

Министерство образования Республики Беларусь
Учреждение образования
«Брестский Государственный технический университет»
Кафедра ИИТ

Лабораторная работа №4
По дисциплине «ОИИС»
Тема: «Трекинг множественных объектов»

Выполнил:
Студент 4 курса
Группы ИИ-24
Крейдич А. А.
Проверила:
Андренко К. В.

Брест 2025

Цель: исследовать применение алгоритмов трекинга на базе обученной сети-детектора объектов

Общее задание

1. Используя сеть-детектор, обученный в ЛР 3, реализовать логику для отслеживания множественных объектов, используя библиотеку Ultralytics YOLO;
2. Применять алгоритмы BoT-Sort и ByteTrack (задействовать соответствующие конфигурационные файлы);
3. Исследовать изменения параметров в конфигурационных файлах и их влияние на качество трекинга;
4. В качестве исходных видеоматериалов для экспериментов использовать видео-ролики из сети (например, из YouTube), содержащие множественные объекты классов из ЛР 3;
5. Оформить отчет по выполненной работе, залить исходный код и отчет в соответствующий репозиторий на github.

Код программы :

```
import os
import torch
from ultralytics import YOLO
from google.colab import drive, files

MODEL_PATH =
"/content/drive/MyDrive/yolo_rock_paper_scissors_train/train/weights/best.pt"
VIDEO_PATH =
"/content/drive/MyDrive/yolo_rock_paper_scissors_train/train/IMG_1829.MOV"
SAVE_DIR = "/content/drive/MyDrive/yolo_tracking_results"

os.makedirs(SAVE_DIR, exist_ok=True)

print("\n Подключение Google Drive ---")
drive.mount('/content/drive', force_remount=True)
print("Google Drive успешно подключен.")

print("\n Загрузка обученной модели ---")
try:
    model = YOLO(MODEL_PATH)
    print(f"Модель '{MODEL_PATH.split('/')[-1]}' успешно загружена.")
except Exception as e:
    print(f"Ошибка загрузки модели ({e}). Проверьте путь.")
    model = YOLO("yolov10m.pt")
    print("Загружена предобученная модель YOL0v10m.pt.")

print("\n Запуск отслеживания с ByteTrack (bytetrack.yaml) ---")
try:
    model.track(
        source=VIDEO_PATH, tracker="bytetrack.yaml", conf=0.25, iou=0.5,
        save=True, project=SAVE_DIR, name="bytetrack_default"
    )
    print(f"Результаты ByteTrack сохранены в: {SAVE_DIR}/bytetrack_default")
except Exception as e:
    print(f"Ошибка при выполнении ByteTrack: {e}")

print("\n Запуск отслеживания с BoT-Sort (botsort.yaml) ---")
try:
    model.track(
        source=VIDEO_PATH, tracker="botsort.yaml", conf=0.25, iou=0.5,
        save=True, project=SAVE_DIR, name="botsort_default"
    )
```

```

    )
    print(f"Результаты BoT-Sort сохранены в: {SAVE_DIR}/botsort_default")
except Exception as e:
    print(f"Ошибка при выполнении BoT-Sort: {e}")

bytetrack_modified_config = """
tracker_type: bytetrack
track_high_thresh: 0.5
track_low_thresh: 0.1
new_track_thresh: 0.6
track_buffer: 50
match_thresh: 0.8

# Эти параметры обязательны, чтобы не было ошибки 'AttributeError':
fuse_score: True          # Включает слияние оценок (fuse score)
gmc_method: sparseOptFlow # Метод компенсации движения камеры
proximity_thresh: 0.5
appearance_thresh: 0.25
with_reid: False
"""

with open("bytetrack_modified.yaml", "w") as f:
    f.write(bytetrack_modified_config)

print("Создан 'bytetrack_modified.yaml' (track_buffer=50)")

print("\n Запуск отслеживания с ByteTrack (track_buffer=50) ---")

try:
    results_byte_modified = model.track(
        source=VIDEO_PATH,
        tracker="bytetrack_modified.yaml",
        project=SAVE_DIR,
        name='bytetrack_50',
        save=True
    )
    print(f"✓ Результаты ByteTrack (buffer=50) сохранены в:
{SAVE_DIR}/bytetrack_50")
except Exception as e:
    print(f"Ошибка при выполнении измененного ByteTrack: {e}")

```

Результат программы:

Изучив полученные видео-результаты пришла к выводу:

Трекер	Конфигурация	Результат	Причина
BoT-Sort	По умолчанию	Лучший	Демонстрирует наибольшую стабильность, меньше ошибок переключения ID (ID Switch) и лучше справляется с временными потерями объектов (например, когда рука быстро движется или меняет позу).
ByteTrack	По умолчанию	Хуже, чем BoT-Sort	Чаще теряет треки или "переключает" ID, особенно в моменты быстрых смен жестов.

Вывод: исследовал применение алгоритмов трекинга на базе обученной сети-детектора объектов.