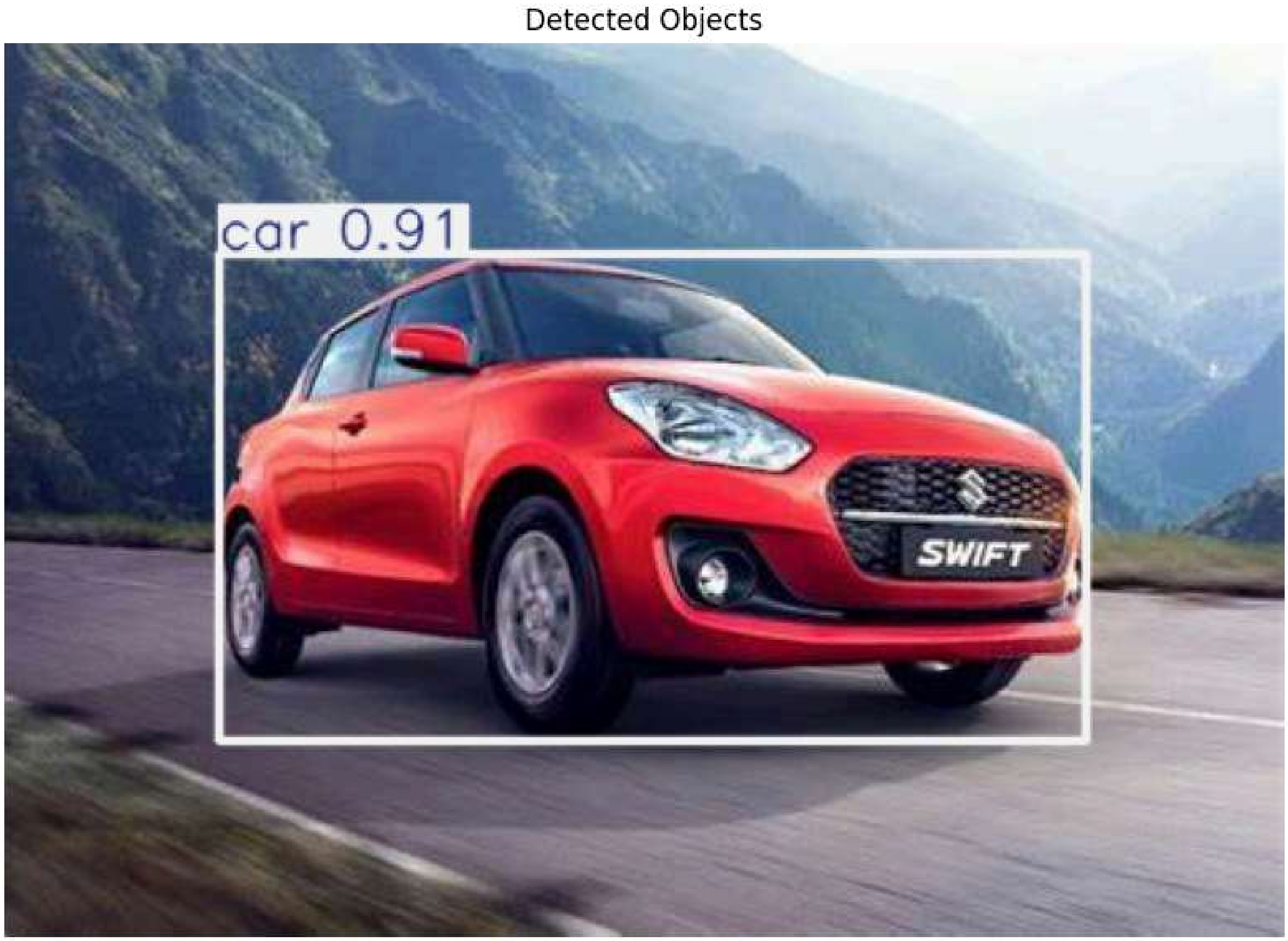
```
from ultralytics import YOLO
import cv2
import matplotlib.pyplot as plt

def display_image(image, title="Image"):
    """Display an image using matplotlib."""
    plt.figure(figsize=(10, 10))
    plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
    plt.title(title)
    plt.axis("off")
    plt.show()

# Load the YOLOv5 model (you can use 'yolov5s', 'yolov5m', etc.)
model = YOLO('yolov8n.pt')  #specify a custom model path.

# Load an image for object detection
image_path = '/content/OIP.jpeg'
image = cv2.imread(image_path)

# Perform object detection
results = model(image)

# Visualize results on the image
annotated_image = results[0].plot()

# Display the annotated image
display_image(annotated_image, title="Detected Objects")

# Save the annotated image
output_path = "output_image.jpg"
cv2.imwrite(output_path, annotated_image)
print(f"Annotated image saved to {output_path}")
```

```
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 se
Downloading https://github.com/ultralytics/assets/releases/download/v8.3.0/yolov8n.pt to 'yolov8n.pt'.
100%|██████████| 6.25M/6.25M [00:00<00:00, 281MB/s]

0: 448x640 1 car, 325.1ms
Speed: 15.6ms preprocess, 325.1ms inference, 25.2ms postprocess per image at shape (1, 3, 448, 640)
```



```
Annotated image saved to output_image.jpg
```