталонных изображений обучающей выборки (расстояний по Хэммингу, то есть количества несовпадающих пикселей). Это даст матрицу R_{ij} , где i — номер образа, j — номер обучающего изображения для i -го образа.

4. Вычисление потенциалов, создаваемых обучающими изображениями, в точке распознаваемого изображения по формуле

$$\varphi(R) = \frac{1000000}{1 + R^2}$$

5. Суммирование потенциалов, создаваемых каждым из образов в точке распознаваемого изображения и выбор образа, создающего наибольший потенциал. Это и будет означать, что распознаваемое изображение принадлежит выбранному образу (например, что это – цифра 2).



ЗАДАНИЕ К ЛАБОРАТОРНОЙ РАБОТЕ

Разработать программу, реализующую следующие функции:

1. Определить три образа: круг, треугольник и квадрат
2. Определить обучающую выборку для каждого из образов, состоящую из 5 отличающихся друг от друга изображений соответствующей геометрической фигуры
3. Задать распознаваемое изображение
4. Определить расстояние по Хэммингу от распознаваемого изображения до изображений обучающей выборки
5. Определить потенциал, создаваемый каждым из образов в распознаваемой точке и выбрать образ с наибольшим потенциалом.