Configuration Manual

MSc Research Project

Programme Name: MSc in Data Analytics

Name : Lakshmi Narasimha Kundeti

Student ID: x22244972@student.ncirl.ie

School of Computing

National College of Ireland

Supervisor:Arjun Chikkankod

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: ……. Lakshmi…Narasimha…Kundeti …………………………………………………….

Student ID: x22244972 …………………………………………………………………………………………….……

Programme: …… MSc..in…Data…Analytics… ………………… **Year:** ……2024…………………..

Module: …………Research Project…………………………………………………………………………

Supervisor: …… …Arjun…Chikkankod… ……………………………………………………………………

Submission Due Date: …………… 16/09/2024… ………………………………………………………………………

Project Title: Redefining Public Safety: A Comparative Analysis of RT-DETR and YOLOv8 - Unveiling the Future of Real-Time Handgun Detection ………………………………………………………………………………………………….………

Word Count: ……………679…………………… **Page Count**…8…………………………………………….…….. ✓

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……………k.l.Narasimha………………………………………………………………………………………

Date: ………………16/09/2024……………………………………………………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Name : Lakshmi Narasimha Kundeti
Student ID: x22244972@student.ncirl.ie

1 INTRODUCTION

This research project is aimed at making a comparative study between the RT-DETR and YOLOv8 models about real-time handgun detection in public places. In this paper, an evaluation will be made for the two outstanding object detection models with respect to several conditions such as accuracy, speed, robustness, and adaptability. In this work, a complete dataset of handgun images is being used with rigorous data augmentation methods so that the hyperparameter tuning might be done efficiently. This will eventually be helpful in getting insights into the development of more effective and efficient AI-driven security solutions for public safety and surveillance systems.

2 SETUP GOOGLE COLAB

2.1 GOOGLE DRIVE COLAB CONNECTION SETUP

➔ Go to colab.google.com and select new notebook and upload the notebook

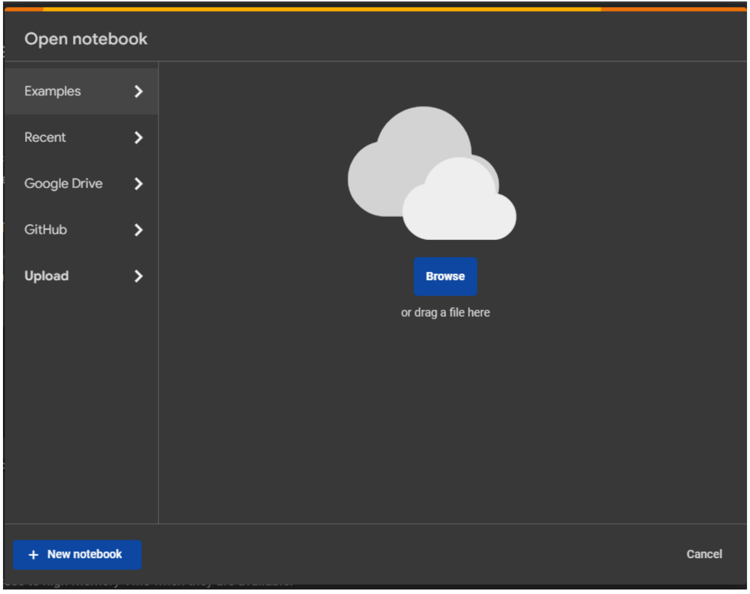
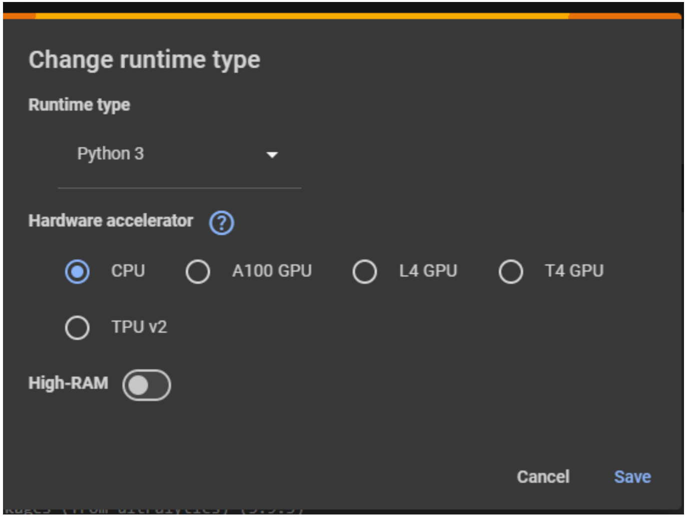


Figure 1 Upload Notebook

➔ Connect the session by selecting the T4 GPU



Change runtime type

Runtime type: Python 3

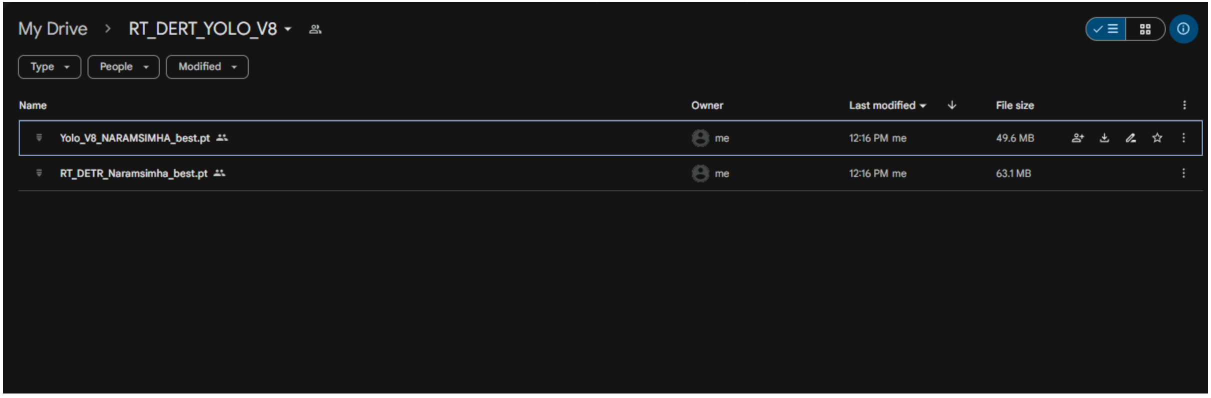
Hardware accelerator

● CPU ○ A100 GPU ○ L4 GPU ○ T4 GPU

○ TPU v2

High-RAM

➔ NEXT GO DRIVE.GOOGLE.COM AND create directory any name and upload the model file

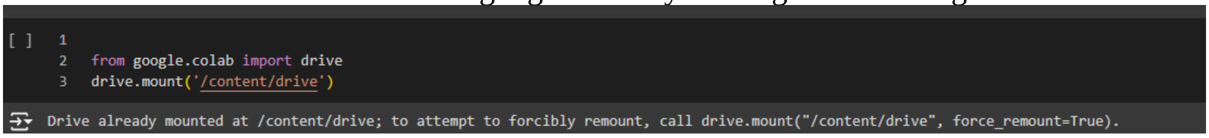


My Drive › RT_DERT_YOLO_V8

Name	Owner	Last modified	File size
Yolo_V8_NARAMSIMHA_best.pt	me	12:16 PM me	49.6 MB
RT_DETR_Naramsimha_best.pt	me	12:16 PM me	63.1 MB

Figure 2 Google Drive

→Now mount the colab notebook to google drive by running the following cell:



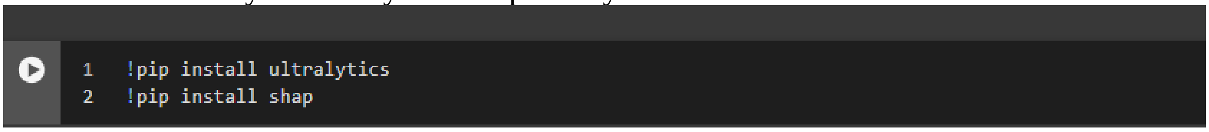
```

from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

Figure 3 Mounting the google drive

→Install the ultralytics library and Shap library

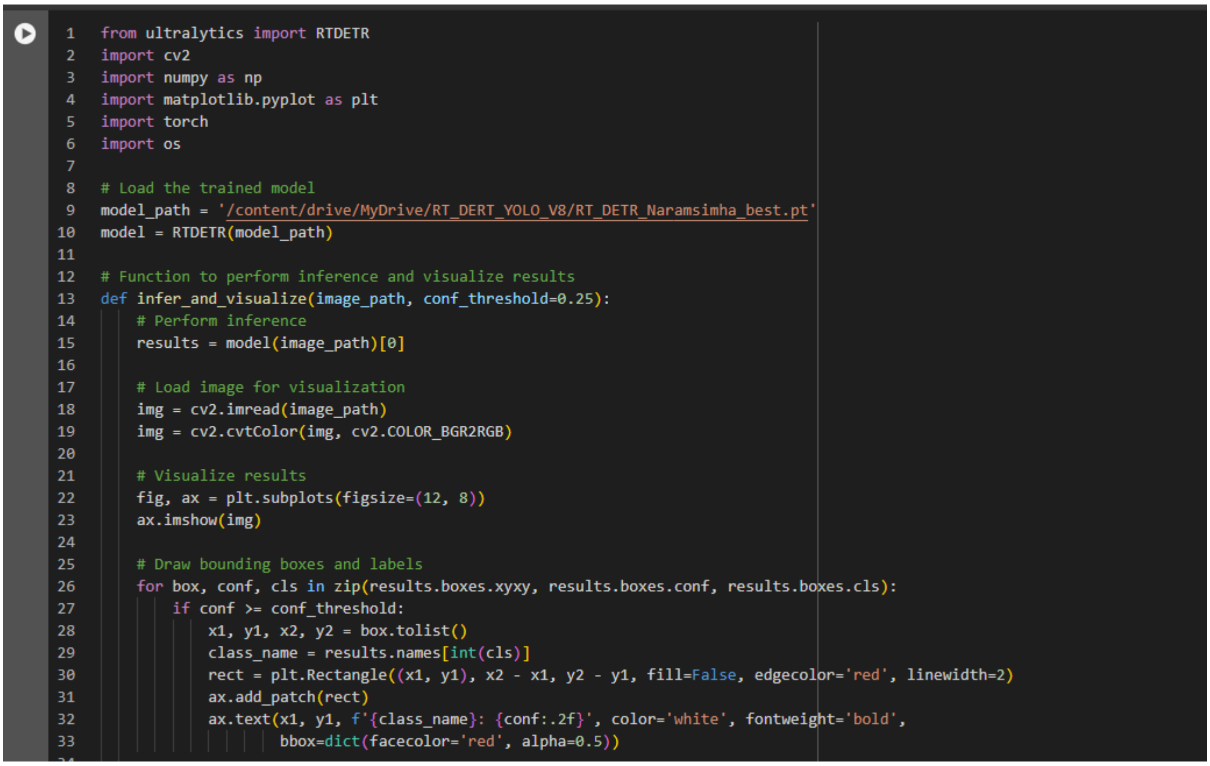


```
!pip install ultralytics
!pip install shap
```

Figure 4 Installation of the library

- ➔ Change the model paths for YOLO v8 and RT DETR

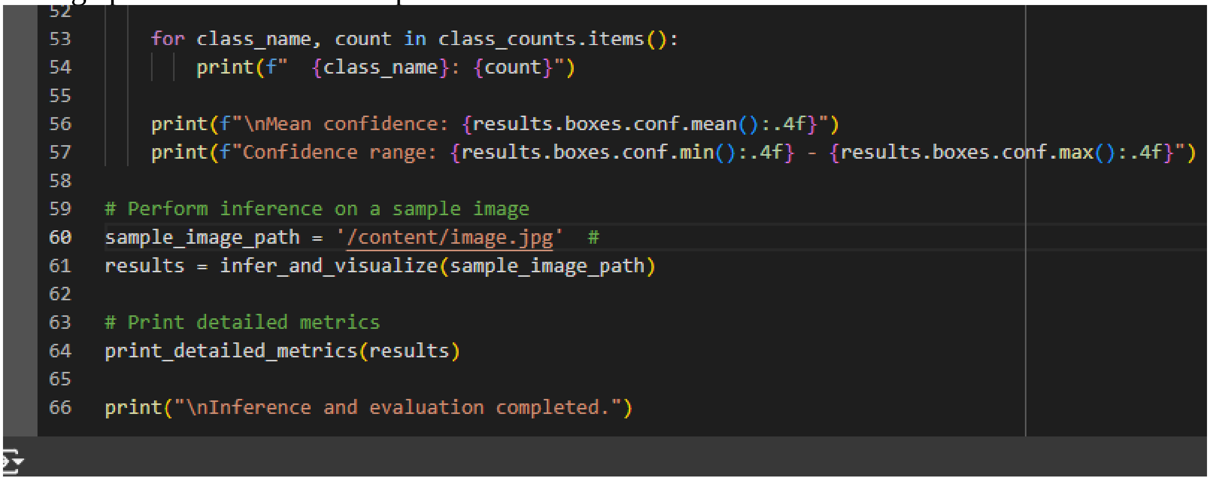
For YOLO v8 change the model and image path with respect to current working directory;
chan



```
 1  from ultralytics import RTDETR
 2  import cv2
 3  import numpy as np
 4  import matplotlib.pyplot as plt
 5  import torch
 6  import os
 7
 8  # Load the trained model
 9  model_path = '/content/drive/MyDrive/RT_DERT_YOLO_V8/RT_DETR_Naramsimha_best.pt'
10  model = RTDETR(model_path)
11
12  # Function to perform inference and visualize results
13  def infer_and_visualize(image_path, conf_threshold=0.25):
14      # Perform inference
15      results = model(image_path)[0]
16
17      # Load image for visualization
18      img = cv2.imread(image_path)
19      img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
20
21      # Visualize results
22      fig, ax = plt.subplots(figsize=(12, 8))
23      ax.imshow(img)
24
25      # Draw bounding boxes and labels
26      for box, conf, cls in zip(results.boxes.xyxy, results.boxes.conf, results.boxes.cls):
27          if conf >= conf_threshold:
28              x1, y1, x2, y2 = box.tolist()
29              class_name = results.names[int(cls)]
30              rect = plt.Rectangle((x1, y1), x2 - x1, y2 - y1, fill=False, edgecolor='red', linewidth=2)
31              ax.add_patch(rect)
32              ax.text(x1, y1, f'{class_name}: {conf:.2f}', color='white', fontweight='bold',
33                      bbox=dict(facecolor='red', alpha=0.5))
```

Figure 5 Model path for RT DETR

Change path in line 9 with respect where model file is located.

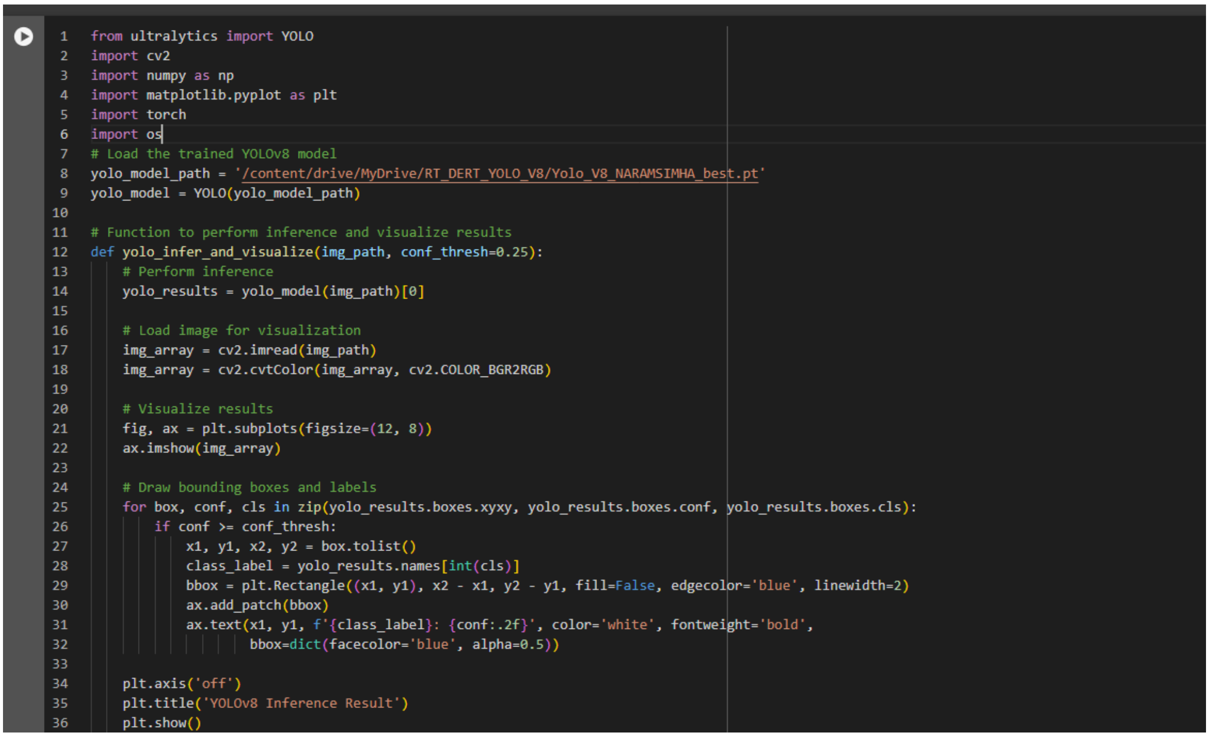


```
52
53      for class_name, count in class_counts.items():
54          print(f"  {class_name}: {count}")
55
56      print(f"\nMean confidence: {results.boxes.conf.mean():.4f}")
57      print(f"Confidence range: {results.boxes.conf.min():.4f} - {results.boxes.conf.max():.4f}")
58
59  # Perform inference on a sample image
60  sample_image_path = '/content/image.jpg'  #
61  results = infer_and_visualize(sample_image_path)
62
63  # Print detailed metrics
64  print_detailed_metrics(results)
65
66  print("\nInference and evaluation completed.")
```

Figure 6 Image path for RT DETR

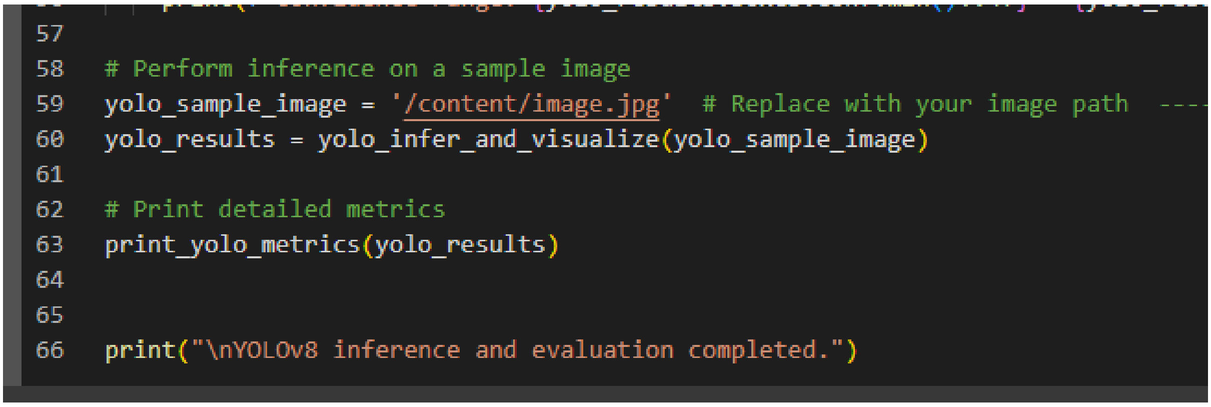
Change path at line 60 ,with respect to the location of the image .

- ➔ Follow the same procedure for YOLO v8 and Shap analysis as well. Change the model path and image according to their location.



```
1   from ultralytics import YOLO
2   import cv2
3   import numpy as np
4   import matplotlib.pyplot as plt
5   import torch
6   import os
7   # Load the trained YOLOv8 model
8   yolo_model_path = '/content/drive/MyDrive/RT_DERT_YOLO_V8/Yolo_V8_NARAMSIMHA_best.pt'
9   yolo_model = YOLO(yolo_model_path)
10
11  # Function to perform inference and visualize results
12  def yolo_infer_and_visualize(img_path, conf_thresh=0.25):
13      # Perform inference
14      yolo_results = yolo_model(img_path)[0]
15
16      # Load image for visualization
17      img_array = cv2.imread(img_path)
18      img_array = cv2.cvtColor(img_array, cv2.COLOR_BGR2RGB)
19
20      # Visualize results
21      fig, ax = plt.subplots(figsize=(12, 8))
22      ax.imshow(img_array)
23
24      # Draw bounding boxes and labels
25      for box, conf, cls in zip(yolo_results.boxes.xyxy, yolo_results.boxes.conf, yolo_results.boxes.cls):
26          if conf >= conf_thresh:
27              x1, y1, x2, y2 = box.tolist()
28              class_label = yolo_results.names[int(cls)]
29              bbox = plt.Rectangle((x1, y1), x2 - x1, y2 - y1, fill=False, edgecolor='blue', linewidth=2)
30              ax.add_patch(bbox)
31              ax.text(x1, y1, f'{class_label}: {conf:.2f}', color='white', fontweight='bold',
32                      bbox=dict(facecolor='blue', alpha=0.5))
33
34      plt.axis('off')
35      plt.title('YOLOv8 Inference Result')
36      plt.show()
```

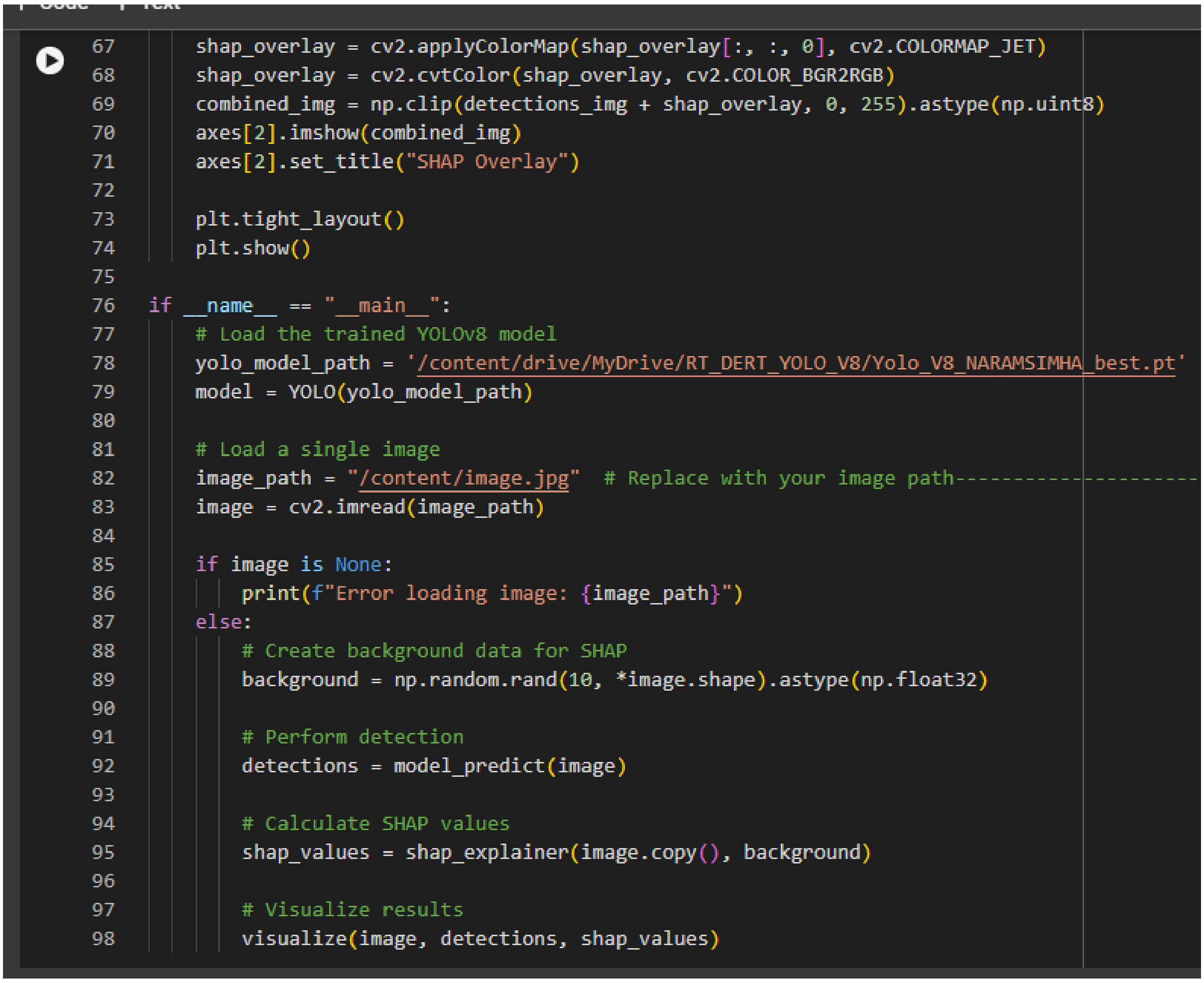
Figure 7 Change path in line 8 of YOLO v8 Model



```
57
58  # Perform inference on a sample image
59  yolo_sample_image = '/content/image.jpg'   # Replace with your image path  ----
60  yolo_results = yolo_infer_and_visualize(yolo_sample_image)
61
62  # Print detailed metrics
63  print_yolo_metrics(yolo_results)
64
65
66  print("\nYOLOv8 inference and evaluation completed.")
```

Figure 8 Change path in line 59 for the image

2.2 Shap analysis

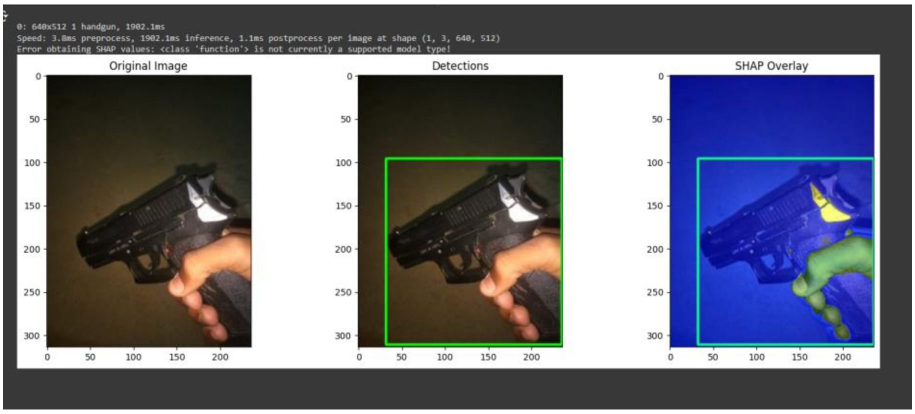


```
67      shap_overlay = cv2.applyColorMap(shap_overlay[:, :, 0], cv2.COLORMAP_JET)
68      shap_overlay = cv2.cvtColor(shap_overlay, cv2.COLOR_BGR2RGB)
69      combined_img = np.clip(detections_img + shap_overlay, 0, 255).astype(np.uint8)
70      axes[2].imshow(combined_img)
71      axes[2].set_title("SHAP Overlay")
72
73      plt.tight_layout()
74      plt.show()
75
76  if __name__ == "__main__":
77      # Load the trained YOLOv8 model
78      yolo_model_path = '/content/drive/MyDrive/RT_DERT_YOLO_V8/Yolo_V8_NARAMSIMHA_best.pt'
79      model = YOLO(yolo_model_path)
80
81      # Load a single image
82      image_path = "/content/image.jpg"  # Replace with your image path--------------------
83      image = cv2.imread(image_path)
84
85      if image is None:
86          print(f"Error loading image: {image_path}")
87      else:
88          # Create background data for SHAP
89          background = np.random.rand(10, *image.shape).astype(np.float32)
90
91          # Perform detection
92          detections = model_predict(image)
93
94          # Calculate SHAP values
95          shap_values = shap_explainer(image.copy(), background)
96
97          # Visualize results
98          visualize(image, detections, shap_values)
```

Figure 9 Shap analysis

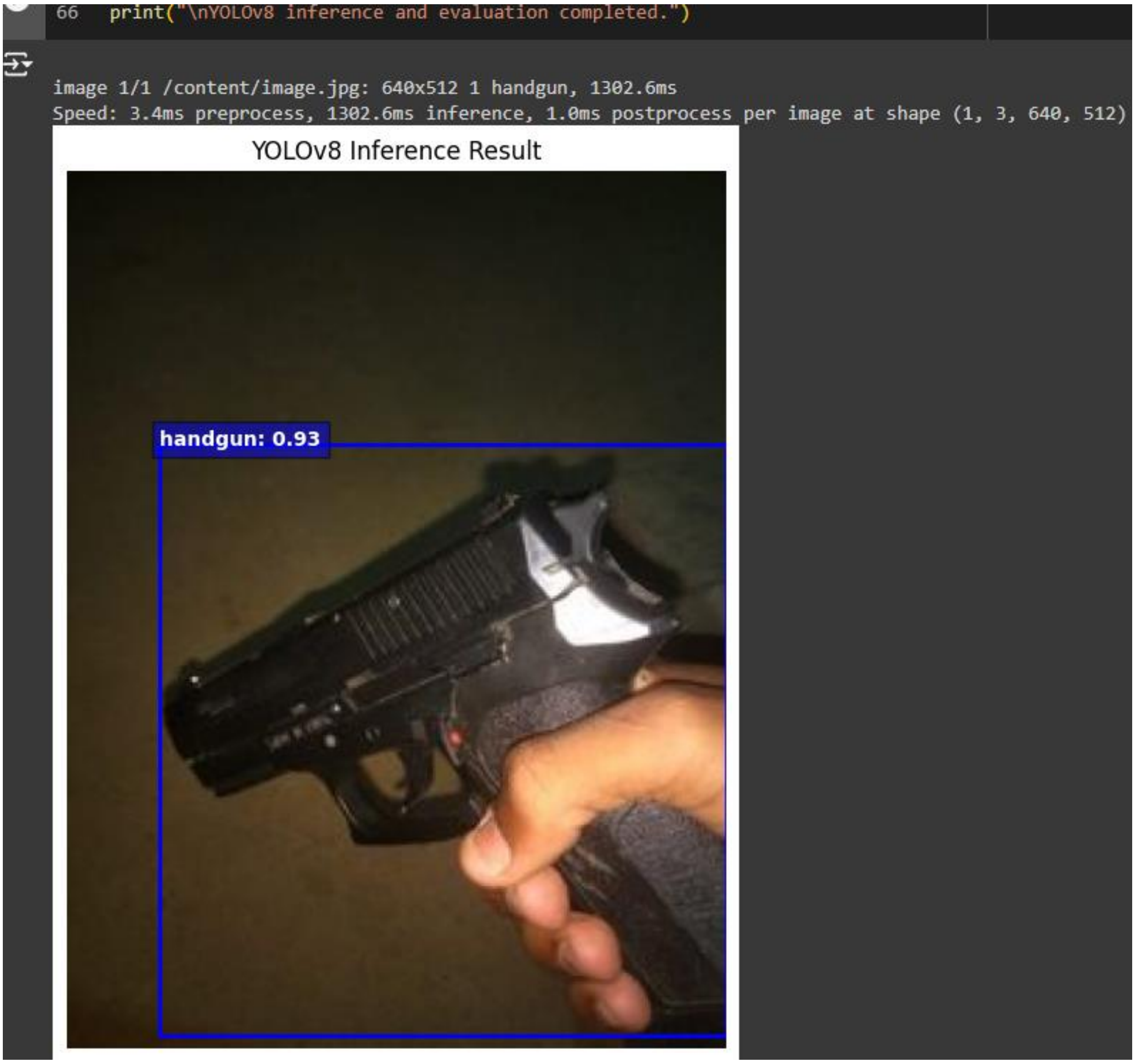
Change model path and image at line 78 and 82

2.3 outputs



```
0: 640x512 1 handgun, 1902.1ms
Speed: 3.8ms preprocess, 1902.1ms inference, 1.1ms postprocess per image at shape (1, 3, 640, 512)
Error obtaining SHAP values: <class 'function'> is not currently a supported model type!
```

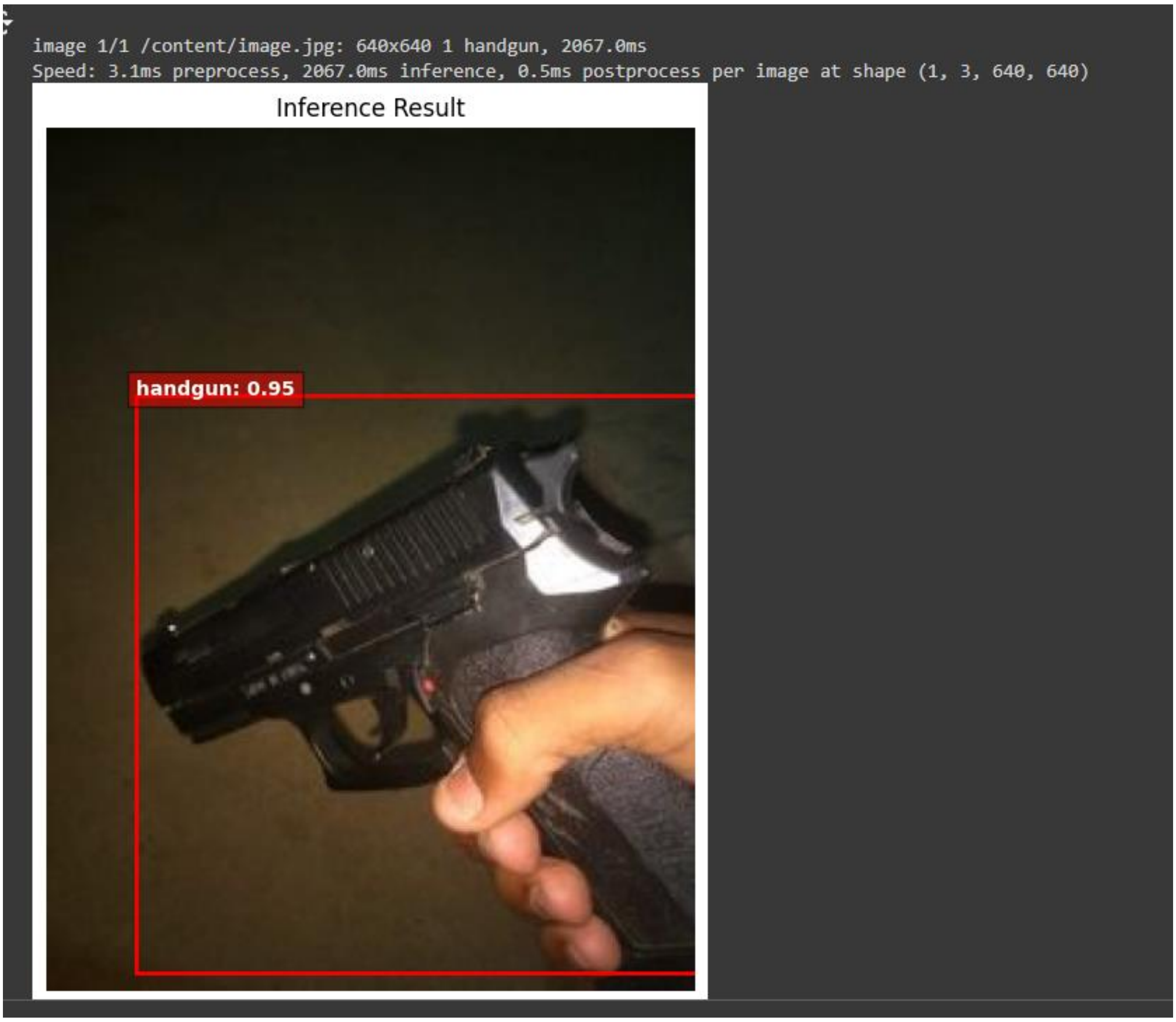
Figure 10 shap analysis with YOLO v8



```
print("\nYOLOv8 inference and evaluation completed.")
```

```
image 1/1 /content/image.jpg: 640x512 1 handgun, 1302.6ms
Speed: 3.4ms preprocess, 1302.6ms inference, 1.0ms postprocess per image at shape (1, 3, 640, 512)
```

Figure 11 YOLO v8 results



```
image 1/1 /content/image.jpg: 640x640 1 handgun, 2067.0ms
Speed: 3.1ms preprocess, 2067.0ms inference, 0.5ms postprocess per image at shape (1, 3, 640, 640)
```

Figure 12 RT DETR results

References

Ultralytics. (n.d.). *GitHub - ultralytics/ultralytics: NEW - YOLOv8 🚀 in PyTorch > ONNX > OpenVINO > CoreML > TFLite*. GitHub. https://github.com/ultralytics/ultralytics

Welcome to the SHAP documentation — SHAP latest documentation. (n.d.). https://shap.readthedocs.io/en/latest/

Ultralytics. (2024, July 4). *RT-DETR (Realtime Detection Transformer)*. Ultralytics YOLO Docs. https://docs.ultralytics.com/models/rtdetr/

Google Colab. (n.d.). https://colab.research.google.com/