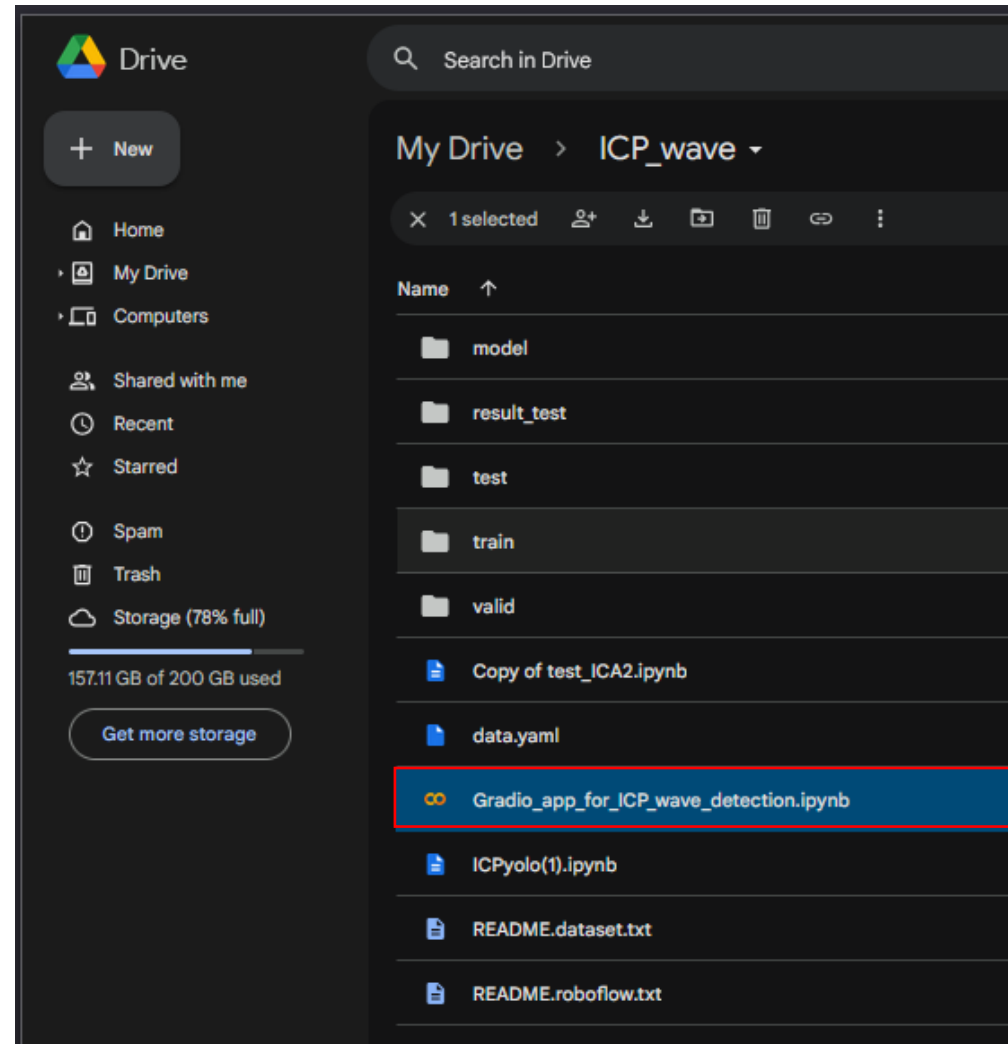
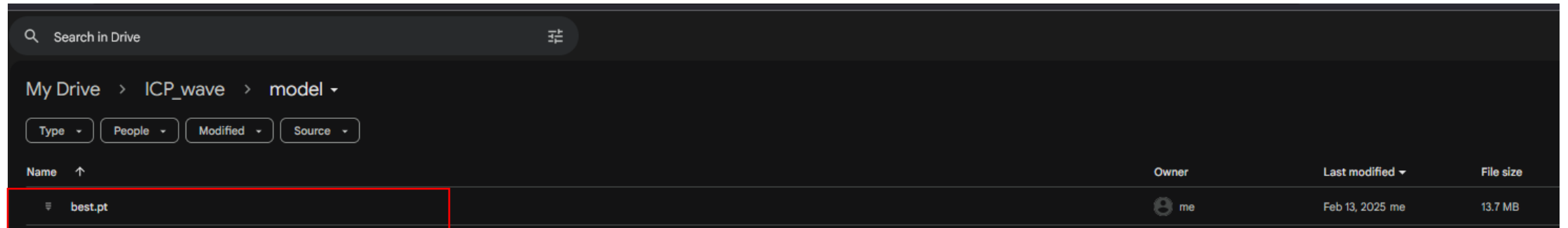


Gradio application for detection
of poor compliance ICP wave

Upload file.ipynb in your google drive



Upload best.pt in your google drive

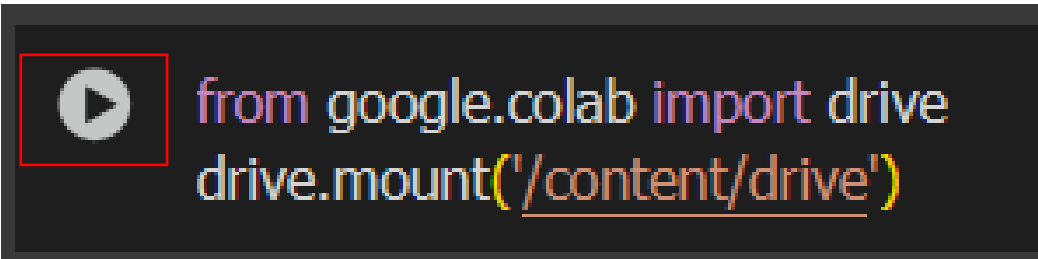


The screenshot shows the Google Drive interface. At the top, there is a search bar labeled 'Search in Drive'. Below it, the breadcrumb navigation shows 'My Drive > ICP_wave > model'. There are four filter buttons: 'Type', 'People', 'Modified', and 'Source'. A table lists the files in the 'model' folder. The first row shows a file named 'best.pt' with a size of 13.7 MB, owned by 'me', and last modified on 'Feb 13, 2025'. The 'best.pt' text in the first row is highlighted with a red box.

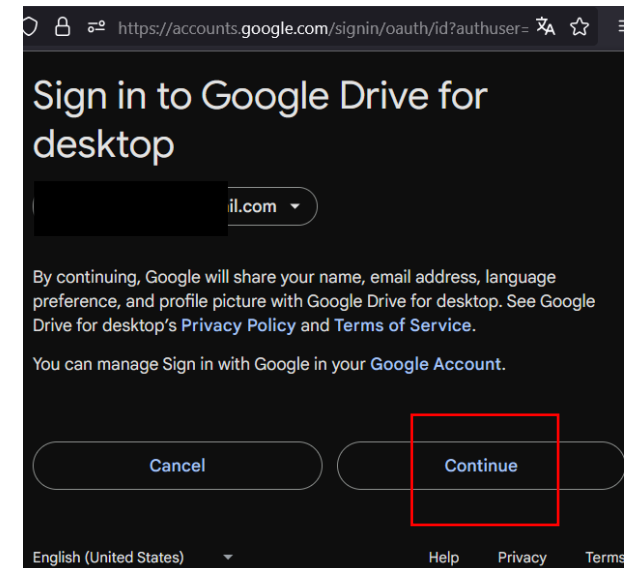
Name	Owner	Last modified	File size
best.pt	me	Feb 13, 2025	13.7 MB

Open file.ipynb and run 1st script to mount your drive

click



Then, consent to the mounting of your drive



run 2nd script to install gradio

click

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

Mounted at /content/drive

#Install Gradio Make sure Gradio is installed in your environment:
!pip install gradio
!pip install gradio ultralytics
```

In 3rd script, modify to your path and then start script

```
#YOLOv5 Inference with Gradio
import torch
import gradio as gr
from PIL import Image
import cv2
import numpy as np

# Load the trained YOLOv5 model
model = torch.hub.load('ultralytics/yolov5', 'custom', path='/content/drive/MyDrive/ICP_wave/model/best.pt') # Replace with your model path

# Define a function for inference
def detect_objects(image):
    # Convert Gradio's image to a format suitable for YOLOv5
    input_image = Image.fromarray(image)

    # Run inference
    results = model(input_image)

    # Render the results on the original image
    results.render() # Render adds boxes and labels to the image

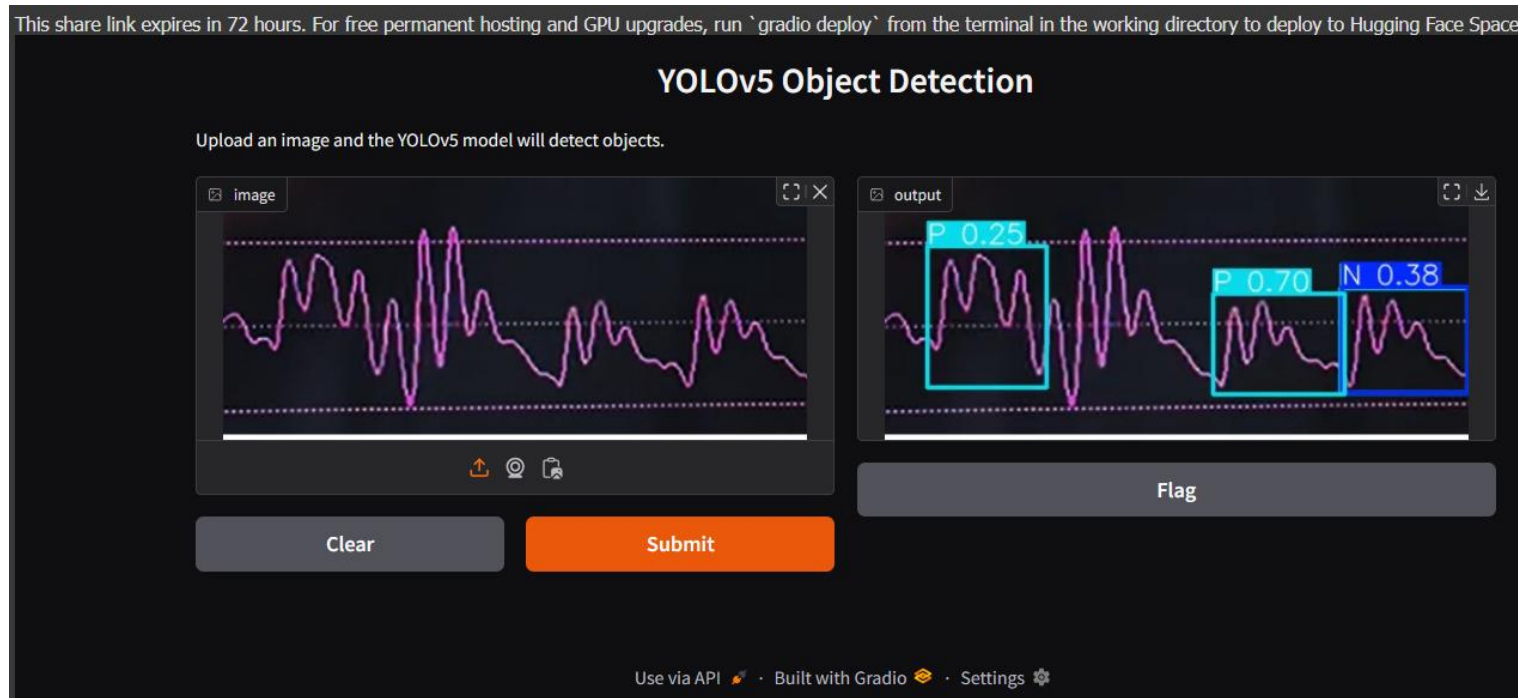
    # Convert to numpy for displaying in Gradio
    output_image = results.ims[0]

    return output_image

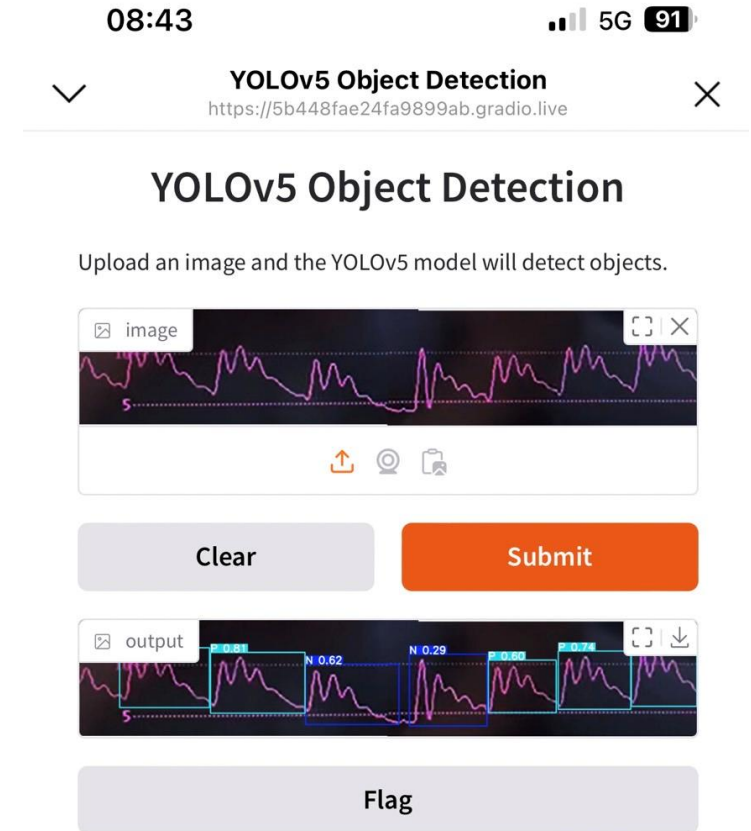
# Set up the Gradio interface
iface = gr.Interface(
    fn=detect_objects,
    inputs=gr.Image(type="numpy"), # Input as numpy array
    outputs="image", # Output as an image with bounding boxes
    title="YOLOv5 Object Detection",
    description="Upload an image and the YOLOv5 model will detect objects."
)
```

edit to your path

Gradio application will be launched



On laptop



On mobile phone