

УДК 004.93

## **СИСТЕМА РАСПОЗНАВАНИЯ ОБРАЗОВ: РАЗРАБОТКА СИСТЕМЫ ОБНАРУЖЕНИЯ ТРАНСПОРТНЫХ СРЕДСТВ С ИСПОЛЬЗОВА- НИЕМ НОВЫХ БИБЛИОТЕК**

Гатауллин Б.И.<sup>1</sup>, студент гр. 4504, V курс, Хаерова Э.И.<sup>2</sup>, студент гр. 4165, I курс

Научный руководитель: Тумбинская М.В.<sup>1</sup>, к.т.н., доцент

<sup>1</sup> Казанский национальный исследовательский технический университет им.  
А.Н. Туполева–КАИ

г. Казань

<sup>2</sup> Казанский национальный исследовательский технический университет им.  
А.Н. Туполева–КАИ, г. Казань

В современном мире задача обнаружения объектов на изображениях является ключевой в области компьютерного зрения. Разработка программных решений, способных точно и эффективно обнаруживать объекты, играет важную роль во многих приложениях, таких как автономное вождение, системы безопасности, анализ видеоматериалов и многие другие [1].

Одним из широко используемых алгоритмов для обнаружения объектов является алгоритм YOLOv5 (You Only Look Once, версия 5). YOLOv5 представляет собой глубокую сверточную нейронную сеть, которая способна выполнять обнаружение объектов в реальном времени с высокой точностью. Этот алгоритм отличается высокой вычислительной эффективностью и может быть успешно применен в различных задачах компьютерного зрения.

Данное исследование имеет практическое значение в различных областях, включая автономное вождение, системы видеонаблюдения, анализ транспортных потоков и другие. Результаты работы могут быть использованы в разработке систем автоматического обнаружения транспортных средств, мониторинга дорожного движения и других приложений, требующих точного и быстрого обнаружения объектов на изображениях. Задача в рамках работы заключается в разработке программы для обнаружения транспортных средств на изображениях с использованием алгоритма YOLOv5 [2].

Программа должна использовать предварительно обученную модель YOLOv5 для обнаружения объектов на изображениях. Входом программы является изображение в формате «JPG». Выходом программы будет являться изображение с отмеченными транспортными средствами и информация о количестве обнаруженных объектов.

Для проверки правильности работы программы необходимо протестировать ее на различных изображениях с транспортными средствами. Программа будет

считаться правильно работающей в случае, если она корректно обнаруживает транспортные средства и отмечает их на изображении.

При разработке программы необходимо использовать язык программирования высокого уровня Python и соответствующие библиотеки для работы с нейронными сетями и графическим интерфейсом [3].

Алгоритм решения задачи обнаружения транспортных средств на изображениях с использованием алгоритма YOLOv5 (You Only Look Once, версия 5) включает следующие основные шаги:

1. Загрузка и подготовка модели: на первом этапе загружается предварительно обученная модель YOLOv5, которая способна выполнять обнаружение объектов на изображениях. Модель загружается из репозитория «ultralytics/yolov5» и готовится к использованию для обработки входных изображений.

2. Загрузка изображения: Пользователь загружает изображение в формате «JPG» через графический интерфейс. Изображение открывается и преобразуется в формат, подходящий для обработки моделью YOLOv5.

3. Обработка изображения нейросетевой моделью: Загруженное изображение передается в модель YOLOv5, которая выполняет обнаружение объектов на изображении. Модель возвращает координаты прямоугольных рамок вокруг обнаруженных объектов, их классы и вероятности обнаружения.

4. Фильтрация обнаруженных объектов: на этом этапе выполняется фильтрация обнаруженных объектов по двум критериям: вероятность обнаружения объекта должна быть выше 70%, обнаруженный объект должен принадлежать к классу транспортных средств (например, автомобиль, мотоцикл, автобус, грузовик).

5. Отображение результатов на изображении: для каждого обнаруженного и отфильтрованного объекта на изображении рисуется рамка. Рамка состоит из белого прямоугольника с черной обводкой. Также рядом с рамкой выводится метка с названием класса объекта и вероятностью его обнаружения.

6. Отображение результатов в графическом интерфейсе: Обработанное изображение с отмеченными транспортными средствами отображается в графическом интерфейсе. Дополнительно в интерфейсе выводится информация о количестве обнаруженных транспортных средств.

Алгоритм YOLOv5 является быстрым и эффективным методом для обнаружения объектов на изображениях. Он обладает высокой вычислительной эффективностью и хорошей устойчивостью к изменениям освещенности и геометрическим искажениям [4]. В дальнейшем этот алгоритм может быть использован для различных задач компьютерного зрения, таких как автономное вождение, системы видеонаблюдения и анализ транспортных потоков.

Алгоритм решения задачи строится со следующими этапами:

1. Подготовка и настройка модели. Загрузка предварительно обученной модели YOLOv5 из репозитория ultralytics/yolov5.

2. Загрузка изображения. Создание графического интерфейса для загрузки изображения, выбор изображения в формате «JPG» через диалоговое окно,

загрузка и преобразование изображения в формат, подходящий для обработки моделью YOLOv5.

3. Обработка изображения моделью YOLOv5. Передача загруженного изображения в модель YOLOv5, получение координат прямоугольных рамок вокруг обнаруженных объектов, их классов и вероятностей обнаружения.

4. Фильтрация обнаруженных объектов: отбор обнаруженных объектов с вероятностью более 80%, фильтрация объектов, относящихся к классам транспортных средств (автомобиль, мотоцикл, автобус, грузовик).

5. Отображение результатов на изображении, создание белой рамки с черной обводкой для каждого обнаруженного транспортного средства, отрисовка рамки вокруг обнаруженного объекта, добавление метки с названием класса объекта и вероятностью его обнаружения рядом с рамкой.

6. Отображение результатов в графическом интерфейсе: преобразование обработанного изображения с отмеченными транспортными средствами для отображения в графическом интерфейсе: отображение обработанного изображения в окне приложения, вывод информации о количестве обнаруженных транспортных средств в графическом интерфейсе [7-8].

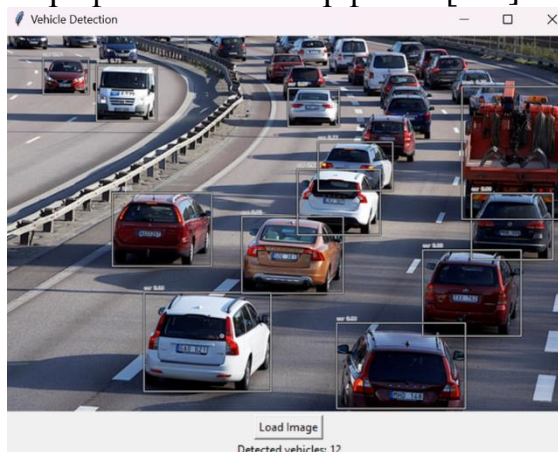


Рис. 1. Результат для изображения №1

Программная реализация алгоритма выполнена на языке Python в среде программирования VS Code с использованием библиотек numpy, opencv, pillow, tkinter и torch. Реализация и алгоритм работы программы:

1. Подготовка и настройка модели. В этом блоке импортируются необходимые библиотеки для работы с нейронными сетями и изображениями. Загрузка предварительно обученной модели YOLOv5 из репозитория ultralytics/yolov5.

2. Обработка изображения моделью YOLOv5. В этом блоке создается функция для передачи загруженного изображения в модель YOLOv5 и получения результатов обнаружения объектов.

3. Загрузка изображения. В этом блоке создается функция для загрузки изображения, его преобразования и передачи в модель YOLOv5 для обработки.

4. Фильтрация обнаруженных объектов. В этом блоке выполняется фильтрация обнаруженных объектов по вероятности и классу транспортных средств.

5. Отображение результатов на изображении. В этом блоке создаются белая рамка с черной обводкой и метки для обнаруженных объектов, затем результаты отображаются на изображении.

6. Отображение результатов в графическом интерфейсе. В этом блоке преобразуется обработанное изображение для отображения в графическом интерфейсе, а также выводится информация о количестве обнаруженных транспортных средств.

7. Создание графического интерфейса. В этом блоке создается графический интерфейс для загрузки изображения, отображения результатов и вывода информации о количестве обнаруженных транспортных средств [6].

В ходе выполнения работы была разработана программа, предназначенная для обнаружения транспортных средств на изображениях с использованием алгоритма YOLOv5 [6]. В ходе экспериментов с различными изображениями было показано, что разработанная программа способна с точностью выше 98% распознавать транспортные средства, такие как автомобили, мотоциклы, автобусы и грузовики. Программа фильтрует обнаруженные объекты по классу и вероятности, отображая только те объекты, вероятность которых выше 70%. Результаты работы программы отображаются в графическом интерфейсе, где пользователь может видеть изображение с отмеченными транспортными средствами и количеством обнаруженных объектов. Таким образом, разработанная программа представляет собой эффективное средство для обнаружения транспортных средств на изображениях с использованием алгоритма YOLOv5 и может быть полезной для различных приложений, таких как системы видеонаблюдения, анализ транспортных потоков и автономное вождение.

#### **Список литературы:**

1. OpenCV-Python Tutorials [Электронный ресурс] - [https://docs.opencv.org/4.x/d6/d00/tutorial\\_py\\_root.html](https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html) (дата обращения: 13.01.2025)
2. Детекция объектов с помощью YOLOv5 [Электронный ресурс] - <https://habr.com/ru/articles/576738/> (дата обращения: 01.02.2025)
3. Официальный репозиторий YOLOv5 на GitHub [Электронный ресурс] - <https://github.com/ultralytics/yolov5> (дата обращения: 22.01.2025)
4. Хаерова Э.И. Обнаружение поддельных новостей с использованием нейронных сетей LSTM // Бюллетень I Международной молодежной конференции по информационной безопасности, 16 мая 2023 г.: Сборник тезисов/ отв. ред. А.Ж. Мартиросян, Р.Н. Шангараев Дипломатическая академия МИД России. – М., 2023. — С. 88-95. ISBN ISBN-978-5-6048376-1-0
5. Гатауллин Б.И. Исследование перспективных методов сокрытия информации в цифровых и аналоговых сигналах // МНПК-2024: материалы Всероссийской молодёжной научно-практической конференции. Казань, 27–29 мая 2024 г. Сборник докладов. Казань: Изд-во ИП Сагиев А.Р., 2024. — С. 237-239. ISBN ISBN 978-5-6053147-0-7

6. Мухаматханов Р.М., Михайлов А.А., Баянов Б.И., Тумбинская М.В. Классификация DDOS-атак на основе нейросетевой модели // Прикладная информатика. 2019. Т. 14. № 1 (79). С. 96-103.
7. Тумбинская М.В., Галиев Р.А. Идентификация фейк-новостей с помощью веб-ресурса на основе нейронных сетей // Программные продукты и системы. 2023. № 4. С. 590-599.
8. Sharipov R., Tumbinskaya M., Abzalov A. Analysis of users' keyboard handwriting based on gaussian reference signals. В сборнике: Proceedings - 2019 International Russian Automation Conference, RusAutoCon 2019.