

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

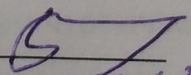
Кафедра технологий
программирования

**КОМПЬЮТЕРНОЕ ЗРЕНИЕ. РЕАЛИЗАЦИЯ СИСТЕМЫ ОБРАБОТКИ
ИЗОБРАЖЕНИЙ И РАСПОЗНАВАНИЯ ОБЪЕКТОВ СРЕДСТВАМИ
OPENCV**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

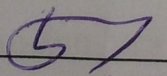
Студентки 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные
технологии
факультета КНиИТ
Некрасовой Яны Антоновны

Научный руководитель
к. ф.-м. н..


10.6.17

И. А. Батраева

Заведующий кафедрой
к. ф.-м. н.


10.6.17

И. А. Батраева

Саратов 2017

ВВЕДЕНИЕ

Компьютерное зрение — теория и технология, позволяющая проводить выявление, наблюдение и классификацию объектов.

Компьютерное зрение, как научная дисциплина, относится к теории и технологии создания искусственных систем, получающих информацию в виде изображений. Видеоданные могут быть представлены в виде многих форм, таких как видеопоследовательность или изображения с разных камер.

Целью компьютерного зрения является формирование выводов на основе анализа изображений, полученных с помощью датчиков.

Для формирования выводов относительно объектов реального мира чаще всего необходимо построить некоторое описание или модель этих объектов на основе изображения. Поэтому можно сделать вывод, что цель компьютерного зрения состоит в формировании описаний сцен по изображениям.

Если рассматривать компьютерное зрение, как технологическую дисциплину, то можно сказать, что основная задача здесь — это создание системы компьютерного зрения с применением теории и моделей. Примерами таких систем могут быть:

- системы управления процессами (промышленные роботы, автономные транспортные средства):
- системы видеонаблюдения:
- системы организации информации (например, для индексации баз данных изображений):
- системы моделирования объектов или окружающей среды (анализ медицинских изображений, топографическое моделирование):
- системы взаимодействия (например, устройства ввода для систем человеко-машинного взаимодействия).

Одной из областей для применения технологий компьютерного зрения является редактирование изображений.

В настоящее время существует огромное количество программного обеспечения, позволяющего производить обработку изображений. Одним из самых популярных редакторов можно назвать программу Adobe Illustrator, которая применяется в издательском деле при подготовке иллюстраций для журналов, книг и газет. Человеку, не очень разбирающемуся в особенностях систем для редактирования фотографий, трудно выбрать нужный редактор.

Графический редактор — программа(или пакет программ), позволяющая создавать и редактировать двумерные изображения с помощью компьютера.

Вычислительная обработка изображений может выполняться различными способами. Можно взять готовую программу, в которой реализованы все необходимые операции для обработки. Многие из таких редакторов находятся в общем доступе, а некоторые являются коммерческими продуктами. Так же существуют программные библиотеки, содержащие большое количество подпрограмм, реализующих модификацию изображений. Пользователь может сам написать свою программу, которая будет вызывать функции библиотеки для выполнения нужных операций. Одной из таких библиотек является OpenCV, которая будет рассмотрена в данной работе и использована для написания собственного приложения для редактирования изображений.

В задачи работы входят:

1. Обзор предметной области.
2. Рассмотрение алгоритмов редактирования изображений, обзор работы с цветовыми пространствами и последовательность действий для распознавания лиц и глаз на основе технологий библиотеки OpenCV.
3. Программная реализация системы подобной графическому редактору с возможностью модификации изображения.

В разработке используется язык программирования Python с возможностями кросс-платформенной графической библиотеки на основе средств Tk (широко распространённая в мире GNU/Linux, портирована также и на Microsoft Windows) в среде PyCharm — интегрированная среда разработки для языка программирования Python.

Дипломная работа состоит из введения, 2 разделов, заключения, списка использованных источников и 4 приложений. Общий объем работы — 74 страницы, из них 49 — основное содержание, включая 40 рисунков, список использованных источников из 21 наименования.

1 Краткое содержание

В разделе 1 «Компьютерное зрение» дается общая информация о задачах решаемых с помощью компьютерного зрения.

Согласно источнику [1], создание разумных машин всегда было главной идеей для авторов фантастической и научно-популярной литературы, а способность визуального восприятия — главным требованием.

Зрение является очень важной функцией человеческого мозга. Люди решают задачи зрительного восприятия очень быстро и без особого понимания. Алан Тьюринг, принимавший участие в создании современного цифрового компьютера и являвшийся основателем области искусственного интеллекта, считал, что с такими задачами может справиться компьютер, если будет обладать разумом.

Возможностей применения компьютеров для обработки изображений достаточно много. В первом подразделе данной части работы перечислены только несколько различных задач, а также описаны примеры:

- Первоначальное рассмотрение структуры цифрового изображения.
- Обработка сканированных страниц текста.
- Машинное зрение.

Цифровые изображения могут представлять различные образы. Цифровое изображение состоит из определенного количества строк и столбцов пикселей (pixels). В [2] этот термин определяется как «элемент изображения» (picture element). Пиксели хранят дискретные значения — небольшие числа, часто от 0 до 255, представляющие яркость точек изображения. В зависимости от схемы кодирования 0 может соответствовать самой малой (темной), а 255 — самой большой (светлой) яркости, или наоборот.

Графическое изображение документа получается после обработки документа сканером (графический образ). Но результат еще не является текстовым документом. Человек может с легкостью решить задачу распознавания текста просто взглянув на лист бумаги. Компьютер же модифицирует графический образ в множество точек. Задача распознавания текста для компьютера является весьма сложной.

Такие задачи решаются с помощью специальных программных средств, называемых средствами распознавания образов.

Существует значительное количество программ для распознавания тек-

ста, так как потребность в этом достаточно велика. Научные методы распознавания текста разивались независимо друг от друга. Именно по этой причине программы используют разную логику построения решений данной задачи.

Машинное зрение — это применение компьютерного зрения в промышленности и производстве. Объектами машинного зрения являются цифровые устройства ввода и вывода и компьютерные сети, осуществляющие контроль производственного оборудования. Машинное зрение явно превосходит возможности зрения человека. Соответственно, развитие этого направления науки считается достаточно актуальным.

Зрительная система, согласно источнику [3], производит сравнение входного изображения с изображением, сгенерированным на основе данных о модели и информации о положении объекта в пространстве. Мало совпадающие детали исключаются из рассмотрения, а более похожие проходят стадию уточнения.

Операции обработки изображений можно разделить на несколько категорий в зависимости от структуры уровня и назначения. Некоторые операции предназначены исключительно для повышения качества изображений при человеческом восприятии, в то время как остальные предназначены для извлечения информации, пригодной для автоматической обработки. Одни операции реализуют создание новых изображений, а другие генерируют в качестве выходных данных результат в неграфической форме. Во втором подразделе первой части рассмотрены операции обработки изображений, такие как:

- Модификация пикселей в небольшой окрестности.
- Глобальное улучшение качества изображения.
- Комбинация нескольких изображений.

В подразделе 3 описаны области смежные с компьютерным зрением. Компьютерное зрение является частью многих областей. Во-первых, важно различать обработку и распознавание изображений. В обработке изображений главная цель — модификация входного изображения, а в распознавании образов — получить вывод на основании подаваемых на вход данных.

Важны для исследований ряд разделов физики, в том числе оптика и теория цвета. Большую роль играет связь с компьютерной графикой. В источнике [4] говорится, что обе сферы имеют дело с возможностью отображения и моделирования объектов; главное отличие между ними в том, что компьютер-

ное зрение связано с описанием и распознаванием объектов по изображениям, а компьютерная графика — с генерацией изображений по описаниям объектов. Между ними существует тесная взаимосвязь по двум направлениям: компьютерная графика нужна для отображения результатов последовательности действий компьютерного зрения, а компьютерное зрение требуется для построения моделей объектов. Согласно источнику [5], цифровые изображения часто используются в качестве входных данных для программ компьютерной графики.

Второй раздел работы посвящен реализации системы обработки изображений и распознавания объектов средствами OpenCV.

Целью данной работы является разработка графического редактора с помощью языка программирования Python, с использованием графических функций и методов библиотеки OpenCV и возможности распознавания лиц и глаз.

Для начала рассмотрена сама библиотека OpenCV: загрузка и установка, необходимые для работы библиотеки NumPy и SciPy, а также встроенные подпрограммы.

Библиотеки NumPy и SciPy, согласно источнику [6], нужны для успешной работы OpenCV.

NumPy — библиотека для языка программирования Python, который (среди прочего) оказывает поддержку для больших, многомерных массивов. Почему это важно? Используя NumPy, мы можем представить изображения как многомерный массив. Представление изображений в виде массивов NumPy не только удобно в вычислительном отношении, но и многие другие операции с изображениями также используют NumPy. Кроме того, при помощи встроенных математических функций NumPy высокого уровня, мы можем быстро выполнить числовой анализ изображения.

Пользуясь NumPy, не нужно забывать о SciPy. SciPy добавляет дальнейшую поддержку научно-технического вычисления.

В подразделе 2.2 данной работы описывается часть используемого функционала библиотеки, ее основные модули и теоретическая основа алгоритмов.

OpenCV — библиотека компьютерного зрения с открытым исходным кодом. Библиотека написана на C и C++ и работает на компьютерах под управлением Linux, Windows, Mac OS X. Так же активно развиваются интерфейсы

библиотеки для Python, Ruby, Matlab и других языков программирования.

Библиотека OpenCV была разработана с целью повышения вычислительной эффективности и с уклоном на приложения, работающие в реальном времени.

Главная цель OpenCV, согласно источнику [7], — предоставление простого в использовании интерфейса, который позволит людям без особых усилий реализовывать сложные приложения для решения задач компьютерного зрения. Библиотека OpenCV содержит большое количество функций, которые охватывают многие области компьютерного зрения, такие как: медицина, безопасность, пользовательский интерфейс, калибровка камеры и робототехника.

В работе исследуются простые методы обработки изображения. Во-первых, рассматриваются основные типы методов модификации изображения, включающие арифметику, битовые операции и маскировку.

Также, рассмотрены варианты из источника [8], как разделить изображение на соответствующие каналы и затем соединить их снова, исследованы различные цветовые пространства, которые поддерживает OpenCV. А так же преимущества и ограничения каждого из них.

Основные модули библиотеки можно отнести к 4 группам (разделам):

- Модули core, highgui, реализующие базовую функциональность (базовые структуры, математические функции, генераторы случайных чисел, линейная алгебра, быстрое преобразование Фурье, ввод/вывод изображений и видео, ввод/вывод в форматах XML, YAML и др.).
- Модули imgproc, features2d для обработки изображений (фильтрация, геометрические преобразования, преобразование цветовых пространств, сегментация, обнаружение особых точек и ребер, контурный анализ и др.).
- Модули video, objdetect, calib3d (калибровка камеры, анализ движения и отслеживание объектов, вычисление положения в пространстве, построение карты глубины, детектирование объектов, оптический поток).
- Модуль ml, реализующий алгоритмы машинного обучения (метод ближайших соседей, наивный байесовский классификатор, деревья решений, случайный лес, машина опорных векторов, нейронные сети и др.).

Согласно источнику [9], все начинается с захвата изображений (модуль highgui). Изображение читается из файла или получается из видеопотока.

После осуществляется предварительная обработка (модуль `imgproc`), такая, как устранение шума, выравнивание яркости, контраста, выделение и удаление бликов, теней. Один и тот же объект может выглядеть по-разному при различном освещении. Например, объект, находящийся под лучами солнца, будет иметь другой оттенок, если поставить его в тень. В таких случаях решением проблемы является выравнивание цветов. Предварительная обработка далеко не самый сложный этап, но может заключать в себе сложную логику модификации изображения.

Следующий этап — выделение особенностей (модули `imgproc`, `features2d`). Например, одна из задач компьютерного зрения, касаемая слежения за объектом, решается на основе данного модуля с помощью поиска особых точек.

После происходит детектирование объектов и образов, выделение интересующих частей изображения (модули `imgproc`, `objdetect`). Данная задача может быть решена на основе ранее описанного алгоритма вычитания изображений.

Далее происходит поиск решения основной задачи: определение местонахождения объекта в пространстве, анализ структуры и т. п. (модули `calib3d`, `contrib`, `video`, `stitching`, `videostab`, `ml`). Например, в задаче видеонаблюдения это восстановление последовательности выполненных передвижений объектом.

Последняя выполняемая задача — принятие решений и обработка выводов (модуль `ml`). Например, в системе видеонаблюдения определяется результат по поставленному «тревожному» правилу — появился объект в кадре или нет.

Реализованный графический редактор позволяет осуществлять работу с как уже имеющимися изображениями, так и с изображениями, сделанными на камеру.

Созданная система в процессе деятельности может быть представлена рядом состояний, которые осуществляют те или иные действия. В системе можно выделить некоторое начальное состояние и конечное, завершающее работу состояние. Можно проанализировать жизненный цикл приложения, в котром возможны следующие действия:

1. нажали кнопку «Good image!»;
2. нажали кнопку «Try again»;
3. выбран Median Filter;

4. выбран Gaussian Filter;
5. выбран Box Filter;
6. выбран Bilateral Filter;
7. выбран Non-Local Means Filter;
8. выбран Sobel Filter;
9. выбран Laplacian Filter;
10. выбран Canny Edge Detector;
11. выбран Blending Filter;
12. выбран Text Filter;
13. выбран Contrast Filter;
14. выбран FaceDetect Filter;
15. выбран Red Eye Filter;
16. выбраны Other Filters;
17. выбор координат;
18. попробовать действие фильтра;
19. применить фильтр;
20. нажать на кнопку «Exit».

Размытие — это то, что происходит, когда камера делает снимок не в фокусе. Более четкие объекты на изображении теряют детали.

Практически, это означает, что каждый пиксель по изображению смешан с его окружающей пиксельной интенсивностью. Эта «смесь» пикселей в районе становится стертым пикселем.

В то время как этот эффект обычно нежелателен на фотографиях, это, на самом деле, довольно полезно при выполнении задачи обработки изображения.

Многие функции компьютерного зрения, такие как обнаружение границ, выполняются лучше, если изображение сначала сглаживается.

Рассмотрены линейные фильтры для редактирования изображений:

- Average Filter;
- Gaussian Filter.

Описана теоретическая основа нелинейных фильтров:

- Median Filter;
- Sigma-Filter.

Рассмотрены цветовые пространства, поддерживаемые OpenCV, теория методов определения границ таких как:

- Sobel Filter;
- Laplacian Filter;
- Canny Edge Detection.

Джон Канни занимался математической задачей относительно поиска фильтра, который будет реагировать на границы и точно определять их, но игнорировать ложные. Он ввел понятие Non-Maximum Suppression (подавление не-максимумов), которое говорит, что пикселями границ будут те точки, в которых достигается локальный максимум градиента в направлении вектора градиента.

Алгоритм работы фильтра можно представить как последовательность следующих шагов:

1. Сглаживание. Размытие изображения для удаления шума.
2. Поиск градиентов. Где градиент изображения достиг максимального значения, там граница.
3. Подавление не-максимумов. Только локальные максимумы отмечаются как границы.
4. Двойная пороговая фильтрация. Предполагаемые границы определяются порогами.
5. Трассировка области неоднозначности. Подавление не явно выраженных краев, несвязанных с определенными (сильными) границами.

Для начала производится преобразование изображения в оттенки серого, чтобы вычисление происходило быстрее.

В OpenCV детектор границ Канни реализуется функцией `cv2.Canny`.

Описан один из самых популярных способов в сфере распознавания объектов, согласно источнику [10], Метод Виолы-Джонса. Для этого метода изображение должны быть представлено в интегральном представлении по признакам Хаара. Классификатор строится на основе адаптивного бустинга, а после классификаторы соединяются в каскадную структуру.

Вообще говоря, все признаки можно назвать определяющими границы, т. е. детекторами. На основе последовательности этих признаков делается вывод о наличии объекта на изображении.

Не менее важным моментом в использовании метода Виола-Джонса является применение каскадной модели или вырожденного дерева принятия решений: в каждом узле дерева с помощью каскада делается вывод о наличии

объекта на изображении. Если объекта нет, то алгоритм завершает свою работу. В противном случае переходит к следующему узлу. Решение построено таким образом, что большинство областей изображения, на которых точно не может быть объекта, отбрасываются на начальном этапе. В случае распознавания лиц — первый уровень содержит всего 2 слабых классификатора.

В разделе 2.3 описывается содержание всего процесса разработки приложения.

В первом подразделе данной части находится описание проектного решения, и в виде рисунка представлен жизненный цикл системы.

В разделе «Программная реализация» описано создание самого приложения, фильтры и возможности системы редактирования, а именно:

1. Размытие изображения разных видов: Median, Gaussian, Bilateral, Non-Local Means.
2. Выделение границ: Sobel, Laplacian, Canny.
3. Наложение «стикеров» на изображение: Blending.
4. Наложение текста.
5. Увеличение контрастности картинки: цветовое пространство LAB.
6. Применение таких фильтров, как черно-белый фильтр, сепия, негатив, насыщенность: цветовые пространства RGB, GRAY, HSV.
7. Размытие лиц.
8. Избавление от эффекта красных глаз.

В данной части приведены снимки экрана, иллюстрирующие процесс взаимодействия пользователя с разработанной программой.

Завершается работа следующим перечнем приложений:

- приложением А — «Листинг программы EasyWebcam.py» — модуль, который позволяет включить камеру своего компьютера и сделать снимок средствами OpenCV;
- приложением Б — «Листинг программы Gui.py» — модуль предназначен для отображения главных функций приложения;
- приложением В — «Листинг программы Filters.py» — модуль с реализацией всех фильтров;
- приложением Г — «CD-диск с отчетом о дипломной работе» — список файлов, содержащихся на приложенном к работе диске.

ЗАКЛЮЧЕНИЕ

Данная дипломная работа потребовала изучения большого количества литературного материала, в результате чего были изучены возможности библиотеки OpenCV, ряд графических редакторов.

На основе исследований было реализовано приложение на языке Python, позволяющее редактировать уже существующие изображения, а также сделанные на камеру. С помощью алгоритмов компьютерного зрения и функций библиотеки OpenCV в приложении используются методы распознавания лиц и глаз для таких фильтров, как размытие лиц на фото и удаление эффекта красных глаз.

Интерфейс созданной программы удобен, прост, наглядно отображает ее возможности. Главное меню редактора содержит команды работы с файлами, сохранения измененного изображения, а также выбор нового.

Всё это свидетельствует о работоспособности приложения и позволяет сделать вывод о его пригодности для создания и редактирования графических изображений.

Таким образом, все поставленные задачи были решены.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 *Форсайт, Д.* Компьютерное зрение. Современный подход / Д. Форсайт, Ж. Понс. — Вильямс, 2004.
- 2 *Шапиро, Л.* Компьютерное зрение / Л. Шапиро, Д. Стокман, А. Богуславский, С. Соколов. — Лаборатория знаний, 2015.
- 3 *Визильтер, Ю.* Обработка и анализ цифровых изображений с примерами на LabVIEW IMAQVision / Ю. Визильтер, С. Желтов, В. Князь, А. Ходарев, А. Моржин. — ДМК Пресс, 2007.
- 4 Зачем компьютеру зрение [Электронный ресурс]. — URL: <http://compress.ru/article/> (Дата обращения 07.05.2017). Загл. с экр. Яз. рус.
- 5 *Карпов, А.* Алгоритмические основы растровой машинной графики: учебное пособие / А. Карпов, Д. Иванов, Е. Кузьмин, В. Лемпицкий, А. Хропов. — Интернет-Университет Информационных Технологий, 2007.
- 6 *Rosebrock, A.* Practical Python and OpenCV: An Introductory, Example Driven Guide to Image Processing and Computer Vision / A. Rosebrock. — PyImageSearch, 2014.
- 7 OpenCV library [Электронный ресурс]. — URL: <http://www.opencv.org> (Дата обращения 21.04.2017). Загл. с экр. Яз. англ.
- 8 *Bradski, G.* Learning OpenCV / G. Bradski, A. Kaehler. — OREILLY, 2008.
- 9 OpenCV шаг за шагом. Введение [Электронный ресурс]. — URL: <http://robocraft.ru/blog/computervision/> (Дата обращения 26.04.2017). Загл. с экр. Яз. рус.
- 10 *Мерков, А.* Распознавание образов. Введение в методы статистического обучения / А. Мерков. — Едиториал УРСС, 2011.

15.06.2017

Handwritten signature