

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

Отчет по лабораторной работе 2

Специальность ИИ-23

Выполнил:

Макарович Н.Р.

Студент группы ИИ-23

Проверил:

Андренко К. В.

Преподаватель-стажёр
Кафедры ИИТ,

«___» _____ 2025 г.

Цель: осуществлять обучение нейросетевого детектора для решения задачи обнаружения заданных объектов

Общее задание

1. Базируясь на своем варианте, ознакомится с выборкой для обучения детектора, выполнить необходимые преобразования данных для организации процесса обучения (если это нужно!);
2. Для заданной архитектуры нейросетевого детектора организовать процесс обучения для своей выборки. Оценить эффективность обучения на тестовой выборке (mAP);
3. Реализовать визуализацию работы детектора из пункта 1 (обнаружение знаков на отдельных фотографиях из сети Интернет);
4. Оформить отчет по выполненной работе, залить исходный код и отчет в соответствующий репозиторий на github.

Код программы:

```
!pip install ultralytics roboflow --quiet
```

```
import torch, ultralytics, roboflow, os
```

```
print("Torch:", torch.__version__)
```

```
print("CUDA доступна:", torch.cuda.is_available())
```

```
print("Ultralytics:", ultralytics.__version__)
```

```
print("Roboflow:", roboflow.__version__)
```

```
from roboflow import Roboflow
```

```
rf = Roboflow(api_key="AnQbLkQeH26zFaNvRRv9")
```

```
project = rf.workspace("roboflow-100").project("vehicles-q0x2v")
```

```
version = project.version(2)
```

```
dataset = version.download("yolov11")
```

```
from ultralytics import YOLO
```

```
model = YOLO("yolo11s.pt")
```

```
data_yaml_path = os.path.join(dataset.location, "data.yaml")
```

```
results = model.train(  
    data=data_yaml_path,  
    epochs=3,  
    imgsz=416,  
    batch=8,  
    workers=1,  
    name="vehicles_yolo11s_fast",  
    device=0 if torch.cuda.is_available() else 'cpu',  
)
```

```
from ultralytics import YOLO
```

```
best_model_path = "runs/detect/vehicles_yolo11s_fast/weights/best.pt"
```

```
model = YOLO(best_model_path)
```

```
metrics = model.val(data=data_yaml_path)
```

```
print(metrics)
```

```
import matplotlib.pyplot as plt
```

```
from PIL import Image
```

```
results_img = "runs/detect/vehicles_yolo11s_fast/results.png"
```

```
if os.path.exists(results_img):
```

```
    img = Image.open(results_img)
```

```
    plt.imshow(img)
```

```
    plt.axis("off")
```

```
    plt.show()
```

```
else:
```

```
    print("Файл results.png не найден (проверь путь).")
```

```
from ultralytics import YOLO
```

```
from PIL import Image
```

```
from IPython.display import display
```

```
import torch
```

```
model = YOLO(best_model_path)
```

```
image_paths = [
```

```
    "/content/111.jpg",
```

```
    "/content/222.jpg",
```

```
]
```

```
for i, img_path in enumerate(image_paths):
```

```
    img = Image.open(img_path).convert("RGB")
```

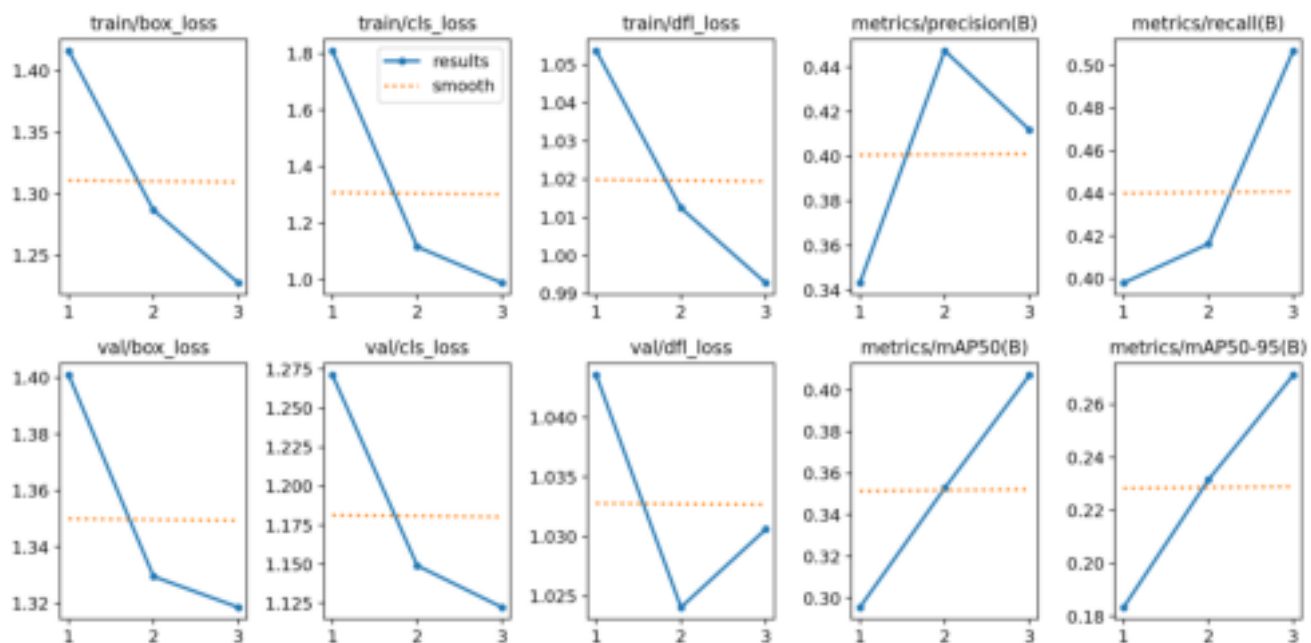



Рис. 3 Метрики



Рис. 4 Пример детекции

Вывод:

Осуществил обучение нейросетевого детектора для решения задачи обнаружения заданных объектов.