

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий

(наименование института полностью)

Кафедра «Прикладная математика и информатика»

(наименование)

01.03.02 Прикладная математика и информатика

(код и наименование направления подготовки, специальности)

Системное программирование и компьютерные технологии

(направленность (профиль)/специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему: «Анализ цветового распределения и тональности текста на изображениях в социальной сети «ВКонтакте»»

Студент

Д.В. Тяпкин

(И.О. Фамилия)

(личная подпись)

Руководитель

к.тех.н, Т.Г. Султанов

(ученая степень, звание, И.О. Фамилия)

Консультант

К.А. Селиверстова

(ученая степень, звание, И.О. Фамилия)

Тольятти 2020

АННОТАЦИЯ

Тема бакалаврской работы: «Анализ цветового распределения и тональности текста на изображениях в социальной сети «ВКонтакте».

В данной бакалаврской работе исследуются практические аспекты реализации программного обеспечения для анализа цветового распределения и тональности текста на изображениях в социальной сети «ВКонтакте».

В бакалаврской работе проводится сбор изображений, рассматривается выявление требуемых для классификации изображений характеристик, выбор оптимальных методов для типизации, анализ полученных данных, использование анализа данных для оценивания аккаунта пользователя сети «ВКонтакте» с эмоциональной точки зрения, а также приводятся результаты разработки собственной программы на языке Python, позволяющей классифицировать фотографии по заданным параметрам.

Структура бакалаврской работы представлена введением, тремя главами, заключением, списком литературы.

Во введении описывается актуальность проводимого исследования, даётся краткая характеристика проделанной работы.

В первой главе рассматривается анализ изображений с эмоциональной точки зрения. Во второй главе описывается математический аппарат для анализа цветового распределения и тональности текста на изображениях в социальной сети «ВКонтакте». В третьей главе приведено описание разработанного программного обеспечения для анализа цветового распределения и тональности текста на изображениях в социальной сети «ВКонтакте» и рассматривается пример его использования. В заключении представлены выводы по проделанной работе.

В работе использовано 3 таблицы, 17 рисунков, список литературы содержит 20 литературных источников. Общий объем выпускной квалификационной работы составляет 56 страницы.

ABSTRACT

The theme of the bachelor's work: "Analysis of the color distribution and tonality of the text in the images on the social network "VKontakte".

In this bachelor's work, practical aspects of implementing software for analyzing the color distribution and tonality of text in images on the VKontakte social network are investigated.

In the bachelor's work, the images are collected, the identification of the characteristics required for image classification, the selection of optimal methods for typing, the analysis of the data obtained, the use of data analysis to evaluate the user account of the VKontakte network from an emotional point of view are considered, and the results of developing your own program in the language are also given Python, which allows you to classify photos according to specified parameters.

The structure of the bachelor's work is presented by the introduction, three chapters, conclusion, list of references.

The introduction describes the relevance of the study, gives a brief description of the work done.

The first chapter examines image analysis from an emotional point of view. The second chapter describes the mathematical apparatus for analyzing the color distribution and tonality of text on images on the VKontakte social network. The third chapter describes the developed software for analyzing the color distribution and tonality of text on images on the VKontakte social network and discusses an example of its use. In conclusion, conclusions on the work done are presented.

The work used 3 tables, 17 figures, the bibliography contains 20 literature. The total volume of graduation qualification work is 56 pages.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
1 ОБЗОР СОВРЕМЕННОГО СОСТОЯНИЯ ИССЛЕДОВАНИЯ В ОБЛАСТИ АНАЛИЗА ИЗОБРАЖЕНИЙ	8
1.1 Анализ основных понятий анализа изображений	8
1.2 Обзор литературы по анализу изображений	9
1.3 Анализ психологического воздействия цвета	12
1.4 Выбор средств автоматизации сбора изображений	16
2 МАТЕМАТИЧЕСКИЕ МЕТОДЫ ДЛЯ АНАЛИЗА ЦВЕТОВОГО РАСПРЕДЕЛЕНИЯ И ТОНАЛЬНОСТИ ТЕКСТА НА ИЗОБРАЖЕНИЯХ... ..	18
2.1 Обзор методов анализа и выявления содержимого на изображении	18
2.2 Обзор математических методов кластеризации	24
2.3 Неиерархическая кластеризация данных с использованием k-means	27
2.4 Сегментации изображений	31
2.5 Распознавание слов	34
2.6 Оценка тональности текста	36
3 АНАЛИЗ ЭМОЦИОНАЛЬНОЙ СОСТАВЛЯЮЩЕЙ ИЗОБРАЖЕНИЙ ИЗ СОЦИАЛЬНОЙ СЕТИ ВКОНТАКТЕ	38
3.1 Алгоритм работы программного обеспечения	38
3.2 Тестирование программного обеспечения	42
3.3 Программный код	48
ЗАКЛЮЧЕНИЕ	53
СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ	54

ВВЕДЕНИЕ

В настоящее время компьютеры применяются во всех сферах деятельности человека, при этом важнейшую роль играют компьютерная обработка фотографий и растровая графика в целом.

Современное программное обеспечение в сфере цифровой обработки изображений позволяют человеку ежедневно создавать большое число фотографий, которые затем заполняют собой профили пользователей в различных социальных сетях, являясь основным информационным контентом.

Эта форму контента по сути является своеобразной презентацией пользователя, так как представляемые в профиле изображения человек выбирает на уровне подсознания, основываясь на принципе «нравится – не нравится».

Формируемый в профиле пользователя информационный контент соотносится с ключевыми особенностями личности пользователя, с его эмоциональной составляющей, с его психологическими потребностями.

Значительное количество пользователей по всему миру пользуется разнообразными социальными сетями, постоянно наполняя свой профиль актуальным информационным содержанием, в первую очередь, новыми фотографиями и видеоматериалами.

Посторонний человек имеет возможность дать психоэмоциональную оценку представленному профилю в социальной сети путём проведения визуального анализа.

Современные компьютерные технологии предоставляют возможность провести подобный анализ огромного количества профилей социальной сети за приемлемое время.

Чтобы автоматизировать такой анализ, требуется обучить электронно-вычислительную машину воспринимать визуальный контент также, как его воспринимают пользователи социальной сети.

На основе использования цифровой обработки можно решить целый комплекс задач: повысить качество изображений, выполнить распознавание, в том числе текстовой информации на изображении и много других.

В бакалаврской работе цифровая обработка изображений позволяет проанализировать психоэмоциональную составляющую фотографий, типизировать изображения по цветовому распределению и текстовой информации, оценив таким образом тональность фотографии.

Затем фотографии типизируют, и вычисляется психоэмоциональная характеристика аккаунта на основе количества «агрессивных», «нейтральных», «отрицательных» и «положительных» фотоизображений.

Целью бакалаврской работы является разработка алгоритма вычисления психоэмоциональной характеристики профиля пользователя на основе типизации фотоизображений в социальной сети «ВКонтакте» путём анализа цветового распределения и тональности текста на фотоизображениях.

Для достижения поставленной цели требуется выполнение следующих задач.

1. Проанализировать воздействие цветового распределения на психоэмоциональное состояние.
2. Автоматизировать получение фотоизображений пользователей социальной сети.
3. Разработать алгоритм типизации фотоизображений на основе цветового распределения.
4. Разработать алгоритм типизации фотоизображений по тональности текста на изображениях.
5. Интерпретировать результаты типизации с реальным психоэмоциональным воздействием на пользователя.

Актуальность бакалаврской работы вызвана тем обстоятельством, что в современных условиях каждый руководитель желает принять на работу

наиболее надёжных, эмоционально устойчивых работников, готовых работать на результат.

Разработанный алгоритм позволит кадровым службам организации экономно расходовать ресурсы, направленные на набор команды ответственных работников, а сотрудникам организаций в сфере образования – обеспечить безопасность и требуемую атмосферу в группе обучающихся.

Разработанный алгоритм можно использовать для выявления пограничных состояний у детей старшего и среднего школьного возраста, в том числе имеющих проблемы с речью. Это связано с воздействием цветов на психоэмоциональное состояние личности.

Цветовое распределение может быть привлекающим, отталкивающим, взбадривающим, успокаивающим.

Данные о том, какие цветовые оттенки предпочитает тот или иной пользователь, получить просто – люди на эмоциях ежедневно или еженедельно наполняют имеющиеся профили в социальных сетях фотографиями.

Автоматически анализируя это информационное наполнение, можно оперативно получать психоэмоциональные характеристики.

В первой главе рассматривается анализ изображений с точки зрения психологии, анализируется воздействие цветового распределения на психоэмоциональное состояние. Выбираются источник получения изображений для последующего анализа, цветовая модель для задания цвета, инструментарий для получения изображений, метод этого инструментария.

Во второй главе рассматриваются подходы к анализу изображений по цветовому распределению и к анализу тональности текста. Выбираются библиотеки для реализации программного обеспечения.

В третьей главе представлена разработка программы на основе анализа размещённых пользователем в открытом доступе изображений.

1 ОБЗОР СОВРЕМЕННОГО СОСТОЯНИЯ ИССЛЕДОВАНИЯ В ОБЛАСТИ АНАЛИЗА ИЗОБРАЖЕНИЙ

1.1 Анализ основных понятий анализа изображений

Цветом называют свойство объектов излучать и отражать свет некоторой части спектра. В широком смысле значение цвета определяется системой взаимодействий, градаций, изменчивостью оттенков и тонов.

Как говорил Иоганн Гёте, «цвета действуют на душу: они могут вызывать чувства, пробуждать эмоции и мысли, которые успокаивают нас или волнуют, а не печалят или радуют» [1]. Из этого следует, что цвета занимают большое место в психоэмоциональной сфере личности. У каждого человека цвет, присутствующий на изображении, вызывает воспоминания и какие-либо чувства.

Изображением называют графическое описание реального или выдуманного события путём создания определённой иллюстрации, которую можно визуальным образом воспринимать и детально анализировать. В современном мире информационных технологий обычно применяются средства компьютерной графики, позволяющие воспроизводить изображение с помощью точек.

Человек воспринимает два компонента изображения:

- главный компонент изображения, в качестве которого выступает некоторый объект или предмет;
- фон изображения, который представляет собой цвет или тон, на котором располагается главный компонент.

Фон изображения особенно важен, так как позволяет донести полную информацию. Если фон изображения подобран верно, то главные компоненты изображения представлены более выразительно, тональность текста усилена. Это говорит о том, что фон изображения занимает важное место в психоэмоциональной составляющей изображения.

Для получения определённой информации, которую содержит изображение, необходимо его проанализировать. Анализ изображения позволяет выделить предметы на изображении, охарактеризовать цветовое распределение, найти текстовые объекты для определения тональности и другое.

По данным глобального отчёта Digital 2020 [2], в России активными пользователями социальных сетей являются 70 миллионов жителей, что составляет 48% от всего населения.

Согласно StatCounter [3] в Российской Федерации наиболее популярной социальной сетью является «VKontakte». Её доля среди других социальных сетей за последний год составила 30%. Из этой статистики следует, что максимальное число фотоизображений, которые публикуют пользователи, размещается в сети «VKontakte». Поэтому эта социальная сеть выбрана для получения изображений в бакалаврской работе. Чтобы проанализировать цветовое распределение и тональности текста на изображениях в социальной сети «VKontakte», необходимо сначала проанализировать и классифицировать изображения, которые представлены в профилях пользователей.

1.2 Обзор литературы по анализу изображений

Изображение содержит определённую информацию, которую легко считывает человек, но не компьютер. Для получения информации для принятия решений, которую содержит изображение, необходимо проанализировать изображение. Для этого применяется технически аналогичное человеческому машинное зрение, основой для которого служит математический анализ.

Ключевыми задачами машинного зрения являются получение, обработка и анализ информации на изображении. Эти задачи выполняются путём математической обработки и оценки информации с применением

компьютерных средств. При этом анализируемые изображения представляются двумерными массивами, которые составляются регистрационными системами. Изображение представляется в виде элементов двумерных массивов, каждый из которых соответствует преобразованному в конфигурацию света значению яркости точки анализируемого изображения.

В настоящее время важное значение имеет информация в форме мультимедиа. Подобный контент стремительно развивается, так как каждую секунду публикуются новые фотографии и видеозаписи, мультимедийный контент в сферах науки и развлечений и так далее. Чтобы получить максимум полезной информации, нужно научиться выполнять обработку и поиск необходимых изображений. Для этого нужно развивать математические методы машинного зрения, которые могут точно и эффективно классифицировать изображения.

Для классификации изображений имеется большое число математических методов, позволяющих отобрать и выделить существенные признаки. Например, в статье [4] изображения классифицируются на основе сочетаний разных информативных признаков, образуя 19 классов. Эффективность методов отбора и выделения информативных признаков для ряда предметов составляет около 93,6%, а у некоторых достигает практически 100%.

Основой для других методов отбора и выделения информативных признаков служит уменьшение количества признаков. Например, в статье Глумова [5] количество признаков уменьшено в 10 раз, при этом признаками выступают точки представленного изображения. Такая методика применяется при обработке медицинских изображений.

Уменьшение пространства признаков позволяет добиться хорошей эффективности при снижении трудозатрат. Так, в статье [6] выявили пространство, которой состоит только лишь из 5 признаков, при этом изначально в пространстве признаков их было 169. В статье анализируются

признаки, которые характеризуют хроническую обструктивную болезнь лёгких. Авторы используют дискриминантный анализ.

В последнее время из-за появления новых технологий в работу психотерапевтов внедряются новые методики. Изображение становится новым инструментом диагностики психических отклонений.

Значительный вклад в развитие цветовой психодиагностики внёс психолог из Швейцарии Макс Люшер. Он рассмотрел выбранные цвета около 35 тысяч человек, обладающих разными эмоциональными характеристиками, и сделал вывод, что психодиагностика на основе воздействия цвета предоставляет большие возможности и даёт достаточную эффективность. В настоящее время его проективная методика используется по всему миру.

В 1980 году проективная методика Макса Люшера [7] была исследована советскими учёными Устиновичем и Дашковым. В этом же году зависимость цветовых предпочтений от психоэмоционального состояния человека рассматривал Плишко. В том же году Эткингом были разработаны методы изучения психоэмоциональных составляющих отношения личности к себе и к людям, которые имеют для неё наибольшее значение. В тесте Эткинга взаимное расположение отдельных пар цветов из набора в восемь цветов выполняется с чётко заданной целью.

Осуществление выбора цветов человеком при отсутствии повторов в комбинации цветов происходит по законам психологии, которые основываются на сложившихся психологических характеристиках человека и его текущем психоэмоциональном состоянии. Распределение цветов без повторений соответствует распределению Гаусса. Чувствительность к небольшим переменам текущего психоэмоционального состояния выявляется на основании полученных данных, которые подтверждают гипотезу.

1.3 Анализ психологического воздействия цвета

Видимый цвет формируется под воздействием световых волн. Цвет также образуется путём воздействия электромагнитного излучения той или иной частоты на органы зрения. Также на восприятие цвета воздействуют психические и физиологические характеристики человека.

Цвет рассматривается физиками, психологами и физиологами как комплексная характеристика, так как влияет на личность не только на уровне зрительного восприятия, но и с точки зрения психологии. Химики разрабатывают компоненты композиционных материалов, придающие цвет. Оптика рассматривает электромагнитное излучение, формирующее цвет. Физиология исследует цветовое влияние на нервную систему и зрительный аппарат. Психологи рассматривают цветовое влияние на психоэмоциональное состояние личности.

Цвета подразделяются на 2 типа: ахроматические, к которым относятся белый цвет, чёрный цвет, и все оттенки серого от ближайшего к белому до полностью чёрного; хроматические, к которым относят все остальные цвета и оттенки (рисунок 1.1). Органы зрения человека способны отличать примерно четыреста ахроматических цветов.

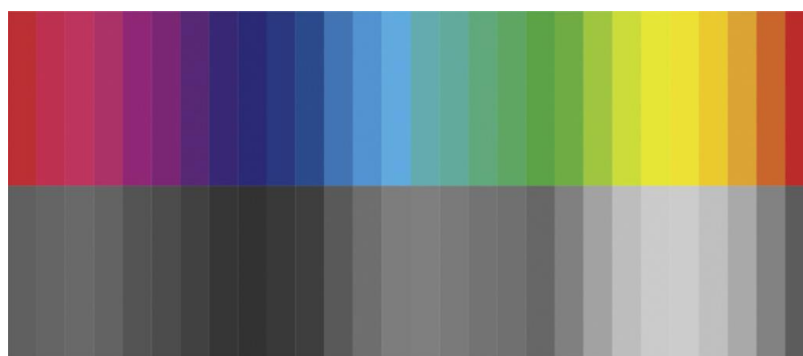


Рисунок 1.1 – Хроматические и ахроматические цвета

По другой классификации цвета подразделяются на 4 типа: пастельные, светлые, спектральные и тёмные. Цвета обладают такими параметрами, как

насыщенность, светлота и тон. Насыщенность показывает, насколько сильно отличаются одинаковые по светлоте хроматический цвет от ахроматического. Тон – это цветовая характеристика, которая группирует схожие цвета, например синие или жёлтые. Светлота – это цветовая качественная характеристика, которая тем больше, чем тот или иной цвет ярче.

Цветовая модель – модель, математически описывающая цветовые оттенки в виде числовых наборов, задачей которой – задать соотношение между цветовыми оттенками, формирующимися техническими приборами, и цветовыми оттенками, воспринимаемыми человеческим зрением. Цветовые модели предназначены для качественного и числового задания цветовых оттенков в разнообразном программном обеспечении.

Цветовые модели подразделяются на 3 класса: аддитивные, перцепционные и субтрактивные. Первый тип моделей основывается на сложении цветов, например модель RGB (Red, Green, Blue), которая используется в бакалаврской работе. К перцепционным цветовым моделям относят модели, основанные на том, как цвет воспринимается. Это модели HLS, HSB, LAB, YCC. Субтрактивные модели основываются на вычитании цветов, например модель CMYK (Cyan, Magenta, Yellow, Key или Black).

Влияние цвета на человека – сложнейший феномен, который зависит от множества условий и основывается на работе мозга. Живописец Василий Кандинский в Баухаусе рассказывал своим ученикам о физике цвета, выделяя базовые цвета (жёлтый, красный, синий) и соответствующие им базовые фигуры – треугольник, квадрат, круг. Кандинский делал акцент на отдельном воздействии каждого цвета. Красный цвет, по его мнению, обозначал внутреннее движение личности, огонь, равновесие, устойчивость, прямоугольность. Жёлтый же, по его мнению, обозначал движение личности извне, подвижность, остроугольность. Синий цвет обозначает холодность, движение вовнутрь, тупоугольность. Чёрный и белый цвета, по его мнению,

символизируют тишину: чёрный цвет обозначает абсорбцию цвета, а белый цвет обозначает возникновение нового.

От того, как человек воспринимает цвет, зависит то, как цвета будут сочетаться, образовывать целостную картину, вызывать чувства и эмоции, изменяя психоэмоциональное состояние личности. В результате взаимодействия различных цветовых оттенков или групп цветов формируется цветовая гармония. Человек воспринимает цветовые оттенки в зависимости от своего возраста, физиологии, окружения, положения в обществе, культурного развития.

Цвета по-разному воздействуют на психоэмоциональное состояние. Например, жёлтый, красный, оранжевый раздражают, заставляют человека действовать. Их относят к тёплым цветам. Холодные цвета и оттенки оказывают противоположное воздействие. Пастельные тона оказывают успокаивающее, смягчающее, сдерживающее воздействие. Цвет может воздействовать на то, как человек воспринимает расстояние. Холодные цвета увеличивают расстояние, тёплые цвета – сокращают.

Выбор цвета человеком рассматривается на протяжении последних десятилетий, учёные проводят разнообразные опыты, исследуют отличия во влиянии цвета на мужчин и женщин, на людей с разными психологическими характеристиками, с разным окружением, из разных территорий и так далее.

Различный человеческий опыт приводит к разному цветовому восприятию. Например, человеку, проживающему в северной местности, не хватает солнца и теплоты. Такой человек предпочитает тёплые цвета. Человек, проживающий там, где тепло, предпочитает холодные цвета при выборе мебели и обоев.

Цветная символика представляет собой способ придания цветовым оттенкам определённых свойств, пробуждающих эмоции, вызывающих воспоминания. Человек может считывать цветовой настрой и осуществлять выбор цвета, который больше всего подходит к его психоэмоциональному состоянию. Человек выбирает цвет неосознанно, но используя тот или иной

цвет, возможно легко воздействовать на людей на основе грамотного использования информации о психоэмоциональном влиянии цветового оттенка.

На основе проективной методики Люшера Клар сделал своё заключение о цветной символике. Красный цвет он считал возбуждающим. При смещении оттенка цвета от красного к жёлтому, по его мнению, возбуждающее действие цвета нарастает. Если добавить коричневый цвет, то оттенок цвета будет успокаивать. Если перемешать синий и красный цвета, полученный цвет будет побуждать к действию. Если перемешать белый и красный цвета, по мнению учёного, возбуждение уменьшится.

По мнению Клар, тёмные оттенки синего успокаивают. Когда человек смотрит на тёмно-синий цвет, нервная система становится спокойной, частота дыхательных движений, сердцебиение, частота сердечных сокращений, артериальное давление значительно понижаются. В дохристианскую эпоху считалось, что тёмно-синий цвет хорошо подходит, чтобы сосредоточиться и достичь просветления. Клар отделял светлые тона синего цвета, полагая, что они несут простодушие и непосредственность. Смешение синего и зелёного символизирует неуверенность в себе.

Смешение красного и синего символизирует стремление к единству. Оттенки цвета между синим и красным, по мнению учёного, символизируют эмоциональную близость.

Психологи считают, что оттенки зелёного символизируют безучастность. По мнению Клар, светлый оттенок является самым холодным, а тёмный – навеивает воспоминания о весне, символизирует развитие и жизнелюбие.

По мнению Клар, жёлтый цвет заставляет действовать, вызывает радостные эмоции, успокаивает.

Учёный считает, что серый цвет не оказывает никакого влияния. Белый цвет олицетворяет начинание, чёрный – символизирует напряжение, вызывает страх.

С психологической точки зрения, цветная символика и то, как человек воспринимает цвет, говорят, насколько важны преобладающие цвета в изображениях, которые являются базовой составляющей информации в профиле пользователя. По преобладающим на изображении цветам можно понять, какой посыл содержит то или иное изображение. Если классифицировать фотографии пользователя в социальной сети в зависимости от создаваемого ими настроения, то можно определить его психоэмоциональное состояние и тип личности.

1.4 Выбор средств автоматизации сбора изображений

Чтобы провести полный анализ изображений, необходимо достаточное их количество, так как невозможно произвести оценку на основе одной фотографии из-за большой погрешности. Следовательно, чем выше количество анализируемых изображений при приемлемых временных затратах, тем более репрезентативной будет выборка, и тем ниже погрешность результата.

Получение изображений вручную занимает значительный временной промежуток. Значит, данный процесс требует автоматизации на основе современных информационных технологий.

В бакалаврской работе получение изображений из сети «ВКонтакте» будет автоматизировано на основе использования инструментария API ВКонтакте.

API (application programming interface) представляет собой особым образом организованный комплекс инструментов, позволяющих взаимодействовать с разрабатываемой прикладной программой или веб-сайтом, автоматизируя получение информации. Этот инструментальный предоставляет возможность упростить запросы и сделать их более удобными, не вникая в особенности реализации функций.

API Вконтакте – интерфейс между создателем прикладного программного обеспечения (ПО) и приложением или веб-интерфейсом, которые взаимодействуют с этим ПО. API Вконтакте позволяет серьёзно упростить разработку приложения или веб-интерфейса, так как разработчик будет использовать уже имеющиеся в API функции или структуры данных.

API Вконтакте предоставляет возможность получать информацию, хранящуюся в социальной сети «ВКонтакте» путём отправления HTTP-запросов к серверу. Применение этого API предпочтительнее по сравнению с разработкой собственных GET- и POST-запросов к серверу, так как упрощает процесс автоматизации получения изображений.

В бакалаврской работе для разработки приложения будет использован метод API `photos.getAll`, возвращающий все изображения профиля пользователя или группы «ВКонтакте» в обратной хронологической последовательности. На вход программы будут подаваться гиперссылки на фотоальбом, содержащие идентификатор пользователя-владельца фотоальбома и идентификатор фотоальбома. Этот метод имеет недостаток, который заключается в невозможности скачивания более сотни фотографий за один запрос.

Для реализации приложения, обеспечивающего выполнение задачи анализа цветового распределения и тональности текста на изображениях в социальной сети «ВКонтакте», был выбран язык Python.

Вывод по первой главе

В этой главе был рассмотрен анализ изображений с точки зрения психологии, проанализировано воздействие цветового распределения на психоэмоциональное состояние. Выбран источник получения изображений для последующего анализа. Выбрана цветовая модель для задания цвета. Выбран инструментарий для получения изображений. Выбран метод этого инструментария.

2 МАТЕМАТИЧЕСКИЕ МЕТОДЫ ДЛЯ АНАЛИЗА ЦВЕТОВОГО РАСПРЕДЕЛЕНИЯ И ТОНАЛЬНОСТИ ТЕКСТА НА ИЗОБРАЖЕНИЯХ

2.1 Обзор методов анализа и выявления содержимого на изображении

Выбор метода анализ и выявления содержимого на изображении прямо связан с качеством изображения и ценностью информации на изображении. Для анализа изображений можно использовать диаграмму, которая распределяет точки изображения по яркости и цветам. Можно также выделить и исследовать на изображении интересующую информацию. В качестве примеров определения качества изображения можно привести оценивание резкости, поточечный анализ, исследование баланса цветного изображения. В качестве примеров выделения нужной информации можно привести поиск предметов, оценка их цвета, размера и других параметров. Анализ изображения часто выполняется одновременно с обработкой изображения. Чтобы выполнить анализ изображения и провести необходимые замеры, сначала требуется определить границы заданных предметов и выполнить шумопонижение для неважных объектов. В бакалаврской работе интерес будут представлять текстовые знаки. Поэтому требуется проанализировать методы определения текстовых символов на изображении, которые будут использованы в дальнейшем для анализа изображений.

На смысл изображения влияет значение текстовых символов, которые на нём расположены. Так, если на изображении располагаются текстовые символы, то методы анализа тональности текста позволят определить посыл изображения.

Таким образом, сначала требуется достоверно определить, имеются ли текстовые символы на изображении или нет. Так как количество вариантов текстовых символов огромно, обнаружение текста является трудной задачей.

Текстовые символы различаются по гарнитуре шрифта, начертанию, кеглю и прочим параметрам. Диапазон изменения каждого из этих параметров достаточно широк. При этом входное изображение может быть невысокого качества, неконтрастное, фон изображения может быть неоднороден. В связи с тем, что эта проблема имеет разнообразные трудности, имеются разные способы выявления текстовых символов на изображении. Уже после решения этой задачи требуется распознать выявленные текстовые символы.

Для решения задачи выявления текстовых символов существуют алгоритмы, основанные на компонентах графа и описанные в работах [8], [9], [10], [11]. Другой способ использует гибридные алгоритмы, которые представлены в работе [12]. Оба вида алгоритмов развиваются за счёт имеющихся первоначальных данных о текстовой структуре. Алгоритмы, основанные на компонентах графа, содержат следующие этапы.

1. Сначала определяются информационно насыщенные области изображения. Чтобы разбить изображения на такие области, применяются методы, основанные на методе MSER (maximally stable extremal regions) [13]. Метод MSER не привязан к языку, так как он базируется на интенсивности предметов на изображении, а текстовые символы обычно характеризуются одинаковой интенсивностью. Это означает, что метод MSER работает для любых языков.

2. Затем необходимо сгруппировать обнаруженные текстовые символы в единый текст. Чтобы сгруппировать текстовые символы, применяется 2 способа. Первый способ основан на применении некоторых правил, на основе которых текстовые области попарно объединяются. Второй способ основан на машинном обучении. Чтобы объединить текстовые области, составляется перечень критериев, основанных на расстоянии между символами, размере символов относительно изображения, границ текстовых областей и интенсивности текста. Если текстовый символ состоит из нескольких компонентов, определённые признаки, например размер области текста или выравнивание, могут привести к неверной группировке, что в

свою очередь приведёт к необходимости дополнительно произвести выполнение этой части алгоритма.

Имеются методы, основанные на областях изображения (region based). Чтобы классифицировать области с текстовыми символами и области, не содержащие текст, применяется машинное обучение. Среди методов, основанных на областях изображения, хорошо себя зарекомендовали методы, которые позволяют произвести обучение представлениям (representation learning), и методы, использующие признаки текстур. Чтобы распознавать и классифицировать объекты, зачастую применяют методы, основанные на глубоком машинном обучении (deep machine learning). В алгоритмах, позволяющих обнаруживать текстовые символы, часто применяют свёрточные нейронные сети [14], [15].

Свёрточные нейронные сети представляют собой инструментальный машинного обучения, который предназначен для типизации графических изображений. Он широко применяется из-за возможности эффективно распознавать объекты на изображениях. Распознавание представляется на выходе чередующимися полносвязными, свёрточными и субдискретизирующими слоями. Суть свёрточной нейронной сети заключается в компьютерной имитации жёстко связанных между собою нейронов. Требуется обеспечить быструю обучаемость такой сети для последующего выделения объектов и поддержки принятия решений, аналогичных решениям человека. Основное достоинство нейронных сетей заключается в том, что они могут изменять алгоритмы своей работы на основе накопленных данных без перенастройки. Нейронная сеть включает в себя слои, состоящие из нейронов. Входной слой состоит из нейронов, которые поступают извне и принимают формы, содержащие информацию. Выходной слой включает в себя нейроны, которые реагируют на поступающую информацию. Промежуточные слои содержат нейроны, которые выполняют обработку информации. Слои нейронов соединяются весами, которые представляют собой особым образом организованные связи.

В зависимости от влияния предшествующего слоя на последующий веса могут быть положительными либо отрицательными.

Имеющееся знание о текстовой структуре, которое предшествует получению опыта, представляет собой использование разнообразных правил и методов исследования, которые помогают распознавать текстовые символы и используются, чтобы выделить области, которые содержат текст, и области, которые не содержат текстовые символы. Результативность предложенной методики зависит от конкретных применяемых характеристик, затрагиваемых в предложенном наборе методов исследования. Помимо этого, возможно подобрать необходимые характеристики ручным способом, например в случае применения машинного обучения.

Такие алгоритмы включают алгоритмы, в основе которых находятся компоненты графа, и гибридные алгоритмы.

Потенциальные текстовые области – это области изображения, которые образуются в случае применения методов, основанных на компонентах графа. Группировка текстовых символов выполняется из оставшихся областей изображений, которые являются компонентами графа. Обычно компоненты графа формируются одним из двух способов: либо путём кластеризации, либо на основании их геометрии.

Гибридные алгоритмы в сочетании с применением машинного обучения сначала формируют ряд регионов, которые содержат текстовые символы определённой величины. Затем формируются компоненты графа и удаляются регионы, которые не содержат текст. В итоге выполняется группировка регионов, содержащих текстовые символы.

Важно отметить, что методика, основанная на компонентах графах, обладает максимальными производительностью и эффективностью. Это подтверждено итогами экспериментов, проведённых в 2013 году на международной конференции ICDAR [16].

Алгоритмы, в основе которых лежат регионы, обычно применяют машинное обучение и метод скольжения окон. К ним относятся текстурные

алгоритмы и алгоритмы, в которых применяется глубокое машинное обучение.

Алгоритмы на основе глубокого машинного обучения выявляют текстуры, затем путём классификации определяется, имеются текстовые символы или нет. Признаки текстур обычно формируются путём применения метода скольжения окон для всех областей изображения. Так, признаки текстур формируются в итоге свёрточной обработки области с использованием банка цифровых фильтров. Подобную свёрточную обработку нужно провести для различных масштабов изображения, что требует значительных трудозатрат. Необходимо отметить, что в случае, если входное изображение имеет недостаточное разрешение, точность таких алгоритмов серьёзно падает. Это является основным недостатком этих алгоритмов.

Алгоритмам глубокого машинного обучения не требуется выбирать признаки наличия текстовых символов на изображении ручным образом. Эти алгоритмы получают такие признаки из обучающей выборки изображений. На вид этих признаков влияет способ обучения, например признаки масок свёртки и сочетание масок свёртки с алгоритмами на основе свёрточных нейронных сетей [17].

Чтобы классифицировать изображения по цветовым оттенкам, используется алгоритм кластеризации k-means. На основе этого алгоритма были проанализированы изображения для определения цветовой тональности каждого изображения.

Алгоритм кластеризации k-means [18, 19, 20] достаточно прост и широко используется для классификации. Кластеризация по методу k-means состоит в разделении множества N точек изображения на число кластеров K , которое задаёт пользователь. Кластер – это набор точек, состоящий из точек изображения, но эти точки группируются независимо от их местоположения. Если же местоположение точек влияет на важные для классификации параметры, тогда оно учитывается при разбиении на кластеры.

Допустим, есть множество X , состоящее из N точек $X = \{x_1, \dots, x_N\}$, и вектор свойств $V(x_i)$. Алгоритм k-means в этом случае заключается в следующем.

1. В каждом из k кластеров инициализируются значения возможных векторов свойств $V(x_i)$. Алгоритм k-means традиционно выбирает значение каждого элемента вектора свойств $V(x_i)$ случайным образом из всех возможных вариантов. Например, при инициализации вектора свойств (R,G,B) он формируется из интенсивности соответственно красного, зелёного и синего компонентов, причём элемент R формируется случайным образом из диапазона значений интенсивности красного компонента.

2. Затем нужно определить все точки в свой кластер. Для этого применяется функция расстояния, которая измеряет дистанцию между двумя векторами свойств. На основе этой дистанции точки распределяются в кластер с ближайшим центром, который обозначают μ_k . После распределения каждая точка x_i входит только в один кластер C_k .

3. Затем на основе значений векторов свойств центры кластеров μ_k пересчитываются по формуле (2.1):

$$\{V(x_i) | x_i \in C_k\} \quad (2.1)$$

Далее второй и третий пункты алгоритма повторяются до тех пор, пока значения μ_k не перестанут изменяться. Неизменность значений означает, что точки после последнего витка цикла не поменяли кластер. Этот момент носит название признака остановки. На рисунке 2.1 изображены итоги кластеризации изображения с помощью алгоритма k-means. Были образованы четыре кластера. Важно отметить, что асфальт и крыша Вашингтонского университета распределены в один и тот же кластер, так как и цветовые оттенки похожи.

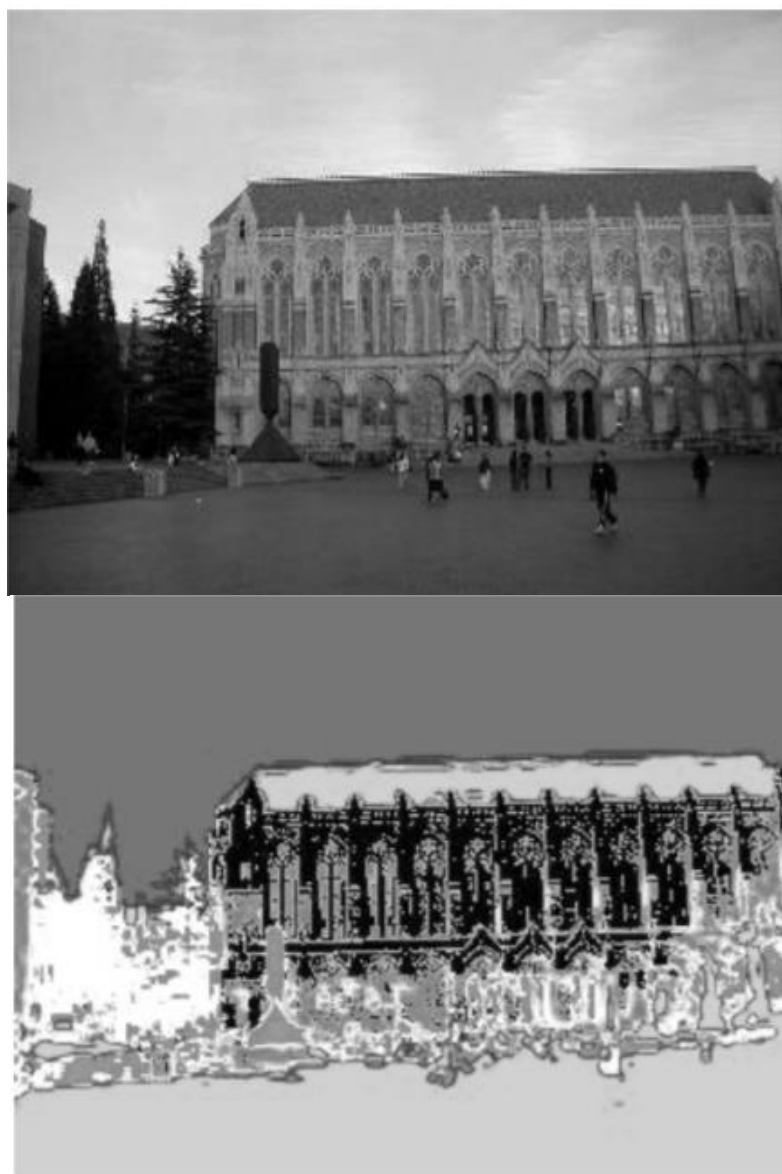


Рисунок 2.1 – Пример анализа изображения методом кластеризации k-means

2.2 Обзор математических методов кластеризации

Основная цель кластеризации заключается в распределении исходных данных по кластерам, в которые они объединяются по близким характеристикам. То есть исходные данные X распределяются в множества G , каждое из которых принадлежит одному из кластеров m . Формирование кластеров происходит так, чтобы объекты, принадлежащие кластеру,

обладали как можно более похожими характеристиками, при этом объекты разных кластеров должны максимально отличаться.

Алгоритмы кластерного анализа данных разбивают на 2 класса: графовые алгоритмы и неиерархические. Каждый класс алгоритмов используют разные методы.

Графовые алгоритмы кластерного анализа данных основываются на последовательном слиянии или разделении кластеров в зависимости от того, какой подход выбран или какая задача поставлена.

При использовании графовых агломеративных методов все объекты сначала образуют каждый свой кластер. Затем на каждом шаге находящиеся рядом кластеры объединяются, пока число кластеров не станет равным заданному минимуму. Формирование кластеров путём слияния на основе агломеративного метода изображено на рисунке 2.2:

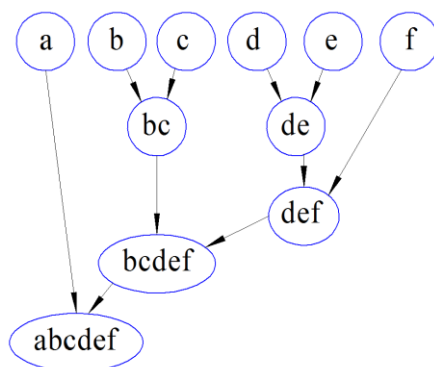


Рисунок 2.2 – Пример дендрограммы для агломеративного метода кластерного анализа

Графовые дивизимные методы противоположны графовым агломеративным методам кластерного анализа. Здесь сначала формируется кластер, в который сливаются все объекты. Затем на каждом шаге кластеры разбиваются, формируя новые кластеры, содержащие объекты, максимально

близкие по своим характеристикам. Разбиение заканчивается, как только будет достигнут заданный максимум.

Способ реализации графового алгоритма зависит от правил распределения по кластерам. Правила распределения определяются условиями поглощения или слияния в кластеры по близким характеристикам. Необходимо заметить, что наилучшее применение графовые алгоритмы кластерного анализа получают лишь при небольшом количестве исходных данных.

При использовании алгоритма ближайшего соседа расстояние между двумя различными кластерами определяется как расстояние между ближайшими объектами, которые относятся к различным кластерам.

При использовании принципа дальнего соседа дистанция между двумя различными кластерами определяется как расстояние между самыми дальними объектами, которые относятся к различным кластерам.

При использовании метода Варда дистанцию между двумя различными кластерами измеряют от центров кластеров и вычисляют как прирост суммы квадратов расстояний объектов до этих центров. В отличие от описанных выше методов здесь применяется дисперсионный анализ. Очередной этап итерации, на котором кластеры объединяются, определяется таким образом, чтобы у целевой функции, которая в свою очередь вычисляется как сумма квадратов расстояний внутри кластера, был минимальный прирост.

Неиерархические алгоритмы широко используются в тех случаях, когда количество исходных данных велико. По сравнению с графовыми алгоритмами в неиерархических необходимо задать определённое число кластеров. Такие алгоритмы можно реализовать, выявив границы кластеров вокруг максимального скопления точек. Другой способ реализации неиерархических алгоритмов заключается в том, чтобы снизить разницу между элементами.

2.3 Неиерархическая кластеризация данных с использованием k-means

Задача кластеризации данных заключается в разбиении объектов на группы (кластеры). При этом распределение объектов по группам должно отвечать критерию оптимальности. В общем случае, оптимальным распределением является то, при котором объекты внутри кластера максимально близки по сочетанию параметров, а объекты из разных кластеров – максимально отличаются.

Критерий оптимальности кластерной структуры задается с помощью целевой функции W , которая для метода k-means выглядит так (2.2):

$$\begin{cases} W = \sum_{c=1}^k W_c, & W \rightarrow \min \\ W_c = \sum_{j=1}^n (x_j - \bar{x})^2 = \sum_{j=1}^n x_j^2 - \frac{1}{n} \left(\sum_{j=1}^n x_j \right)^2, \end{cases} \quad (2.2)$$

где k – заданное количество групп (кластеров), c – номер группы (кластера), n – количество объектов в кластере с номером c , x – вектор параметров объекта с номером j из кластера с номером c , \bar{x} – центр кластера с номером c .

Для определения сходства (близости) объектов с номерами i и j из исходной выборки данных можно использовать любую функцию $d(a,b)$ отвечающую следующим требованиям:

1. Требование симметрии $d(X_i, X_j) = d(X_j, X_i) \geq 0$. Функция должна возвращать одинаковые значения, если поменять ее аргументы местами. При этом функция всегда должна возвращать или положительное значение или 0.

2. Требования различимости нетождественных объектов $X_i \neq X_j \Rightarrow d(X_i, X_j) \neq 0$. Если параметры объектов не одинаковые, то значение возвращаемое функций для этих пар объектов не может быть равно 0.

3. Требования неразличимости тождественных объектов $X_i = X_j \Rightarrow d(X_i, X_j) = 0$. Если параметры объектов одинаковые, то значение, возвращаемое функций для этих пар объектов должно быть равно 0.

4. Требование соблюдения неравенство треугольника $d(X_i, X_j) \leq d(X_i, X_k) + d(X_k, X_j)$. Значение функции для двух объектов X_i, X_j не может быть больше, чем сумма значений функции для пар объектов (X_i, X_k) и (X_k, X_j)

Если функция отвечает этим свойствам, то ее называют метрикой, определяющей меру отличия объектов друг от друга. Это значит, что чем больше значение, возвращаемое данной функцией, тем сильнее параметры этих объектов отличаются друг от друга. Противоположностью таким функциям являются корреляционные меры сходства (чем больше значение корреляционные меры, тем больше сходство сравниваемых объектов).

Стоит отметить также отметить, что значения возвращаемые метриками не ограничены сверху, и их значения зависят от выбора шкалы измерения признаков.

Хоть в качестве метрики при кластеризации данных можно использовать любую функцию, удовлетворяющих описанным выше требованиям, но наибольшую популярность получили следующие метрики: Евклидова d_2 (2.3), Чебышева d_s (2.4), Минковского d_m (2.5), расстояние Манхэттена d_1 (2.6).

Если принять, что каждый объект выборки описывается вектором параметров $(x_1, x_2, x_3, \dots, x_p)$, где p – количество параметров, то формулы расчета сходства объектов с номерами i и j , по данным метрикам можно представить, как (2.3) – (2.6):

$$d_2(X_i, X_j) = \left[\sum_{k=1}^p (x_{ki} - x_{kj})^2 \right]^{\frac{1}{2}} \quad (2.3)$$

$$d_s(X_i, X_j) = \sup_{k=1 \dots p} (|x_{ki} - x_{kj}|) \quad (2.4)$$

$$d_m(X_i, X_j) = \left[\sum_{k=1}^p |x_{ki} - x_{kj}|^p \right]^{\frac{1}{p}} \quad (2.5)$$

$$d_1(X_i, X_j) = \sum_{k=1}^p |x_{ki} - x_{kj}| \quad (2.6)$$

Исходными данными для кластеризации данных с использованием алгоритма k-means является:

- количество k групп в итоговой кластерной структуре;
- выбор одной функции из числа (2.3) – (2.6), используемой для сравнения объектов;
- набор объектов, подвергающихся кластеризации $(X_1, X_2, \dots, X_m) \in R^p$, где m – количество объектов выборки, p – количество параметров, описывающих каждый объект. При этом i -й объект выборки описывается как (2.7):

$$X_i = (x_1, x_2, x_3, \dots, x_p) \quad (2.7)$$

Перед началом выполнения кластеризации данных в соответствии с алгоритмом k-means производится первоначальное определение параметров k центроидов (центров кластеров). Причем их параметры выбираются случайным образом: центроидами считают наборы параметров $(x_1, x_2, x_3, \dots, x_p)$ случайно выбранных объектов (в количестве k штук).

Следующим шагом после определения параметров множества центроидов $\{S_1, S_2, \dots, S_k\}$ является распределение объектов по группам. Каждый объект может принадлежать только одной из k групп. Объект будет отнесен алгоритмом к той группе, с центроидом которой метрика будет возвращать наименьшее значение.

Таким образом, формируется k множеств (C_1, C_2, \dots, C_k) объектов.

После того, как объекты распределены по группам (C_1, C_2, \dots, C_k) , в соответствии с алгоритмом k-means, осуществляется перерасчет параметров центроидов. Если вначале параметры центроидов определялись случайным образом, то при дальнейших расчетах до получения окончательной кластерной структуры используются расчеты, представленные на 2.8:

$$S_i = \frac{1}{|C_i|} \sum_{X \in C_i} X, \quad i = 1 \dots k \quad (2.8)$$

Параметры центроидов S_i рассчитываются как среднеарифметическое от параметров объектов X , попавших при распределении в i -ю группу C_i .

Теперь, когда параметры центроидов изменились проводится перераспределение объектов по группам аналогичным образом, как это описывалось выше.

Таким образом, процесс расчета кластерной структуры продолжается на протяжении нескольких итераций. Каждая последующая итерация приводит к снижению значения целевой функции, показанной на (2.2).

Кластеризация продолжается до тех пор, параметры центроидов не перестанут изменяться, т.е. пока на текущей итерации t не выполнится условие, показанное на (2.9):

$$(C_1(t-1), C_2(t-1), \dots, C_k(t-1)) = (C_1(t), C_2(t), \dots, C_k(t)) \quad (2.9)$$

После выполнения данного условия, результаты кластеризации предъявляются в качестве результата пользователю.

2.4 Сегментации изображений

Задачу сегментации изображения можно описать следующим образом. Необходимо разделить исходное изображение на фрагменты таким образом, чтобы элементы изображения внутри фрагмента были объединены по смыслу.

Сегментация является промежуточным этапом анализа изображений. Разбиение изображения на фрагменты позволяет упрощать задачу его анализа, применяя к различным ее частям необходимые алгоритмы. В простейшем случае при сегментации изображения формируются непересекающиеся области, т.е. каждый пиксель изображения может принадлежать только одной области.

Сегментация изображений используется, например, в системах компьютерного зрения, анализирующих дорожно-транспортную обстановку, при анализе медицинских изображений, при разметке изображений в системах информационного поиска.

В настоящее время существует большое количество алгоритмов для сегментации изображений, с использованием разного набора признаков на основе различных подходов. Общей чертой алгоритмов является их инерционность, которая заключается в том, что сегментация выполняется на основе расчетов в вычислительном цикле. Каждая итерация цикла заканчивается расчетом оценки критерия качества сегментации на основе целевой функции. Таким образом, алгоритм сегментации состоит из:

- вычислений внутри цикла (решение задачи классификации относительно каждого пикселя);
- расчета на каждой итерации критерия качества сегментации.

Распределение пикселей по классам для цветного изображения задается парой:

$$S = (I, L), \quad (2.10)$$

где I – исходное изображение, относительно которого решается задача сегментации (I задается как прямоугольная матрица, элементы которой соответствуют цветам пикселей в формате RGB). L – матрица, такого же размера как и I , значения матрицы указывают на номер сегмента изображения, к которому относится пиксель, находящийся в матрице I по аналогичным координатам.

Пусть имеется множество V , которое будет являться связанным, когда для произвольной пары пикселей p_1, p_2 имеется такая последовательность $(p_1, q_1, q_2 \dots q_m, p_2)$, пиксели в которой $(q_k, q_{k+1}), \forall k \in (1, 2, \dots, m-1)$ и $(p_1, q_1), (p_2, q_m)$ являются смежными. Если разбить исходное множество, то мы получим непересекающиеся связанные промежутки $C_1 \dots C_n$.

Полученные компоненты $C_1 \dots C_n$ представляют собой результат деления матрицы L . Каждый компонент матрицы L представляет собой пару (B, M) , B – бинарная маска изображения, а M – метка класса (указатель на сегмент) (2.11):

$$L = \sum_{i=1}^n B_i M_i \quad (2.11)$$

Результат сегментации изображения можно представить так, как это показано на (2.12):

$$S = (I, ((B_1 M_1), \dots, (B_n M_n))) \quad (2.12)$$

Если выборка изображений представлена как $Images = \{S_1, S_2, \dots, S_W\}$, то результат сегментации i -ого изображения (2.13):

$$\hat{S}_i = (I, ((\hat{B}_1 \hat{M}_1), \dots, (\hat{B}_m \hat{M}_m))), \quad (2.13)$$

при этом $\forall i \in \{1, 2, \dots, W\}$ $C_j \in S_i$ и $\hat{C}_j \in \hat{S}_i$.

Введем следующие обозначения: $center(C_j)$ – центр масс компоненты C_j . $square(C_j)$ – площадь компоненты C_j (в данном случае площадь измеряется по количеству пикселей). Тогда на основе введенных обозначений можно задать меру сходства двух бинарных масок B и B' по (2.14):

$$HD(B, B') = \frac{|xor(B, B')|_{L_1}}{\sqrt{square(B)square(B')}} \quad (2.14)$$

При $\forall j \in \{1, 2, \dots, m\}$: $\hat{C}_{j'} = (\hat{B}_{j'}, \hat{M}_{j'})$ будет соответствовать $C_j = (B_j, M_j)$ при соблюдении (2.15):

$$\begin{cases} |center(\hat{C}_{j'}) - center(C_j)|_{L_2} \leq T_c, \\ HD(\hat{B}_{j'}, B_j) \leq T_s \end{cases}, \quad (2.15)$$

где T_s – порог меры сходства, T_c – порог близости центра.

При восстановлении соответствия могут существовать такие $C_j \in S_i$ и $\hat{C}_j \in \hat{S}_i$, для которых не будут найдены пары. Это возможно при двух вариантах:

- множество компонент без пар из разметки, получение с помощью алгоритма и не имеющие пары в оригинальной разметке:

$$False_Components_Without_Pair_i = FCWP_i = \{\hat{C}_{j'}\};$$

- множество компонент без пар из истинной разметки, не найденные алгоритмом – $True_Components_Without_Pair_i = TCWP_i = \{C_j\}$.

Качество бинарной сегментации $(\hat{S}_1, \dots, \hat{S}_W)$ сгенерированной алгоритмом Alg можно оценить с использованием параметра Q_b , который рассчитывается по (2.16):

$$Q_b(Images, Alg) = \frac{1}{W} \sum_{i=1}^W \frac{|TCWL_i| + |FCWP_i|}{n_i + m_i}, \quad (2.16)$$

где $Images = \{S_1, S_2, \dots, S_W\}$ – выборка изображений, на которых проверяется качество сегментации. Чем меньше значение Q_b , качественней сегментация.

2.5 Распознавание слов

Рассмотрим слово в виде распознаваемого образа. В этом случае необходимо ввести меру, по значению которой можно оценить близость распознаваемого образа к известным шаблонам.

Введем следующие обозначения: $\{t_1, t_2, \dots, t_j\}$ – множество тестовых образов, $\{r_1, r_2, \dots, r_i\}$ – множество шаблонов.

Соответствие между тестовым образом слова t и шаблоном r устанавливается в том случае, если между ними есть общие символы.

При распознавании используется функция, показанная на (2.17).

$$\rho(r_i, t_i) = \begin{cases} 1, & r_i \neq t_i \\ 0, & r_i = t_i \end{cases} \quad (2.17)$$

Мерой схожести двух образов t и r , с учетом (2.17), будет являться:

$$v(\bar{r}, \bar{t}) = \min_s \mu(S), \text{ где } \mu(S) = \sum_{(i,j) \in S} \rho(r_i, t_i), \quad (2.18)$$

где соответствие S – двудольный граф, в котором отсутствуют некоторые вершины и ребра.

Далее рассматривается задача сравнения упорядоченного набора символов, в котором возможны следующие ошибки:

- неправильная замена символа (компьютер – помпьютер);
- неправильная вставка символа (компьютер – кмпьютер);
- неправильное удаление символа (компьютер – омпьютер).

Важным понятием при распознавании образов является понятие «расстояние редактирования». Это минимальное количество преобразований, которое необходимо провести над неким образом A , чтобы получить из него образ B .

$$D(A, B) = \min_j [C(j) + I(j) + R(j)] \quad (2.19)$$

Примем условия, показанные на (2.20):

$$d(i, j | i-1, j-1) = \begin{cases} 1, & \text{при } t(i) = r(i) \\ 0, & \text{при } y(i) \neq r(i) \end{cases}, \quad (2.20)$$

в этом случае, можно считать:

$$d(i, j | i-1, j) = d(i, j | i, j-1) = 1. \quad (2.21)$$

Эти данные можно представить в виде таблицы. Столбцы таблицы – образы символов, строки – шаблоны образов. Данную таблицу можно представить в виде графа. Тогда маршрут графа – это реализация

соответствия слов. Например, из точки (r_2, t_1) возможно три варианта перемещения: $(r_2, t_2), (r_3, t_2), (r_3, t_1)$. В этом случае задача распознавания образа превращается в задачу поиска кратчайшего пути в графе.

Описанные выше подходы используются компанией Google в предоставляемой ей API «Google Cloud Vision», которая используется в данной работе для выделения текста из анализируемых изображений.

2.6 Оценка тональности текста

Под оценкой тональности текста в нашем случае мы будем рассматривать эмоциональную составляющую текстовой информации. По принципу действия известные методы для оценки тональности текста можно разделить на следующие три группы.

1. Оценка эмоциональности текста на основе поиска присутствующих в нем термов в специализированных тональных словарях. Методы, относящиеся к данной группе, основаны на технологиях информационного поиска.

2. Классификация текста к одному из заранее известных классов эмоционального наполнения на основе анализа векторного представления текста. Данные методы основаны на алгоритмах машинного обучения, способных генерировать модели классификации данных.

3. Комбинирование технологий информационного поиска из первой группы методов с алгоритмами машинного обучения, используемыми во второй группе методов.

Алгоритм анализа эмоциональной составляющей, используемой в данной бакалаврской работе, относится к первой группе методов. Алгоритм состоит из следующих шагов.

1. Извлечение текста из изображения.
2. Выделение из полученного текста множество токенов (слов).

3. Поиск по имеющемуся словарю значения эмоциональной оценки для каждого токена.

4. Расчет на основе множества значений, полученных на предыдущем шаге, конечной числовой оценки эмоциональности всего текста.

При реализации алгоритма анализа тональности текста будет использована библиотека TextBlob. Данная библиотека предназначена для анализа текстовых данных и решения различных задач обработки естественного языка (Natural language processing). На основе использования данной библиотеки на языке программирования Python будет разработано программное обеспечение, дающее числовую оценку тональность текста. На выходе программа будет выдавать число в диапазоне от -1 до +1, где отрицательные значения соответствуют негативной тональности, а положительные позитивной тональности.

Вывод по второй главе

Были рассмотрены подходы к анализу изображений по цветовому распределению, а также подходы к анализу тональности текста. Выбраны библиотеки, которые будут использованы при реализации программного обеспечения: библиотека TextBlob для Python и API «Google Cloud Vision».

3 АНАЛИЗ ЭМОЦИОНАЛЬНОЙ СОСТАВЛЯЮЩЕЙ ИЗОБРАЖЕНИЙ ИЗ СОЦИАЛЬНОЙ СЕТИ ВКОНТАКТЕ

3.1 Алгоритм работы программного обеспечения

На языке программирования Python было разработано программное обеспечение, позволяющее проводить оценку эмоционального состояния выбранного пользователя социальной сети «ВКонтакте» на основе анализа размещённых им в открытом доступе изображений.

Программа работает по следующему алгоритму, представленному на рисунке 3.1. На первом этапе с использованием API социальной сети «ВКонтакте» производится аутентификация приложения в социальной сети и загрузка фотографий выбранного пользователя.

На следующем этапе производится отдельный анализ всех полученных изображений. Для этого каждое изображение с использованием API «Google Cloud Vision» проверяется на наличие текста. Если текст найден, то он распознаётся с помощью облачного сервиса Google. Распознанный текст анализируется с помощью библиотеки TextBlob для получения числовой оценки тональности текста. На выходе такой оценки получаем число из диапазона от -1 до +1. Отрицательное значение означает, что эмоциональная окраска текста негативная, положительное значение говорит о том, что текст позитивный, а значения около 0 говорят о нейтральности текста.

Следующим шагом является оценка эмоциональной составляющей на основе цветового распределения. Для этого производится кластеризация пикселей и определяется 3 доминирующих цвета по таблице X11 color names. Считается, что доминирующие цвета определяют эмоциональную составляющую изображения. Последним шагом является объединение текстовой оценки и цветового распределения для отнесения анализируемого изображения к одному из 4 кластеров.

Эта процедура повторяется для всех изображений пользователя.
Результат выводится на экран.



Рисунок 3.1 – Блок-схема анализа изображений

В данном исследовании принято, что существует 4 кластера эмоциональной составляющей изображения. Их перечисление с описанием представлено в таблице 1, распределение текстовой оценки по кластерам – в таблице 2, распределение доминирующих цветов на изображении по кластерам – в таблице 3 (название цветов в соответствии с X11 color names):

Таблица 1 – Характеристики кластеров по эмоциональной составляющей

Номер кластера	Описание
Кластер №1	Положительные эмоции, приподнятое настроение, чувство радости, дружелюбность, игривость, позитив.
Кластер №2	Грусть, уныние подавленность, чувство печали, плохое настроение.
Кластер №3	Негативные эмоции, злость, ненависть, агрессия, раздражение
Кластер №4	Эмоциональная нейтральность, чувство спокойствия, чувство душевного комфорта.

Таблица 2 – Распределение значений оценки тональности текста

Номер кластера	Диапазон значений оценки тональности текста
Кластер №1	[+0,2; +1,0]
Кластер №2	[-1,0; -0,2] дальнейшее уточнение кластера производится на основе цветового распределения
Кластер №3	
Кластер №4	(-0,2; +0,2)

Таблица 3 – Распределение доминирующих цветов по кластерам

Кластер №1
Crimson, Salmon, Red, HotPink, DeepPink, Tomato, OrangeRed, DarkOrange, Gold, Yellow, PeachPuff, Violet, Fuchsia, Magenta, DarkMagenta, Purple, FireBrick, SandyBrown, Goldenrod, DarkGoldenRod, Chocolate, Brown, Maroon, GreenYellow, Chartreuse, LawnGreen, Lime, LimeGreen, MediumSpringGreen, SpringGreen, ForestGreen, Green, DarkGreen, YellowGreen, MediumAquamarine, Aqua, Cyan, DeepSkyBlue, DodgerBlue, Blue, White, Snow, Honeydew, MintCream, Azure, AliceBlue, GhostWhite, WhiteSmoke, Seashell, Beige, OldLace, FloralWhite, Ivory, AntiqueWhite, Linen, LavenderBlush, MistyRose, Gainsboro, LightGrey, Silver, Pink
Кластер №2
DarkSalmon, MediumVioletRed, DarkKhaki, BlueViolet, DarkViolet, DarkOrchid, DarkSlateBlue, SaddleBrown, SeaGreen, DarkCyan, Teal, LightCyan, CadetBlue, SteelBlue, LightSteelBlue, LightSkyBlue, CornflowerBlue, MediumSlateBlue, RoyalBlue, MediumBlue, Gainsboro, LightGrey, Silver, DarkGrey, Grey, DimGrey, LightSlateGrey, SlateGrey, DarkSlateGrey, Black
Кластер №3
Red, Indigo, DarkRed, DarkOliveGreen, DarkBlue, Navy, MidnightBlue, Maroon, Black, DarkSlateGrey, DimGrey, Grey
Кластер №4
IndianRed, LightCoral, LightSalmon, LightPink, PaleVioletRed, Orange, Coral, LemonChiffon, LightGoldenrodYellow, PapayaWhip, Moccasin, PaleGoldenrod, Khaki, LightYellow, Lavender, Thistle, Plum, Orchid, MediumOrchid, MediumPurple, SlateBlue, Cornsilk, BlanchedAlmond, Bisque, NavajoWhite, Wheat, BurlyWood, Tan, RosyBrown, Peru, Sienna, PaleGreen, LightGreen, MediumSeaGreen, Green, OliveDrab, Olive, LightSeaGreen, LightCyan, PaleTurquoise, Aquamarine, Turquoise, MediumTurquoise, DarkTurquoise, PowderBlue, LightBlue, SkyBlue, Blue, White, Snow, Honeydew, MintCream, Azure, AliceBlue, GhostWhite, WhiteSmoke, Seashell, Beige, OldLace, FloralWhite, Ivory, AntiqueWhite, Linen, LavenderBlush, MistyRose, Gainsboro, LightGrey, Silver, DarkGrey, Grey, DimGrey, LightSlateGrey, SlateGrey, DarkSlateGrey

3.2 Тестирование программного обеспечения

Для тестирования работы программного обеспечения было выбрано несколько изображений. В первом тестовом изображении не содержится текст (рисунок 3.2).



Рисунок 3.2 – Тестовое изображение №1 (без текста)

В своем выводе программа определила следующие доминирующие цвета (здесь и далее применяются стандартизированные названия цветов «X11 color names»): deeppink, gold, goldenrod.

Последняя строчка вывода «оценка: 0» сообщает, что изображение относится к кластеру №1 – позитивные изображения (рисунок 3.3).

```
for i in range(0,len(list_file)) > else > if text_img>0.1
Debug: main_analis_img x
Debugger Console +
goldenrod gold deeppink
eJpdFFeQmEc.jpg
цветовая оценка: 0
```

Рисунок 3.3 – Вывод программы на тестовом изображении №1

Во втором тестовом изображении также не содержится текст (рисунок 3.4):

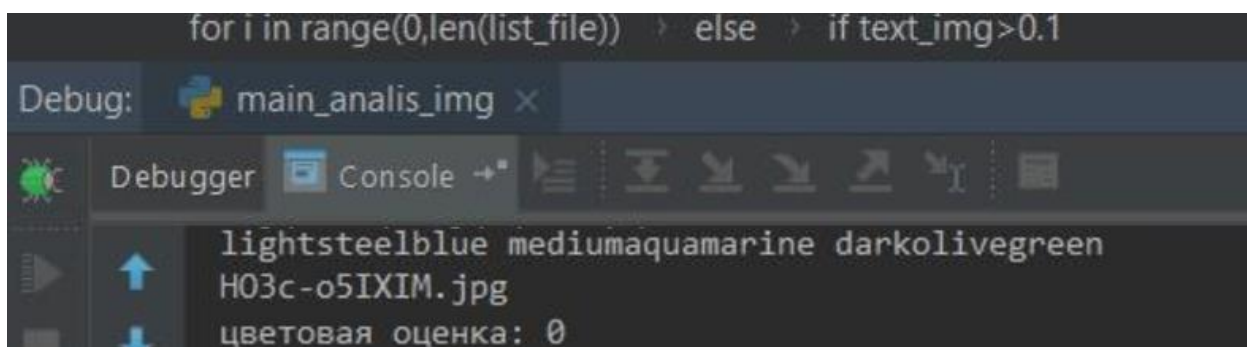


Рисунок 3.4 – Тестовое изображение №2 (без текста)

В своем выводе программа определила следующие доминирующие цвета («X11 color names»): darkolivegreen, lightsteelblue, mediumquamarine.

Последняя строчка вывода «оценка: 0» сообщает, что изображение относится к кластеру №1 – позитивные изображения (рисунок 3.5).

```
for i in range(0,len(list_file)) > else > if text_img>0.1
```



```
Debug: main_analis_img x
```

```
lightsteelblue medianaquamarine darkolivegreen  
H03c-o5IXIM.jpg  
цветовая оценка: 0
```

Рисунок 3.5 – Вывод программы на тестовом изображении №2

На третьем тестовом изображении содержится текст, поэтому программа должна проводить анализ на основе содержания текста (рисунок 3.6).

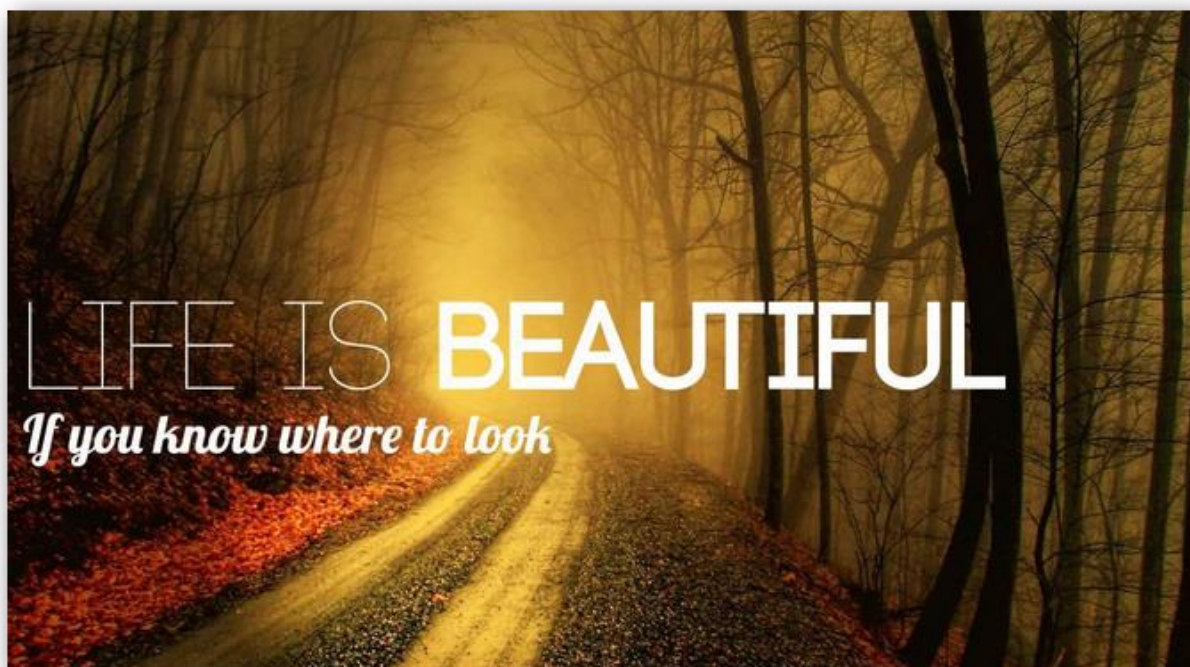


Рисунок 3.6 – Тестовое изображение №3 (с текстом «Life is beautiful»)

В своём выводе программа определила следующие доминирующие цвета («X11 color names»): whitesmoke, lightgrey, black. Но эмоциональная оценка изображения в данном случае проводится по тональности текста.

Последняя строчка вывода «оценка: 0.85» сообщает, что текст является позитивным (значение >0) (рисунок 3.7).

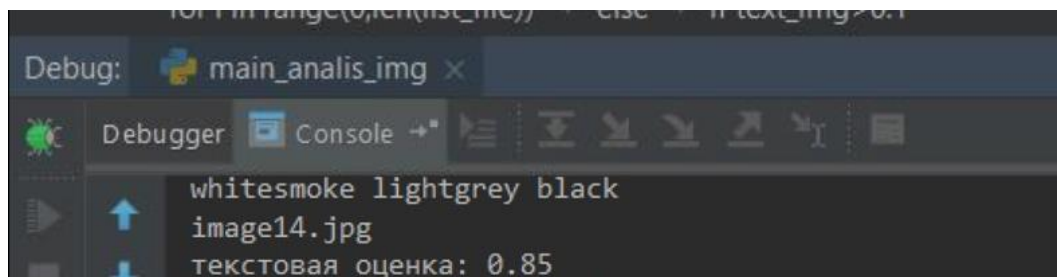


Рисунок 3.7 – Вывод программы на тестовом изображении №3

На четвертом тестовом изображении также содержится текст (рисунок 3.8).

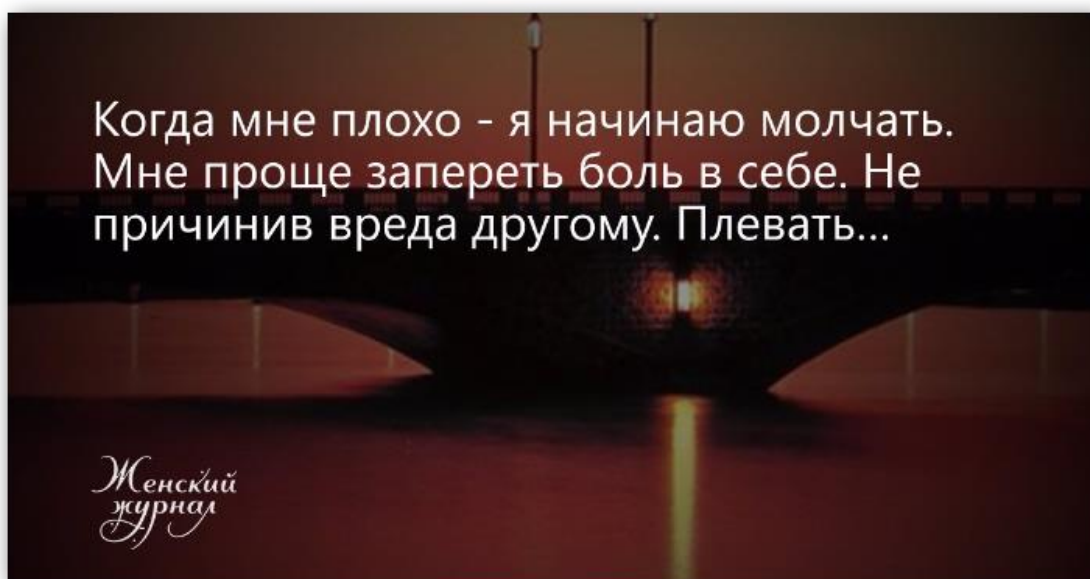


Рисунок 3.8 – Тестовое изображение №4 (с текстом)

В своем выводе программа определила следующие доминирующие цвета («X11 color names»): black, dimgrey. Эмоциональная оценка изображения в данном случае проводится по тональности текста.

Последняя строчка вывода «оценка: -0.306» сообщает, что текст является негативным (значение <0) (рисунок 3.9).

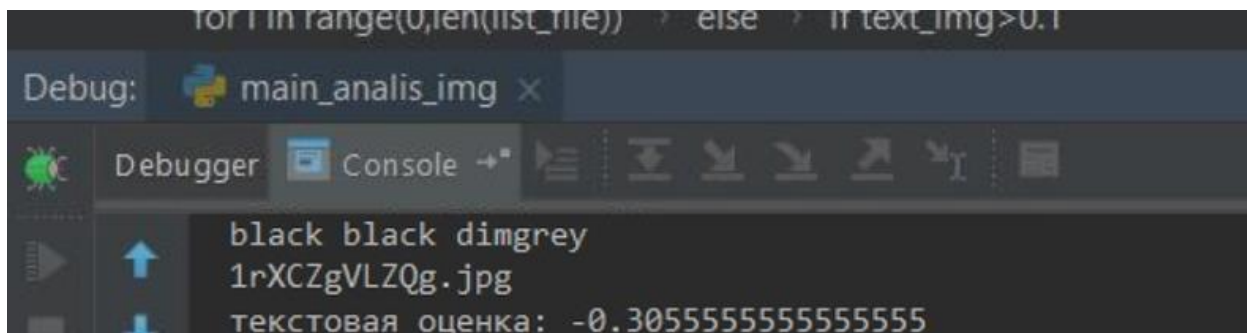


Рисунок 3.9 – Вывод программы на тестовом изображении №4

Теперь, когда ясно, что программа правильно анализирует отдельные изображения, проверим её работу в условиях анализа 12 фотографий пользователя социальной сети «ВКонтакте». Результат выполнения показан на рисунке 3.10.

```
↑ image14.jpg
текстовая оценка: 0.85
↓ dimgrey darkslategrey grey
qsCAPOBApiE.jpg
цветовая оценка: 2
silver darkslategrey dimgrey
quj_bFOAetA.jpg
цветовая оценка: 3
white rosybrown silver
rsLm5vSp0c4.jpg
цветовая оценка: 3
darkslategrey dimgrey gainsboro
vXadUCoTu7w.jpg
цветовая оценка: 3
[3, 1, 2, 6]
```

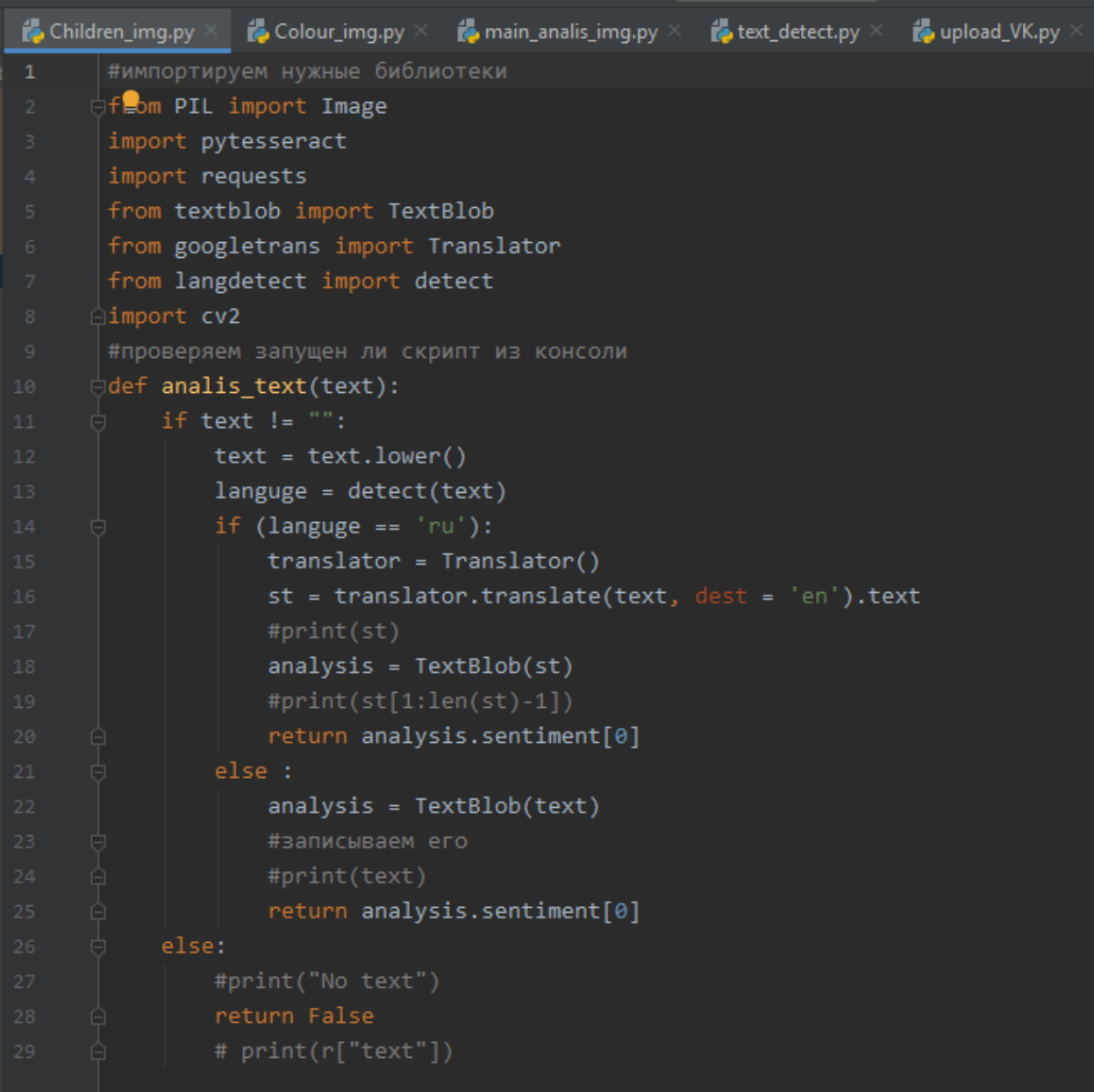
IDE and Plugin updates: PyCharm is ready to update.

Рисунок 3.10 – Распределение по кластерам 13 изображений пользователя социальной сети «ВКонтакте»

В результате выполнения программа распределяет 12 изображений по 4 кластерам (таблица 1). Как видим, 4 изображения были отнесены к первому кластеру, 1 изображение ко второму, 2 изображения к третьему и 6 изображений к четвертому. Больше всего изображений в четвертом кластере, значит можно сделать вывод, что данный пользователь в настоящее время проявляет эмоциональную нейтральность.

3.3 Программный код

Ниже приведен программный код разработанного приложения (рисунки 3.11 – 3.14). Дадим описание программного кода.

The image shows a screenshot of a code editor with five tabs: Children_img.py, Colour_img.py, main_analis_img.py, text_detect.py, and upload_VK.py. The active tab is text_detect.py, which contains the following Python code:

```
1 #импортируем нужные библиотеки
2 from PIL import Image
3 import pytesseract
4 import requests
5 from textblob import TextBlob
6 from googletrans import Translator
7 from langdetect import detect
8 import cv2
9 #проверяем запущен ли скрипт из консоли
10 def analis_text(text):
11     if text != "":
12         text = text.lower()
13         language = detect(text)
14         if (language == 'ru'):
15             translator = Translator()
16             st = translator.translate(text, dest = 'en').text
17             #print(st)
18             analysis = TextBlob(st)
19             #print(st[1:len(st)-1])
20             return analysis.sentiment[0]
21         else :
22             analysis = TextBlob(text)
23             #записываем его
24             #print(text)
25             return analysis.sentiment[0]
26     else:
27         #print("No text")
28         return False
29     # print(r["text"])
```

Рисунок 3.11 – Программный код, отвечающий за получение числовой оценки тональности текста

Логически программный код разделен между 4 файлам: children_img.py, color_img.py, main_analis.py, text_detect.py и upload_vk.py.

В файле children_img.py содержится программный код, отвечающий за получение числовой оценки тональности текста. Для этого используются отдельные функции библиотеки textblob. Так как данная библиотека работает только с английским текстом, то весь текст на русском языке переводиться с помощью оброчных сервисов Google. За это отвечает библиотека googletans.

```
Children_img.py x Colour_img.py x main_analis_img.py x text_detect.py x upload_VK.py x
1 from __future__ import print_function
2 import binascii
3 import struct
4 from PIL import Image
5 import numpy as np
6 import scipy
7 import scipy.misc
8 import webcolors
9 import scipy.cluster
10
11 NUM_CLUSTERS = 5
12 def closest_colour(requested_colour):
13     min_colours = {}
14     for key, name in webcolors.css3_hex_to_names.items():
15         r_c, g_c, b_c = webcolors.hex_to_rgb(key)
16         rd = (r_c - requested_colour[0]) ** 2
17         gd = (g_c - requested_colour[1]) ** 2
18         bd = (b_c - requested_colour[2]) ** 2
19         min_colours[(rd + gd + bd)] = name
20     return min_colours[min(min_colours.keys())]
21 def get_colour_name(requested_colour):
22     try:
23         closest_name = actual_name = webcolors.rgb_to_name(requested_colour)
24     except ValueError:
25         closest_name = closest_colour(requested_colour)
26         actual_name = None
27     return actual_name, closest_name
28 def analis_img(path):
29     im = Image.open(path)
30     im = im.resize((150, 150)) # optional, to reduce time
31     ar = np.asarray(im)
32     shape = ar.shape
33     ar = ar.reshape(scipy.product(shape[:2]), shape[2]).astype(float)
34     if im.mode == 'P':
35         im.putalpha(0)
```

Рисунок 3.12 – Программный код, отвечающий за определение доминирующих цветов изображения

В файле `colour_img.py` содержится программный код, отвечающий за определение доминирующих цветов изображения. Для работы с изображениями используется библиотека `pil`. Как было описано во второй главе, доминирующие цвета определяются путем кластеризации данных. Для этого подключается библиотека `scipy`, в которой содержатся реализации алгоритмов кластеризации.

```
Children_img.py x Colour_img.py x main_analis_img.py x text_detect.py x upload_VK.py x
1 import os
2 import text_detect
3 import Colour_img
4 import Children_img
5 import upload_VK
6 login = ''#необходим логин от vk.com
7 password = '' #пароль от vk.com
8 vk_id = ''#id созданного приложения vk.com
9 url = "" #ссылку на альбомы пользователей
10 id = ''#id пользователя
11 photo_folder = upload_VK.auth_session_vk(login,password,vk_id,url,id)
12 list_file = os.listdir(path=photo_folder)
13 kol = [0, 0, 0, 0]
14 l=0
15 p=0
16 for i in range(0,len(list_file)):
17     #text_img = Children_img.analis_text(text_detect.detect_text(photo_folder+"
18     text_img = False
19     colour_img = Colour_img.analis_img(photo_folder+"/"+list_file[i])
20     if text_img == False:
21         kol[colour_img] += 1
22         print(list_file[i])
23         print("цветовая оценка:", colour_img)
24     else:
25         if text_img < -0.5:
26             kol[2] +=1
27             print(list_file[i])
28             print("текстовая оценка:", text_img)
29         if -0.5< text_img <-0.2 :
30             kol[1] +=1
31             print(list_file[i])
32             print("текстовая оценка:", text_img)
33         if -0.2< text_img <0.2 :
34             kol[3] +=1
35             print(list_file[i])
36             print("текстовая оценка:", text_img)
```

Рисунок 3.13 – Программный код, отвечающий за логику работы всего приложения, а также управляющий процессом кластеризации изображений

```
Children_img.py × Colour_img.py × main_analis_img.py × text_detect.py × upload_VK.py ×
1 from urllib.request import urlretrieve
2 from collections import defaultdict
3 import vk
4 import os
5 import time
6 import math
7 # Авторизация
8 def auth_session_vk(login,password,vk_id,url,id):
9     session = vk.AuthSession(app_id=vk_id, user_login=login, user_password=password)
10    api = vk.API(session, v='5.95')
11    # Разбираем ссылку
12    # #album_id = url.split('/')[-1].split('_')[1]
13    owner_id = url.split('/')[-1].split('_')[0].replace('albums', '')
14    photos_count = api.photos.getAll(owner_id=owner_id)['count']
15    counter = 0 # текущий счетчик
16    prog = 0 # процент загруженных
17    broken = 0 # не загружено из-за ошибки
18    time_now = time.time() # время старта
19    owner_name = api.users.get(user_ids = id)[0]['first_name']
20    owner_surname = api.users.get(user_ids = id)[0]['last_name']
21    # &nbsp;Создадим каталоги
22    if not os.path.exists('saved'):
23        os.mkdir('saved')
24    photo_folder = 'saved/album_{0}_{1} ({2})'.format(owner_name, owner_surname, ow
25    if not os.path.exists(photo_folder):
26        os.mkdir(photo_folder)
27    for j in range(math.ceil(photos_count / 1000)): # Подсчитаем&nbsp;сколько раз ну
28        photos = api.photos.getAll(owner_id=owner_id, count=200, offset=j * 1000) #
```

Рисунок 3.14 – Программный код получения изображений с помощью
VK API

В файле text_detect.py содержатся функции для распознавания текста на изображениях. Для этого производится подключение к сервисам Google Cloud Vision, куда отправляются анализируемые изображения. В файле upload_VK.py содержатся функции для работы с VK API, в том числе загрузки изображений выбранного пользователя.

За логику работы всего приложения отвечает программный код, содержащийся в файле main_analis.py.

Вывод по третьей главе

На языке программирования Python было разработано программное обеспечение, позволяющее проводить оценку эмоционального состояния выбранного пользователя социальной сети «ВКонтакте» на основе анализа размещённых им в открытом доступе изображений. При этом использовались библиотека TextBlob для Python и API «Google Cloud Vision».

ЗАКЛЮЧЕНИЕ

В ходе работы была проанализирована различная литература по теме исследования. Рассмотрен анализ изображений с точки зрения психологии, проанализировано воздействие цветового распределения на психоэмоциональное состояние.

Был разработан алгоритм классификации эмоциональной составляющей изображений. Алгоритм включает в себя автоматизированный сбор изображений из социальной сети «ВКонтакте», определение в изображениях доминирующих цветов путем кластеризации пикселей методом k-means, определение наличия текста на изображениях и анализ его тональности. Выходными данными алгоритма является классификация изображений по четырем классам: «позитивные» изображения, «нейтральные», «грустные» и «агрессивные».

Предложенный алгоритм был реализован на языке программирования Python с использованием VK API для получения изображений выбранного пользователя, API «Google Cloud Vision» для обнаружения и распознавания текста и библиотеки TextBlob для анализа тональности текста.

Разработанный алгоритм классификации изображений контента аккаунта социальной сети может быть использован в образовательных учреждениях с целью своевременного контроля перепадов психического состояния учащихся для поддержания безопасности окружающих. Также будет удобен кадровым специалистам в выявлении наиболее подходящего кандидата на конкретную вакансию.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Иоганн Вольфганг Гёте и его учение о цвете (часть первая) / С.В. Месяц. – М.: Кругъ, 2012. – 464 с. - ISBN 978-5-7396-0250-3. – Текст : непосредственный.
2. Digital 2020: Global digital overview. – Текст : электронный. – URL: <https://datareportal.com/reports/digital-2020-global-digital-overview> (дата обращения: 28.04.2020).
3. Statcounter Globalstats. Social Media Stats Russian Federation. – Текст : электронный. – URL: <https://gs.statcounter.com/social-media-stats/all/russian-federation> (дата обращения: 28.04.2020).
4. Sheng, G. High-resolutionsatellite scene classification using a sparse coding based multiple featurecombination / G. Sheng, W. Yang, T. Xu, H. Sun // International Journal of Remote Sensing, 2012, vol. 33(8). – Pp. 2395-2412.
5. Глумов, Н.И. Метод отбора информативных признаков на цифровых изображениях / Н.И. Глумов // Компьютерная оптика, 2007, Т. 31, № 3. – С. 73-76. Текст : непосредственный.
6. Гайдель, А.В. Возможности текстурного анализа компьютерных томограмм в диагностике хронической обструктивной болезни / А.В. Гайдель, П.М. Зельтер, А.В. Капишников, А.Г. Храмов // Компьютерная оптика, 2014, Т. 38, № 4. – С. 843-850. Текст : непосредственный.
7. Цветовой тест Люшера / М. Люшер. – М.: Экспо, 2002. – 192 с. – ISBN 5-699-11508-6. – Текст : непосредственный.
8. Fu, L. A Robust Text Segmentation Approach in Complex Background Based on Multiple Constraints [Text] / L. Fu, W. Wang, Y. Zhan // Pacific-Rim Conference on Multimedia – 6th Pacific Rim Conference on Multimedia, Jeju Island, Korea, November 13-16, 2005, Proceedings, Part I: Advances in Multimedia Information Processing – PCM 2005. – Springer-Verlag Berlin Heidelberg 2005. – Pp. 594-605.

9. Gllavata, J. Adaptive Fuzzy Text Segmentation in Images with Complex Backgrounds Using Color and Texture [Text] / J. Gllavata, B. Freisleben // International Conference on Computer Analysis of Images and Patterns – 11th International Conference, CAIP 2005, Versailles, France, September 5-8, 2005. – Proceedings: Computer Analysis of Images and Patterns. – Springer-Verlag Berlin Heidelberg 2005. – Pp. 756-765.

10. Yalniz, I.Z. Efficient Exploration of Text Regions in Natural Scene Images Using Adaptive Image Sampling [Text] / I.Z. Yalniz, D. Gray, R. Manmatha // European Conference on Computer Vision. – ECCV 2016: Computer Vision – ECCV 2016 Workshops. – 2016. – Pp. 427-439.

11. Андрианов А.И. Локализация текста на изображениях сложных графических сцен // Современные проблемы науки и образования. – № 3. – 2013. – Текст : электронный. URL: <http://science-education.ru/ru/article/view?id=9311> (дата обращения: 28.04.2020).

12. Болотова, Ю.А. Обзор алгоритмов детектирования текстовых областей на изображениях и видеозаписях / Ю.А. Болотова, В.Г. Спицын, П.М. Осина // Компьютерная оптика. – 2017. – Т. 41, № 3. – С. 441-452. Текст : непосредственный.

13. Matas, J. Robust wide baseline stereo from maximally stable extremal regions [Text] / J. Matas, O. Chum, M. Urban, T. Pajdl // British Machine Vision Conference. – 2002. Pp. 384-393.

14. Ха, Л.М. Свёрточная нейронная сеть для решения задачи классификации / Л.М. Ха // Труды МФТИ. – 2016. – Т. 8, № 3. – С. 91-97. Текст : непосредственный.

15. Наумов, С.В. Методы решения задачи детекции текста на изображениях / С.В. Наумов, В.В. Прокопова, А.Б. Тельбухов, Д.О. Инишева // Молодой учёный. Международный научный журнал. – 2019. – № 18 (256). – С. 93-95. Текст : непосредственный.

16. Kumar, D. Multi-script robust reading competition in ICDAR 2013 [Text] / D. Kumar, A. Prasad, A.G. Ramakrishnan // International Workshop on Multilingual OCR (MOCR 2013). – Washington DC, USA, 2013.

17. Фёдорова, А.А. Распознавание английского текста свёрточной нейронной сетью / А.А. Фёдорова // Молодой учёный. Международный научный журнал. – 2016. – № 14 (118). – С. 97-102. Текст : непосредственный.

18. Тюрин, А.Г. Кластерный анализ, методы и алгоритмы кластеризации / А.Г. Тюрин, И.О. Зуев // Вестник МГТУ МИРЭА. – 2014. – №2, выпуск 3. – С. 86-97. Текст : непосредственный.

19. MacQueen, J.B. Some methods for classification and analysis of multivariate observations [Text] / J.B. MacQueen // Proceedings of 5th Berkeley Mathematical Statistics and Probability. University of California Press, 1967. Pp. 281-297.

20. Steinhaus, H. Sur la division des corps matériels en parties [Text] / H. Steinhaus // Bull. Acad. Polon. Sci, 1957. Pp. 801-804.