

# Распознавание образов

## Цель работы :

Получение практических навыков работы с простейшими алгоритмами распознавания объектов с качественными характеристиками.

## Задание :

На любом языке программирования реализовать программу, определяющую степень сходства образцов, поданных на вход. Использовать при этом все формулы сходства и проанализировать полученные результаты. Сформулировать признаки, по которым будет осуществляться сравнение (не менее 5).

## Задание с учетом варианта :

Номер варианта	Эталоны
6	локальная сеть, Интернет, интранет

## Общее описание алгоритма :

В большинстве случаев образы и отдельные изображения характеризуются с помощью количественных характеристик: геометрических размеров, веса, площади, объема и т. д. В этих случаях количественные изменения характеристик конкретного изображения обычно не сразу ведут к изменению образа, к которому относится распознаваемое изображение. Только достигнув определенных для каждого образа границ, количественные изменения вызывают качественный скачок - переход к другому образу. Образы и конкретные изображения могут характеризоваться не только количественными, но и качественными характеристиками (свойствами, признаками, атрибутами). Эти признаки не могут быть описаны (или обычно не описываются) количественно, например, цвет, вкус, ощущение, запах. Образы либо обладают какими-то качественными характеристиками, либо не обладают.

Существуют различные подходы к распознаванию изображений с качественными характеристиками. В данной лабораторной работе мы используем один из них, основанный на двоичном кодировании наличия или отсутствия какого-либо качественного признака. В рассматриваемом подходе конкретное изображение  $X_k$  некоторого образа с качественными характеристиками представляется в виде двоичного вектора

Для указанных в данном варианте эталонов выделим 5 качественных признаков таким образом, что бы соответствующие данным эталонам векторы значений качественных признаков не совпадали :

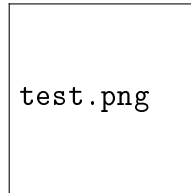
- Частность
- Глобальность
- Высокоуровневость
- Наличие администратора
- Наличие владельца (хозяин)

Для кодирования наличия этих признаков у эталонов используем двоичные векторы характеристик (в программной реализации это одномерные массивы элементов типа int с значениями 1 ("наличие") или 0 ("отсутствие"))

В результате имеем следующую таблицу :

Эталон	Частность	Глобальность	Высокоур.	Администратор	Владелец
Инtranет	Да	Нет	Да	Да	Да
Локальная сеть	Да	Нет	Нет	Да	Нет
Интернет	Нет	Да	Да	Нет	Нет

Реализация которой тривиальна и представлена на скриншоте :



Имея готовые характеристические векторы для заданных эталонов, мы должны заполнить аналогичным образом характеристический вектор для заданной пользователем сущности. Заполнив соответствующим образом этот вектор, мы можем произвести более тонкую классификацию объектов с качественными признаками получается при введении для каждой пары объектов  $X_j, X_n$ , для которых введено двоичное кодирование качественных признаков, переменных :  $a, b, g, h$ , формулы для нахождения которых представлены ниже :

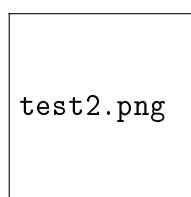
$$a = \sum_{k=1}^n x_{jk} \cdot x_{ik} \quad (1)$$

$$b = \sum_{k=1}^n (1 - x_{jk}) \cdot (1 - x_{ik}) \quad (2)$$

$$g = \sum_{k=1}^n x_{jk} \cdot (1 - x_{ik}) \quad (3)$$

$$h = \sum_{k=1}^n (1 - x_{jk}) \cdot x_{ik} \quad (4)$$

Пример реализации вычисления одного из таких коэффициентов представлен на скриншоте :



Из анализа переменных  $a, b, g, h$  следует, что, чем больше сходство между объектами  $X_j$ , и  $X_i$ , тем больше должна быть переменная  $a$ , т.е. мера близости объектов или функция сходства должна быть возрастающей функцией от  $a$ , функция сходства должна быть симметричной относительно переменных  $g$  и  $h$ . Относительно переменной  $b$  однозначный вывод сделать не удастся, поскольку, с одной стороны, отсутствие одинаковых признаков у объектов может свидетельствовать об их сходстве, однако, с другой стороны, если у объектов общим является только отсутствие одинаковых признаков, то они не могут относиться к одному классу.

Полученные значения переменных  $a, b, g, h$  используются в 7 наиболее часто встречаемых функциях сходства :

- Функция сходства Рассела и Рао
- Функция сходства Жокера и Нидмена

- *Функция сходства Дайса*
- *Функция сходства Сокаля и Снифа*
- *Функция сходства Сокаля и Мишнера*
- *Функция сходства Кульжсинского*
- *Функция сходства Юла*

Пользуясь вычисленными коэффициентами и используя функции сходства можно обозначить общий ход алгоритма программы :

- Для каждого характеристического вектора заданных эталонов :
  - Вычисляем переменные  $a, b, g, h$
  - Подставляем полученные переменные в каждую из семи функций сходства
  - Вычисляем максимум из полученных значений функций сходства и сохраняем его в какой-нибудь структуре данных (обрабатывая особые случаи)
- Находим максимум из максимумов полученных функций сходства для каждой итерации сравнения с каждым эталоном и получаем эталон, на который наиболее похожа сущность, определенная пользователем.

### *Анализ функций сходства :*

#### **1. Функция сходства Рассела и Рао**

Аналитический вид :

$$S(x_i, x_j) = \frac{a}{a + b + g + h} = \frac{a}{n} \quad (5)$$

Пределы функции :

$$\begin{aligned} \lim_{a \rightarrow \infty} \frac{a}{a+b+g+h} &= 1 \\ \lim_{b \rightarrow \infty} \frac{a}{a+b+g+h} &= 0 \\ \lim_{g \rightarrow \infty} \frac{a}{a+b+g+h} &= 0 \\ \lim_{h \rightarrow \infty} \frac{a}{a+b+g+h} &= 0 \end{aligned}$$

Поскольку переменные  $a, b, g, h$  не могут одновременно быть равными нулю, то функция сходства Рассела и Рао не возвращает деления на 0. Функция возвращает 0 если  $a = 0$ . Если  $a = b = g = h = 1$  то функция вернет свое максимальное значение, равное 1. При использовании функции сходства Рассела и Рао степень распознавания будет тем выше, чем больше одинаковых признаков у сравниваемого объекта и объекта-образца.

#### **2. Функция сходства Жокара и Нидмена**

Аналитический вид :

$$S(x_i, x_j) = \frac{a}{n - b} \quad (6)$$

Пределы функции :

$$\begin{aligned} \lim_{a \rightarrow \infty} \frac{a}{a+g+h} &= 1 \\ \lim_{g \rightarrow \infty} \frac{a}{a+g+h} &= 0 \\ \lim_{h \rightarrow \infty} \frac{a}{a+g+h} &= 0 \end{aligned}$$

В данном случае функция сходства Жокара и Нидмена имеет точки разрыва, то есть случаи, когда в знаменателе получается 0 - когда векторы сравниваемого объекта и объекта-образца представляют собой нулевые вектора. В такой ситуации мы получим неопределенность  $\frac{0}{0}$ , что дает нам право предполагать, что, вообще говоря, в этом случае функция сходства может принимать любое значение, т.к. о природе сравниваемого объекта и о природе объекта-образца ничего не известно, потому что они оба не обладают ни одним признаком, по которым производится сравнение.

### 3. Функция сходства Дайса

Аналитический вид :

$$S(x_i, x_j) = \frac{a}{2 \cdot a + g + h} \quad (7)$$

Пределы функции :

$$\lim_{a \rightarrow \infty} \frac{a}{2 \cdot a + g + h} = 0.5$$

$$\lim_{g \rightarrow \infty} \frac{a}{2 \cdot a + g + h} = 0$$

$$\lim_{h \rightarrow \infty} \frac{a}{2 \cdot a + g + h} = 0$$

Аналогично функции Жокара-Нидмена, но с максимумом 0.5.

### 4. Функция сходства Сокаля и Снифа

Аналитический вид :

$$S(x_i, x_j) = \frac{a}{a + 2 \cdot (g + h)} \quad (8)$$

Пределы функции :

$$\lim_{a \rightarrow \infty} \frac{a}{a + 2 \cdot (g + h)} = 1$$

$$\lim_{g \rightarrow \infty} \frac{a}{a + 2 \cdot (g + h)} = 0$$

$$\lim_{h \rightarrow \infty} \frac{a}{a + 2 \cdot (g + h)} = 0$$

Аналогично функции Жокара-Нидмена, однако различие в наличии определенного признака у сравниваемого объекта и объекта-образца снижает значение функции сходства.

### 5. Функция сходства Сокаля-Мишнера

Аналитический вид :

$$S(x_i, x_j) = \frac{a + b}{n} \quad (9)$$

Пределы функции :

$$\lim_{a \rightarrow \infty} \frac{a + b}{n} = 1$$

$$\lim_{b \rightarrow \infty} \frac{a + b}{n} = 1$$

$$\lim_{g \rightarrow \infty} \frac{a + b}{n} = 0$$

$$\lim_{h \rightarrow \infty} \frac{a + b}{n} = 0$$

Минимальное значение функции равно 0, а максимальное - 1. Всякое расхождение в наличии признаков ведет к уменьшению значения этой функции сходства.

### 6. Функция сходства Кульжинского

Аналитический вид :

$$S(x_i, x_j) = \frac{a}{g + h} \quad (10)$$

Пределы функции :

$$\lim_{g \rightarrow \infty} \frac{a}{g+h} = 0$$

$$\lim_{h \rightarrow \infty} \frac{a}{g+h} = 0$$

Аналогично функции Жокара-Нидмена с добавлением еще одного критичного случая, когда имеется хотя бы один признак, который присутствует у сравниваемого объекта и объекта-образца, и нет признаков, которые есть у одного объекта, и нет у другого. В таком случае получаем неопределенность  $\frac{a}{0}$ , что дает значение функции, равное  $\infty$ , т.е. как бы сравниваемый объект и объект-образец идентичны.

## 7. Функция сходства Юла

Аналитический вид :

$$S(x_i, x_j) = \frac{a \cdot b - g \cdot h}{a \cdot b + g \cdot h} \quad (11)$$

Пределы функции :

$$\lim_{a \rightarrow \infty} \frac{a \cdot b - g \cdot h}{a \cdot b + g \cdot h} = 1$$

$$\lim_{b \rightarrow \infty} \frac{a \cdot b - g \cdot h}{a \cdot b + g \cdot h} = 1$$

$$\lim_{g \rightarrow \infty} \frac{a \cdot b - g \cdot h}{a \cdot b + g \cdot h} = -1$$

$$\lim_{h \rightarrow \infty} \frac{a \cdot b - g \cdot h}{a \cdot b + g \cdot h} = -1$$

Значения функции лежат в отрезке  $[-1; 1]$ . При этом существуют точки разрыва, дающие неопределенность вида  $\frac{0}{0}$ . Это говорит о том, что должен быть хотя бы 1 признак, который есть у обоих объектов, 1 признак, которого нет у обоих объектов ( $ab \neq 0$ ), или 1 признак, который есть у одного объекта и которого нет у другого (и наоборот) ( $gh \neq 0$ ).

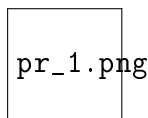
В противном случае функцию сходства Юла применять нельзя, т.к. она может принимать любое значение.

Из анализа вышеперечисленных семи функций сходства ясно, что особое внимание при программной реализации стоит уделять функции сходства Кульжинского и функции сходства Юла. В данном варианте при заданных характеристических векторах эталонов только функция сходства Кульжинского возвращала неопределенность вида  $\frac{n}{0}$  в случаях полного совпадения эталона и образца, а также если характеристический вектор образца содержал одни нули и единицы.

### Программная реализация :

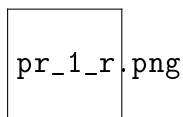
Данная лабораторная работа была написана на языке программирования C++ (с некоторыми расширениями). Для реализации GUI приложения был выбран кроссплатформенный фреймворк для разработки desktop-приложений QT (точнее - его модули QtCore и QtGui). Интерфейс приложения состоит из вкладочный вид и состоит из 2х вкладок : ввод данных и вкладка результатов.

Вкладка "Ввод данных" имеет следующий вид :



Эта вкладка содержит флажки, соответствующие наличию определенных признаков у исследуемого объекта; вджжет-таблицу, где отображаются соответствующие признаки у исследуемого объекта и предопределенных эталонов в виде значений типа Да/Нет. Переключение флажков ведет к изменению значений соответствующих признаков в таблице. Сравнение эталона с образцом происходит при нажатии соответствующей кнопки.

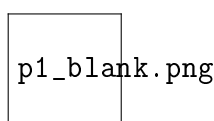
Вкладка "Результаты" имеет следующий вид :



Эта вкладка содержит текстовую метку где указано на какой эталон больше всего похож данный образец. При этом в текстовом поле подробно указаны результаты работы алгоритма на каждой итерации сравнения с каждым эталоном :

- значения переменных  $a, b, g, h$
- значения всех функций сходства на данном наборе переменных  $a, b, g, h$
- текущий максимум функций сходства
- глобальный максимум функций сходства для всех эталонов

В начале работы приложения эта вкладка пуста :

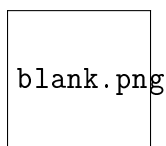


### ***Протокол работы :***

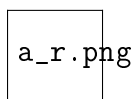
Для проверки корректности работы приложения проведем 4 тестовых прогона :

#### ***1. Граничные случаи :***

Как было выяснено при анализе функции сравнения Кульжинского, эта функция возвращает неопределенность вида  $\frac{n}{0}$  в случае если характеристический вектор образца содержит только 1 или 0. Проверим это:



В результате как уже было сказано функция Кульжинского вернет деление на ноль для всех эталонов :

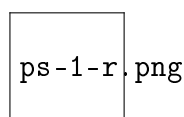


#### ***2. 1 прогон :***

Имеем следующий образец :

Частное	Глобальное	Высокоур.	Админ.	Контроль доступа
Да	Нет	Да	Да	Нет

Результат :

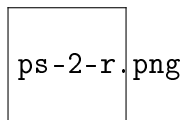


3. 2 прогон :

Имеем следующий образец :

Частное	Глобальное	Высокоур.	Админ.	Контроль доступа
Да	Нет	Нет	Да	Нет

Результат :

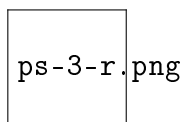


4. 3 прогон :

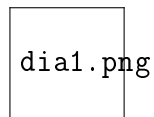
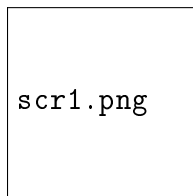
Имеем следующий образец :

Частное	Глобальное	Высокоур.	Админ.	Контроль доступа
Да	Да	Нет	Да	Нет

Результат :

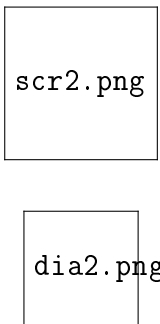


1. Процедура поиска максимального элемента в массиве (метод *getMax()* класса *MainWindow*) :





2. Процедура, повторяемая на каждой итерации алгоритма (метод *DisplayDiff()* класса *MainWindow*):



scr2.png

dia2.png

3. Процедура, реализующая основную логику приложения (обработчик кнопки на форме) :

diam.png