yq2021_Hw2_Problem5

October 14, 2022

Problem 5

Option 1: Finetuning a pretrained torchvision object detection model

*Some of the code comes from: https://pytorch.org/tutorials/intermediate/torchvision_tutorial.html

[1]:
```
!pip install cython
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-
wheels/public/simple/
Requirement already satisfied: cython in /usr/local/lib/python3.7/dist-packages
(0.29.32)
```

[2]:
```
!pip install pycocotools
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-
wheels/public/simple/
Requirement already satisfied: pycocotools in /usr/local/lib/python3.7/dist-
packages (2.0.5)
Requirement already satisfied: matplotlib>=2.1.0 in
/usr/local/lib/python3.7/dist-packages (from pycocotools) (3.2.2)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages
(from pycocotools) (1.21.6)
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.7/dist-
packages (from matplotlib>=2.1.0->pycocotools) (0.11.0)
Requirement already satisfied: kiwisolver>=1.0.1 in
/usr/local/lib/python3.7/dist-packages (from matplotlib>=2.1.0->pycocotools)
(1.4.4)
Requirement already satisfied: python-dateutil>=2.1 in
/usr/local/lib/python3.7/dist-packages (from matplotlib>=2.1.0->pycocotools)
(2.8.2)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in
/usr/local/lib/python3.7/dist-packages (from matplotlib>=2.1.0->pycocotools)
(3.0.9)
Requirement already satisfied: typing-extensions in
/usr/local/lib/python3.7/dist-packages (from
kiwisolver>=1.0.1->matplotlib>=2.1.0->pycocotools) (4.1.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.7/dist-
packages (from python-dateutil>=2.1->matplotlib>=2.1.0->pycocotools) (1.15.0)
```

Download the dataset and unzip it

[3]:

```
!wget https://www.cis.upenn.edu/~jshi/ped_html/PennFudanPed.zip .
```

```
--2022-10-14 20:04:37--
https://www.cis.upenn.edu/~jshi/ped_html/PennFudanPed.zip
Resolving www.cis.upenn.edu (www.cis.upenn.edu)… 158.130.69.163,
2607:f470:8:64:5ea5::d
Connecting to www.cis.upenn.edu (www.cis.upenn.edu)|158.130.69.163|:443…
connected.
HTTP request sent, awaiting response… 200 OK
Length: 53723336 (51M) [application/zip]
Saving to: ‘PennFudanPed.zip’

PennFudanPed.zip    100%[===================>]  51.23M  10.7MB/s    in 5.9s

2022-10-14 20:04:44 (8.67 MB/s) - ‘PennFudanPed.zip’ saved [53723336/53723336]

--2022-10-14 20:04:44--  http://./
Resolving . (.)… failed: No address associated with hostname.
wget: unable to resolve host address ‘.’
FINISHED --2022-10-14 20:04:44--
Total wall clock time: 7.1s
Downloaded: 1 files, 51M in 5.9s (8.67 MB/s)
```

[4]:

```
!unzip -q PennFudanPed.zip
```

Add the corresponding class for this dataset

[5]:

```
import os
import numpy as np
import torch
from PIL import Image


class PennFudanDataset(torch.utils.data.Dataset):
    def __init__(self, root, transforms):
        self.root = root
        self.transforms = transforms
        # load all image files, sorting them to
        # ensure that they are aligned
        self.imgs = list(sorted(os.listdir(os.path.join(root, "PNGImages"))))
        self.masks = list(sorted(os.listdir(os.path.join(root, "PedMasks"))))

    def __getitem__(self, idx):
        # load images and masks
        img_path = os.path.join(self.root, "PNGImages", self.imgs[idx])
        mask_path = os.path.join(self.root, "PedMasks", self.masks[idx])
        img = Image.open(img_path).convert("RGB")
        # note that we haven't converted the mask to RGB,
        # because each color corresponds to a different instance
        # with 0 being background
        mask = Image.open(mask_path)
        # convert the PIL Image into a numpy array
        mask = np.array(mask)
        # instances are encoded as different colors
        obj_ids = np.unique(mask)
        # first id is the background, so remove it
        obj_ids = obj_ids[1:]

        # split the color-encoded mask into a set
        # of binary masks
        masks = mask == obj_ids[:, None, None]

        # get bounding box coordinates for each mask
        num_objs = len(obj_ids)
        boxes = []
        for i in range(num_objs):
            pos = np.where(masks[i])
            xmin = np.min(pos[1])
            xmax = np.max(pos[1])
            ymin = np.min(pos[0])
            ymax = np.max(pos[0])
            boxes.append([xmin, ymin, xmax, ymax])

        # convert everything into a torch.Tensor
        boxes = torch.as_tensor(boxes, dtype=torch.float32)
        # there is only one class
        labels = torch.ones((num_objs,), dtype=torch.int64)
        masks = torch.as_tensor(masks, dtype=torch.uint8)

        image_id = torch.tensor([idx])
        area = (boxes[:, 3] - boxes[:, 1]) * (boxes[:, 2] - boxes[:, 0])
        # suppose all instances are not crowd
        iscrowd = torch.zeros((num_objs,), dtype=torch.int64)

        target = {}
        target["boxes"] = boxes
        target["labels"] = labels
        target["masks"] = masks
        target["image_id"] = image_id
        target["area"] = area
        target["iscrowd"] = iscrowd

        if self.transforms is not None:
            img, target = self.transforms(img, target)

        return img, target

    def __len__(self):
        return len(self.imgs)
```

Check the downloaded Dataset

[6]:

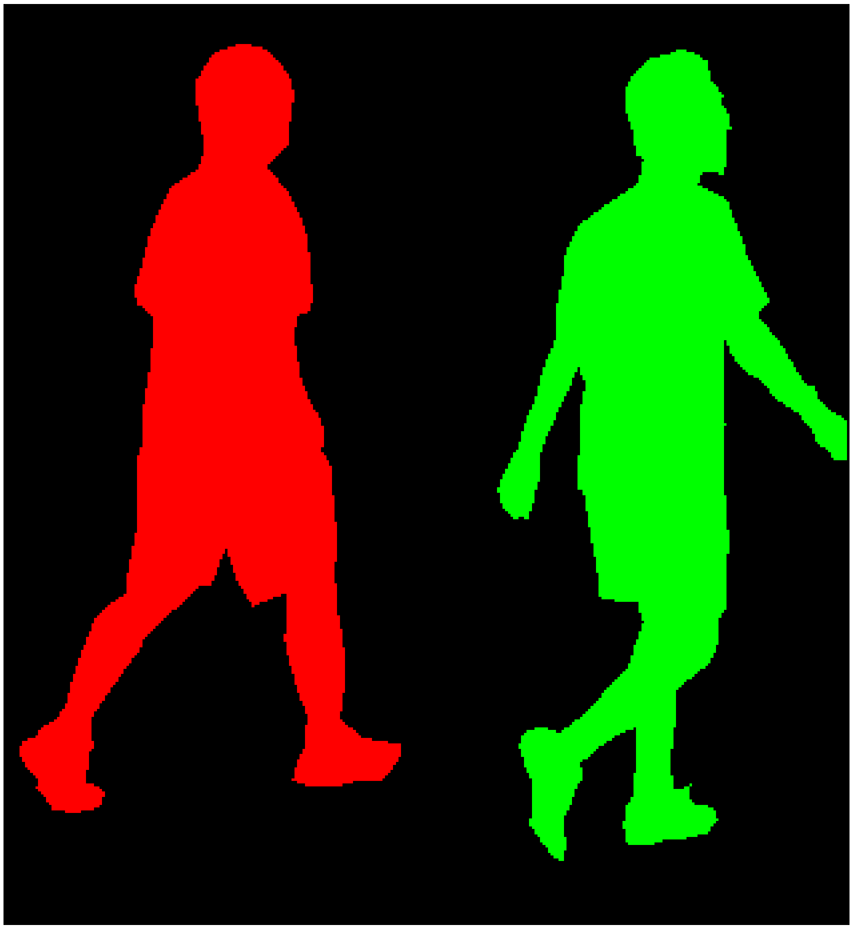
```
Image.open('PennFudanPed/PNGImages/FudanPed00028.png')
```

[6]:

[7]:

```
test_mask = Image.open('PennFudanPed/PedMasks/FudanPed00028_mask.png')
test_mask.putpalette([
    0, 0, 0, # black background
    255, 0, 0, # index 1 is red
    0, 255, 0, # index 2 is green
    0, 0, 255, # index 3 is blue
])
test_mask
```

[7]:



Import supportive functions

[8]:

```
%%shell

git clone https://github.com/pytorch/vision.git
cd vision
git checkout v0.13.0

cp references/detection/utils.py ../
cp references/detection/transforms.py ../
cp references/detection/coco_eval.py ../
cp references/detection/engine.py ../
cp references/detection/coco_utils.py ../
```

```
Cloning into 'vision'…
remote: Enumerating objects: 231713, done.
remote: Counting objects: 100% (5384/5384), done.
remote: Compressing objects: 100% (555/555), done.
remote: Total 231713 (delta 4960), reused 5189 (delta 4820), pack-reused
```

```
226329
Receiving objects: 100% (231713/231713), 467.54 MiB | 16.93 MiB/s, done.
Resolving deltas: 100% (210050/210050), done.
Note: checking out 'v0.13.0'.

You are in 'detached HEAD' state. You can look around, make experimental
changes and commit them, and you can discard any commits you make in this
state without impacting any branches by performing another checkout.

If you want to create a new branch to retain commits you create, you may
do so (now or later) by using -b with the checkout command again. Example:

  git checkout -b <new-branch-name>

HEAD is now at da3794e90 Fix all broken URLs (#6176) (#6177)
```

[8]:

Initialize the dataset

[34]:

```python
import transforms as T
from engine import train_one_epoch, evaluate
import utils

def get_transform(train):
    transforms = []
    transforms.append(T.PILToTensor())
    transforms.append(T.ConvertImageDtype(torch.float))
    if train:
        transforms.append(T.RandomHorizontalFlip(0.5))
    return T.Compose(transforms)

dataset = PennFudanDataset('PennFudanPed/', get_transform(train=True))
dataset[0]
```

[34]:

```
(tensor([[[0.5804, 0.5725, 0.5608,  …, 0.8314, 0.8235, 0.8275],
          [0.5020, 0.5020, 0.4941,  …, 0.6667, 0.6784, 0.7020],
          [0.5098, 0.5137, 0.5098,  …, 0.7373, 0.7608, 0.8000],
          …,
          [0.7255, 0.7216, 0.7176,  …, 0.8275, 0.8510, 0.8863],
          [0.7294, 0.7333, 0.7333,  …, 0.8235, 0.8588, 0.9059],
          [0.7333, 0.7451, 0.7451,  …, 0.8431, 0.8902, 0.8824]],

         [[0.3961, 0.3882, 0.3765,  …, 0.7882, 0.7804, 0.7843],
          [0.3176, 0.3176, 0.3098,  …, 0.6235, 0.6353, 0.6588],
          [0.3255, 0.3294, 0.3255,  …, 0.6941, 0.7176, 0.7569],
          …,
          [0.7255, 0.7216, 0.7176,  …, 0.8039, 0.8275, 0.8627],
          [0.7294, 0.7333, 0.7333,  …, 0.8000, 0.8353, 0.8824],
          [0.7333, 0.7451, 0.7451,  …, 0.8196, 0.8667, 0.8588]],

         [[0.3255, 0.3176, 0.3059,  …, 0.7176, 0.7098, 0.7137],
          [0.2471, 0.2471, 0.2392,  …, 0.5529, 0.5647, 0.5882],
          [0.2549, 0.2588, 0.2549,  …, 0.6235, 0.6471, 0.6863],
          …,
          [0.7255, 0.7216, 0.7176,  …, 0.8039, 0.8275, 0.8627],
          [0.7294, 0.7333, 0.7333,  …, 0.8000, 0.8353, 0.8824],
          [0.7333, 0.7451, 0.7451,  …, 0.8196, 0.8667, 0.8588]]]),
 {'boxes': tensor([[258., 181., 400., 430.],
          [ 25., 170., 140., 485.]]),
  'labels': tensor([1, 1]),
  'masks': tensor([[[0, 0, 0,  …, 0, 0, 0],
           [0, 0, 0,  …, 0, 0, 0],
           [0, 0, 0,  …, 0, 0, 0],
           …,
           [0, 0, 0,  …, 0, 0, 0],
           [0, 0, 0,  …, 0, 0, 0],
           [0, 0, 0,  …, 0, 0, 0]],

          [[0, 0, 0,  …, 0, 0, 0],
           [0, 0, 0,  …, 0, 0, 0],
           [0, 0, 0,  …, 0, 0, 0],
           …,
           [0, 0, 0,  …, 0, 0, 0],
           [0, 0, 0,  …, 0, 0, 0],
           [0, 0, 0,  …, 0, 0, 0]]], dtype=torch.uint8),
  'image_id': tensor([0]),
  'area': tensor([35358., 36225.]),
  'iscrowd': tensor([0, 0])})
```

Using Mask R-CNN to compute the instance segmentation masks

[35]:

```python
import torchvision
from torchvision.models.detection.faster_rcnn import FastRCNNPredictor
from torchvision.models.detection.mask_rcnn import MaskRCNNPredictor


def get_model_instance_segmentation(num_classes):
    # load an instance segmentation model pre-trained on COCO
    model = torchvision.models.detection.
 ↪maskrcnn_resnet50_fpn(weights="DEFAULT")

    # get number of input features for the classifier
    in_features = model.roi_heads.box_predictor.cls_score.in_features
    # replace the pre-trained head with a new one
    model.roi_heads.box_predictor = FastRCNNPredictor(in_features, num_classes)

    # now get the number of input features for the mask classifier
    in_features_mask = model.roi_heads.mask_predictor.conv5_mask.in_channels
    hidden_layer = 256
    # and replace the mask predictor with a new one
    model.roi_heads.mask_predictor = MaskRCNNPredictor(in_features_mask,
                                                       hidden_layer,
                                                       num_classes)

    return model
```

Testing forward() method

[11]:
```
model = torchvision.models.detection.fasterrcnn_resnet50_fpn(weights="DEFAULT")
dataset = PennFudanDataset('PennFudanPed', get_transform(train=True))
data_loader = torch.utils.data.DataLoader(
 dataset, batch_size=2, shuffle=True, num_workers=4,
 collate_fn=utils.collate_fn)
# For Training
images,targets = next(iter(data_loader))
images = list(image for image in images)
targets = [{k: v for k, v in t.items()} for t in targets]
output = model(images,targets)    # Returns losses and detections
# For inference
model.eval()
x = [torch.rand(3, 300, 400), torch.rand(3, 500, 400)]
predictions = model(x)            # Returns predictions
```

```
Downloading:
"https://download.pytorch.org/models/fasterrcnn_resnet50_fpn_coco-258fb6c6.pth"
to /root/.cache/torch/hub/checkpoints/fasterrcnn_resnet50_fpn_coco-258fb6c6.pth
```

```
  0%|          | 0.00/160M [00:00<?, ?B/s]
```

```
/usr/local/lib/python3.7/dist-packages/torch/utils/data/dataloader.py:566:
UserWarning: This DataLoader will create 4 worker processes in total. Our
suggested max number of worker in current system is 2, which is smaller than
what this DataLoader is going to create. Please be aware that excessive worker
creation might get DataLoader running slow or even freeze, lower the worker
number to avoid potential slowness/freeze if necessary.
  cpuset_checked))
```

Putting Everything together

Initialize the dataset loader

[36]:
```
# use our dataset and defined transformations
dataset = PennFudanDataset('PennFudanPed', get_transform(train=True))
dataset_test = PennFudanDataset('PennFudanPed', get_transform(train=False))
```

```
# split the dataset in train and test set
torch.manual_seed(3407)
indices = torch.randperm(len(dataset)).tolist()
dataset = torch.utils.data.Subset(dataset, indices[:-50])
dataset_test = torch.utils.data.Subset(dataset_test, indices[-50:])

# define training and validation data loaders
data_loader = torch.utils.data.DataLoader(
    dataset, batch_size=2, shuffle=True, num_workers=4,
    collate_fn=utils.collate_fn)

data_loader_test = torch.utils.data.DataLoader(
    dataset_test, batch_size=1, shuffle=False, num_workers=4,
    collate_fn=utils.collate_fn)
```

```
/usr/local/lib/python3.7/dist-packages/torch/utils/data/dataloader.py:566:
UserWarning: This DataLoader will create 4 worker processes in total. Our
suggested max number of worker in current system is 2, which is smaller than
what this DataLoader is going to create. Please be aware that excessive worker
creation might get DataLoader running slow or even freeze, lower the worker
number to avoid potential slowness/freeze if necessary.
  cpuset_checked))
```

Initialize the parameters and optimizers

[37]:

```
device = torch.device('cuda') if torch.cuda.is_available() else torch.
 ↪device('cpu')

# our dataset has two classes only - background and person
num_classes = 2

# get the model using our helper function
model = get_model_instance_segmentation(num_classes)
# move model to the right device
model.to(device)

# construct an optimizer
params = [p for p in model.parameters() if p.requires_grad]
optimizer = torch.optim.SGD(params, lr=0.005,
                            momentum=0.9, weight_decay=0.0005)

# and a learning rate scheduler which decreases the learning rate by
# 10x every 3 epochs
lr_scheduler = torch.optim.lr_scheduler.StepLR(optimizer,
                                               step_size=3,
                                               gamma=0.1)
```

Train 10 epochs using fine tuned model

[38]:
```python
# let's train it for 10 epochs
from torch.optim.lr_scheduler import StepLR
num_epochs = 10

for epoch in range(num_epochs):
    # train for one epoch, printing every 10 iterations
    train_one_epoch(model, optimizer, data_loader, device, epoch, print_freq=10)
    # update the learning rate
    lr_scheduler.step()
    # evaluate on the test dataset
    evaluate(model, data_loader_test, device=device)
```

```
Epoch: [0]  [ 0/60]  eta: 0:01:06  lr: 0.000090  loss: 2.9246 (2.9246)
loss_classifier: 0.5471 (0.5471)  loss_box_reg: 0.4399 (0.4399)  loss_mask:
1.9163 (1.9163)  loss_objectness: 0.0184 (0.0184)  loss_rpn_box_reg: 0.0028
(0.0028)  time: 1.1105  data: 0.4197  max mem: 6172
Epoch: [0]  [10/60]  eta: 0:00:29  lr: 0.000936  loss: 1.6280 (1.9091)
loss_classifier: 0.4320 (0.4095)  loss_box_reg: 0.3044 (0.3298)  loss_mask:
0.7351 (1.1391)  loss_objectness: 0.0242 (0.0241)  loss_rpn_box_reg: 0.0070
(0.0066)  time: 0.5899  data: 0.0464  max mem: 6172
Epoch: [0]  [20/60]  eta: 0:00:22  lr: 0.001783  loss: 0.9292 (1.3253)
loss_classifier: 0.2026 (0.2964)  loss_box_reg: 0.2471 (0.2780)  loss_mask:
0.3726 (0.7260)  loss_objectness: 0.0144 (0.0189)  loss_rpn_box_reg: 0.0060
(0.0061)  time: 0.5253  data: 0.0094  max mem: 6172
Epoch: [0]  [30/60]  eta: 0:00:16  lr: 0.002629  loss: 0.5777 (1.0515)
loss_classifier: 0.1136 (0.2269)  loss_box_reg: 0.1805 (0.2499)  loss_mask:
0.2140 (0.5528)  loss_objectness: 0.0112 (0.0162)  loss_rpn_box_reg: 0.0049
(0.0057)  time: 0.5036  data: 0.0091  max mem: 6172
Epoch: [0]  [40/60]  eta: 0:00:10  lr: 0.003476  loss: 0.4099 (0.8950)
loss_classifier: 0.0613 (0.1856)  loss_box_reg: 0.1561 (0.2310)  loss_mask:
0.1589 (0.4589)  loss_objectness: 0.0059 (0.0137)  loss_rpn_box_reg: 0.0040
(0.0058)  time: 0.5199  data: 0.0097  max mem: 6172
Epoch: [0]  [50/60]  eta: 0:00:05  lr: 0.004323  loss: 0.3021 (0.7904)
loss_classifier: 0.0498 (0.1591)  loss_box_reg: 0.1269 (0.2125)  loss_mask:
0.1438 (0.4007)  loss_objectness: 0.0037 (0.0119)  loss_rpn_box_reg: 0.0040
(0.0061)  time: 0.5472  data: 0.0096  max mem: 6172
Epoch: [0]  [59/60]  eta: 0:00:00  lr: 0.005000  loss: 0.3007 (0.7154)
loss_classifier: 0.0401 (0.1407)  loss_box_reg: 0.0900 (0.1948)  loss_mask:
0.1379 (0.3635)  loss_objectness: 0.0012 (0.0103)  loss_rpn_box_reg: 0.0038
(0.0060)  time: 0.5421  data: 0.0083  max mem: 6172
Epoch: [0] Total time: 0:00:32 (0.5440 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:25  model_time: 0.1559 (0.1559)  evaluator_time:
0.0047 (0.0047)  time: 0.5019  data: 0.3401  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0908 (0.1024)  evaluator_time:
```

```
0.0054 (0.0076)  time: 0.1124  data: 0.0045  max mem: 6172
Test: Total time: 0:00:06 (0.1262 s / it)
Averaged stats: model_time: 0.0908 (0.1024)  evaluator_time: 0.0054 (0.0076)
Accumulating evaluation results…
DONE (t=0.01s).
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.732
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.987
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.924
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.659
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.741
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.312
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.774
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.774
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.725
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.779
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.732
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.990
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.927
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.401
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.741
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.302
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.771
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.771
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.750
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.773
Epoch: [1]  [ 0/60]  eta: 0:01:06  lr: 0.005000  loss: 0.2624 (0.2624)
loss_classifier: 0.0314 (0.0314)  loss_box_reg: 0.0726 (0.0726)  loss_mask:
0.1521 (0.1521)  loss_objectness: 0.0007 (0.0007)  loss_rpn_box_reg: 0.0056
(0.0056)  time: 1.1027  data: 0.4638  max mem: 6172
Epoch: [1]  [10/60]  eta: 0:00:30  lr: 0.005000  loss: 0.2346 (0.2546)
loss_classifier: 0.0314 (0.0329)  loss_box_reg: 0.0722 (0.0830)  loss_mask:
0.1407 (0.1327)  loss_objectness: 0.0007 (0.0011)  loss_rpn_box_reg: 0.0035
(0.0049)  time: 0.6024  data: 0.0496  max mem: 6172
Epoch: [1]  [20/60]  eta: 0:00:22  lr: 0.005000  loss: 0.2323 (0.2472)
loss_classifier: 0.0291 (0.0319)  loss_box_reg: 0.0686 (0.0736)  loss_mask:
0.1326 (0.1352)  loss_objectness: 0.0009 (0.0017)  loss_rpn_box_reg: 0.0038
(0.0050)  time: 0.5373  data: 0.0083  max mem: 6172
Epoch: [1]  [30/60]  eta: 0:00:16  lr: 0.005000  loss: 0.2316 (0.2590)
loss_classifier: 0.0291 (0.0351)  loss_box_reg: 0.0612 (0.0735)  loss_mask:
0.1423 (0.1437)  loss_objectness: 0.0012 (0.0017)  loss_rpn_box_reg: 0.0039
```

```
(0.0050)  time: 0.5443  data: 0.0085  max mem: 6172
Epoch: [1]  [40/60]  eta: 0:00:11  lr: 0.005000  loss: 0.2955 (0.2642)
loss_classifier: 0.0349 (0.0364)  loss_box_reg: 0.0645 (0.0747)  loss_mask:
0.1541 (0.1464)  loss_objectness: 0.0011 (0.0018)  loss_rpn_box_reg: 0.0034
(0.0049)  time: 0.5501  data: 0.0089  max mem: 6172
Epoch: [1]  [50/60]  eta: 0:00:05  lr: 0.005000  loss: 0.2238 (0.2547)
loss_classifier: 0.0312 (0.0346)  loss_box_reg: 0.0597 (0.0714)  loss_mask:
0.1249 (0.1423)  loss_objectness: 0.0008 (0.0016)  loss_rpn_box_reg: 0.0034
(0.0048)  time: 0.5248  data: 0.0093  max mem: 6172
Epoch: [1]  [59/60]  eta: 0:00:00  lr: 0.005000  loss: 0.2432 (0.2613)
loss_classifier: 0.0368 (0.0360)  loss_box_reg: 0.0634 (0.0745)  loss_mask:
0.1249 (0.1441)  loss_objectness: 0.0004 (0.0016)  loss_rpn_box_reg: 0.0038
(0.0051)  time: 0.5392  data: 0.0093  max mem: 6172
Epoch: [1] Total time: 0:00:33 (0.5559 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:24  model_time: 0.1455 (0.1455)  evaluator_time:
0.0049 (0.0049)  time: 0.4891  data: 0.3375  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0896 (0.1025)  evaluator_time:
0.0048 (0.0064)  time: 0.1104  data: 0.0044  max mem: 6172
Test: Total time: 0:00:06 (0.1251 s / it)
Averaged stats: model_time: 0.0896 (0.1025)  evaluator_time: 0.0048 (0.0064)
Accumulating evaluation results…
DONE (t=0.01s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.759
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.995
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.926
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.715
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.765
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.319
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.798
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.798
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.750
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.803
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.752
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.995
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.919
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.601
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.757
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.315
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.785
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.785
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.775
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.786
Epoch: [2]  [ 0/60]  eta: 0:01:05  lr: 0.005000  loss: 0.2863 (0.2863)
loss_classifier: 0.0435 (0.0435)  loss_box_reg: 0.0752 (0.0752)  loss_mask:
0.1568 (0.1568)  loss_objectness: 0.0024 (0.0024)  loss_rpn_box_reg: 0.0084
(0.0084)  time: 1.0846  data: 0.4351  max mem: 6172
Epoch: [2]  [10/60]  eta: 0:00:28  lr: 0.005000  loss: 0.1999 (0.2222)
loss_classifier: 0.0240 (0.0306)  loss_box_reg: 0.0514 (0.0565)  loss_mask:
0.1246 (0.1297)  loss_objectness: 0.0007 (0.0016)  loss_rpn_box_reg: 0.0037
(0.0039)  time: 0.5775  data: 0.0464  max mem: 6172
Epoch: [2]  [20/60]  eta: 0:00:22  lr: 0.005000  loss: 0.2067 (0.2410)
loss_classifier: 0.0290 (0.0347)  loss_box_reg: 0.0558 (0.0692)  loss_mask:
0.1211 (0.1313)  loss_objectness: 0.0007 (0.0014)  loss_rpn_box_reg: 0.0031
(0.0044)  time: 0.5482  data: 0.0088  max mem: 6172
Epoch: [2]  [30/60]  eta: 0:00:17  lr: 0.005000  loss: 0.2004 (0.2266)
loss_classifier: 0.0290 (0.0320)  loss_box_reg: 0.0496 (0.0622)  loss_mask:
0.1118 (0.1271)  loss_objectness: 0.0006 (0.0013)  loss_rpn_box_reg: 0.0033
(0.0041)  time: 0.5614  data: 0.0092  max mem: 6172
Epoch: [2]  [40/60]  eta: 0:00:11  lr: 0.005000  loss: 0.1869 (0.2231)
loss_classifier: 0.0244 (0.0310)  loss_box_reg: 0.0474 (0.0609)  loss_mask:
0.1169 (0.1262)  loss_objectness: 0.0006 (0.0013)  loss_rpn_box_reg: 0.0028
(0.0038)  time: 0.5329  data: 0.0088  max mem: 6172
Epoch: [2]  [50/60]  eta: 0:00:05  lr: 0.005000  loss: 0.2214 (0.2263)
loss_classifier: 0.0291 (0.0316)  loss_box_reg: 0.0523 (0.0619)  loss_mask:
0.1269 (0.1277)  loss_objectness: 0.0003 (0.0014)  loss_rpn_box_reg: 0.0025
(0.0037)  time: 0.5538  data: 0.0095  max mem: 6172
Epoch: [2]  [59/60]  eta: 0:00:00  lr: 0.005000  loss: 0.2058 (0.2215)
loss_classifier: 0.0298 (0.0302)  loss_box_reg: 0.0473 (0.0595)  loss_mask:
0.1129 (0.1266)  loss_objectness: 0.0004 (0.0013)  loss_rpn_box_reg: 0.0029
(0.0040)  time: 0.5570  data: 0.0090  max mem: 6172
Epoch: [2] Total time: 0:00:33 (0.5589 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:24  model_time: 0.1596 (0.1596)  evaluator_time:
0.0048 (0.0048)  time: 0.4975  data: 0.3318  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0908 (0.1024)  evaluator_time:
0.0048 (0.0056)  time: 0.1097  data: 0.0044  max mem: 6172
Test: Total time: 0:00:06 (0.1243 s / it)
Averaged stats: model_time: 0.0908 (0.1024)  evaluator_time: 0.0048 (0.0056)
Accumulating evaluation results…
DONE (t=0.01s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.809
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.991
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.946
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.693
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.821
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.345
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.845
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.845
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.750
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.854
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.774
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.983
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.939
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.571
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.782
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.325
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.802
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.802
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.767
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.805
Epoch: [3]  [ 0/60]  eta: 0:00:52  lr: 0.000500  loss: 0.1661 (0.1661)
loss_classifier: 0.0232 (0.0232)  loss_box_reg: 0.0222 (0.0222)  loss_mask:
0.1182 (0.1182)  loss_objectness: 0.0016 (0.0016)  loss_rpn_box_reg: 0.0009
(0.0009)  time: 0.8757  data: 0.3491  max mem: 6172
Epoch: [3]  [10/60]  eta: 0:00:29  lr: 0.000500  loss: 0.1787 (0.1937)
loss_classifier: 0.0253 (0.0262)  loss_box_reg: 0.0380 (0.0444)  loss_mask:
0.1163 (0.1194)  loss_objectness: 0.0003 (0.0006)  loss_rpn_box_reg: 0.0020
(0.0031)  time: 0.5821  data: 0.0396  max mem: 6172
Epoch: [3]  [20/60]  eta: 0:00:22  lr: 0.000500  loss: 0.1787 (0.1896)
loss_classifier: 0.0243 (0.0246)  loss_box_reg: 0.0370 (0.0405)  loss_mask:
0.1129 (0.1205)  loss_objectness: 0.0003 (0.0009)  loss_rpn_box_reg: 0.0018
(0.0031)  time: 0.5398  data: 0.0089  max mem: 6172
Epoch: [3]  [30/60]  eta: 0:00:16  lr: 0.000500  loss: 0.1646 (0.1820)
loss_classifier: 0.0214 (0.0242)  loss_box_reg: 0.0307 (0.0376)  loss_mask:
0.1115 (0.1162)  loss_objectness: 0.0004 (0.0010)  loss_rpn_box_reg: 0.0017
(0.0029)  time: 0.5303  data: 0.0088  max mem: 6172
Epoch: [3]  [40/60]  eta: 0:00:11  lr: 0.000500  loss: 0.1763 (0.1848)
loss_classifier: 0.0228 (0.0251)  loss_box_reg: 0.0346 (0.0388)  loss_mask:
0.1107 (0.1166)  loss_objectness: 0.0004 (0.0010)  loss_rpn_box_reg: 0.0023
(0.0032)  time: 0.5671  data: 0.0089  max mem: 6172
Epoch: [3]  [50/60]  eta: 0:00:05  lr: 0.000500  loss: 0.1824 (0.1883)
loss_classifier: 0.0236 (0.0264)  loss_box_reg: 0.0352 (0.0402)  loss_mask:
0.1127 (0.1177)  loss_objectness: 0.0002 (0.0009)  loss_rpn_box_reg: 0.0023
(0.0032)  time: 0.5709  data: 0.0091  max mem: 6172
Epoch: [3]  [59/60]  eta: 0:00:00  lr: 0.000500  loss: 0.1839 (0.1872)
```

```
loss_classifier: 0.0247 (0.0261)  loss_box_reg: 0.0382 (0.0399)  loss_mask:
0.1214 (0.1174)  loss_objectness: 0.0003 (0.0009)  loss_rpn_box_reg: 0.0016
(0.0030)  time: 0.5565  data: 0.0085  max mem: 6172
Epoch: [3] Total time: 0:00:33 (0.5621 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:25  model_time: 0.1518 (0.1518)  evaluator_time:
0.0048 (0.0048)  time: 0.5074  data: 0.3496  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0896 (0.1026)  evaluator_time:
0.0050 (0.0055)  time: 0.1097  data: 0.0043  max mem: 6172
Test: Total time: 0:00:06 (0.1242 s / it)
Averaged stats: model_time: 0.0896 (0.1026)  evaluator_time: 0.0050 (0.0055)
Accumulating evaluation results…
DONE (t=0.01s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.826
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.990
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.943
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.716
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.838
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.351
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.862
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.862
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.783
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.869
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.772
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.983
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.939
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.529
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.781
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.322
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.803
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.803
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.775
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.806
Epoch: [4]  [ 0/60]  eta: 0:00:57  lr: 0.000500  loss: 0.1606 (0.1606)
loss_classifier: 0.0258 (0.0258)  loss_box_reg: 0.0292 (0.0292)  loss_mask:
0.1013 (0.1013)  loss_objectness: 0.0017 (0.0017)  loss_rpn_box_reg: 0.0025
(0.0025)  time: 0.9666  data: 0.3818  max mem: 6172
Epoch: [4]  [10/60]  eta: 0:00:30  lr: 0.000500  loss: 0.1779 (0.1839)
loss_classifier: 0.0258 (0.0271)  loss_box_reg: 0.0298 (0.0369)  loss_mask:
```

```
0.1033 (0.1154)  loss_objectness: 0.0006 (0.0012)  loss_rpn_box_reg: 0.0025
(0.0032)  time: 0.6027  data: 0.0425  max mem: 6172
Epoch: [4]  [20/60]  eta: 0:00:23  lr: 0.000500  loss: 0.1670 (0.1870)
loss_classifier: 0.0255 (0.0276)  loss_box_reg: 0.0317 (0.0401)  loss_mask:
0.1057 (0.1152)  loss_objectness: 0.0003 (0.0009)  loss_rpn_box_reg: 0.0025
(0.0033)  time: 0.5679  data: 0.0088  max mem: 6172
Epoch: [4]  [30/60]  eta: 0:00:17  lr: 0.000500  loss: 0.1663 (0.1857)
loss_classifier: 0.0238 (0.0272)  loss_box_reg: 0.0345 (0.0398)  loss_mask:
0.1057 (0.1147)  loss_objectness: 0.0002 (0.0011)  loss_rpn_box_reg: 0.0021
(0.0029)  time: 0.5550  data: 0.0101  max mem: 6172
Epoch: [4]  [40/60]  eta: 0:00:11  lr: 0.000500  loss: 0.1715 (0.1837)
loss_classifier: 0.0224 (0.0260)  loss_box_reg: 0.0359 (0.0387)  loss_mask:
0.1079 (0.1150)  loss_objectness: 0.0003 (0.0010)  loss_rpn_box_reg: 0.0016
(0.0031)  time: 0.5502  data: 0.0098  max mem: 6172
Epoch: [4]  [50/60]  eta: 0:00:05  lr: 0.000500  loss: 0.1681 (0.1843)
loss_classifier: 0.0224 (0.0260)  loss_box_reg: 0.0309 (0.0391)  loss_mask:
0.1064 (0.1153)  loss_objectness: 0.0003 (0.0009)  loss_rpn_box_reg: 0.0019
(0.0029)  time: 0.5622  data: 0.0088  max mem: 6172
Epoch: [4]  [59/60]  eta: 0:00:00  lr: 0.000500  loss: 0.1510 (0.1812)
loss_classifier: 0.0204 (0.0258)  loss_box_reg: 0.0306 (0.0382)  loss_mask:
0.1002 (0.1136)  loss_objectness: 0.0002 (0.0009)  loss_rpn_box_reg: 0.0017
(0.0028)  time: 0.5623  data: 0.0087  max mem: 6172
Epoch: [4] Total time: 0:00:34 (0.5683 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:25  model_time: 0.1464 (0.1464)  evaluator_time:
0.0050 (0.0050)  time: 0.5019  data: 0.3493  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0900 (0.1035)  evaluator_time:
0.0045 (0.0056)  time: 0.1110  data: 0.0045  max mem: 6172
Test: Total time: 0:00:06 (0.1254 s / it)
Averaged stats: model_time: 0.0900 (0.1035)  evaluator_time: 0.0045 (0.0056)
Accumulating evaluation results…
DONE (t=0.02s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.828
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.990
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.942
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.729
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.841
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.350
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.865
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.865
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.792
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.872
```

```
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.775
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.982
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.939
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.524
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.782
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.324
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.805
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.805
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.783
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.808
Epoch: [5]  [ 0/60]  eta: 0:01:01  lr: 0.000500  loss: 0.2103 (0.2103)
loss_classifier: 0.0309 (0.0309)  loss_box_reg: 0.0322 (0.0322)  loss_mask:
0.1454 (0.1454)  loss_objectness: 0.0006 (0.0006)  loss_rpn_box_reg: 0.0011
(0.0011)  time: 1.0329  data: 0.4769  max mem: 6172
Epoch: [5]  [10/60]  eta: 0:00:30  lr: 0.000500  loss: 0.1709 (0.1848)
loss_classifier: 0.0228 (0.0266)  loss_box_reg: 0.0322 (0.0386)  loss_mask:
0.1075 (0.1164)  loss_objectness: 0.0002 (0.0003)  loss_rpn_box_reg: 0.0031
(0.0028)  time: 0.6062  data: 0.0528  max mem: 6172
Epoch: [5]  [20/60]  eta: 0:00:23  lr: 0.000500  loss: 0.1709 (0.1827)
loss_classifier: 0.0227 (0.0263)  loss_box_reg: 0.0290 (0.0365)  loss_mask:
0.1075 (0.1166)  loss_objectness: 0.0002 (0.0006)  loss_rpn_box_reg: 0.0021
(0.0027)  time: 0.5621  data: 0.0101  max mem: 6172
Epoch: [5]  [30/60]  eta: 0:00:17  lr: 0.000500  loss: 0.1601 (0.1798)
loss_classifier: 0.0163 (0.0260)  loss_box_reg: 0.0286 (0.0356)  loss_mask:
0.1075 (0.1149)  loss_objectness: 0.0003 (0.0007)  loss_rpn_box_reg: 0.0018
(0.0026)  time: 0.5591  data: 0.0094  max mem: 6172
Epoch: [5]  [40/60]  eta: 0:00:11  lr: 0.000500  loss: 0.1510 (0.1769)
loss_classifier: 0.0177 (0.0253)  loss_box_reg: 0.0233 (0.0345)  loss_mask:
0.1043 (0.1136)  loss_objectness: 0.0002 (0.0008)  loss_rpn_box_reg: 0.0018
(0.0027)  time: 0.5461  data: 0.0090  max mem: 6172
Epoch: [5]  [50/60]  eta: 0:00:05  lr: 0.000500  loss: 0.1621 (0.1816)
loss_classifier: 0.0268 (0.0264)  loss_box_reg: 0.0290 (0.0368)  loss_mask:
0.1043 (0.1145)  loss_objectness: 0.0002 (0.0008)  loss_rpn_box_reg: 0.0023
(0.0030)  time: 0.5581  data: 0.0091  max mem: 6172
Epoch: [5]  [59/60]  eta: 0:00:00  lr: 0.000500  loss: 0.1751 (0.1785)
loss_classifier: 0.0267 (0.0255)  loss_box_reg: 0.0371 (0.0359)  loss_mask:
0.1065 (0.1136)  loss_objectness: 0.0002 (0.0007)  loss_rpn_box_reg: 0.0023
(0.0029)  time: 0.5816  data: 0.0088  max mem: 6172
Epoch: [5] Total time: 0:00:34 (0.5735 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:24  model_time: 0.1676 (0.1676)  evaluator_time:
0.0041 (0.0041)  time: 0.4806  data: 0.3075  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0908 (0.1044)  evaluator_time:
0.0039 (0.0054)  time: 0.1117  data: 0.0048  max mem: 6172
```

```
Test: Total time: 0:00:06 (0.1256 s / it)
Averaged stats: model_time: 0.0908 (0.1044)  evaluator_time: 0.0039 (0.0054)
Accumulating evaluation results…
DONE (t=0.01s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.831
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.990
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.938
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.738
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.841
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.350
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.863
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.863
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.800
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.869
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.774
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.982
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.919
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.513
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.783
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.324
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.803
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.803
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.775
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.806
Epoch: [6]  [ 0/60]  eta: 0:00:56  lr: 0.000050  loss: 0.1186 (0.1186)
loss_classifier: 0.0127 (0.0127)  loss_box_reg: 0.0150 (0.0150)  loss_mask:
0.0894 (0.0894)  loss_objectness: 0.0002 (0.0002)  loss_rpn_box_reg: 0.0013
(0.0013)  time: 0.9372  data: 0.3950  max mem: 6172
Epoch: [6]  [10/60]  eta: 0:00:29  lr: 0.000050  loss: 0.1873 (0.1908)
loss_classifier: 0.0281 (0.0287)  loss_box_reg: 0.0329 (0.0383)  loss_mask:
0.1094 (0.1185)  loss_objectness: 0.0010 (0.0021)  loss_rpn_box_reg: 0.0030
(0.0032)  time: 0.5878  data: 0.0437  max mem: 6172
Epoch: [6]  [20/60]  eta: 0:00:22  lr: 0.000050  loss: 0.1779 (0.1852)
loss_classifier: 0.0240 (0.0269)  loss_box_reg: 0.0329 (0.0353)  loss_mask:
0.1094 (0.1183)  loss_objectness: 0.0003 (0.0018)  loss_rpn_box_reg: 0.0023
(0.0028)  time: 0.5484  data: 0.0089  max mem: 6172
Epoch: [6]  [30/60]  eta: 0:00:17  lr: 0.000050  loss: 0.1698 (0.1877)
loss_classifier: 0.0222 (0.0269)  loss_box_reg: 0.0353 (0.0379)  loss_mask:
0.1087 (0.1186)  loss_objectness: 0.0002 (0.0014)  loss_rpn_box_reg: 0.0018
(0.0029)  time: 0.5784  data: 0.0091  max mem: 6172
```

```
Epoch: [6]  [40/60]  eta: 0:00:11  lr: 0.000050  loss: 0.1514 (0.1779)
loss_classifier: 0.0188 (0.0248)  loss_box_reg: 0.0255 (0.0339)  loss_mask:
0.1048 (0.1155)  loss_objectness: 0.0003 (0.0012)  loss_rpn_box_reg: 0.0016
(0.0026)  time: 0.5715  data: 0.0086  max mem: 6172
Epoch: [6]  [50/60]  eta: 0:00:05  lr: 0.000050  loss: 0.1514 (0.1776)
loss_classifier: 0.0188 (0.0246)  loss_box_reg: 0.0261 (0.0345)  loss_mask:
0.1033 (0.1145)  loss_objectness: 0.0003 (0.0012)  loss_rpn_box_reg: 0.0021
(0.0027)  time: 0.5737  data: 0.0084  max mem: 6172
Epoch: [6]  [59/60]  eta: 0:00:00  lr: 0.000050  loss: 0.1619 (0.1754)
loss_classifier: 0.0217 (0.0243)  loss_box_reg: 0.0296 (0.0341)  loss_mask:
0.1030 (0.1133)  loss_objectness: 0.0002 (0.0010)  loss_rpn_box_reg: 0.0026
(0.0027)  time: 0.5817  data: 0.0083  max mem: 6172
Epoch: [6] Total time: 0:00:34 (0.5782 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:24  model_time: 0.1573 (0.1573)  evaluator_time:
0.0041 (0.0041)  time: 0.4806  data: 0.3180  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0929 (0.1038)  evaluator_time:
0.0041 (0.0055)  time: 0.1111  data: 0.0041  max mem: 6172
Test: Total time: 0:00:06 (0.1253 s / it)
Averaged stats: model_time: 0.0929 (0.1038)  evaluator_time: 0.0041 (0.0055)
Accumulating evaluation results…
DONE (t=0.01s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.833
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.991
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.931
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.735
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.843
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.352
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.865
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.865
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.800
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.872
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.779
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.982
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.920
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.518
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.789
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.327
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.807
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.807
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.792
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.808
Epoch: [7]  [ 0/60]  eta: 0:00:53  lr: 0.000050  loss: 0.1325 (0.1325)
loss_classifier: 0.0098 (0.0098)  loss_box_reg: 0.0160 (0.0160)  loss_mask:
0.1062 (0.1062)  loss_objectness: 0.0001 (0.0001)  loss_rpn_box_reg: 0.0005
(0.0005)  time: 0.8950  data: 0.3721  max mem: 6172
Epoch: [7]  [10/60]  eta: 0:00:29  lr: 0.000050  loss: 0.1486 (0.1647)
loss_classifier: 0.0231 (0.0219)  loss_box_reg: 0.0298 (0.0309)  loss_mask:
0.1062 (0.1096)  loss_objectness: 0.0002 (0.0004)  loss_rpn_box_reg: 0.0015
(0.0019)  time: 0.5890  data: 0.0418  max mem: 6172
Epoch: [7]  [20/60]  eta: 0:00:23  lr: 0.000050  loss: 0.1624 (0.1701)
loss_classifier: 0.0243 (0.0229)  loss_box_reg: 0.0340 (0.0325)  loss_mask:
0.1082 (0.1116)  loss_objectness: 0.0002 (0.0004)  loss_rpn_box_reg: 0.0023
(0.0026)  time: 0.5605  data: 0.0090  max mem: 6172
Epoch: [7]  [30/60]  eta: 0:00:17  lr: 0.000050  loss: 0.1565 (0.1678)
loss_classifier: 0.0207 (0.0237)  loss_box_reg: 0.0290 (0.0318)  loss_mask:
0.1078 (0.1092)  loss_objectness: 0.0002 (0.0006)  loss_rpn_box_reg: 0.0023
(0.0025)  time: 0.5627  data: 0.0089  max mem: 6172
Epoch: [7]  [40/60]  eta: 0:00:11  lr: 0.000050  loss: 0.1448 (0.1695)
loss_classifier: 0.0163 (0.0235)  loss_box_reg: 0.0218 (0.0328)  loss_mask:
0.1101 (0.1102)  loss_objectness: 0.0003 (0.0005)  loss_rpn_box_reg: 0.0019
(0.0026)  time: 0.5715  data: 0.0087  max mem: 6172
Epoch: [7]  [50/60]  eta: 0:00:05  lr: 0.000050  loss: 0.1841 (0.1749)
loss_classifier: 0.0212 (0.0241)  loss_box_reg: 0.0340 (0.0343)  loss_mask:
0.1113 (0.1129)  loss_objectness: 0.0002 (0.0008)  loss_rpn_box_reg: 0.0027
(0.0027)  time: 0.5844  data: 0.0089  max mem: 6172
Epoch: [7]  [59/60]  eta: 0:00:00  lr: 0.000050  loss: 0.1798 (0.1758)
loss_classifier: 0.0232 (0.0243)  loss_box_reg: 0.0243 (0.0343)  loss_mask:
0.1113 (0.1136)  loss_objectness: 0.0003 (0.0008)  loss_rpn_box_reg: 0.0024
(0.0027)  time: 0.5675  data: 0.0085  max mem: 6172
Epoch: [7] Total time: 0:00:34 (0.5769 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:23  model_time: 0.1528 (0.1528)  evaluator_time:
0.0040 (0.0040)  time: 0.4673  data: 0.3091  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0905 (0.1036)  evaluator_time:
0.0039 (0.0054)  time: 0.1117  data: 0.0048  max mem: 6172
Test: Total time: 0:00:06 (0.1249 s / it)
Averaged stats: model_time: 0.0905 (0.1036)  evaluator_time: 0.0039 (0.0054)
Accumulating evaluation results…
DONE (t=0.01s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.831
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.991
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.930
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.737
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.842
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.352
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.866
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.866
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.808
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.872
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.774
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.982
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.929
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.523
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.782
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.325
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.805
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.805
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.792
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.806
Epoch: [8]  [ 0/60]  eta: 0:00:57  lr: 0.000050  loss: 0.1174 (0.1174)
loss_classifier: 0.0051 (0.0051)  loss_box_reg: 0.0128 (0.0128)  loss_mask:
0.0958 (0.0958)  loss_objectness: 0.0002 (0.0002)  loss_rpn_box_reg: 0.0035
(0.0035)  time: 0.9555  data: 0.4094  max mem: 6172
Epoch: [8]  [10/60]  eta: 0:00:28  lr: 0.000050  loss: 0.1461 (0.1516)
loss_classifier: 0.0171 (0.0188)  loss_box_reg: 0.0255 (0.0247)  loss_mask:
0.0995 (0.1058)  loss_objectness: 0.0002 (0.0003)  loss_rpn_box_reg: 0.0020
(0.0020)  time: 0.5746  data: 0.0447  max mem: 6172
Epoch: [8]  [20/60]  eta: 0:00:23  lr: 0.000050  loss: 0.1564 (0.1661)
loss_classifier: 0.0218 (0.0225)  loss_box_reg: 0.0257 (0.0306)  loss_mask:
0.1035 (0.1096)  loss_objectness: 0.0003 (0.0011)  loss_rpn_box_reg: 0.0014
(0.0023)  time: 0.5605  data: 0.0085  max mem: 6172
Epoch: [8]  [30/60]  eta: 0:00:16  lr: 0.000050  loss: 0.1774 (0.1734)
loss_classifier: 0.0242 (0.0239)  loss_box_reg: 0.0310 (0.0341)  loss_mask:
0.1058 (0.1120)  loss_objectness: 0.0005 (0.0010)  loss_rpn_box_reg: 0.0016
(0.0024)  time: 0.5613  data: 0.0106  max mem: 6172
Epoch: [8]  [40/60]  eta: 0:00:11  lr: 0.000050  loss: 0.1696 (0.1753)
loss_classifier: 0.0210 (0.0246)  loss_box_reg: 0.0310 (0.0353)  loss_mask:
0.0984 (0.1120)  loss_objectness: 0.0003 (0.0009)  loss_rpn_box_reg: 0.0022
(0.0026)  time: 0.5528  data: 0.0108  max mem: 6172
Epoch: [8]  [50/60]  eta: 0:00:05  lr: 0.000050  loss: 0.1619 (0.1748)
loss_classifier: 0.0223 (0.0244)  loss_box_reg: 0.0259 (0.0344)  loss_mask:
0.1105 (0.1125)  loss_objectness: 0.0003 (0.0010)  loss_rpn_box_reg: 0.0019
(0.0025)  time: 0.5794  data: 0.0095  max mem: 6172
Epoch: [8]  [59/60]  eta: 0:00:00  lr: 0.000050  loss: 0.1636 (0.1756)
loss_classifier: 0.0237 (0.0246)  loss_box_reg: 0.0290 (0.0349)  loss_mask:
```

```
0.1110 (0.1126)  loss_objectness: 0.0004 (0.0010)  loss_rpn_box_reg: 0.0019
(0.0025)  time: 0.5968  data: 0.0092  max mem: 6172
Epoch: [8] Total time: 0:00:34 (0.5783 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:22  model_time: 0.1608 (0.1608)  evaluator_time:
0.0040 (0.0040)  time: 0.4594  data: 0.2916  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0906 (0.1043)  evaluator_time:
0.0041 (0.0054)  time: 0.1112  data: 0.0044  max mem: 6172
Test: Total time: 0:00:06 (0.1249 s / it)
Averaged stats: model_time: 0.0906 (0.1043)  evaluator_time: 0.0041 (0.0054)
Accumulating evaluation results…
DONE (t=0.01s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.830
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.991
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.930
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.732
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.841
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.352
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.864
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.864
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.800
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.870
IoU metric: segm
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.774
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.982
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.929
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.521
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.782
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.324
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.803
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.803
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.783
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.805
Epoch: [9]  [ 0/60]  eta: 0:01:10  lr: 0.000005  loss: 0.1382 (0.1382)
loss_classifier: 0.0135 (0.0135)  loss_box_reg: 0.0210 (0.0210)  loss_mask:
0.0982 (0.0982)  loss_objectness: 0.0040 (0.0040)  loss_rpn_box_reg: 0.0015
(0.0015)  time: 1.1709  data: 0.4315  max mem: 6172
Epoch: [9]  [10/60]  eta: 0:00:30  lr: 0.000005  loss: 0.1464 (0.1678)
loss_classifier: 0.0203 (0.0226)  loss_box_reg: 0.0240 (0.0317)  loss_mask:
0.1024 (0.1102)  loss_objectness: 0.0005 (0.0008)  loss_rpn_box_reg: 0.0016
```

```
(0.0026)  time: 0.6173  data: 0.0455  max mem: 6172
Epoch: [9]  [20/60]  eta: 0:00:23  lr: 0.000005  loss: 0.1594 (0.1706)
loss_classifier: 0.0211 (0.0238)  loss_box_reg: 0.0240 (0.0302)  loss_mask:
0.1046 (0.1135)  loss_objectness: 0.0001 (0.0008)  loss_rpn_box_reg: 0.0016
(0.0023)  time: 0.5627  data: 0.0079  max mem: 6172
Epoch: [9]  [30/60]  eta: 0:00:17  lr: 0.000005  loss: 0.1813 (0.1764)
loss_classifier: 0.0247 (0.0248)  loss_box_reg: 0.0349 (0.0341)  loss_mask:
0.1204 (0.1142)  loss_objectness: 0.0001 (0.0009)  loss_rpn_box_reg: 0.0016
(0.0024)  time: 0.5751  data: 0.0088  max mem: 6172
Epoch: [9]  [40/60]  eta: 0:00:11  lr: 0.000005  loss: 0.1865 (0.1767)
loss_classifier: 0.0243 (0.0247)  loss_box_reg: 0.0401 (0.0346)  loss_mask:
0.1171 (0.1141)  loss_objectness: 0.0003 (0.0008)  loss_rpn_box_reg: 0.0018
(0.0025)  time: 0.5759  data: 0.0089  max mem: 6172
Epoch: [9]  [50/60]  eta: 0:00:05  lr: 0.000005  loss: 0.1659 (0.1773)
loss_classifier: 0.0234 (0.0250)  loss_box_reg: 0.0276 (0.0347)  loss_mask:
0.1147 (0.1142)  loss_objectness: 0.0003 (0.0009)  loss_rpn_box_reg: 0.0023
(0.0025)  time: 0.5656  data: 0.0091  max mem: 6172
Epoch: [9]  [59/60]  eta: 0:00:00  lr: 0.000005  loss: 0.1567 (0.1738)
loss_classifier: 0.0176 (0.0242)  loss_box_reg: 0.0232 (0.0337)  loss_mask:
0.1008 (0.1127)  loss_objectness: 0.0003 (0.0008)  loss_rpn_box_reg: 0.0020
(0.0025)  time: 0.5521  data: 0.0086  max mem: 6172
Epoch: [9] Total time: 0:00:34 (0.5781 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:25  model_time: 0.1645 (0.1645)  evaluator_time:
0.0046 (0.0046)  time: 0.5018  data: 0.3316  max mem: 6172
Test:  [49/50]  eta: 0:00:00  model_time: 0.0906 (0.1047)  evaluator_time:
0.0041 (0.0055)  time: 0.1121  data: 0.0045  max mem: 6172
Test: Total time: 0:00:06 (0.1264 s / it)
Averaged stats: model_time: 0.0906 (0.1047)  evaluator_time: 0.0041 (0.0055)
Accumulating evaluation results…
DONE (t=0.02s).
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.830
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.991
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.930
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.732
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.841
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.352
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.865
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.865
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.800
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.871
IoU metric: segm
```

```
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.772
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.982
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.929
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.521
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.779
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.323
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.801
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.801
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.783
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.803
```

Check the output of the trained model

[41]:

```python
# pick random one image from the test set
import random
index = random.randint(0, 50)
img, _ = dataset_test[index]
# put the model in evaluation mode
model.eval()
with torch.no_grad():
    prediction = model([img.to(device)])
```

[42]:

```python
# check the predicted result
prediction
```

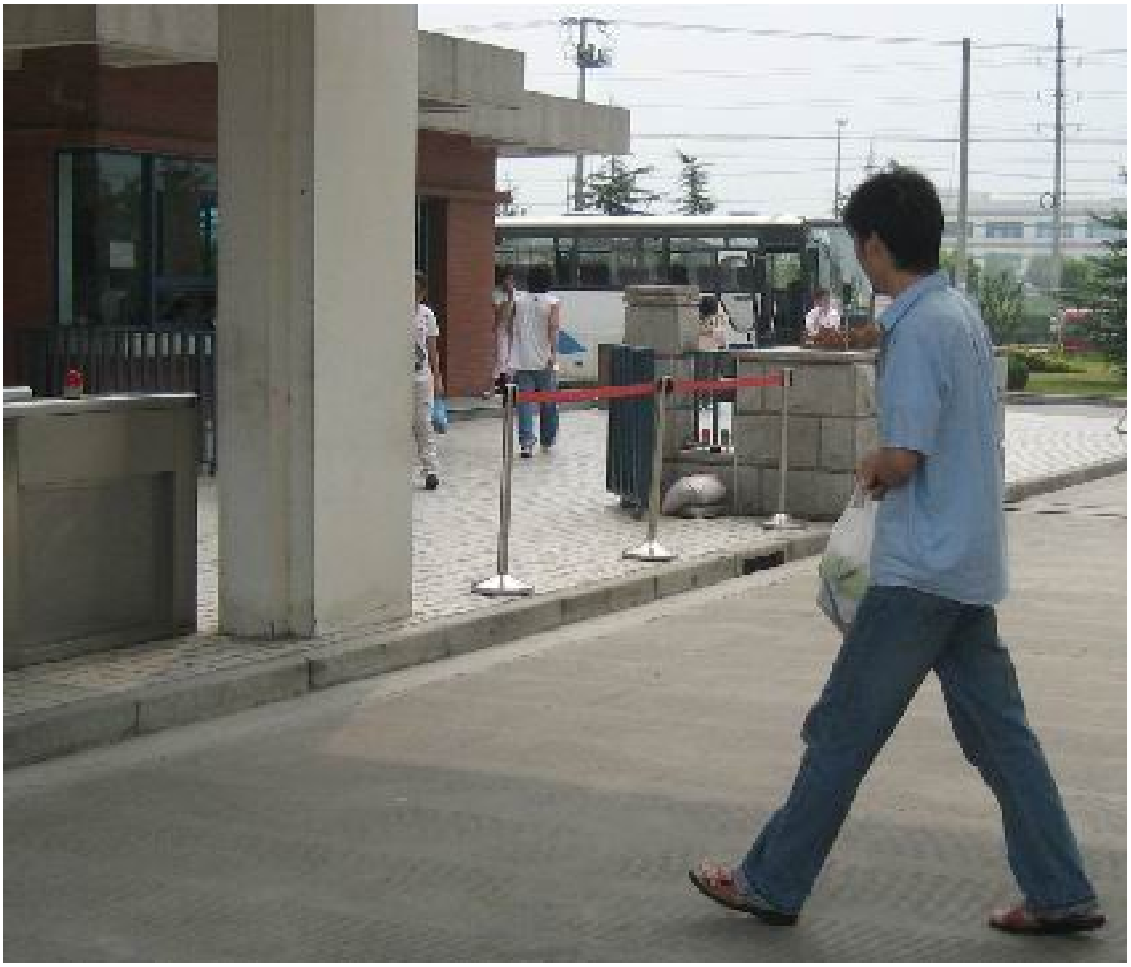
[42]:

```
[{'boxes': tensor([[257.1166,  64.9268, 413.6607, 351.0188],
          [189.0505,  96.6888, 210.3089, 171.2245],
          [303.5939,  60.2320, 388.4765, 314.0001]], device='cuda:0'),
  'labels': tensor([1, 1, 1], device='cuda:0'),
  'scores': tensor([0.9975, 0.9773, 0.2118], device='cuda:0'),
  'masks': tensor([[[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]],


          [[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]],
          [[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]]], device='cuda:0')}]
```

Reformat the image and check the result

```
[43]: Image.fromarray(img.mul(255).permute(1, 2, 0).byte().numpy())
```

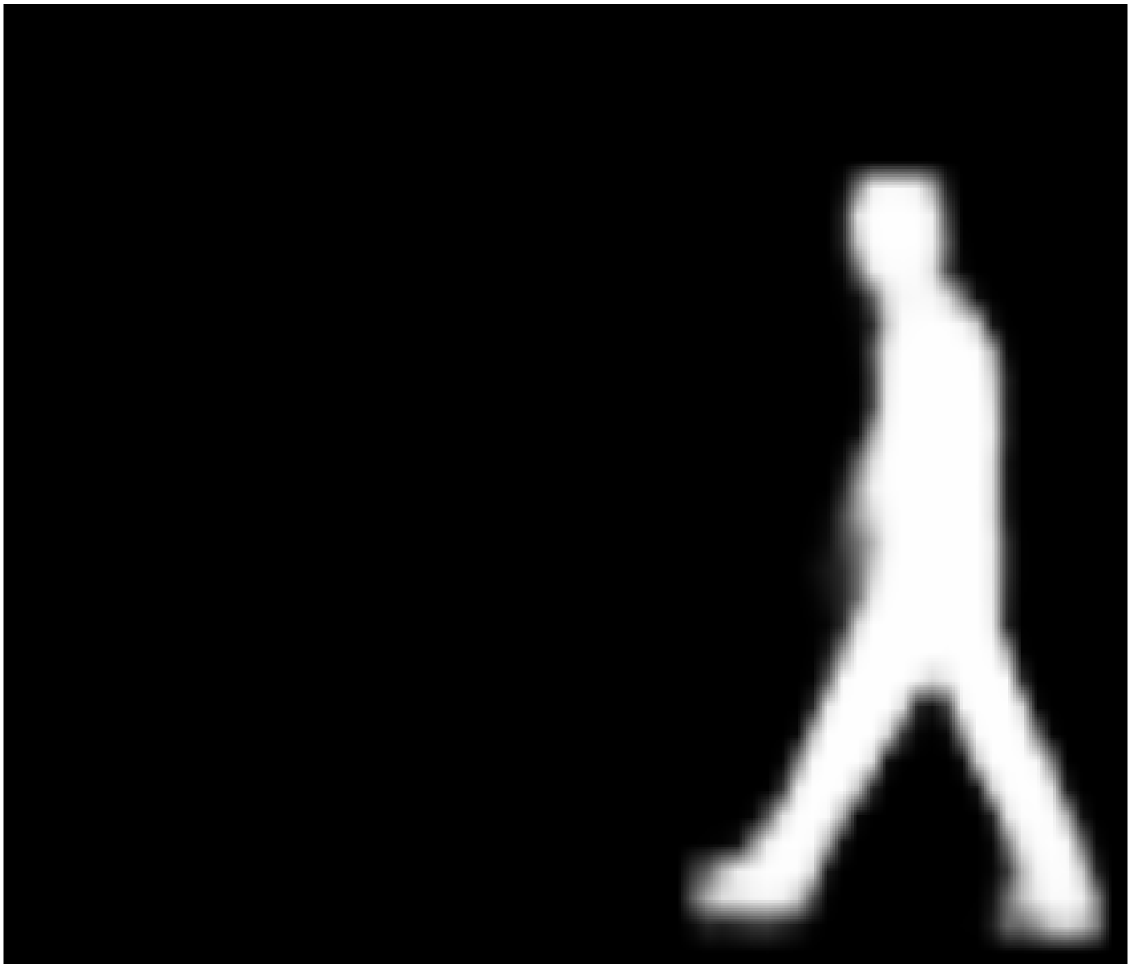
[43]:



Check the segmentation mask

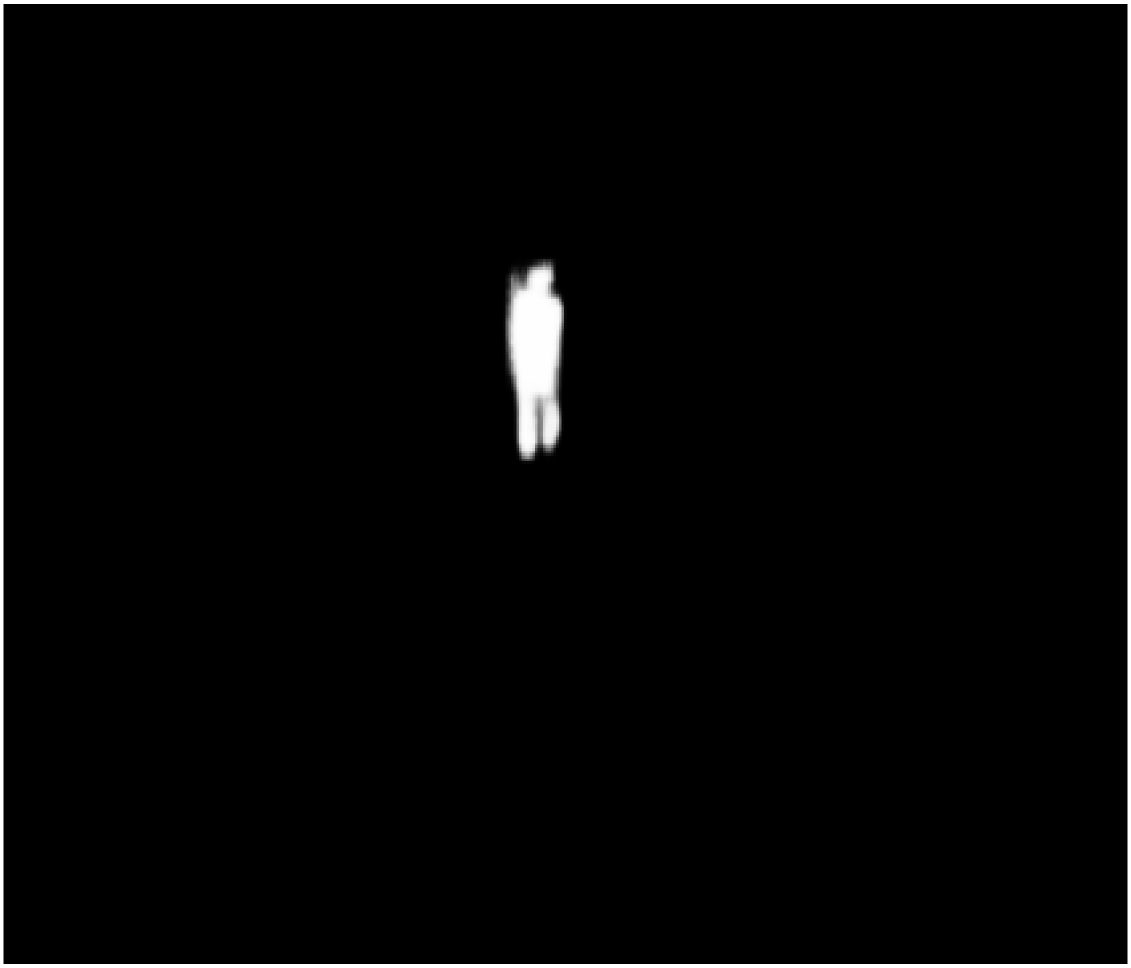
```
[44]: Image.fromarray(prediction[0]['masks'][0, 0].mul(255).byte().cpu().numpy())
```

[44]:



[45]:
```
Image.fromarray(prediction[0]['masks'][1, 0].mul(255).byte().cpu().numpy())
```

[45]:



Option 2: Add another backbone

[46]:

```
import torchvision
from torchvision.models.detection import FasterRCNN
from torchvision.models.detection.rpn import AnchorGenerator

# load a pre-trained model for classification and return
# only the features
backbone = torchvision.models.mobilenet_v2(weights="DEFAULT").features
# FasterRCNN needs to know the number of
# output channels in a backbone. For mobilenet_v2, it's 1280
# so we need to add it here
backbone.out_channels = 1280

# let's make the RPN generate 5 x 3 anchors per spatial
# location, with 5 different sizes and 3 different aspect
# ratios. We have a Tuple[Tuple[int]] because each feature
# map could potentially have different sizes and
# aspect ratios
anchor_generator = AnchorGenerator(sizes=((32, 64, 128, 256, 512),),
                                   aspect_ratios=((0.5, 1.0, 2.0),))

# let's define what are the feature maps that we will
# use to perform the region of interest cropping, as well as
# the size of the crop after rescaling.
# if your backbone returns a Tensor, featmap_names is expected to
# be [0]. More generally, the backbone should return an
# OrderedDict[Tensor], and in featmap_names you can choose which
# feature maps to use.
roi_pooler = torchvision.ops.MultiScaleRoIAlign(featmap_names=['0'],
                                                output_size=7,
                                                sampling_ratio=2)

# put the pieces together inside a FasterRCNN model
model_option_2 = FasterRCNN(backbone,
                   num_classes=2,
                   rpn_anchor_generator=anchor_generator,
                   box_roi_pool=roi_pooler)

device = torch.device('cuda') if torch.cuda.is_available() else torch.
 ↪device('cpu')

# move model to the right device
# Changes
model_option_2.to(device)

# construct an optimizer
params = [p for p in model_option_2.parameters() if p.requires_grad]
optimizer = torch.optim.SGD(params, lr=0.005,
                            momentum=0.9, weight_decay=0.0005)

# and a learning rate scheduler which decreases the learning rate by
# 10x every 3 epochs
lr_scheduler = torch.optim.lr_scheduler.StepLR(optimizer, step_size=3, gamma=0.
 ↪1)

for epoch in range(num_epochs):
    # train for one epoch, printing every 10 iterations
    train_one_epoch(model_option_2, optimizer, data_loader, device, epoch,␣
 ↪print_freq=10)
    # update the learning rate
    lr_scheduler.step()
    # evaluate on the test dataset
    evaluate(model_option_2, data_loader_test, device=device)
```

```
Epoch: [0]  [ 0/60]  eta: 0:00:52  lr: 0.000090  loss: 1.3941 (1.3941)
loss_classifier: 0.6496 (0.6496)  loss_box_reg: 0.0400 (0.0400)
loss_objectness: 0.6887 (0.6887)  loss_rpn_box_reg: 0.0158 (0.0158)  time:
0.8816  data: 0.4675  max mem: 6172
Epoch: [0]  [10/60]  eta: 0:00:18  lr: 0.000936  loss: 1.3941 (1.3767)
loss_classifier: 0.6156 (0.5823)  loss_box_reg: 0.0700 (0.0690)
loss_objectness: 0.6834 (0.6828)  loss_rpn_box_reg: 0.0387 (0.0426)  time:
0.3769  data: 0.0491  max mem: 6172
Epoch: [0]  [20/60]  eta: 0:00:14  lr: 0.001783  loss: 1.2109 (1.2129)
loss_classifier: 0.4132 (0.4440)  loss_box_reg: 0.0804 (0.0921)
loss_objectness: 0.6480 (0.6397)  loss_rpn_box_reg: 0.0345 (0.0371)  time:
0.3255  data: 0.0079  max mem: 6172
Epoch: [0]  [30/60]  eta: 0:00:10  lr: 0.002629  loss: 0.9969 (1.1282)
loss_classifier: 0.2775 (0.3978)  loss_box_reg: 0.1245 (0.1165)
loss_objectness: 0.5145 (0.5772)  loss_rpn_box_reg: 0.0309 (0.0367)  time:
0.3255  data: 0.0084  max mem: 6172
Epoch: [0]  [40/60]  eta: 0:00:06  lr: 0.003476  loss: 0.8696 (1.0501)
loss_classifier: 0.2537 (0.3638)  loss_box_reg: 0.1726 (0.1344)
loss_objectness: 0.3614 (0.5140)  loss_rpn_box_reg: 0.0379 (0.0378)  time:
0.3286  data: 0.0092  max mem: 6172
Epoch: [0]  [50/60]  eta: 0:00:03  lr: 0.004323  loss: 0.7329 (0.9764)
loss_classifier: 0.2304 (0.3390)  loss_box_reg: 0.1774 (0.1427)
loss_objectness: 0.2598 (0.4581)  loss_rpn_box_reg: 0.0372 (0.0365)  time:
0.3278  data: 0.0094  max mem: 6311
Epoch: [0]  [59/60]  eta: 0:00:00  lr: 0.005000  loss: 0.5542 (0.9140)
loss_classifier: 0.2203 (0.3174)  loss_box_reg: 0.1513 (0.1441)
loss_objectness: 0.1963 (0.4166)  loss_rpn_box_reg: 0.0310 (0.0359)  time:
0.3243  data: 0.0087  max mem: 6311
Epoch: [0] Total time: 0:00:20 (0.3383 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:22  model_time: 0.0861 (0.0861)  evaluator_time:
0.0043 (0.0043)  time: 0.4502  data: 0.3587  max mem: 6311
Test:  [49/50]  eta: 0:00:00  model_time: 0.0360 (0.0385)  evaluator_time:
0.0024 (0.0030)  time: 0.0444  data: 0.0042  max mem: 6311
Test: Total time: 0:00:02 (0.0576 s / it)
Averaged stats: model_time: 0.0360 (0.0385)  evaluator_time: 0.0024 (0.0030)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.045
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.150
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.006
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.001
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.098
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.047
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.255
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.352
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.017
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.386
Epoch: [1]  [ 0/60]  eta: 0:00:48  lr: 0.005000  loss: 0.8100 (0.8100)
loss_classifier: 0.2839 (0.2839)  loss_box_reg: 0.2739 (0.2739)
loss_objectness: 0.1958 (0.1958)  loss_rpn_box_reg: 0.0563 (0.0563)  time:
0.8016  data: 0.3917  max mem: 6311
Epoch: [1]  [10/60]  eta: 0:00:18  lr: 0.005000  loss: 0.4534 (0.5334)
loss_classifier: 0.1523 (0.1795)  loss_box_reg: 0.1344 (0.1588)
loss_objectness: 0.1527 (0.1623)  loss_rpn_box_reg: 0.0270 (0.0328)  time:
0.3706  data: 0.0435  max mem: 6311
Epoch: [1]  [20/60]  eta: 0:00:14  lr: 0.005000  loss: 0.4769 (0.5590)
loss_classifier: 0.1566 (0.1898)  loss_box_reg: 0.1545 (0.1810)
loss_objectness: 0.1515 (0.1555)  loss_rpn_box_reg: 0.0293 (0.0327)  time:
0.3308  data: 0.0087  max mem: 6311
Epoch: [1]  [30/60]  eta: 0:00:10  lr: 0.005000  loss: 0.5281 (0.5530)
loss_classifier: 0.1672 (0.1865)  loss_box_reg: 0.1838 (0.1859)
loss_objectness: 0.1397 (0.1485)  loss_rpn_box_reg: 0.0310 (0.0321)  time:
0.3331  data: 0.0094  max mem: 6311
Epoch: [1]  [40/60]  eta: 0:00:06  lr: 0.005000  loss: 0.4674 (0.5419)
loss_classifier: 0.1397 (0.1805)  loss_box_reg: 0.1539 (0.1848)
loss_objectness: 0.1286 (0.1429)  loss_rpn_box_reg: 0.0317 (0.0337)  time:
0.3316  data: 0.0095  max mem: 6311
Epoch: [1]  [50/60]  eta: 0:00:03  lr: 0.005000  loss: 0.4674 (0.5381)
loss_classifier: 0.1368 (0.1775)  loss_box_reg: 0.1667 (0.1909)
loss_objectness: 0.1212 (0.1365)  loss_rpn_box_reg: 0.0340 (0.0331)  time:
0.3398  data: 0.0091  max mem: 6618
Epoch: [1]  [59/60]  eta: 0:00:00  lr: 0.005000  loss: 0.3847 (0.5144)
loss_classifier: 0.1274 (0.1689)  loss_box_reg: 0.1514 (0.1852)
loss_objectness: 0.0913 (0.1296)  loss_rpn_box_reg: 0.0162 (0.0307)  time:
0.3363  data: 0.0087  max mem: 6618
Epoch: [1] Total time: 0:00:20 (0.3444 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:19  model_time: 0.0760 (0.0760)  evaluator_time:
0.0129 (0.0129)  time: 0.3836  data: 0.2936  max mem: 6618
Test:  [49/50]  eta: 0:00:00  model_time: 0.0302 (0.0360)  evaluator_time:
0.0025 (0.0025)  time: 0.0410  data: 0.0047  max mem: 6618
Test: Total time: 0:00:02 (0.0546 s / it)
Averaged stats: model_time: 0.0302 (0.0360)  evaluator_time: 0.0025 (0.0025)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.203
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.591
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.046
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.004
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.224
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.125
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.364
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.382
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.017
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.419
Epoch: [2]  [ 0/60]  eta: 0:00:48  lr: 0.005000  loss: 0.2413 (0.2413)
loss_classifier: 0.0676 (0.0676)  loss_box_reg: 0.0788 (0.0788)
loss_objectness: 0.0775 (0.0775)  loss_rpn_box_reg: 0.0175 (0.0175)  time:
0.8008  data: 0.4975  max mem: 6618
Epoch: [2]  [10/60]  eta: 0:00:18  lr: 0.005000  loss: 0.3504 (0.3711)
loss_classifier: 0.1112 (0.1168)  loss_box_reg: 0.1132 (0.1311)
loss_objectness: 0.0987 (0.0956)  loss_rpn_box_reg: 0.0214 (0.0276)  time:
0.3653  data: 0.0497  max mem: 6618
Epoch: [2]  [20/60]  eta: 0:00:14  lr: 0.005000  loss: 0.3504 (0.3814)
loss_classifier: 0.1112 (0.1175)  loss_box_reg: 0.1396 (0.1504)
loss_objectness: 0.0907 (0.0888)  loss_rpn_box_reg: 0.0214 (0.0248)  time:
0.3302  data: 0.0070  max mem: 6618
Epoch: [2]  [30/60]  eta: 0:00:10  lr: 0.005000  loss: 0.3664 (0.4233)
loss_classifier: 0.1116 (0.1308)  loss_box_reg: 0.1791 (0.1742)
loss_objectness: 0.0747 (0.0918)  loss_rpn_box_reg: 0.0218 (0.0265)  time:
0.3367  data: 0.0092  max mem: 6618
Epoch: [2]  [40/60]  eta: 0:00:06  lr: 0.005000  loss: 0.4420 (0.4344)
loss_classifier: 0.1350 (0.1334)  loss_box_reg: 0.2059 (0.1845)
loss_objectness: 0.0806 (0.0894)  loss_rpn_box_reg: 0.0254 (0.0270)  time:
0.3337  data: 0.0089  max mem: 6618
Epoch: [2]  [50/60]  eta: 0:00:03  lr: 0.005000  loss: 0.4420 (0.4320)
loss_classifier: 0.1350 (0.1326)  loss_box_reg: 0.2158 (0.1873)
loss_objectness: 0.0700 (0.0858)  loss_rpn_box_reg: 0.0254 (0.0263)  time:
0.3382  data: 0.0089  max mem: 6618
Epoch: [2]  [59/60]  eta: 0:00:00  lr: 0.005000  loss: 0.3496 (0.4169)
loss_classifier: 0.1072 (0.1275)  loss_box_reg: 0.1482 (0.1788)
loss_objectness: 0.0682 (0.0849)  loss_rpn_box_reg: 0.0214 (0.0257)  time:
0.3352  data: 0.0092  max mem: 6618
Epoch: [2] Total time: 0:00:20 (0.3442 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:19  model_time: 0.0727 (0.0727)  evaluator_time:
0.0047 (0.0047)  time: 0.3965  data: 0.3178  max mem: 6618
Test:  [49/50]  eta: 0:00:00  model_time: 0.0360 (0.0395)  evaluator_time:
0.0024 (0.0027)  time: 0.0450  data: 0.0048  max mem: 6618
Test: Total time: 0:00:02 (0.0584 s / it)
Averaged stats: model_time: 0.0360 (0.0395)  evaluator_time: 0.0024 (0.0027)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
```

```
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.185
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.561
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.031
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.024
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.204
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.138
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.353
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.394
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.058
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.428
Epoch: [3]  [ 0/60]  eta: 0:00:43  lr: 0.000500  loss: 0.3568 (0.3568)
loss_classifier: 0.1135 (0.1135)  loss_box_reg: 0.1811 (0.1811)
loss_objectness: 0.0437 (0.0437)  loss_rpn_box_reg: 0.0185 (0.0185)  time:
0.7236  data: 0.3614  max mem: 6618
Epoch: [3]  [10/60]  eta: 0:00:18  lr: 0.000500  loss: 0.2734 (0.3089)
loss_classifier: 0.0861 (0.0929)  loss_box_reg: 0.1339 (0.1443)
loss_objectness: 0.0506 (0.0530)  loss_rpn_box_reg: 0.0185 (0.0188)  time:
0.3735  data: 0.0411  max mem: 6618
Epoch: [3]  [20/60]  eta: 0:00:14  lr: 0.000500  loss: 0.3210 (0.3442)
loss_classifier: 0.1048 (0.1073)  loss_box_reg: 0.1563 (0.1613)
loss_objectness: 0.0506 (0.0559)  loss_rpn_box_reg: 0.0164 (0.0197)  time:
0.3404  data: 0.0088  max mem: 6618
Epoch: [3]  [30/60]  eta: 0:00:10  lr: 0.000500  loss: 0.3541 (0.3658)
loss_classifier: 0.1152 (0.1118)  loss_box_reg: 0.1734 (0.1701)
loss_objectness: 0.0540 (0.0614)  loss_rpn_box_reg: 0.0227 (0.0224)  time:
0.3444  data: 0.0090  max mem: 6618
Epoch: [3]  [40/60]  eta: 0:00:07  lr: 0.000500  loss: 0.4011 (0.3683)
loss_classifier: 0.1207 (0.1127)  loss_box_reg: 0.1689 (0.1714)
loss_objectness: 0.0627 (0.0623)  loss_rpn_box_reg: 0.0211 (0.0219)  time:
0.3413  data: 0.0092  max mem: 6660
Epoch: [3]  [50/60]  eta: 0:00:03  lr: 0.000500  loss: 0.3294 (0.3640)
loss_classifier: 0.1037 (0.1119)  loss_box_reg: 0.1594 (0.1699)
loss_objectness: 0.0524 (0.0610)  loss_rpn_box_reg: 0.0173 (0.0211)  time:
0.3357  data: 0.0088  max mem: 6660
Epoch: [3]  [59/60]  eta: 0:00:00  lr: 0.000500  loss: 0.3350 (0.3582)
loss_classifier: 0.1049 (0.1099)  loss_box_reg: 0.1551 (0.1673)
loss_objectness: 0.0474 (0.0597)  loss_rpn_box_reg: 0.0190 (0.0213)  time:
0.3360  data: 0.0083  max mem: 6660
Epoch: [3] Total time: 0:00:20 (0.3486 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:21  model_time: 0.1053 (0.1053)  evaluator_time:
0.0199 (0.0199)  time: 0.4358  data: 0.3027  max mem: 6660
Test:  [49/50]  eta: 0:00:00  model_time: 0.0353 (0.0391)  evaluator_time:
0.0025 (0.0031)  time: 0.0437  data: 0.0041  max mem: 6660
Test: Total time: 0:00:02 (0.0576 s / it)
```

```
Averaged stats: model_time: 0.0353 (0.0391)  evaluator_time: 0.0025 (0.0031)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.257
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.654
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.145
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.014
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.284
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.150
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.448
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.457
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.033
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.500
Epoch: [4]  [ 0/60]  eta: 0:00:46  lr: 0.000500  loss: 0.2900 (0.2900)
loss_classifier: 0.0966 (0.0966)  loss_box_reg: 0.1223 (0.1223)
loss_objectness: 0.0514 (0.0514)  loss_rpn_box_reg: 0.0198 (0.0198)  time:
0.7710  data: 0.3953  max mem: 6660
Epoch: [4]  [10/60]  eta: 0:00:18  lr: 0.000500  loss: 0.3474 (0.3662)
loss_classifier: 0.1049 (0.1149)  loss_box_reg: 0.1795 (0.1757)
loss_objectness: 0.0507 (0.0555)  loss_rpn_box_reg: 0.0190 (0.0201)  time:
0.3713  data: 0.0431  max mem: 6660
Epoch: [4]  [20/60]  eta: 0:00:14  lr: 0.000500  loss: 0.3243 (0.3359)
loss_classifier: 0.0967 (0.1048)  loss_box_reg: 0.1308 (0.1504)
loss_objectness: 0.0491 (0.0597)  loss_rpn_box_reg: 0.0190 (0.0210)  time:
0.3385  data: 0.0086  max mem: 6660
Epoch: [4]  [30/60]  eta: 0:00:10  lr: 0.000500  loss: 0.3399 (0.3623)
loss_classifier: 0.0969 (0.1114)  loss_box_reg: 0.1323 (0.1696)
loss_objectness: 0.0490 (0.0590)  loss_rpn_box_reg: 0.0211 (0.0224)  time:
0.3450  data: 0.0097  max mem: 6660
Epoch: [4]  [40/60]  eta: 0:00:06  lr: 0.000500  loss: 0.2919 (0.3460)
loss_classifier: 0.0995 (0.1066)  loss_box_reg: 0.1504 (0.1614)
loss_objectness: 0.0492 (0.0571)  loss_rpn_box_reg: 0.0179 (0.0210)  time:
0.3340  data: 0.0094  max mem: 6660
Epoch: [4]  [50/60]  eta: 0:00:03  lr: 0.000500  loss: 0.2919 (0.3454)
loss_classifier: 0.0964 (0.1062)  loss_box_reg: 0.1393 (0.1623)
loss_objectness: 0.0477 (0.0556)  loss_rpn_box_reg: 0.0170 (0.0214)  time:
0.3386  data: 0.0087  max mem: 6660
Epoch: [4]  [59/60]  eta: 0:00:00  lr: 0.000500  loss: 0.3109 (0.3447)
loss_classifier: 0.1001 (0.1066)  loss_box_reg: 0.1401 (0.1614)
loss_objectness: 0.0452 (0.0558)  loss_rpn_box_reg: 0.0166 (0.0209)  time:
0.3458  data: 0.0084  max mem: 6660
Epoch: [4] Total time: 0:00:21 (0.3501 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:20  model_time: 0.0852 (0.0852)  evaluator_time:
```

```
0.0043 (0.0043)  time: 0.4017  data: 0.3109  max mem: 6660
Test:  [49/50]  eta: 0:00:00  model_time: 0.0346 (0.0386)  evaluator_time:
0.0019 (0.0025)  time: 0.0436  data: 0.0046  max mem: 6660
Test: Total time: 0:00:02 (0.0572 s / it)
Averaged stats: model_time: 0.0346 (0.0386)  evaluator_time: 0.0019 (0.0025)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.251
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.694
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.099
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.026
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.279
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.155
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.422
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.443
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.083
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.480
Epoch: [5]  [ 0/60]  eta: 0:00:49  lr: 0.000500  loss: 0.4341 (0.4341)
loss_classifier: 0.1329 (0.1329)  loss_box_reg: 0.2123 (0.2123)
loss_objectness: 0.0530 (0.0530)  loss_rpn_box_reg: 0.0359 (0.0359)  time:
0.8325  data: 0.4548  max mem: 6660
Epoch: [5]  [10/60]  eta: 0:00:19  lr: 0.000500  loss: 0.3539 (0.3643)
loss_classifier: 0.1027 (0.1085)  loss_box_reg: 0.1730 (0.1745)
loss_objectness: 0.0511 (0.0566)  loss_rpn_box_reg: 0.0238 (0.0247)  time:
0.3892  data: 0.0481  max mem: 6660
Epoch: [5]  [20/60]  eta: 0:00:14  lr: 0.000500  loss: 0.3401 (0.3573)
loss_classifier: 0.0937 (0.1069)  loss_box_reg: 0.1689 (0.1723)
loss_objectness: 0.0508 (0.0549)  loss_rpn_box_reg: 0.0208 (0.0232)  time:
0.3440  data: 0.0090  max mem: 6660
Epoch: [5]  [30/60]  eta: 0:00:10  lr: 0.000500  loss: 0.2920 (0.3398)
loss_classifier: 0.0870 (0.1037)  loss_box_reg: 0.1556 (0.1642)
loss_objectness: 0.0412 (0.0504)  loss_rpn_box_reg: 0.0203 (0.0215)  time:
0.3412  data: 0.0103  max mem: 6660
Epoch: [5]  [40/60]  eta: 0:00:07  lr: 0.000500  loss: 0.2920 (0.3483)
loss_classifier: 0.0958 (0.1059)  loss_box_reg: 0.1535 (0.1640)
loss_objectness: 0.0488 (0.0571)  loss_rpn_box_reg: 0.0180 (0.0214)  time:
0.3382  data: 0.0095  max mem: 6660
Epoch: [5]  [50/60]  eta: 0:00:03  lr: 0.000500  loss: 0.3315 (0.3370)
loss_classifier: 0.0941 (0.1034)  loss_box_reg: 0.1555 (0.1590)
loss_objectness: 0.0458 (0.0543)  loss_rpn_box_reg: 0.0161 (0.0202)  time:
0.3399  data: 0.0088  max mem: 6660
Epoch: [5]  [59/60]  eta: 0:00:00  lr: 0.000500  loss: 0.2708 (0.3371)
loss_classifier: 0.0890 (0.1039)  loss_box_reg: 0.1263 (0.1590)
loss_objectness: 0.0443 (0.0543)  loss_rpn_box_reg: 0.0126 (0.0198)  time:
0.3443  data: 0.0084  max mem: 6660
```

```
Epoch: [5] Total time: 0:00:21 (0.3539 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:22  model_time: 0.0900 (0.0900)  evaluator_time:
0.0048 (0.0048)  time: 0.4539  data: 0.3578  max mem: 6660
Test:  [49/50]  eta: 0:00:00  model_time: 0.0353 (0.0393)  evaluator_time:
0.0023 (0.0024)  time: 0.0437  data: 0.0047  max mem: 6660
Test: Total time: 0:00:02 (0.0582 s / it)
Averaged stats: model_time: 0.0353 (0.0393)  evaluator_time: 0.0023 (0.0024)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.275
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.711
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.133
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.030
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.300
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.149
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.439
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.451
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.108
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.486
Epoch: [6]  [ 0/60]  eta: 0:00:53  lr: 0.000050  loss: 0.1932 (0.1932)
loss_classifier: 0.0597 (0.0597)  loss_box_reg: 0.0820 (0.0820)
loss_objectness: 0.0363 (0.0363)  loss_rpn_box_reg: 0.0152 (0.0152)  time:
0.8935  data: 0.5026  max mem: 6660
Epoch: [6]  [10/60]  eta: 0:00:19  lr: 0.000050  loss: 0.3186 (0.3075)
loss_classifier: 0.0996 (0.0959)  loss_box_reg: 0.1537 (0.1470)
loss_objectness: 0.0481 (0.0471)  loss_rpn_box_reg: 0.0187 (0.0175)  time:
0.3885  data: 0.0519  max mem: 6660
Epoch: [6]  [20/60]  eta: 0:00:14  lr: 0.000050  loss: 0.3332 (0.3224)
loss_classifier: 0.1003 (0.0987)  loss_box_reg: 0.1537 (0.1507)
loss_objectness: 0.0490 (0.0529)  loss_rpn_box_reg: 0.0205 (0.0201)  time:
0.3474  data: 0.0075  max mem: 6660
Epoch: [6]  [30/60]  eta: 0:00:10  lr: 0.000050  loss: 0.3388 (0.3358)
loss_classifier: 0.1008 (0.1022)  loss_box_reg: 0.1621 (0.1597)
loss_objectness: 0.0469 (0.0532)  loss_rpn_box_reg: 0.0224 (0.0207)  time:
0.3506  data: 0.0083  max mem: 6660
Epoch: [6]  [40/60]  eta: 0:00:07  lr: 0.000050  loss: 0.3205 (0.3227)
loss_classifier: 0.0886 (0.0986)  loss_box_reg: 0.1492 (0.1550)
loss_objectness: 0.0387 (0.0496)  loss_rpn_box_reg: 0.0184 (0.0195)  time:
0.3415  data: 0.0086  max mem: 6660
Epoch: [6]  [50/60]  eta: 0:00:03  lr: 0.000050  loss: 0.2788 (0.3159)
loss_classifier: 0.0868 (0.0960)  loss_box_reg: 0.1301 (0.1506)
loss_objectness: 0.0387 (0.0496)  loss_rpn_box_reg: 0.0165 (0.0197)  time:
0.3415  data: 0.0094  max mem: 6660
```

```
Epoch: [6]  [59/60]  eta: 0:00:00  lr: 0.000050  loss: 0.3402 (0.3250)
loss_classifier: 0.0984 (0.0998)  loss_box_reg: 0.1544 (0.1547)
loss_objectness: 0.0495 (0.0506)  loss_rpn_box_reg: 0.0198 (0.0199)  time:
0.3459  data: 0.0089  max mem: 6660
Epoch: [6] Total time: 0:00:21 (0.3570 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:21  model_time: 0.0960 (0.0960)  evaluator_time:
0.0136 (0.0136)  time: 0.4279  data: 0.3171  max mem: 6660
Test:  [49/50]  eta: 0:00:00  model_time: 0.0355 (0.0396)  evaluator_time:
0.0023 (0.0026)  time: 0.0436  data: 0.0044  max mem: 6660
Test: Total time: 0:00:02 (0.0576 s / it)
Averaged stats: model_time: 0.0355 (0.0396)  evaluator_time: 0.0023 (0.0026)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.262
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.691
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.124
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.020
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.294
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.177
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.439
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.451
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.075
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.489
Epoch: [7]  [ 0/60]  eta: 0:00:47  lr: 0.000050  loss: 0.2634 (0.2634)
loss_classifier: 0.0798 (0.0798)  loss_box_reg: 0.1262 (0.1262)
loss_objectness: 0.0376 (0.0376)  loss_rpn_box_reg: 0.0197 (0.0197)  time:
0.7959  data: 0.3951  max mem: 6660
Epoch: [7]  [10/60]  eta: 0:00:19  lr: 0.000050  loss: 0.2942 (0.3072)
loss_classifier: 0.0832 (0.0900)  loss_box_reg: 0.1278 (0.1397)
loss_objectness: 0.0528 (0.0606)  loss_rpn_box_reg: 0.0178 (0.0169)  time:
0.3823  data: 0.0433  max mem: 6660
Epoch: [7]  [20/60]  eta: 0:00:14  lr: 0.000050  loss: 0.2974 (0.3286)
loss_classifier: 0.0893 (0.0997)  loss_box_reg: 0.1457 (0.1539)
loss_objectness: 0.0528 (0.0569)  loss_rpn_box_reg: 0.0179 (0.0181)  time:
0.3416  data: 0.0083  max mem: 6660
Epoch: [7]  [30/60]  eta: 0:00:10  lr: 0.000050  loss: 0.3806 (0.3508)
loss_classifier: 0.1128 (0.1061)  loss_box_reg: 0.1885 (0.1669)
loss_objectness: 0.0503 (0.0581)  loss_rpn_box_reg: 0.0206 (0.0198)  time:
0.3449  data: 0.0088  max mem: 6660
Epoch: [7]  [40/60]  eta: 0:00:07  lr: 0.000050  loss: 0.3522 (0.3436)
loss_classifier: 0.1106 (0.1044)  loss_box_reg: 0.1847 (0.1639)
loss_objectness: 0.0470 (0.0556)  loss_rpn_box_reg: 0.0206 (0.0197)  time:
0.3469  data: 0.0088  max mem: 6660
```

```
Epoch: [7]  [50/60]  eta: 0:00:03  lr: 0.000050  loss: 0.2993 (0.3382)
loss_classifier: 0.0824 (0.1029)  loss_box_reg: 0.1278 (0.1615)
loss_objectness: 0.0420 (0.0538)  loss_rpn_box_reg: 0.0186 (0.0201)  time:
0.3434  data: 0.0093  max mem: 6660
Epoch: [7]  [59/60]  eta: 0:00:00  lr: 0.000050  loss: 0.2993 (0.3301)
loss_classifier: 0.0840 (0.1006)  loss_box_reg: 0.1278 (0.1571)
loss_objectness: 0.0422 (0.0531)  loss_rpn_box_reg: 0.0151 (0.0193)  time:
0.3404  data: 0.0093  max mem: 6660
Epoch: [7] Total time: 0:00:21 (0.3532 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:20  model_time: 0.0931 (0.0931)  evaluator_time:
0.0146 (0.0146)  time: 0.4030  data: 0.2940  max mem: 6660
Test:  [49/50]  eta: 0:00:00  model_time: 0.0349 (0.0393)  evaluator_time:
0.0025 (0.0027)  time: 0.0442  data: 0.0046  max mem: 6660
Test: Total time: 0:00:02 (0.0576 s / it)
Averaged stats: model_time: 0.0349 (0.0393)  evaluator_time: 0.0025 (0.0027)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.256
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.664
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.102
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.049
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.282
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.134
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.445
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.450
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.117
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.484
Epoch: [8]  [ 0/60]  eta: 0:00:51  lr: 0.000050  loss: 0.3596 (0.3596)
loss_classifier: 0.1115 (0.1115)  loss_box_reg: 0.1839 (0.1839)
loss_objectness: 0.0439 (0.0439)  loss_rpn_box_reg: 0.0203 (0.0203)  time:
0.8504  data: 0.4487  max mem: 6660
Epoch: [8]  [10/60]  eta: 0:00:19  lr: 0.000050  loss: 0.3550 (0.3759)
loss_classifier: 0.1056 (0.1162)  loss_box_reg: 0.1732 (0.1820)
loss_objectness: 0.0556 (0.0556)  loss_rpn_box_reg: 0.0219 (0.0221)  time:
0.3879  data: 0.0482  max mem: 6660
Epoch: [8]  [20/60]  eta: 0:00:14  lr: 0.000050  loss: 0.3060 (0.3574)
loss_classifier: 0.0911 (0.1085)  loss_box_reg: 0.1659 (0.1757)
loss_objectness: 0.0524 (0.0526)  loss_rpn_box_reg: 0.0198 (0.0206)  time:
0.3401  data: 0.0086  max mem: 6660
Epoch: [8]  [30/60]  eta: 0:00:10  lr: 0.000050  loss: 0.2957 (0.3427)
loss_classifier: 0.0913 (0.1052)  loss_box_reg: 0.1455 (0.1639)
loss_objectness: 0.0428 (0.0545)  loss_rpn_box_reg: 0.0152 (0.0190)  time:
0.3410  data: 0.0092  max mem: 6660
```

```
Epoch: [8]  [40/60]  eta: 0:00:07  lr: 0.000050  loss: 0.3075 (0.3406)
loss_classifier: 0.0925 (0.1054)  loss_box_reg: 0.1455 (0.1631)
loss_objectness: 0.0410 (0.0533)  loss_rpn_box_reg: 0.0176 (0.0188)  time:
0.3468  data: 0.0094  max mem: 6660
Epoch: [8]  [50/60]  eta: 0:00:03  lr: 0.000050  loss: 0.3086 (0.3364)
loss_classifier: 0.0925 (0.1048)  loss_box_reg: 0.1502 (0.1603)
loss_objectness: 0.0447 (0.0523)  loss_rpn_box_reg: 0.0182 (0.0190)  time:
0.3458  data: 0.0088  max mem: 6746
Epoch: [8]  [59/60]  eta: 0:00:00  lr: 0.000050  loss: 0.2822 (0.3279)
loss_classifier: 0.0922 (0.1024)  loss_box_reg: 0.1314 (0.1558)
loss_objectness: 0.0379 (0.0508)  loss_rpn_box_reg: 0.0154 (0.0188)  time:
0.3419  data: 0.0080  max mem: 6746
Epoch: [8] Total time: 0:00:21 (0.3553 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:21  model_time: 0.1226 (0.1226)  evaluator_time:
0.0048 (0.0048)  time: 0.4221  data: 0.2935  max mem: 6746
Test:  [49/50]  eta: 0:00:00  model_time: 0.0344 (0.0397)  evaluator_time:
0.0024 (0.0026)  time: 0.0440  data: 0.0047  max mem: 6746
Test: Total time: 0:00:02 (0.0577 s / it)
Averaged stats: model_time: 0.0344 (0.0397)  evaluator_time: 0.0024 (0.0026)
Accumulating evaluation results…
DONE (t=0.01s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.238
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.690
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.074
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.018
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.265
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.146
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.417
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.422
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.017
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.464
Epoch: [9]  [ 0/60]  eta: 0:00:48  lr: 0.000005  loss: 0.4472 (0.4472)
loss_classifier: 0.1144 (0.1144)  loss_box_reg: 0.2432 (0.2432)
loss_objectness: 0.0601 (0.0601)  loss_rpn_box_reg: 0.0296 (0.0296)  time:
0.8097  data: 0.4448  max mem: 6746
Epoch: [9]  [10/60]  eta: 0:00:18  lr: 0.000005  loss: 0.3388 (0.3266)
loss_classifier: 0.0903 (0.0994)  loss_box_reg: 0.1642 (0.1565)
loss_objectness: 0.0532 (0.0527)  loss_rpn_box_reg: 0.0172 (0.0180)  time:
0.3740  data: 0.0471  max mem: 6746
Epoch: [9]  [20/60]  eta: 0:00:14  lr: 0.000005  loss: 0.3066 (0.3390)
loss_classifier: 0.0853 (0.1042)  loss_box_reg: 0.1563 (0.1630)
loss_objectness: 0.0438 (0.0530)  loss_rpn_box_reg: 0.0156 (0.0189)  time:
0.3384  data: 0.0086  max mem: 6746
```

```
Epoch: [9]  [30/60]  eta: 0:00:10  lr: 0.000005  loss: 0.2782 (0.3328)
loss_classifier: 0.0883 (0.1025)  loss_box_reg: 0.1316 (0.1588)
loss_objectness: 0.0438 (0.0528)  loss_rpn_box_reg: 0.0155 (0.0187)  time:
0.3469  data: 0.0094  max mem: 6830
Epoch: [9]  [40/60]  eta: 0:00:07  lr: 0.000005  loss: 0.2940 (0.3329)
loss_classifier: 0.0883 (0.1012)  loss_box_reg: 0.1345 (0.1582)
loss_objectness: 0.0459 (0.0538)  loss_rpn_box_reg: 0.0180 (0.0197)  time:
0.3472  data: 0.0090  max mem: 6830
Epoch: [9]  [50/60]  eta: 0:00:03  lr: 0.000005  loss: 0.3042 (0.3365)
loss_classifier: 0.0942 (0.1027)  loss_box_reg: 0.1346 (0.1599)
loss_objectness: 0.0474 (0.0544)  loss_rpn_box_reg: 0.0180 (0.0194)  time:
0.3460  data: 0.0093  max mem: 6830
Epoch: [9]  [59/60]  eta: 0:00:00  lr: 0.000005  loss: 0.3316 (0.3402)
loss_classifier: 0.1007 (0.1041)  loss_box_reg: 0.1413 (0.1616)
loss_objectness: 0.0547 (0.0547)  loss_rpn_box_reg: 0.0188 (0.0199)  time:
0.3421  data: 0.0087  max mem: 6830
Epoch: [9] Total time: 0:00:21 (0.3532 s / it)
creating index…
index created!
Test:  [ 0/50]  eta: 0:00:21  model_time: 0.0845 (0.0845)  evaluator_time:
0.0053 (0.0053)  time: 0.4263  data: 0.3352  max mem: 6830
Test:  [49/50]  eta: 0:00:00  model_time: 0.0336 (0.0394)  evaluator_time:
0.0020 (0.0025)  time: 0.0425  data: 0.0044  max mem: 6830
Test: Total time: 0:00:02 (0.0582 s / it)
Averaged stats: model_time: 0.0336 (0.0394)  evaluator_time: 0.0020 (0.0025)
Accumulating evaluation results…
DONE (t=0.02s).
IoU metric: bbox
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.241
 Average Precision  (AP) @[ IoU=0.50      | area=   all | maxDets=100 ] = 0.678
 Average Precision  (AP) @[ IoU=0.75      | area=   all | maxDets=100 ] = 0.039
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Precision  (AP) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.026
 Average Precision  (AP) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.273
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=  1 ] = 0.144
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets= 10 ] = 0.390
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=   all | maxDets=100 ] = 0.394
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= small | maxDets=100 ] = -1.000
 Average Recall     (AR) @[ IoU=0.50:0.95 | area=medium | maxDets=100 ] = 0.042
 Average Recall     (AR) @[ IoU=0.50:0.95 | area= large | maxDets=100 ] = 0.430
```

After 10 epochs, we could notice that the option 1 (fine-tuned) model has lower triaining loss. And its AP and the AR metric are also better than the option 2 (different backbone) model.

Option 1 model AP and AR (bounding box):

Option 2 model AP and AR (bounding box):

Test option1 model and option2 model using Beatles picture

[56]:

```
# First we download the picture
!wget https://upload.wikimedia.org/wikipedia/en/4/42/Beatles_-_Abbey_Road.jpg
Image.open('Beatles_-_Abbey_Road.jpg')
```

```
--2022-10-14 21:53:00--
https://upload.wikimedia.org/wikipedia/en/4/42/Beatles_-_Abbey_Road.jpg
Resolving upload.wikimedia.org (upload.wikimedia.org)… 103.102.166.240,
2001:df2:e500:ed1a::2:b
Connecting to upload.wikimedia.org
(upload.wikimedia.org)|103.102.166.240|:443… connected.
HTTP request sent, awaiting response… 200 OK
Length: 89688 (88K) [image/jpeg]
Saving to: ‘Beatles_-_Abbey_Road.jpg.2’

Beatles_-_Abbey_Roa 100%[===================>]  87.59K  --.-KB/s    in 0.003s

2022-10-14 21:53:01 (26.6 MB/s) - ‘Beatles_-_Abbey_Road.jpg.2’ saved
[89688/89688]
```

[56]:

Validation of option 1 model:

[94]:

```
from torchvision import transforms
test_Beatles_Image  = Image.open('Beatles_-_Abbey_Road.jpg').convert("RGB")
test_Beatles_Image = test_Beatles_Image.resize((450,450))
test_Beatles_Image_Tensor = transforms.ToTensor()(test_Beatles_Image)

model.eval()
with torch.no_grad():
    predicted_Output = model([test_Beatles_Image_Tensor.to(device)])

predicted_Output
```

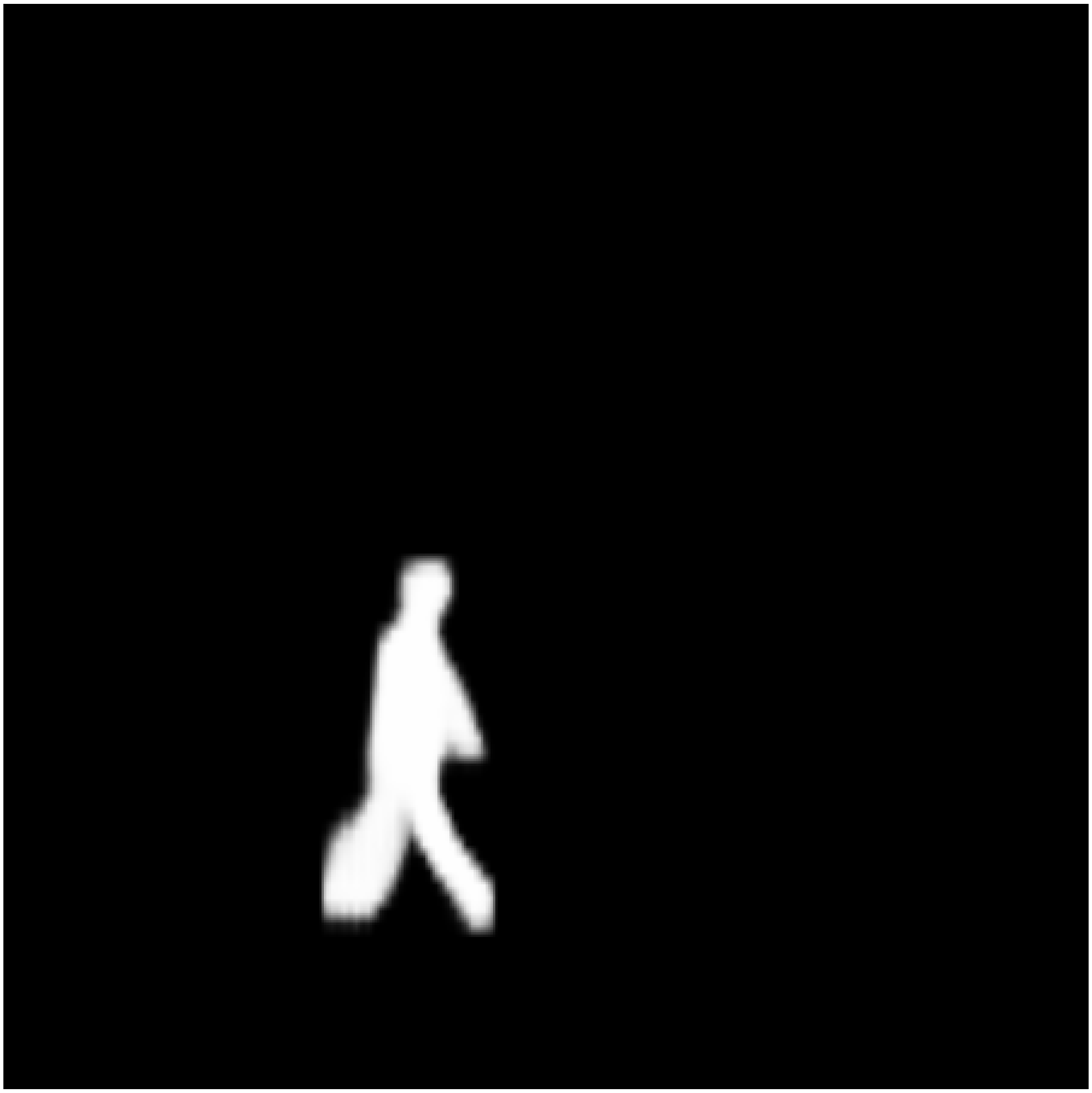
[94]:

```
[{'boxes': tensor([[132.7701, 230.7768, 202.8403, 383.5952],
          [215.5315, 238.9845, 302.2977, 396.4629],
          [ 32.9992, 229.8836, 116.1947, 382.2405],
          [311.6980, 236.8884, 400.8449, 411.6074],
          [336.9757, 210.0914, 345.8099, 244.1632],
          [ 78.9452, 225.4330, 193.7413, 382.2825],
          [ 49.1275, 230.3554, 333.3038, 389.2376]], device='cuda:0'),
  'labels': tensor([1, 1, 1, 1, 1, 1, 1], device='cuda:0'),
  'scores': tensor([0.9827, 0.9757, 0.9483, 0.8186, 0.1556, 0.0613, 0.0561],
         device='cuda:0'),
  'masks': tensor([[[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]],


          [[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]],


          [[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]],
          …,


          [[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]],


          [[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]],


          [[[0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            …,
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.],
            [0., 0., 0.,  …, 0., 0., 0.]]]], device='cuda:0')}]
```

[95]:

```
Image.fromarray(predicted_Output[0]['masks'][0, 0].mul(255).byte().cpu().
 ↪numpy())
```

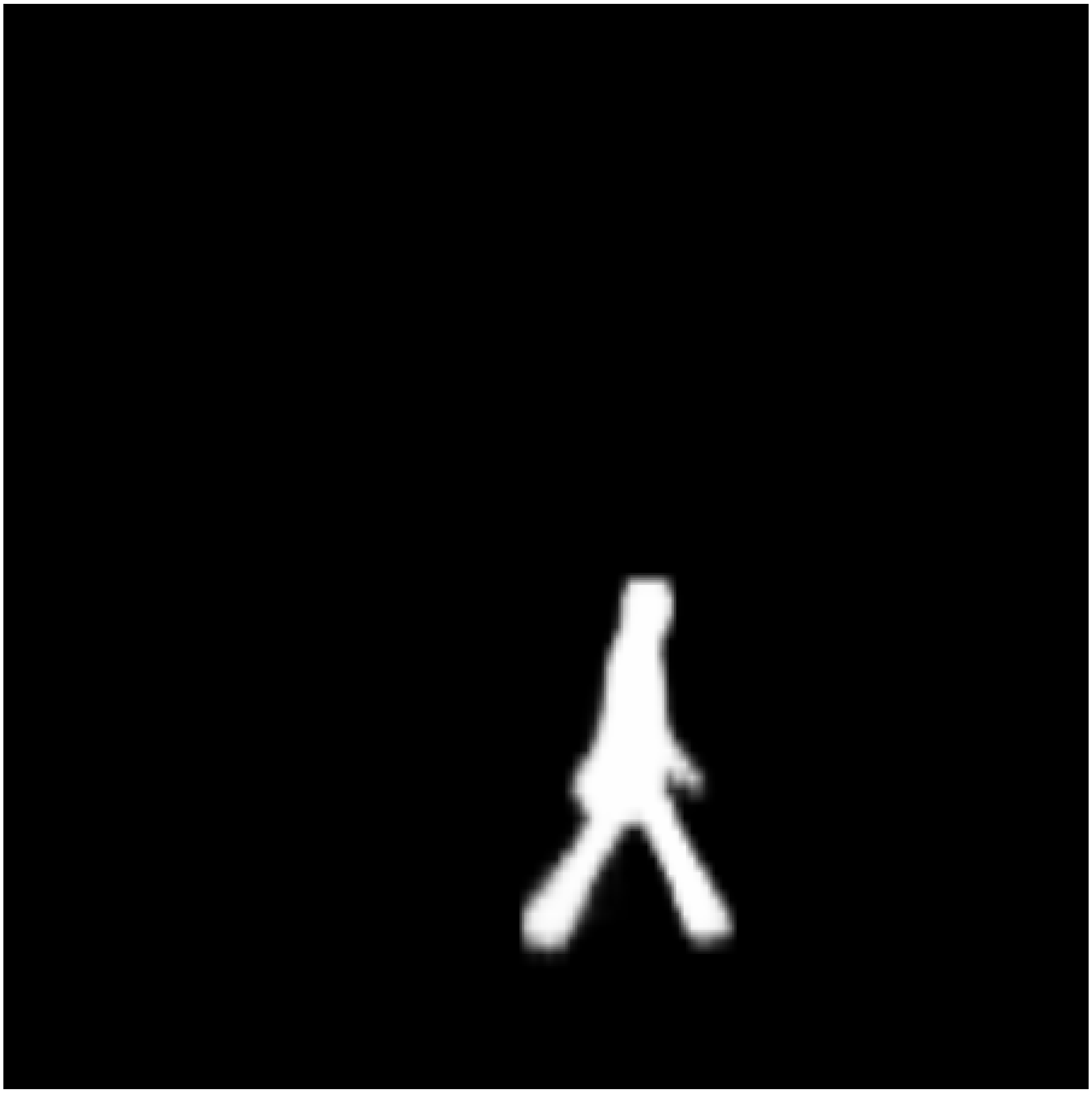
[95]:



[96]:

```
Image.fromarray(predicted_Output[0]['masks'][1, 0].mul(255).byte().cpu().
 ↪numpy())
```

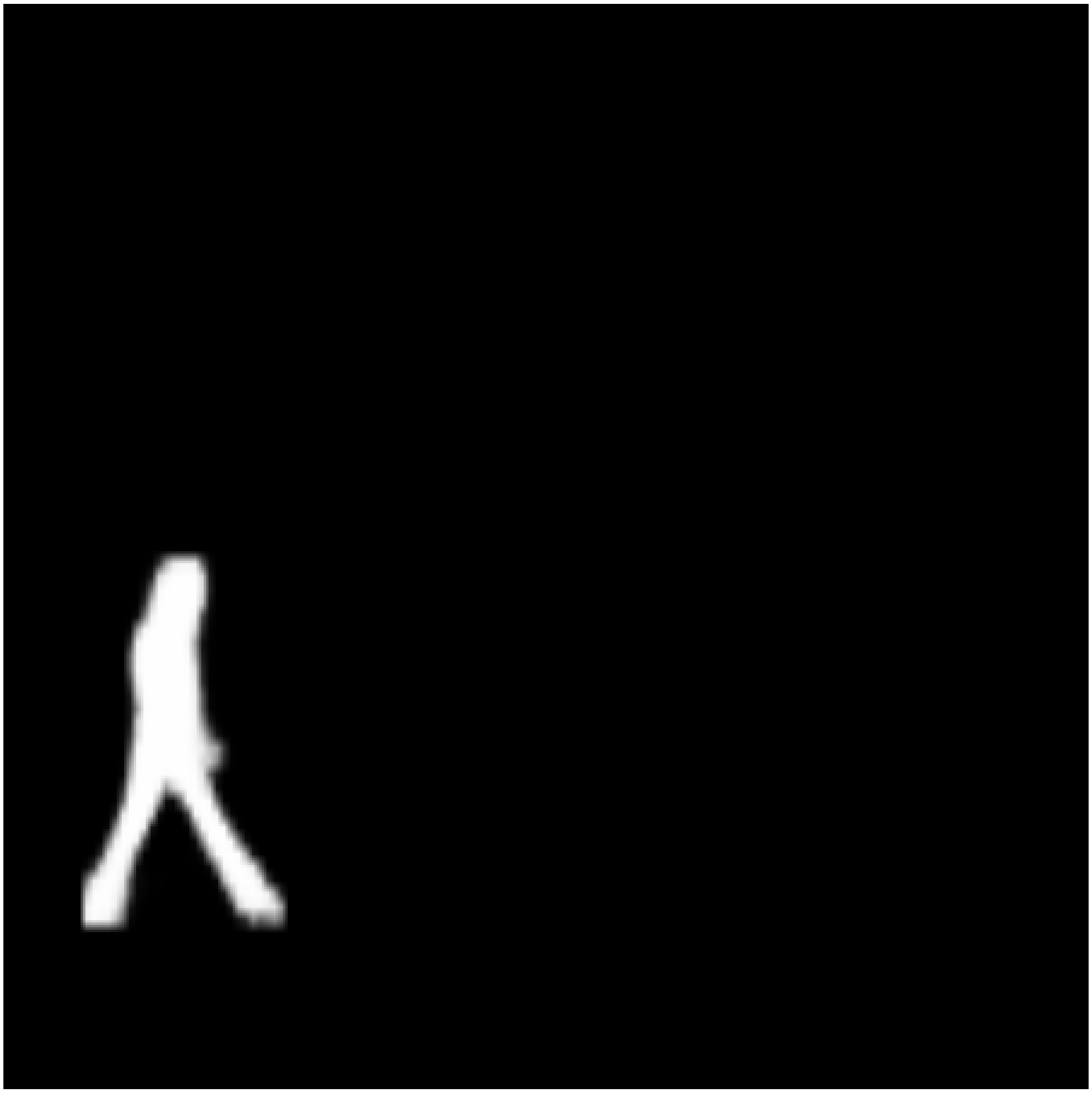
[96]:



[97]:

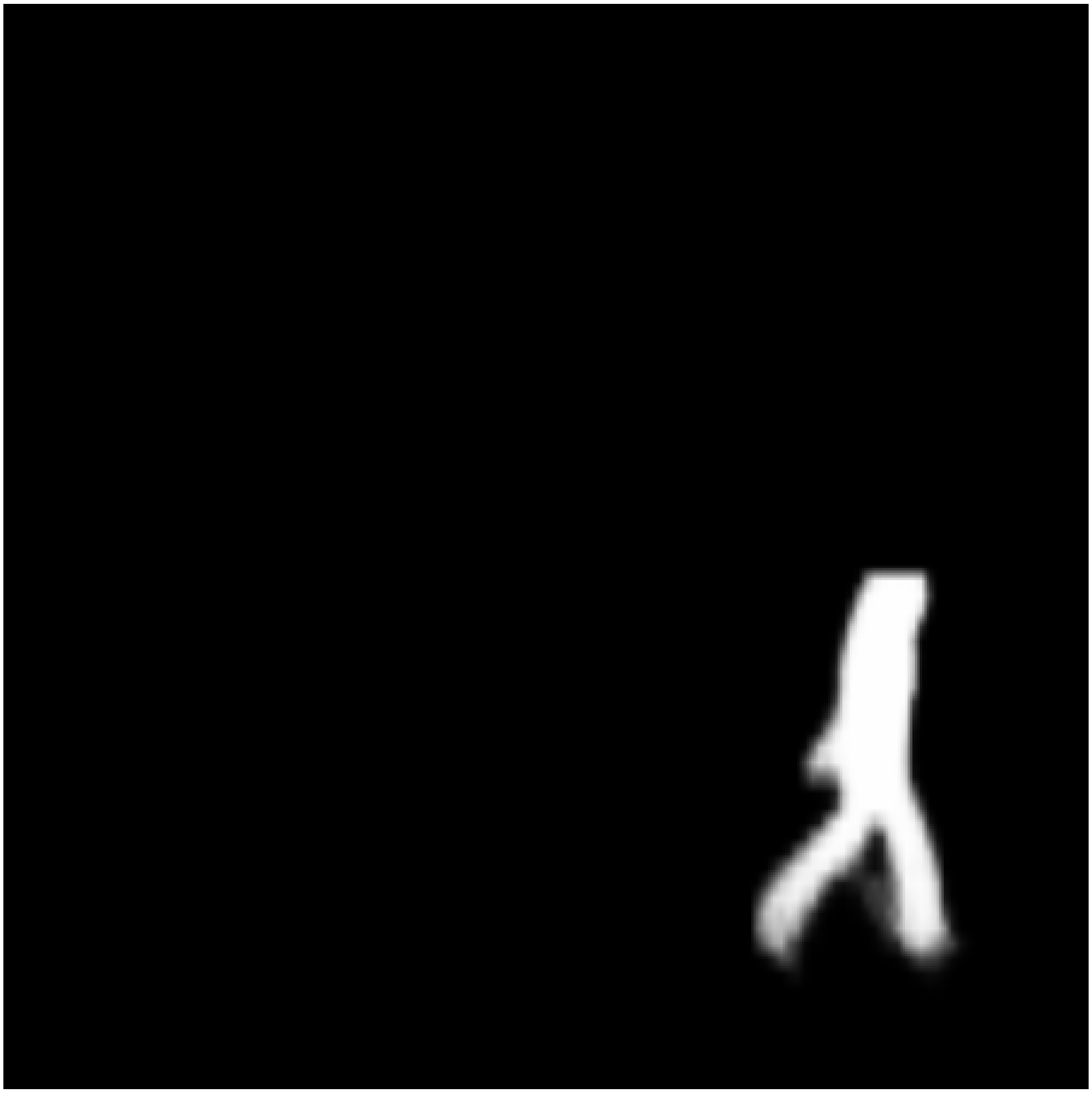
```
Image.fromarray(predicted_Output[0]['masks'][2, 0].mul(255).byte().cpu().
 ↪numpy())
```

[97]:



```
[98]: Image.fromarray(predicted_Output[0]['masks'][3, 0].mul(255).byte().cpu().
      ↪numpy())
```

```
[98]:
```



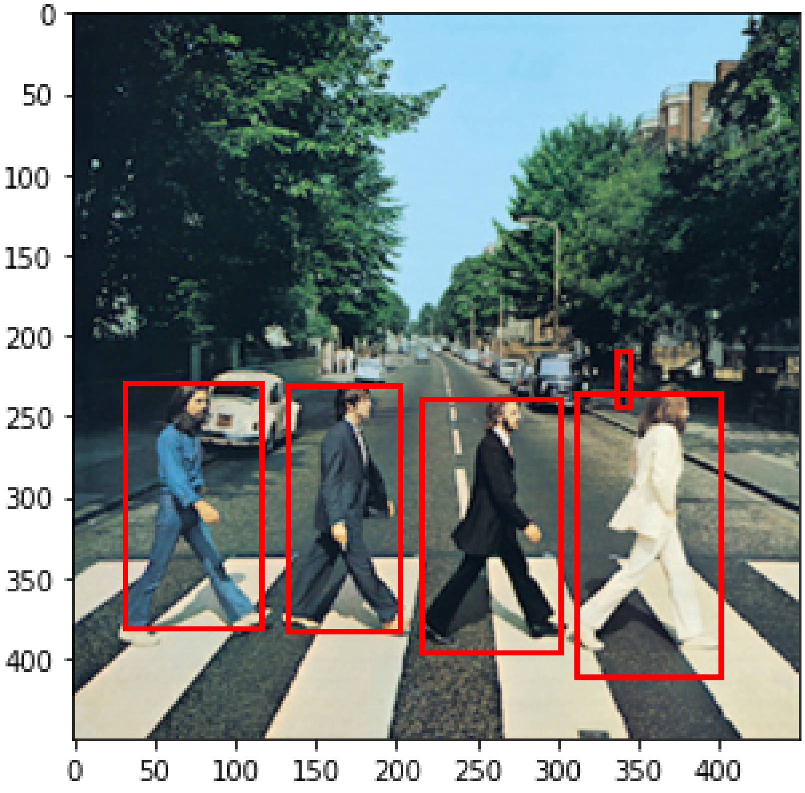
Draw the bounding box

Here we choose the top-5 scored box

*This “plot_img_bbox” function is refered to : https://www.kaggle.com/code/konstanter/fasterrcnn-pytorch-maskdetection

[89]:
```
import matplotlib.pyplot as plt
import matplotlib.patches as patches
# Function to visualize bounding boxes in the image

def plot_img_bbox(img, target):
    # plot the image and bboxes
    # Bounding boxes are defined as follows: x-min y-min width height
    fig, a = plt.subplots(1,1)
    fig.set_size_inches(5,5)
    a.imshow(img)
    bounding_Boxes = target['boxes'].cpu().numpy()
    for i in range(5):
        box = bounding_Boxes[i]
        x, y, width, height = box[0], box[1], box[2]-box[0], box[3]-box[1]
        rect = patches.Rectangle((x, y), width, height, linewidth = 2,␣
↪edgecolor = 'r', facecolor = 'none')

        # Draw the bounding box on top of the image
        a.add_patch(rect)
    plt.show()
```

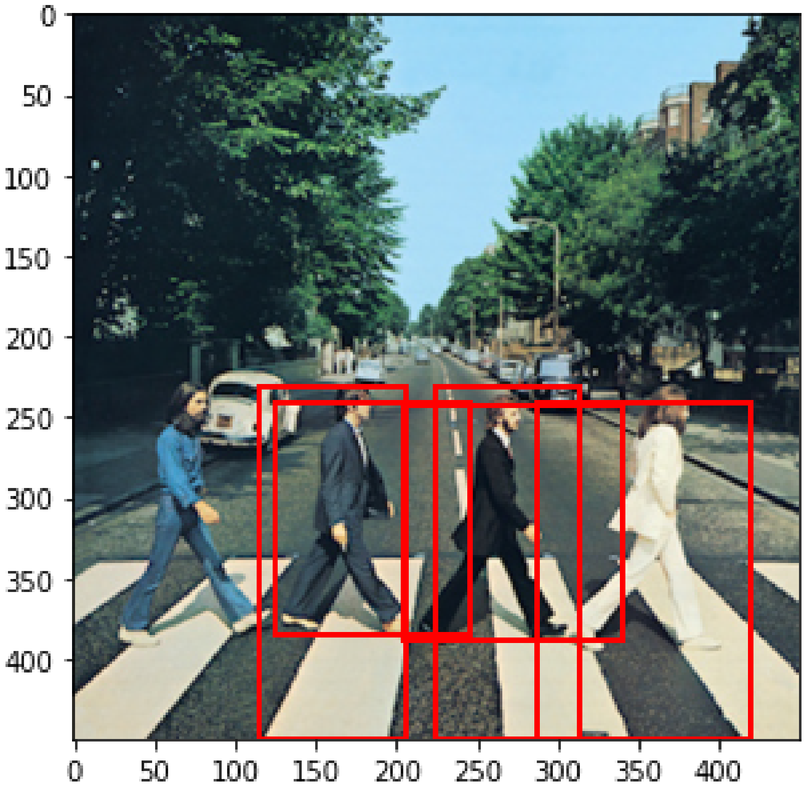
[93]:
```
plot_img_bbox(test_Beatles_Image, predicted_Output[0])
```



Validation of option 2 model:

The top-5 scored bounding boxes are:

[90]:
```
model_option_2.eval()
with torch.no_grad():
    predicted_Output_2 = model_option_2([test_Beatles_Image_Tensor.
↪to(device)])[0]

plot_img_bbox(test_Beatles_Image, predicted_Output_2)
```



So compared the top-5 scored bounding boxes of model1 and model2, we can notice that model 1 successfully select all 4 people occured in the picture, while model 2 missed the most left people in the picture.

Thus, on this Beatles image, model 1 performs better than model 2.

[86]:

```
predicted_Output_2
```

[86]:

```
{'boxes': tensor([[223.6211, 231.1144, 313.8610, 450.0000],
          [114.5353, 230.9814, 206.5275, 450.0000],
          [203.9372, 243.5188, 340.9922, 389.3011],
          [287.7523, 241.7243, 419.7959, 450.0000],
          [124.6823, 240.9502, 246.3496, 385.5459],
          [ 84.3286, 227.8522, 272.1978, 450.0000],
          [195.9931, 214.4947, 435.7801, 450.0000],
          [ 44.6954, 225.3032, 133.7435, 450.0000],
          [ 60.3157, 211.2322, 185.7823, 450.0000],
          [ 22.0326, 157.3285, 114.2121, 414.2127]], device='cuda:0'),
  'labels': tensor([1, 1, 1, 1, 1, 1, 1, 1, 1, 1], device='cuda:0'),
  'scores': tensor([0.5101, 0.4976, 0.4890, 0.4545, 0.3267, 0.2958, 0.2167,
 0.2154, 0.0886,
          0.0630], device='cuda:0')}
```