СОДЕРЖАНИЕ

Введение1

1. Теоретическая часть2
   1. Терминология по проекту и глоссарий2
   2. Стек технологий2
2. Практическая часть3
   1. Архитектура ПО3
   2. Разработка проекта3
   3. Контроль выполнения плана3

Заключение4

Список литературы5

ВВЕДЕНИЕ

В современном мире компьютерное зрение и обработка изображений становятся неотъемлемой частью множества областей - от систем безопасности и медицинской диагностики до робототехники и мобильных приложений. Библиотека OpenCV в Python позволяет эффективно разрабатывать и интегрировать алгоритмы распознавания и детекции объектов. Это особенно актуально в условиях стремительного роста данных и необходимости автоматического анализа визуальной информации.

Цель проекта - разработать систему для определения объектов в изображениях с использованием библиотеки OpenCV на языке Python. Для достижения данной цели необходимо решить следующие задачи:

* Изучить теоретические основы компьютерного зрения и современные методы детекции объектов.
* Ознакомиться с возможностями и функционалом библиотеки OpenCV, а также изучить сопутствующие инструменты и библиотеки, используемые в обработке изображений.
* Разработать и реализовать алгоритм, позволяющий эффективно детектировать и классифицировать объекты на изображениях.
* Провести эксперименты для оценки точности, скорости и производительности предложенных методов.
* Проанализировать полученные результаты и предложить рекомендации по оптимизации и дальнейшему развитию системы.

Объектом исследования является система компьютерного зрения, разработанная на базе библиотеки OpenCV с использованием Python, предназначенная для автоматического определения объектов на изображениях.

Предмет исследования включает методы и алгоритмы обработки изображений и детекции объектов, направленные на повышение эффективности и точности распознавания. Особое внимание уделяется анализу процессов предобработки данных, выбору оптимальных алгоритмов и параметров, которые обеспечивают наилучшие результаты при работе с реальными изображениями.

1. Теоретическая часть
   1. Терминология по проекту и глоссарий

OpenCV (open source computer vision library) - библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом. Реализована на С/C++, также разрабатывается для Python, Java, Ruby, Matlab, Lua и других языков. Может свободно использоваться в академических и коммерческих целях - распространяется в условиях лицензии BSD.

Первая версия OpenCV была написана на С, начиная со второй версии был осуществлён переход на C++. Все новые разработки и алгоритмы появляются в API C++. Поддерживается привязки на Python, Java и MATLAB/Octave.

Значительная часть проекта разработана российской компанией Itseez, поглощённой в мае 2016 года корпорацией Intel.

Где используется OpenCV?

OpenCV может использоваться везде, где нужно компьютерное зрение. Эта отрасль IT работает с технологиями, которые позволяют устройству «увидеть», распознать и описать изображение. Компьютерное зрение дает точную информацию о том, что изображено на картинке, с описанием, характеристиками и размерами (с определенной степенью достоверности).

Также библиотека работает с машинным обучением - отраслью, которая обучает алгоритмы действовать тем или иным образом.

OpenCV применяется:

* в робототехнике - для ориентирования робота в пространстве, распознавания объектов и взаимодействия с ними;
* медицинских технологиях - для создания точных методов диагностики, например 3D-визуализации органа при МРТ;
* промышленных технологиях - для автоматизированного контроля качества, считывания этикеток, сортировки продуктов и пр.;
* безопасности - для создания «умных» камер видеонаблюдения, которые реагируют на подозрительные действия, для считывания и распознавания биометрии;
* мобильной фотографии - для создания бьюти-фильтров, изменяющих лицо приложений;
* на транспорте - для разработки автопилотов.

Функции OpenCV:

* Работа со структурами данных

Для хранения и работы с изображениями OpenCV использует векторы и скаляры, матрицы и диапазоны.

* Видоизменение изображений

С помощью OpenCV с картинкой можно работать как в графическом редакторе: обрезать, увеличивать или уменьшать, вращать.

* Добавление эффектов

Картинку можно сделать в оттенках серого или полностью черно-белой.

* Рисование поверх изображения

На картинку можно нанести линии и геометрические фигуры, сделать подпись, например, чтобы выделить найденное программой лицо.

* Распознавание объектов

Для распознавания элементов в OpenCV используются очертания объектов, сегментация по цветам, встроенные методы распознавания, которые можно настраивать в зависимости от объекта и чувствительности алгоритма.

* Работа с видеороликами

Новые версии библиотеки поддерживают работу не только с картинками, но и с видео. Они могут считывать ролики с использованием кодеков, анализировать происходящее в них, отслеживать движения и элементы.

* 1. Стек технологий
* Python

  Язык программирования, выбранный для реализации проекта. Благодаря простоте синтаксиса, большому количеству библиотек и активному сообществу, Python позволяет быстро разрабатывать прототипы и интегрировать различные алгоритмы обработки изображений.

* OpenCV

  Основная библиотека для компьютерного зрения, которая предоставляет инструменты для захвата, обработки и анализа изображений. OpenCV обеспечивает реализацию алгоритмов детекции объектов, распознавания лиц, сегментации и других операций, необходимых для проекта.

* NumPy

  Библиотека для эффективных числовых вычислений и работы с многомерными массивами. NumPy часто используется в комбинации с OpenCV для обработки пиксельных данных, выполнения матричных операций и оптимизации вычислительных процессов.

* Matplotlib / Seaborn

  Инструменты для визуализации данных. Они позволяют строить графики и изображения, что полезно для анализа результатов работы алгоритмов, отладки кода и представления результатов экспериментов.

* SciPy

  Библиотека для научных вычислений, которая предоставляет расширенные математические алгоритмы. SciPy может использоваться для обработки сигналов, оптимизации и решения различных задач, возникающих при разработке систем компьютерного зрения.

* Дополнительные библиотеки машинного обучения (опционально)

  При необходимости можно интегрировать такие библиотеки, как scikit-learn для традиционных алгоритмов классификации, или TensorFlow/PyTorch для построения и обучения нейронных сетей, если задача требует более сложного подхода к детекции объектов.

* Jupyter Notebook или Visual Studio Code

  Среды разработки, способствующие интерактивному программированию и быстрому тестированию алгоритмов. Jupyter Notebook полезен для исследования данных и отладки, а VS Code обеспечивает мощную поддержку разработки больших проектов.

* Версионный контроль (Git)

  Инструмент для управления изменениями в проекте, позволяющий отслеживать историю разработки, работать в команде и эффективно проводить ревью кода.

* Средство управления зависимостями (pip/conda)

  Инструменты для установки и обновления необходимых библиотек и пакетов, что позволяет обеспечить стабильность и воспроизводимость окружения разработки.

Данный стек технологий обеспечивает гибкость, масштабируемость и высокую производительность системы, предназначенной для автоматического определения объектов на изображениях с использованием библиотек и инструментов, проверенных временем в области компьютерного зрения.