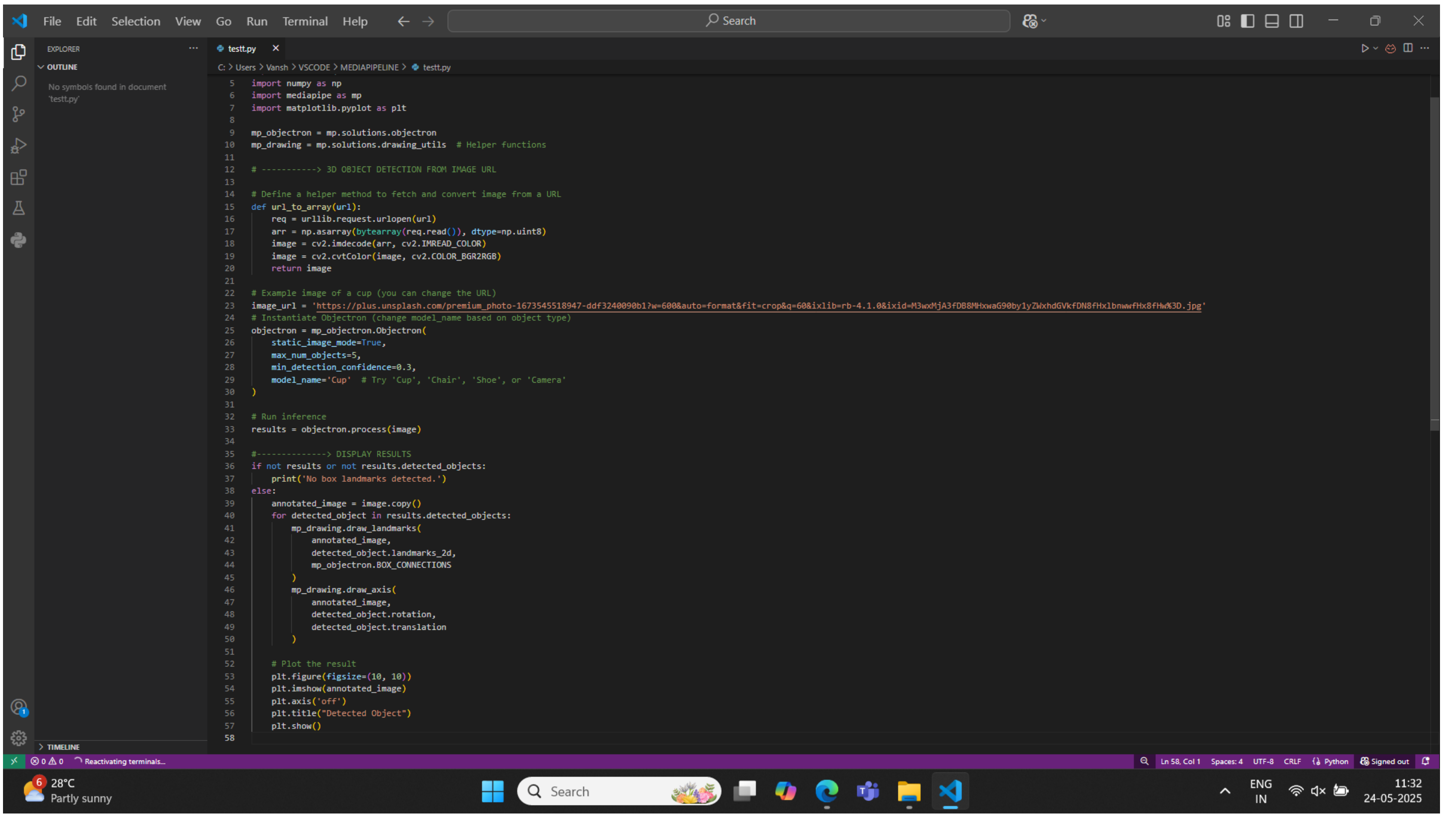
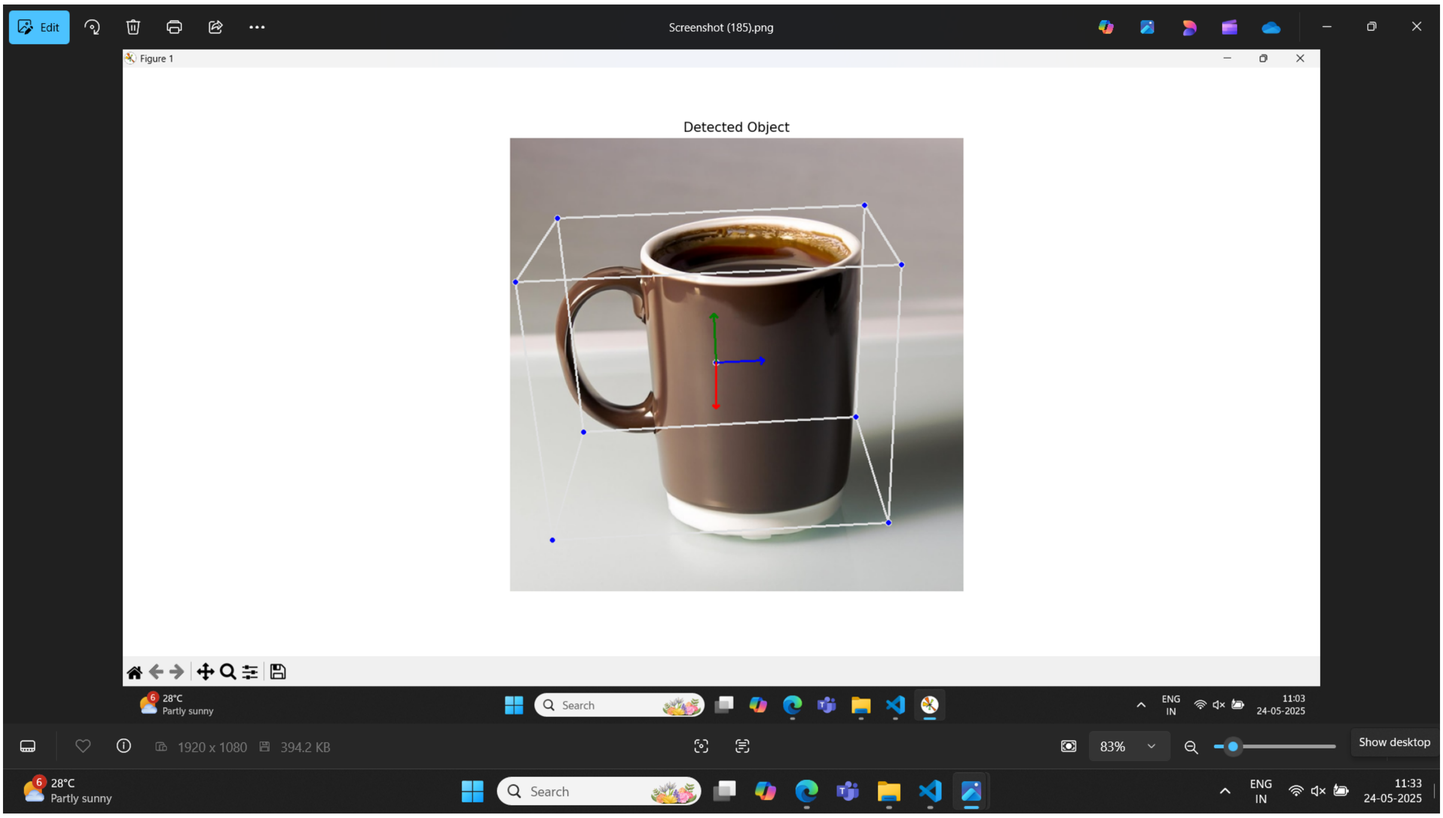
Python Code for Objectron



```python
import numpy as np
import mediapipe as mp
import matplotlib.pyplot as plt

mp_objectron = mp.solutions.objectron
mp_drawing = mp.solutions.drawing_utils  # Helper functions

# ----------> 3D OBJECT DETECTION FROM IMAGE URL

# Define a helper method to fetch and convert image from a URL
def url_to_array(url):
    req = urllib.request.urlopen(url)
    arr = np.asarray(bytearray(req.read()), dtype=np.uint8)
    image = cv2.imdecode(arr, cv2.IMREAD_COLOR)
    image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    return image

# Example image of a cup (you can change the URL)
image_url = 'https://plus.unsplash.com/premium_photo-1673545518947-ddf3240090b1?w=600&auto=format&fit=crop&q=60&ixlib=rb-4.1.0&ixid=M3wxMjA3fDB8MHxwaG90by1yZWxhdGVkfDN8fHx1bnwwfHx8fHw%3D.jpg'
# Instantiate Objectron (change model_name based on object type)
objectron = mp_objectron.Objectron(
    static_image_mode=True,
    max_num_objects=5,
    min_detection_confidence=0.3,
    model_name='Cup'  # Try 'Cup', 'Chair', 'Shoe', or 'Camera'
)

# Run inference
results = objectron.process(image)

#-------------> DISPLAY RESULTS
if not results or not results.detected_objects:
    print('No box landmarks detected.')
else:
    annotated_image = image.copy()
    for detected_object in results.detected_objects:
        mp_drawing.draw_landmarks(
            annotated_image,
            detected_object.landmarks_2d,
            mp_objectron.BOX_CONNECTIONS
        )
        mp_drawing.draw_axis(
            annotated_image,
            detected_object.rotation,
            detected_object.translation
        )

    # Plot the result
    plt.figure(figsize=(10, 10))
    plt.imshow(annotated_image)
    plt.axis('off')
    plt.title("Detected Object")
    plt.show()
```

Detected Object (Cup) Output



Edit
Screenshot (185).png
Figure 1
Detected Object
28°C
Partly sunny
Search
ENG
IN
11:03
24-05-2025
1920 x 1080
394.2 KB
83%
Show desktop
28°C
Partly sunny
Search
ENG
IN
11:33
24-05-2025