МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра вычислительных технологий**

**КУРСОВАЯ РАБОТА**

**РАСПОЗНАВАНИЕ ЛИЦ С ПОМОЩЬЮ БИБЛИОТЕКИ OPENCV**

Работу выполнил\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Петренко А.Г.

(подпись)

Направление 02.03.02 — Фундаментальные информатика и информационные технологии

Направленность (профиль) Вычислительные технологии

Научный руководитель

д.т.н., проф \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Вишняков Ю.М.

(подпись)

Нормоконтролер

канд.техн.наук, доц.\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Е.Е.Полупанова

(подпись)

Краснодар

2018

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 2](#_Toc533271255)

[1 Теоретический материал по теме «Распознавание лиц» 5](#_Toc533271256)

[1.1. Распознавание лиц человеком 5](#_Toc533271257)

[1.2. Эксперимент А.Л. Ярбуса 7](#_Toc533271258)

[1.3 Распознавание лиц машиной 9](#_Toc533271259)

[2 Распознавание лиц с библиотекой opencv 17](#_Toc533271260)

[2.1 Признаки Хаара 17](#_Toc533271261)

[2.2 Метод Виолы-Джонса 18](#_Toc533271262)

[2.3 Использование признаков Хаара 20](#_Toc533271263)

[3 Описание и реализация алгоритма 24](#_Toc533271260)

[3.1 Распознавание лица с камеры 24](#_Toc533271261)

[3.2 Создание базы лиц 29](#_Toc533271262)

[3.3 Обучение распознавателя 31](#_Toc533271263)

[3.4 Определение лица 32](#_Toc533271263)

[Заключение 35](#_Toc533271264)

[Список использованных источников 36](#_Toc533271264)

**ВВЕДЕНИЕ**

В настоящее время для идентификации человека используются биометрические методы.  В связи с простотой распознавания лица и обилием камер во всех аспектах жизни человека, все более актуальным становятся разработки в распознавании лиц. Задача распознавания лиц актуальна как в области интеллектуальных сред, так и в системах безопасности. В связи с востребованностью разработкой алгоритмов распознавания лиц работают и крупные компании, но среди продуктов с открытым исходным кодом можно выделить OpenCV. Это библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения.

Распознавание лица можно разбить на три пункта:

1) yайти лицо в режиме реального времени;

2) cравнить найденное лицо с лицами, хранящимися в базе данных;

3) cравнить найденное лицо с эталонным в базе данных.

Но если человек может определить и сравнить лицо с лицами, хранящимися в памяти, за доли секунды, то без должной технологии распознавания машина не сможет отличить человека от столба. Человек узнает знакомое лицо, ориентируясь на индивидуальные черты, а именно расстояние между глаз, их цвет, высота губ и их ширина. Для начала компьютер должен не просто распознать человека, но и понять, что находится перед ним, лицо ли это или ваза. При том ракурс, с которого камера принимает лицо человека, игра света или же лишние предметы на лице человека играют огромную роль.

В своей работе я рассматриваю распознавание лиц людей с использованием локальных бинарных шаблонов (Local Binary Patterns - LBP). Оператор LBP может быть использован для поиска объекта на изображении (например лица), а также проверки этого объекта на принадлежность некоторому классу (верификация, распознавание эмоций, пола по лицу).

В первой части курсовой работы будет рассмотрен теоретический материал по распознаванию лиц, такой как определения, понятия, распознавание лица человеком и машиной и практическое использование. Во второй части речь пойдет и практической реализации.

Объектом исследования является распознавания лица в анфас, определение человека в реальном времени, обучение программы и точность самого распознавания.

Поставленной целью будет создание такой программы с помощью библиотеки opencv, языка высокого уровня Python и среды разработки PyCharm.

Для этого выполним ряд поставленных задач.

задачи:

1) рассмотреть теоретический материл по теме Распознавание лиц;

2) распознавание лиц с библиотекой opencv;

3) предложить и реализовать алгоритм на языке Python.

1. **Теоретический материал по теме «Распознавание лиц»**
   1. **Распознавание лиц человеком**

Человек способен идентифицировать объекты окружающего мира в течение десятков миллисекунд. Такая высокая скорость распознавания предметов возможна потому, что наш мозг постоянно создает предположения по отношению того, что находится в поле зрения, и сравнивает эти прогнозы с поступающей извне информацией.

Существует три основных этапа в процессе распознавания человеческих лиц:

1) определение физических характеристик лица, на которое мы смотрим;  
2) определение личности человека, в ходе которого мы понимаем, знаком ли нам тот или иной человек;

3) мы распознаем человека, но все еще не знаем, известно ли нам его имя.

Исследователи обнаружили, что на каждом этапе активируются определенные участки мозга.

Распознавание лиц, как отмечают психологи, — это процесс, больше связанный с когнитивной стороной восприятия. Дело в том, что, по сути, распознавание лиц, происходит следующим образом: мозг постоянно сравнивает то, что видит, с тем, что хранится в его долговременной памяти. Как ни странно, почти все алгоритмы распознавания конкретных лиц, уже заложенных в их базу, работают именно так.

К примеру, когда мы будем смотреть на такой объект, как часы, мы признаем его засчет сравнения того, что мы видим, и тех характеристик,

которые присущи ментальному образу часов. Хотя не все часы одинаковы и некоторые копии этого объекта могут отличаться от ментального прототипа, у каждых часов есть набор присущих только ему ключевых характеристик, таких как минутная и часовая стрелка, циферблат, которые и помогают нам распознавать его. Затем образ объекта классифицируется и сохраняется в памяти под определенной категорией. Чем больше разновидностей часов запомнил наш мозг, тем проще ему будет распознать новый объект.

Классификация объекта — это этап, на котором, как правило, процесс распознавания подходит к концу, но в случае распознавания лиц — это только начало. Если часы достаточно признать часами, то человеческое лицо недостаточно просто признать человеческим лицом. Практически сразу же мы определяем пол и возраст человека, его расу и даже то, нравится он нам или нет. Кроме того, мы тут же определяем, знакомо ли нам это лицо. Если человек нам знаком, мы тут же начинаем извлекать из этого информацию, как и алгоритм распознавания лиц. Он может определить конкретного человека, после чего выдать доступ к информации, отправить сигнал в криминальные службы, в зависимости от того, для чего создан.

**1.2 Эксперимент А.Л. Ярбуса**

Прежде всего рассмотрим крайне интересные психофизические аспекты восприятия лица, в частности, вопрос о том, а как вообще человек воспринимает или распознает лицо.

Здесь нам помогут классические эксперименты А.Л.Ярбуса, основная особенность которых – независимость решения вопроса о роли движений глаз в зрительном восприятии от решения вопроса о регуляции самих движений глаз и их детерминации. Известно, что движения глаза представляют собой крайне сложную картину его активности и состоят, как

минимум, из нескольких форм движений: нистагмы, отдельные саккады, инверсионные нистагмы, синусоидальные колебания низкой .

При длительном наблюдении весь образ или его фрагменты поочередно то угасают, то появляются вновь. Такая фрагментация образа (поочередное угасание и восстановление его отдельных частей) зависит от характера и

В известной мере эти наблюдения подкрепляют теорию нейронных ансамблей, согласно которой для реализации способности к восприятию необходим опыт: тот или иной образ воспринимается в результате комбинаций отдельных следов в мозге, образовавшихся там ранее усвоенным элементам. Эти данные согласуются и с другими, казалось бы, противоположными теориями восприятия, согласно которым образ сразу, без всякого предварительного опыта воспринимается как целое, без какого-либо синтеза из отдельных частей, благодаря способности воспринимать «форму», «целостность» и «организацию».

Из этого и других экспериментальных данных следует важный вывод: при восприятии изображения на первый план выступает независимое поведение отдельных частей фигуры, то есть выделение групп структурных элементов или организованных структур.

Итак, движения глаз приводят к дестабилизации изображения на сетчатке. Однако мы, тем не менее, воспринимаем изображение стабильным, постоянным, несмотря на движение глаз, тела и рассматриваемых объектов. В том, как это реализуется мозгом, состоит так называемая проблема пространственной константности зрительного восприятия, которая разделяется на две: проблема константности, стабильности зрительного поля и проблема инвариантного восприятия объектов.

Пространственная константность восприятия формируется в результате специальной деятельности зрительной системы, которая может подразделяться на пять основных этапов:

1) формирование сетчатых изображений;

2) локальный анализ возбуждения фоторецепторного слоя;

3) непредметные механизмы инвариантности;

4) инвариантный синтез образа объекта;

5) анализ предметного окружения.

Основная роль движения глаз состоит в перемещении оси зрения так, чтобы изображение пристально рассматриваемого объекта (или части объекта) всегда оказывалось в центральной части сетчатки, в зоне наилучшего зрения фовеа. Именно в этой зоне имеется наибольшая плотность цветочувствительных рецепторов. Здесь же располагаются рецептивные поля тонических корковых нейронов, анализирующих форму и т.п.

Один и тот же объект сначала обнаруживается, первично обрабатывается периферией с передачей полученной информации в мозг по «быстрому» каналу, а затем, после скачка, он исследуется более детально (если это необходимо) путем размещения изображения в зоне фовеа. Детальная информация в этом случае передается в мозг уже по медленному каналу. При фиксации какого-либо участка изображения зрительной системой обрабатывается не только этот участок, проецирующийся на фовеа, но и получаемая с периферии информация, необходимая для расчета следующего скачка.

Тем не менее данный процесс не столь очевиден. Дело в том, глаз фиксирует основные фрагменты слабоконтрастного изображения, которые обычно рассматриваются как информативные признаки описания изображения и которые в дальнейшем используются в формальных логических утверждениях.

Особенности восприятия слабоконтрастных изображений зрительным трактом человека заключаются также в том, что в процессе узнавания мозг выступает как активная распознающая система (с проверкой правильности решений). Этапами распознавания при этом могут быть: выделение признаков, предварительный анализ, выдвижение гипотезы, проверка гипотезы – сличение изображений с эталоном, взятым из памяти. Поэтому

принцип активного распознавания, должен закладываться на этапе первичного проектирования систем кибернетического видения слабоконтрастных объектов, в частности при распознавании человеческих лиц.

**1.3 Распознавание лиц машиной**

Задача идентификации и распознавания лиц – это одна из первых практических задач, которая стимулировала становление и развитие теории распознавания и идентификации объектов. Существует девять категорий объектов, которые соответствуют гностическим областям и вызывают зрительные образы:

1) объекты, которыми можно манипулировать;

2) объекты, которыми можно частично манипулировать;

3) объекты не манипулируемые;

4) лица;

5) выражения лиц;

6) живые существа;

7) печатные знаки;

8) рукописные изображения;

9) характеристики и расположение источников света.

Интерес к процедурам, лежащим в основе процесса узнавания и распознавания лиц, всегда был значительным, особенно в связи с возрастающими практическими потребностями: охранные системы, верификация, криминалистическая экспертиза, телеконференции и т.д. Несмотря на ясность того житейского факта, что человек хорошо идентифицирует лица людей, совсем не очевидно, как научить ЭВМ проводить эту процедуру, в том числе как декодировать и хранить цифровые изображения лиц. Еще менее ясными являются оценки схожести лиц, включая их комплексную обработку. Можно выделить несколько направлений

исследований проблемы распознавания лиц:

1) нейропсихологические модели;

2) нейрофизиологические модели;

3) информационно – процессуальные модели;

4) компьютерные модели распознавания.

Проблема распознавания лиц рассматривалась еще на ранних стадиях компьютерного зрения. Ряд компаний на протяжении более 40 лет активно разрабатывают автоматизированные, а сейчас и автоматические системы распознавания человеческих лиц: Smith & Wesson (система ASID – Automated Suspect Identification System); ImageWare (система FaceID); Imagis, Epic Solutions, Spillman, Miros (система Trueface); Vissage Technology (система Vissage Gallery); Visionics (система FaceIt).

Для решения задачи распознавания лиц были предложены различные методики, среди которых можно выделить подходы, основанные на нейронных сетях, на разложении Карунена – Лоэва, на алгебраических моментах, линиях одинаковой интенсивности, эластичных (деформируемые) эталонах сравнения. В разработках алгоритмов распознавания особые усилия направлены на автоматическое выделение элементов лица (глаза, нос, рот, подбородок и др.) на его различных изображениях: фас, профиль и произвольный ракурс. Далее эти геометрические характеристики используются в решении задачи распознавания. Типичным при описании этих подходов является отсутствие сравнения на статистически значимой базе данных лиц. Можно выделить два способа распознавания лиц:

1) сравнение типа соответствия между стимулами один против одного;

2) сравнение между накопленным, репрезентативным рядом лиц.

Геометрическое сравнение, основано на определении элементов лица – Элементы лица: глаза, нос, рот, подбородок и др. Лицо может быть распознано, даже когда индивидуальные Элементы лица видны недостаточно. Идея подхода заключается в нахождении относительного положения и собственных характеристик отдельных Элементы лица. Было показано, что

если даже Элементы лица извлекаются вручную, то компьютерное распознавание дает очень хорошие результаты.

Эталонное сравнение построено на идее, что изображение, представлено в виде массива байтов – величин интенсивности, сравнивается в подходящей метрике с эталоном – целым лицом. Существуют несколько путей подготовки эталонов и их представления. Несколько эталонов используются для распознавания с разных ракурсов.

Заслуживает внимания подход, когда лицо представляется в виде набора малых различных эталонов. Предпочтительным и более комплексным подходом является путь в использовании одного эталона совместно с точной априорной моделью, которая позволяет оценить трансформацию основного лица, при изменении ракурса наблюдения. Деформируемая модель затем используется в построении метрики сравнения эталонных лиц. Данная идея является основой методики деформируемых эталонов.

Схема эталонного сравнения в работе Bruce V. достаточно сильно модифицирована, чтобы называть ее корреляционно – экстремальной. Она использует нормализацию изображения, которое переводит его в карту величин градиентов и является свободной от карты краев. Одним из успешных находок является использование нескольких разрешений и малых по размерам эталонов для глаз, рта и носа. На этих подходах построены детекторы элементов лица. Важно отметить, что следующий шаг является конструктивным: сначала детектировать глаза (путем эталонного сравнения), потом автоматически нормализовать изображение по масштабу и ориентации.

Можно заметить, что такой подход содержит элементы распознавания на основе эталона всего лица: ЭЛ (глаза) используются для нормализации изображения, и эталонное сравнение проводится раздельно по отдельным характерным чертам лица (глаза, нос, рот). Однако как показали эксперименты, успешнее всего распознавание лица происходит на архитектуре, комбинирующей подход распознавания всего лица с подходом на основе эталонного сравнения его элементов.

Следует также рассмотреть схему распознавания, основанную на К-L декомпозиции. Отметим, что поскольку объекты распознавания в К-L декомпозиции представляются в виде линейной суммы базисных эталонов, то алгоритм распознавания не может дать лучше результата, чем корреляционный. Однако на этом пути можно значительно уменьшить вычислительные затраты, сравнимые со схемами распознавания на основе геометрических характеристик ЭЛ. Так Ellis H.D. показал, что снижение вычислительных затрат при том же уровне качества распознавания достигает 96%. Аналогичный алгоритм Т.Poggio работает лучше чем алгоритм R.J.Baron, поскольку использует более малые эталоны, что устойчивее к дисторсии изображения.

Представляет интерес схема распознавания на нейронных сетях. В частности, использование сети гипер базисных функций в синтезировании вектора признаков Элементы лица для распознавания 3D объектов с произвольного ракурса. В этом случае входами сети являются параметры Элементы лица, в том числе их позиция на изображении. Гипербазисная функциональная сеть имеет входы как амплитуды градиентов для каждого пикселя и как центры соответствующих эталонов различные центры при различных сдвигах, что напоминает описанную ранее схему сравнения эталонов Элементы лица. Это может соответствовать линейной классификации на гауссовских функциях корреляционных коэффициентов взамен просто метода максимума на коэффициентах корреляции.

Вопрос о зависимости результатов распознавания от ракурса съемки можно решать несколькими путями. Если для каждого человека имеются изображения, снятые с разных ракурсов, то можно использовать те же схемы распознавания, за счет увеличения вычислительных затрат. Использование гипер базисных функций – классификации с возможностью интерполяции между различными точками проекций достаточно рискованно. Однако в реальности, тем не менее, может быть только одно фронтальное изображение лица, доступное для генерации эталона. Очевидно, что одно изображение 3D

объекта (без теней) не содержит достаточной информации. Если, тем не менее, объект принадлежит классу сходных объектов (прототипов), для которых различные точки проекций известны, то возможно разумная экстраполяция и можно предложить корректную проекцию для данного объекта только по одной 2D проекции. Люди определенно способны распознавать лица, повернутые на 20-300 относительно фронтальной проекции. Возможно, они просто используют свои полученные данные о структуре типичного лица.

Другим вариантом решения этой проблемы является задача использования 3D моделей лица, для поддержки распознавания на не фронтальных изображениях лиц. Как указывает R.Brunelli, возможны постановки задач и их решения, включая отработки на экспертной БД, связанные с получением других проекций лица, используя знания о проекциях других типичных объектов этого класса.

Для различных контуров Элементы лица используются разные методы их извлечения на исходном портрете. Фигуры глаз и рта имеют устойчивые геометрические формы, поэтому они извлекаются в терминах модели деформируемого эталона. Другие элементы лица, такие как брови, нос и контур лица настолько изменчивы, что для их обнаружения применяется модель активного контура, которая устойчиво обнаруживает подобные объекты. На рис.3 показаны все элементы лица, которые используются при портретной экспертизе, и которые крайне желательно идентифицировать при автоматическом распознавании лица, что обеспечивает легитимность метода.

**Модель деформируемого эталона**. Деформируемые эталоны определяются параметрами, задаваемыми априорными знаниями об ожидаемой форме ЭЛ и которые определяются численно в процессе обучения при контурном дешифрировании.

Эталоны достаточно гибки при изменении их размеров и других задающих параметров, при этом их можно числено сравнивать, а полученные значения параметров можно использовать для описания конкретного

Элементы лица. Деформируемые эталоны взаимодействуют с текущим цифровым изображением в динамическом режиме. Энергетическая функция определяется набором компонентов, которые притягивают эталон к изображению Элементы лица на основе характеристик графиков срезов интенсивности, таких как максимумы и минимумы, краев и само значение интенсивности. Минимум энергетической функции соответствует лучшему выбору для данного изображения. Обычно деформируемые эталоны используют для обнаружения глаз и рта.

До начала поиска Элементы лица необходимо задать яркостные границы, которые можно было бы использовать для идентификации Элементы лица от других фрагментов лица и грубый контур каждого Элементы лица как начальный контур для последующих итераций. Обычно используют масштабный пространственный фильтр, для вычисления гистограммы и определения нулевых яркостей на различных масштабах, и метод грубой оценки контура для грубого определения положения контура объекта. Исключением является только грубый контур лица, который меньше чем его точный контур.

После того как получен грубый контур, происходит нахождение физического контура на каждом Элементы лица. Общепринятые детекторы краев не позволяют точно определить контуры глаз или рта, исходя только из локального набора краев. Дело в том, что обычные детекторы краев не позволяют синтезировать локальную информацию в целостный глобальный контур объекта. Поэтому проектирование детектора глаза основано на методе деформируемого эталона, который задается набором параметров, определяемым априорной информацией об ожидаемой форме и используемым в процессе обучения. Эти эталоны достаточно гибки и меняют свои размеры и форму путем вариации их значений параметров, так как эталон взаимодействует с изображением. Полученные значения величин параметров описания эталона используются для описания конкретного Элементы лица.

**Модель активного контура** модель змеи. Активный контур определяется как энергетически минимальный сплайн, обучаемый путем введения внешних   
притягивающих вынужденных потенциалов и влиянием потенциалов изображения, которые натягивают его на Элементы лица, на основе характеристик линий и краев. Змеи фиксируются на ближайших краях и более аккуратно и точно локализуясь в последующем. Поскольку змея есть энергетически минимальный сплайн (ЭМС), необходимо исследовать потенциальные функции, которые включают в себя локальные минимумы, а также альтернативные решения на более высоком уровне анализа процесса. Выбор необходимого решения будет достаточным, если выбрать путь добавления членов ряда, которые будут продвигать ЭМС по изображению для получения необходимого решения. Метод активных контуров (метод змей) обычно применяют для обнаружения и определения элементов лица, как брови, нос, овал лица. Формы бровей, ноздрей и овала лица, в отличие от глаз и рта, значительно различаются у разных людей и их контуры не могут быть определены с помощью деформируемых эталонов.

Для этих целей наиболее эффективной оказывается модель активного контура МАК. Активный контур змея есть энергетически минимальный сплайн, направляемый активными внешними силами и влиянием сил собственно изображением, которые натягивают змею на признаки Элементы лица. В качестве признаков здесь принимаются элементарные признаки изображения: линии и края. Начальная змея локализуется на ближайших краях, а затем точно определяется ее локализация и форма.

Если сравнить два подхода: идентификацию лиц на основе вектора признаков, представляющих собой геометрические характеристики ЭЛ и идентификацию лиц на основе сравнения полутоновых эталонов, то видно, что корреляционно – экстремальный подход на базе полутоновых эталонов работает эффективнее. Этот подход не требует специальных априорных знаний о структуре Элементы лица. В тоже время, методика, основанная на

характеристиках Элементы лица, дает значительную скорость распознавания, не требует специализированного программно-аппаратного обеспечения и больших объемов памяти.

1. **Распознавание лиц с библиотекой opencv**

Для начала давайте разберемся, как распознать лицо на фотографии. Во-первых, нужно найти, где на фото расположено лицо человека и не спутать его с часами на стене и кактусом на подоконнике. Казалось бы, простая задача для человека, оказывается не такой простой для компьютера. Для того, чтобы найти лицо мы должны выделить его основные компоненты, такие как нос, лоб, глаза, губы. Для этого будем использовать шаблоны, изображенные на рисунке 1.

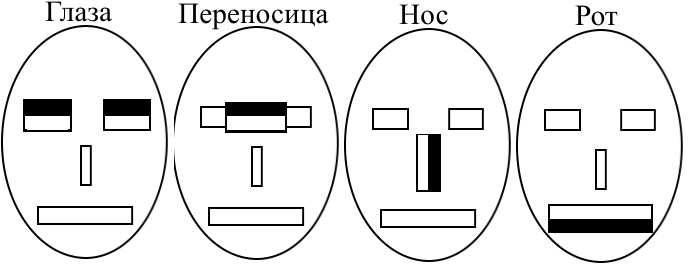


Рисунок 1 – Основные шаблоны

**2.1 Признаки Хаара**

Для начала я бы хотел рассмотреть признаки Хаара и их виды. Прямоугольные признаки Хаара. Простейший прямоугольный признак Хаара можно определить как разность сумм пикселей двух смежный областей внутри прямоугольника, который может занимать различные положения и масштабы на изображении. Такой вид признаков называется 2-прямоугольным. Виола и Джонс также определили 3-прямоугольные и 4-прямоугольные признаки. Каждый признак может показать наличие (или отсутствие) какой-либо конкретной характеристики изображения, такой как

границы или изменение текстур. Например, 2-прямоугольный признак может показать, где находится граница между темным и светлым регионами.

Наклонные признаки Хаара. Линхарт и Майд представили идею наклоненных под 45° признаков Хаара. Это было сделано для увеличения размерности пространства признаков. Способ оказался удачным и некоторые наклонные признаки были способны лучше описывать объект. Например, 2-прямоугольный наклонный признак Хаара может показать наличие края, наклоненного на 45 градусов. Мессом и Барзак дополнили концепцию наклонных признаков Хаара. Хоть идея и является математически верной, на практике при использовании признаков под разными углами возникают проблемы. Для ускорения вычислений, детектор использует изображения низкого разрешения, что приводит к ошибке округления. Исходя из этого, наклонные признаки Хаара обычно не используются.

**2.2 Метод Виолы-Джонса**

Обычно у каждого метода есть основа, то, без чего этот метод не мог бы существовать в принципе, а уже над этой основой строится вся остальная часть. В методе Виолы-Джонса эту основу составляют примитивы Хаара, представляющие собой разбивку заданной прямоугольной области на наборы разнотипных прямоугольных подобластей, изображенных на рисунке 2:

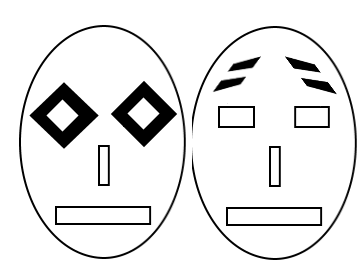


Рисунок 2 – Шаблоны с поворотом

В оригинальной версии алгоритма Виолы-Джонса использовались только примитивы без поворотов, а для вычисления значения признака сумма яркостей пикселей одной подобласти вычиталась из суммы яркостей другой подобласти. В развитии метода были предложены примитивы с наклоном на 45 градусов и несимметричных конфигураций. Также вместо вычисления обычной разности, было предложено приписывать каждой подобласти определенный вес и значения признака вычислять как взвешенную сумму пикселей разнотипных областей, используя формулу 1:

https://habrastorage.org/getpro/habr/post_images/412/e91/fab/412e91fabeb24e4a352cfa522f17cb52.gif (1)

Сложность вычисления признака так же как и получения значения пикселя остается O(1): значение каждой подобласти можно вычислить скомбинировав 4 значения интегрального представления (Summed Area Table — SAT), которое в свою очередь можно построить заранее один раз для всего изображения за O(n), здесь n — число пикселей в изображении, используя формулу 2.

https://habrastorage.org/getpro/habr/post_images/403/cfb/8db/403cfb8db58b1c33919002155e3b9f75.gif  
https://habrastorage.org/getpro/habr/post_images/3cd/12d/327/3cd12d3270f8c11b3076876dd84eac81.gif (2)

Это позволило создать быстрый алгоритм поиска объектов, который поль-зуется успехом уже больше десятилетия. Но вернемся к нашим признакам. Для определения принадлежности к классу в каждом каскаде, находиться сумма значений слабых классификаторов этого каскада. Каждый слабый классификатор выдает два значения в зависимости от того больше или меньше заданного порога значение признака, принадлежащего этому классификатору. В конце сумма значений слабых классификаторов срав-нивается с порогом каскада и выносится решения найден объект или нет данным каскадом.

**2.3 Использование признаков Хаара**

Если шаблоны, они же примитивы, соответствуют конкретным областям на изображении, будем считать, что на изображении есть человеческое лицо. Для каждого из них считается разность между яркостью белой и черной областей. Это значение сравнивается с эталоном и принимается решение о том, есть ли здесь часть человеческого лица или нет. Этот метод называется методом Виолы-Джонса (так же известен как каскады Хаара). Давайте представим, что у нас на фотографии не одно большое лицо, а много мелких. Если применить шаблоны ко всей картинке мы не найдем там лиц, т.к. они будут меньше шаблонов. Для того чтобы искать на всем фото лица разных размеров используется метод скользящего окна. Именно внутри этого окна и высчитываются примитивы. Окно как бы скользит по всему изображению. После каждого прохождения изображения окно увеличивается, чтобы найти лица большего масштаба.

Найдено лицо на фотографии, но для определения конкретного человека требуется выполнить еще несколько действий. Для решения этой задачи будем использовать алгоритм Local Binary Patterns. Суть его заключается в том, что мы разбиваем изображение на части и в каждой такой части каждый пиксель сравнивается с соседними 8 пикселями. Если значение центрального пикселя больше соседнего, то пишем 0, в противном случае. И так для каждого пикселя у нас получается некоторое число. Далее на основе этих чисел для всех частей, на которые мы разбивали фотографию, считается гистограмма. Все гистограммы со всех частей объединяются в один вектор характеризующий изображение в целом. Если мы хотим узнать насколько похожи два лица, нам придется вычислить для каждого из них такой вектор и сравнить их, вычисление вектора изображено на рисунке 3.

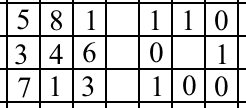


Рисунок 3 – расчет веса LBP

Вектор записывается в строку 11010001.

На рисунке 4 представлен алгоритм нахождения лица. Параметр cascadePath содержит имя файла с уже готовыми значениями для распознавания лиц. Этот файл был взят с GitHub.

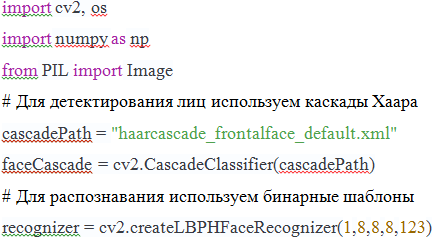


Рисунок 4 – Начало работы с opencv

Параметр cascadePath содержит имя файла с уже готовыми значениями для распознавания лиц. Этот файл был взят с GitHub.

Далее создаем объект CascadeClassifier и объект распознавания лиц LBPHFaceRecognizer. На последнем остановимся поподробнее, точнее, на его параметрах, которые схематично изображены на рисунке 5. Первые два значения 1 и 8 характеризуют окрестности пикселя. 1 – радиус выборки пикселей. 8 – количество выбираемых пикселей. Чем больше пикселей выбрано, тем точнее алгоритм распознает лицо.

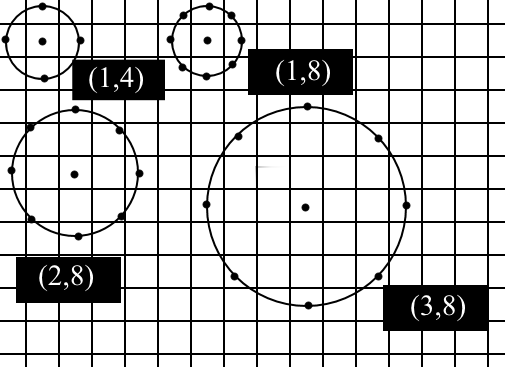
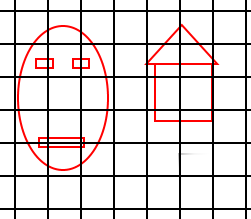


Рисунок 5 – радиус LBP

Следующие параметры (8,8) характеризуют размеры областей, изображенные на рисунке 6, на которые мы разбиваем исходное изображение с лицом. Чем оно меньше, тем больше будет таких областей и тем качественнее распознавание.



(8,8)

Рисунок 6 – Разбиение на области

И наконец, последнее значение это параметр confidence threshold, определяющий пороговое значение для распознавания лица. Чем меньше confidence тем больше алгоритм уверен в том, что на фотографии изображено известное ему лицо. Порог означает, что когда уверенности мало, алгоритм просто считает это лицо незнакомым. В данном случае порог равен 123.

1. **Описание и реализация алгоритма**

В начале необходимо познакомиться с простым алгоритмом распознавания лица.

Для того, чтобы определить конкретного человека в режиме реального времени потребуется создать базу его фотографий. После чего перевести его в формат yml, для того, чтобы алгоритм смог сравнить кадры, которые он имеет в базе и те, что поступают с камеры. В завершении алгоритм должен информировать пользователя о том, кто именно находится перед камерой.

Разумной реализацией такой задачи будет разбиение ее на несколько алгоритмов, работающих отдельно и выполняющих каждый свою задачу.

**3.1 Распознавание лица с камеры**

На рисунке изображен алгоритм распознавания лица в режиме реального времени.

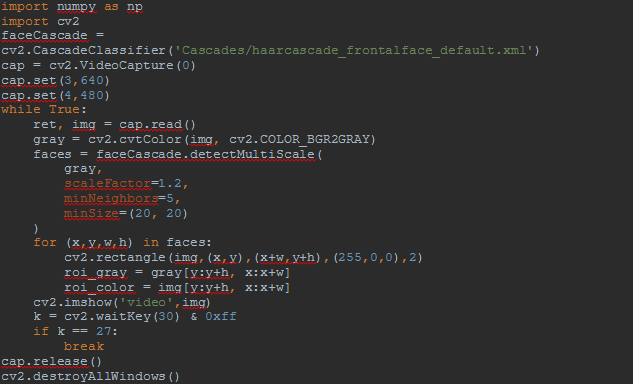


Рисунок 7 – Нахождение лица

После тривиального импорта numpy и cv, требуется импортировать xml файл с положительными и отрицательными примерами лиц. Первоначально алгоритм требует много положительных изображений (изображений лиц) и негативных изображений (изображений без лиц) для обучения классификатора. Затем нужно извлечь из него функции. Описание этого этапа займет много времени, поэтому эта часть будет опущена ниже.

Для неединоразового запуска алгоритма требуется запустить цикл без условия выхода.

Переменная img принимает картинку с разрешением 640X480, с которой будет работать алгоритм. В черно-белом формате программе гораздо проще распознать лицо, так как алгоритм, работающий по признакам Хаара, с большей точностью может найти резкий разрыв оттенков, gray все та же картинка img, но черно-белая.

Создавая функцию классификатора, мы может не только задать параметры для определения лиц, но и упростить задачу для алгоритма.

Scalefactor - параметр, определяющий размер изображения при каждой шкале изображения.

Таким образом мы можем создать масштабируемую пирамиду. В примере указан параметр 1.2, а значит изображение будет уменьшено на 80%, но не один раз. Масштабируемая пирамида представляет из себя набор одного и того же изображения, но разных размеров.

minNeighbors – минимальное количество соседей. В рисунке описано о каких соседях идет речь. Чем параметр выше, тем точнее алгоритм распознает лица.

minSize – это минимальный размер изображения, которое может быть принято за лицо, чем minSize меньше, тем чаще алгоритм будет срабатывать.

maxSize – аналог minSize, но работающий наоборот, в алгоритме не использован, поэтому найденное лицо может быть любых размеров.

Цикл For никак не относится к распознаванию лица, он просто рисует прямоугольник вокруг найденного лица.

Создание xml файла. Работа с расширенным каскадом слабых классификаторов включает в себя два основных этапа: этап обучения и этап обнаружения.

Подготовка данных тренинга заключается в следующем. Для обучения усиленного каскада слабых классификаторов нам нужен набор положительных выборок (содержащих реальные объекты, которые вы хотите обнаружить) и набор отрицательных изображений (содержащих все, что вы не хотите обнаруживать). Набор отрицательных образцов должен быть подготовлен вручную, тогда как набор положительных образцов создается с помощью приложения opencv\_createsamples.

Отрицательные образцы взяты из произвольных изображений, не содержащих объектов, которые вы хотите обнаружить. Эти негативные изображения, из которых генерируются образцы, должны быть перечислены в специальном файле негативных изображений, содержащем один путь изображения на строку (может быть абсолютным или относительным) . Обратите внимание, что негативные образцы и образцы изображений также называются фоновыми образцами или фоновыми изображениями и используются взаимозаменяемо в этом документе.

Описанные изображения могут быть разных размеров. Тем не менее, каждое изображение должно быть равно или больше, чем требуемый размер окна обучения (который соответствует размерам модели, в большинстве случаев это средний размер вашего объекта) , потому что эти изображения используются для выборки данного отрицательного изображения в несколько изображений. образцы, имеющие этот размер окна обучения.

Пример такого файла отрицательного описания:

/ IMG

img1.jpg

img2.jpg

bg.txt

Файл bg.txt:

IMG / img1.jpg

IMG / img2.jpg

Положительные образцы создаются приложением opencv\_createsamples. Они используются процессом повышения, чтобы определить, что именно должна искать модель, пытаясь найти интересующие вас объекты. Приложение поддерживает два способа создания набора данных положительного образца:

1) вы можете генерировать много позитивов из одного изображения позитивного объекта;

2) вы можете предоставить все положительные результаты самостоятельно и использовать инструмент, чтобы вырезать их, изменить их размер и поместить их в необходимый двоичный формат opencv.

Хотя первый подход работает хорошо для фиксированных объектов, таких как очень жесткие логотипы, он имеет тенденцию вскоре потерпеть неудачу для менее жестких объектов. В этом случае мы предлагаем использовать второй подход. Во многих учебных пособиях в Интернете даже утверждается, что 100 изображений реальных объектов могут привести к лучшей модели, чем 1000 искусственно созданных позитивов, с помощью приложения opencv\_createsamples. Однако если вы решите выбрать первый подход, помните о некоторых вещах:

1) стоит обратить внимание, что нужно больше, чем один положительный образец, прежде чем передать его в указанное приложение, потому что он применяет только перспективное преобразование;

2) если нам нужна надежная модель, требуется взять образцы, которые охватывают широкий спектр разновидностей, которые могут встречаться в вашем классе объектов, например, в случае лиц вы должны учитывать различные расы и возрастные группы, эмоции и, возможно, стили бороды. Это также применимо при использовании второго подхода.

Первый подход берет одно изображение объекта с, например, логотипом компании и создает большой набор положительных образцов из данного

изображения объекта путем случайного вращения объекта, изменения интенсивности изображения, а также размещения изображения на произвольных фонах. Количество и диапазон случайности могут контролироваться аргументами командной строки приложения opencv\_createsamples.

Аргументы командной строки:

1. vec <vec\_filename>: имя выходного файла, содержащего положительные образцы для обучения;

2) img <image\_file\_name> : исходное изображение объекта (например, логотип компании);

3) bg <background\_file\_name>: файл описания фона; содержит список изображений, которые используются в качестве фона для случайно искажен-ных версий объекта;

4) num <number\_of\_samples> : количество положительных образцов для генерации;

5) bgcolor <background\_color>: цвет фона (в настоящее время предполагаются изображения в оттенках серого); цвет фона обозначает прозрачный цвет, поскольку могут быть артефакты сжатия, величина допуска на цвет может быть задана параметром –bgthresh, а все пиксели в пределах диапазона bgcolor-bgthresh и bgcolor + bgthresh интерпретируются как прозрачные;

6) bgthresh <background\_color\_threshold>;

7) inv : если указано, цвета будут инвертированы;

8) randinv : если указано, цвета будут инвертированы случайным образом;

10) maxidev <max\_intensity\_deviation> : максимальное отклонение интенсивности пикселей в образцах переднего плана;

11) maxxangle <max\_x\_rotation\_angle> : максимальный угол поворота к оси x, должен быть указан в радианах;

12) maxyangle <max\_y\_rotation\_angle> : максимальный угол поворота к оси Y, должен быть указан в радианах;

13) maxzangle <max\_z\_rotation\_angle> : максимальный угол поворота к оси z, должен быть указан в радианах;

14) show: полезная опция отладки, если указано, каждый образец будет показан, а нажатие Esc продолжит процесс создания семплов, не показывая каждый семпл;

15) w <sample\_width> : ширина (в пикселях) выходных выборок;

16) h <sample\_height> : высота (в пикселях) выходных выборок.

При выполнении opencv\_createsamples для создания образца экземпляра объекта используется следующая процедура: Заданное исходное изображение поворачивается случайным образом вокруг всех трех осей. Выбранный угол ограничен -maxxangle, -maxyangleи -maxzangle. Затем пиксели, имеющие интенсивность от bg\_color-bg\_color\_threshold; Диапазон bg\_color + bg\_color\_threshold интерпретируется как прозрачный. Белый шум добавляется к интенсивности переднего плана. Если –inv ключ указан, то интенсивность пикселей переднего плана инвертируется. Если –randinv указан ключ, алгоритм случайным образом выбирает, следует ли применять инверсию к этому образцу. Наконец, полученное изображение помещается на произвольный фон из файла описания фона с изменением размера до желаемого размера, указанного -w-h сохраняется в vec-файле, указанном параметром –vec командной строки.

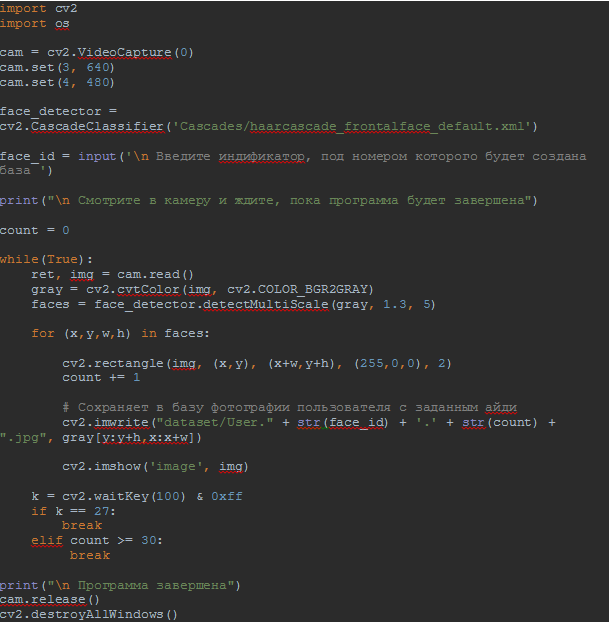
Положительные образцы также могут быть получены из коллекции ранее размеченных изображений, что является желательным способом при построении надежных объектных моделей. Эта коллекция описывается текстовым файлом, похожим на файл описания фона. Каждая строка этого файла соответствует изображению. Первым элементом строки является имя файла, за которым следует число аннотаций объектов, за которыми следуют числа, описывающие координаты объектов, ограничивающих прямоугольники (x, y, ширина, высота).

Только что был рассмотрен алгоритм распознавания лица, но для того, чтобы определить лицо, требуется создать базу лиц.

**3.2 Создание базы лиц**

На картинке 8 изображен алгоритм создания базы лиц.

Можно заметить, что код практически не изменился, но добавилось несколько функций.

Рисунок 8 – Создание базы лиц

Пользователь должен ввести идентификатор, под номером которого будет сохранено 30 фотографий. Именно благодаря им в дальнейшем программа сможет распознавать конкретного человека.

Каждый раз, разпознав лицо, алгоритм сохранит изображения как User.face\_id.count.jpg, где face\_id пользователь вводит самостоятельно, а count – номер сделанного изображения. Как только алгоритм сделает 30 изображений, он завершит работу.

После завершения работы будет создана база изображений, изображенная на рисунке 9.

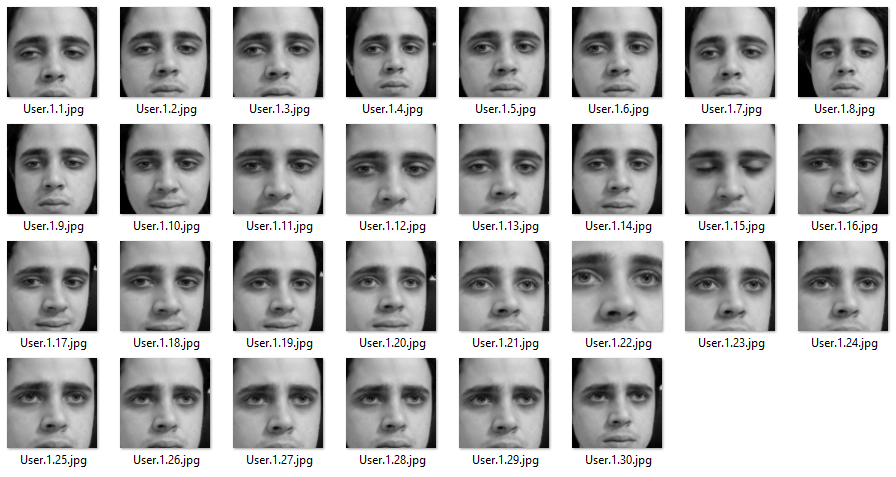


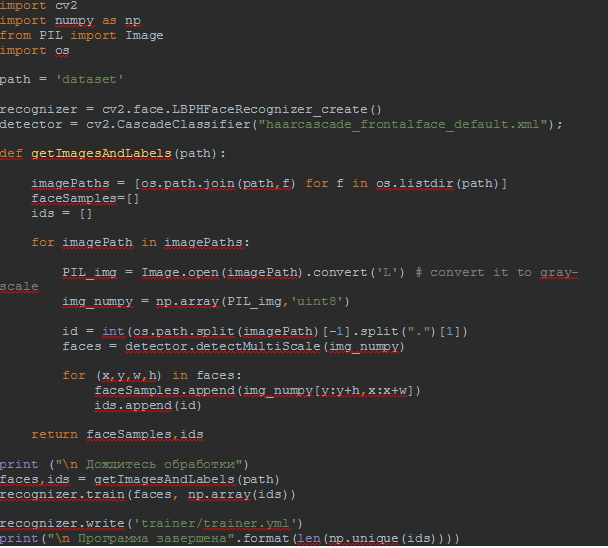
Рисунок 9 – База лиц

**3.3 Обучение распознавателя**

На рисунке 10 изображен алгоритм перекодировки базы лиц в yml файл.

recognizer = cv2.face.LBPHFaceRecognizer\_create()

В качестве распознавателя используется LBPH.

Рисунок 10 – Передача данных в распознаватель

recognizer = cv2.face.LBPHFaceRecognizer\_create()

В качестве распознавателя используется LBPH.

Функция getImagesAndLabels (path) будет принимать все фотографии в каталоге: «dataset /», возвращая 2 массива: «Идентификаторы(lds)» и «Лица(Faces)». С этими массивами в качестве входных данных будет обучен распознаватель.

В результате файл с именем «trainer.yml» будет сохранен в каталоге тренера, который был ранее создан нами.

**3.4 Определение лица**

На рисунке 11 изображен алгоритм распознавания лица конкретного человека, занесенного в базу лиц.

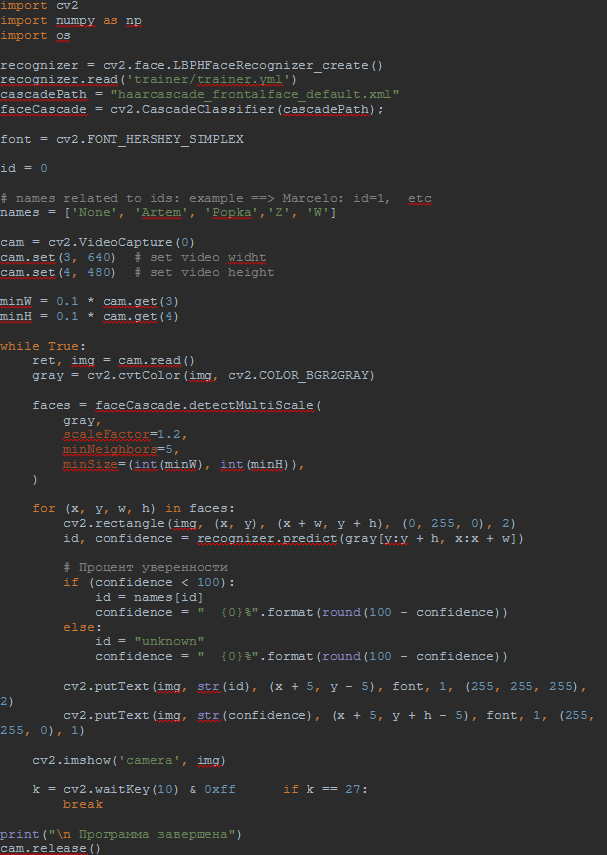


Рисунок 11 – Распознавание лица

Функция recognizer.predict () Принимает в качестве параметра захваченную часть лица, подлежащую анализу, и возвращает своего вероятного владельца, указывая его идентификатор и степень уверенности распознавателя в связи с этим совпадением.

Алгоритм не только старается определить лицо, но и проинформировать пользователя о степени своей уверенности, при том выводится погрешность, то есть 0% - это 100% уверенность алгоритма.

**ЗАКЛЮЧЕНИЕ**

Цель курсовой работы – изучить способы распознавания лица, научиться работать с библиотекой opencv и реализовать алгоритм, способный выполнить поставленную задачу.

В курсовой работе был рассмотрен теоретический материл по теме «Распознавание лиц», были рассмотрены основные способы решения задачи, основанные на теоретическом материале, реализован алгоритм, способный распознавать и определять лица в режиме реального времени. Задача выполнена на языке программирования высокого уровня Python и среды разработки PyCharm.

Планируется обучение распознавателя определять лица не только в анфас, но и в профиль.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Компьютерное зрение. Современный подход / Дэвид А. Форсайт // 2004.
2. Learning OpenCV/ Гари Брадски и Эйдриан Калер // Сентябрь 2008.
3. Русскоязычное сообщество habr [Электронный ресурс]. – URL: [https://habr.com/post/301096/](https://habr.com/post/301096/%20) (дата обращения: 22.11.2018г).