

Санкт-Петербургский государственный университет  
Факультет прикладной математики - процессов управления  
Кафедра технологии программирования

**Целикова Светлана Олеговна**

**Выпускная квалификационная работа бакалавра**

**Разработка системы распознавания эмоций с  
использованием нейросетевых технологий**

Направление 01.03.02

Прикладная математика и информатика

Научный руководитель,  
старший преподаватель  
Стученков А. Б.

Санкт-Петербург

2019

# Содержание

Введение.....	3
Постановка задачи.....	5
Обзор литературы.....	6
Глава 1. Обзор существующих подходов и реализаций .....	8
1.1 Существующие подходы к решению задачи распознавания эмоций.....	8
1.2 Существующие решения задачи распознавания эмоций.....	12
Глава 2. Основные понятия и методы.....	14
2.1 Искусственная нейронная сеть .....	14
2.2 Сверточная нейронная сеть.....	15
2.3 Теоретический обзор используемых методов.....	20
Глава 3. Решение поставленной задачи .....	25
3.1 Используемые программные средства.....	25
3.2 Обучающая выборка .....	25
3.3 Предобработка данных для обучения нейронной сети.....	26
3.4 Создание и обучение нейронной сети .....	29
3.5 Объектная модель приложения .....	32
3.6 Разработка пользовательского интерфейса программы .....	33
Глава 4. Тестирование результатов .....	36
Выводы .....	39
Заключение .....	40
Список литературы .....	41

## Введение

«Эмоции - субъективные состояния человека и животных, возникающие в ответ на воздействие внешних или внутренних раздражителей и проявляющиеся в форме непосредственных переживаний» [1]. Эмоции являются неотъемлемой частью жизни людей и межличностного общения. Они могут демонстрироваться различными способами: мимикой, жестами, интонацией, позой, однако наибольшей выразительностью обладает лицо человека.

Задачи, связанные с распознаванием лицевых экспрессий человека, давно являлись предметом исследований психологов и ученых. Так Чарльзом Дарвином была опубликована работа «О выражении эмоций у человека и животных». [2]. Он первым высказал гипотезу об универсальности мимических проявлений эмоций, причем не только для различных культур, но и для различных видов. Данная теория была проверена и продолжена психологом Полом Экманом [3]. В ходе наблюдения за лицевыми экспрессиями жителей одного из племен в Новой Гвинее, ни разу не имевшими долговременного контакта с представителями западной или восточной культуры, ученый заметил, что некоторые эмоции выражались таким же образом, что и во всем мире. Пол Экман установил, что такие эмоции как: гнев, удивление, печаль, радость, страх и отвращение являются универсальными и могут быть поняты человеком, независимо от его культуры.

Задача автоматического распознавания эмоций по мимике является актуальной в наши дни, поскольку имеет большое количество применений в различных сферах человеческой деятельности. Так, детекция эмоциональных реакций может использоваться в маркетинге. Система, автоматически определяя эмоциональный отклик человека на демонстрируемый ему контент, показывает степень его заинтересованности в том или ином товаре или услуге. Автоматическое распознавание эмоций может применяться правоохранительными органами и охранными службами. С их помощью следователи могут определять истинность показаний подозреваемого при допросе. Кроме того, системы могут помочь обнаружить людей,

демонстрирующих эмоциональные реакции, характерные для психически нездоровых лиц, нарушителей правопорядка и т. д. Системы распознавания эмоций могут повысить безопасность на дорогах. Оснащение автомобилей подобными средствами в ряде случаев может позволить избежать аварий, связанных с плохим самочувствием или сонливостью водителей. Кроме всего прочего, программы для определения эмоционального состояния могут повысить степень эффективности взаимодействия человека и компьютера, обеспечить правильную реакцию на состояние пользователя.

Особенностью данной работы является то, что помимо системы распознавания эмоциональных реакций, разработанная программа включает также систему тренировки эмоций. Из-за проблем со здоровьем, вследствие пережитых потрясений и других причин люди могут иметь проблемы с демонстрацией своих эмоций с помощью мимики. У подобного человека могут возникать трудности при взаимодействии с окружающими. Системы тренировки эмоций способны помочь в подобной ситуации.

## **Постановка задачи**

**Целью данной работы** является разработка системы распознавания и тренировки эмоций, основанной на нейросетевых технологиях. В контексте данной работы рассматриваются следующие 6 эмоций (базовые эмоции): гнев, удивление, печаль, радость, страх и отвращение.

Для осуществления поставленной цели необходимо решить следующие **задачи**:

- Провести теоретический анализ исследуемой области.
- Сформировать обучающую выборку для тренировки нейронной сети.
- Осуществить предобработку обучающей выборки.
- Выбрать архитектуру нейронной сети, обучить ее и применить полученную модель для распознавания эмоций на изображении.
- Выбрать метод детектирования лиц на изображении и применить его
- Разработать интерфейс программы.
- Протестировать полученную программу.

Подробнее перечисленные выше этапы будут рассмотрены в следующих главах.

## Обзор литературы

В разделе представлен краткий обзор основных литературных источников, используемых при написании данной работы и разработке рассматриваемой системы.

**1. Y. Chen, H. Wu, T. Wang, Y. Wang “A Comparison of Methods of Facial Expression Recognition”, 2018**

В статье представлена классификация существующих подходов к задаче автоматического распознавания лицевых экспрессий, а также дан краткий обзор данных подходов.

**2. D. A. Pitaloka, A. Wulandari, T. Basaruddin, D. Y. Liliana “Enhancing CNN with Preprocessing Stage in Automatic Emotion Recognition”, 2017**

В ходе эксперимента, описанного в статье, производится анализ методов предварительной обработки изображений с целью повышения точности нейронной сети, разрабатываемой для решения задачи автоматического распознавания эмоций, в частности, сравниваются способы нормализации изображений. При разработке системы в рамках данной работы использовался метод глобальной нормализации контраста, продемонстрировавший лучшие результаты согласно представленной статье.

**3. P. Viola and M.J. Jones, «Rapid Object Detection using a Boosted Cascade of Simple Features», 2001**

В статье представлен метод для обнаружения объектов на изображении в режиме реального времени, применяемый в данной работе для детектирования лиц на кадрах при работе приложения, а также при предварительной обработке данных.

**4. N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”, 2014**

В статье представлен метод прореживания - один из способов борьбы с проблемой переобучения в нейронных сетях. При обучении нейронной сети в рамках данной работы использовался представленный метод.

**5. S. Ioffe, C. Szegedy “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”, 2015**

В статье представлен метод ускорения обучения нейронной сети - батч-нормализация. Рассматриваемый метод также применялся при обучении нейронной сети в рамках данной работы.

# Глава 1. Обзор существующих подходов и реализаций

## 1.1 Существующие подходы к решению задачи распознавания эмоций

Большинство методов, применяемых для решения задачи классификации лицевых эмоций, состоят из следующих шагов [4]: сбор данных и их предварительная обработка, извлечение и выбор признаков, классификация и принятие решения (рис.1).

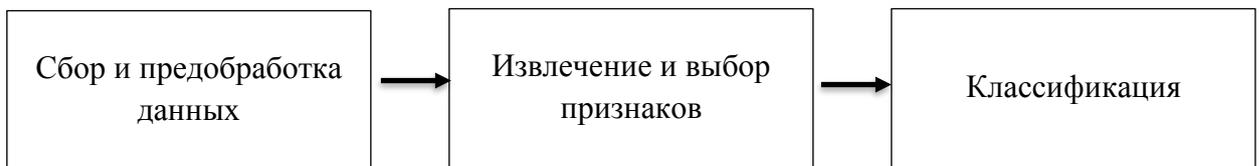


Рис.1: Схема работы систем распознавания эмоций

В отличие от используемого в данной работе метода на основе сверточной нейронной сети, внутри которой происходит и выбор признаков и последующее определение класса снимка, в ряде подходов выделение особенностей изображения и классификация представляют собой отдельные, не связанные между собой задачи, каждая из которых может быть решена одним из возможных способов. В случае же использования некоторых современных методов машинного обучения, таких как глубокое обучение [5],[6],[7], на вход алгоритму подается не набор признаков, а непосредственно само изображение и впоследствии выполняется определение его класса.

Рассмотрим существующие методы извлечения признаков, применяемые в подходах первого типа. Данные подходы могут быть разделены на группы, в зависимости от используемого пространства признаков. Наборы признаков, вычисляемые по изображениям лица, могут включать в себя текстурные или геометрические признаки [4].

**Геометрическими признаками** могут являться расстояния, углы и т.д. Они характеризуют особенности расположения, ориентации объектов на изображении. 2D или 3D модели обычно используется для поиска ключевых точек лица.

### **Active Shape Models (ASM)**

Данный метод [8] заключается в использовании статистических моделей форм объектов, которые могут быть деформированы, для подгонки под исследуемый объект на новом изображении.

На лицах со снимков обучающей выборки размечаются расположения антропометрических точек, при этом порядок нумерации точек на изображениях одинаков. Локализация точек ASM-модели на новом изображении происходит за счет решения задачи оптимизации. Для каждой точки определяется наилучшее совпадающее с шаблоном положение путем использования градиента текстуры изображения вблизи этой точки. В результате работы данного метода происходит локализация ключевых точек лица на изображении. Рассматриваемый метод можно использовать совместно с системой кодирования лицевых движений Пола Экмана [9].

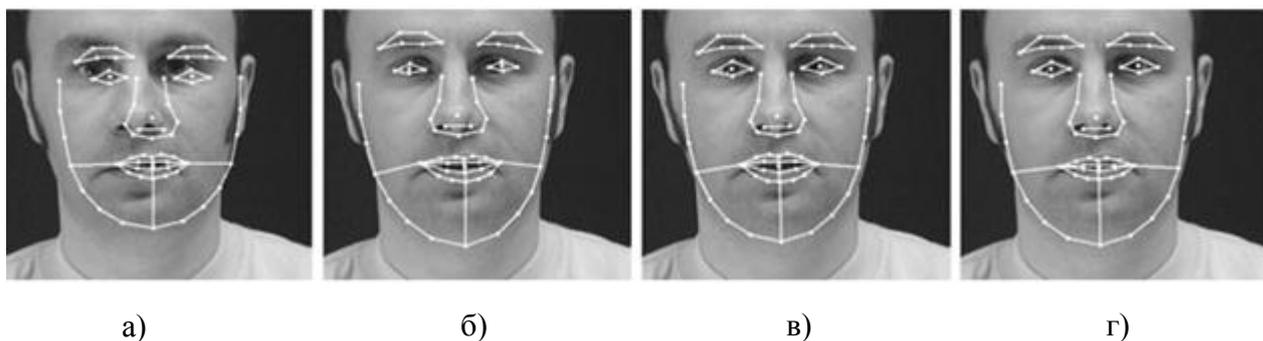


Рис.2: Процесс локализации ASM на изображении. а) – начальная позиция, б) – спустя 5 итераций, в) – спустя 10 итераций, г) – итоговая модель

**Текстурные признаки** отражают однородность изображения. Текстурными особенностями изображений лица являются в основном изменения образа текстуры, такие как морщины на коже и выпуклости.

### **Local Binary Pattern**

В процессе работы данного метода [10] каждый пиксель исходного изображения представляется в виде бинарного числа, которое зависит от интенсивностей соседних пикселей.

Вектор признаков создается следующим образом:

1. Исследуемое изображение делится на ячейки.
2. Каждый пиксель поочередно сравнивается с одним из его 8-ми соседей.
3. Если значение центрального пикселя больше значения соседнего, то в данную ячейку записывается 0, иначе - 1.
4. Совокупность значений всех соседних ячеек формирует восьмизначное двоичное число, которое описывает окрестность пикселя.
5. Для каждой ячейки вычисляют гистограмму частоты каждого встречающегося числа. Гистограмма может рассматриваться как 256-мерный вектор признаков.
6. Гистограммы всех ячеек объединяются, что дает вектор признаков всего изображения.

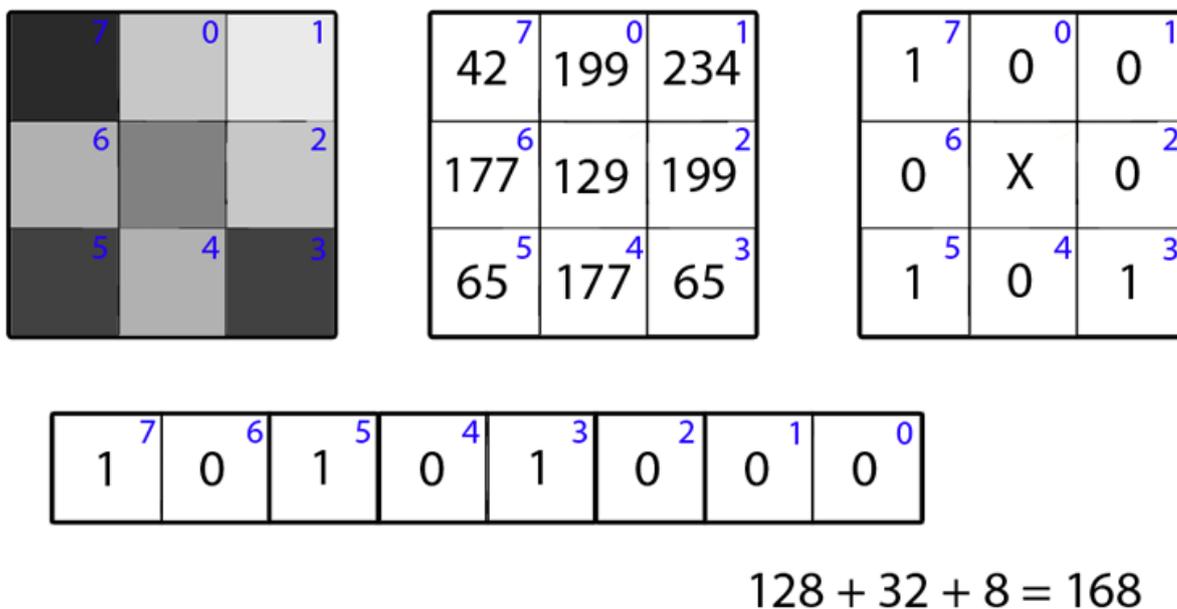


Рис.3: Иллюстрация принципа работы LBP

### Вейвлеты (фильтры) Габора

Текстурные особенности изображения также могут быть извлечены с помощью вейвлетов Габора [11]. Вейвлет-преобразование представляет собой

свертку изображения с функцией  $\psi$  –вейвлетом. В случае использования двумерного фильтра Габора с ориентацией  $\mu$  и масштабом  $\nu$  данная функция имеет вид:

$$\psi_{\mu,\nu}(z) = \frac{\|k_{\mu,\nu}\|^2}{\sigma^2} e^{-\frac{\|k_{\mu,\nu}\|^2 \|z\|^2}{2\sigma^2}} [e^{ik_{\mu,\nu}z} - e^{-\frac{\sigma^2}{2}}], \text{ где}$$

$z=(x,y)$  - координаты точки,

$\|\cdot\|$  - норма вектора,

$k_{\mu,\nu} = k_{\nu} e^{i\phi_{\mu}}$ , где

$$k_{\nu} = k_{max}/f^{\nu}, \phi_{\nu} = \frac{\pi\nu}{8},$$

$k_{max}$ - максимальная частота,  $f$  — пространственный множитель между ядрами в частотной области.

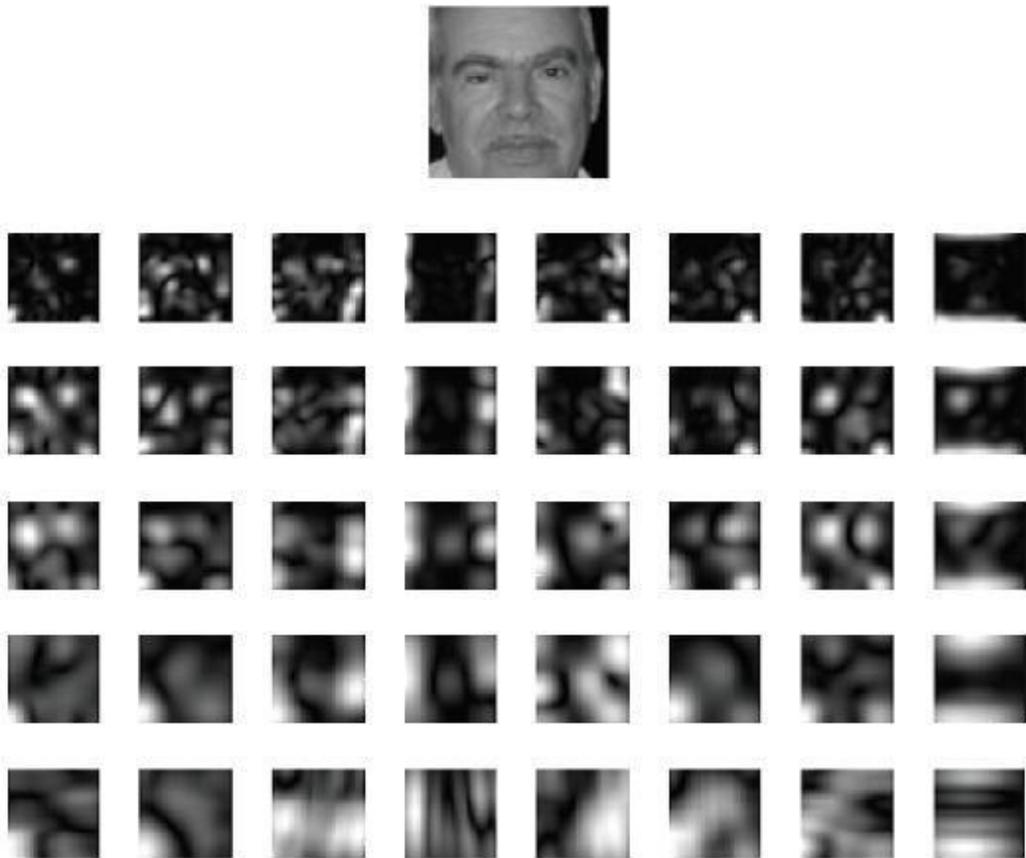


Рис.4: Выходы операции свертки для изображения

## Классификация

На последнем этапе работы рассматриваемых систем распознавания эмоций извлеченные признаки подаются на вход классификатору. Для решения задачи

классификации могут использоваться такие подходы как: метод опорных векторов, случайный лес, нейронные сети и другие.

## 1.2 Существующие решения задачи распознавания эмоций FaceReader

Разработка нидерландской компании Noldus Information Technology [12]. Программа способна классифицировать лицевые экспрессии по 6 базовым эмоциям, а также определять нейтральное выражение лица. Помимо этого, FaceReader способен определять пол и возраст людей.

Программа использует геометрические признаки изображения для классификации, в частности, используется метод Active Appearance Model, создающий модель лица человека по ключевым точкам с учетом деталей поверхности. Решение задачи классификации осуществляется нейронной сетью.



Рис.5: Иллюстрация работы программы FaceReader

Достоинства:

- Система способна анализировать как статические кадры, так и видеофайлы.
- Для распознавания эмоции наклон лица в плоскости может быть любым, если система сможет его обнаружить.
- Программа хорошо визуализирована: присутствуют диаграммы, гistogramмы, проценты выражаемых эмоций.

Недостатки:

- Система не натренирована для распознавания эмоций детей младше пяти лет.
- Точность распознавания снижается, если пользователь в очках.
- Программа не детектирует лица, повернутые в профиль.

## **Project Oxford**

Компания Microsoft разработала проект под названием Project Oxford [13]. Данный проект представляет собой набор готовых REST API, реализующих алгоритмы компьютерного зрения. Помимо сервиса распознавания лиц, речи и текста, данный проект включает в себя также сервис по распознаванию эмоций. Программа способна различать следующие типы эмоций: счастье, страх, грусть, отвращение, гнев, презрение, удивление, а также нейтральное выражение лица. Разработчики могут использовать данные сервисы внутри своих приложений.

Достоинства:

- Возможность использования внутри различных приложений.
- Демонстрация процентов выражаемых эмоций.

Недостатки:

- Программа не может распознать лица с наклоном более 40 градусов.
- Распознавание эмоций ведется только на статичных изображениях.

## Глава 2. Основные понятия и методы

### 2.1 Искусственная нейронная сеть

Искусственная нейронная сеть - математическая модель, имитирующая работу нервной системы человека, в особенности ее способность к самообучению и исправлению ошибок.

Нервная система человека состоит из нейронов, соединенных нервными волокнами. Благодаря нервным волокнам между нейронами передаются электрические импульсы. Нейрон получает возбуждение через специальные отростки - дендриты, а передает полученный сигнал с помощью другого отростка, называемого аксоном. Место контакта между двумя нейронами называется синапсом. Синапсы могут ослаблять или усиливать проходящий сигнал. Главная функция нейрона - передача возбуждения с дендритов на аксон. Сигналы же поступают из разных дендритов и оказывают влияние на итоговое значение сигнала в аксоне.

Нейрон отвечает на возбуждение, если суммарное значение сигнала превысит некоторую границу, иначе - сигнал не будет выдан на аксон. Искусственные нейроны имитируют такое поведение биологических нейронов.

Искусственный нейрон представляет собой некую функцию от линейной комбинации всех своих входных сигналов. Для рассмотрения модели искусственного нейрона необходимо ввести понятие веса. Синапс является звеном между двумя нейронами. Синапсы имеют параметр — вес. Благодаря ему при передаче сигнала от одного нейрона к другому меняется значение входной информации. Информация, поступившая от синапса с наибольшим весом, будет доминирующей. Именно благодаря этим весам, входная информация обрабатывается и превращается в результат. Именно веса обеспечивают обработку входной информации.

Рассмотрим модель искусственного нейрона, представленную на рисунке.

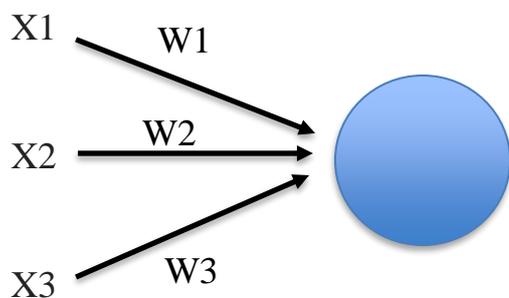


Рис.6: Представление искусственного нейрона

На вход нейрону поступают сигналы  $x_1, x_2, x_3$ . Каждый сигнал имеет свой вес. Входные значения умножаются на значения соответствующих весов и суммируются. Таким образом, нейрон имеет следующие входные данные:  $w_1x_1 + w_2x_2 + w_3x_3$ . Данная сумма представляет собой общее возбуждение нейрона. Для получения выходных данных нейрона необходимо передать полученные значения в функцию активации.

Функция активации, в зависимости от задачи, может иметь разные виды. К примеру, в качестве функции активации может использовать линейная функция, сигмоида или же гиперболический тангенс.

## 2.2 Сверточная нейронная сеть

Сверточные нейронные сети - особая архитектура нейронных сетей, основным предназначением которой является эффективное распознавание образов. Сверточная нейронная сеть является неким прототипом зрительной коры головного мозга. В ходе эксперимента в 1962 году Хьюбел и Визел установили, что отдельные нейроны активируются только при визуальном восприятии границ конкретной ориентации. Некоторые нейроны реагировали, когда воспринимали горизонтальные грани, некоторые - диагональные или вертикальные. Ученые пришли к выводу, что все эти нейроны сконцентрированы в виде стержневой архитектуры и все вместе формируют визуальное восприятие человека. Идея, основанная на нейронах зрительной коры, отыскивающих специальные характеристики легла в основу сверточных нейронных сетей.

Сверточная нейронная сеть состоит из различных видов слоев: сверточных, субдискретизирующих и слоев обычного персептрона. Изображение, проходя через чередующиеся сверточные и субдискретизирующие слои преобразуется во входной вектор признаков для слоев персептрона. Рассмотрим слои сверточной нейронной сети более подробно.

### **Сверточный слой**

Первый слой в сверточной нейронной сети должен быть сверточным. Название данного слоя, как и самой архитектуры происходит от операции свертки, применяемой в данном слое. На вход данному слою поступает изображение - матрица пикселей, с размерами  $h \times w \times c$ , где  $h \times w$  - размеры изображения,  $c$  - количество каналов.

Сверточный слой - набор карт признаков, каждая из которых обладает собственным фильтром. Фильтр (ядро свертки, матрица весов) производит операцию свертки путем умножения каждого фрагмента исходной матрицы на матрицу фильтра поэлементно, результаты суммируются (рис.7). Полученное значение передается функции активации, представляющей собой некоторую нелинейную функцию. Результат ее работы записывается в позицию выходной матрицы, соответствующей обрабатываемому в данный момент фрагменту. Каждое ядро выделяет определенные признаки на изображении. Размер у всех карт сверточного слоя – одинаковы и вычисляются по формуле:

$(w, h) = (prW - kW + 1, prH - kH + 1)$ , где

$prW, prH$  - размеры предыдущей карты,

$kW, kH$  - размеры фильтра.

В отличие от обычной полносвязной сети, в которой каждый нейрон слоя связан со всеми нейронами предыдущего слоя и при этом каждая связь имеет свой вес, в сверточной нейронной сети, благодаря наличию операции свертки для различных нейронов выходного слоя используется одна и та же матрица весов небольшого размера.

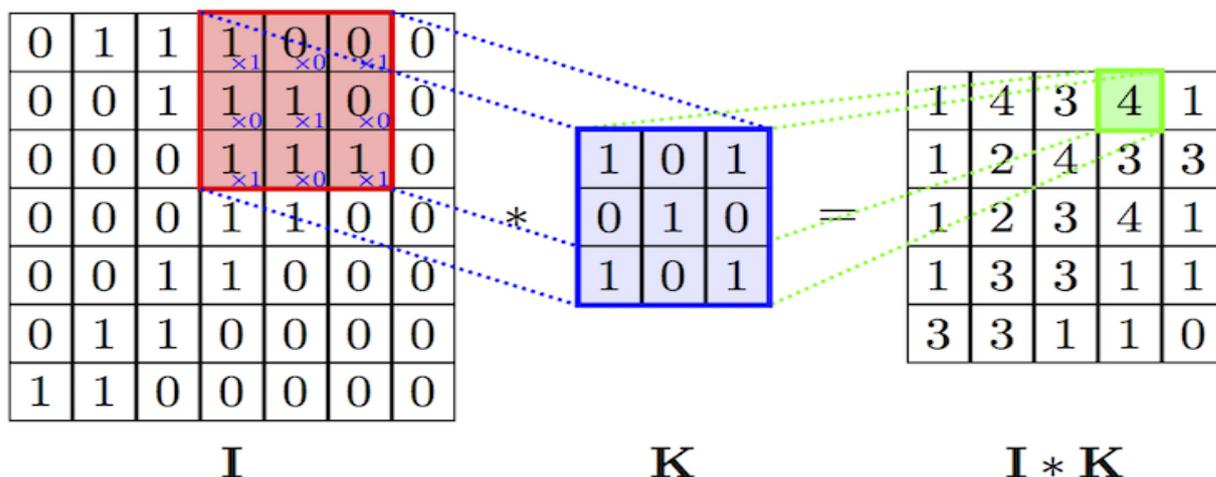


Рис.7: Иллюстрация операции свертки

### Субдискретизирующий слой (слой пулинга)

Основная задача данного слоя - снижение размерности карт, поступивших с предыдущего слоя для уменьшения количества параметров для следующего слоя. Это позволяет уменьшить вычислительную сложность, также помогает избежать переобучения. Обычно данный слой используется между сверточными слоями.

В данном слое также применяется скользящий по карте предыдущего слоя фильтр, однако вместо умножения на значения весов матрицы, к содержимому сканируемого фрагмента применяется некая функция. Одной из часто применяемых функций является функция максимума.

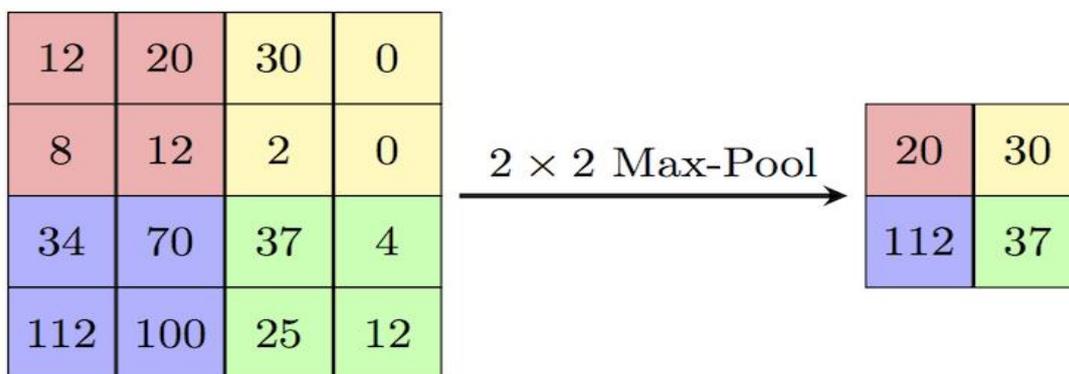


Рис.8: Иллюстрация операции пулинга

### Полносвязный слой

Данные, полученные после прохождения изображения через несколько сверточных и субдискретизирующих слоев, передаются на вход обычному многослойному персептрону. Данный слой предназначен для классификации и получения итогового результата. Нейроны карт предыдущего слоя пулинга связаны с одним нейроном полносвязного слоя.

### **Обучение сверточной нейронной сети**

Обучение сверточной нейронной сети происходит с помощью метода обратного распространения ошибки.

На первом этапе, когда нейронная сеть еще не обучена, веса инициализируются случайным образом. Для обучения нейронной сети ей на вход подаются данные из тренировочного набора. В тренировочном наборе все данные являются размеченными. Нейронная сеть выдает результат для обучающего экземпляра. Полученный результат сравнивается с правильным ответом и вычисляется функция ошибки. Тренировка нейронной сети заключается в подборе весовых коэффициентов, минимизирующих функцию ошибки.

Для минимизации функции ошибки используется метод градиентного спуска, суть которого состоит в том, что для поиска минимума функции необходимо двигаться в направлении антиградиента.

Введем обозначения:

$E_k$ - функция ошибки,

$t_{k,j}$  - желаемый выход нейрона  $j$  для  $k$ -го примера

$y_{k,j}$  - фактический выход нейрона  $j$  для  $k$ -го примера

$w_{k,i,j}$  - вес связи между  $i$  и  $j$  нейронами для  $k$ -го примера

Неактивированная сумма для каждого нейрона записывается в виде:

$$s_{k,j} = \sum_i w_{k,i,j} y_{k,i}$$

Выход  $j$ -го нейрона для  $k$ -го примера вычисляется по формуле:

$y_{k,j} = f_j(s_{k,j})$ , где  $f$ -функция активации  $j$ -го нейрона.

Градиент функции ошибки - вектор частных производных функции ошибки по весам сети.

$\Delta E = \left[ \frac{\partial E}{\partial w_1}, \dots, \frac{\partial E}{\partial w_n} \right]$ , где  $\Delta E$ -градиент функции ошибки,

$\frac{\partial E}{\partial w_k}$ - частная производная функции ошибки по k-му весу.

$$\frac{\partial E}{\partial w_k} = \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial s_j} \frac{\partial s_j}{\partial w_j}$$

На данном этапе значения ошибки определены явно только для нейронов выходного слоя. Для нейронов скрытого слоя нужно применить метод обратного распространения ошибки.

Смысл его состоит в том, что вычисление ошибок скрытых слоев происходит с использованием значений ошибок выходного слоя, таким образом, значения ошибки распространяются обратно по сети.

Ошибка i-го нейрона в скрытом слое вычисляется по формуле:

$$\delta_i = \frac{\partial y_i}{\partial s_i} \sum_j \delta_j w_{i,j}, \text{ где } \delta_j - \text{ошибка нейрона } j \text{ из следующего слоя}$$

Таким образом, алгоритм имеет следующий вид:

1. Прямое распространение сигнала по сети
2. Обратное распространение ошибки
3. Обновление значений весов

### **Расчет ошибки на субдискретизирующем слое**

В случае если слой пулинга идет перед полносвязным, то вычисление ошибки происходит также, как для нейроном скрытого слоя многослойного персептрона.

Рассмотрим случай, когда слой субдискретизирующий слой идет перед сверточным слоем. В таком случае расчет ошибки происходит с использованием операции обратной свертки.

Обратная свертка – способ для вычисления ошибок, заключающийся в повороте фильтра на 180 градусов и скользящем сканировании карты ошибок сверточного слоя с измененными краевыми эффектами.

### **Расчет ошибок в сверточном слое**

Расчет ошибок в сверточном слое производится за счет знаний о значениях ошибок в субдискретизирующем слое. Поскольку слой пулинга получается из сверточного путем выбора в каждом фрагменте нейрона, то ошибки для соответствующих нейронов просто копируются, а остальные нейроны получают нулевое значение ошибки.

## 2.3 Теоретический обзор используемых методов

### Метод прореживания

Переобучение является проблемой для глубоких нейронных сетей. Метод прореживания (анг. dropout), предложенный в статье [14] является способом решения данной проблемы.

Идея метода заключается в том, что вместо одной нейронной сети обучается ансамбль сетей, а полученные результаты усредняются. На очередном градиентном шаге изменяется структура сети: каждая нейрон может быть отброшен с некоторой вероятностью  $p$ . Для оставшихся в сети весов производится градиентный шаг, после него все отброшенные нейроны возвращаются в сеть. Поскольку отброшенные нейроны не влияют на процесс обучения в процессе алгоритма обратного распространения ошибки, исключение хотя бы одного нейрона фактически приводит к обучению новой сети.

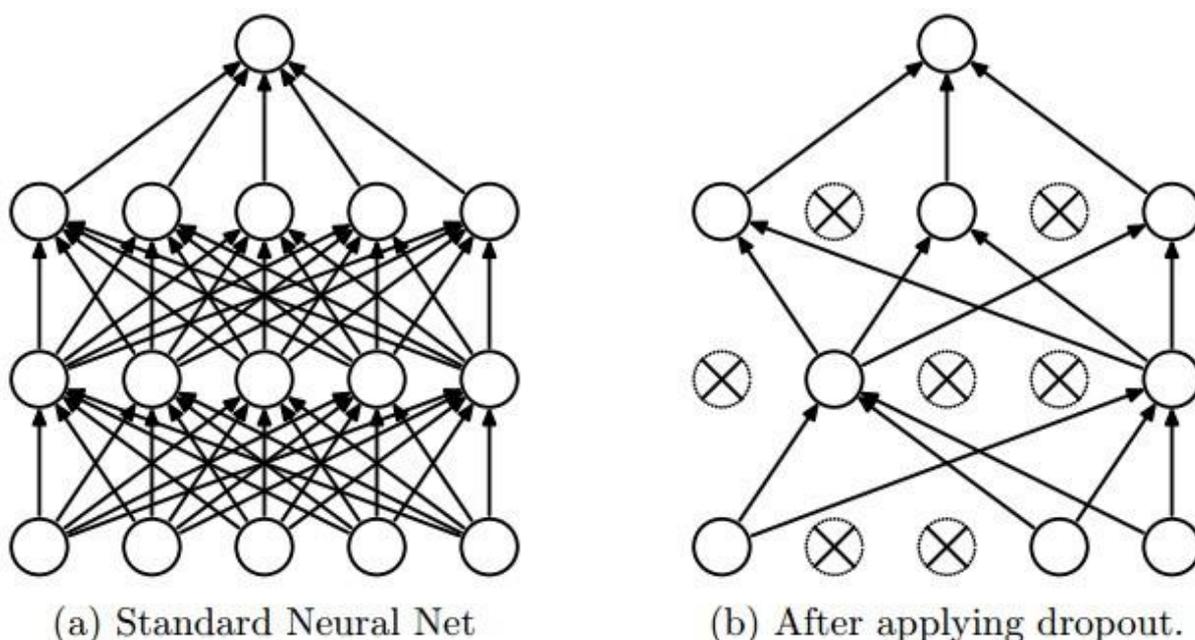


Рис.10: Иллюстрация метода прореживания

## L2 - регуляризация

Регуляризация - метод, суть которого заключается в добавлении некоторых ограничений к условию с целью предотвращения переобучения.

Одним из проявлений переобучения являются слишком большие весовые коэффициенты. Суть данного метода в том, что он штрафует нейронную сеть за слишком большие коэффициенты. Для каждого веса  $\omega$  к целевой функции добавляется слагаемое  $\frac{1}{2}\lambda\omega^2$ , где  $\lambda$  – коэффициент регуляризации. Таким образом, метод сильно штрафует веса, имеющие большие значения и оказывает слабое влияние на веса, имеющие умеренные значения.

## Батч-нормализация

Батч-нормализация - один из способов ускорения обучения нейронной сети, предложенный в статье [15]. Метод позволяет решить следующую проблему: по мере распространения по сети, даже нормализованный на входе сигнал на внутренних слоях может, исказиться по матожиданию и дисперсии, что приводит к несоответствиям между градиентами различных уровней.

С помощью батч-нормализации входные данные перед входом на слой нормализуются таким образом, чтобы матожидание было равно 0, а дисперсия 1 на всем батче.

$$B = x_1, \dots, x_k \text{ - батч,}$$

$$M_b = \frac{1}{k} \sum_{i=1}^k x_i \text{ - матожидание,}$$

$$\sigma^2 = \frac{1}{k} \sum_{i=1}^k (x_i - M_b)^2 \text{ - дисперсия,}$$

$$\hat{x}_i = \frac{x_i - M_b}{\sqrt{\sigma_i^2 + \epsilon}} \text{ - нормализованное значение,}$$

Подобная нормализация может изменить представление данных в слое. Для решения данной проблемы вводятся параметра сжатия и сдвига, которые подбираются в процессе обучения подобно весам:

$$y_i = \hat{x}_i \gamma + \beta$$

## Глобальная нормализация контраста

Метод, используемый для того, чтобы избежать разной контрастности в изображении. В ходе метода выполняется вычитание среднего значения всех пикселей изображения из значения каждого пикселя, а затем полученный результат делится на стандартное отклонение. Таким образом, вычисляется среднее значение каждого изображения, а затем выполняется масштабирование изображения таким образом, чтобы стандартное отклонение по пикселям было постоянным и равным  $s$ .

$$X'_{i,j,k} = s \frac{X_{i,j,k} - \bar{X}}{\max\left\{\varepsilon, \sqrt{\lambda + \frac{1}{3rc} \sum_{i=1}^r \sum_{j=1}^c \sum_{k=1}^3 (X_{i,j,k} - \bar{X})^2}\right\}}, \text{ где}$$

$X_{i,j,k}$  - значение пикселя,

$\bar{X}$  - среднее значение всех пикселей изображения,

$r, c$  - размеры изображения,

$s$  – новое значение контраста,

$\lambda$  - положительный параметр регуляризации, используемый для смещения стандартного отклонения,

$\varepsilon$  - минимальное значение стандартного отклонения.

Параметры  $\varepsilon, \lambda$  введены для того, чтобы для обработки изображений с очень низким контрастом. Деление на стандартное отклонение для таких изображений не даст ничего, кроме усиления шума.

### **Метод Виолы-Джонса**

Метод Виолы-Джонса [16] - алгоритм, использующийся для обнаружения объектов на изображении в режиме реального времени. Основными преимуществами метода является высокая точность и быстрая скорость работы.

Детектор Виолы-Джонса основан на интегральном представлении изображений, использовании примитивов Хаара, методе построения классификатора с использованием бустинга, и методе комбинирования классификаторов в каскадную структуру.

### **Интегральное представление изображений**

Для ускорения работы с данными в методе Виолы-Джонса используется интегральное представление изображений. Интегральное представление изображения - матрица, размерность которой совпадает с размерностью исходного изображения. Ее элементы вычисляются по формуле:

$$L(x,y) = \sum_{i=0,x}^{i \leq x, j=0,y} I(x,y), \text{ где}$$

$I(i,j)$  — яркость пикселя изображения.

То есть каждый элемент интегрального представления изображения  $L[x,y]$  содержит в себе сумму пикселей изображения в прямоугольнике от  $(0,0)$  до  $(x,y)$ . Время расчета интегрального изображения линейно, пропорционально числу пикселей в исходном изображении.

Для интегрального представления справедлива следующая формула:

$$L(x,y) = I(x,y) - L(x-1,y-1) + L(x,y-1) + L(x-1,y).$$

Главное преимущество интегрального представления изображения - возможность быстро вычислить сумму пикселей произвольного прямоугольника.

### Признаки Хаара

Признаки(примитивы) Хаара — признаки изображения, используемые в распознавании образов. Вычисляемое значение признака  $F = X - Y$ , где  $X$  – сумма яркостей пикселей закрываемых светлой областью примитива, а  $Y$  – сумма яркостей пикселей, закрываемых темной областью примитива. Для вычисления значений исходные изображения представляются в интегральном виде.

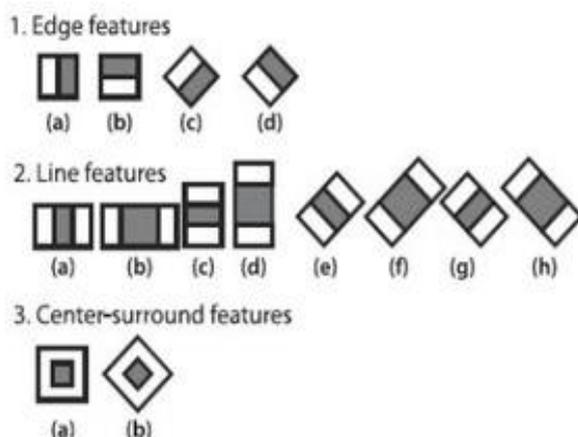


Рис.10: Признаки Хаара

Примитивы Хаара дают точечное значение перепада яркости по осям X и Y соответственно.

### **Принцип сканирующего окна**

Для вычисления признаков на изображении используется принцип сканирующего окна. Выбирается небольшое окно сканирования. Внутри окна вычисляется примерно 200000 различных признаков за счет изменения их масштаба и положения внутри окна. Затем окно передвигается по изображению с шагом в 1 ячейку. После нахождения всех признаков меняется масштаб сканирующего окна, и затем снова происходит расчет признаков. Найденные признаки подаются на вход классификатору, который выносит вердикт о принадлежности объекта сканирующему окну.

### **AdaBoost и каскад классификаторов**

Для принятия решения о наличии или отсутствии объекта на изображении, в методе Виолы-Джонса используется так называемый rejection cascade узлов, где каждый узел - это классификатор, при этом данные классификаторы построены с помощью алгоритма AdaBoost - частного случая технологии бустинга. На начальных этапах обработки изображение проходит через самые простые классификаторы, которые отбрасывают те части изображения, которые явно не являются интересующими объектами. Чем дальше по каскаду проходит изображение, тем более сложные классификаторы применяются к нему. При прохождении через весь каскад изображение определяется как объект.

## Глава 3. Решение поставленной задачи

### 3.1 Используемые программные средства

- **Язык программирования Python.** Выбор данного языка обусловлен тем, что он позволяет использовать библиотеку компьютерного зрения OpenCV, различные библиотеки глубокого обучения, пакеты для предварительной обработки данных.
- **OpenCV** – библиотека [17], используемая для решения задач компьютерного зрения [18]. Содержит различные средства для обработки изображений, а также готовые алгоритмы компьютерного зрения.
- **Keras** - нейросетевая библиотека. Включает в себя различные реализации широко применяемых строительных блоков нейронных сетей и множество средств для работы с изображениями.
- **PyQt** - модуль для создания графических интерфейсов приложений на Python на базе платформы Qt.
- **Qt Designer** — среда для разработки графических интерфейсов программ, использующих Qt.
- **Google Colaboratory** - облачный сервис для исследований в области задач машинного обучения, позволяющий бесплатно получить доступ к GPU (NVidia Tesla K80).

### 3.2 Обучающая выборка

Для обучения нейронной сети применялся тренировочный набор данных, состоящий из снимков из трех наборов данных (датасетов): Fer2013, CK+, Japanese Female Facial Expression (JAFFE) database.

#### **Fer 2013**

Датасет [19] состоит из 28709 изображений с размерами 48x48 в градациях серого. В наборе представлены снимки людей различных возрастных категорий, с различными положениями головы, с различной степенью освещенно-

сти; на некоторых изображениях часть лица закрыта чем-либо. В наборе представлены 6 базовых типов эмоций, а также нейтральное выражение лица. Изображения с нейтральным выражением лица не использовались в тренировочном наборе данных, поскольку в рамках данной работы рассматриваются только 6 базовых эмоций.

### **СК+**

Датасет [20] включает в себя 593 последовательности изображений размером 640\*490 в градациях серого - последовательностей кадров видео, на которых люди изображают заданную эмоцию. На видео демонстрировались 7 типов эмоций: грусть, удивление, счастье, страх, гнев, отвращение, презрение. Однако цепочки с последним типом эмоции не рассматривались, поскольку презрение не входит в типы эмоций, рассматриваемых в данной работе. Из данных последовательностей извлекаются последние кадры: последний содержит максимальное выражение эмоции, демонстрируемой на видео.

### **Japanese Female Facial Expression**

Датасет [21] состоит из 213 изображений в градациях серого с размерами 256x256, на которых 10 женщин изображают одну из 6 базовых эмоций или демонстрируют нейтральное выражение лица. Снимки с нейтральным выражением не добавлялись в обучающую выборку.

## **3.3 Предобработка данных для обучения нейронной сети.**

### **Обнаружение лиц и обрезка изображений**

Изображения, на которых обучается нейронная сеть должны быть как можно более чистыми, поэтому датасет очищается от снимков, на которых лицо не может быть найдено. Изображения подаются на вход каскаду Хаара, реализованному в библиотеке компьютерного зрения OpenCV.

После фильтрации в выборке остаются только те снимки, лица на которых могут быть детектированы с помощью метода Виолы-Джонса (данный метод также используется для детектирования лиц при работе данной системы).

Оставшиеся же изображения обрезаются, и на них остается лишь область лица человека.

### **Дальнейшая фильтрация**

Разрабатываемая система предполагает, что взаимодействующий с ней пользователь находится в условиях с нормальной освещенностью, не закрывает лицо посторонними предметами и смотрит в камеру. Поэтому для повышения точности работы сети, производится ручная фильтрация выборки от снимков с нестандартными положениями головы, а также изображений, на которых лицо закрыто руками, очками и так далее. Кроме того, удаляются искаженные изображения и размытые снимки.

### **Масштабирование изображений**

В ходе экспериментов было установлено, что оптимальным для изображений обучающей выборки является размер 64x64. Такой размер позволяет нейронной сети достаточно быстро обучаться и работать, при этом изображения не искажаются при масштабировании до рассматриваемых пропорций, что положительно отражается на точности результатов работы сети.

### **Глобальная нормализация контраста**

В статье [22] рассматриваются способы повышения точности работы сверточной нейронной сети в задаче распознавания эмоций с помощью предварительной обработки данных. В исследовании сравниваются различные способы нормализации изображений и делается вывод, что наибольшее увеличение точности обеспечивает такой тип нормализации, как глобальная нормализация контраста. Подробности данного метода детально описаны в предыдущей главе. В ходе данной работы была написана программная реализация этого метода, и с его помощью были обработаны все изображения обучающей выборки.

Таким образом, обработка отдельно взятого изображения выглядит следующим образом: Рис.11.



Рис.11: Схема предварительной обработки изображения

### Проверка выборки на сбалансированность, сравнение распределений обучающей и тестовой выборок, нормализация данных

Большинство алгоритмов машинного обучения работают лучше всего, когда количество экземпляров каждого класса примерно одинаково. В ситуации, когда количество экземпляров одного класса намного превышает другой, могут возникнуть проблемы с точностью нейронной сети. Однако, после фильтрации не было выявлено классов, разница между количеством снимков в которых сильно велика.

Общее количество снимков в обучающей выборке – 6085.

После проверки на сбалансированность выполняется небольшая корректировка обучающей и тестовой выборок таким образом, чтобы коэффициент корреляции Спирмена для данных выборок был равен 1.

Обучающая и тестовые выборки имеют следующие распределения: Таблица 1.

Эмоция	Обучающая выборка	Тестовая выборка
Счастье	1052	170
Злость	1041	169
Отвращение	1033	161

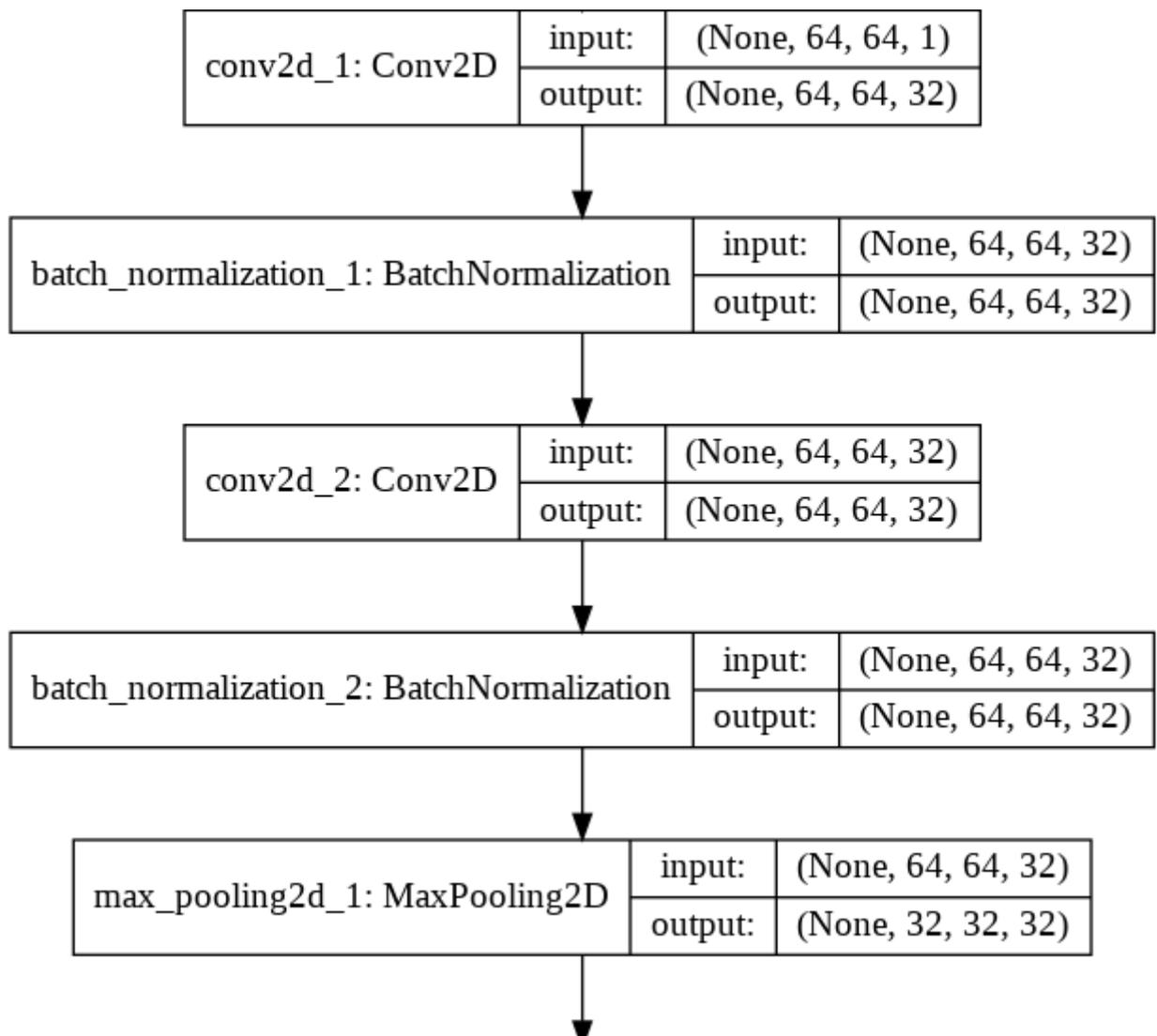
Удивление	991	148
Грусть	985	146
Испуг	983	144

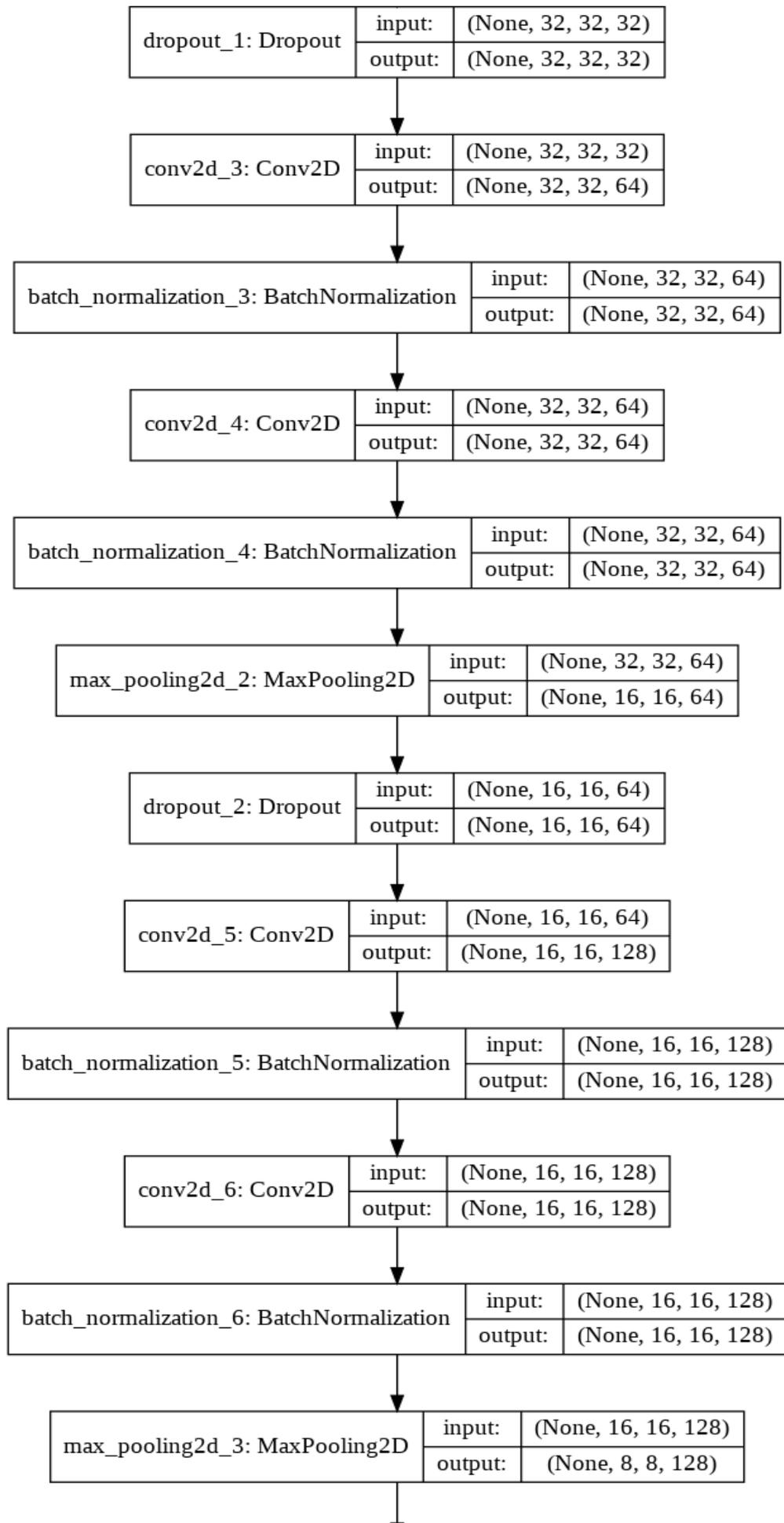
Таблица 1: Распределение обучающей и тестовой выборки

На последнем этапе предобработки все изображения нормализуются, чтобы значение каждого пикселя было в отрезке [0;1].

### 3.4 Создание и обучение нейронной сети

Используемая в работе нейросеть имеет следующую архитектуру:





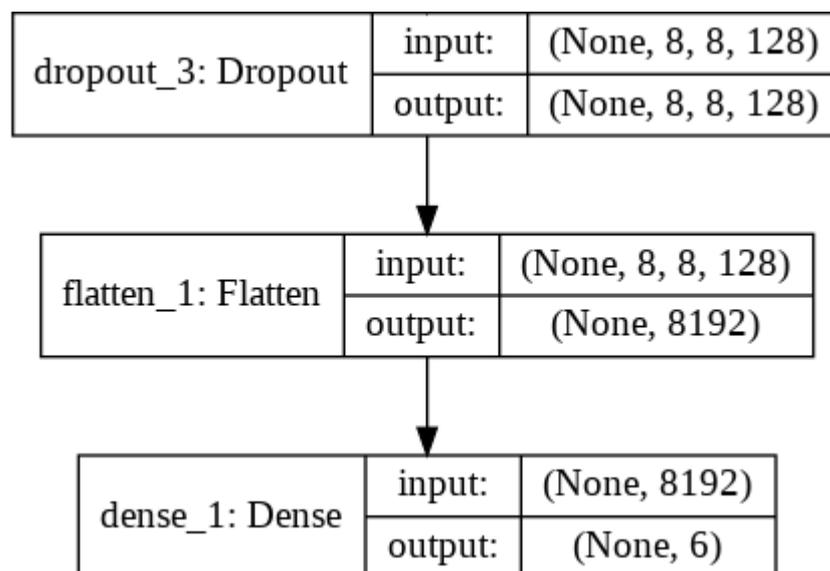


Рис.12: Архитектура нейронной сети

Особенности слоев, представленных в данной архитектуре подробно рассмотрены в предыдущей главе. Как видно из представленной схемы, сеть состоит из 6 сверточных слоев - Conv2D (входные изображения рассматриваются как двумерные матрицы). После 2-го, 4-го и 6-го сверточных слоев расположены субдискретизирующие слои (MaxPooling2D) с фильтром размера 2x2. Перед полносвязным слоем (Dense) находится слой выравнивания (Flatten), служащий соединительным узлом между слоями. Помимо структуры сети, на схеме представлены входные и выходные данные для каждого слоя. При обучении сети использовались такие методы как: метод прореживания, нормализация батчей и L2-регуляризация, также описанные выше.

Для обучения сверточной нейронной сети используются следующие гиперпараметры:

batch-size = 64 - количество изображений из обучающей выборки, обрабатываемых одновременно за одну итерацию метода градиентного спуска,

kernel-size = 3 - размер фильтра,

num-epochs = 100 - количество итераций по всей обучающей выборке,

drop\_prob - вероятность исключения нейрона в методе прореживания:

drop\_prob\_1 = 0.8,

drop\_prob\_2 = 0.7,

drop\_prob\_3 = 0.6.

conv\_depth - количество фильтров в сверточных слоях, слои имеют различные значения данного параметра:

conv\_depth\_1 = 32,

conv\_depth\_2 = 32,

conv\_depth\_3 = 64,

conv\_depth\_4 = 64,

conv\_depth\_5 = 128,

conv\_depth\_6 = 128.

pool\_size = 2 -размер подвыборки в слоях пулинга.

В сверточных слоях используется активационная функция ReLu:

$f(x)=\max(0,x)$ .

Последний слой имеет функцию активации softmax:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{k=1}^K e^{z_k}}, \text{ где}$$

Z – исходный вектор,

K – размерность исходного вектора.

Благодаря данной функции сеть вернет распределение вероятностей по классам.

В качестве функции потерь выступает категориальная перекрестная энтропия:  $-\sum_{i=1}^n (x_i \log(y_i))$ . Данная функция определяет расстояние между распределениями вероятности на выходе нейронной сети и фактическим распределением меток.

### 3.5 Объектная модель приложения

Рассмотрим основные классы приложения:

- App – класс окна для создания снимков и загрузки фотографий.
- Train – класс окна для выбора эмоций и вывода результата.
- PhotoHandler – класс, внутри которого происходит анализ снимка и делается вывод о преобладающей на изображении эмоции.
- FaceDetector – класс, предназначенный для поиска лиц на изображениях.

- Layout – класс для получения и отображения данных видеопотока с веб-камеры.

При запуске приложения создается экземпляр класса App, в конструкторе которого вызывается метод `setUpUi`, отвечающий за отрисовку элементов интерфейса первого окна. Создаются экземпляры классов Layout, FaceDetector, PhotoHandler. С помощью механизма слотов и сигналов новые кадры, полученные из видеопотока внутри класса Layout, передаются методу `image_data_slot` класса FaceDetector и обрабатываются. На изображениях локализуются лица, и происходит отрисовка снимков. При нажатии на кнопку для создания снимка вызывается метод `make_photo` класса PhotoHandler, используя в качестве аргумента текущий кадр, хранящийся в созданном экземпляре класса Layout. В случае использования загрузки снимка с компьютера также вызывается метод `make_photo` с полученным изображением в качестве аргумента. При вызове данного метода создается экземпляр класса Train, и появляется второе окно программы. После выбора нужной эмоции или режима автоматического определения эмоции вызывается метод `check_emotion` класса PhotoHandler, внутри которого происходит определения преобладающей эмоции на основе построенной модели нейронной сети.

### **3.6 Разработка пользовательского интерфейса программы**

Пользовательский интерфейс программы реализован с использованием библиотеки PyQt и среды Qt Designer. В среде Qt Designer было создан макет интерфейса программы, а добавление интерактивности выполнялось с использованием PyQt.

Интерфейс программы состоит из двух окон. Первое окно предназначено для создания снимков, которые в дальнейшем передаются нейронной сети для анализа. Пользователь, изображая ту или иную эмоцию, может сделать снимок с помощью веб-камеры. При использовании веб-камеры программа автоматически детектирует лицо человека в видеопотоке, применяя каскад Хаара, входящий в библиотеку компьютерного зрения OpenCV. При желании можно загрузить изображение с компьютера. Окно также содержит примеры того, как можно демонстрировать ту или иную эмоцию.

После того, как пользователь делает снимок или загружает фотографию с компьютера, открывается второе окно программы (рис.13). Человеку показывается полученное изображение и предлагается выбрать, какую эмоцию он хотел изобразить. Возможно также выбрать вариант определения эмоции, без самостоятельного выбора класса. В случае выбора определенной эмоции программа, основываясь на результате работы нейронной сети, делает вывод, соответствует ли выбранная пользователем эмоция той, которую выявила на данном снимке нейронная сеть. В случае же, если заявленная эмоция является второй наиболее вероятной эмоцией, программа выносит вердикт, что эмоция близка к выбранной. После принятия решения пользователю показывается результат работы. В случае, если заявленная эмоция расходится с той, что обнаружила нейросеть, приложение сообщает, какая, по его мнению, эмоция преобладает на снимке. При выборе режима определения эмоции программой, программа показывает пользователю решение нейронной сети о преобладающей на снимке эмоции.

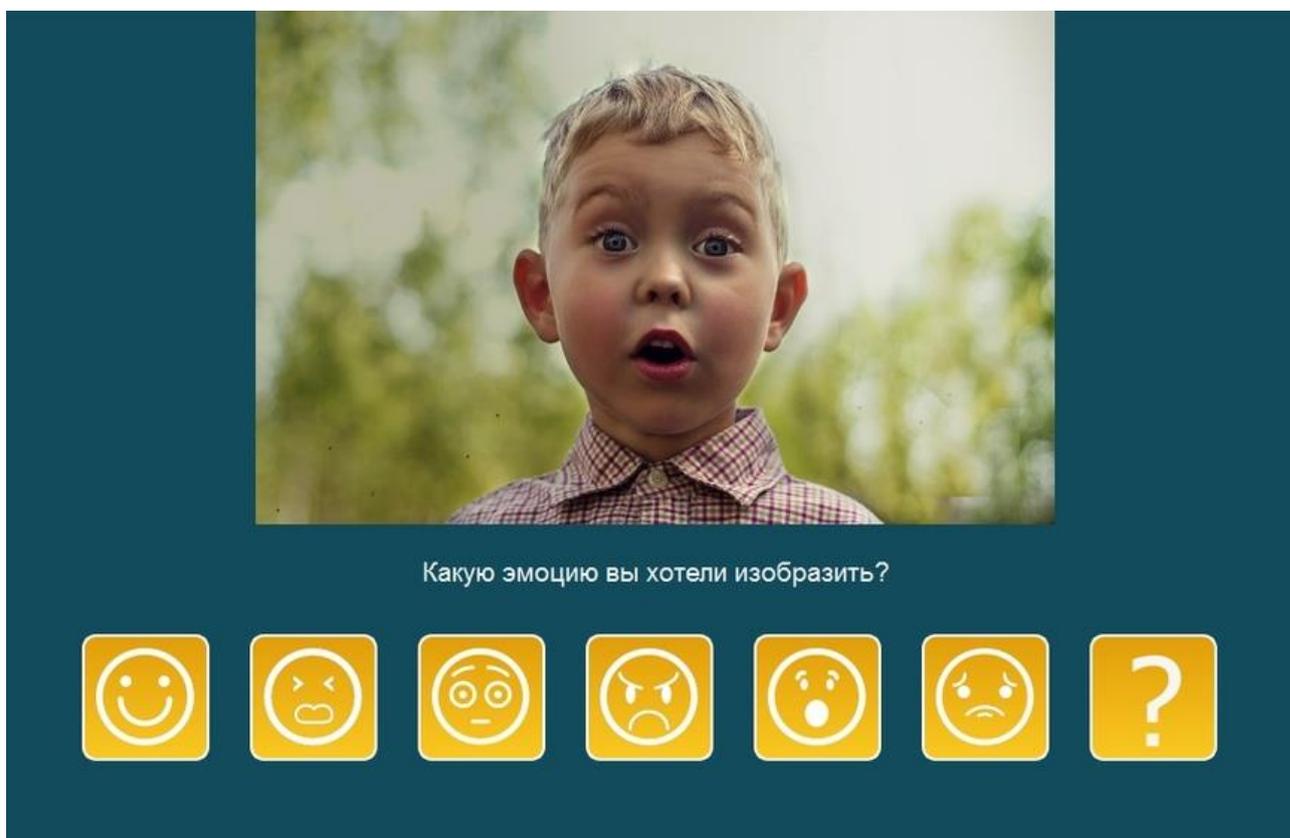


Рис.13: Окно выбора эмоции

Таким образом, разработанная программа способна распознавать эмоции на снимках, позволяет пользователю тренировать выражение эмоциональных реакций, а также предоставляет удобный интерфейс для работы с приложением.

## Глава 4. Тестирование результатов

На рисунке ниже представлены пример результатов работы программы.

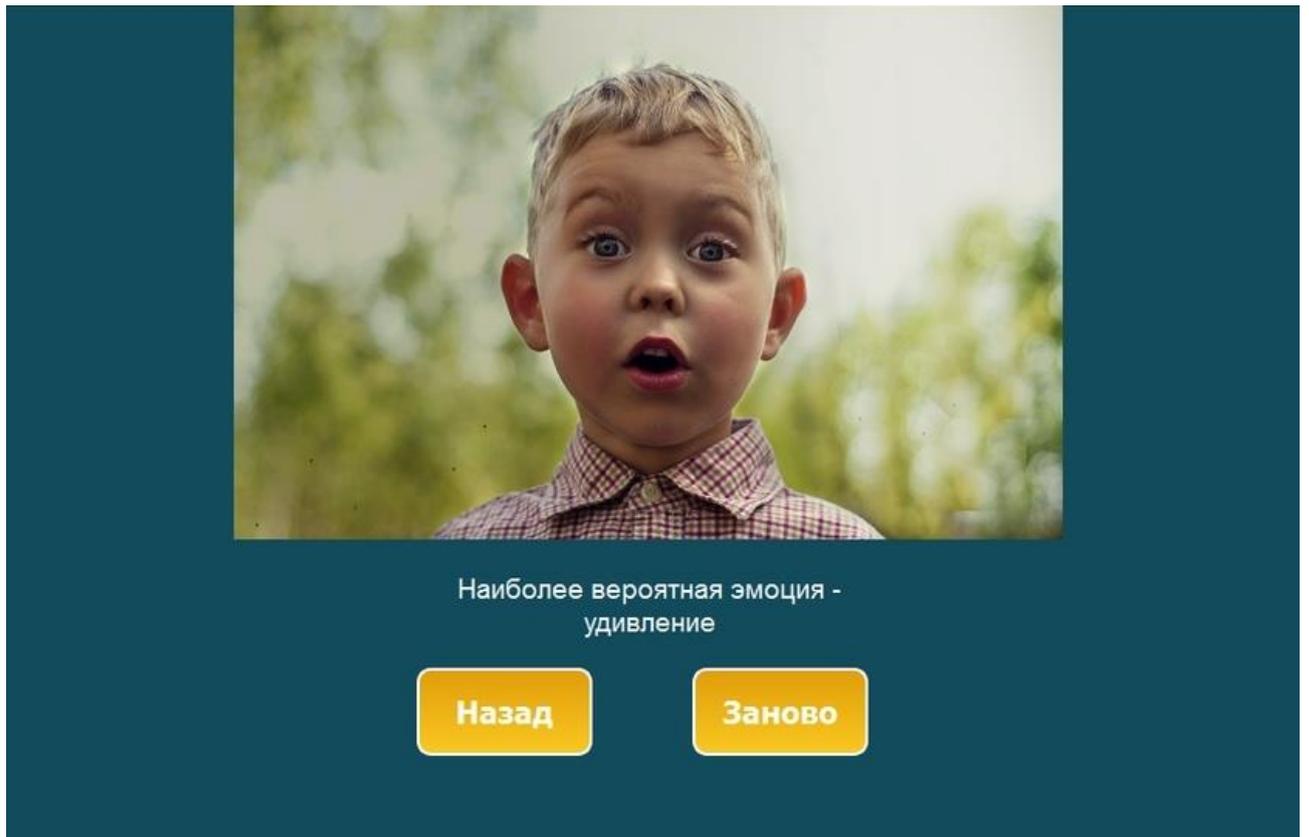


Рис.14: Пример работы программы

Общее количество изображений в тестовом наборе данных - 938. Распределение тестовой выборки представлено в таблице 1 предыдущей главы.

Общая точность работы нейронной сети при тестировании составила 82%. Рассмотрим матрицу неточностей для полученных результатов, а также ее нормированную версию (таблицы 2, 3).

	Злость	Отвращение	Страх	Радость	Грусть	Удивление
Злость	144	15	7	0	3	0
Отвращение	20	115	18	3	4	1
Страх	5	12	94	2	5	26
Радость	1	0	1	166	0	2

Грусть	6	4	19	1	112	4
Удивление	0	0	9	1	0	138

Таблица 2: Матрица неточностей результатов работы + метки классов

	Злость	Отвращение	Страх	Радость	Грусть	Удивление
Злость	<b>0.852</b>	0.089	0.041	0	0.018	0
Отвращение	0.124	<b>0.714</b>	0.112	0.019	0.025	0.006
Страх	0.035	0.083	<b>0.653</b>	0.014	0.035	0.18
Радость	0.006	0	0.006	<b>0.976</b>	0	0.012
Грусть	0.041	0.027	0.13	0.007	<b>0.767</b>	0.027
Удивление	0	0	0.06	0.007	0	<b>0.932</b>

Таблица 3: Нормированная матрица неточностей результатов работы + метки классов

Из представленных данных видно, что наилучшие результаты сеть демонстрирует при распознавании радости, удивления, злости - точность на рассматриваемых классах составляет 98, 93, 85 процентов соответственно. Несколько меньшая точность обнаруживается на изображениях с метками грусть, отвращение, страх. Результаты распознавания данных эмоций, следующие: 77, 71 и 65 процентов соответственно.

Полученные результаты могут быть объяснены особенностями рассматриваемых эмоций. Например, 18% изображений класса страх классифицированы как “удивление”. Это объяснимо тем, что рассматриваемые эмоции имеют черты в некоторых случаях: поднятые брови, открытый рот. Сети не всегда удается уловить мельчайшие особенности эмоции “страх”, отличающие ее от “удивления”, несмотря на то что для человека данная задача в большинстве случаев не составляет большого труда. Подобную ситуацию можно наблюдать также на паре эмоций “отвращение” и “злость”.

Стоит также отметить, что зачастую при неправильной классификации фактическая метка находится на втором месте по вероятности в выходном векторе. А поскольку разработанная система позволяет учитывать данную особенность, точность ее работы в случае выбора режима тренировки эмоций несколько возрастает и составляет 91%.

## **Выводы**

В ходе выполнения работы были решены следующие задачи:

- Проведен теоретический анализ исследуемой области.
- Сформирована обучающая выборка для тренировки нейронной сети, и осуществлена ее предобработка.
- Создана и обучена сверточная нейронная сеть для решения задачи автоматического распознавания эмоций.
- Выбран и применен метод для детектирования лиц на изображениях.
- Разработан интерфейс для удобного пользования программой.
- Полученная программа была протестирована и показала приемлемый результат работы.

## Заключение

Задача автоматического распознавания эмоций является актуальной в наши дни, поскольку имеет широкое применение в различных сферах человеческой деятельности. В рамках данной работы рассмотрены и изучены основные подходы к решению данной задачи. Рассмотрены основные этапы предварительной обработки изображений и подготовки обучающей выборки. Реализована система распознавания и тренировки эмоций на основе сверточной нейронной сети, а также разработан интерфейс для удобного взаимодействия с ней.

Разработанная система имеет достаточно высокий уровень точности, однако может быть улучшена с помощью увеличения объема обучающей выборки и повышения ее разнообразия. Кроме того, возможно совершенствование данной программы с целью ее эффективной работы на изображениях с нестандартными положениями головы, а также со снимками, где видна лишь часть лица.

В дальнейшем планируется расширение реализованной системы таким образом, чтобы распознавание эмоций могло осуществляться не только по статическим кадрам, но и по видеофрагментам, с учетом всей последовательности кадров. Разработанная в контексте данной работы программа может стать фундаментом подобной системы.

## Список литературы

1. Большая медицинская энциклопедия под редакцией Петровского Б.В., 3-е издание [Электронный ресурс]: URL: <https://xn--90aw5c.xn--clavg/> (дата обращения: 14.04.2019).
2. С. Darwin. The Expression of the Emotions in Man and Animals. With an introduction, afterword, and commentaries by Paul Ekman. 3rd ed. Oxford University Press, 1998.
3. P. Ekman, Universals and cultural differences in facial expressions of emotion. Nebraska, Lincoln University of Nebraska Press, 1971.
4. Y. Chen, H. Wu, T. Wang, Y. Wang. A Comparison of Methods of Facial Expression Recognition. Proceedings of the 1st WRC Symposium on Advanced Robotics and Automation 2018 Beijing, 2018. P. 261-262
5. O. Arriaga, P. Plöger, Matias Valdenegro. Real-time Convolutional Neural Networks for Emotion and Gender Classification. ArXiv, 2017.
6. D. Duncan, G. Shine, C. English. Facial Emotion Recognition in Real Time. 2016
7. I. Swathi, S.S. Gantayat. Facial Expression Recognition Based On Emotion For Visually Challenged People. ICICSS, 2018
8. Gil-Jin Jang, Jeong-Sik Park, Ahra Jo, Ji-Hwan Kim. Facial Emotion Recognition Using Active Shape Models and Statistical Pattern Recognizers. Ninth International Conference on Broadband and Wireless Computing, Communication and Applications, 2014.
9. Ekman, P., Friesen, W. V., & Hager, J. C. Facial Action Coding System. Manual and Investigator's Guide, Salt Lake City, UT: Research Nexus, 2002.
10. C. Shan, S. Gong, and P. W. McOwan, Facial expression recognition based on local binary patterns: A comprehensive study, Image and Vision Computing, 2009. P. 803 – 816
11. N. Mehta, S. Jadhav. Facial Emotion recognition using Log Gabor filter and PCA. International Conference on Computing Communication Control and automation, 2016.

- 12.** FaceReader [Электронный ресурс]: URL: <https://www.noldus.com/human-behavior-research/products/facereader> (дата обращения 02.04.2019).
- 13.** Project Oxford [Электронный ресурс]: URL: <https://azure.microsoft.com/ru-ru/services/cognitive-services/> (дата обращения 02.04.2019).
- 14.** N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *Journal of Machine Learning Research*, 2014.
- 15.** S. Ioffe, C. Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning*, pages, 2015. P. 448–456
- 16.** P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition*, 2001. P. 511–518
- 17.** OpenCV documentation [Электронный ресурс]: URL: <https://docs.opencv.org/> (дата обращения 15.01.2019).
- 18.** Gregori E. Introduction To Computer Vision Using OpenCV. *Embedded Systems Conference in San Jose*, 2012.
- 19.** База фотографий fer2013 [Электронный ресурс]: URL: <https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data> (дата обращения 27.01.2019).
- 20.** Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. The Extended Cohn-Kanade Dataset (CK+): A complete expression dataset for action unit and emotion-specified expression. *Proceedings of the Third International Workshop on CVPR for Human Communicative Behavior Analysis (CVPR4HB 2010)*, San Francisco, USA, 2010, P. 94-101
- 21.** M. J. Lyons, S. Akemastu, M. Kamachi, J. Gyoba. Coding Facial Expressions with Gabor Wavelets, *3rd IEEE International Conference on Automatic Face and Gesture Recognition*, 1998, P. 200-205

**22.** D. A. Pitaloka, A. Wulandari, T. Basaruddin, D. Y. Liliana. Enhancing CNN with Preprocessing Stage in Automatic Emotion Recognition. *Procedia Computer Science*, 2017.