

Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики и фундаментальной информатики
Базовая кафедра вычислительных и информационных технологий

УТВЕРЖДАЮ

Заведующий кафедрой

_____ / В.В. Шайдуров

« ___ » _____ 2017 г.

БАКАЛАВРСКАЯ РАБОТА

Направление 02.03.01 Математика и компьютерные науки

**ИССЛЕДОВАНИЕ МЕТОДОВ РЕШЕНИЯ ЗАДАЧИ
РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ**

Научный руководитель
кандидат физико-математических наук,
доцент

_____ / И.В. Баранова

Выпускник

_____ / Д.А. Докучаев

Красноярск 2017

РЕФЕРАТ

Бакалаврская работа по теме «Исследование методов решения задачи распознавания изображений» содержит 49 страниц текста, 26 использованных источников.

РАСПОЗНАВАНИЕ ОБРАЗОВ, РАСПОЗНАВАНИЕ ИЗОБРАЖЕНИЙ, ПРЕЦЕДЕНТ, ОБРАЗ, НЕЙРОННАЯ СЕТЬ, МАШИНА ОПОРНЫХ ВЕКТОРОВ, ОБУЧАЮЩАЯ ВЫБОРКА.

Цель работы – исследование и программная реализация основных алгоритмов решения задачи распознавания изображений.

В результате исследования были изучены основные методы распознавания: метод опорных векторов, метод потенциальной функции и искусственные нейронные сети. Решен практический пример задачи распознавания изображений – задача распознавания на изображении пешеходов. Разработано программное обеспечение «Распознавание пешеходов», реализующее работу перечисленных методов распознавания по решению поставленной практической задачи. Проведены вычислительные эксперименты по сравнению вычислительной сложности, скорости сходимости и качеству обучения методов на подобранных обучающих выборках. Проведено сравнение результатов, полученных в результате работы каждого метода.

СОДЕРЖАНИЕ

Введение.....	3
1 Задача распознавания изображений.....	5
1.1 Понятие распознавания образов.....	5
1.2 Постановка задачи распознавания изображений.....	6
1.3 Сведения о получения векторов признаков.....	7
2 Необходимые теоретические сведения и положения о сходимости методов .	11
2.1 Необходимые определения из теории алгоритмов и вычислительной математики.....	11
2.2 Некоторые положения о сходимости методов.....	12
3 Метод опорных векторов.....	13
3.1 Идея метода.....	13
3.2 Линейно разделимый случай задачи.....	14
3.3 Линейно неразделимый случай задачи.....	17
3.4 Метод сопряженных градиентов.....	19
3.5 Метод наискорейшего спуска.....	21
4 Решение задачи с помощью нейронных сетей.....	23
4.1 Сети прямого распространения.....	23
4.2 Метод обратного распространения ошибки.....	25
5 Метод потенциальной функции.....	30
5.1 Физический смысл.....	30
5.2 Описание метода.....	30
6 Решение практической задачи распознавания изображений.....	33
6.1 Вычислительный эксперимент.....	33
6.2 Сравнение представленных алгоритмов.....	36
7 Описание программного модуля.....	39
7.1 Программный модуль вектора опорных векторов.....	40
7.2 Программный модуль алгоритма нейронных сетей.....	42
7.3. Программный модуль метода потенциальной функции.....	43
Заключение.....	45
Список использованных источников.....	46

ВВЕДЕНИЕ

Во многих сферах деятельности человека требуется классифицировать объекты на несколько категорий с определенными характеристиками. Задачи подобного вида относятся к задаче распознавания образов, являющейся одной из наиболее практически важных и сложных задач анализа данных.

Распознавание образов представляет собой научное направление, возникшее и развивающееся на базе теоретической информатики, прикладной статистики, искусственного интеллекта и смежных дисциплин. Его назначением является разработка принципов и методов классификации объектов различной природы. Методы распознавания образов имеют широкий спектр применений практически во всех областях человеческой деятельности, связанных с изучением объектов и процессов: медицине, биологии, химии, маркетинге, психологии, социологии, менеджменте и других.

В настоящее время существует множество частных случаев задачи распознавания образов: распознавание речи, распознавание изображений, распознавание символов, классификация документов, поиск знаний в базах данных и другие. В данной работе решается задача распознавания изображений.

Целью работы является исследование и программная реализация основных алгоритмов решения задачи распознавания изображений.

В первой главе работы приводятся основные понятия и постановка задачи распознавания изображений.

Вместе с развитием информационных систем и ресурсов, постоянно появляются новые методы классификации и машинного обучения, которые решают задачу распознавания образов. В бакалаврской работе изучаются три наиболее распространённых метода бинарной классификации: метод опорных векторов, метод потенциальной функции и искусственные нейронные сети.

В третьей, четвёртой и пятой главах работы дается подробное описание перечисленных алгоритмов, и указываются особенности их работы.

В работе решается практический пример задачи распознавания изображений – задачи распознавания на изображении пешеходов.

Разработано программное обеспечение «Распознавание пешеходов», реализующее метод опорных векторов, нейронную сеть с обратным распространением ошибки и метод потенциальной функции. Для каждого метода предусматривается отдельный элемент управления. Данное программное средство решает поставленную практическую задачу.

Значительная часть работы посвящена исследованию вышеперечисленных методов распознавания образов. Выполняется серия вычислительных экспериментов, позволяющих оценить работу алгоритмов на ряде тестовых обучающих выборок. В ходе вычислительного эксперимента методы сравниваются по вычислительной сложности, скорости сходимости и качеству обучения.

1 Задача распознавания изображений

1.1 Понятие распознавания образов

В нынешних реалиях сложно представить деятельность человека, в которой не используется распознавание образов. Существует постоянная потребность решать подобные задачи с помощью информационных ресурсов, используя различные методы машинного обучения.

Определение 1.1 *Распознавание образов* – это наука о принципах и методах классификации объектов различной природы.

Распознавание образов является востребованным и успешно развивающимся направлением в современной теоретической информатике и анализе данных. Необходимость в решении задач распознавания возникает в самых разных областях человеческой деятельности, связанных с изучением объектов и процессов: медицине, биологии, химии, маркетинге, психологии, социологии, менеджменте и других.

Определение 1.2 *Машинное обучение (теория обучения машин)* – научное направление, находящееся на стыке прикладной статистики, численных методов, методов оптимизации, дискретного анализа, теории вероятностей (и ряда других дисциплин) и изучающее методы искусственного интеллекта, т.е. алгоритмы, которые не напрямую решают задачу, а обучаются её решению в процессе решения множества сходных задач.

Методы машинного обучения делятся на два основных типа: обучение с учителем и обучение без учителя. В этой работе все далее рассматриваемые методы будут являться методами обучения с учителем. Такого рода методы требуют наличия обучающей выборки, состоящей из прецедентов.

Определение 1.3 *Обучающая выборка* – множество объектов, для которых известно какому классу они принадлежат.

Определение 1.4 *Прецедент* – это образ, правильная классификация которого заранее известна.

Задача распознавания образов состоит в том, чтобы отнести новый распознаваемый объект к какому-либо классу.

1.2 Постановка задачи распознавания изображений

Пусть задано множество $X = \{x_1, \dots, x_N\}$, где $x_i, i = \overline{1, N}$ является образом объектом распознавания. Существует индикаторная функция, которая неизвестна [17].

$$g(x) : X \rightarrow M, \quad M = \{1, \dots, m\} \quad (1.1)$$

Определение 1.5 *Индикаторная функция* – это функция вида (1.1), разбивающая пространство образов X на m непересекающихся классов [17].

Пусть F – это пространство признаков, функция $f(x) : X \rightarrow F$ ставит в соответствие каждому объекту $x_i \in X$ вектор $f(x)$. Вектор $f(x)$ называется образом объекта x . В пространстве образов определены непересекающиеся множества точек $K_i \in F, i = \overline{1, m}$, соответствующие образам одного класса [17].

Определение 1.6 *Решающим правилом* называется функция вида $\overline{g(x)} : F \rightarrow M$, представляющая собой оценку для $g(x)$ на основании $f(x)$ [17].

Пусть $f_j = f(x_j), j = \overline{1, N}$ – доступная информация о функциях $g(x)$ и $f(x)$, но сами эти функции неизвестны. Тогда (g_j, f_j) – множество прецедентов [17].

Требуется построить такое решающее правило $\overline{g(x)}$, чтобы классификация объектов проводилась с минимальным числом ошибок.

Как было сказано выше, набор объектов описывается признаками, которые могут быть следующих образов:

- Бинарный (0 или 1),
- Номинальный (качественный) признак,
- Порядковый или количественный признак.

В данной работе используются качественные признаки для описания объектов.

Будем рассматривать частный случай задачи распознавания образов – распознавание изображений, т.е. в качестве множества объектов распознавания возьмем некоторое множество изображений. Так же условимся, что классификацию этих изображений будем проводить по 2 классам, следовательно, множество $M = \{0,1\}$.

1.3 Сведения о получения векторов признаков

Как уже было сказано в предыдущем разделе, все методы, решающие задачу распознавания (в том числе и задачу распознавания изображений) выполняют классификацию объектов на основании их характеристик (т.е. вектора значений характеристик). Для того чтобы было удобно использовать эти методы, проведем некоторые преобразования над изображениями. Следует оговорить, что в качестве изображений в данной работе используются *растровые изображения*.

Определение 1.7 *Растровое изображение* – изображение, представляющее собой сетку пикселей (цветных точек) на мониторе, бумаге и других отображающих устройствах.

Важными характеристиками изображения являются: размер изображения в пикселях (в виде количества пикселей по ширине и по высоте) и количество используемых цветов или глубина цвета.

В качестве таких преобразований можно использовать операцию получения градиента изображения.

Определение 1.8 *Градиент* – направления максимального изменения яркости изображения $\nabla f = \left[\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y} \right]$.

Градиент – это вектор, прежде всего, который обладает длиной (силой края) и направлением. Для вычисления длины будем использовать евклидову

метрику $\|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$, а для вычисления направления воспользуемся

следующим выражением $\theta = \arctan \frac{\frac{\partial f}{\partial y}}{\frac{\partial f}{\partial x}}$.

Очевидно, что нет прямой возможности посчитать производные от изображения. Для этого вводится операция свертки, которая позволяет численно вычислить производную в каждом пикселе.

Операция свертки состоит в том, чтобы пересчитать значение некоторого пикселя с помощью наложения ядра. Пересчет производится посредством произведения нецентральных элементов ядра и пикселей изображения, на которые это ядро наложено, и дальнейшего суммирования этих произведений. Алгоритм применения такой операции проиллюстрирован на рисунке 1.1.

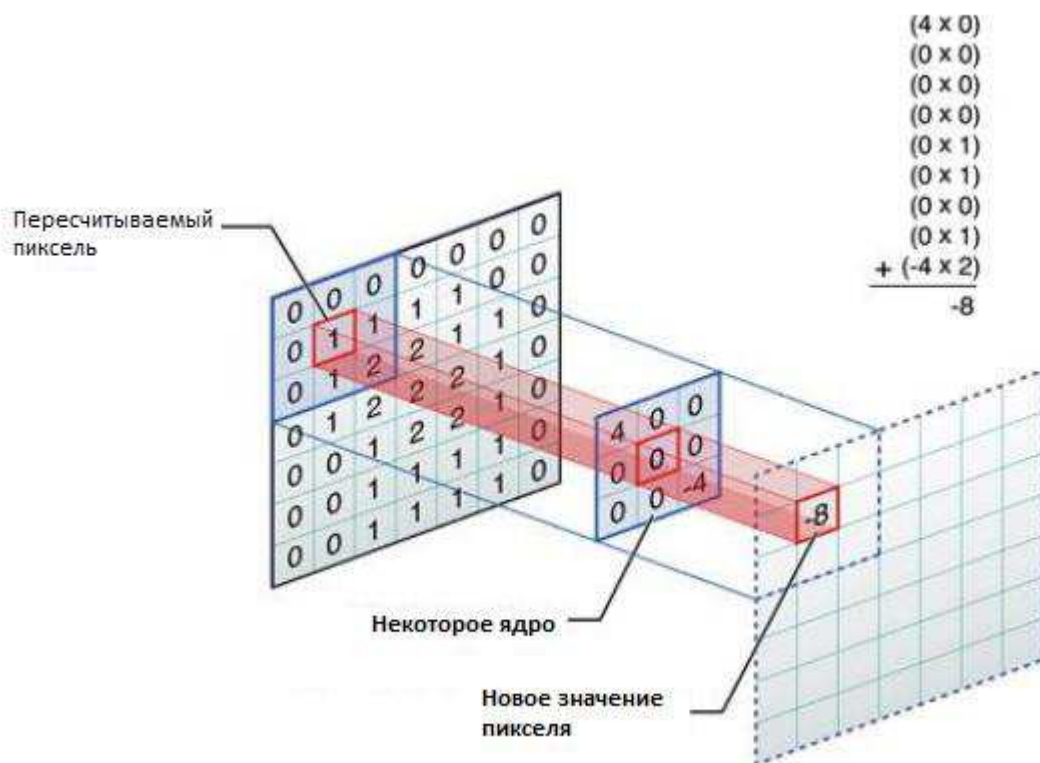


Рисунок 1.1 – Применение операции свертки

Для вычисления производной по x в качестве ядра используется оператор Собеля вида (1.2), а для вычисления производной в каждом пикселе изображения по y используется оператор Собеля вида (1.3).

$$\begin{matrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{matrix} \quad (1.2)$$

$$\begin{matrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{matrix} \quad (1.3)$$

Таким образом, мы можем получить производные в каждом пикселе изображения, вычислить градиент, длину градиента и его направление.

Изображения на входе могут быть совершенно разными по размеру и цветовой палитре. Как видно, в операции свертки используется одна матрица, но в палитре RGB подразумевается использование трех таких матриц.

Преобразуем изображение к разрешению $64*128$ пикселей, либо графическим редактором, либо какими-либо другими доступными средствами. Далее, используя три матрицы цветовой палитры RGB, преобразуем изображение к градации серого, получая, таким образом, одну матрицу. Переход в градацию серого осуществляется по формуле $0,299 * R + 0,587 * G + 0,114 * B$, которая применяется для каждого пикселя. После этого можно применять операцию свертки с приведенными выше операторами.

Разобьем изображение на ячейки $8*8$ пикселей. В таких ячейках для каждого пикселя есть градиент, его длина и направление. На основе этих данных составим гистограмму для каждой ячейки.

Гистограмма будет состоять из 8 элементов, которые будут представлять различные углы направления вектора градиента, т.е.:

- $[0, \frac{\pi}{4}]$ – элемент номер 1,

- $[\frac{\pi}{4}, \frac{\pi}{2}]$ – элемент номер 2,
- $[\frac{\pi}{2}, \frac{3\pi}{4}]$ – элемент номер 3,
- $[\frac{3\pi}{4}, \pi]$ – элемент номер 4,
- $[\pi, \frac{5\pi}{4}]$ – элемент номер 5,
- $[\frac{5\pi}{4}, \frac{3\pi}{2}]$ – элемент номер 6,
- $[\frac{3\pi}{2}, \frac{7\pi}{4}]$ – элемент номер 7,
- $[\frac{7\pi}{4}, 2\pi]$ – элемент номер 8.

Таким образом, анализируя 64 пикселя в ячейке, исследуем каждый пиксель отдельно. Находим направление градиента в этом пикселе, а затем – соответствующий элемент гистограммы для этого градиента. Как только такой элемент найден, прибавляем к нему единицу, умноженную на длину вектора градиента.

Для каждой ячейки получится отдельная гистограмма. Всего таких гистограмм будет 64, как и ячеек. Каждый элемент гистограммы будем считать признаком изображения. Для одного изображения, таким образом, получится 1024 признака.

2 Необходимые теоретические сведения и положения о сходимости методов

2.1 Необходимые определения из теории алгоритмов и вычислительной математики

Определение 2.1 *Аппроксимация* – научный метод, состоящий в замене одних объектов другими, в каком-то смысле близкими к исходным, но более простыми [16].

Пусть A – положительно определенный оператор в гильбертовом пространстве H , а b – элемент этого пространства.

Определение 2.2 Говорят, что метод *сходится*, если разность точного значения функции и вычисленного по абсолютной величине, стремится к нулю.

Определение 2.3 *Устойчивость* – непрерывная зависимость от входных данных.

Определение 2.4 *Квадратичным функционалом* называется функционал вида [22]

$$J(u) = (Au, u) - 2(u, b). \quad (2.1)$$

Квадратичный функционал имеет область определения, очевидно, совпадающую с областью определения оператора A , так что $D(J) = D(A)$.

Теорема 2.1 *Для того чтобы некоторый элемент $u_0 \in D(A)$ сообщал минимальное значение функционалу, необходимо и достаточно, чтобы этот элемент удовлетворял уравнению*

$$Au_0 = b, \quad (2.2)$$

причем такой элемент единственный [22].

Далее, в работе в качестве положительно определенного оператора A используется положительно определенная матрица размерности $n \times n$, а в качестве элемента b – вектор размерности n , элементы которых принадлежат множеству R .

2.2 Некоторые положения о сходимости методов

Существует множество различных методов распознавания изображений и машинного обучения. Выделяют детерминистские, вероятностные, логические и структурные методы. В данной работе рассматриваются исключительно детерминистские методы, такие как:

- Метод опорных векторов,
- Нейронная сеть и метод обратного распространения ошибки,
- Метод потенциальных функций.

Из определения (1.6) видно, что решающее правило – это некоторая функция, которая может быть, как линейной, так и нелинейной. Поэтому метод опорных векторов существует двух видов: линейный и нелинейный. Будем считать, что объект, представленный на изображении, является объектом достаточно сложной структуры, которая определяется, исходя из признаков изображения, 1024 переменными. Так как не существует возможности сказать точно, к какому случаю будет относиться задача распознавания изображений, то в данной работе будет использоваться нелинейный подход.

Такое положение делается из соображений, что метод будет сходиться лучше, предоставляя лучшее из решений.

Нейронные сети обладают достаточно хорошим качеством. Они могут приближать функции различной сложности:

- Нелинейные функции,
- Кусочно-заданные функции,
- Неоднозначные функции.

Использование данного метода целесообразно, с учетом сложности структуры объекта, т.е. пешехода.

Метод потенциальной функции всегда ищет решающее правило в виде нелинейной функции, поэтому данный метод был также включен в эту работу.

3 Метод опорных векторов

3.1 Идея метода

Пусть дано некоторое множество точек в пространстве. Требуется разделить данные точки на два класса по какому-либо признаку. Для этого мы ищем некоторую функцию – решающее правило. Все точки, которые лежат выше будут отнесены к первому классу, остальные ко второму.

Таких функций может быть бесконечное множество, причем они могут быть, как линейными, так и нелинейными. Как следует из определения задачи распознавания образов, нам нужно найти такое решающее правило, чтобы классификация проводилась с минимальным число ошибок. Такая функция будет единственной. Пример такой функции представлен на рисунке 3.1 прямой 2.

Решение такой задачи предложил В. Вапник в 1960-80 гг. Данный метод называется *методом опорных векторов*. Этот метод относится к методам машинного обучения с учителем и предназначен для нахождения оптимальных в некотором смысле решающих функций [15].

Решающие функции строятся с учетом граничных точек этих двух классов. Такие точки называют *опорными*. На рисунке 3.1 опорные точки расположены на прямых 1 и 3.

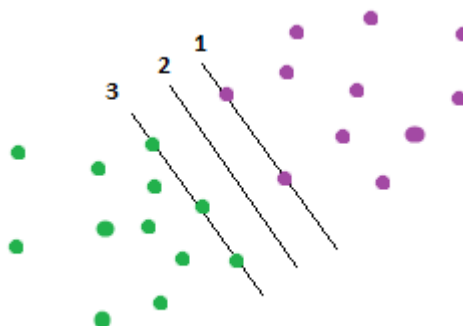


Рисунок 3.1 – Линейный случай метода опорных векторов

3.2 Линейно разделимый случай задачи

Рассмотрим задачу нахождения наилучшего в некотором смысле линейного разделения множества векторов на два класса. Пусть задано некоторое множество прецедентов (X, Y) , где $X = \{x_1, \dots, x_N\}$ – обучающая выборка, а $Y = \{y_1, \dots, y_N\}$ – множество меток двух классов ω_1 и ω_2 . Требуется по обучающей выборке построить линейную решающую функцию $f(x)$, которая удовлетворяла бы условию

$$f(x) > 0, \forall x_i \in \omega_1; f(x) < 0, \forall x_i \in \omega_2. \quad (3.1)$$

Без ограничения общности можно считать, что метки классов равны $y_i = \begin{cases} 1, x_i \in \omega_1 \\ -1, x_i \in \omega_2 \end{cases}$. Тогда представленную выше задачу можно переформулировать следующим образом:

требуется найти линейную решающую функцию $f(x)$, которая бы удовлетворяла условию

$$y_i f(x_i) > 0, \forall x_i \in X. \quad (3.2)$$

При умножении функции f на некоторое положительное число система неравенств (3.2) будет равносильна системе $y_i f(x_i) > 1, \forall x_i \in X$. Так как $f(x)$ – линейная функция, то система неравенств примет вид

$$y_i((w, x_i) + b) \geq 1, i = 1, \dots, N, \quad (3.3)$$

где w – вектор весовых коэффициентов, b – некоторое число.

Тогда разделяющей на два класса гиперплоскостью будет $(w, x) + b = 0$. Нетрудно заметить, что и все гиперплоскости вида $(w, x) + b = 0$, где $b \in (b - 1, b + 1)$ также будут разделяющими [15]. На рисунке 3.2 показан пример гиперплоскостей указанного вида.

Страница изъята

Страница изъята

Страница изъята

исследуем свойства функции $K(x, y) = (\varphi(x), \varphi(y))$, которая называется ядром. Приведем следующую теорему, которая полностью характеризует ядро. Она была доказана Джеймсом Мерсером в 1909 году в работе [15].

Теорема 3.1 *Функция $K(x, y) = (\varphi(x), \varphi(y))$ является ядром тогда и только тогда, когда она удовлетворяет условию симметричности $K(x, y) = K(y, x)$ и функция $K(x, y)$ неотрицательно определена, т.е. матрица $K = (K_{i,j})$, $K_{i,j} = K(x_i, x_j)$ является неотрицательно определенной для любых векторов x_1, \dots, x_m .*

Теорема 3.2 *Справедливы следующие свойства ядер: сумма ядер – ядро, произведение ядер – ядро, сумма равномерно сходящегося ряда ядер – ядро и композиция ядра и любого отображения – ядро.*

Следствие 3.1 *Многочлен с положительными коэффициентами от ядра – ядро,*

Следствие 3.2 *Экспонента от ядра – ядро,*

Следствие 3.3 *Функция $e^{-\|x-y\|^2}$ – ядро.*

В силу теоремы 3.2 любые $m+1$ векторов могут быть разделены на любые два класса с помощью мономинального отображения степени не больше m . Поэтому, если $\varphi: x \rightarrow \{x_1^{i_1}, \dots, x_n^{i_n}\}, i_1 + \dots + i_n \leq m$ такое отображение, то ядро, соответствующее этому отображению, можно искать в виде $K(x, y) = (\varphi(x), \varphi(y)) = ((x, y) + 1)^m$. Таким образом, это ядро гарантирует разделение любых $m+1$ векторов на любые два класса. В этом случае нахождение разделяющей функции осуществляется по следующему алгоритму.

1. Находим наибольшее значение функции в виде

$$\Phi(\lambda) = \sum_{i=2}^N \lambda_i - 0.5 \sum_{i=1}^N \lambda_i \lambda_j y_i y_j K(x_i, x_j) \quad (3.9)$$

при условии $\sum_{i=1}^N \lambda_i y_i = 0$ в области $\lambda_i \geq 0 (i = 1, \dots, N)$, получим вектор

$$\lambda^{(0)} = (\lambda_1^{(0)}, \dots, \lambda_N^{(0)}),$$

2. Разделяющую функцию находим в виде:

Страница изъята

Страница изъята

- $\beta_k = \frac{(r_{k+1}, r_{+1})}{(r_k, r_k)},$
- $p_{k+1} = r_{k+1} + \beta_k p_k.$

Если все вычисления и исходные данные точны, то метод сходится к решению системы не более чем за n итераций, где n – размерность системы. Более тонкий анализ показывает, что число итераций не превышает m , где m – число различных собственных значений матрицы A [3].

3.5 Метод наискорейшего спуска

Существует достаточно большое количество градиентных методов. Выделяют метод покоординатного спуска с постоянным шагом, с дробным шагом и метод наискорейшего спуска. В данной работе будет использоваться метод наискорейшего спуска, так как данный метод, среди всех выше представленных, сходится к решению с наименьшим числом итераций. Рассмотрим алгоритм данного метода.

В работе рассматривается задача поиска минимума $f(x): R^n \rightarrow R$, записываемая в виде $f(x) \rightarrow \min_{x \in R^n}$. Пусть функция $f(x)$ такова, что можно вычислить ее градиент [6].

Основная идея метода заключается в том, чтобы осуществлять оптимизацию в направлении наискорейшего спуска, а это направление задается антиградиентом $-\nabla f$ [6]:

$$x^{[k+1]} = x^{[k]} - \lambda^{[k]} \nabla f(x^{[k]}). \quad (3.11)$$

Этот вариант градиентного метода основывается на выборе шага из следующего соображения. Из точки $x^{[k]}$ будем двигаться в направлении антиградиента до тех пор, пока не достигнем минимума функции на этом направлении, т.е. на луче $L = \{x = x^{[k]} - \lambda \nabla f(x^{[k]}); \lambda \geq 0\}$ [6],

$$\lambda^{[k]} = \arg \min_{\lambda \in (0, \infty)} f(x^{[k]} - \lambda f'(x^{[k]})). \quad (3.12)$$

Другими словами, $\lambda^{[k]}$ выбираются так, чтобы следующая итерация была точкой минимума функции f на луче L . Такой вариант градиентного метода называется методом наискорейшего спуска. Заметим, кстати, что в этом методе направление соседних шагов ортогональны [6].

Метод наискорейшего спуска требует решения на каждом шаге задачи одномерной оптимизации. Практика показывает, что этот метод часто требует меньшего числа операций, чем градиентный метод с постоянным шагом [6].

В общей ситуации, тем не менее, теоретическая скорость сходимости метода наискорейшего спуска не выше скорости сходимости градиентного метода с постоянным (оптимальным) шагом [6].

4 Решение задачи с помощью нейронных сетей

4.1 Сети прямого распространения

Определение 4.1 *Искусственная нейронная сеть* – математическая модель, а также ее программное и аппаратное воплощение, построенная по принципу организации и функционирования биологических нейронных сетей – сетей нервных клеток живого организма [4].

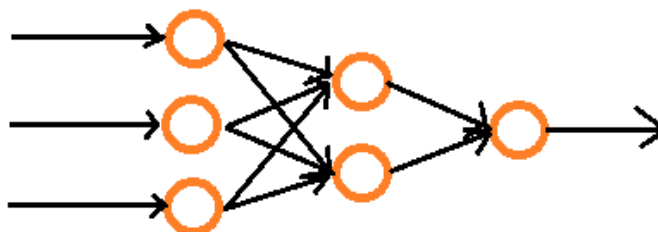


Рисунок 4.1 – Многослойная нейронная сеть

Нейронная сеть представляет собой систему связанных между собой нейронов.

Определение 4.2 *Нейрон* – электрически возбудимая клетка, которая обрабатывает, хранит и передает информацию с помощью электрических и химических сигналов.

Нейрон, в данной модели, может выполнять три функции: сбор информации связанных с ним нейронов на предыдущем слое, активация полученного значения некоторой активационной функцией, выдача активированного значения нейронам на следующем слое.

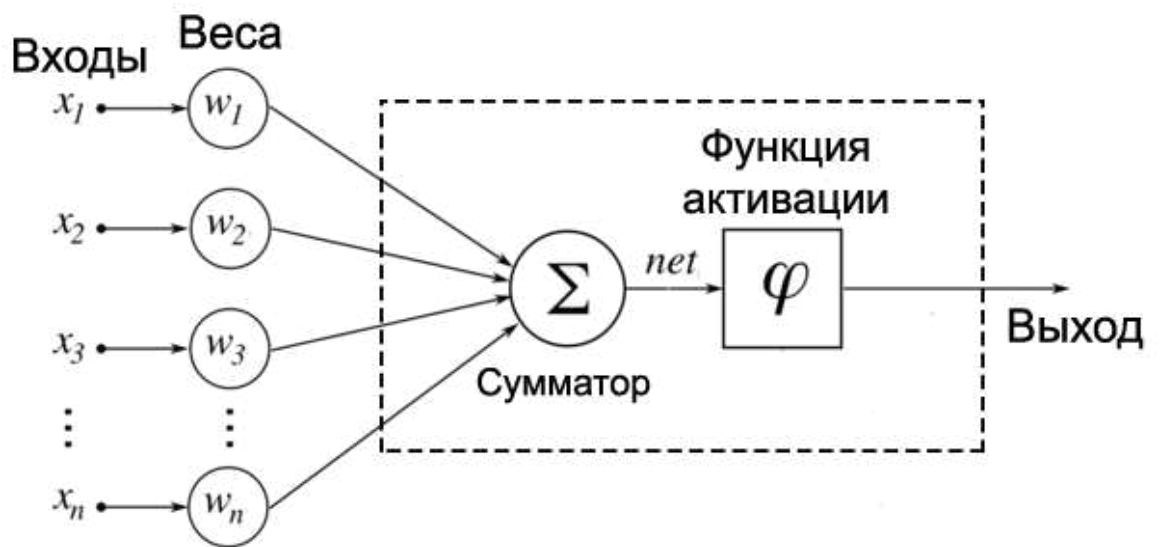


Рисунок 3.2 – Нейрон

Среди активационных функций чаще всего используют следующие:

- Линейно пороговая функция $f(x) = \begin{cases} 0, & x < 0.5 \\ 1, & x \geq 0.5 \end{cases}$,
- Сигмоидная функция $f(x) = \frac{1}{1 + \exp(-\alpha x)}$,
- Гиперболический тангенс $f(x) = th\left(\frac{x}{\alpha}\right)$.

Наибольшее внимание стоит уделить связям нейронов. Каждая такая связь называется *синапсом*. Синапсы имеют некоторый коэффициент, называемый весом синапса. Если некоторое значение проходит через синапс, то это значение будет изменено путем перемножения этого значения и веса синапса. Как правило, обучение нейронных сетей направлено на вычисление таких значений, чтобы определенная поставленная задача для нейронной сети решалась с минимальным числом ошибок.

Нейронные сети активно исследуются и изучаются в настоящее время. Их используют во многих областях информатики. Классификация, кластеризация, прогнозирование, построение регрессии – вот неполный перечень задач, которые могут быть решены с использованием различных моделей нейронных сетей.

Существует, довольно, большая классификация нейронных сетей: сети прямого распространения, рекуррентные нейронные сети, радиально – базисные нейронные сети, самоорганизующиеся карты. В данной работе будут рассматриваться сети прямого распространения.

Определение 4.3 *Многослойная нейронная сеть* – сеть, которая имеет хотя бы один скрытый слой.

Определение 4.4 *Сеть прямого распространения* – сеть, в которой нейроны скрытого слоя не имеют связей с нейронами на предыдущих слоях.

Явный пример сети прямого распространения проиллюстрирован на рисунке 4.1. На практике чаще используют именно многослойные нейронные сети, так как область применения однослойных сетей крайне невелика. В данной работе рассматривается многослойная нейронная сеть.

Существует большое количество методов обучения нейронных сетей, как с учителем, так и без него. В данной работе мы будем использовать метод обратного распространения ошибки.

4.2 Метод обратного распространения ошибки

Алгоритм обратного распространения ошибки представляет собой градиентный алгоритм обучения многослойного персептрона, основанный на минимизации среднеквадратической ошибки выходов сети. Алгоритм был предложен в 1974 году П. Дж. Вербосом, а в 1986 году его развили Д. Румельхарт и Р. Вильямс. Появление алгоритма дало дополнительный толчок развитию нейросетевых технологий [21].

Основная идея алгоритмов обучения с учителем, к которому относится и алгоритм обратного распространения ошибки, заключается в том, что на основе разности между желаемым и целевым выходами сети можно вычислить выходную ошибку сети. Цель определения выходной ошибки – управление процессом обучения нейронной сети, то есть корректировки весов ее межнейронных связей для минимизации функции ошибки [21].

Страница изъята

Страница изъята

Страница изъята

Страница изъята

5 Метод потенциальной функции

5.1 Физический смысл

Каждая предъявляемая выборка может быть представлена точкой в пространстве признаков. Представим, что в каждую такую точку помещен электрический заряд. В некоторой произвольной точке M совокупность всех зарядов создает электрический потенциал V , являющийся суммой отдельных потенциалов, создаваемых каждым отдельным зарядом.

Известно, что величина потенциала вычисляется как сумма

$$V = \frac{1}{4\pi\epsilon_0} \sum_i \frac{q_i}{r_i} = \sum_i V_i, \quad (5.1)$$

где q_i – заряд в точке p_i ; r_i – расстояние от точки p_i до точки M .

Потенциал выражается функцией, симметричной относительно точки, в которую помещен заряд; в этой точке потенциал, по определению, равен бесконечности. Линии, соединяющие точки равного потенциала, называются *экипотенциальными*.

Можно представить, что всякое облако точек, отображающих некоторый класс, выглядит как некое потенциальное плато, отделенное от другого плато, отображающего другой класс, глубокой долиной, потенциал в которой минимален или даже равен нулю. Определение минимальной (или нулевой) экипотенциали позволит найти границу раздела между классами.

5.2 Описание метода

Обозначим потенциальную функцию $K(x, x_k)$, центрированную относительно x_k . Для любой точки x и для любого x_k можно выбрать некоторое K , имеющее вид

$$K(x, x_k) = \sum_{i=2}^{\infty} \lambda_i^2 \varphi_i(x) \varphi_i(x_k), \quad (5.2)$$

где λ_i выбраны такими, чтобы удовлетворялись условия: $\lambda_i \neq 0, \sum_i \lambda_i^2 < \infty$, а функции $\varphi_i(x)$ представляют собою элементы последовательностей ортонормированных функций.

Другой метод создания потенциальных функций состоит в использовании свойства симметрии относительно x_k , т.е. $K(x_k, x) = K(x, x_k)$.

Можно принять, например, такие функции:

- $K(x, x_k) = \exp(-\alpha \|x - x_k\|^2)$,
- $K(x, x_k) = \frac{1}{\alpha \|x - x_k\|^2}$,
- $K(x, x_k) = \frac{\sin(\alpha \|x - x_k\|^2)}{\alpha \|x - x_k\|^2}$, где α – положительная константа.

Рассмотрим процесс обучения, при котором распознающей машине предъявляются образы, о которых заранее известно, что они принадлежат к одному из двух классов – либо Ω_1 , либо Ω_2 .

Система предварительной обработки выделяет надлежащим образом выбранные параметры (признаки). Набор признаков позволяет построить вектор x_{i+1} для $(i + 1)$ -го образа.

Разделяющая функция находится с помощью суммарного потенциала $K(x)$, вычисляемого как сумма частных потенциалов $K(x, x_k)$, связанных с каждым отдельным источником.

Он находится путем применения следующего алгоритма, в котором через i обозначен номер этапа:

$$K_{i+1}(x) = K_i(x) + \rho_{i+1} K(x, x_{i+1}). \quad (5.3)$$

Корректирующий член ρ_{i+1} удовлетворяет следующим условиям:

$$\rho_{i+1} = \begin{cases} +1, & x_{i+1} \in \Omega_1, K_i(x_{i+1}) \leq 0, \\ -1, & x_{i+1} \in \Omega_2, K_i(x_{i+1}) > 0, \\ 0, & \text{else.} \end{cases} \quad (5.4)$$

Правильная классификация соответствует случаям, когда $K_i(x_{i+1} > 0)$ при $x_{i+1} \in \Omega_1$ и $K_i(x_{i+1} \leq 0)$ при $x_{i+1} \in \Omega_2$. Поэтому можно использовать $K_i(x)$ как разделяющую функцию и определить ее итеративным методом

$$d_{i+1}(x) = d_i(x) + \rho_{i+1}K(x, x_{i+1}). \quad (5.5)$$

6 Решение практической задачи распознавания изображений

Рассмотрим частную задачу распознавания образов и машинного обучения – распознавания изображений. Целью задачи распознавания изображений является анализ изображения и классификация его к определенному классу. На практике наиболее часто решают такие задачи распознавания изображений как: распознавание лиц, распознавание символов, оптическое распознавание рукописного текста, распознавание автомобильных номеров, распознавание штрих-кодов, распознавание отпечатков пальцев, распознавание объектов специфических категорий на изображениях и многие другие.

В данной работе решается практический пример задачи распознавания изображений – задачи распознавания пешеходов, т.е. задача нахождения на изображении человека.

Был разработан программный модуль «Распознавание пешеходов», в котором была реализована работа метода опорных векторов, нейронной сети и потенциальной функции. Данное программное средство предназначено для решения поставленной практической задачи. Для каждого метода предусматривается отдельный элемент управления. Подробно вид программного средства и вид отображения результатов решения задачи приведены в следующей главе.

На вход программному модулю подавалось некоторое изображение. Далее, программа, используя один из трёх перечисленных методов, должна была дать соответствующий ответ – имеется ли на изображении пешеход или нет (выдать сообщение «Пешеход» или «Не пешеход»).

6.1 Вычислительный эксперимент

В работе была проведена серия вычислительных экспериментов с использованием реализованного программного обеспечения «Распознавание пешеходов». Данные эксперименты были выполнены на персональном компьютере со следующими характеристиками:

- Amd A4 – 3305M (1900 Ghz – x2),
- ОЗУ 6 Гб,
- Amd 7670m (1024 Mb),
- Windows 10

Вычислительные эксперименты позволяют оценить работу алгоритмов на ряде тестовых обучающих выборок. Были созданы три различные обучающие выборки, состоящие из 5, 10 и 15 изображений соответственно. В каждой выборке находились как изображения с пешеходами, так и без них. Размер данных изображений мог быть любым, далее программный модуль меняет разрешение на 64*128 пикселей. Изображения могут быть как цветные, так и черно-белые. На прецеденты, относящиеся к классу «Пешеход» накладывается важное условие. В фокусе данных изображений обязательно должен быть человек, расположенный по центру, как можно крупнее.

Итак, каждому из трёх методов, реализованных в программном средстве, на вход была подана обучающая выборка. После обучения алгоритму были даны 10 заранее подобранных изображений, для которых он должен был решить задачу распознавания пешеходов, т.е. решить – есть ли на изображении пешеход или нет. На входные изображения накладываются те же условия, что и на файлы обучающей выборки. Все результаты фиксировались экспертом.

Проведем исследование работы методов по следующим критериям: определим количество ошибок распознавания первого и второго рода, найдём скорость сходимости и процент верно распознанных изображений, а также оценим вычислительную сложность алгоритмов. Проведем сравнение представленных методов на основе полученных результатов, составив сравнительную таблицу.

Определение 6.1 *Ошибка первого рода* – вероятность принять основной класс за вторичный.

Определение 6.2 *Ошибка второго рода* – вероятность принять вторичный класс за основной.

В таблице 1 приведены результаты проведенного вычислительного эксперимента.

Таблица 1 – Результаты вычислительного эксперимента

Метод	Обучающая выборка	Число ошибок первого рода	Число ошибок второго рода	Процент верно распознанных изображений
Метод опорных векторов и метод сопряженных градиентов	5	2	1	70%
	10	3	0	70%
	15	3	1	60%
Метод опорных векторов и метод наискорейшего спуска	5	2	1	70%
	10	3	0	70%
	15	3	1	60%
Нейронная сеть	5	3	0	70%
	10	2	2	60%
	15	1	3	60%
Метод потенциальной функции	5	2	1	70%
	10	3	0	70%
	15	4	0	60%

Из результатов, представленных в таблице 1, видно, что наилучшим образом себя показывает метод потенциальной функции. Это единственный метод, прошедший обучение на всех трёх обучающих выборках. Стоит отметить,

что на обучающей выборке из 15 файлов, данный метод допустил 4 ошибки первого рода. Это говорит о том, что обучающую выборку требуется увеличивать для увеличения качества распознавания.

Так же видно, что методы допускают ошибки первого и второго рода в одинаковом количестве. Это говорит о том, что данные классификаторы достаточно близки и дают схожие результаты.

6.2 Сравнение представленных алгоритмов

В таблице 2 приведены результаты по скорости обучения данных алгоритмов, для всех ранее подобранных выборок и их вычислительной сложности.

Таблица 2 – Анализ алгоритмов по скорости обучения и вычислительной сложности

Метод	Обучающая выборка	Скорость обучения	Вычислительная сложность
Метод опорных векторов и метод сопряженных градиентов	5	< 1 секунды	Сходится, в теории, не более чем за n итераций. $O[n^4]$ операций за итерацию
	10	< 2 секунд	
	15	< 3 минут	

Продолжение таблицы 2 – Анализ алгоритмов по скорости обучения и вычислительной сложности

Метод опорных векторов и метод наискорейшего спуска	5	< 5 минут	Итерационный. $O[n^3]$ операций за итерацию
	10	< 60 минут	
	15	< 180 минут	
Нейронная сеть	5	< 180 минут	Итерационный. $O[n^4]$ операций за итерацию
	10	> 24 часов	
	15	> 48 часов	
Метод потенциальной функции	5	< 1 секунд	Итерационный. $O[n^3]$ операций за итерацию
	10	< 5 секунд	
	15	< 30 секунд	

По результатам таблицы 2 видно, что наилучшим образом себя показывает метод потенциальной функции. Он отличается простой реализацией и обладает сходимостью на самых различных обучающих выборках.

Метод опорных векторов сложно реализуем и требует решения многомерной оптимизационной задачи. В случае, когда для решения оптимизационной задачи используется алгоритм наискорейшего спуска, требуется дополнительно решать много одномерных оптимизационных задач, что увеличивает вычислительную сложность метода и приводит к непредвиденным результатам. Если используется метод сопряженных градиентов для решения оптимизационной задачи, то алгоритм распознавания ведет себя стабильно и дает достаточно хорошие результаты.

Нейронные сети способны решать различные задачи, связанные с анализом данных, что несомненно является положительной чертой данного подхода. Обучение нейронной сети можно свести к задаче о нахождении минимума некоторой функции. Правило Видлоу – Хоффа предусматривает подбор

некоторого коэффициента обучения, который отвечает за скорость спуска в точку минимума. На практике, такой коэффициент достаточно трудно подобрать, что является минусом данного алгоритма. Возникают ситуации, когда сеть проходит необходимую точку, либо спускается к данной точке слишком медленно.

7 Описание программного модуля

Как было сказано в предыдущей главе, в работе решается практический пример задачи распознавания изображений – задачи распознавания пешеходов. Для решения данной задачи было разработано программное средство, в котором реализована работа метода опорных векторов, нейронной сети и потенциальной функции.

Программный модуль носит название «Распознавание пешеходов».

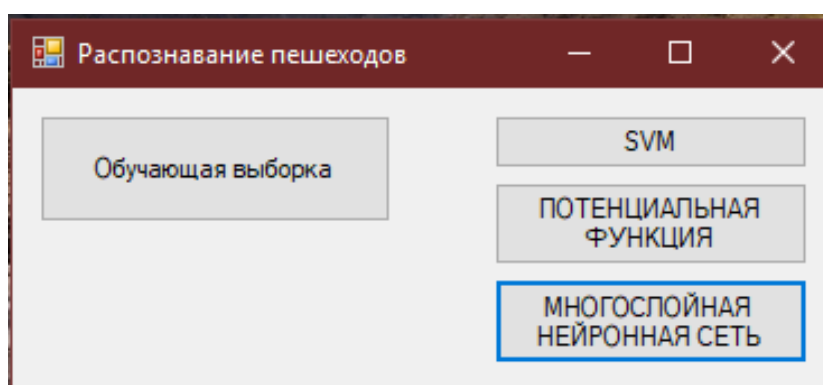


Рисунок 7.1 – Стартовое окно

На рисунке 7.1. изображено стартовое окно, которое используется для выбора одного из трёх реализованных методов распознавания изображений (кнопки **SVM**, **Потенциальная функция** и **Многослойная нейронная сеть**, соответственно) и для перехода к исследованию обучающей выборки.

В программном модуле реализована возможность наблюдать за обучающей выборкой. Можно выбрать любое изображение и посмотреть на его градиент. Также имеется возможность наблюдать за гистограммами ячеек изображения и проводить с данными результатами дополнительные исследования. Данная возможность продемонстрирована на рисунке 7.2.



Рисунок 7.2 – Обучающая выборка

7.1 Программный модуль вектора опорных векторов

Для каждого рассмотренного метода реализован отдельный программный модуль в виде формы. На рисунках 7.3 и 7.4. изображена форма метода опорных векторов. Из рисунка видно, что имеется возможность выбрать метод для решения оптимизационной задачи в методе. Реализована загрузка изображения в программу и проверка на наличие пешехода.

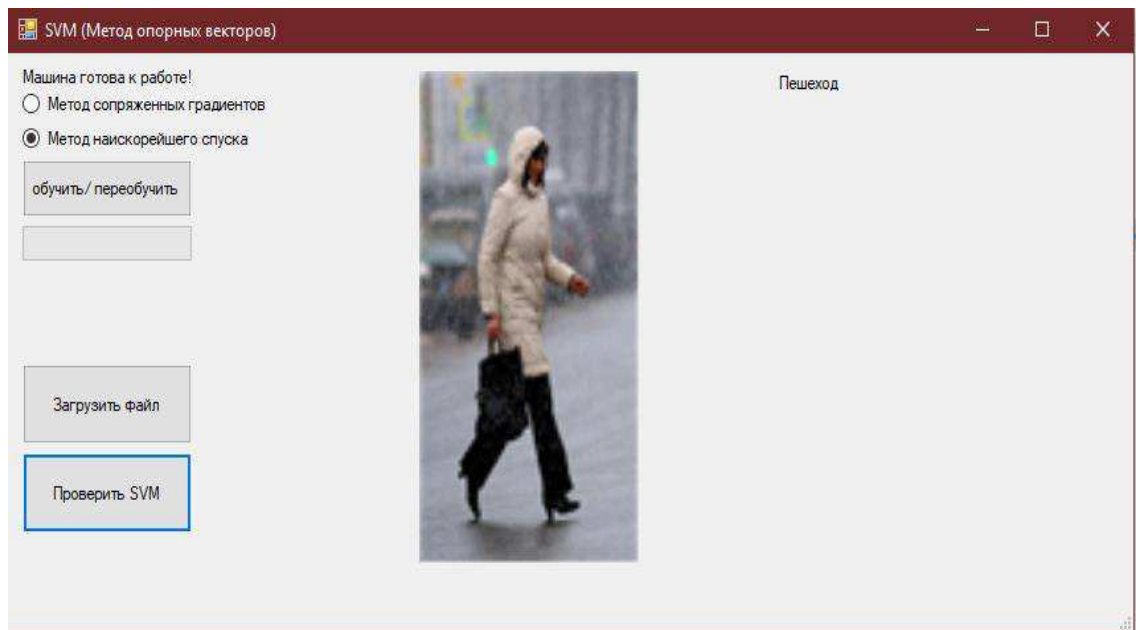


Рисунок 7.3 – Форма метода опорных векторов

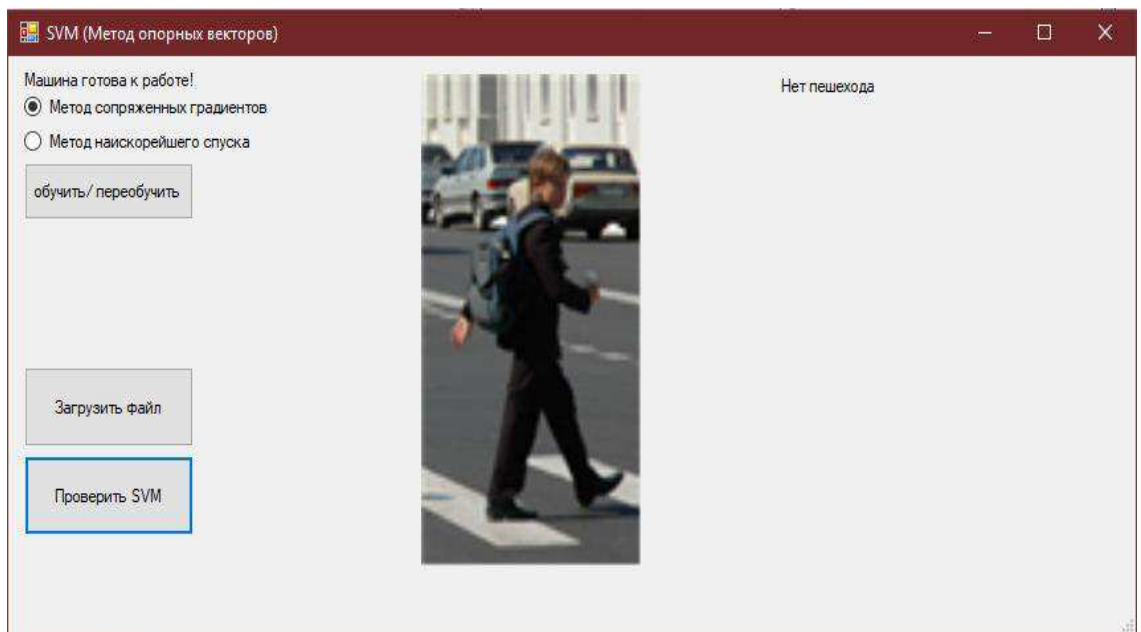


Рисунок 7.4 – Форма метода опорных векторов

На рисунке 7.3 видно, что распознавание произведено без ошибки, а на изображении 7.4 допущена ошибка первого рода.

7.2 Программный модуль алгоритма нейронных сетей

Для нейронных сетей реализована отдельная форма. Пользователю дана возможность выбрать параметры сети, такие как: точность, скорость обучения, наличие биас-нейронов, количество скрытых слоев и количество нейронов на скрытом слое. На рисунках 7.5 и 7.6. изображена форма алгоритма нейронной сети.

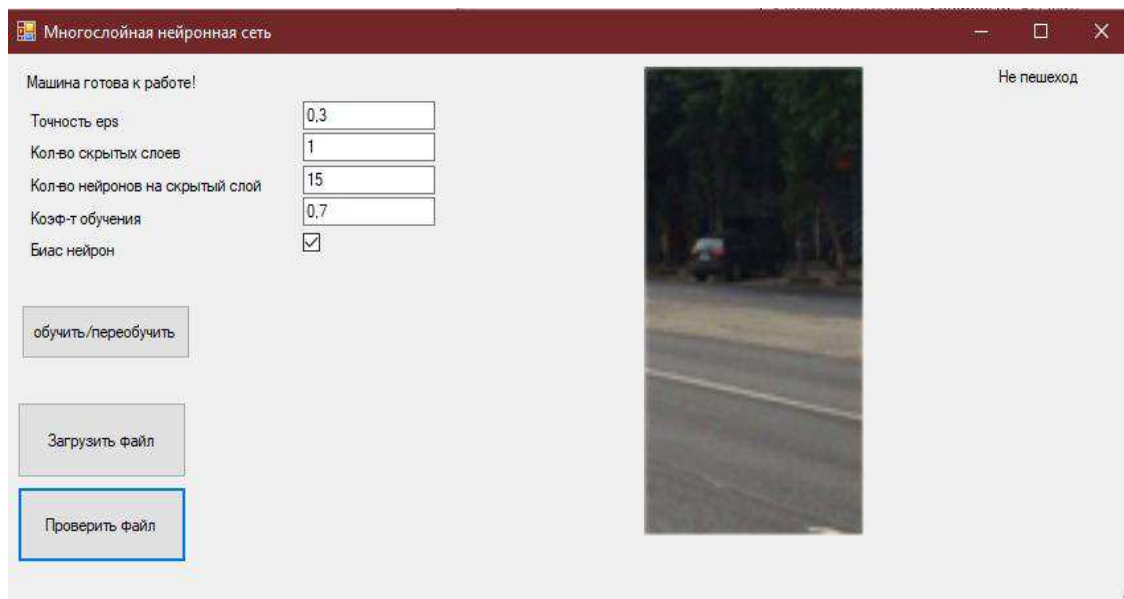


Рисунок 7.5 – Форма алгоритма нейронных сетей

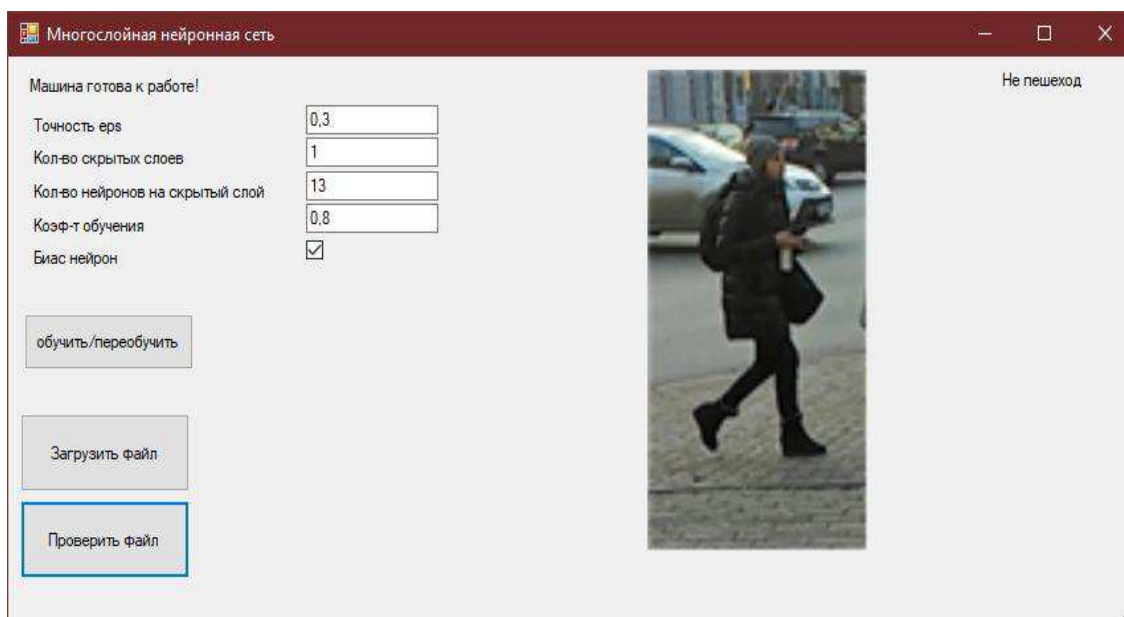


Рисунок 7.6 – Форма алгоритма нейронных сетей

Также на данной форме предусмотрена возможность загрузить изображение и применить к нему обученный классификатор. На рисунке 7.5 приведен пример верной классификации, а на рисунке 7.6 допущена ошибка первого рода.

7.3. Программный модуль метода потенциальной функции

Для метода потенциальной функции также реализована отдельная форма. Пример этой формы приведен на изображениях 7.7 и 7.8 соответственно.

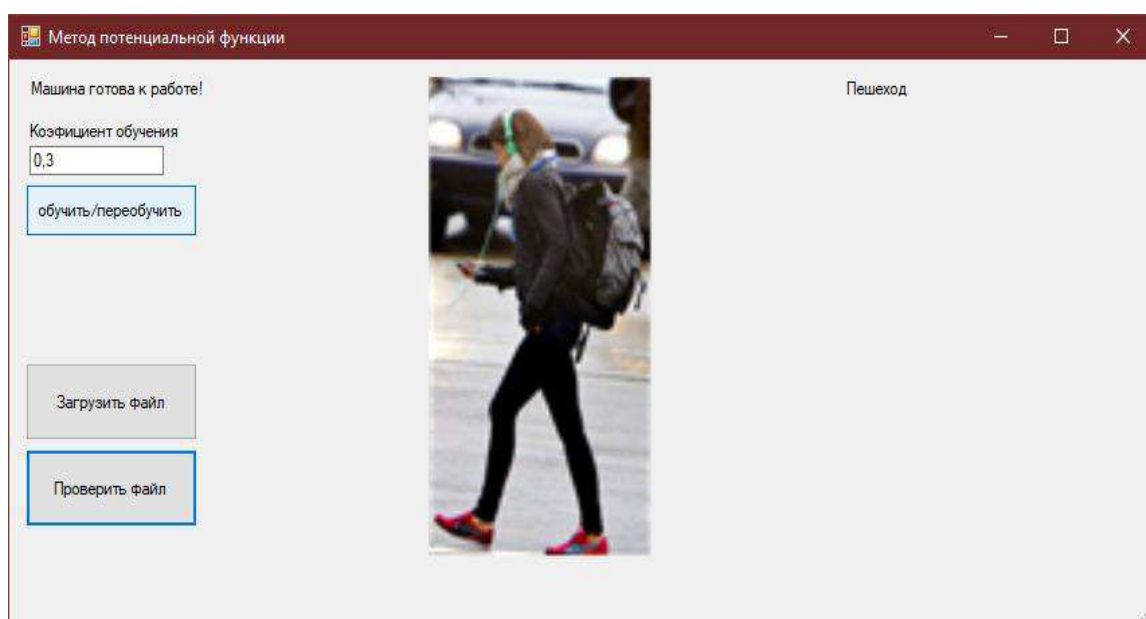


Рисунок 7.7 – Форма метода потенциальной функции

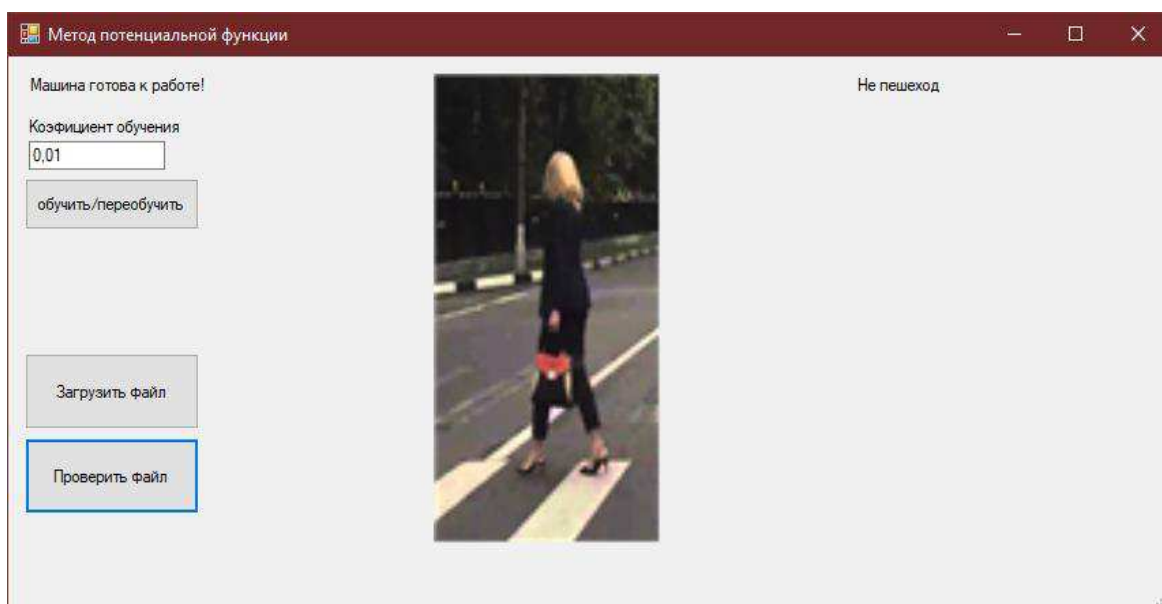


Рисунок 7.8 – Форма метода потенциальной функции

Пользователю предоставлена возможность подобрать коэффициент скорости обучения. Имеется возможность загрузить изображение в программу и применить к нему классификатор. На рисунке 7.7 приведен пример верной классификации, а на рисунке 7.8 допущена ошибка первого рода.

ЗАКЛЮЧЕНИЕ

В работе были получены следующие результаты:

- Изучены постановка задачи распознавания образов и основные методы распознавания.
- Разработано программное обеспечение, реализующее данные алгоритмы.
- Проведены вычислительные эксперименты по сравнению вычислительной сложности и условий сходимости методов.
- Решена практическая задача распознавания образов.
- Выполнен сравнительный анализ полученных результатов.

Результаты работы опубликованы на международной научно-технической конференции студентов, аспирантов и молодых ученых «Перспектив Свободный – 2016» (Красноярск, 2016) и международной научно-технической конференции студентов, аспирантов и молодых ученых «Перспектив Свободный – 2017» (Красноярск, 2017).

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Аксенов, С. В. Организация и использование нейронных сетей (методы и технологии) / С. В. Аксенов; под общ. ред. В. Б. Новосельцева. – Томск: Изд-во НТЛ, 2006. – 128 с.
2. Бахвалов, Н.С. Численные методы / Н. С. Бахвалов, Н. П. Жидков, Г. М. Кобельков. – Изд. 6-е. – М.: БИНОМ: Лаборатория знаний, 2008. — 636 с.
3. Васильев Ф. П. Методы оптимизации / Ф. П. Васильев. – Москва: Факториал пресс, 2002. – 824 с.
4. Генетические алгоритмы, искусственные нейронные сети и проблемы виртуальной реальности / Г. К. Вороновский [и др.] – Харьков: Основа, 1997. – 107 с.
5. Воронцов, К. В. Лекции по алгоритмам кластеризации и многомерного шкалирования / К. В. Воронцов. – Москва: МГУ, 2007. – 18 с.
6. Глебов, Н. И. Методы оптимизации: учебное пособие / Н. И. Глебов, Ю. А. Кочетов, А. В. Плясунов. – Новосибирск: НГУ, 2000. – 105 с.
7. Нейроинформатика / А. Н. Горбань [и др.]. – Новосибирск: Наука, 1998. – 296 с.
8. Горбань, А. Н. Нейронные сети на персональном компьютере / А. Н. Горбань, Д. А. Россиев. — Новосибирск: Наука, 1996. — 276 с.
9. Докучаев, Д. А. Методы решения задач распознавания изображений / Д. А. Докучаев // Сборник материалов международной научно-технической конференция студентов, аспирантов и молодых учёных «Перспектив Свободный-2016». – Красноярск, СФУ, 2016. – С. 22-24.
10. Докучаев, Д. А. Исследование методов распознавания изображений / Д. А. Докучаев // Сборник материалов международной научно-технической конференция студентов, аспирантов и молодых учёных «Перспектив Свободный-2017». – Красноярск, СФУ, 2017. (в печати).

11. Еремин, Д. М. Искусственные нейронные сети в интеллектуальных системах управления: учеб. пособие / Д. М. Еремин, И. Б. Гарцеев. – Москва: МГИРЭА, 2004. – 75 с.
12. Дуда, Р. Распознавание образов и анализ сцен: пер. с англ. Г. Г. Вайнштейна, А. М. Васьковского / Р. Дуда, П. Харт; под ред. В. Л. Стефанюка. – Москва: Мир, 1976. – 502 с.
13. Загоруйко, Н. Г. Прикладные методы анализа данных и знаний / Н. Г. Загоруйко. – Новосибирск: ИМ СО РАН, 1999. – 270 с.
14. Калиткин, Н. Н. Численные методы / Н. Н. Калиткин; под ред. А. А. Самарского. – Москва: Наука, 1978. – 512 с.
15. Лепский, А. Е. Математические методы распознавания образов / А. Е. Лепский, А. Г. Броневиц. – Таганрог: Изд-во ТТИ ЮФУ, 2009. – 155 с.
16. Лоран, П. Ж. Аппроксимация и оптимизация: пер. с франц. Ю. С. Завьялова, Р. А. Звягиной, В. И. Квасова, В. И. Шмырева / П. Ж. Лоран; под ред. Г. Ш. Рубинштейна, Н. Н. Яненко. – Москва: Мир, 1975. – 496 с.
17. Местецкий, Л. М. Математические методы распознавания образов / Л. М. Местецкий. – Москва: МГУ, 2002. – 139 с.
18. Патрик, Э. А. Основы теории распознавания образов : пер. с англ. В. М. Баронкина, Б. А. Смиренина, Ю. С. Шинакова под ред. Б. Р. Левина / Э. А. Патрик. – Москва: Советское радио, 1980. – 403 с.
19. Потапов, А. С. Распознавание образов и машинное восприятие / А. С. Потапов. — М.: Политехника, 2007. — 552 с.
20. Рутковская, Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Д. Рутковская; пер. с польского И. Д. Рудинского. — М.: Горячая линия - Телеком, 2006. — 452с.
21. Сахнюк П. А. Алгоритм обратного распространения ошибки / П. А. Сахнюк. [Электронный ресурс] – Режим доступа: <http://www.basegroup.ru>
22. Сизов, А. С. Задача о минимуме квадратичного функционала / А. С. Сизов, П. А. Скрипниченко. [Электронный ресурс] – Режим доступа: <http://vi.horizontalru.com/23.html>

23. Ту, Д. Принципы распознавания образов : пер с англ. И. Б. Гуревича под ред. Ю. И. Журавлева / Д. Ту, Р. Гонсалес. – Москва: Мир, 1978. – 401.
24. Хант, Э. Искусственный интеллект: пер. с англ. Д. А. Белова, Ю. И. Крюкова под ред. В. Л. Стефанюка / Э. Хант. – Москва: Мир, 1978. – 550 с.
25. Ясницкий, Л. Н. Введение в искусственный интеллект / Л. Н. Ясницкий. – М.: Академия, 2005. – 176 с.
26. Nocedal J. Numerical Optimization / J. Nocedal, S. J. Wright. – New York: Springer-Verlag. – 634 с.

Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики и фундаментальной информатики
Базовая кафедра вычислительных и информационных технологий

УТВЕРЖДАЮ
Заведующий кафедрой
Шайду / В.В. Шайдуров
«16» июня 2017 г.

БАКАЛАВРСКАЯ РАБОТА

Направление 02.03.01 Математика и компьютерные науки

ИССЛЕДОВАНИЕ МЕТОДОВ РЕШЕНИЯ ЗАДАЧИ РАСПОЗНАВАНИЯ ИЗОБРАЖЕНИЙ

Научный руководитель
кандидат физико-математических наук,
доцент

И.В. Баранова / И.В. Баранова
16.06.17

Выпускник

Докучаев / Д.А. Докучаев
16.06.17

Красноярск 2017