Step 1: Model Loading

```
import torch

# Load the pre-trained YOLO model (YOLOv5)
model = torch.hub.load('ultralytics/yolov5', 'yolov5s', pretrained=True)
```

I load a pre-trained YOLOv5 model using the PyTorch Hub.

Step 2: Load the Image

```
import cv2

# Load the image
image_path = 'image1.jpg'
image = cv2.imread(image_path)
```

In this step, I load an image from a specified path. The image should contain multiple objects based on the instructions to demonstrate the model's detection capabilities.

Step 3: Object Detection

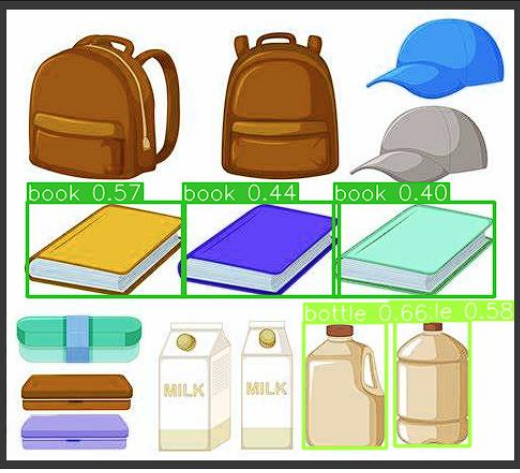
```
# Perform inference
results = model(image)
```

Here, I feed the loaded image to the YOLO model for object detection. The model processes the image and returns the results, including detected objects' coordinates, class labels, and confidence scores.

Step 4: Visualize the results

```
import matplotlib.pyplot as plt

# Display results
results.show()  # This will open a window showing the image with detected objects
```



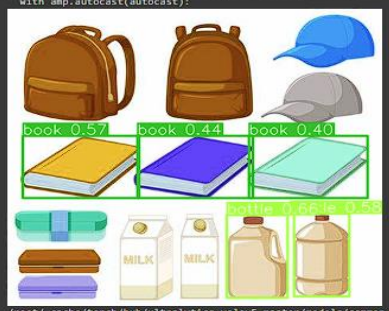
In this step, I visualize the detected objects. The `results.show()` method displays the image with bounding boxes around detected objects and labels indicating the class and confidence score. As we can see the model didn't detect all the objects in the image and detect the books and bottle's only.

Step 5: Test with other images

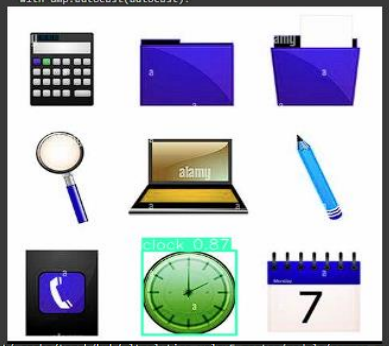
```
# List of images to test
test_images = ['image1.jpg', 'image2.jpg', 'image.jpg']

for img_path in test_images:
    img = cv2.imread(img_path)
    results = model(img)
    results.show()
```

```
/root/.cache/torch/hub/ultralytics_yolov5_master/models/common.py:892: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with amp.autocast(autocast):
```




```
/root/.cache/torch/hub/ultralytics_yolov5_master/models/common.py:892: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with amp.autocast(autocast):
```




```
/root/.cache/torch/hub/ultralytics_yolov5_master/models/common.py:892: FutureWarning: `torch.cuda.amp.autocast(args...)` is deprecated. Please use `torch.amp.autocast('cuda', args...)` instead.
  with amp.autocast(autocast):
```

I then test the model on multiple images to observe its performance. This step helps me to observe how well the model generalizes to different images with varying contents. For each image in the test set, we repeat the detection and visualization steps. Based on the results the model didn’t detect all the objects in different images too.

Conclusion

Using a pre-trained YOLOv5 model for object detection proved effective, combining speed and accuracy. Even though the model didn’t perform as I expected. This document provides a clear framework for implementing object detection and demonstrates the model's capabilities across multiple images.