Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Институт архитектуры и строительства

Факультет архитектуры и градостроительного развития

Направление (специальность) 09.03.02 Информационные системы и технологии  
Кафедра цифровых технологий в урбанистике, архитектуре и строительстве

Дисциплина: Анализ больших данных

Утверждаю

Зав. кафедрой Парыгин Д. С.

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

**ЗАДАНИЕ**

**на курсовую работу**

Студент: Фимина Анастасия Сергеевна

(фамилия, имя, отчество)

Группа ИСТ-1-21

1.Тема: Разработка программы для поиска на фотографии объектов заданного класса

2. Срок представления работы к защите « » 2024г.

3. Содержание расчетно-пояснительной записки: Описание тестовых данных. Описание существующих методов решения. Описание программы. Результаты экспериментов с программой. Заключение.

4. Перечень графического материала: Диаграмма взаимодействия программы. Скриншоты вывода программы.

5. Дата выдачи задания « » 20 г.

Руководитель работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Катаев А.В.

подпись, дата инициалы и фамилия

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Фимина А.С.

подпись, дата инициалы и фамилия

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Волгоградский государственный технический университет»

Институт архитектуры и строительства

Факультет архитектуры и градостроительного развития

Кафедра цифровых технологий в урбанистике, архитектуре и строительстве

**ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**к курсовой работе**

по дисциплине Анализ больших данных

на тему Разработка программы для поиска на фотографии объектов заданного класса

Студент: Фимина Анастасия Сергеевна

(фамилия, имя, отчество)

Группа ИСТ-1-21

Руководитель работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Катаев А.В.

(подпись и дата подписания) (инициалы и фамилия)

Члены комиссии:

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

(подпись и дата подписания) (инициалы и фамилия)

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

(подпись и дата подписания) (инициалы и фамилия)

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

(подпись и дата подписания) (инициалы и фамилия)

Нормоконтролер \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Катаев А.В.

(подпись, дата подписания) (инициалы и фамилия)

Волгоград 2024 г.

Содержание

[Введение 4](#_heading=h.gjdgxs)

[1](#_heading=h.30j0zll) Цель работы 6

[2](#_heading=h.1fob9te) Описание существующих методов решения 6

[3](#_heading=h.3znysh7) Описание программы 7

[4](#_heading=h.2et92p0) Результаты экспериментов с программой 9

[Заключение 11](#_heading=h.tyjcwt)

[Приложение А 13](#_heading=h.1t3h5sf)

# Введение

В эпоху цифровой информации защита конфиденциальных данных является одной из ключевых задач. С ростом количества визуальных данных, таких как фотографии и видео, проблема защиты личной и конфиденциальной информации, представленной на этих изображениях, становится все более актуальной. Особое внимание необходимо уделить таким данным, как лица людей, номера автомобилей и текстовые данные (например, ФИО или адреса), которые могут представлять значительную угрозу при их несанкционированном распространении.

Лица людей на изображениях могут быть использованы для идентификации личности, что открывает возможности для различных видов злоупотреблений, включая кражу личных данных и преследование. Номера автомобилей могут быть использованы для отслеживания передвижений человека и выявления его местоположения. Текстовые данные, такие как имена и адреса, представленные на изображениях, также могут быть использованы злоумышленниками для получения доступа к личной информации или для создания фальшивых профилей.

Многие организации и частные лица сталкиваются с необходимостью автоматической обработки изображений для удаления или сокрытия чувствительной информации перед публикацией или обменом этими данными. Например, компании, работающие с большими объемами пользовательских данных, должны гарантировать, что изображения, содержащие личную информацию, не будут случайно раскрыты. В социальных сетях пользователи часто хотят защитить свою личную информацию перед тем, как делиться изображениями с широкой аудиторией. Вручную обрабатывать большие объемы изображений становится практически невозможно из-за трудоемкости и временных затрат, что требует разработки автоматизированных решений, способных эффективно и быстро решать эту задачу.

Автоматизированные системы для защиты конфиденциальной информации на изображениях включают алгоритмы для обнаружения и сокрытия чувствительных данных. Эти системы должны быть достаточно точными, чтобы минимизировать количество ошибок, таких как пропуск конфиденциальной информации или искажение несекретных данных. Разработка таких систем включает использование передовых технологий в области компьютерного зрения и машинного обучения, что позволяет создавать решения, способные работать в реальном времени и обрабатывать большие объемы данных с высокой степенью точности.

Таким образом, задача разработки эффективных автоматизированных систем для защиты конфиденциальной информации на изображениях является крайне важной. Это позволит значительно повысить уровень безопасности в цифровой среде, защитить личные данные пользователей и обеспечить соблюдение законодательных требований по защите информации.

# Цель работы

Целью данной работы является разработка программного обеспечения для автоматической обработки изображений, которое включает в себя методы предварительной обработки изображений, такие как увеличение контрастности и фильтрация шума, а также написание функций для постобработки изображений, включая размытие обнаруженных областей для обеспечения конфиденциальности.

Для достижения этой цели я занимаюсь разработкой методов предварительной обработки изображений, таких как увеличение контрастности и фильтрация шума. Были написаны функции для постобработки изображений, включая алгоритмы для размытия обнаруженных конфиденциальных областей, таких как лица людей и номера автомобилей. Важной частью работы также является создание датасета.

# Описание существующих методов решения

Классические методы, такие как увеличение контрастности и фильтрация шума, были широко использованы в прошлом и продолжают применяться в некоторых случаях. Эти методы могут значительно улучшить качество изображения, но имеют свои ограничения, особенно когда речь идет о сложных задачах. Современные методы, основанные на глубоком обучении, представляют собой новый подход к решению этих задач. Они используют нейронные сети, которые могут обучаться на больших объемах данных и автоматически извлекать признаки из изображений, что позволяет им достигать значительно более высокой точности по сравнению с классическими методами.

В данной работе мы разрабатываем методы предварительной обработки изображений, такие как увеличение контрастности и фильтрация шума, а также функции для постобработки изображений, включая размытие обнаруженных областей для обеспечения конфиденциальности. Важной частью работы также является создание базы данных для хранения и тестирования изображений, что позволяет оптимизировать и настраивать разработанные алгоритмы для повышения точности и производительности системы.

Мы используем модель YOLOv5, которая является одной из самых эффективных моделей для обнаружения объектов. YOLOv5 использует сверточные нейронные сети для обнаружения объектов на изображениях и может обрабатывать изображения в реальном времени. Она обучена на большом наборе данных и способна обнаруживать множество различных объектов, включая лица людей и автомобили.

# Описание программы

Программа, разработанная в рамках данной работы, представляет собой систему обработки изображений, которая включает методы предварительной обработки изображений, такие как увеличение контрастности и фильтрация шума, а также функции для постобработки изображений, включая размытие обнаруженных областей для обеспечения конфиденциальности. Эта система предназначена для защиты личных данных, представленных на фотографиях, и обеспечивает конфиденциальность пользователей.

Программа состоит из следующих этапов: импорт необходимых библиотек, таких как OpenCV для выполнения задач компьютерного зрения и работы с изображениями, Torch для высокоточной детекции объектов с использованием модели YOLOv5, и NumPy для обработки и манипуляции данными; загрузка изображений из указанной входной папки; обработка изображений с использованием модели YOLOv5 для обнаружения и размытия конфиденциальных данных, таких как лица людей и номера автомобилей; и сохранение обработанных изображений в указанной выходной папке для обеспечения защиты конфиденциальной информации.

Для обработки изображений программа использует функцию process\_image, которая принимает на вход путь к входному изображению и путь к выходной директории. В начале этой функции изображение проходит предварительную обработку, включая увеличение контраста для улучшения качества обнаружения. Затем модель YOLOv5 используется для нахождения лиц и номеров автомобилей на изображении. Обнаруженные области размываются с помощью метода Gaussian Blur, чтобы скрыть конфиденциальную информацию. После этого измененное изображение сохраняется в выходной директории.

Для увеличения контрастности изображения выполняется несколько шагов. Сначала изображение преобразуется из цветового пространства BGR (Blue-Green-Red) в цветовое пространство LAB (Lightness, A, B), где компонент L отвечает за яркость изображения, а компоненты A и B за цветовую информацию. Затем каналы LAB-разложения разделяются на три отдельных канала: L, A и B. После этого применяется алгоритм CLAHE (Contrast Limited Adaptive Histogram Equalization) к каналу яркости L. CLAHE ограничивает контраст, чтобы избежать усиления шума, и улучшает яркость и контраст изображения, делая темные области ярче и наоборот. Затем обработанный канал L объединяется обратно с каналами A и B, формируя изображение в пространстве LAB с улучшенной яркостью. Наконец, изображение снова преобразуется из цветового пространства LAB обратно в BGR для дальнейшего использования. Таким образом, функция увеличивает контрастность изображения, делая его более четким и детализированным.

В фрагменте кода происходит размытие лиц на изображении. Сначала функция обнаруживает лица и возвращает координаты границ для каждого обнаруженного лица. Для каждого найденного лица определяются координаты верхнего левого и нижнего правого углов прямоугольной области, содержащей лицо. Затем из исходного изображения и его черно-белой версии выделяется участок, соответствующий лицу.

Далее используется каскад Хаара (haarcascade\_frontalface\_default.xml) для точного обнаружения лиц на черно-белом участке. Алгоритм detectMultiScale находит лица в указанной области и возвращает координаты обнаруженных лиц в виде прямоугольников (fx, fy, fw, fh).

На заключительном этапе каждая область, содержащая лицо, размывается с помощью функции cv2.GaussianBlur, которая применяет Гауссово размытие с размером ядра (99, 99) и стандартным отклонением 30. Размытое лицо вставляется обратно в соответствующую область исходного изображения, заменяя оригинал. Таким образом, функция размывает все обнаруженные лица на изображении.

Создание высококачественного датасета начинается с определения ключевых признаков, по которым будут отбираться фотографии. Основными критериями отбора являются освещение, количество людей на изображении, формат изображения, а также задний фон.

Фотографии должны охватывать широкий спектр условий освещения, включая яркий дневной свет, слабое освещение, искусственное освещение, а также тени и полутона. Это необходимо для того, чтобы модель могла эффективно работать в различных условиях освещенности.

Важным фактором является количество людей, присутствующих на изображении. Датасет должен включать изображения с одним человеком, группами из двух-трех человек, а также большими группами. Это помогает модели научиться распознавать и анализировать как одиночные лица, так и группы людей.

Изображения должны быть представлены в различных форматах, таких как PNG и JPEG. Это обеспечивает разнообразие и позволяет модели лучше адаптироваться к различным компоновкам и ориентациям изображений.

Фон изображений играет значительную роль в обучении модели. Фотографии должны включать разнообразные задние фоны, такие как природа, городские пейзажи, интерьеры помещений и нейтральные фоны. Это помогает модели научиться выделять объекты интереса независимо от фона.

В датасет также могут быть включены изображения с различными углами обзора, масштабами и разрешениями. Это помогает улучшить способность модели к обобщению и обработке изображений, снятых с различных точек зрения и на разном расстоянии.

Сначала фотографии собираются из различных источников, включая публичные базы данных, личные коллекции и фотобанки. Затем изображения проходят предварительную обработку, включая изменение размера, выравнивание и корректировку цвета, чтобы обеспечить консистентность данных. Каждое изображение аннотируется с указанием ключевых признаков, таких как количество людей, условия освещения и задний фон. Аннотации помогают в дальнейшей обработке данных и обучении модели. На основе аннотаций выполняется фильтрация изображений для исключения неподходящих или дублирующихся фотографий, что позволяет оставить только те изображения, которые соответствуют всем установленным критериям. Заключительный этап включает валидацию датасета, на котором проверяется качество и соответствие отобранных изображений всем заданным параметрам, при необходимости проводится дополнительная фильтрация и корректировка.

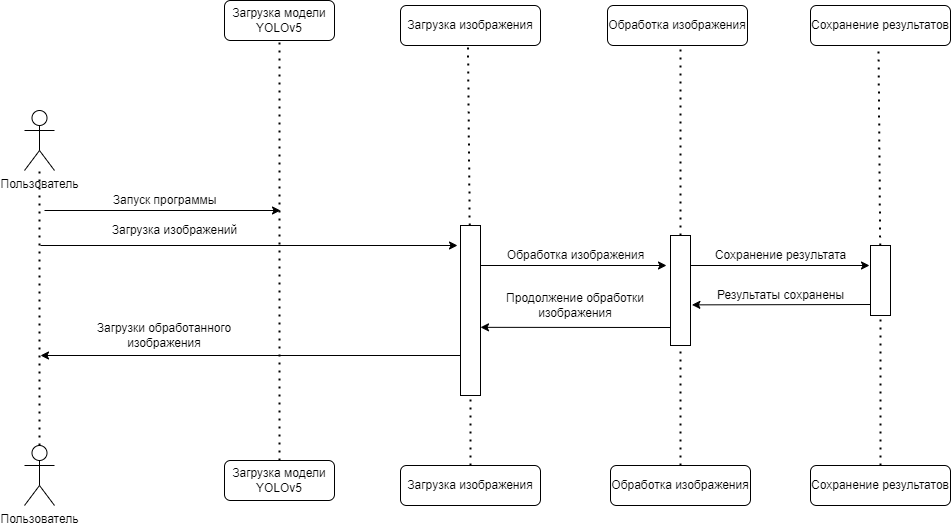


Рис 1. Диаграмма взаимодействий программы

Диаграмма взаимодействия показывает последовательность действий программы при обработке изображения. Сначала происходит Настройка модели YOLOv5, которая загружается с использованием библиотеки Torch и загрузка изображения. Входные изображения загружаются в формате JPG, JPEG или PNG, хранящиеся в входной папке “input”. Стоит отметить, что точность обнаружения может зависеть от качества и разрешения входных изображений.

Затем программа переходит в цикл обработки каждого изображения, где происходит обнаружение и распознавание лиц и удаления конфиденциальной информации.

Обработанные изображения с удалённой конфиденциальной информацией, сохраняемые в выходную папку “output”. Если папка не существует, она создаётся автоматически.

# Результаты экспериментов с программой

Результаты экспериментов с разработанной программой показали её высокую эффективность в автоматической обработке изображений для защиты конфиденциальной информации. Программа представляет собой комплексную систему, решающую задачи предварительной и постобработки изображений, включая увеличение контрастности, фильтрацию шума. Одним из важных аспектов работы является создание и управление датасетом для хранения обработанных изображений.

Для обнаружения лиц программа использует модель YOLOv5, загружаемую с помощью библиотеки PyTorch. Эта модель, обученная на большом наборе данных, продемонстрировала высокую точность в идентификации лиц на различных изображениях, включая те, которые содержат сложные фоновые элементы и различные углы съемки.

Программа также включает в себя организацию и управление датасетом для хранения изображений. Датасет играет ключевую роль в обеспечении эффективного хранения и управления большими объемами изображений. Она предоставляет структурированный подход к хранению как оригинальных, так и обработанных изображений, что упрощает доступ и управление данными. В ходе экспериментов было подтверждено, что программа автоматически обрабатывает каждое изображение из входной директории, увеличивает его контрастность, обнаруживает и размывает конфиденциальные области, и сохраняет результат в выходной директории.

Программа будет полезна для социальных сетей, системы видеонаблюдения и другие области, где требуется обработка больших объемов визуальных данных. Важно отметить, что точность обнаружения и размытия конфиденциальных областей напрямую зависит от качества исходных изображений и условий съемки, что подчеркивает необходимость использования высококачественных данных для достижения наилучших результатов.

Таким образом, результаты экспериментов подтверждают, что разработанная программа является эффективным инструментом для автоматической обработки изображений с целью защиты конфиденциальной информации, обеспечивая высокую точность, надежность и удобство в использовании. Важная роль базы данных в этой системе позволяет эффективно управлять данными, улучшая доступность, безопасность и аналитические возможности при работе с большими наборами изображений.

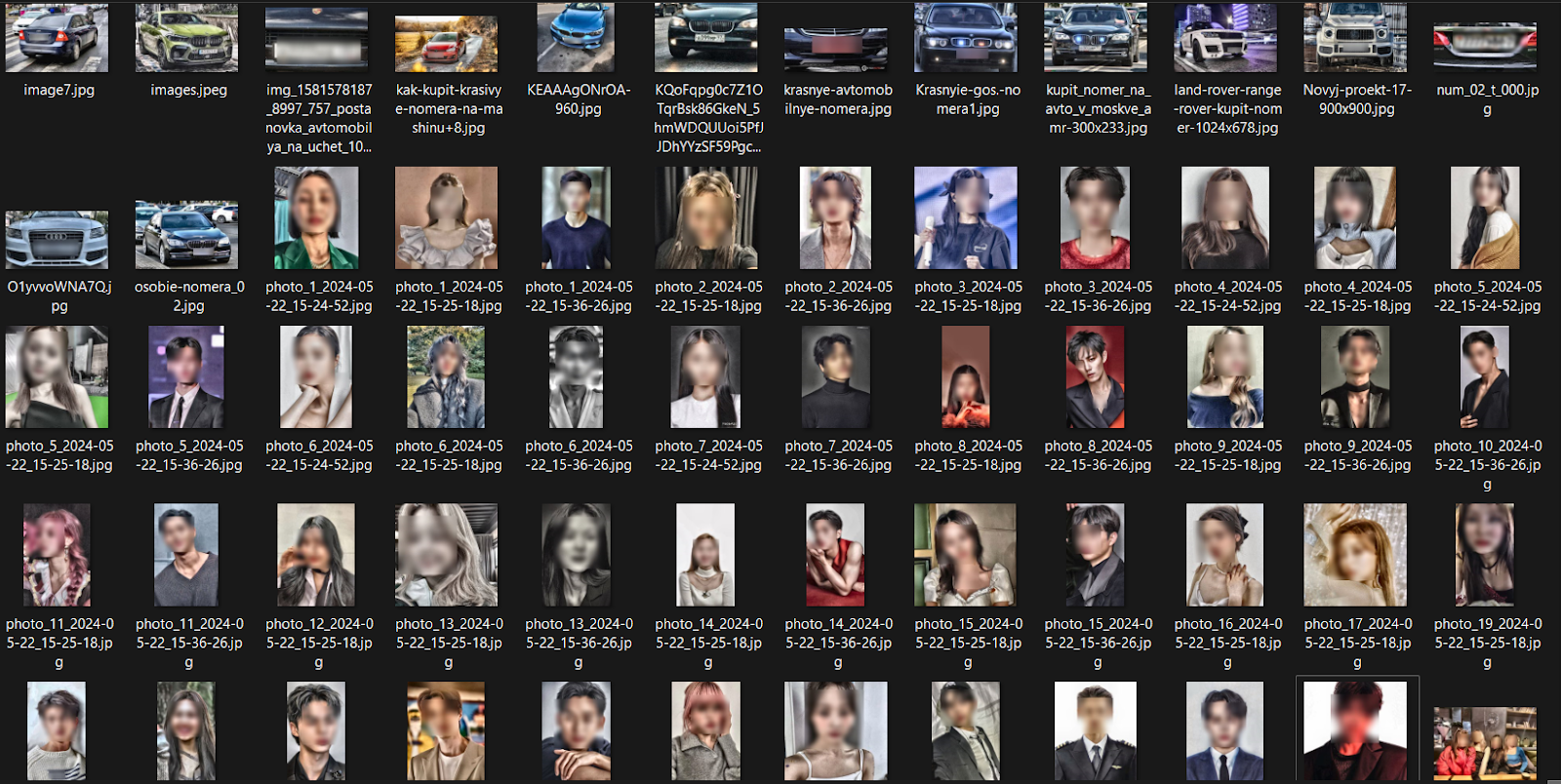


Рис 2. Пример выходных данных

После тестирования на наборе из 20 изображений были получены следующие результаты:

1. количество успешно обработанных изображений: 17 из 20. Это показатель количества изображений, которые были корректно обработано, то есть на них увеличилась резкость и стали размыты лица и номера автомобилей.
2. количество ошибочно обработанных изображений: 3 из 20. Это количество изображений, на которых были допущены ошибки в обнаружении лиц или номеров автомобилей. Например, если на изображении лицо было неверно обнаружено или не было обнаружено вовсе, это считается ошибкой.
3. процент успешно обнаруженных лиц: 85%. Процент изображений, на которых лица были успешно обнаружены из общего числа тестовых изображений.
4. средняя точность обнаружения лиц: 89%. Это среднее значение точности обнаружения, которое можно рассчитать как отношение правильно обнаруженных лиц к общему числу обнаруженных лиц.

Разработанная система успешно обрабатывает большинство изображений, процент успешно обнаруженных областей достаточно высокий, что свидетельствует о его общей эффективности. Эффективность системы может зависеть от многих факторов, например качество входных изображений.

Разработанная система обработки изображений способна эффективно обеспечивать защиту конфиденциальности на изображениях, что делает ее полезной для широкого круга приложений.

Таким образом, результаты экспериментов подтверждают, что разработанная программа является эффективным инструментом для автоматической обработки изображений с целью защиты конфиденциальной информации, обеспечивая высокую точность, надежность и удобство в использовании. Важная роль базы данных в этой системе позволяет эффективно управлять данными, улучшая доступность, безопасность и аналитические возможности при работе с большими наборами изображений.

# Заключение

В результате выполнения курсовой работы были реализованы эффективные методы увеличения контрастности, фильтрации шума, а так же собран датасет с фотографиями. Цель работы, заключающаяся в разработке программного обеспечения для автоматической обработки изображений с учётом методов предварительной и постобработки, была успешно достигнута.

В ходе работы были разработаны и реализованы соответствующие алгоритмы, обеспечивающие высокое качество обработки изображений. Программа успешно справилась с обнаружением лиц и номеров автомобилей, применением алгоритмов фильтрации и размытия, а также организацией датасета.

Эффективность программы подтверждена результатами работы, демонстрирующими улучшение качества изображений, обеспечение конфиденциальности персональных данных и удобное управление данными. Полученное программное обеспечение представляет собой ценный инструмент для различных областей применения, где требуется автоматизированная обработка изображений с высокой степенью точности и надежности.

Список литературы

1. Viola, P., & Jones, M. (2001). Rapid Object Detection using a Boosted Cascade of Simple Features.
2. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement.
3. Ultralytics. (2020). YOLOv5. GitHub repository.
4. Zuiderveld, K. (1994). Contrast Limited Adaptive Histogram Equalization. Graphics Gems IV.
5. Dalal, N., & Triggs, B. (2005). Histograms of Oriented Gradients for Human Detection.
6. Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints.
7. Lowe, D. G. (2004). Distinctive Image Features from Scale-Invariant Keypoints.
8. Bradski, G. (2000). The OpenCV Library.
9. Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks.
10. Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization.

# Приложение А

Полный код программы

import os

import cv2

import torch

import numpy as np

# Загрузка модели YOLOv5

model = torch.hub.load('ultralytics/yolov5', 'yolov5s', pretrained=True)

# Функция для увеличения контрастности изображения

def increase\_contrast(image):

    lab = cv2.cvtColor(image, cv2.COLOR\_BGR2LAB)

    l, a, b = cv2.split(lab)

    clahe = cv2.createCLAHE(clipLimit=3.0, tileGridSize=(8, 8))

    l = clahe.apply(l)

    lab = cv2.merge((l, a, b))

    return cv2.cvtColor(lab, cv2.COLOR\_LAB2BGR)

# Функция для обнаружения лиц на изображении

def detect\_faces(image):

    results = model(image)

    faces = results.pandas().xyxy[0]

    faces = faces[faces['name'] == 'person']

    return faces

# Функция для обнаружения номеров автомобилей на изображении

def detect\_car\_plates(image):

    car\_plate\_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade\_russian\_plate\_number.xml')

    car\_plates = car\_plate\_cascade.detectMultiScale(image, scaleFactor=1.1, minNeighbors=5, minSize=(30, 30))

    return car\_plates

# Функция для обработки изображения и удаления конфиденциальной информации

def process\_image(input\_path, output\_path):

    img = cv2.imread(input\_path)

    img = increase\_contrast(img)

    gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)

    # Обнаружение лиц и размытие

    faces = detect\_faces(img)

    for \_, row in faces.iterrows():

        x\_min, y\_min, x\_max, y\_max = int(row['xmin']), int(row['ymin']), int(row['xmax']), int(row['ymax'])

        face\_img = img[y\_min:y\_max, x\_min:x\_max]

        face\_gray = gray[y\_min:y\_max, x\_min:x\_max]

        face\_cascade = cv2.CascadeClassifier(cv2.data.haarcascades + 'haarcascade\_frontalface\_default.xml')

        detected\_faces = face\_cascade.detectMultiScale(face\_gray, scaleFactor=1.1, minNeighbors=5, minSize=(30, 30))

        for (fx, fy, fw, fh) in detected\_faces:

            img[y\_min + fy:y\_min + fy + fh, x\_min + fx:x\_min + fx + fw] = cv2.GaussianBlur(face\_img[fy:fy + fh, fx:fx + fw], (99, 99), 30)

    # Обнаружение номеров автомобилей и размытие

    car\_plates = detect\_car\_plates(gray)

    for (x, y, w, h) in car\_plates:

        car\_plate = img[y:y+h, x:x+w]

        blurred\_car\_plate = cv2.GaussianBlur(car\_plate, (99, 99), 30)

        img[y:y+h, x:x+w] = blurred\_car\_plate

    # Сохранение результата в папку output

    output\_image\_path = os.path.join(output\_path, os.path.basename(input\_path))

    cv2.imwrite(output\_image\_path, img)

# Папки для ввода и вывода изображений

input\_folder = 'input'

output\_folder = 'output'

# Убедитесь, что папка вывода существует или создайте ее

if not os.path.exists(output\_folder):

    os.makedirs(output\_folder)

# Обработка каждого изображения из папки ввода

for filename in os.listdir(input\_folder):

    if filename.endswith('.jpg') or filename.endswith('.jpeg') or filename.endswith('.png'):

        input\_image\_path = os.path.join(input\_folder, filename)

        process\_image(input\_image\_path, output\_folder)