EXERCISE NO. 8
OBJECT DETECTION WITH SINGLE SHOT DETECTOR USING PYTORCH

AIM:

To implement object detection with a Single Shot Detector using PyTorch.

ALGORITHM:

1. Import the necessary libraries.
2. Load the pre-trained SSD model.
3. Load the COCO labels.
4. Apply image transforms.
5. Load and preprocess the image.
6. Predict the class.
7. Load image for OpenCV display.
8. Draw boxes for detection above the confidence threshold.
9. Display the objects detected with label and confidence.

PROGRAM:

```
import torch
from torchvision.models.detection import ssd300_vgg16, SSD300_VGG16_Weights
from torchvision import transforms
import cv2
from PIL import Image
import numpy as np
from google.colab.patches import cv2_imshow

weights = SSD300_VGG16_Weights.DEFAULT
model = ssd300_vgg16(weights=weights)
model.eval()

labels = weights.meta["categories"]

preprocess = weights.transforms()

img_path = '.../giraffe.jpg'
img = Image.open(img_path).convert("RGB")
input_tensor = preprocess(img).unsqueeze(0)
```

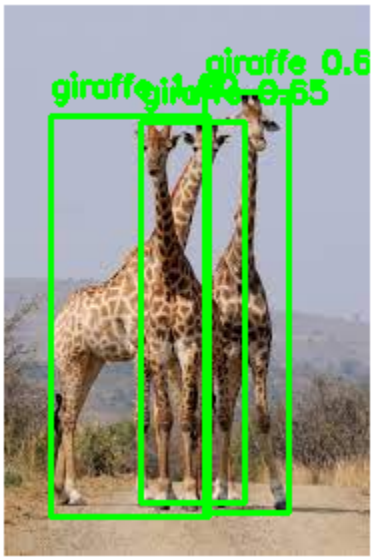
```
with torch.no_grad():
    outputs = model(input_tensor)[0]

img_cv = cv2.imread(img_path)

threshold = 0.5
for box, label, score in zip(outputs['boxes'], outputs['labels'], outputs['scores']):
    if score > threshold:
        x1, y1, x2, y2 = box.int().numpy()
        class_name = labels[label]
        cv2.rectangle(img_cv, (x1, y1), (x2, y2), (0, 255, 0), 2)
        cv2.putText(img_cv, f'{class_name} {score:.2f}', (x1, y1 - 10),
                    cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0, 255, 0), 2)

cv2_imshow(img_cv)
```

OUTPUT:



RESULT:

Thus the program has been successfully implemented and verified.