МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Российский государственный университет им. А. Н. Косыгина   
(Технологии. Дизайн. Искусство)»

Кафедра информационных технологий

**Курсовая работа**

по дисциплине «Объектно-ориентированное программирование»

на тему: «Разработка приложения для определения номерных знаков автомобилей»

Выполнил:

студент 2 курса Сидоров Д. С.  
группа ИТС-123 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*подпись*

Проверил:   
к.т.н., доцент Семенов А.А.

Оценка: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*подпись*

Дата «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_20\_\_\_ г.

РГУ им. А.Н. Косыгина, 2025 г.

Содержание

[**Введение** 3](#_Toc168256392)

[**Основная часть** 4](#_Toc168256393)

[Постановка задачи 4](#_Toc168256394)

[Обоснование выбора программных средств для решения задачи 4](#_Toc168256395)

[Описание использованных средств языка программирования 4](#_Toc168256396)

[Руководство пользователя 6](#_Toc168256397)

[**Листинг кода** 13](#_Toc168256398)

[main.cpp 13](#_Toc168256399)

[mainwindow.h 13](#_Toc168256400)

[mainwindow.cpp 15](#_Toc168256401)

[codeeditor.h 29](#_Toc168256402)

[codeeditor.cpp 31](#_Toc168256403)

[settingsdialog.h 39](#_Toc168256404)

[settingsdialog.cpp 40](#_Toc168256405)

[editorsettings.h 42](#_Toc168256406)

[editorsettings.cpp 42](#_Toc168256407)

[cppsyntaxhighlighter.h 43](#_Toc168256408)

[cppsyntaxhighlighter.cpp 44](#_Toc168256409)

[**Вывод** 47](#_Toc168256410)

[**Список литературы** 48](#_Toc168256411)

# Введение

В современном мире автоматизация процессов становится неотъемлемой частью нашей жизни. Одной из важных задач является автоматическое распознавание автомобильных номеров, которое находит применение в различных сферах: от управления парковками до систем безопасности и контроля дорожного движения.

Разработка приложения для определения номерных знаков автомобилей представляет собой комплексную задачу, требующую применения современных технологий компьютерного зрения и обработки изображений. Данная работа направлена на создание программного обеспечения, способного эффективно выделять и распознавать автомобильные номера на изображениях.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Разработка архитектуры приложения с использованием принципов ООП
2. Создание модуля захвата и предварительной обработки изображений
3. Реализация алгоритма обнаружения области номерного знака на изображении
4. Разработка системы распознавания символов на номерном знаке
5. Тестирование и отладка работы приложения

Для реализации проекта предлагается использовать следующие технологии и инструменты:

* Язык программирования: Python (версия 3.8 или выше)
* Библиотеки компьютерного зрения: OpenCV
* Библиотека для распознавания текста: Easy OCR
* Среда разработки: PyCharm

Выбор Python обусловлен его простотой, большим количеством доступных библиотек для работы с изображениями и машинного обучения, а также широкими возможностями для создания графического интерфейса. OpenCV предоставляет необходимые инструменты для обработки изображений и выделения областей интереса, а Easy OCR является проверенным решением для распознавания текста.

Результатом выполнения курсовой работы должно стать функционирующее приложение, способное определять и распознавать автомобильные номера на статических изображениях.

# Основная часть

## Постановка задачи

В данной работе необходимо разработать консольную программу для автоматического распознавания автомобильных номеров по изображению и проверки их наличия в списке разрешенных номеров. Программа должна принимать на вход путь к файлу изображения, выполнять его обработку с помощью алгоритмов компьютерного зрения, распознавать текст номерного знака и сравнивать полученный результат со списком разрешенных номеров. На выходе программа должна сообщать, найден ли данный номер в списке разрешенных. Решение должно быть реализовано с использованием принципов объектно-ориентированного программирования и обеспечивать надежное распознавание номеров при различных условиях съемки.

## Обоснование выбора программных средств для решения задачи

Python был выбран в качестве основного языка программирования благодаря его простоте, читаемости кода и богатой экосистеме библиотек для работы с компьютерным зрением и машинным обучением. Python также предоставляет удобные инструменты для работы с файлами и обработки строк, что важно при работе со списками номеров. Язык отлично подходит для быстрого прототипирования и разработки приложений, связанных с обработкой изображений.

EasyOCR представляет собой современную библиотеку для распознавания текста, которая показывает высокую точность при работе с различными языками и шрифтами. Важным преимуществом EasyOCR является возможность использования кастомных каскадных таблиц, что позволяет настроить систему специально под распознавание автомобильных номеров. Это существенно повышает точность распознавания по сравнению с использованием стандартных настроек.

OpenCV (Open Source Computer Vision Library) является ведущей библиотекой для работы с компьютерным зрением. Она предоставляет широкий набор инструментов для обработки изображений: методы предварительной обработки (фильтрация шумов, изменение контраста), выделение областей интереса, работа с контурами. OpenCV оптимизирована для работы в реальном времени и имеет отличную документацию, что упрощает процесс разработки.

Стандартные системные библиотеки Python используются для обеспечения базового функционала программы: работа с файловой системой (os, sys), обработка путей к файлам (pathlib), работа со строками и регулярными выражениями (re). Эти библиотеки являются частью стандартной поставки Python и не требуют дополнительной установки, что упрощает развертывание программы.

Выбранный стек технологий позволяет эффективно решить поставленную задачу, обеспечивая необходимую функциональность и производительность при сохранении простоты разработки и поддержки кода.

## Руководство пользователя

Данный раздел содержит информацию о том, как установить, настроить и использовать программу для определения автомобильных номеров по фотографии.

Системные требования: Для работы программы необходима операционная система Windows, Linux или MacOS с установленным Python версии 3.8 или выше. Также требуется не менее 4 ГБ оперативной памяти и 1 ГБ свободного места на жестком диске.

Установка и настройка: Перед началом работы необходимо установить требуемые зависимости. Откройте терминал или командную строку и выполните следующую команду для установки необходимых библиотек:

pip install opencv-python easyocr pillow

После установки зависимостей необходимо скачать файлы программы и распаковать их в удобное место на компьютере. В корневой директории программы должны находиться следующие основные компоненты: папка src с исходным кодом программы, папка assets для тестовых изображений, файл allowed\_car\_numbers.txt для списка разрешенных номеров и папка progress для отладочной информации.

Настройка списка разрешенных номеров: Для работы с программой необходимо предварительно заполнить файл allowed\_car\_numbers.txt. Каждый разрешенный номер автомобиля должен быть записан в отдельной строке. Формат записи номеров должен соответствовать стандартному формату автомобильных номеров без пробелов и дополнительных символов. При необходимости добавления новых номеров просто добавьте их в конец файла, начиная с новой строки.

Использование программы: Для распознавания номера автомобиля на изображении используется консольный интерфейс. Чтобы обработать изображение, необходимо открыть терминал или командную строку, перейти в директорию с программой и выполнить команду следующего формата:

python3 car\_number\_detect.py путь\_к\_изображению

Например, для обработки тестового изображения из папки assets команда будет выглядеть так: python3 car\_number\_detect.py ./assets/1.jpg

После выполнения команды программа проанализирует указанное изображение и выведет в консоль результат проверки - находится ли распознанный номер в списке разрешенных.

Отладка и диагностика: При работе программы в папке progress автоматически сохраняются промежуточные результаты обработки изображения. Это может быть полезно при возникновении проблем с распознаванием или для понимания процесса работы программы. Здесь можно найти изображения с выделенными областями номеров, результаты предварительной обработки и другую служебную информацию.

Рекомендации по использованию: Для достижения наилучших результатов распознавания рекомендуется использовать четкие изображения с хорошим освещением. Номерной знак на фотографии должен быть читаемым и не иметь сильных загрязнений. Оптимальным является фронтальный ракурс съемки номерного знака. Поддерживаются основные форматы изображений: JPEG, PNG, BMP.

В случае возникновения ошибок при работе программы убедитесь, что указан корректный путь к изображению, файл существует и имеет поддерживаемый формат. Также проверьте правильность установки всех необходимых зависимостей и наличие достаточных прав доступа к файлам программы.

# Листинг кода

Проект состоит из 5 основных классов: mainwindow, codeeditor, settingsdialog, editorsettings и cppsyntaxhighlighter, а также файл main, содержащий основную функцию и точку входа для приложения.

## main.cpp

#include "mainwindow.h"

#include <QApplication>

#include <QStyleFactory>

int main(int argc, char \*argv[])

{

QApplication a(argc, argv);

MainWindow w;

w.show();

return a.exec();

}

## mainwindow.h

*… программный код удален, так как это образец.*

# Вывод

В ходе выполнения курсовой работы на тему "Разработка приложения для редактирования, компиляции и выполнения исходного кода на C++" были достигнуты значимые результаты. Прежде всего, было создан редактор исходного кода, предоставляющее пользователям удобный и интуитивно понятный интерфейс для написания, компиляции и выполнения исходного кода на языке C++. Приложение поддерживает автозаполнение кода, подсветку синтаксиса, автоматические отступы и номера строк, что значительно упрощает процесс написания и отладки программного обеспечения.

Интеграция возможности компиляции и выполнения программ непосредственно из интерфейса приложения ускоряет цикл разработки и тестирования, позволяя пользователям компилировать и запускать программы одним нажатием кнопки. Реализация механизма сигналов и слотов Qt обеспечила создание реактивного и отзывчивого интерфейса, что улучшает пользовательский опыт.

Использование возможностей языка C++, таких как классы, объекты, наследование, полиморфизм и шаблоны, обеспечило модульность и расширяемость кода. Это позволяет легко поддерживать и расширять функциональность приложения в будущем. Механизмы автосохранения и обработки ошибок повышают надежность и устойчивость приложения, предупреждая пользователей о несохраненных изменениях перед закрытием приложения или файла.

Выбор программных средств, таких как Qt и Qt Creator, оказался обоснованным, так как они значительно упростили процесс разработки и предоставили мощные инструменты для создания пользовательских интерфейсов и управления проектами. Разработанное приложение успешно решает поставленные задачи и представляет собой функциональное средство для разработки программ на языке C++. Оно может быть использовано как учебный инструмент для студентов, так и как полезный инструмент для программистов, занимающихся разработкой на C++.

Исходный код проекта:

<https://github.com/folivorra/CuTE.git>

# Список литературы

1. Интернет-ресурс: https://www.learncpp.com/ (дата обращения 20.03.2024)

2. Интернет-ресурс: https://doc.qt.io/ (дата обращения 09.04.2024)

3. Интернет-ресурс: https://metanit.com/cpp/qt/ (дата обращения 15.03.2024)

4. Интернет-ресурс: https://github.com/ (дата обращения 12.05.2024)