EXP NO: 15

27.10.25

IMPLEMENT A YOLO MODEL TO DETECT OBJECTS

AIM

To implement a YOLO model to detect objects

Objective

To study how YOLO detects multiple objects in a single image.

To understand the architecture and working a pre-trained YOLO model.

To use transfer learning for custom object decisions.

pseudocode

Install and import (YOLO package)

load a pre-trained YOLO model

load a test image or use a camera frame.

Run the model predict() method to detect objects.

Display bonding boxes and class label.

Save the annotated output image

Observation

The pre-trained YOLO VSS model successfully detected multiple objects such as cars, buses person in single frame.

Each object was enclosed in a bounding box with a class label and confidence score.

The inference time per image was very low

The model demonstrated strong generalization without additional training.

The visualization clearly showed YOLO's ability to detect overlapping objects complex scenes.

Result

Successfully Implement pre-trained YOLO.

image 1 / 1 content / IMG-20250523-WA0017.jpg 640X480

3 persons. 7.8 ms

Speed: 8 preprocess: 7.3 ms inference 2.5 ms

postprocess pon image A shape (1, 3, 640, 480)

output

Detected objects

person

person

person

```
detected_img_path = os.path.join(output_folder, os.path.basename(img_path))
if os.path.exists(detected_img_path):
    detected_img = Image.open(detected_img_path)
    plt.imshow(detected_img)
    plt.axis('off')
    plt.title("YOLOv5 Object Detection Result")
    plt.show()
```

```
 Image already exists: street.jpg

 Loading YOLOv5 model (this may take a few seconds)...
Creating new Ultralytics Settings v0.0.6 file ✅
View Ultralytics Settings with 'yolo settings' or at '/root/.config/Ultralytics/settings.json'
Update Settings with 'yolo settings key=value', i.e. 'yolo settings runs_dir=path/to/dir'. For help see https://docs.ultraly
Using cache found in /root/.cache/torch/hub/ultralytics_yolov5_master
YOLOv5 🚀 2025-11-1 Python-3.12.12 torch-2.8.0+cu126 CPU
```

https://colab.research.google.com/drive/1n1XghI5JqaDe3R2jFQdd5Sn9MUNRvhMf#scrollTo=z_4zjquYa71U

9.ipynb

File Edit View Insert Runtime Tools Help

Share

Commands + Code + Text Run all Reconnect

```
print("\n Loading YOLOv5 model (this may take a few seconds)...")
model = torch.hub.load('ultralytics/yolov5', 'yolov5s', pretrained=True)

# Step 3: Perform inference on the image
img = Image.open(img_path)
results = model(img)

# Step 4: Print detection results
print("\n Detection Results")
results.print()  # prints detected objects, confidence, coordinates

# Step 5: Displa, the image with bounding boxes
results.show()  # opens the image window with labels and boxes

# Step 6: (Optional) Save output image to a folder
output_folder = "yolo_output"
results.save(save_dir=output_folder)
print(f"\n Detection completed! Output saved in folder: '{output_folder}'")

# -------------------------
```

Follow link (ctrl + click)

Variables Terminal

9:19 AM

ENG IN 14:59 03-11-2025

https://colab.research.google.com/drive/1n1XghI5JqaDe3R2jFQdd5Sn9MUNRvhMf#scrollTo=z_4zjquYa71U

9.ipynb

File Edit View Insert Runtime Tools Help

Share

Commands + Code + Text Run all Reconnect

[ ]

```
#lab 15

import torch
from PIL import Image
import matplotlib.pyplot as plt
import urllib.request
import os

# Step 1: Automatically download a sample image
img_path = "street.jpg"
if not os.path.exists(img_path):
    url = "https://ultralytics.com/images/zidane.jpg"  # sample image
    urllib.request.urlretrieve(url, img_path)
    print(" Sample image downloaded successfully: street.jpg")
else:
    print(" Image already exists:", img_path)

# Step 2: Load the pre-trained YOLOv5 model
# This will download YOLOv5 automatically from Ultralytics (first time only)
```

Variables Terminal

9:19 AM

ENG IN 14:59 03-11-2025

https://colab.research.google.com/drive/1n1XghI5JqaDe3R2jFQdd5Sn9MUNRvhMf#scrollTo=Dgr7136VtflX

9.ipynb

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all

Reconnect

Share

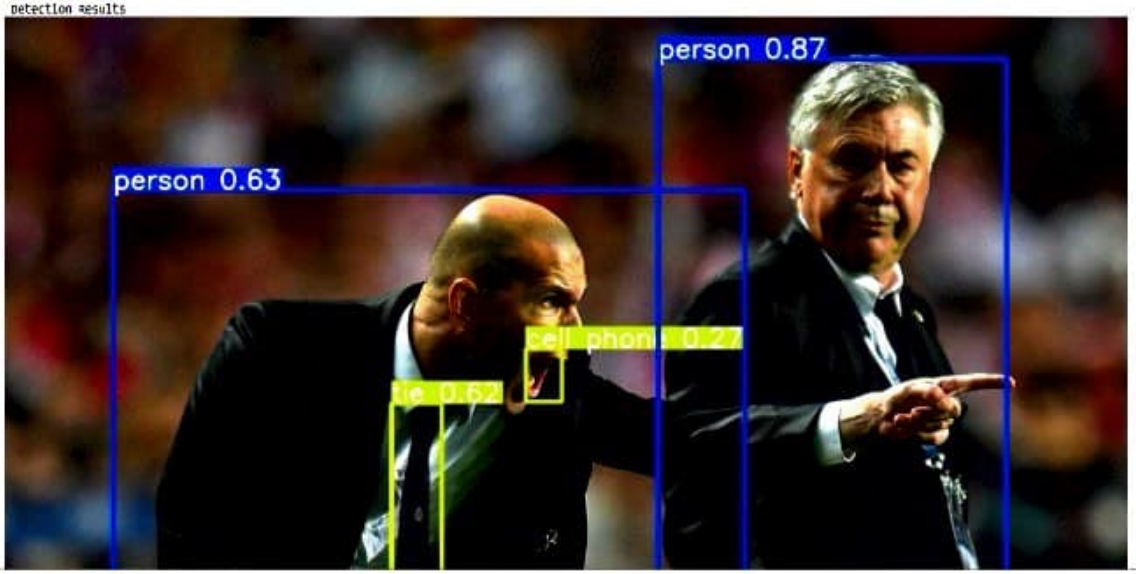
```
Image already exists: street.jpg

Loading YOLOv5 model (this may take a few seconds)...
Creating new Ultralytics Settings v0.0.6 file
View Ultralytics Settings with 'yolo settings' or at '/root/.config/Ultralytics/settings.json'
Update Settings with 'yolo settings key=value', i.e. 'yolo settings runs_dir=path/to/dir'. For help see https://docs.ultralytics.com/quickstart/#ultralytics-settings.
Using cache found in /root/.cache/torch/hub/ultralytics_yolov5_master
YOLOv5 🚀 2025-11-1 Python-3.12.12 torch-2.8.0+cu126 CPU

Downloading https://github.com/ultralytics/yolov5/releases/download/v7.0/yolov5s.pt to yolov5s.pt...
100%|██████████| 14.1M/14.1M [00:00<00:00, 114MB/s]

Fusing layers...
YOLOv5s summary: 213 layers, 7225885 parameters, 0 gradients, 16.4 GFLOPs
Adding AutoShape...
/root/.cache/torch/hub/ultralytics_yolov5_master/models/common.py:906: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with amp.autocast(autocast):
image 1/1: 
Speed: 69..
 1 phone
, 45.7ms NMS per image at shape (1, 3, 384, 640)

Detection Results
```



Variables Terminal

9:19 AM

ENG IN 15:01 03-11-2025