Assignment 3 : Multi-label Image Classification

In [0]:

```
from google.colab import drive
drive.mount('/content/gdrive')
```

```
Drive already mounted at /content/gdrive; to attempt to forcibly remount, call
drive.mount("/content/gdrive", force_remount=True).
```

In [0]:

```
import os
os.chdir("gdrive/My Drive/MP-1/MP-3")
```

In [0]:

```
!pip3 install torch torchvision
```

```
Requirement already satisfied: torch in /usr/local/lib/python3.6/dist-packages (0.4.1)
Requirement already satisfied: torchvision in /usr/local/lib/python3.6/dist-packages (0.2.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.6/dist-packages (from torchvision)
(1.14.6)
Requirement already satisfied: pillow>=4.1.1 in /usr/local/lib/python3.6/dist-packages (from
torchvision) (5.3.0)
Requirement already satisfied: six in /usr/local/lib/python3.6/dist-packages (from torchvision)
(1.11.0)
```

In [0]:

```
!pip install --no-cache-dir -I pillow
```

```
Collecting pillow
  Downloading
https://files.pythonhosted.org/packages/62/94/5430ebaa83f91cc7a9f687ff5238e26164a779cca2ef99032322
318/Pillow-5.3.0-cp36-cp36m-manylinux1_x86_64.whl (2.0MB)
    100% |████████████████████████████████| 2.0MB 18.0MB/s
Installing collected packages: pillow
Successfully installed pillow-5.3.0
```

In [0]:

```
!bash download_data.sh
```

In [0]:

```
import os
import numpy as np
import torch
import torch.nn as nn
from torchvision import transforms
from sklearn.metrics import average_precision_score
from PIL import Image, ImageDraw
import matplotlib.pyplot as plt
from kaggle_submission import output_submission_csv
#from classifier import Classifier
from voc_dataloader import VocDataset, VOC_CLASSES
import pickle

%matplotlib inline
%load_ext autoreload
%autoreload 2
```

In this assignment, you train a classifier to do multi-label classificaton on the PASCAL VOC 2007 dataset. The dataset has 20

different class which can appear in any given image. Your classifier will predict whether each class appears in an image. This task is slightly different from exclusive multiclass classification like the ImageNet competition where only a single most appropriate class is predicted for an image.

Reading Pascal Data

Loading Training Data

In the following cell we will load the training data and also apply some transforms to the data. Feel free to apply more transforms for data augmentation which can lead to better performance.

In [0]:

```
# Transforms applied to the training data
normalize = transforms.Normalize(mean=[0.485, 0.456, 0.406],
                                 std= [0.229, 0.224, 0.225])

train_transform = transforms.Compose([

            transforms.RandomHorizontalFlip(),
            #transforms.RandomRotation(45),
            transforms.RandomResizedCrop(224,scale=(0.08, 1.0), ratio=(0.75, 1.3333333333333333), i
nterpolation=2),

            transforms.Resize(227),
            transforms.CenterCrop(227),
            transforms.ToTensor(),
            normalize,


        ])
```

In [0]:

```
ds_train = VocDataset('VOCdevkit_2007/VOC2007/','train',train_transform)
```

Loading Validation Data

We will load the test data for the PASCAL VOC 2007 dataset. Do **NOT** add data augmentation transforms to validation data.

In [0]:

```
# Transforms applied to the testing data
test_transform = transforms.Compose([
            transforms.Resize(227),
            transforms.CenterCrop(227),
            transforms.ToTensor(),
            normalize,
        ])
```

In [0]:

```
ds_val = VocDataset('VOCdevkit_2007/VOC2007/','val',test_transform)
```

In [0]:

```
with open('ds_val.pkl', 'wb') as f:
    pickle.dump(ds_val, f)

with open('ds_train.pkl', 'wb') as f:
    pickle.dump(ds_train, f)
```

Loading data objects saved in pickle!

In [0]:

In [0]:

```
with open('ds_val.pkl', 'rb') as f:
    ds_val = pickle.load(f)

with open('ds_train.pkl', 'rb') as f:
    ds_train = pickle.load(f)
```

saving new transformation

In [0]:

```
ds_train.transform = train_transform
```

updating new transform object into pickle

In [0]:

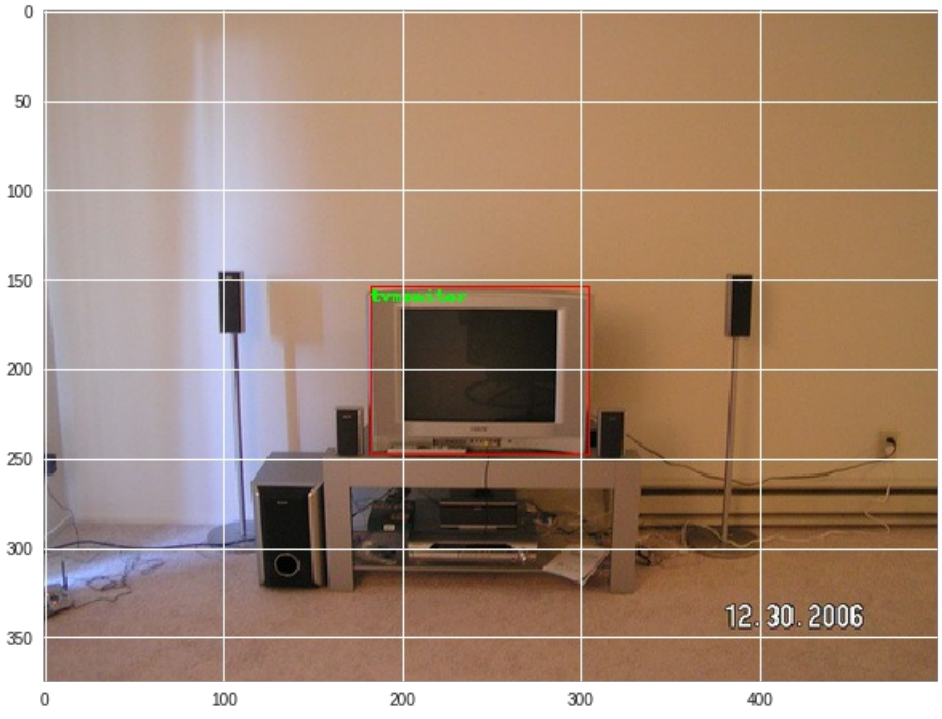
```
with open('ds_train.pkl', 'wb') as f:
    pickle.dump(ds_train, f)
```

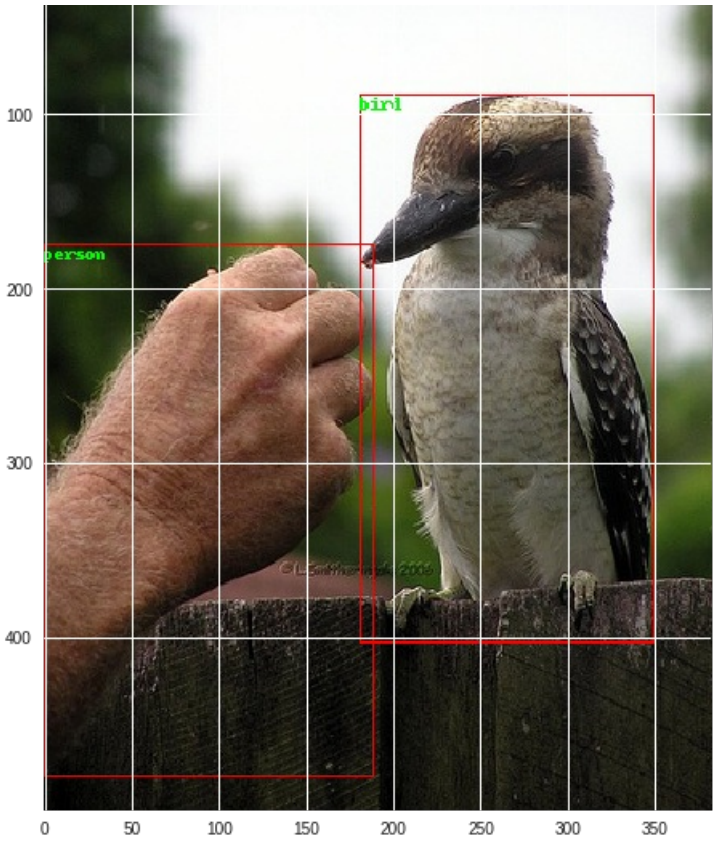
Visualizing the Data

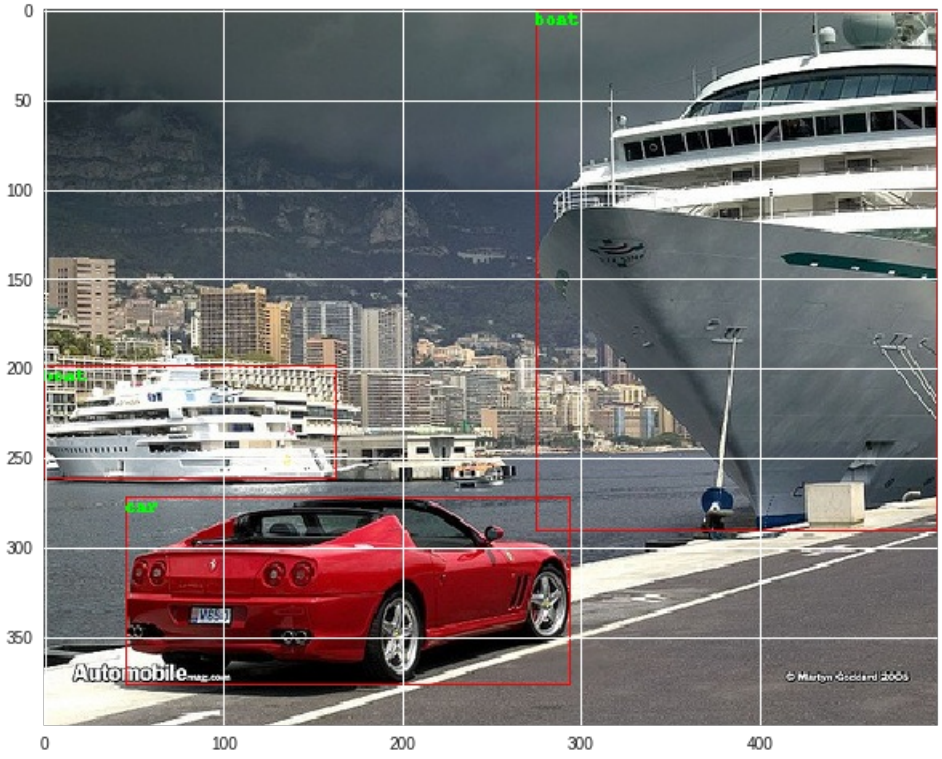
PASCAL VOC has bounding box annotations in addition to class labels. Use the following code to visualize some random examples and corresponding annotations from the train set.

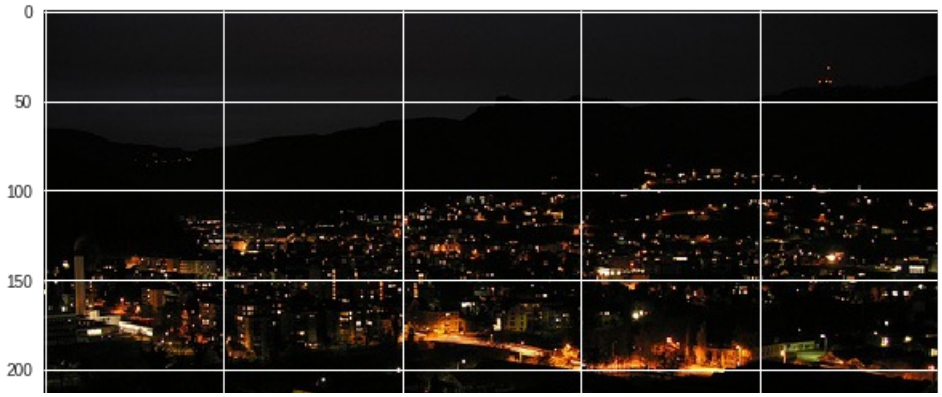
In [0]:

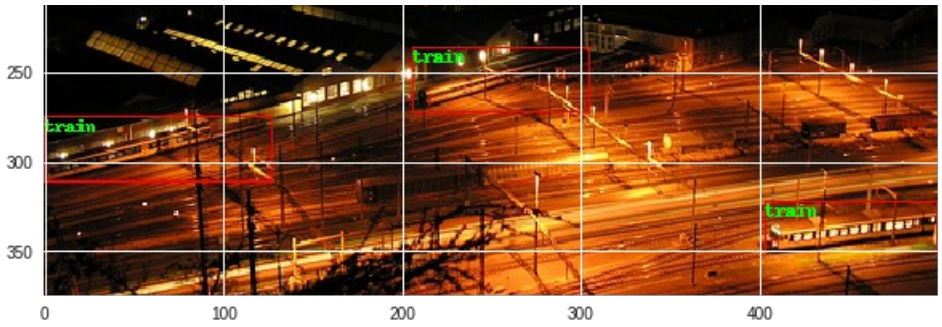
```
for i in range(5):
    idx = np.random.randint(0, len(ds_train.names)+1)
    _imgpath = os.path.join('VOCdevkit_2007/VOC2007/', 'JPEGImages', ds_train.names[idx]+'.jpg')
    img = Image.open(_imgpath).convert('RGB')
    draw = ImageDraw.Draw(img)
    for j in range(len(ds_train.box_indices[idx])):
        obj = ds_train.box_indices[idx][j]
        draw.rectangle(list(obj), outline=(255,0,0))
        draw.text(list(obj[0:2]), ds_train.classes[ds_train.label_order[idx][j]], fill=(0,255,0))
    plt.figure(figsize = (10,10))
    plt.imshow(np.array(img))
```

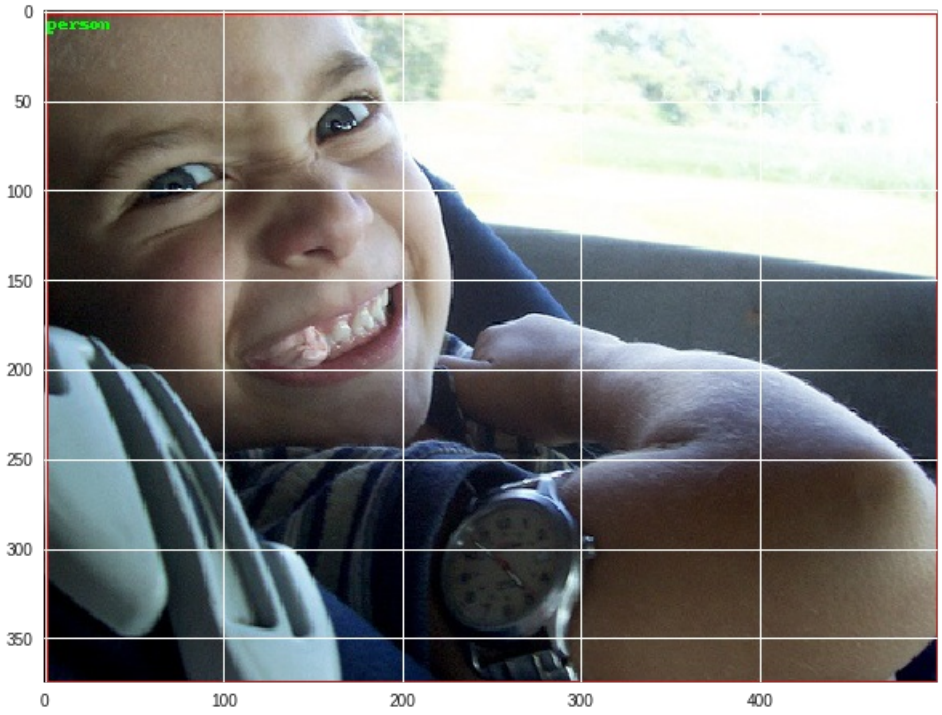



bird
person
0
50
100
150
200
250
300
350
100
200
300
400


boat
car
Automobile
0
100
200
300
400
0
50
100
150
200
250
300
350


0
50
100
150
200





Classification

In [0]:

```
device = torch.device("cuda:0" if torch.cuda.is_available() else "cpu")
```

In [0]:

```
train_loader = torch.utils.data.DataLoader(dataset=ds_train,
                                           batch_size=50,
                                           shuffle=True,
                                           num_workers=0)
```

In [0]:

```
val_loader = torch.utils.data.DataLoader(dataset=ds_val,
                                           batch_size=50,
                                           shuffle=True,
                                           num_workers=1)
```

In [0]:

```
def train_classifier(train_loader, classifier, criterion, optimizer):
    classifier.train()
    loss_ = 0.0
    losses = []
    for i, (images, labels, _) in enumerate(train_loader):
        images, labels = images.to(device), labels.to(device)
        optimizer.zero_grad()
        #print(images.shape)
```

```
        logits = classifier(images)
        loss = criterion(logits, labels)
        loss.backward()
        optimizer.step()
        losses.append(loss)
    return torch.stack(losses).mean().item()
```

In [0]:

```
def test_classifier(test_loader, classifier, criterion, print_ind_classes=True):
    classifier.eval()
    losses = []
    with torch.no_grad():
        y_true = np.zeros((0,21))
        y_score = np.zeros((0,21))
        for i, (images, labels, _) in enumerate(test_loader):
            images, labels = images.to(device), labels.to(device)
            logits = classifier(images)
            y_true = np.concatenate((y_true, labels.cpu().numpy()), axis=0)
            y_score = np.concatenate((y_score, logits.cpu().numpy()), axis=0)
            loss = criterion(logits, labels)
            losses.append(loss)
        aps = []
        # ignore first class which is background
        for i in range(1, y_true.shape[1]):
            ap = average_precision_score(y_true[:, i], y_score[:, i])
            if print_ind_classes:
                print('-------  Class: {:<12}     AP: {:>8.4f}  -------'.format(VOC_CLASSES[i], ap)
            aps.append(ap)

        mAP = np.mean(aps)
        test_loss = np.mean(losses)
        print('mAP: {0:.4f}'.format(mAP))
        print('Avg loss: {}'.format(test_loss))

    return mAP, test_loss, aps
```

Modifying the network

The network you are given as is will allow you to reach around 0.15-0.2 mAP. To meet the benchmark for this assignment you will need to improve the network. There are a variety of different approaches you should try:

- Network architecture changes
  - Number of layers: try adding layers to make your network deeper
  - Batch normalization: adding batch norm between layers will likely give you a significant performance increase
  - Residual connections: as you increase the depth of your network, you will find that having residual connections like those in ResNet architectures will be helpful
- Optimizer: Instead of plain SGD, you may want to add a learning rate schedule, add momentum, or use one of the other optimizers you have learned about like Adam. Check the `torch.optim` package for other optimizers
- Data augmentation: You should use the `torchvision.transforms` module to try adding random resized crops and horizontal flips of the input data. Check `transforms.RandomResizedCrop` and `transforms.RandomHorizontalFlip` for this
- Epochs: Once you have found a generally good hyperparameter setting try training for more epochs
- Loss function: You might want to add weighting to the `MultiLabelSoftMarginLoss` for classes that are less well represented or experiment with a different loss function

In [0]:

```
classifier = Classifier().to(device)
# You can can use this function to reload a network you have already saved previously
#classifier.load_state_dict(torch.load('voc_classifier.pth'))
```

In [0]:

```
import numpy as np
#arr=np.array([1,1,1,3,1,1,1,1,1,4,1,2,1,1,1,3,3,2,1,2,1])
#weight=torch.from_numpy(arr)
#weight=weight.to(device)

criterion = nn.MultiLabelSoftMarginLoss()
```

```
criterion = nn.MultiLabelSoftMarginLoss()
#optimizer = torch.optim.SGD(classifier.parameters(), lr=0.001, momentum=0.92)
optimizer =torch.optim.Adam(classifier.parameters(), lr=0.0001, betas=(0.9, 0.999))
```

In [0]:

```
# Training the Classifier
NUM_EPOCHS = 160
TEST_FREQUENCY = 20

for epoch in range(1, NUM_EPOCHS+1):
    print("Starting epoch number " + str(epoch))
    train_loss = train_classifier(train_loader, classifier, criterion, optimizer)
    print("Loss for Training on Epoch " +str(epoch) + " is "+ str(train_loss))
    if(epoch%TEST_FREQUENCY==0):
        mAP_val, val_loss, _ = test_classifier(val_loader, classifier, criterion)
        print('Evaluating classifier')
        print("Mean Precision Score for Testing on Epoch " +str(epoch) + " is "+ str(mAP_val))
```

```
Starting epoch number 1
Loss for Training on Epoch 1 is 0.38385292887687683
Starting epoch number 2
Loss for Training on Epoch 2 is 0.24156561493873596
Starting epoch number 3
Loss for Training on Epoch 3 is 0.23414763808250427
Starting epoch number 4
Loss for Training on Epoch 4 is 0.22995591163635254
Starting epoch number 5
Loss for Training on Epoch 5 is 0.2261081337928772
Starting epoch number 6
Loss for Training on Epoch 6 is 0.22265003621578217
Starting epoch number 7
Loss for Training on Epoch 7 is 0.21953028440475464
Starting epoch number 8
Loss for Training on Epoch 8 is 0.21649180352687836
Starting epoch number 9
Loss for Training on Epoch 9 is 0.21611854434013367
Starting epoch number 10
Loss for Training on Epoch 10 is 0.21153473854064941
Starting epoch number 11
Loss for Training on Epoch 11 is 0.21422407031059265
Starting epoch number 12
Loss for Training on Epoch 12 is 0.21198274195194244
Starting epoch number 13
Loss for Training on Epoch 13 is 0.20877470076084137
Starting epoch number 14
Loss for Training on Epoch 14 is 0.20792993903160095
Starting epoch number 15
Loss for Training on Epoch 15 is 0.20263825356960297
Starting epoch number 16
Loss for Training on Epoch 16 is 0.20670893788337708
Starting epoch number 17
Loss for Training on Epoch 17 is 0.2019513100385666
Starting epoch number 18
Loss for Training on Epoch 18 is 0.1999925971031189
Starting epoch number 19
Loss for Training on Epoch 19 is 0.20195026695728302
Starting epoch number 20
Loss for Training on Epoch 20 is 0.19764390587806702
-------  Class: aeroplane        AP:   0.4551  -------
-------  Class: bicycle          AP:   0.1495  -------
-------  Class: bird             AP:   0.1625  -------
-------  Class: boat             AP:   0.2465  -------
-------  Class: bottle           AP:   0.1039  -------
-------  Class: bus              AP:   0.1407  -------
-------  Class: car              AP:   0.4428  -------
-------  Class: cat              AP:   0.3232  -------
-------  Class: chair            AP:   0.3801  -------
-------  Class: cow              AP:   0.1171  -------
-------  Class: diningtable      AP:   0.2166  -------
-------  Class: dog              AP:   0.2380  -------
-------  Class: horse            AP:   0.1841  -------
-------  Class: motorbike        AP:   0.1367  -------
-------  Class: person           AP:   0.7349  -------
-------  Class: pottedplant      AP:   0.0840  -------
```

```
-------  Class: sheep            AP:   0.1160  -------
-------  Class: sofa             AP:   0.2341  -------
-------  Class: train            AP:   0.1683  -------
-------  Class: tvmonitor        AP:   0.2190  -------
mAP: 0.2427
Avg loss: 0.20746831595897675
Evaluating classifier
Mean Precision Score for Testing on Epoch 20 is 0.24265638576284307
Starting epoch number 21
Loss for Training on Epoch 21 is 0.19488359987735748
Starting epoch number 22
Loss for Training on Epoch 22 is 0.1984357088804245
Starting epoch number 23
Loss for Training on Epoch 23 is 0.19538426399230957
Starting epoch number 24
Loss for Training on Epoch 24 is 0.19278009235858917
Starting epoch number 25
Loss for Training on Epoch 25 is 0.19236013293266296
Starting epoch number 26
Loss for Training on Epoch 26 is 0.19009003043174744
Starting epoch number 27
Loss for Training on Epoch 27 is 0.19006142020225525
Starting epoch number 28
Loss for Training on Epoch 28 is 0.1866062879562378
Starting epoch number 29
Loss for Training on Epoch 29 is 0.18323169648647308
Starting epoch number 30
Loss for Training on Epoch 30 is 0.18761304020881653
Starting epoch number 31
Loss for Training on Epoch 31 is 0.18799152970314026
Starting epoch number 32
Loss for Training on Epoch 32 is 0.18708433210849762
Starting epoch number 33
Loss for Training on Epoch 33 is 0.18514224886894226
Starting epoch number 34
Loss for Training on Epoch 34 is 0.18272961676120758
Starting epoch number 35
Loss for Training on Epoch 35 is 0.1847359836101532
Starting epoch number 36
Loss for Training on Epoch 36 is 0.18451546132564545
Starting epoch number 37
Loss for Training on Epoch 37 is 0.1818249672651291
Starting epoch number 38
Loss for Training on Epoch 38 is 0.1801609992980957
Starting epoch number 39
Loss for Training on Epoch 39 is 0.17741984128952026
Starting epoch number 40
Loss for Training on Epoch 40 is 0.17924915254116058
-------  Class: aeroplane        AP:   0.5525  -------
-------  Class: bicycle          AP:   0.1795  -------
-------  Class: bird             AP:   0.2234  -------
-------  Class: boat             AP:   0.3542  -------
-------  Class: bottle           AP:   0.1362  -------
-------  Class: bus              AP:   0.1929  -------
-------  Class: car              AP:   0.5882  -------
-------  Class: cat              AP:   0.3785  -------
-------  Class: chair            AP:   0.3914  -------
-------  Class: cow              AP:   0.1538  -------
-------  Class: diningtable      AP:   0.2868  -------
-------  Class: dog              AP:   0.2856  -------
-------  Class: horse            AP:   0.3812  -------
-------  Class: motorbike        AP:   0.3179  -------
-------  Class: person           AP:   0.7387  -------
-------  Class: pottedplant      AP:   0.1518  -------
-------  Class: sheep            AP:   0.2083  -------
-------  Class: sofa             AP:   0.2399  -------
-------  Class: train            AP:   0.4200  -------
-------  Class: tvmonitor        AP:   0.2040  -------
mAP: 0.3192
Avg loss: 0.19188477098941803
Evaluating classifier
Mean Precision Score for Testing on Epoch 40 is 0.31924344041219366
Starting epoch number 41
Loss for Training on Epoch 41 is 0.1799895167350769
Starting epoch number 42
Loss for Training on Epoch 42 is 0.1777503937482834
Starting epoch number 43
```

Loss for Training on Epoch 43 is 0.17584370076656342
Starting epoch number 44
Loss for Training on Epoch 44 is 0.17545920610427856
Starting epoch number 45
Loss for Training on Epoch 45 is 0.18247787654399872
Starting epoch number 46
Loss for Training on Epoch 46 is 0.17807705700397491
Starting epoch number 47
Loss for Training on Epoch 47 is 0.17091944813728333
Starting epoch number 48
Loss for Training on Epoch 48 is 0.17246389389038086
Starting epoch number 49
Loss for Training on Epoch 49 is 0.17336353659629822
Starting epoch number 50
Loss for Training on Epoch 50 is 0.17120380699634552
Starting epoch number 51
Loss for Training on Epoch 51 is 0.17169666290283203
Starting epoch number 52
Loss for Training on Epoch 52 is 0.17053382098674774
Starting epoch number 53
Loss for Training on Epoch 53 is 0.17355826497077942
Starting epoch number 54
Loss for Training on Epoch 54 is 0.17192566394805908
Starting epoch number 55
Loss for Training on Epoch 55 is 0.17133034765720367
Starting epoch number 56
Loss for Training on Epoch 56 is 0.16610555350780487
Starting epoch number 57
Loss for Training on Epoch 57 is 0.16908134520053864
Starting epoch number 58
Loss for Training on Epoch 58 is 0.1676468700170517
Starting epoch number 59
Loss for Training on Epoch 59 is 0.17038646340370178
Starting epoch number 60
Loss for Training on Epoch 60 is 0.16419732570648193
------- Class: aeroplane AP: 0.6068 -------
------- Class: bicycle AP: 0.1277 -------
------- Class: bird AP: 0.2723 -------
------- Class: boat AP: 0.4373 -------
------- Class: bottle AP: 0.1366 -------
------- Class: bus AP: 0.1771 -------
------- Class: car AP: 0.5469 -------
------- Class: cat AP: 0.4010 -------
------- Class: chair AP: 0.4225 -------
------- Class: cow AP: 0.1448 -------
------- Class: diningtable AP: 0.3100 -------
------- Class: dog AP: 0.2858 -------
------- Class: horse AP: 0.4545 -------
------- Class: motorbike AP: 0.3567 -------
------- Class: person AP: 0.7870 -------
------- Class: pottedplant AP: 0.1804 -------
------- Class: sheep AP: 0.1724 -------
------- Class: sofa AP: 0.2753 -------
------- Class: train AP: 0.4651 -------
------- Class: tvmonitor AP: 0.2228 -------
mAP: 0.3391
Avg loss: 0.19850996136665344
Evaluating classifier
Mean Precision Score for Testing on Epoch 60 is 0.3391468675108501
Starting epoch number 61
Loss for Training on Epoch 61 is 0.1608818769454956
Starting epoch number 62
Loss for Training on Epoch 62 is 0.16274918615818024
Starting epoch number 63
Loss for Training on Epoch 63 is 0.16199974715709686
Starting epoch number 64
Loss for Training on Epoch 64 is 0.16202490031719208
Starting epoch number 65
Loss for Training on Epoch 65 is 0.1605551838874817
Starting epoch number 66
Loss for Training on Epoch 66 is 0.16217979788780212
Starting epoch number 67
Loss for Training on Epoch 67 is 0.16307853162288666
Starting epoch number 68
Loss for Training on Epoch 68 is 0.15830865502357483
Starting epoch number 69
Loss for Training on Epoch 69 is 0.1603788584470749

```
Starting epoch number 70
Loss for Training on Epoch 70 is 0.158578559756279
Starting epoch number 71
Loss for Training on Epoch 71 is 0.16084273159503937
Starting epoch number 72
Loss for Training on Epoch 72 is 0.15597794950008392
Starting epoch number 73
Loss for Training on Epoch 73 is 0.15829665958881378
Starting epoch number 74
Loss for Training on Epoch 74 is 0.1620354801416397
Starting epoch number 75
Loss for Training on Epoch 75 is 0.15851882100105286
Starting epoch number 76
Loss for Training on Epoch 76 is 0.15897907316684723
Starting epoch number 77
Loss for Training on Epoch 77 is 0.16004081070423126
Starting epoch number 78
Loss for Training on Epoch 78 is 0.1587800234556198
Starting epoch number 79
Loss for Training on Epoch 79 is 0.1630675047636032
Starting epoch number 80
Loss for Training on Epoch 80 is 0.1649705022573471
-------  Class: aeroplane        AP:   0.6041  -------
-------  Class: bicycle          AP:   0.4191  -------
-------  Class: bird             AP:   0.3901  -------
-------  Class: boat             AP:   0.4513  -------
-------  Class: bottle           AP:   0.1718  -------
-------  Class: bus              AP:   0.2995  -------
-------  Class: car              AP:   0.6507  -------
-------  Class: cat              AP:   0.4314  -------
-------  Class: chair            AP:   0.4823  -------
-------  Class: cow              AP:   0.1777  -------
-------  Class: diningtable      AP:   0.3631  -------
-------  Class: dog              AP:   0.3099  -------
-------  Class: horse            AP:   0.5514  -------
-------  Class: motorbike        AP:   0.4715  -------
-------  Class: person           AP:   0.7970  -------
-------  Class: pottedplant      AP:   0.1701  -------
-------  Class: sheep            AP:   0.1895  -------
-------  Class: sofa             AP:   0.3331  -------
-------  Class: train            AP:   0.6269  -------
-------  Class: tvmonitor        AP:   0.3288  -------
mAP: 0.4110
Avg loss: 0.16709783673286438
Evaluating classifier
Mean Precision Score for Testing on Epoch 80 is 0.41096631307612974
Starting epoch number 81
Loss for Training on Epoch 81 is 0.15545541048049927
Starting epoch number 82
Loss for Training on Epoch 82 is 0.15282070636749268
Starting epoch number 83
Loss for Training on Epoch 83 is 0.15575280785560608
Starting epoch number 84
Loss for Training on Epoch 84 is 0.15288926661014557
Starting epoch number 85
Loss for Training on Epoch 85 is 0.1538861244916916
Starting epoch number 86
Loss for Training on Epoch 86 is 0.15478140115737915
Starting epoch number 87
Loss for Training on Epoch 87 is 0.15645438432693481
Starting epoch number 88
Loss for Training on Epoch 88 is 0.15056195855140686
Starting epoch number 89
Loss for Training on Epoch 89 is 0.15211115777492523
Starting epoch number 90
Loss for Training on Epoch 90 is 0.15307262539863586
Starting epoch number 91
Loss for Training on Epoch 91 is 0.14994275569915771
Starting epoch number 92
Loss for Training on Epoch 92 is 0.14763766527175903
Starting epoch number 93
Loss for Training on Epoch 93 is 0.15007218718528748
Starting epoch number 94
Loss for Training on Epoch 94 is 0.14933981001377106
Starting epoch number 95
Loss for Training on Epoch 95 is 0.1495475023984909
Starting epoch number 96
```

```
Starting epoch number 96
Loss for Training on Epoch 96 is 0.14766447246074677
Starting epoch number 97
Loss for Training on Epoch 97 is 0.1497674435377121
Starting epoch number 98
Loss for Training on Epoch 98 is 0.14634442329406738
Starting epoch number 99
Loss for Training on Epoch 99 is 0.14532841742038727
Starting epoch number 100
Loss for Training on Epoch 100 is 0.14977754652500153
-------  Class: aeroplane        AP:   0.6251  -------
-------  Class: bicycle          AP:   0.3727  -------
-------  Class: bird             AP:   0.4098  -------
-------  Class: boat             AP:   0.4992  -------
-------  Class: bottle           AP:   0.1786  -------
-------  Class: bus              AP:   0.3148  -------
-------  Class: car              AP:   0.6670  -------
-------  Class: cat              AP:   0.4068  -------
-------  Class: chair            AP:   0.4646  -------
-------  Class: cow              AP:   0.1927  -------
-------  Class: diningtable      AP:   0.3380  -------
-------  Class: dog              AP:   0.3175  -------
-------  Class: horse            AP:   0.5806  -------
-------  Class: motorbike        AP:   0.4243  -------
-------  Class: person           AP:   0.8065  -------
-------  Class: pottedplant      AP:   0.1928  -------
-------  Class: sheep            AP:   0.1864  -------
-------  Class: sofa             AP:   0.3233  -------
-------  Class: train            AP:   0.6782  -------
-------  Class: tvmonitor        AP:   0.2823  -------
mAP: 0.4131
Avg loss: 0.17450839281082153
Evaluating classifier
Mean Precision Score for Testing on Epoch 100 is 0.4130648649112308
Starting epoch number 101
Loss for Training on Epoch 101 is 0.14703461527824402
Starting epoch number 102
Loss for Training on Epoch 102 is 0.14874564111232758
Starting epoch number 103
Loss for Training on Epoch 103 is 0.14995060861110687
Starting epoch number 104
Loss for Training on Epoch 104 is 0.14760740101337433
Starting epoch number 105
Loss for Training on Epoch 105 is 0.1484629511833191
Starting epoch number 106
Loss for Training on Epoch 106 is 0.14802870154380798
Starting epoch number 107
Loss for Training on Epoch 107 is 0.14465972781181335
Starting epoch number 108
Loss for Training on Epoch 108 is 0.14385518431663513
Starting epoch number 109
Loss for Training on Epoch 109 is 0.14015229046344757
Starting epoch number 110
Loss for Training on Epoch 110 is 0.14079758524894714
Starting epoch number 111
Loss for Training on Epoch 111 is 0.13926686346530914
Starting epoch number 112
Loss for Training on Epoch 112 is 0.14277470111846924
Starting epoch number 113
Loss for Training on Epoch 113 is 0.13925793766975403
Starting epoch number 114
Loss for Training on Epoch 114 is 0.13894888758659363
Starting epoch number 115
Loss for Training on Epoch 115 is 0.13727203011512756
Starting epoch number 116
Loss for Training on Epoch 116 is 0.13745437562465668
Starting epoch number 117
Loss for Training on Epoch 117 is 0.13910537958145142
Starting epoch number 118
Loss for Training on Epoch 118 is 0.14063622057437897
Starting epoch number 119
Loss for Training on Epoch 119 is 0.13987873494625092
Starting epoch number 120
Loss for Training on Epoch 120 is 0.14659708738327026
-------  Class: aeroplane        AP:   0.6527  -------
-------  Class: bicycle          AP:   0.4198  -------
-------  Class: bird             AP:   0.4318  -------
-------  Class: boat             AP:   0.5141  -------
```

```
-------  Class: boat             AP:   0.5141  -------
-------  Class: bottle           AP:   0.1823  -------
-------  Class: bus              AP:   0.4045  -------
-------  Class: car              AP:   0.6746  -------
-------  Class: cat              AP:   0.4210  -------
-------  Class: chair            AP:   0.5182  -------
-------  Class: cow              AP:   0.2190  -------
-------  Class: diningtable      AP:   0.4023  -------
-------  Class: dog              AP:   0.3482  -------
-------  Class: horse            AP:   0.5932  -------
-------  Class: motorbike        AP:   0.5758  -------
-------  Class: person           AP:   0.8114  -------
-------  Class: pottedplant      AP:   0.1979  -------
-------  Class: sheep            AP:   0.2071  -------
-------  Class: sofa             AP:   0.3562  -------
-------  Class: train            AP:   0.6335  -------
-------  Class: tvmonitor        AP:   0.3221  -------
mAP: 0.4443
Avg loss: 0.16878525912761688
Evaluating classifier
Mean Precision Score for Testing on Epoch 120 is 0.44428469202422327
Starting epoch number 121
Loss for Training on Epoch 121 is 0.14666177332401276
Starting epoch number 122
Loss for Training on Epoch 122 is 0.14224708080291748
Starting epoch number 123
Loss for Training on Epoch 123 is 0.13649596273899078
Starting epoch number 124
Loss for Training on Epoch 124 is 0.1372068226337433
Starting epoch number 125
Loss for Training on Epoch 125 is 0.13836292922496796
Starting epoch number 126
Loss for Training on Epoch 126 is 0.1384437531232834
Starting epoch number 127
Loss for Training on Epoch 127 is 0.13939239084720612
Starting epoch number 128
Loss for Training on Epoch 128 is 0.13615871965885162
Starting epoch number 129
Loss for Training on Epoch 129 is 0.13926725089550018
Starting epoch number 130
Loss for Training on Epoch 130 is 0.13693054020404816
Starting epoch number 131
Loss for Training on Epoch 131 is 0.1412055939435959
Starting epoch number 132
Loss for Training on Epoch 132 is 0.140755757689476
Starting epoch number 133
Loss for Training on Epoch 133 is 0.14216697216033936
Starting epoch number 134
Loss for Training on Epoch 134 is 0.1419692486524582
Starting epoch number 135
Loss for Training on Epoch 135 is 0.1417274922132492
Starting epoch number 136
Loss for Training on Epoch 136 is 0.13119544088840485
Starting epoch number 137
Loss for Training on Epoch 137 is 0.13721832633018494
Starting epoch number 138
Loss for Training on Epoch 138 is 0.13484768569469452
Starting epoch number 139
Loss for Training on Epoch 139 is 0.1341766119003296
Starting epoch number 140
Loss for Training on Epoch 140 is 0.13460035622119904
-------  Class: aeroplane        AP:   0.6402  -------
-------  Class: bicycle          AP:   0.4639  -------
-------  Class: bird             AP:   0.3887  -------
-------  Class: boat             AP:   0.4563  -------
-------  Class: bottle           AP:   0.1748  -------
-------  Class: bus              AP:   0.4263  -------
-------  Class: car              AP:   0.6542  -------
-------  Class: cat              AP:   0.4263  -------
-------  Class: chair            AP:   0.4874  -------
-------  Class: cow              AP:   0.2060  -------
-------  Class: diningtable      AP:   0.3885  -------
-------  Class: dog              AP:   0.3424  -------
-------  Class: horse            AP:   0.6279  -------
-------  Class: motorbike        AP:   0.5562  -------
-------  Class: person           AP:   0.8162  -------
-------  Class: pottedplant      AP:   0.2424  -------
-------  Class: sheep            AP:   0.1856  -------
```

```
-------  Class: sheep            AP:   0.1696  -------
-------  Class: sofa             AP:   0.3467  -------
-------  Class: train            AP:   0.6578  -------
-------  Class: tvmonitor        AP:   0.3311  -------
mAP: 0.4410
Avg loss: 0.17453141510486603
Evaluating classifier
Mean Precision Score for Testing on Epoch 140 is 0.44095223223215035
Starting epoch number 141
Loss for Training on Epoch 141 is 0.14334529638290405
Starting epoch number 142
Loss for Training on Epoch 142 is 0.1432388424873352
Starting epoch number 143
Loss for Training on Epoch 143 is 0.13403095304965973
Starting epoch number 144
Loss for Training on Epoch 144 is 0.12924863398075104
Starting epoch number 145
Loss for Training on Epoch 145 is 0.13110597431659698
Starting epoch number 146
Loss for Training on Epoch 146 is 0.13122375309467316
Starting epoch number 147
Loss for Training on Epoch 147 is 0.13466192781925201
Starting epoch number 148
Loss for Training on Epoch 148 is 0.13344396650791168
Starting epoch number 149
Loss for Training on Epoch 149 is 0.1255396157503128
Starting epoch number 150
Loss for Training on Epoch 150 is 0.12805941700935364
Starting epoch number 151
Loss for Training on Epoch 151 is 0.12832264602184296
Starting epoch number 152
Loss for Training on Epoch 152 is 0.13213442265987396
Starting epoch number 153
Loss for Training on Epoch 153 is 0.1343706250190735
Starting epoch number 154
Loss for Training on Epoch 154 is 0.1291821151971817
Starting epoch number 155
Loss for Training on Epoch 155 is 0.1294546127319336
Starting epoch number 156
Loss for Training on Epoch 156 is 0.12573006749153137
Starting epoch number 157
Loss for Training on Epoch 157 is 0.12180715799331665
Starting epoch number 158
Loss for Training on Epoch 158 is 0.12842108309268951
Starting epoch number 159
Loss for Training on Epoch 159 is 0.12860669195652008
Starting epoch number 160
Loss for Training on Epoch 160 is 0.1276235282421112
-------  Class: aeroplane        AP:   0.6507  -------
-------  Class: bicycle          AP:   0.4945  -------
-------  Class: bird             AP:   0.4569  -------
-------  Class: boat             AP:   0.4502  -------
-------  Class: bottle           AP:   0.1726  -------
-------  Class: bus              AP:   0.4076  -------
-------  Class: car              AP:   0.6372  -------
-------  Class: cat              AP:   0.4293  -------
-------  Class: chair            AP:   0.5074  -------
-------  Class: cow              AP:   0.2073  -------
-------  Class: diningtable      AP:   0.4145  -------
-------  Class: dog              AP:   0.3388  -------
-------  Class: horse            AP:   0.6250  -------
-------  Class: motorbike        AP:   0.5547  -------
-------  Class: person           AP:   0.8235  -------
-------  Class: pottedplant      AP:   0.2342  -------
-------  Class: sheep            AP:   0.2146  -------
-------  Class: sofa             AP:   0.3321  -------
-------  Class: train            AP:   0.6640  -------
-------  Class: tvmonitor        AP:   0.3520  -------
mAP: 0.4484
Avg loss: 0.18069632351398468
Evaluating classifier
Mean Precision Score for Testing on Epoch 160 is 0.4483512501245811
```

In [0]:

```
# Save the clssifier network
# Suggestion: you can save checkpoints of your network during training and reload them later
```

```
# Suggestion: You can save checkpoints of your network during training and reload them later
torch.save(classifier.state_dict(), './voc_classifier.pth')
```

In [0]:

```
print(classifier.state_dict())
```

In [0]:

```
from PIL import Image
def register_extension(id, extension):
    Image.EXTENSION[extension.lower()] = id.upper()
    Image.register_extension = register_extension
def register_extensions(id, extensions):
    for extension in extensions: register_extension(id, extension)
    Image.register_extensions = register_extensions
```

In [0]:

```
model=classifier.state_dict()
import copy
saved_trainer = copy.deepcopy(model)
```

In [0]:

```
with open("my_trainer_object.pkl", "wb") as output_file:
    pickle.dump(saved_trainer, output_file)
```

In [0]:

```
with open("my_trainer_object.pkl", "rb") as output_file:
    save=pickle.load(output_file)
```

In [0]:

```
classifier.load_state_dict(save)
```

In [0]:

```
print(classifier.state_dict())
```

```
OrderedDict([('conv1.weight', tensor([[[[-0.1117, -0.0383,  0.0684,  0.0720, -0.0262],
          [ 0.0078,  0.0090, -0.0697,  0.0699,  0.0398],
          [-0.0088, -0.1016,  0.0252, -0.0899,  0.0363],
          [-0.0366,  0.0949, -0.0356,  0.0668, -0.0041],
          [ 0.1052, -0.1277, -0.0693,  0.0860, -0.0291]],

         [[-0.1325,  0.0210,  0.0239, -0.0083, -0.1161],
          [ 0.0555, -0.0319, -0.1209,  0.0572,  0.1050],
          [-0.0648,  0.0540,  0.0229,  0.0453,  0.0617],
          [ 0.0773, -0.0152, -0.0646, -0.0707,  0.0854],
          [ 0.0098, -0.0726,  0.0754, -0.0134,  0.0997]],

         [[-0.1052,  0.0369,  0.0233,  0.0727, -0.0143],
          [ 0.0763, -0.0173,  0.0092,  0.0013,  0.0429],
          [-0.0337, -0.0533,  0.0319, -0.0662,  0.0623],
          [-0.0624, -0.0025, -0.0958,  0.0822,  0.1082],
          [-0.0852,  0.0996, -0.0935, -0.0849,  0.0323]]],


        [[[-0.0416,  0.0010,  0.0267, -0.0810, -0.0012],
          [ 0.0086,  0.0752,  0.0813, -0.0711,  0.1102],
          [-0.1137, -0.0021, -0.0228, -0.0501, -0.0295],
          [-0.0746, -0.0710,  0.0951,  0.1035, -0.0740],
          [-0.0737, -0.1071,  0.0047, -0.0243, -0.0388]],

         [[ 0.0093,  0.1223, -0.0154,  0.1012, -0.0338],
          [ 0.1070,  0.0383,  0.1284, -0.0143,  0.0890],
          [-0.0834, -0.0682, -0.0363, -0.0532,  0.0075],
          [ 0.0078, -0.0139,  0.0492,  0.0115, -0.0510],
```

```
  [-0.0395, -0.0346,  0.0939,  0.0836,  0.0451]],

 [[ 0.0846, -0.0461, -0.0194, -0.0226,  0.0745],
  [-0.0333, -0.0277,  0.0898, -0.0624, -0.0802],
  [-0.0310,  0.0668, -0.0869,  0.0536,  0.0770],
  [-0.0635, -0.0356,  0.0482,  0.0399,  0.0037],
  [-0.0424, -0.0400, -0.0220, -0.0206, -0.0895]]],


[[[-0.0135,  0.0615, -0.0656, -0.0045, -0.0999],
  [-0.0143,  0.0621,  0.0553,  0.0071,  0.0306],
  [ 0.1045, -0.0555,  0.0927,  0.0301,  0.0274],
  [-0.0221,  0.0630,  0.0025,  0.0137,  0.0676],
  [ 0.1115,  0.1037, -0.0736,  0.0321, -0.0942]],

 [[-0.1013, -0.0218, -0.0929,  0.0574, -0.0475],
  [-0.1089, -0.0951,  0.0673, -0.0964,  0.0480],
  [-0.0994, -0.0303, -0.0208, -0.0741, -0.0585],
  [ 0.0022,  0.0568,  0.0071, -0.0186, -0.0402],
  [-0.0000,  0.0438, -0.0908,  0.0912,  0.0357]],

 [[-0.0416, -0.0504,  0.0158, -0.0826,  0.1085],
  [ 0.0524, -0.0529, -0.0916, -0.0467, -0.1175],
  [-0.0559,  0.0397, -0.0268, -0.0239,  0.0047],
  [ 0.0832,  0.1108,  0.0809,  0.1205,  0.0592],
  [-0.0190, -0.0290,  0.0075, -0.0681,  0.0069]]],


...,


[[[ 0.0820,  0.1206, -0.0019, -0.0630, -0.0073],
  [-0.0369,  0.0872, -0.0166,  0.0385, -0.0998],
  [ 0.0299, -0.0947, -0.0038, -0.0883,  0.0437],
  [-0.0762,  0.0338, -0.0017, -0.0262, -0.1063],
  [-0.0750,  0.0328,  0.1014,  0.0228,  0.0714]],

 [[ 0.0086,  0.0295, -0.0363,  0.0085, -0.0351],
  [ 0.1097, -0.0565, -0.0394,  0.0111,  0.0212],
  [ 0.0441, -0.0530, -0.1128, -0.0164, -0.0925],
  [ 0.0706, -0.1073,  0.0108, -0.1250,  0.0077],
  [-0.0709,  0.0660, -0.0498, -0.0374, -0.0130]],

 [[ 0.0273,  0.0700, -0.0694,  0.0427,  0.0681],
  [-0.0201,  0.0625,  0.0061, -0.0602,  0.1000],
  [ 0.0660, -0.0439, -0.0666,  0.0786, -0.0986],
  [ 0.0361,  0.0039, -0.0084, -0.1055, -0.0265],
  [-0.0647,  0.1105, -0.0296,  0.0292, -0.0169]]],


[[[ 0.0070, -0.0554,  0.0251,  0.0367,  0.1141],
  [ 0.0093,  0.0111, -0.0709, -0.0829, -0.0384],
  [-0.0018, -0.0277, -0.0380,  0.1009,  0.0850],
  [-0.0206, -0.0645, -0.1065, -0.0550,  0.0105],
  [ 0.0639,  0.0793,  0.0519, -0.0561,  0.0520]],

 [[ 0.0145,  0.0419,  0.0929,  0.0819,  0.0458],
  [-0.0417,  0.0155, -0.0961, -0.0081, -0.0893],
  [-0.0109,  0.0876, -0.1124, -0.1325,  0.0590],
  [-0.0550, -0.1237, -0.0761, -0.1295, -0.0175],
  [ 0.0988, -0.1085, -0.0082, -0.0837, -0.0877]],

 [[ 0.1108, -0.0941, -0.0810,  0.0369, -0.0765],
  [-0.0814,  0.0906, -0.0735, -0.0814, -0.0469],
  [-0.0437,  0.0273,  0.0512, -0.0761,  0.0389],
  [ 0.0016,  0.0777, -0.0639,  0.0555,  0.0711],
  [ 0.0536,  0.0015,  0.0252,  0.0246,  0.1154]]],


[[[ 0.0023, -0.0562,  0.0702, -0.1079, -0.0474],
  [ 0.0234,  0.0029,  0.0860, -0.0004, -0.0883],
  [ 0.0099, -0.0417, -0.0609,  0.0539,  0.0233],
  [ 0.0751,  0.0253, -0.0175, -0.0415, -0.0891],
  [-0.0850, -0.0678,  0.0745,  0.0597, -0.0410]],

 [[-0.0448, -0.0602,  0.0237, -0.1277,  0.0539],
  [-0.0969, -0.0505,  0.0097,  0.0986, -0.0480],
```

```
          [-0.0714, -0.0512,  0.0332,  0.0911,  0.0482],
          [-0.1053,  0.0511,  0.1073,  0.0530, -0.0905],
          [-0.0087,  0.1234,  0.0658,  0.1136,  0.0553]],

         [[-0.0359,  0.0505, -0.0456, -0.0854, -0.0548],
          [-0.0805, -0.0040,  0.0628,  0.0100, -0.0233],
          [ 0.0727, -0.0192,  0.0238, -0.1240,  0.0483],
          [ 0.0052,  0.1049,  0.0688,  0.0626, -0.0492],
          [ 0.0811, -0.0230, -0.0080, -0.0862,  0.0499]]]], device='cuda:0')), ('conv1.bias', tensor([ 0.0381,  0.0611, -0.0277, -0.0595,  0.0988, -0.0596, -0.0686, -0.0568,
        -0.0863, -0.0678,  0.0456, -0.0050,  0.0717,  0.0757,  0.0874, -0.0907,
        -0.0931,  0.0101,  0.0144,  0.0931,  0.1148, -0.0413,  0.0245,  0.0745,
         0.0565, -0.0101,  0.0014, -0.0643,  0.0633,  0.1015, -0.1070,  0.0911],
       device='cuda:0')), ('conv1_bn.weight', tensor([0.5735, 0.3667, 0.4259, 0.7143, 0.2196, 0.7226, 0.9366, 0.7213, 0.7883,
        0.7855, 0.7495, 0.7693, 0.1116, 0.4845, 0.7984, 0.5589, 0.6922, 0.7276,
        0.2417, 0.7547, 0.6929, 0.9592, 0.0954, 0.0442, 0.7201, 0.1408, 0.0648,
        0.6891, 0.2569, 0.0832, 0.0392, 0.0401], device='cuda:0')), ('conv1_bn.bias', tensor([ 0.0387, -0.0468,  0.0004,  0.0342, -0.0048, -0.0013,  0.0847, -0.0528,
         0.0138, -0.0294, -0.0246,  0.0052, -0.0431,  0.0414, -0.0181,  0.0479,
        -0.0107,  0.0388,  0.0517, -0.0214,  0.0084, -0.0466, -0.0126, -0.0122,
         0.0017, -0.0071,  0.0038,  0.0297,  0.0320, -0.0416, -0.0268,  0.0024],
       device='cuda:0')), ('conv1_bn.running_mean', tensor([ 0.0398,  0.0825, -0.0025, -0.0224,  0.1341, -0.0632, -0.0684, -0.1408,
        -0.0324, -0.0220,  0.0363,  0.0011, -0.2769,  0.0637,  0.0138, -0.0916,
        -0.1751,  0.0086,  0.0247,  0.1880,  0.1747,  0.0095, -0.2763,  0.1038,
        -0.0197, -0.0124,  0.0186, -0.0643,  0.0779,  0.2000,  0.0197,  0.1042],
       device='cuda:0')), ('conv1_bn.running_var', tensor([0.0554, 0.0796, 0.1485, 0.0924, 0.0789, 0.1132, 0.0158, 0.8077, 0.1170,
        0.1024, 0.2274, 0.0788, 2.8782, 0.0298, 0.1518, 0.0168, 0.2631, 0.0287,
        0.0569, 0.3429, 0.2245, 0.1531, 1.9393, 0.2293, 0.2023, 0.0830, 0.0782,
        0.0113, 0.0438, 0.2454, 0.4163, 0.0707], device='cuda:0')),
('conv1_bn.num_batches_tracked', tensor(8160, device='cuda:0')), ('conv2.weight', tensor([[[[-0.0083, -0.0250,  0.0285,  0.0206, -0.0149],
          [ 0.0187, -0.0410, -0.0091,  0.0340, -0.0159],
          [-0.0329, -0.0098,  0.0342,  0.0132,  0.0164],
          [-0.0205, -0.0003, -0.0172, -0.0143,  0.0175],
          [-0.0011, -0.0414,  0.0331,  0.0198, -0.0290]],

         [[ 0.0314, -0.0223, -0.0047, -0.0096,  0.0068],
          [ 0.0078,  0.0298,  0.0324, -0.0263, -0.0116],
          [ 0.0043,  0.0057, -0.0124, -0.0031,  0.0185],
          [ 0.0011, -0.0055, -0.0221,  0.0291,  0.0196],
          [-0.0329,  0.0045, -0.0237, -0.0118,  0.0015]],

         [[ 0.0425,  0.0347, -0.0129,  0.0020, -0.0056],
          [ 0.0116, -0.0132, -0.0056, -0.0080, -0.0021],
          [ 0.0019,  0.0365,  0.0209, -0.0136, -0.0012],
          [ 0.0235,  0.0306, -0.0063,  0.0371, -0.0028],
          [-0.0076,  0.0309, -0.0074, -0.0306, -0.0256]],

         ...,

         [[ 0.0399, -0.0175,  0.0068,  0.0445,  0.0216],
          [ 0.0413,  0.0127, -0.0202,  0.0260,  0.0353],
          [ 0.0298,  0.0240,  0.0404, -0.0269, -0.0024],
          [-0.0055, -0.0272,  0.0149, -0.0210, -0.0233],
          [ 0.0161,  0.0148,  0.0033,  0.0334,  0.0348]],

         [[ 0.0319, -0.0038,  0.0378, -0.0170,  0.0383],
          [ 0.0344,  0.0062,  0.0081,  0.0359,  0.0444],
          [ 0.0024,  0.0156,  0.0074, -0.0048,  0.0237],
          [ 0.0037,  0.0303, -0.0252,  0.0117, -0.0267],
          [-0.0221, -0.0295,  0.0153,  0.0251,  0.0084]],

         [[ 0.0367,  0.0180,  0.0283,  0.0100, -0.0217],
          [-0.0134,  0.0011,  0.0037,  0.0116,  0.0279],
          [ 0.0022,  0.0067,  0.0143,  0.0233,  0.0077],
          [-0.0046, -0.0088, -0.0321,  0.0031, -0.0163],
          [ 0.0328,  0.0038,  0.0167, -0.0198,  0.0053]]],


        [[[ 0.0132,  0.0144, -0.0462, -0.0314, -0.0458],
          [-0.0135, -0.0017,  0.0165, -0.0168, -0.0234],
          [-0.0332, -0.0130,  0.0175,  0.0349,  0.0255],
          [ 0.0068,  0.0496,  0.0299,  0.0279,  0.0274],
          [ 0.0116,  0.0414,  0.0244, -0.0045, -0.0305]],
```

```
  [ ...,  ...,  ...,  ...,  ...]],

 [[-0.0372,  0.0233, -0.0272, -0.0269,  0.0187],
  [ 0.0225, -0.0228,  0.0138, -0.0257,  0.0227],
  [-0.0117,  0.0021,  0.0036,  0.0031, -0.0121],
  [ 0.0061,  0.0088, -0.0045,  0.0172,  0.0054],
  [ 0.0138,  0.0011,  0.0132, -0.0143,  0.0292]],

 [[ 0.0137, -0.0149, -0.0187,  0.0136, -0.0200],
  [ 0.0095,  0.0338,  0.0271,  0.0277,  0.0073],
  [-0.0189, -0.0139, -0.0277, -0.0065,  0.0099],
  [-0.0145,  0.0182, -0.0425, -0.0493, -0.0394],
  [-0.0158, -0.0187, -0.0230, -0.0505, -0.0387]],

 ...,

 [[ 0.0252, -0.0221, -0.0322,  0.0008, -0.0037],
  [-0.0023,  0.0096,  0.0226, -0.0021,  0.0175],
  [-0.0228, -0.0321,  0.0075, -0.0136, -0.0436],
  [-0.0056, -0.0041,  0.0173, -0.0285,  0.0242],
  [ 0.0428,  0.0433, -0.0139, -0.0072, -0.0257]],

 [[ 0.0291, -0.0453, -0.0269, -0.0204,  0.0012],
  [ 0.0099,  0.0222, -0.0401, -0.0067,  0.0049],
  [ 0.0019, -0.0093, -0.0356, -0.0365, -0.0045],
  [ 0.0095,  0.0071,  0.0086,  0.0278, -0.0033],
  [ 0.0540,  0.0266, -0.0148,  0.0357,  0.0008]],

 [[ 0.0032,  0.0030,  0.0042, -0.0015, -0.0300],
  [-0.0242,  0.0257,  0.0168,  0.0363,  0.0002],
  [-0.0314,  0.0392,  0.0165,  0.0489,  0.0194],
  [-0.0083, -0.0063, -0.0124,  0.0505, -0.0148],
  [ 0.0390,  0.0087,  0.0485,  0.0332,  0.0447]]],


[[[-0.0055,  0.0231,  0.0389, -0.0202,  0.0092],
  [-0.0217,  0.0187, -0.0366,  0.0213, -0.0292],
  [ 0.0095, -0.0337, -0.0468,  0.0387,  0.0169],
  [-0.0497, -0.0328,  0.0038,  0.0139, -0.0230],
  [-0.0332, -0.0233, -0.0394,  0.0241,  0.0256]],

 [[ 0.0228, -0.0098,  0.0005,  0.0319,  0.0349],
  [-0.0163,  0.0216, -0.0015, -0.0115,  0.0261],
  [-0.0178, -0.0262,  0.0213, -0.0149, -0.0283],
  [ 0.0195, -0.0162, -0.0104, -0.0309,  0.0003],
  [-0.0365, -0.0204, -0.0288, -0.0210,  0.0133]],

 [[-0.0098,  0.0196, -0.0345,  0.0081,  0.0107],
  [ 0.0043, -0.0536, -0.0429, -0.0517,  0.0033],
  [-0.0317, -0.0588, -0.0158, -0.0528,  0.0259],
  [-0.0431, -0.0058, -0.0170,  0.0079,  0.0083],
  [-0.0123,  0.0058, -0.0329,  0.0147,  0.0324]],

 ...,

 [[-0.0025,  0.0248,  0.0291, -0.0167, -0.0228],
  [ 0.0172, -0.0258,  0.0247,  0.0264,  0.0060],
  [ 0.0093, -0.0048, -0.0322, -0.0100, -0.0103],
  [-0.0114, -0.0010,  0.0349, -0.0070, -0.0267],
  [-0.0147, -0.0346, -0.0188, -0.0086,  0.0128]],

 [[-0.0101, -0.0123, -0.0120, -0.0284, -0.0253],
  [-0.0120, -0.0017,  0.0194, -0.0150,  0.0230],
  [-0.0065, -0.0375, -0.0222,  0.0330,  0.0170],
  [-0.0159, -0.0131,  0.0157,  0.0161,  0.0293],
  [ 0.0127, -0.0047,  0.0246,  0.0019,  0.0001]],

 [[ 0.0048, -0.0223, -0.0173, -0.0134,  0.0141],
  [-0.0039,  0.0163,  0.0178, -0.0114, -0.0380],
  [-0.0217, -0.0059, -0.0376, -0.0230,  0.0162],
  [ 0.0292,  0.0192, -0.0240, -0.0135, -0.0201],
  [ 0.0121, -0.0166, -0.0426, -0.0362, -0.0204]]],


...,


[[[-0.0230,  0.0353,  0.0245, -0.0355,  0.0198],
```

```
[[ 0.0256,  0.0333,  0.0213,  0.0333,  0.0196],
  [-0.0417,  0.0130, -0.0241,  0.0270,  0.0352],
  [ 0.0059, -0.0440, -0.0204,  0.0021, -0.0313],
  [-0.0111,  0.0126, -0.0043, -0.0035,  0.0248],
  [-0.0161, -0.0133, -0.0014, -0.0403, -0.0419]],

 [[-0.0040,  0.0266, -0.0203,  0.0048, -0.0027],
  [-0.0214, -0.0276, -0.0010,  0.0002, -0.0286],
  [-0.0099, -0.0240,  0.0146, -0.0282, -0.0196],
  [ 0.0095,  0.0190,  0.0271, -0.0262, -0.0240],
  [ 0.0031,  0.0218, -0.0182,  0.0096,  0.0308]],

 [[ 0.0315, -0.0049, -0.0095, -0.0232, -0.0373],
  [-0.0104, -0.0330,  0.0051, -0.0140, -0.0241],
  [ 0.0307,  0.0322, -0.0046,  0.0097,  0.0036],
  [ 0.0170,  0.0231, -0.0282,  0.0192, -0.0078],
  [ 0.0133, -0.0308, -0.0031,  0.0197, -0.0160]],

 ...,

 [[-0.0058, -0.0134,  0.0005, -0.0294,  0.0367],
  [ 0.0042,  0.0024, -0.0115, -0.0244,  0.0113],
  [-0.0180,  0.0058, -0.0151, -0.0024,  0.0030],
  [ 0.0013, -0.0402, -0.0106, -0.0130, -0.0409],
  [-0.0303, -0.0179, -0.0230, -0.0199,  0.0118]],

 [[ 0.0306,  0.0183, -0.0307, -0.0130, -0.0256],
  [-0.0060, -0.0085, -0.0283, -0.0027, -0.0158],
  [-0.0301, -0.0088, -0.0152, -0.0142,  0.0003],
  [ 0.0135, -0.0311, -0.0471, -0.0020, -0.0141],
  [-0.0031,  0.0151, -0.0207,  0.0060, -0.0298]],

 [[ 0.0073, -0.0142,  0.0339, -0.0308,  0.0175],
  [ 0.0138, -0.0298,  0.0198, -0.0122,  0.0167],
  [-0.0220, -0.0040, -0.0013,  0.0094,  0.0045],
  [ 0.0008, -0.0123, -0.0367,  0.0097, -0.0053],
  [ 0.0050, -0.0334, -0.0442, -0.0198, -0.0224]]],


[[[ 0.0052, -0.0133,  0.0383,  0.0268, -0.0207],
  [-0.0109,  0.0334, -0.0064, -0.0233, -0.0254],
  [ 0.0225, -0.0300,  0.0067, -0.0227,  0.0151],
  [-0.0244, -0.0208, -0.0121, -0.0190, -0.0131],
  [-0.0180,  0.0066, -0.0154,  0.0132, -0.0042]],

 [[ 0.0145,  0.0060,  0.0036,  0.0224, -0.0347],
  [-0.0319,  0.0279, -0.0005,  0.0083,  0.0031],
  [-0.0302, -0.0415, -0.0219,  0.0072,  0.0102],
  [ 0.0001, -0.0184, -0.0187,  0.0096,  0.0070],
  [-0.0402, -0.0506,  0.0017,  0.0166, -0.0360]],

 [[-0.0359, -0.0454, -0.0297,  0.0332, -0.0072],
  [ 0.0039, -0.0386, -0.0396,  0.0048,  0.0218],
  [ 0.0320, -0.0281, -0.0100, -0.0125, -0.0174],
  [ 0.0346,  0.0173,  0.0312,  0.0047,  0.0005],
  [ 0.0383,  0.0317, -0.0237, -0.0004,  0.0402]],

 ...,

 [[ 0.0312, -0.0124, -0.0116, -0.0183, -0.0351],
  [ 0.0028, -0.0034,  0.0065, -0.0384, -0.0000],
  [ 0.0269, -0.0400, -0.0412, -0.0032,  0.0156],
  [-0.0225,  0.0010, -0.0293,  0.0092, -0.0027],
  [-0.0160,  0.0141, -0.0524, -0.0447, -0.0124]],

 [[-0.0095, -0.0063, -0.0094, -0.0002, -0.0105],
  [-0.0401,  0.0313, -0.0184, -0.0220, -0.0029],
  [ 0.0188,  0.0159,  0.0071, -0.0206,  0.0142],
  [-0.0395, -0.0222, -0.0118, -0.0432, -0.0214],
  [-0.0376, -0.0028, -0.0149, -0.0146, -0.0507]],

 [[ 0.0003, -0.0256, -0.0141,  0.0434, -0.0086],
  [-0.0065, -0.0179, -0.0144,  0.0269, -0.0165],
  [-0.0184,  0.0217, -0.0011,  0.0031,  0.0304],
  [-0.0045,  0.0004, -0.0245, -0.0229,  0.0147],
  [ 0.0264, -0.0240, -0.0155,  0.0282, -0.0157]]],
```

```
        [[[ 0.0348,  0.0469,  0.0102, -0.0079, -0.0050],
          [ 0.0163,  0.0442,  0.0231, -0.0314,  0.0125],
          [-0.0059,  0.0420,  0.0450, -0.0173, -0.0123],
          [ 0.0004,  0.0310,  0.0389,  0.0153, -0.0154],
          [-0.0111,  0.0283,  0.0077,  0.0244, -0.0117]],

         [[ 0.0227, -0.0097,  0.0291,  0.0210,  0.0159],
          [-0.0047,  0.0306, -0.0007,  0.0123,  0.0155],
          [ 0.0429,  0.0139,  0.0327, -0.0281, -0.0265],
          [ 0.0364,  0.0215, -0.0124,  0.0204,  0.0041],
          [ 0.0115, -0.0217, -0.0294, -0.0080,  0.0030]],

         [[-0.0013,  0.0192,  0.0154,  0.0073,  0.0036],
          [ 0.0001, -0.0172,  0.0130, -0.0023,  0.0120],
          [ 0.0453,  0.0326,  0.0516,  0.0434,  0.0321],
          [-0.0201, -0.0010,  0.0266,  0.0522,  0.0436],
          [-0.0092, -0.0084,  0.0452,  0.0414,  0.0139]],

         ...,

         [[-0.0216,  0.0256,  0.0030, -0.0008, -0.0178],
          [ 0.0058, -0.0040, -0.0145,  0.0061,  0.0163],
          [-0.0241, -0.0108, -0.0393,  0.0026, -0.0183],
          [-0.0096, -0.0306, -0.0229,  0.0197,  0.0085],
          [-0.0036, -0.0295,  0.0278, -0.0027,  0.0089]],

         [[ 0.0244,  0.0283,  0.0348, -0.0269,  0.0091],
          [-0.0062, -0.0192,  0.0047, -0.0219,  0.0208],
          [ 0.0084,  0.0429,  0.0182,  0.0205,  0.0107],
          [ 0.0215,  0.0331,  0.0354, -0.0293,  0.0048],
          [-0.0073, -0.0085,  0.0077,  0.0259, -0.0467]],

         [[ 0.0191,  0.0261, -0.0027, -0.0118,  0.0252],
          [ 0.0208, -0.0120,  0.0036, -0.0219, -0.0111],
          [-0.0221,  0.0234, -0.0011, -0.0229, -0.0170],
          [ 0.0154, -0.0207, -0.0079,  0.0350,  0.0223],
          [-0.0114, -0.0279, -0.0005,  0.0084, -0.0002]]]], device='cuda:0')), ('conv2.bias', tensor([-0.0329, -0.0190, -0.0035, -0.0087,  0.0272, -0.0122, -0.0171,  0.0028,
        -0.0249,  0.0112,  0.0042,  0.0278,  0.0267, -0.0352, -0.0001, -0.0211,
         0.0071,  0.0070,  0.0168,  0.0310, -0.0164,  0.0264, -0.0109, -0.0072,
        -0.0079, -0.0058, -0.0032,  0.0136, -0.0012,  0.0090, -0.0129, -0.0193,
         0.0063,  0.0100, -0.0279, -0.0278, -0.0051, -0.0141,  0.0153, -0.0010,
        -0.0221, -0.0258, -0.0253,  0.0165,  0.0190, -0.0308, -0.0158,  0.0201,
         0.0138,  0.0085,  0.0034,  0.0115, -0.0149, -0.0090,  0.0285,  0.0153,
         0.0003,  0.0059, -0.0336,  0.0040,  0.0286,  0.0300,  0.0222, -0.0296],
       device='cuda:0')), ('conv2_bn.weight', tensor([0.5243, 0.9142, 0.6366, 0.2765, 0.6375, 0.7431, 0.0798, 0.4566, 0.1851,
        0.2931, 0.6286, 0.4646, 0.8127, 0.5795, 0.8807, 0.5380, 0.7072, 0.4262,
        0.0377, 0.1207, 0.7565, 0.7231, 0.7009, 0.0495, 0.5928, 0.5178, 0.7923,
        0.2856, 0.5032, 0.6817, 0.4434, 0.8622, 0.6865, 0.2926, 0.3185, 0.5599,
        0.6613, 0.6795, 0.8405, 0.2020, 0.1078, 0.4785, 0.1885, 0.1268, 0.6912,
        0.0676, 0.0691, 0.9399, 0.2217, 0.7959, 0.8452, 0.5605, 0.1397, 0.4489,
        0.4939, 0.7077, 0.4050, 0.7349, 0.3480, 0.8540, 0.6507, 0.6619, 0.0692,
        0.3360], device='cuda:0')), ('conv2_bn.bias', tensor([-0.0223, -0.0178, -0.0165, -0.0180, -0.0314, -0.0181, -0.0233, -0.0064,
        -0.0173, -0.0221, -0.0160, -0.0198, -0.0095, -0.0415, -0.0202, -0.0134,
        -0.0411, -0.0053, -0.0070, -0.0118, -0.0140, -0.0499, -0.0212, -0.0059,
        -0.0236, -0.0355, -0.0427, -0.0043, -0.0301,  0.0132, -0.0065, -0.0136,
        -0.0329, -0.0164, -0.0357, -0.0226, -0.0402, -0.0114, -0.0353, -0.0212,
        -0.0101, -0.0326, -0.0197,  0.0001, -0.0111, -0.0084, -0.0059, -0.0365,
         0.0150, -0.0175, -0.0193, -0.0210, -0.0180, -0.0192, -0.0194, -0.0222,
        -0.0244, -0.0437, -0.0070, -0.0387, -0.0202, -0.0186, -0.0237, -0.0304],
       device='cuda:0')), ('conv2_bn.running_mean', tensor([-0.2958, -0.2646, -0.3654, -0.5392,  0.3974,  0.0065,  0.7984, -0.2033,
         0.3217, -0.3061, -0.2939, -0.2597, -0.1432,  0.0804, -0.3012, -0.1514,
         0.1360, -0.3587, -0.3173, -0.0089, -0.5034,  0.1930, -0.3156,  0.0664,
        -0.4284, -0.3035, -0.2120, -0.4497,  0.5616, -0.2300, -0.0835, -0.3604,
        -0.2400, -0.4203,  0.4153,  0.4558, -0.4340, -0.4993, -0.0171, -0.3412,
        -0.2523, -0.7652, -0.1984, -0.2426, -0.5655, -0.0508, -0.4544,  0.0229,
        -0.2810, -0.2331, -0.2479, -0.2308, -0.1414, -0.6321, -0.4323, -0.4556,
         0.1658,  0.4135, -0.5175, -0.3011, -0.1338, -0.2105,  0.4852,  0.5545],
       device='cuda:0')), ('conv2_bn.running_var', tensor([0.1635, 0.1210, 0.2374, 0.3369, 0.1443, 0.5569, 0.2300, 0.1417, 0.1057,
        0.1613, 0.1723, 0.1296, 0.1610, 0.1691, 0.2162, 0.1438, 0.1327, 0.2918,
        0.1665, 0.1055, 0.2297, 0.2135, 0.1279, 0.1275, 0.3499, 0.2123, 0.1732,
        0.2386, 0.2259, 0.0935, 0.0994, 0.1598, 0.1719, 0.2556, 0.3350, 0.3815,
        0.2924, 0.3197, 0.1513, 0.1875, 0.1659, 0.2965, 0.1072, 0.1130, 0.2062,
```

```
        0.2924, 0.3197, 0.1515, 0.1875, 0.1659, 0.2905, 0.1672, 0.1150, 0.2002,
        0.0809, 0.3647, 0.1855, 0.1759, 0.2119, 0.1480, 0.2188, 0.0903, 0.2558,
        0.2165, 0.1705, 0.4177, 0.1013, 0.2348, 0.2536, 0.0730, 0.1550, 0.1818,
        0.3084], device='cuda:0')), ('conv2_bn.num_batches_tracked', tensor(8160, device='cuda:0'))
, ('conv3.weight', tensor([[[[-0.0170,  0.0291, -0.0172, -0.0010,  0.0258],
          [ 0.0168, -0.0073, -0.0222,  0.0067,  0.0301],
          [ 0.0265, -0.0195,  0.0197,  0.0041, -0.0033],
          [-0.0148, -0.0002,  0.0091, -0.0179,  0.0245],
          [ 0.0057, -0.0076,  0.0166,  0.0053, -0.0069]],

         [[ 0.0208,  0.0317,  0.0122,  0.0290,  0.0221],
          [-0.0040,  0.0295, -0.0017,  0.0372,  0.0421],
          [-0.0174,  0.0219,  0.0246,  0.0094,  0.0331],
          [ 0.0087, -0.0186, -0.0316, -0.0083,  0.0204],
          [-0.0278, -0.0003, -0.0104,  0.0038, -0.0131]],

         [[-0.0022, -0.0168,  0.0149, -0.0215, -0.0078],
          [-0.0032,  0.0064,  0.0087,  0.0234,  0.0209],
          [-0.0017, -0.0250,  0.0135,  0.0179,  0.0090],
          [ 0.0098,  0.0077,  0.0270,  0.0303,  0.0070],
          [-0.0242, -0.0107,  0.0258, -0.0109, -0.0126]],

         ...,

         [[-0.0112,  0.0250, -0.0010,  0.0159, -0.0123],
          [-0.0091, -0.0030,  0.0251, -0.0102, -0.0244],
          [-0.0135, -0.0175, -0.0076,  0.0030,  0.0020],
          [ 0.0009, -0.0102, -0.0247, -0.0326, -0.0450],
          [ 0.0208, -0.0200, -0.0263,  0.0014, -0.0519]],

         [[-0.0165, -0.0145, -0.0206, -0.0195,  0.0223],
          [ 0.0149, -0.0190, -0.0001, -0.0085,  0.0107],
          [ 0.0036, -0.0234,  0.0113,  0.0086, -0.0110],
          [ 0.0015,  0.0049, -0.0183,  0.0139, -0.0251],
          [-0.0399,  0.0055, -0.0186,  0.0164, -0.0207]],

         [[-0.0037, -0.0203, -0.0319, -0.0241, -0.0249],
          [-0.0110, -0.0042, -0.0225, -0.0049, -0.0243],
          [-0.0294, -0.0325, -0.0304, -0.0208, -0.0161],
          [ 0.0044, -0.0362, -0.0512, -0.0090, -0.0251],
          [-0.0013, -0.0164, -0.0161, -0.0464, -0.0231]]],


        [[[-0.0270, -0.0021,  0.0245,  0.0070, -0.0128],
          [-0.0079, -0.0047, -0.0288,  0.0074, -0.0022],
          [-0.0065, -0.0095, -0.0106, -0.0192,  0.0182],
          [ 0.0107,  0.0174,  0.0078,  0.0066,  0.0015],
          [ 0.0076, -0.0182,  0.0157, -0.0175,  0.0226]],

         [[ 0.0041, -0.0397, -0.0038, -0.0092,  0.0011],
          [-0.0083,  0.0081, -0.0076,  0.0016,  0.0153],
          [-0.0078,  0.0022,  0.0188,  0.0125, -0.0098],
          [ 0.0067,  0.0107, -0.0167,  0.0112,  0.0040],
          [ 0.0398,  0.0010,  0.0353,  0.0370,  0.0249]],

         [[-0.0236, -0.0138,  0.0220,  0.0172, -0.0077],
          [ 0.0034,  0.0129,  0.0175, -0.0022,  0.0155],
          [-0.0034,  0.0178,  0.0079,  0.0070, -0.0067],
          [ 0.0285,  0.0161,  0.0101,  0.0087,  0.0303],
          [ 0.0240, -0.0000, -0.0012,  0.0012, -0.0004]],

         ...,

         [[ 0.0122,  0.0224, -0.0071,  0.0145,  0.0092],
          [ 0.0303, -0.0081,  0.0094, -0.0109,  0.0007],
          [ 0.0292, -0.0121,  0.0014,  0.0028,  0.0186],
          [ 0.0309, -0.0118, -0.0096,  0.0131,  0.0052],
          [ 0.0143, -0.0025,  0.0171, -0.0241, -0.0021]],

         [[ 0.0109, -0.0057, -0.0193,  0.0266,  0.0127],
          [-0.0001,  0.0105,  0.0053,  0.0351,  0.0345],
          [ 0.0232, -0.0150,  0.0118,  0.0452,  0.0363],
          [ 0.0217,  0.0251,  0.0324,  0.0416,  0.0163],
          [ 0.0341,  0.0377,  0.0274,  0.0368,  0.0219]],

         [[ 0.0377,  0.0083, -0.0243, -0.0123,  0.0129],
          [ 0.0171, -0.0071, -0.0070, -0.0080,  0.0014],
          [ 0.0117,  0.0195,  0.0120, -0.0138,  0.0093]
```

```
  [ 0.0117,  0.0195,  0.0120, -0.0150,  0.0095],
  [ 0.0054, -0.0149, -0.0039,  0.0202,  0.0293],
  [ 0.0106,  0.0236,  0.0195,  0.0257,  0.0312]]],


[[[ 0.0013, -0.0267, -0.0272, -0.0146,  0.0092],
  [-0.0074,  0.0070,  0.0078,  0.0332, -0.0067],
  [ 0.0029, -0.0184,  0.0231,  0.0395,  0.0402],
  [ 0.0043, -0.0198, -0.0040, -0.0082,  0.0316],
  [-0.0149,  0.0174,  0.0192,  0.0033,  0.0042]],

 [[-0.0029, -0.0237,  0.0113, -0.0048, -0.0211],
  [ 0.0153,  0.0074, -0.0154,  0.0092, -0.0339],
  [-0.0040, -0.0109,  0.0036,  0.0035, -0.0260],
  [-0.0187,  0.0258,  0.0098,  0.0196,  0.0213],
  [ 0.0363,  0.0024,  0.0382,  0.0155,  0.0282]],

 [[-0.0225,  0.0110, -0.0045, -0.0216,  0.0042],
  [ 0.0204, -0.0232, -0.0188,  0.0102,  0.0082],
  [-0.0013, -0.0074,  0.0067,  0.0216,  0.0087],
  [-0.0077, -0.0269,  0.0204, -0.0241, -0.0165],
  [-0.0098,  0.0218, -0.0198, -0.0161, -0.0336]],

 ...,

 [[-0.0174,  0.0159,  0.0048, -0.0021,  0.0157],
  [-0.0125,  0.0137,  0.0198,  0.0355,  0.0243],
  [-0.0148,  0.0059, -0.0091,  0.0303,  0.0109],
  [-0.0173,  0.0268,  0.0091, -0.0133, -0.0085],
  [-0.0168,  0.0022, -0.0433, -0.0168, -0.0087]],

 [[-0.0062,  0.0243,  0.0011, -0.0087, -0.0237],
  [ 0.0224,  0.0231, -0.0099, -0.0146, -0.0329],
  [ 0.0177,  0.0308,  0.0328,  0.0193, -0.0030],
  [ 0.0274,  0.0000, -0.0057, -0.0201,  0.0097],
  [ 0.0368,  0.0210, -0.0197, -0.0043, -0.0297]],

 [[ 0.0076, -0.0279, -0.0090, -0.0218, -0.0165],
  [ 0.0022, -0.0303, -0.0056, -0.0205,  0.0273],
  [ 0.0087, -0.0223, -0.0019,  0.0042,  0.0007],
  [-0.0092, -0.0043,  0.0064,  0.0248,  0.0106],
  [ 0.0127,  0.0229,  0.0230,  0.0060, -0.0218]]],


...,


[[[ 0.0407,  0.0402,  0.0309,  0.0208,  0.0180],
  [ 0.0226,  0.0229,  0.0037,  0.0090, -0.0238],
  [ 0.0108,  0.0190,  0.0128, -0.0254, -0.0233],
  [ 0.0278, -0.0027,  0.0143,  0.0084, -0.0200],
  [ 0.0219,  0.0139,  0.0317, -0.0068, -0.0033]],

 [[ 0.0001,  0.0173, -0.0004,  0.0199, -0.0223],
  [ 0.0293,  0.0196, -0.0276, -0.0425, -0.0321],
  [ 0.0119, -0.0049,  0.0036,  0.0044, -0.0179],
  [-0.0042, -0.0115, -0.0128, -0.0115, -0.0030],
  [-0.0046,  0.0135, -0.0178, -0.0368,  0.0143]],

 [[-0.0136, -0.0111, -0.0048, -0.0107, -0.0209],
  [-0.0210,  0.0018,  0.0260,  0.0278, -0.0018],
  [ 0.0117,  0.0214,  0.0024, -0.0148, -0.0154],
  [-0.0195,  0.0108,  0.0087,  0.0077,  0.0167],
  [ 0.0200,  0.0290,  0.0171, -0.0263, -0.0162]],

 ...,

 [[-0.0168, -0.0003,  0.0048, -0.0048, -0.0107],
  [ 0.0029, -0.0205,  0.0078,  0.0117, -0.0237],
  [-0.0175, -0.0001, -0.0211,  0.0094, -0.0053],
  [-0.0082,  0.0120, -0.0005,  0.0186, -0.0056],
  [-0.0102, -0.0125, -0.0048,  0.0160, -0.0039]],

 [[-0.0128, -0.0171, -0.0245, -0.0065,  0.0142],
  [-0.0087, -0.0022, -0.0129, -0.0060,  0.0033],
  [-0.0029, -0.0246,  0.0001, -0.0015, -0.0100],
  [-0.0343,  0.0022, -0.0018, -0.0374, -0.0102],
  [-0.0304,  0.0180,  0.0108,  0.0373,  0.0160]],
```

```
  [-0.0394, -0.0180, -0.0108, -0.0375, -0.0169]],

 [[ 0.0231,  0.0280, -0.0028, -0.0221,  0.0136],
  [ 0.0283, -0.0169,  0.0023, -0.0257,  0.0017],
  [ 0.0144,  0.0109, -0.0319,  0.0167, -0.0079],
  [-0.0186,  0.0079, -0.0205, -0.0147, -0.0168],
  [-0.0255,  0.0054, -0.0221,  0.0216, -0.0229]]],


[[[ 0.0085, -0.0244,  0.0011, -0.0118,  0.0022],
  [ 0.0073,  0.0187, -0.0116,  0.0099,  0.0027],
  [-0.0053,  0.0147,  0.0277,  0.0058, -0.0123],
  [ 0.0175,  0.0126, -0.0150,  0.0106, -0.0147],
  [-0.0087,  0.0104,  0.0182, -0.0034, -0.0175]],

 [[ 0.0111, -0.0085,  0.0121,  0.0487, -0.0106],
  [-0.0059, -0.0197,  0.0118,  0.0243,  0.0029],
  [-0.0264, -0.0310, -0.0196,  0.0081,  0.0061],
  [-0.0341, -0.0086, -0.0322,  0.0302,  0.0305],
  [-0.0020, -0.0025, -0.0328, -0.0202,  0.0242]],

 [[ 0.0095,  0.0185,  0.0064,  0.0256, -0.0124],
  [ 0.0235,  0.0033,  0.0307, -0.0149, -0.0045],
  [ 0.0264,  0.0463,  0.0106,  0.0162,  0.0095],
  [ 0.0018,  0.0382,  0.0025,  0.0229, -0.0061],
  [-0.0004,  0.0315, -0.0088,  0.0363,  0.0122]],

 ...,

 [[-0.0021, -0.0066, -0.0089, -0.0233, -0.0163],
  [-0.0092, -0.0095,  0.0133, -0.0245, -0.0363],
  [-0.0329,  0.0004, -0.0227, -0.0320, -0.0000],
  [ 0.0097,  0.0130, -0.0199, -0.0075, -0.0234],
  [ 0.0097,  0.0293,  0.0114,  0.0178,  0.0225]],

 [[-0.0288, -0.0241, -0.0103,  0.0103,  0.0088],
  [-0.0352, -0.0360,  0.0049, -0.0199, -0.0191],
  [ 0.0060,  0.0027,  0.0124,  0.0088, -0.0187],
  [-0.0123, -0.0151, -0.0132,  0.0030,  0.0003],
  [-0.0201, -0.0031,  0.0217,  0.0022,  0.0094]],

 [[-0.0229,  0.0134, -0.0089, -0.0348, -0.0237],
  [ 0.0136,  0.0022,  0.0162,  0.0086, -0.0008],
  [-0.0331, -0.0222, -0.0002, -0.0047, -0.0016],
  [-0.0152, -0.0274,  0.0094, -0.0224, -0.0135],
  [-0.0145, -0.0026,  0.0077,  0.0023,  0.0042]]],


[[[ 0.0219,  0.0177, -0.0175,  0.0151, -0.0126],
  [ 0.0121,  0.0111,  0.0276,  0.0157,  0.0084],
  [-0.0165, -0.0041, -0.0242, -0.0151,  0.0073],
  [-0.0009, -0.0132, -0.0153, -0.0152,  0.0052],
  [ 0.0338, -0.0020,  0.0044,  0.0104,  0.0123]],

 [[ 0.0231,  0.0070, -0.0172, -0.0039, -0.0198],
  [ 0.0158, -0.0187, -0.0359, -0.0223, -0.0282],
  [ 0.0197,  0.0197,  0.0197,  0.0240,  0.0047],
  [ 0.0193,  0.0269, -0.0304,  0.0037, -0.0110],
  [ 0.0163, -0.0127, -0.0195, -0.0247, -0.0236]],

 [[-0.0213,  0.0042, -0.0123,  0.0210,  0.0122],
  [-0.0064, -0.0089, -0.0067,  0.0292, -0.0152],
  [-0.0113,  0.0071, -0.0214,  0.0063,  0.0144],
  [-0.0137, -0.0221, -0.0098,  0.0288,  0.0117],
  [-0.0271,  0.0214,  0.0239, -0.0091, -0.0022]],

 ...,

 [[ 0.0028, -0.0234, -0.0232,  0.0076, -0.0152],
  [ 0.0034, -0.0140,  0.0101,  0.0162,  0.0133],
  [-0.0150,  0.0139, -0.0209,  0.0303,  0.0014],
  [ 0.0098, -0.0193, -0.0222,  0.0105,  0.0205],
  [ 0.0052,  0.0052, -0.0164,  0.0184,  0.0105]],

 [[-0.0107, -0.0048, -0.0009, -0.0153,  0.0254],
  [-0.0237, -0.0231, -0.0333, -0.0014, -0.0025],
  [-0.0148, -0.0286,  0.0081,  0.0007, -0.0012],
  [ 0.0212,  0.0130,  0.0040,  0.0005,  0.0124]
```

```
          [ 0.0212,  0.0139, -0.0040,  0.0095,  0.0124],
          [-0.0127,  0.0077, -0.0282, -0.0213, -0.0177]],

         [[-0.0173, -0.0075, -0.0229, -0.0231,  0.0055],
          [-0.0304, -0.0166,  0.0043,  0.0135, -0.0252],
          [-0.0340, -0.0023, -0.0149, -0.0176, -0.0238],
          [-0.0234, -0.0206, -0.0332,  0.0057,  0.0014],
          [-0.0014, -0.0064, -0.0291,  0.0058,  0.0028]]]], device='cuda:0')), ('conv3.bias', tensc
r([ 0.0093,  0.0220,  0.0009, -0.0003,  0.0137,  0.0156, -0.0189,  0.0025,
        -0.0076,  0.0174,  0.0213,  0.0055,  0.0037,  0.0108, -0.0138,  0.0138,
        -0.0219,  0.0152,  0.0209,  0.0018,  0.0018, -0.0062,  0.0232, -0.0089,
        -0.0127,  0.0202,  0.0010,  0.0080, -0.0112,  0.0086, -0.0109, -0.0200],
       device='cuda:0')), ('conv3_bn.weight', tensor([0.7141, 0.4729, 0.8028, 0.3566, 0.1970, 0.504
5, 0.6810, 0.9082, 0.9528,
        0.6940, 0.5592, 0.6561, 0.2367, 0.5608, 0.6263, 0.1924, 0.9443, 0.0558,
        0.2408, 0.9459, 0.7787, 0.7035, 0.8117, 0.3565, 0.9204, 0.3234, 0.1368,
        0.1478, 0.6904, 0.8167, 0.1325, 0.9595], device='cuda:0')), ('conv3_bn.bias', tensor([-0.00
36, -0.0309, -0.0098, -0.0163, -0.0078, -0.0189,  0.0162, -0.0197,
        -0.0003, -0.0066,  0.0222,  0.0152,  0.0042, -0.0191, -0.0021, -0.0004,
         0.0115,  0.0088, -0.0182,  0.0064, -0.0161, -0.0182, -0.0022,  0.0001,
        -0.0115,  0.0415, -0.0155, -0.0059,  0.0030,  0.0323, -0.0260,  0.0430],
       device='cuda:0')), ('conv3_bn.running_mean', tensor([-0.6313, -0.0841,  0.2167, -0.5382,  0.
2081,  0.1838, -0.1008,  0.9627,
        -0.8384, -0.7625, -0.1448, -0.3772, -0.3208, -0.7416, -0.5860, -0.1013,
        -0.4403, -0.9598, -0.4775, -0.2454,  0.0253,  0.9488, -0.8265, -0.4819,
        -0.3313,  0.0057, -0.5561, -0.7658, -0.4637, -0.2083, -0.2901, -0.0679],
       device='cuda:0')), ('conv3_bn.running_var', tensor([0.3805, 0.2030, 0.1639, 0.2643, 0.1084,
0.1575, 0.1610, 0.2321, 0.2607,
        0.2514, 0.2338, 0.1640, 0.1610, 0.2714, 0.2871, 0.2310, 0.2022, 0.2800,
        0.1941, 0.1090, 0.1913, 0.2675, 0.2003, 0.1719, 0.2552, 0.1025, 0.1798,
        0.2799, 0.1437, 0.0890, 0.1760, 0.1067], device='cuda:0')),
('conv3_bn.num_batches_tracked', tensor(8160, device='cuda:0')), ('conv4.weight', tensor([[[[ 0.00
85, -0.0095, -0.0382],
          [-0.0353, -0.0310,  0.0232],
          [-0.0546, -0.0088, -0.0236]],

         [[-0.0186,  0.0012,  0.0442],
          [ 0.0246,  0.0030,  0.0169],
          [-0.0210, -0.0326,  0.0021]],

         [[-0.0192,  0.0039, -0.0138],
          [-0.0423,  0.0494, -0.0255],
          [ 0.0414,  0.0011,  0.0144]],

         ...,

         [[-0.0216, -0.0063, -0.0268],
          [-0.0137,  0.0382,  0.0272],
          [ 0.0094,  0.0549, -0.0041]],

         [[ 0.0117,  0.0452,  0.0401],
          [-0.0263,  0.0357, -0.0244],
          [ 0.0168, -0.0160,  0.0058]],

         [[-0.0174, -0.0496, -0.0582],
          [-0.0559,  0.0319, -0.0276],
          [ 0.0662, -0.0161, -0.0202]]],


        [[[-0.0458, -0.0168, -0.0225],
          [-0.0452,  0.0345,  0.0322],
          [ 0.0288, -0.0170, -0.0541]],

         [[-0.0434,  0.0235, -0.0241],
          [-0.0150,  0.0492, -0.0081],
          [ 0.0403, -0.0052, -0.0109]],

         [[-0.0566, -0.0066, -0.0191],
          [ 0.0214,  0.0297, -0.0493],
          [ 0.0081, -0.0386, -0.0452]],

         ...,

         [[-0.0272,  0.0076,  0.0128],
          [ 0.0558,  0.0334, -0.0215],
          [-0.0469, -0.0375, -0.0519]],

         [[-0.0424, -0.0126, -0.0201],
```

```
 [[ 0.0484, -0.0126, -0.0301],
  [ 0.0180,  0.0313,  0.0014],
  [-0.0286,  0.0556,  0.0144]],

 [[ 0.0188,  0.0420, -0.0501],
  [-0.0176,  0.0160, -0.0505],
  [-0.0087,  0.0507,  0.0187]]],


[[[ 0.0255, -0.0439, -0.0086],
  [ 0.0062, -0.0315,  0.0211],
  [ 0.0265, -0.0427, -0.0165]],

 [[-0.0018,  0.0391,  0.0573],
  [ 0.0242,  0.0475,  0.0563],
  [ 0.0283, -0.0068, -0.0466]],

 [[ 0.0272, -0.0330,  0.0397],
  [-0.0077, -0.0354,  0.0146],
  [ 0.0323,  0.0221, -0.0481]],

 ...,

 [[-0.0208,  0.0364,  0.0373],
  [-0.0367, -0.0284,  0.0521],
  [ 0.0194, -0.0491, -0.0538]],

 [[ 0.0496,  0.0441,  0.0478],
  [ 0.0586, -0.0341, -0.0188],
  [-0.0360,  0.0473, -0.0269]],

 [[ 0.0104, -0.0454, -0.0546],
  [ 0.0087, -0.0619, -0.0255],
  [-0.0140, -0.0191, -0.0742]]],


...,


[[[ 0.0433, -0.0520,  0.0076],
  [-0.0531, -0.0574,  0.0509],
  [ 0.0020,  0.0471, -0.0410]],

 [[-0.0305, -0.0456, -0.0121],
  [ 0.0145, -0.0463,  0.0484],
  [-0.0542, -0.0392,  0.0302]],

 [[ 0.0237, -0.0532, -0.0226],
  [-0.0028,  0.0492,  0.0179],
  [-0.0028, -0.0215,  0.0167]],

 ...,

 [[-0.0313,  0.0513,  0.0231],
  [-0.0345, -0.0443,  0.0350],
  [ 0.0108, -0.0281,  0.0306]],

 [[-0.0341,  0.0352, -0.0592],
  [-0.0501, -0.0281, -0.0627],
  [-0.0405, -0.0110,  0.0345]],

 [[ 0.0332, -0.0347, -0.0076],
  [-0.0556,  0.0333,  0.0596],
  [-0.0242, -0.0157,  0.0694]]],


[[[ 0.0065,  0.0334,  0.0266],
  [ 0.0002, -0.0523,  0.0211],
  [-0.0334, -0.0662,  0.0000]],

 [[ 0.0289, -0.0435,  0.0229],
  [-0.0311,  0.0129, -0.0394],
  [-0.0309, -0.0643,  0.0463]],

 [[-0.0320,  0.0479,  0.0527],
  [-0.0543, -0.0123, -0.0567],
  [ 0.0296, -0.0719, -0.0425]],
```

```
          ...,

          [[-0.0350,  0.0670, -0.0372],
           [ 0.0201, -0.0395,  0.0557],
           [ 0.0011, -0.0256, -0.0185]],

          [[-0.0099, -0.0315,  0.0311],
           [ 0.0152,  0.0526, -0.0036],
           [-0.0026, -0.0381,  0.0217]],

          [[-0.0335, -0.0086,  0.0375],
           [-0.0623, -0.0041,  0.0111],
           [-0.0121,  0.0211,  0.0236]]],


         [[[ 0.0305,  0.0664,  0.0343],
           [ 0.0253,  0.0284, -0.0745],
           [-0.0458,  0.0622,  0.0125]],

          [[-0.0385,  0.0334,  0.0226],
           [-0.0611,  0.0215, -0.0003],
           [-0.0088, -0.0506,  0.0536]],

          [[ 0.0259, -0.0177, -0.0173],
           [-0.0360,  0.0476, -0.0281],
           [-0.0146,  0.0257,  0.0420]],

          ...,

          [[ 0.0567, -0.0239, -0.0271],
           [ 0.0416, -0.0139, -0.0624],
           [ 0.0246, -0.0627, -0.0314]],

          [[ 0.0080,  0.0398, -0.0094],
           [-0.0424, -0.0305, -0.0613],
           [ 0.0577,  0.0128,  0.0315]],

          [[ 0.0296,  0.0402,  0.0717],
           [-0.0692, -0.0112,  0.0339],
           [ 0.0459,  0.0478, -0.0042]]]], device='cuda:0')), ('conv4.bias', tensor([ 0.0496,  0.008
2, -0.0308,  0.0526,  0.0087, -0.0079,  0.0392, -0.0174,
         0.0213,  0.0253,  0.0097,  0.0468, -0.0489, -0.0025,  0.0140, -0.0331,
         0.0536,  0.0543,  0.0198, -0.0148,  0.0176, -0.0259, -0.0310,  0.0553,
         0.0549,  0.0434,  0.0339, -0.0257,  0.0574, -0.0189, -0.0030, -0.0237,
         0.0223, -0.0202, -0.0251,  0.0395, -0.0167, -0.0522, -0.0385, -0.0258,
         0.0172, -0.0256,  0.0452,  0.0112,  0.0465,  0.0153, -0.0146,  0.0257,
         0.0217,  0.0419, -0.0347, -0.0132, -0.0138,  0.0022,  0.0265,  0.0200,
         0.0060,  0.0376, -0.0100,  0.0576, -0.0173, -0.0473,  0.0416, -0.0005],
       device='cuda:0')), ('conv4_bn.weight', tensor([0.4675, 0.1948, 0.3481, 0.7562, 0.5113, 0.158
9, 0.7162, 0.5178, 1.0016,
         0.1335, 0.5650, 0.2057, 0.3942, 0.5628, 0.8956, 0.8043, 0.3952, 0.5657,
         0.9636, 0.8389, 0.9474, 0.5757, 0.9608, 0.2627, 0.4821, 0.5975, 0.1158,
         0.4961, 0.6885, 0.9508, 0.4125, 0.5879, 0.2931, 0.8289, 0.0318, 0.3721,
         0.4727, 0.5521, 0.6838, 0.3100, 0.9552, 0.9525, 0.3127, 0.4344, 0.6637,
         0.8290, 0.4810, 0.0287, 0.7037, 0.8898, 0.4920, 0.1737, 0.6259, 0.0562,
         0.5315, 0.7383, 0.3112, 0.6892, 0.4596, 0.9562, 0.2869, 0.9803, 0.5836,
         0.4839], device='cuda:0')), ('conv4_bn.bias', tensor([-0.0146, -0.0273, -0.0208, -0.0229,
-0.0118, -0.0162,  0.0152, -0.0309,
         -0.0323, -0.0040, -0.0192, -0.0097, -0.0154, -0.0354, -0.0091, -0.0249,
         -0.0267, -0.0265, -0.0531, -0.0224, -0.0431, -0.0239, -0.0001, -0.0083,
          0.0020, -0.0079, -0.0221, -0.0079, -0.0037, -0.0013, -0.0248, -0.0131,
         -0.0321, -0.0114, -0.0163, -0.0218, -0.0439, -0.0063, -0.0050, -0.0257,
         -0.0424, -0.0010,  0.0088, -0.0015, -0.0424, -0.0548, -0.0227, -0.0120,
         -0.0251, -0.0317, -0.0109, -0.0100, -0.0306, -0.0169, -0.0318, -0.0262,
         -0.0399, -0.0122, -0.0363, -0.0256, -0.0140, -0.0199, -0.0101, -0.0151],
       device='cuda:0')), ('conv4_bn.running_mean', tensor([-0.3773,  0.2934, -0.2924,  0.0717,  0.
0090, -0.3154, -0.4189, -0.0419,
         -0.1565, -0.2044,  0.5445,  0.5262, -0.2044,  0.2368, -0.1219, -0.8167,
          0.7078, -0.0220, -0.2629,  0.3396, -0.2997,  0.0874, -0.0478,  0.0665,
          0.1041, -0.5877,  0.5271, -0.4020, -0.2050, -0.3616,  0.5055, -0.0770,
         -0.5221, -0.1362,  0.9304, -0.0909,  1.1701, -0.4391, -0.0651, -0.2678,
         -0.2339, -0.2667, -0.3418, -0.2399, -0.0637,  0.4180, -0.1231, -0.6772,
          0.2089, -0.5070, -0.2707, -0.2476, -0.2973, -0.6817, -0.0880, -0.0373,
          0.5925, -0.1812,  0.8351, -0.2172, -0.1398, -0.0538, -0.2268, -0.0176],
       device='cuda:0')), ('conv4_bn.running_var', tensor([0.1458, 0.1424, 0.1175, 0.1552, 0.1372,
0.1502, 0.1024, 0.2236, 0.0829,
```

```
        0.1088, 0.1783, 0.1132, 0.1955, 0.1230, 0.1738, 0.1800, 0.1178, 0.1500,
        0.1497, 0.1129, 0.1107, 0.2257, 0.1236, 0.0868, 0.1710, 0.1961, 0.2321,
        0.0919, 0.0998, 0.1294, 0.2538, 0.1270, 0.2605, 0.1395, 0.2848, 0.1472,
        0.1762, 0.1380, 0.1288, 0.1027, 0.1814, 0.1081, 0.1017, 0.1334, 0.1278,
        0.1765, 0.0887, 0.1957, 0.1897, 0.1402, 0.2809, 0.1605, 0.1365, 0.2320,
        0.1141, 0.1188, 0.2603, 0.1100, 0.2426, 0.1464, 0.1036, 0.1035, 0.1405,
        0.0999], device='cuda:0')), ('conv4_bn.num_batches_tracked', tensor(8160, device='cuda:0'))
, ('conv5.weight', tensor([[[[-0.0263,  0.0319,  0.0196],
          [ 0.0164,  0.0338, -0.0059],
          [-0.0073,  0.0056,  0.0422]],

         [[ 0.0052,  0.0202, -0.0202],
          [ 0.0529,  0.0241,  0.0389],
          [ 0.0576,  0.0323, -0.0212]],

         [[-0.0400, -0.0254, -0.0110],
          [ 0.0077, -0.0241,  0.0366],
          [-0.0107, -0.0288,  0.0608]],

         ...,

         [[-0.0374,  0.0236,  0.0132],
          [ 0.0203, -0.0166, -0.0449],
          [ 0.0166,  0.0426, -0.0525]],

         [[ 0.0154,  0.0000, -0.0417],
          [-0.0163,  0.0113,  0.0172],
          [-0.0405,  0.0292, -0.0224]],

         [[-0.0065, -0.0002, -0.0079],
          [-0.0519,  0.0238,  0.0100],
          [-0.0180,  0.0212, -0.0067]]],


        [[[ 0.0016, -0.0445,  0.0194],
          [-0.0532, -0.0426, -0.0065],
          [-0.0196, -0.0293,  0.0413]],

         [[ 0.0235,  0.0229, -0.0066],
          [ 0.0257,  0.0023,  0.0082],
          [ 0.0453,  0.0067, -0.0146]],

         [[-0.0376, -0.0016,  0.0357],
          [ 0.0294,  0.0258,  0.0020],
          [ 0.0388,  0.0129, -0.0234]],

         ...,

         [[-0.0344,  0.0048, -0.0260],
          [ 0.0062, -0.0064,  0.0444],
          [ 0.0362,  0.0072,  0.0516]],

         [[ 0.0169,  0.0421,  0.0087],
          [ 0.0271,  0.0372, -0.0152],
          [-0.0173, -0.0120,  0.0006]],

         [[ 0.0278,  0.0061,  0.0007],
          [-0.0165,  0.0328, -0.0388],
          [ 0.0286,  0.0122,  0.0338]]],


        [[[-0.0103, -0.0257,  0.0279],
          [-0.0073, -0.0188, -0.0284],
          [-0.0070,  0.0396,  0.0388]],

         [[-0.0031,  0.0208,  0.0036],
          [-0.0037, -0.0215,  0.0134],
          [-0.0253,  0.0136, -0.0298]],

         [[ 0.0298, -0.0184,  0.0202],
          [ 0.0388, -0.0076, -0.0357],
          [-0.0471, -0.0291, -0.0259]],

         ...,

         [[ 0.0321,  0.0380, -0.0053],
          [ 0.0158, -0.0147, -0.0536],
```

```
  [-0.0364,  0.0093, -0.0635]],

 [[ 0.0355, -0.0058,  0.0278],
  [ 0.0058, -0.0274, -0.0155],
  [-0.0218,  0.0027,  0.0277]],

 [[-0.0445, -0.0466, -0.0383],
  [-0.0334, -0.0351, -0.0415],
  [ 0.0216, -0.0259, -0.0023]]],


...,


[[[-0.0397, -0.0224,  0.0203],
  [-0.0023, -0.0323,  0.0105],
  [ 0.0199, -0.0213, -0.0229]],

 [[-0.0109, -0.0176, -0.0336],
  [-0.0145, -0.0171,  0.0317],
  [-0.0223, -0.0140,  0.0081]],

 [[-0.0008, -0.0081,  0.0497],
  [ 0.0230,  0.0417, -0.0147],
  [-0.0287, -0.0387, -0.0077]],

 ...,

 [[ 0.0438,  0.0125,  0.0369],
  [ 0.0386,  0.0102,  0.0294],
  [-0.0331, -0.0259, -0.0015]],

 [[-0.0307,  0.0013,  0.0247],
  [-0.0393,  0.0419,  0.0288],
  [ 0.0082, -0.0260, -0.0084]],

 [[-0.0378,  0.0118,  0.0145],
  [-0.0023, -0.0164, -0.0303],
  [-0.0189, -0.0609, -0.0289]]],


[[[ 0.0075, -0.0211, -0.0253],
  [-0.0242,  0.0230, -0.0200],
  [ 0.0286, -0.0006, -0.0146]],

 [[ 0.0056, -0.0187, -0.0430],
  [-0.0104,  0.0056,  0.0325],
  [-0.0549, -0.0455,  0.0102]],

 [[-0.0022,  0.0206,  0.0137],
  [-0.0050,  0.0175, -0.0073],
  [-0.0448,  0.0152,  0.0069]],

 ...,

 [[-0.0170,  0.0139, -0.0026],
  [-0.0350, -0.0024, -0.0059],
  [-0.0018, -0.0339,  0.0125]],

 [[-0.0022, -0.0264, -0.0243],
  [ 0.0287, -0.0286, -0.0221],
  [ 0.0164, -0.0156, -0.0333]],

 [[ 0.0219, -0.0400,  0.0068],
  [-0.0095,  0.0009, -0.0457],
  [-0.0282, -0.0235,  0.0060]]],


[[[ 0.0199, -0.0041, -0.0145],
  [-0.0242,  0.0162,  0.0314],
  [-0.0188,  0.0097, -0.0293]],

 [[-0.0301, -0.0367,  0.0234],
  [ 0.0130,  0.0359, -0.0425],
  [ 0.0287,  0.0215, -0.0114]],

 [[ 0.0156, -0.0105, -0.0405],
```

```
          [ 0.0247,  0.0036,  0.0259],
          [ 0.0470,  0.0247, -0.0423]],

         ...,

         [[ 0.0180, -0.0414, -0.0232],
          [ 0.0152, -0.0119, -0.0419],
          [-0.0070,  0.0195, -0.0123]],

         [[ 0.0131,  0.0305,  0.0316],
          [-0.0170, -0.0120,  0.0156],
          [-0.0380, -0.0037,  0.0313]],

         [[-0.0166,  0.0314,  0.0370],
          [-0.0009,  0.0127, -0.0047],
          [-0.0395, -0.0364,  0.0006]]]], device='cuda:0')), ('conv5.bias', tensor([ 0.0185,  0.010
9,  0.0073, -0.0115,  0.0131, -0.0042,  0.0104,  0.0123,
        -0.0313,  0.0225, -0.0372,  0.0358,  0.0029, -0.0327,  0.0163,  0.0079,
         0.0117, -0.0161,  0.0179,  0.0077,  0.0300, -0.0231, -0.0414, -0.0197,
         0.0124, -0.0297,  0.0160, -0.0298, -0.0189, -0.0155, -0.0294,  0.0109,
         0.0174, -0.0207,  0.0281,  0.0226,  0.0020,  0.0086, -0.0081,  0.0237,
         0.0180, -0.0368, -0.0354,  0.0398,  0.0304,  0.0256, -0.0348,  0.0293,
         0.0414,  0.0305, -0.0413,  0.0019,  0.0105, -0.0149, -0.0342, -0.0009,
        -0.0318,  0.0259,  0.0146,  0.0307,  0.0174,  0.0354,  0.0083,  0.0047,
         0.0017, -0.0028, -0.0107, -0.0134, -0.0317, -0.0129, -0.0069, -0.0285,
        -0.0385,  0.0375,  0.0390,  0.0363, -0.0327, -0.0140,  0.0049, -0.0051,
         0.0317, -0.0369,  0.0089,  0.0373,  0.0022, -0.0166, -0.0184,  0.0264,
        -0.0257,  0.0091, -0.0136, -0.0167, -0.0200,  0.0345, -0.0137,  0.0348,
        -0.0023,  0.0199, -0.0252, -0.0119, -0.0207, -0.0377,  0.0097,  0.0050,
         0.0004, -0.0265, -0.0355, -0.0205,  0.0009, -0.0028,  0.0016, -0.0397,
        -0.0349, -0.0296, -0.0033,  0.0025, -0.0020,  0.0014, -0.0314,  0.0109,
         0.0409, -0.0193,  0.0361, -0.0094,  0.0247, -0.0351,  0.0262,  0.0240],
       device='cuda:0')), ('conv5_bn.weight', tensor([0.1914, 0.6326, 0.5565, 0.0260, 0.3462, 0.864
1, 0.0913, 0.7189, 0.4358,
        0.4206, 0.2959, 0.8930, 0.1044, 0.5596, 0.3438, 0.9657, 0.8241, 0.9159,
        0.2390, 0.0523, 0.9919, 0.8096, 0.5451, 0.2202, 0.5772, 0.1811, 0.8201,
        0.7122, 0.2239, 0.9177, 0.6009, 0.7310, 0.0757, 0.3645, 0.2180, 0.6446,
        0.4765, 0.0508, 0.5639, 0.4533, 0.5949, 0.7924, 0.4402, 0.8657, 0.6077,
        0.6641, 0.1788, 0.5645, 0.6665, 0.6100, 0.8155, 0.2965, 0.1882, 0.3560,
        0.7946, 0.6833, 0.6208, 0.8982, 0.3480, 0.7259, 0.2056, 0.7423, 0.5996,
        0.7795, 0.8425, 0.1776, 0.9334, 0.1019, 0.6558, 0.1860, 0.0980, 0.3488,
        0.4999, 1.0035, 0.2759, 0.0468, 0.6919, 0.6153, 0.0830, 0.0980, 0.7912,
        0.9826, 0.7975, 0.5168, 0.9337, 0.6523, 0.5699, 0.4591, 0.0695, 0.8304,
        0.3332, 0.6313, 0.0609, 0.8055, 0.5335, 0.5663, 0.0435, 0.2585, 0.4396,
        0.3782, 0.1199, 0.6890, 0.3308, 0.3365, 0.7483, 0.1488, 0.4072, 0.9667,
        0.1745, 0.1979, 0.8085, 0.9495, 0.5789, 0.0739, 0.0776, 0.6586, 0.9262,
        0.5146, 0.8824, 0.7877, 0.9515, 0.5118, 0.0708, 0.3991, 0.0342, 0.2946,
        0.6027, 0.9830], device='cuda:0')), ('conv5_bn.bias', tensor([-0.0293, -0.0135, -0.0347, -
0.0096, -0.0107, -0.0335, -0.0184, -0.0126,
        -0.0422, -0.0317, -0.0295, -0.0176, -0.0176, -0.0215, -0.0376, -0.0404,
        -0.0046, -0.0295, -0.0081, -0.0196, -0.0109, -0.0238, -0.0138, -0.0217,
        -0.0265, -0.0276, -0.0263, -0.0379, -0.0217, -0.0506, -0.0335, -0.0224,
        -0.0005, -0.0203, -0.0082, -0.0156, -0.0175, -0.0186, -0.0189, -0.0287,
        -0.0319, -0.0385, -0.0327, -0.0083, -0.0162, -0.0244, -0.0212, -0.0003,
        -0.0333, -0.0198, -0.0182, -0.0213, -0.0280, -0.0097, -0.0359, -0.0356,
        -0.0045, -0.0116, -0.0166, -0.0369, -0.0184, -0.0329, -0.0263, -0.0271,
        -0.0355, -0.0085, -0.0291, -0.0146, -0.0275, -0.0121, -0.0254, -0.0193,
        -0.0360, -0.0151, -0.0384, -0.0120, -0.0192, -0.0419, -0.0123, -0.0168,
        -0.0329, -0.0311, -0.0418, -0.0242, -0.0163, -0.0387, -0.0250, -0.0261,
        -0.0266, -0.0324, -0.0261, -0.0135, -0.0121, -0.0061, -0.0418, -0.0260,
         0.0007, -0.0045, -0.0171, -0.0171, -0.0184, -0.0278, -0.0017, -0.0147,
        -0.0060, -0.0210, -0.0033, -0.0235, -0.0101, -0.0091, -0.0285, -0.0199,
        -0.0312, -0.0090, -0.0179, -0.0275, -0.0147, -0.0258, -0.0256, -0.0254,
        -0.0362, -0.0146, -0.0201, -0.0285, -0.0103, -0.0082, -0.0231, -0.0041],
       device='cuda:0')), ('conv5_bn.running_mean', tensor([ 0.1646,  0.1269, -0.2383, -0.2719, -
0.6361,  0.3352,  0.1633,  0.1409,
        -0.1130, -0.1683, -0.0352, -0.6037, -0.2060,  0.0971,  0.6402, -0.3334,
        -0.7281, -0.4596, -0.4998,  0.0493, -0.2213,  0.2218,  0.3143, -0.0204,
         0.3093, -0.0671, -0.2082, -0.4206,  0.0745,  0.3178,  0.0145,  0.0275,
         0.0371,  0.1762,  0.0234, -0.4587,  0.1322,  0.3160, -0.7292, -0.0431,
         0.3352, -0.5629, -0.5686,  0.6522,  0.0340,  0.4584,  0.4619,  0.0856,
         0.7018,  0.5543,  0.2159, -0.6773, -0.4121, -0.3652,  0.3038,  0.1312,
        -0.7614, -0.0791, -0.1615, -0.5119, -0.2627,  0.1908, -0.1087, -0.1383,
         0.4100, -0.0048,  0.3714,  0.5954, -0.4405,  0.3432,  0.5162, -0.0256,
        -0.2471, -0.7206,  0.2512, -0.1429, -0.6321, -0.3339, -0.0704, -0.1717,
        -0.3461, -0.2922,  0.2136, -0.4114,  0.2293, -0.0604, -0.0642,  0.1784,
        -0.1609, -0.2261,  0.3608, -0.3755,  0.6179, -0.2497, -0.1206,  0.0405,
```

```
         0.2692,  0.1757, -0.3363, -0.4266, -0.6333, -0.7447, -0.2864, -0.6929,
        -0.2087, -0.4738, -0.4351,  0.8298,  0.1607,  0.0225,  0.4925,  0.0310,
        -0.1882, -0.1762, -0.2893,  0.2583, -0.1693, -0.6033, -0.4129,  0.0012,
         0.1843, -0.1299,  0.2409, -0.6171, -0.3928, -0.8827, -0.8418, -0.4620],
       device='cuda:0')), ('conv5_bn.running_var', tensor([0.0752, 0.1060, 0.1027, 0.0864, 0.1363,
0.1276, 0.1812, 0.0934, 0.0955,
        0.1153, 0.0936, 0.2192, 0.0935, 0.1178, 0.1783, 0.1217, 0.1520, 0.1490,
        0.1303, 0.1017, 0.1531, 0.1172, 0.0982, 0.0658, 0.0949, 0.1092, 0.1136,
        0.1150, 0.1034, 0.1047, 0.0657, 0.1181, 0.0892, 0.0894, 0.1012, 0.1794,
        0.1390, 0.2030, 0.1742, 0.1011, 0.1120, 0.1770, 0.2379, 0.2054, 0.1492,
        0.1202, 0.1000, 0.1029, 0.1352, 0.1406, 0.0858, 0.1272, 0.1134, 0.0730,
        0.0897, 0.1157, 0.2621, 0.2098, 0.1513, 0.2220, 0.1005, 0.1064, 0.1253,
        0.1231, 0.1180, 0.1142, 0.1645, 0.0966, 0.1162, 0.1306, 0.1179, 0.1391,
        0.0857, 0.1884, 0.2151, 0.0809, 0.1400, 0.1067, 0.0942, 0.0865, 0.1165,
        0.0830, 0.0976, 0.0942, 0.1401, 0.0632, 0.1040, 0.1381, 0.0822, 0.0975,
        0.1358, 0.0847, 0.1262, 0.1021, 0.0934, 0.1199, 0.1032, 0.1076, 0.1158,
        0.1255, 0.1567, 0.1344, 0.0992, 0.1414, 0.1412, 0.1369, 0.1142, 0.1571,
        0.0830, 0.0784, 0.1021, 0.0948, 0.1596, 0.1397, 0.0977, 0.1228, 0.1164,
        0.1280, 0.0879, 0.0907, 0.1257, 0.1431, 0.1222, 0.1876, 0.1438, 0.2421,
        0.1901, 0.1399], device='cuda:0')), ('conv5_bn.num_batches_tracked', tensor(8160, device='c
uda:0')), ('conv6.weight', tensor([[[[-0.0207, -0.0213,  0.0110],
          [-0.0300, -0.0277,  0.0231],
          [-0.0412, -0.0049, -0.0049]],

         [[-0.0351, -0.0060, -0.0230],
          [ 0.0035, -0.0095,  0.0175],
          [-0.0373, -0.0010, -0.0112]],

         [[ 0.0214,  0.0165, -0.0037],
          [ 0.0208, -0.0146, -0.0028],
          [ 0.0003, -0.0248, -0.0099]],

         ...,

         [[ 0.0108, -0.0269, -0.0022],
          [ 0.0235,  0.0206,  0.0204],
          [-0.0155, -0.0095, -0.0054]],

         [[-0.0138, -0.0165, -0.0209],
          [ 0.0158, -0.0208,  0.0260],
          [ 0.0442,  0.0004,  0.0099]],

         [[-0.0399, -0.0567, -0.0380],
          [-0.0344, -0.0163, -0.0270],
          [-0.0026, -0.0228, -0.0316]]],


        [[[ 0.0276, -0.0215, -0.0267],
          [ 0.0032, -0.0044, -0.0208],
          [ 0.0040,  0.0312,  0.0045]],

         [[-0.0357, -0.0323,  0.0221],
          [-0.0303, -0.0229,  0.0222],
          [-0.0394, -0.0288, -0.0402]],

         [[ 0.0018, -0.0380, -0.0418],
          [ 0.0135, -0.0007, -0.0313],
          [ 0.0105,  0.0056, -0.0133]],

         ...,

         [[ 0.0272, -0.0010,  0.0130],
          [ 0.0105, -0.0262, -0.0101],
          [-0.0278,  0.0102, -0.0091]],

         [[-0.0174, -0.0323, -0.0078],
          [-0.0110, -0.0117, -0.0450],
          [-0.0064, -0.0314, -0.0224]],

         [[ 0.0041, -0.0347, -0.0249],
          [-0.0152, -0.0382,  0.0040],
          [-0.0315, -0.0328, -0.0475]]],


        [[[-0.0160, -0.0322, -0.0042],
          [ 0.0146, -0.0119, -0.0258],
          [ 0.0053, -0.0186, -0.0187]],
```

```
 [[ 0.0193,  0.0186, -0.0208],
  [-0.0153, -0.0053, -0.0067],
  [-0.0162, -0.0037, -0.0100]],

 [[ 0.0213, -0.0320, -0.0118],
  [ 0.0153,  0.0235, -0.0328],
  [ 0.0164,  0.0105,  0.0141]],

 ...,

 [[-0.0059, -0.0143, -0.0092],
  [ 0.0193,  0.0208,  0.0297],
  [ 0.0186,  0.0034,  0.0193]],

 [[-0.0174, -0.0125, -0.0088],
  [ 0.0206,  0.0017,  0.0357],
  [ 0.0388,  0.0156, -0.0068]],

 [[ 0.0117,  0.0146,  0.0086],
  [ 0.0173, -0.0080, -0.0112],
  [-0.0307, -0.0024,  0.0054]]],


...,


[[[ 0.0038,  0.0241, -0.0052],
  [-0.0119, -0.0194,  0.0210],
  [-0.0374, -0.0240, -0.0021]],

 [[-0.0076,  0.0173,  0.0248],
  [-0.0243,  0.0195,  0.0288],
  [-0.0118, -0.0088, -0.0115]],

 [[-0.0239, -0.0332, -0.0188],
  [-0.0261, -0.0228,  0.0296],
  [-0.0001, -0.0195, -0.0060]],

 ...,

 [[ 0.0104,  0.0080,  0.0079],
  [ 0.0252, -0.0012, -0.0146],
  [-0.0056, -0.0174, -0.0249]],

 [[-0.0222,  0.0085,  0.0198],
  [ 0.0120, -0.0267, -0.0080],
  [-0.0356, -0.0344,  0.0103]],

 [[ 0.0410,  0.0387,  0.0391],
  [ 0.0163,  0.0106,  0.0227],
  [ 0.0142,  0.0196,  0.0208]]],


[[[-0.0290, -0.0265, -0.0303],
  [-0.0268,  0.0149, -0.0300],
  [ 0.0099, -0.0265,  0.0358]],

 [[ 0.0007,  0.0227, -0.0000],
  [ 0.0282,  0.0362, -0.0187],
  [ 0.0301,  0.0102,  0.0332]],

 [[-0.0102, -0.0304, -0.0202],
  [ 0.0277, -0.0177, -0.0101],
  [ 0.0169,  0.0184, -0.0295]],

 ...,

 [[-0.0407, -0.0070,  0.0117],
  [-0.0197,  0.0115,  0.0030],
  [-0.0254, -0.0137,  0.0084]],

 [[-0.0088,  0.0197,  0.0097],
  [ 0.0096, -0.0043, -0.0425],
  [ 0.0090, -0.0048, -0.0185]],

 [[-0.0008, -0.0161, -0.0061],
```

```
          [ 0.0217, -0.0047, -0.0218],
          [-0.0246, -0.0128, -0.0092]]],


        [[[-0.0113,  0.0046, -0.0370],
          [-0.0257, -0.0232, -0.0307],
          [ 0.0129,  0.0148,  0.0013]],

         [[-0.0161, -0.0015,  0.0120],
          [-0.0521, -0.0203, -0.0050],
          [-0.0228,  0.0001, -0.0218]],

         [[ 0.0060,  0.0364, -0.0024],
          [ 0.0058, -0.0282,  0.0164],
          [ 0.0155,  0.0056,  0.0063]],

         ...,

         [[-0.0340, -0.0100, -0.0094],
          [ 0.0113, -0.0062, -0.0069],
          [-0.0303,  0.0190, -0.0301]],

         [[-0.0337, -0.0191, -0.0278],
          [-0.0198,  0.0287,  0.0310],
          [ 0.0044, -0.0051, -0.0136]],

         [[ 0.0137,  0.0218, -0.0264],
          [ 0.0340,  0.0066, -0.0152],
          [ 0.0399,  0.0189,  0.0198]]]], device='cuda:0')), ('conv6.bias', tensor([-0.0258,  0.014
0,  0.0233,  0.0185,  0.0011, -0.0167, -0.0280,  0.0085,
         0.0115,  0.0280,  0.0228, -0.0030, -0.0207, -0.0171,  0.0044,  0.0266,
        -0.0231,  0.0236, -0.0053,  0.0098, -0.0126,  0.0046, -0.0061,  0.0118,
        -0.0101, -0.0267,  0.0193,  0.0265, -0.0034,  0.0123, -0.0219,  0.0179,
         0.0073,  0.0153, -0.0031, -0.0250, -0.0004,  0.0115, -0.0272,  0.0130,
         0.0064, -0.0039,  0.0072, -0.0239, -0.0172,  0.0234, -0.0172, -0.0025,
         0.0083,  0.0220, -0.0226,  0.0044,  0.0144, -0.0283, -0.0178,  0.0257,
         0.0110,  0.0261,  0.0132, -0.0121, -0.0130,  0.0241, -0.0036, -0.0048],
       device='cuda:0')), ('conv6_bn.weight', tensor([0.6751, 0.6988, 0.2303, 0.5104, 0.1604, 0.021
2, 0.8522, 0.0289, 0.8047,
        0.4433, 0.5874, 0.6523, 0.3084, 0.1554, 0.9484, 0.8707, 0.1192, 0.5513,
        0.4040, 0.6583, 0.5257, 0.9485, 0.9495, 0.0503, 0.3467, 0.8910, 0.6006,
        0.0145, 0.9990, 0.6189, 0.6586, 0.6502, 0.5065, 0.4543, 0.9037, 0.6151,
        0.1337, 0.7185, 0.5335, 0.0661, 0.5675, 0.7426, 0.5587, 0.3582, 0.2712,
        0.2053, 0.3666, 0.3219, 0.4313, 0.3917, 0.7930, 0.5646, 0.2135, 0.8685,
        0.1679, 0.1223, 0.9973, 0.4314, 0.9183, 0.4401, 0.7347, 0.6721, 0.4092,
        0.1497], device='cuda:0')), ('conv6_bn.bias', tensor([-0.0377, -0.0317, -0.0294,  0.0069,
-0.0197, -0.0059,  0.0127,  0.0095,
        -0.0167, -0.0252,  0.0136, -0.0325, -0.0219,  0.0060,  0.0035, -0.0412,
         0.0037,  0.0069,  0.0034, -0.0209,  0.0165, -0.0320,  0.0037,  0.0152,
        -0.0132, -0.0302, -0.0308, -0.0112,  0.0059, -0.0100, -0.0084, -0.0185,
        -0.0436,  0.0024, -0.0170, -0.0428,  0.0191, -0.0070,  0.0145, -0.0236,
         0.0076, -0.0284, -0.0102, -0.0074,  0.0001,  0.0009, -0.0069, -0.0284,
        -0.0085, -0.0148, -0.0216, -0.0211, -0.0108, -0.0181,  0.0032,  0.0152,
        -0.0085, -0.0004, -0.0241, -0.0182, -0.0355, -0.0042, -0.0131, -0.0074],
       device='cuda:0')), ('conv6_bn.running_mean', tensor([ 0.2457, -1.0438,  0.0412, -0.3612, -
0.4698, -0.5011, -0.2927, -0.3666,
        -0.3926,  0.0140, -0.1420, -0.6286, -0.2723,  0.1918,  0.0879, -0.2651,
         0.1141, -0.0454, -0.2030, -0.0023, -0.1630,  0.0583, -0.5851, -0.1581,
        -0.8256, -0.0753,  0.0693, -0.0902, -0.1789, -0.2789, -0.1769, -0.1547,
        -0.0505, -0.0512, -0.0901,  0.2905, -0.3211, -1.0213, -0.1464,  0.1255,
        -0.4407, -0.5202, -0.2091, -0.2207,  0.0130,  0.0023, -0.6206, -0.4912,
        -0.6979,  0.1789,  0.1390,  0.0557, -0.4986,  0.5272, -0.1649,  0.2437,
         0.0269,  0.0718, -0.4540, -0.0252, -0.4000,  0.0692, -0.5773, -0.4310],
       device='cuda:0')), ('conv6_bn.running_var', tensor([0.1370, 0.1886, 0.1308, 0.1326, 0.1130,
0.1337, 0.0947, 0.1793, 0.1850,
        0.0977, 0.2303, 0.1449, 0.1322, 0.1579, 0.0967, 0.0804, 0.2247, 0.1066,
        0.0920, 0.2018, 0.1589, 0.1800, 0.1256, 0.2258, 0.2568, 0.1093, 0.1658,
        0.1596, 0.0875, 0.2357, 0.2223, 0.1492, 0.1632, 0.1596, 0.0992, 0.1792,
        0.3051, 0.1442, 0.2666, 0.1673, 0.1868, 0.0976, 0.1195, 0.1911, 0.1611,
        0.2071, 0.1648, 0.1325, 0.1776, 0.1307, 0.1157, 0.0924, 0.1590, 0.1057,
        0.1692, 0.1926, 0.1539, 0.1768, 0.1454, 0.1759, 0.1424, 0.1718, 0.1756,
        0.1621], device='cuda:0')), ('conv6_bn.num_batches_tracked', tensor(8160, device='cuda:0'))
, ('conv7.weight', tensor([[[[ 0.0108,  0.0033,  0.0119, -0.0111,  0.0104],
          [ 0.0161, -0.0280, -0.0250, -0.0237,  0.0111],
          [ 0.0031,  0.0080,  0.0167,  0.0081, -0.0019],
          [ 0.0134,  0.0163, -0.0134,  0.0191,  0.0058],
          [-0.0249,  0.0069,  0.0053, -0.0101,  0.0237]],
```

```
 [[-0.0178, -0.0189, -0.0072, -0.0030, -0.0208],
  [-0.0019, -0.0158, -0.0356, -0.0398, -0.0038],
  [-0.0017, -0.0346, -0.0336, -0.0019, -0.0063],
  [-0.0093,  0.0076, -0.0047,  0.0193,  0.0217],
  [-0.0041,  0.0291,  0.0257,  0.0011,  0.0299]],

 [[ 0.0297,  0.0081, -0.0174,  0.0062,  0.0211],
  [ 0.0022,  0.0175,  0.0035,  0.0256,  0.0018],
  [-0.0133, -0.0100, -0.0061,  0.0010, -0.0194],
  [-0.0027, -0.0162, -0.0157, -0.0052,  0.0151],
  [ 0.0162, -0.0030,  0.0067,  0.0233,  0.0205]],

 ...,

 [[ 0.0253,  0.0152, -0.0070,  0.0229, -0.0164],
  [-0.0028, -0.0267, -0.0022,  0.0017, -0.0340],
  [-0.0126, -0.0176, -0.0029, -0.0140, -0.0318],
  [ 0.0229,  0.0045,  0.0070, -0.0155, -0.0316],
  [ 0.0189,  0.0163, -0.0067, -0.0200, -0.0270]],

 [[ 0.0340,  0.0294,  0.0095,  0.0041,  0.0229],
  [ 0.0358,  0.0044,  0.0098,  0.0029,  0.0141],
  [ 0.0106,  0.0261,  0.0352,  0.0283,  0.0282],
  [-0.0157, -0.0050, -0.0109,  0.0089,  0.0328],
  [ 0.0242, -0.0018,  0.0310,  0.0031,  0.0159]],

 [[ 0.0312,  0.0052,  0.0110, -0.0176, -0.0124],
  [ 0.0298, -0.0070, -0.0084, -0.0162,  0.0050],
  [-0.0187,  0.0009, -0.0152, -0.0054, -0.0138],
  [-0.0121,  0.0246, -0.0189,  0.0197, -0.0245],
  [-0.0059, -0.0008,  0.0148,  0.0035, -0.0021]]],


[[[-0.0278, -0.0189, -0.0001,  0.0078,  0.0058],
  [-0.0223, -0.0091, -0.0242, -0.0079, -0.0060],
  [-0.0011, -0.0226,  0.0131,  0.0025,  0.0025],
  [-0.0037,  0.0197,  0.0054, -0.0147,  0.0132],
  [ 0.0084, -0.0188,  0.0018, -0.0109,  0.0068]],

 [[-0.0178,  0.0052,  0.0028,  0.0008, -0.0143],
  [-0.0159, -0.0098, -0.0077, -0.0176, -0.0225],
  [-0.0074,  0.0093, -0.0334, -0.0188, -0.0210],
  [-0.0173, -0.0351, -0.0008, -0.0335,  0.0040],
  [-0.0177, -0.0130, -0.0046, -0.0328,  0.0100]],

 [[-0.0065, -0.0070, -0.0036, -0.0174, -0.0017],
  [ 0.0125,  0.0041,  0.0013, -0.0177, -0.0235],
  [ 0.0188,  0.0271,  0.0203, -0.0113, -0.0188],
  [-0.0138,  0.0216,  0.0261,  0.0163, -0.0335],
  [-0.0253, -0.0155,  0.0081,  0.0186, -0.0293]],

 ...,

 [[-0.0013,  0.0226, -0.0143, -0.0181, -0.0110],
  [-0.0059,  0.0230,  0.0007, -0.0135,  0.0305],
  [ 0.0010, -0.0048, -0.0214, -0.0147,  0.0299],
  [ 0.0296, -0.0141,  0.0015, -0.0067,  0.0157],
  [ 0.0165, -0.0023,  0.0004, -0.0066, -0.0101]],

 [[-0.0094, -0.0180, -0.0069,  0.0189,  0.0238],
  [-0.0193, -0.0142,  0.0306,  0.0245,  0.0311],
  [ 0.0002,  0.0009,  0.0031,  0.0114,  0.0254],
  [-0.0097, -0.0113,  0.0257,  0.0172,  0.0128],
  [-0.0131,  0.0199,  0.0211, -0.0063,  0.0078]],

 [[-0.0088,  0.0032,  0.0167, -0.0170,  0.0148],
  [-0.0299,  0.0220, -0.0088,  0.0045,  0.0054],
  [-0.0102,  0.0029, -0.0316, -0.0221, -0.0152],
  [ 0.0004, -0.0150, -0.0239, -0.0445,  0.0026],
  [ 0.0199, -0.0088, -0.0149, -0.0240, -0.0297]]],


[[[-0.0101,  0.0143,  0.0036,  0.0237,  0.0108],
  [ 0.0305, -0.0088,  0.0274, -0.0059, -0.0025],
  [-0.0053,  0.0312,  0.0211, -0.0123,  0.0108],
  [ 0.0014,  0.0296, -0.0077,  0.0244,  0.0296],
```

```
  [ 0.0106,  0.0209, -0.0035,  0.0281,  0.0188]],

 [[-0.0122, -0.0220, -0.0154,  0.0026,  0.0045],
  [ 0.0022, -0.0127,  0.0004,  0.0090,  0.0042],
  [-0.0122,  0.0137, -0.0264, -0.0190, -0.0286],
  [-0.0197, -0.0267, -0.0334,  0.0009, -0.0020],
  [-0.0043, -0.0123,  0.0212,  0.0107,  0.0002]],

 [[ 0.0196, -0.0006, -0.0106, -0.0104, -0.0238],
  [-0.0222, -0.0029, -0.0168, -0.0062, -0.0032],
  [-0.0066, -0.0071, -0.0208, -0.0052,  0.0213],
  [ 0.0142,  0.0185, -0.0090,  0.0062, -0.0049],
  [-0.0217,  0.0084,  0.0031, -0.0017, -0.0257]],

 ...,

 [[-0.0089,  0.0030, -0.0095, -0.0181,  0.0098],
  [ 0.0048, -0.0270,  0.0029, -0.0020,  0.0146],
  [-0.0180, -0.0060, -0.0372, -0.0264,  0.0050],
  [-0.0312, -0.0287, -0.0060,  0.0107,  0.0114],
  [ 0.0003, -0.0107,  0.0155, -0.0133,  0.0190]],

 [[ 0.0181, -0.0005,  0.0077, -0.0289, -0.0172],
  [ 0.0022,  0.0335,  0.0017,  0.0233,  0.0041],
  [ 0.0067, -0.0029,  0.0293, -0.0073, -0.0038],
  [ 0.0291, -0.0017,  0.0343,  0.0180,  0.0097],
  [ 0.0224, -0.0130,  0.0222,  0.0089, -0.0209]],

 [[-0.0257, -0.0252,  0.0160, -0.0306, -0.0261],
  [-0.0308, -0.0072, -0.0004, -0.0305, -0.0329],
  [-0.0131, -0.0028, -0.0318,  0.0135,  0.0136],
  [-0.0053, -0.0208, -0.0214, -0.0080, -0.0059],
  [ 0.0092, -0.0084, -0.0151, -0.0181,  0.0187]]],


...,


[[[ 0.0146, -0.0106,  0.0314,  0.0084,  0.0063],
  [-0.0076, -0.0017,  0.0073, -0.0007,  0.0036],
  [-0.0002,  0.0327, -0.0009,  0.0171,  0.0287],
  [ 0.0064,  0.0010, -0.0054,  0.0239,  0.0389],
  [ 0.0252, -0.0170, -0.0017,  0.0241, -0.0107]],

 [[ 0.0223,  0.0297, -0.0138,  0.0068,  0.0329],
  [-0.0072, -0.0044, -0.0180,  0.0241,  0.0194],
  [-0.0173, -0.0186,  0.0021,  0.0078, -0.0024],
  [-0.0168, -0.0010, -0.0008,  0.0069, -0.0253],
  [-0.0314, -0.0041, -0.0017, -0.0050, -0.0231]],

 [[ 0.0157, -0.0002,  0.0278,  0.0051, -0.0015],
  [ 0.0221, -0.0139,  0.0118, -0.0027,  0.0132],
  [ 0.0127, -0.0132, -0.0017,  0.0215, -0.0203],
  [-0.0066, -0.0289, -0.0044,  0.0021,  0.0064],
  [-0.0346, -0.0415,  0.0047, -0.0069, -0.0293]],

 ...,

 [[ 0.0082,  0.0041, -0.0172, -0.0011, -0.0012],
  [-0.0286, -0.0218,  0.0123,  0.0283,  0.0157],
  [-0.0211, -0.0027,  0.0270,  0.0274,  0.0247],
  [ 0.0065, -0.0080,  0.0258,  0.0105, -0.0240],
  [-0.0138, -0.0087,  0.0200,  0.0173,  0.0178]],

 [[ 0.0017,  0.0286,  0.0106,  0.0318,  0.0270],
  [ 0.0231,  0.0275, -0.0030, -0.0161,  0.0162],
  [ 0.0076, -0.0128,  0.0093, -0.0083,  0.0174],
  [ 0.0197, -0.0110,  0.0140,  0.0051,  0.0147],
  [-0.0071,  0.0218,  0.0024,  0.0122, -0.0210]],

 [[-0.0253,  0.0061,  0.0039,  0.0189, -0.0090],
  [-0.0063,  0.0053, -0.0142, -0.0157,  0.0043],
  [-0.0126,  0.0051,  0.0109,  0.0023,  0.0058],
  [-0.0089, -0.0002,  0.0024, -0.0050, -0.0253],
  [ 0.0174,  0.0121,  0.0116,  0.0162,  0.0262]]],
```

```
          [[[ 0.0172,  0.0126,  0.0100,  0.0059,  0.0036],
            [-0.0197,  0.0210, -0.0211,  0.0126,  0.0097],
            [ 0.0175, -0.0298,  0.0207,  0.0038,  0.0226],
            [-0.0257,  0.0031, -0.0101,  0.0199,  0.0228],
            [-0.0008,  0.0214,  0.0069,  0.0174,  0.0204]],

           [[-0.0019,  0.0001, -0.0237, -0.0137,  0.0221],
            [-0.0057,  0.0054, -0.0145, -0.0123,  0.0290],
            [-0.0072, -0.0206, -0.0106, -0.0159, -0.0157],
            [-0.0039,  0.0419,  0.0016,  0.0301,  0.0364],
            [ 0.0458,  0.0337,  0.0323,  0.0084,  0.0011]],

           [[-0.0050,  0.0097, -0.0112,  0.0077,  0.0087],
            [ 0.0302,  0.0221, -0.0071, -0.0212, -0.0049],
            [ 0.0116,  0.0035, -0.0128, -0.0257,  0.0022],
            [ 0.0068, -0.0203, -0.0134, -0.0182,  0.0080],
            [ 0.0189, -0.0216, -0.0219, -0.0170, -0.0241]],

           ...,

           [[-0.0133,  0.0077,  0.0121, -0.0194, -0.0050],
            [-0.0042, -0.0073,  0.0096,  0.0046,  0.0133],
            [-0.0079,  0.0097,  0.0051, -0.0239, -0.0108],
            [-0.0233, -0.0069, -0.0052, -0.0077,  0.0066],
            [-0.0276, -0.0181, -0.0096,  0.0043,  0.0186]],

           [[ 0.0119, -0.0045, -0.0089,  0.0061,  0.0007],
            [-0.0084, -0.0101, -0.0193,  0.0012, -0.0227],
            [ 0.0083,  0.0050, -0.0331, -0.0105, -0.0244],
            [ 0.0233,  0.0053, -0.0208,  0.0067, -0.0168],
            [-0.0118, -0.0133,  0.0058,  0.0095,  0.0014]],

           [[ 0.0390,  0.0006,  0.0045,  0.0024,  0.0007],
            [ 0.0380,  0.0140, -0.0007,  0.0197,  0.0199],
            [ 0.0333,  0.0126,  0.0276, -0.0037,  0.0015],
            [ 0.0128,  0.0159, -0.0054,  0.0051,  0.0158],
            [ 0.0254,  0.0217,  0.0326,  0.0286, -0.0147]]],


          [[[-0.0289, -0.0236,  0.0024,  0.0161,  0.0186],
            [-0.0013, -0.0099, -0.0025,  0.0384,  0.0392],
            [ 0.0120, -0.0132,  0.0210,  0.0005,  0.0127],
            [ 0.0039, -0.0174,  0.0002, -0.0096, -0.0018],
            [-0.0012, -0.0184,  0.0073,  0.0112, -0.0155]],

           [[ 0.0196,  0.0106,  0.0342,  0.0165,  0.0107],
            [ 0.0162,  0.0005,  0.0205, -0.0180, -0.0140],
            [-0.0018,  0.0012,  0.0114, -0.0114, -0.0114],
            [ 0.0210, -0.0106, -0.0200, -0.0057,  0.0089],
            [-0.0275, -0.0137,  0.0291, -0.0159, -0.0096]],

           [[-0.0037,  0.0028, -0.0214, -0.0153, -0.0044],
            [-0.0076,  0.0045, -0.0079, -0.0012,  0.0078],
            [-0.0163,  0.0109,  0.0162, -0.0103,  0.0037],
            [ 0.0240, -0.0117,  0.0149,  0.0106, -0.0286],
            [-0.0092,  0.0112,  0.0305,  0.0296,  0.0092]],

           ...,

           [[-0.0215, -0.0296, -0.0198,  0.0132, -0.0057],
            [-0.0107, -0.0284, -0.0098, -0.0233,  0.0095],
            [ 0.0186,  0.0200, -0.0198,  0.0045,  0.0081],
            [ 0.0121,  0.0238,  0.0146,  0.0069,  0.0311],
            [-0.0134,  0.0170,  0.0290,  0.0097,  0.0009]],

           [[-0.0215,  0.0126,  0.0055,  0.0049, -0.0154],
            [ 0.0046,  0.0164,  0.0128,  0.0013, -0.0185],
            [-0.0145,  0.0162,  0.0049,  0.0136,  0.0092],
            [-0.0230,  0.0279,  0.0092,  0.0285,  0.0086],
            [-0.0232,  0.0142,  0.0106, -0.0103,  0.0266]],

           [[ 0.0057,  0.0198,  0.0141, -0.0032, -0.0309],
            [-0.0217,  0.0151, -0.0020, -0.0210, -0.0308],
            [-0.0144, -0.0190,  0.0108, -0.0044, -0.0378],
            [-0.0128,  0.0069,  0.0075, -0.0297, -0.0097],
            [ 0.0190, -0.0062,  0.0018, -0.0086,  0.0193]]]], device='cuda:0')), ('conv7.bias', tensc
r([ 0.0100,  0.0142, -0.0061, -0.0228,  0.0169, -0.0248,  0.0145,  0.0224,
```

```
        -0.0235, -0.0075,  0.0202,  0.0100,  0.0000, -0.0237, -0.0115,  0.0149,
         0.0068, -0.0104,  0.0174, -0.0115, -0.0201, -0.0127,  0.0025,  0.0003,
         0.0089, -0.0123,  0.0230, -0.0068, -0.0095,  0.0135, -0.0157, -0.0033,
         0.0075,  0.0065,  0.0154,  0.0150,  0.0124,  0.0215,  0.0004,  0.0050,
         0.0016, -0.0037, -0.0250,  0.0198,  0.0138, -0.0022, -0.0160,  0.0007,
         0.0156,  0.0165,  0.0130,  0.0016, -0.0204, -0.0011, -0.0145, -0.0155,
         0.0021, -0.0132, -0.0005, -0.0091, -0.0109, -0.0151,  0.0176,  0.0241,
         0.0011, -0.0010,  0.0113,  0.0241, -0.0202,  0.0091,  0.0083, -0.0249,
         0.0152, -0.0173,  0.0188,  0.0117,  0.0080,  0.0143, -0.0056, -0.0234,
        -0.0147, -0.0181, -0.0207,  0.0041,  0.0144, -0.0103,  0.0164, -0.0206,
         0.0113,  0.0085, -0.0025, -0.0191,  0.0002,  0.0182, -0.0012,  0.0156,
        -0.0211,  0.0055,  0.0061, -0.0130, -0.0119, -0.0188, -0.0185, -0.0059,
         0.0036, -0.0139, -0.0130, -0.0071, -0.0207, -0.0192, -0.0241,  0.0193,
        -0.0007, -0.0040, -0.0041,  0.0152, -0.0065, -0.0026,  0.0073, -0.0184,
         0.0107,  0.0132, -0.0117,  0.0242,  0.0160,  0.0244, -0.0111, -0.0249],
       device='cuda:0')), ('conv7_bn.weight', tensor([0.6614, 0.4155, 0.9904, 0.3792, 0.7880, 0.752
6, 0.4111, 0.9713, 0.5663,
        0.6527, 0.1365, 0.5185, 0.9057, 0.9388, 0.1845, 0.0719, 0.0809, 0.0717,
        0.5204, 0.3690, 0.4020, 0.7467, 0.8893, 0.1712, 0.5951, 0.9172, 0.4815,
        0.5541, 0.2510, 0.3909, 0.3932, 0.4466, 0.6359, 0.0393, 0.7426, 0.9332,
        0.8662, 0.3551, 0.7460, 0.4015, 0.6886, 0.5012, 0.1008, 0.2294, 0.3195,
        0.8532, 0.8472, 0.1624, 0.8710, 0.9044, 0.8649, 0.8986, 0.2568, 0.2410,
        0.0663, 0.9058, 0.3119, 0.8452, 0.7440, 0.8052, 0.5288, 0.4167, 0.3906,
        0.1816, 0.8933, 0.3376, 0.4249, 0.1075, 0.8108, 0.9500, 0.8477, 0.6723,
        0.8316, 0.0805, 0.8031, 0.4995, 0.7180, 0.3398, 0.5978, 0.6673, 0.7559,
        0.6617, 0.6499, 0.9034, 0.8272, 0.4201, 0.6790, 0.8351, 0.7727, 0.7906,
        0.6372, 0.7390, 0.8721, 0.9232, 0.6296, 0.2807, 0.3719, 0.9308, 0.7221,
        0.4371, 0.1032, 0.1858, 0.1179, 0.1657, 0.5710, 0.1181, 0.1225, 0.7445,
        0.1390, 0.4018, 0.1109, 0.5613, 0.5336, 0.8294, 0.8589, 0.1047, 0.1803,
        0.2644, 0.2803, 0.1447, 0.9477, 0.8822, 0.6214, 0.2961, 0.8770, 0.6787,
        0.6169, 0.4554], device='cuda:0')), ('conv7_bn.bias', tensor([-0.0570, -0.0516, -0.0557, -
0.0539, -0.0756, -0.0724, -0.0347, -0.0642,
        -0.0375, -0.0813, -0.0306, -0.0579, -0.0555, -0.0672, -0.0520, -0.0305,
        -0.0438, -0.0408, -0.0368, -0.0551, -0.0574, -0.0526, -0.0727, -0.0454,
        -0.0678, -0.0877, -0.0601, -0.0446, -0.0363, -0.0642, -0.0469, -0.0474,
        -0.0707, -0.0228, -0.0400, -0.0492, -0.0625, -0.0502, -0.0649, -0.0439,
        -0.0760, -0.0786, -0.0336, -0.0480, -0.0522, -0.0675, -0.0629, -0.0296,
        -0.0575, -0.0423, -0.0645, -0.0536, -0.0441, -0.0272, -0.0249, -0.0598,
        -0.0372, -0.0566, -0.0628, -0.0498, -0.0415, -0.0447, -0.0564, -0.0300,
        -0.0577, -0.0542, -0.0397, -0.0398, -0.0511, -0.0722, -0.0608, -0.0699,
        -0.0353, -0.0234, -0.0367, -0.0420, -0.0541, -0.0420, -0.0490, -0.0267,
        -0.0542, -0.0368, -0.0585, -0.0762, -0.0526, -0.0479, -0.0512, -0.0483,
        -0.0567, -0.0447, -0.0552, -0.0307, -0.1032, -0.0679, -0.0670, -0.0506,
        -0.0547, -0.0584, -0.0714, -0.0453, -0.0324, -0.0527, -0.0421, -0.0541,
        -0.0517, -0.0396, -0.0511, -0.0361, -0.0353, -0.0385, -0.0409, -0.0392,
        -0.0509, -0.0603, -0.0379, -0.0456, -0.0301, -0.0472, -0.0513, -0.0342,
        -0.0485, -0.0672, -0.0605, -0.0469, -0.0671, -0.0533, -0.0676, -0.0377],
       device='cuda:0')), ('conv7_bn.running_mean', tensor([ 0.0085, -0.6893, -0.3168, -0.2323, -0
.0634, -1.1945,  0.3922,  0.6649,
        -0.2367,  0.5537,  0.3511,  0.4410,  0.1140,  0.4734, -0.0603, -1.6170,
        -1.6945,  0.1783, -1.7040,  1.6515,  1.0635, -0.4377, -1.0317, -0.2911,
        -0.5559, -0.5542,  0.0999, -0.2247, -1.8667,  0.2837, -0.2611,  1.2784,
        -0.5555,  0.9236, -1.6844, -1.0301, -0.5494,  1.0526,  0.7149, -0.5787,
        -0.1166, -0.0847,  1.0920,  1.1684,  0.4965, -0.9148, -1.8205,  1.6150,
        -0.8575, -1.5323, -0.4999, -0.2554,  1.0102, -1.1762,  0.5680,  0.2824,
        -1.8454, -0.5707, -0.1685, -0.7056, -2.6167,  1.8677,  0.0280, -1.7920,
         0.1124, -0.8969, -0.6997, -0.5419, -1.0995, -0.2173, -1.5899, -0.0191,
        -3.7384, -0.6104, -1.8464,  1.0430, -0.1539, -0.8480,  1.3419, -2.9317,
         1.0828,  1.9544,  0.1923,  0.4396,  0.5137, -0.9347,  0.6975, -1.2235,
        -0.2068, -0.6373, -0.8300, -1.8440, -0.1930,  0.0736, -1.0386,  2.0968,
        -0.7358,  1.8443, -0.1811, -1.2620, -0.8410,  0.6542,  2.3954, -0.4087,
         0.3686,  1.4950,  1.3186, -1.3282, -2.1248, -0.6441,  2.8355, -1.2551,
         1.9055, -1.7239, -1.5197,  0.6691,  1.3010,  1.4019,  0.4985,  0.7026,
        -1.5255, -1.5002,  2.0425,  0.2324,  0.3264, -0.2769,  0.7101, -0.4580],
       device='cuda:0')), ('conv7_bn.running_var', tensor([0.4435, 0.5521, 0.5112, 0.4202, 0.4538,
0.5023, 0.6018, 0.4553, 0.6052,
        0.5023, 0.4140, 0.4578, 0.3872, 0.4425, 0.4335, 0.5180, 0.6718, 0.3827,
        1.0202, 0.7502, 0.5896, 0.4525, 0.5268, 0.3919, 0.4676, 0.5174, 0.3992,
        0.5704, 0.6242, 0.5040, 0.4594, 0.5113, 0.5238, 0.4610, 0.6170, 0.4337,
        0.5798, 0.4642, 0.3879, 0.5180, 0.3824, 0.5408, 0.3666, 0.4121, 0.3826,
        0.4854, 0.6412, 0.4833, 0.3557, 0.4407, 0.6083, 0.4757, 0.5656, 0.4563,
        0.3867, 0.5755, 0.5076, 0.5334, 0.4591, 0.5147, 0.6899, 0.4919, 0.4565,
        0.6037, 0.4268, 0.4650, 0.5465, 0.4962, 0.4759, 0.3920, 0.4966, 0.5041,
        0.9598, 0.4367, 0.6096, 0.5837, 0.4829, 0.5289, 0.6640, 1.3719, 0.6074,
        0.6211, 0.5407, 0.4022, 0.4433, 0.4926, 0.5182, 0.5843, 0.4942, 0.6217,
        0.4092, 0.5874, 0.4978, 0.6066, 0.5312, 0.5285, 0.4141, 0.7758, 0.4435,
        0.4596, 0.4288, 0.3785, 0.5530, 0.4025, 0.4193, 0.4393, 0.5340, 0.4412,
```

```
        0.9401, 0.6387, 0.7066, 0.5298, 0.5353, 0.6009, 0.8468, 0.5280, 0.4843,
        0.5738, 0.4740, 0.4224, 0.5765, 0.5370, 0.6869, 0.4617, 0.5459, 0.4267,
        0.5735, 0.6179], device='cuda:0')), ('conv7_bn.num_batches_tracked', tensor(8160, device='c
uda:0')), ('conv8.weight', tensor([[[[-0.0141,  0.0049, -0.0139],
          [ 0.0172, -0.0058, -0.0118],
          [ 0.0261, -0.0008,  0.0341]],

         [[-0.0292, -0.0089,  0.0266],
          [-0.0313,  0.0044,  0.0220],
          [ 0.0124, -0.0224, -0.0285]],

         [[ 0.0514,  0.0420,  0.0464],
          [ 0.0572,  0.0022,  0.0257],
          [-0.0063,  0.0288, -0.0015]],

         ...,

         [[-0.0352, -0.0171, -0.0044],
          [-0.0317, -0.0045, -0.0205],
          [ 0.0072,  0.0009,  0.0167]],

         [[ 0.0137, -0.0160, -0.0011],
          [ 0.0215,  0.0287, -0.0096],
          [-0.0226,  0.0203, -0.0239]],

         [[ 0.0197,  0.0106,  0.0126],
          [ 0.0060, -0.0309, -0.0058],
          [-0.0449, -0.0465, -0.0452]]],


        [[[-0.0249,  0.0237,  0.0206],
          [-0.0184, -0.0203,  0.0366],
          [-0.0272, -0.0178,  0.0311]],

         [[ 0.0116, -0.0005,  0.0146],
          [ 0.0143, -0.0120,  0.0163],
          [ 0.0166, -0.0018,  0.0279]],

         [[ 0.0030,  0.0032, -0.0044],
          [ 0.0046, -0.0086,  0.0094],
          [ 0.0250,  0.0131,  0.0174]],

         ...,

         [[-0.0006,  0.0027,  0.0103],
          [-0.0069, -0.0073, -0.0295],
          [ 0.0224,  0.0063, -0.0188]],

         [[-0.0227,  0.0129,  0.0022],
          [-0.0226, -0.0044,  0.0142],
          [ 0.0146, -0.0066, -0.0078]],

         [[ 0.0263,  0.0328,  0.0389],
          [ 0.0076,  0.0202, -0.0169],
          [ 0.0017,  0.0288,  0.0295]]],


        [[[-0.0357, -0.0117,  0.0097],
          [ 0.0038,  0.0110, -0.0203],
          [ 0.0113, -0.0168,  0.0192]],

         [[-0.0031, -0.0099, -0.0265],
          [ 0.0176, -0.0129,  0.0271],
          [-0.0044,  0.0067,  0.0089]],

         [[ 0.0287,  0.0050,  0.0236],
          [ 0.0132,  0.0180, -0.0022],
          [ 0.0305,  0.0063,  0.0375]],

         ...,

         [[-0.0116, -0.0080,  0.0243],
          [-0.0073, -0.0077,  0.0229],
          [ 0.0232,  0.0009,  0.0353]],

         [[-0.0080,  0.0098,  0.0106],
          [-0.0168, -0.0125,  0.0113]
```

```
  [ 0.0100,  0.0125,  0.0115],
  [-0.0303, -0.0212, -0.0320]],

 [[-0.0318, -0.0136,  0.0129],
  [ 0.0041,  0.0054,  0.0187],
  [ 0.0138, -0.0327, -0.0111]]],


...,


[[[-0.0271,  0.0021, -0.0188],
  [ 0.0181,  0.0170, -0.0020],
  [ 0.0123, -0.0096,  0.0059]],

 [[-0.0198, -0.0214, -0.0500],
  [-0.0051, -0.0247, -0.0086],
  [ 0.0160, -0.0012, -0.0087]],

 [[ 0.0201, -0.0138,  0.0231],
  [ 0.0212,  0.0288, -0.0297],
  [ 0.0419,  0.0362, -0.0129]],

 ...,

 [[-0.0270, -0.0002, -0.0088],
  [ 0.0264, -0.0224, -0.0058],
  [ 0.0214,  0.0031, -0.0295]],

 [[-0.0264, -0.0152, -0.0187],
  [-0.0281,  0.0089,  0.0373],
  [ 0.0291, -0.0099,  0.0293]],

 [[ 0.0004, -0.0074, -0.0312],
  [ 0.0282, -0.0274, -0.0139],
  [-0.0021,  0.0140,  0.0253]]],


[[[ 0.0117, -0.0134,  0.0133],
  [-0.0185, -0.0276, -0.0210],
  [ 0.0081,  0.0018, -0.0360]],

 [[-0.0070,  0.0326,  0.0002],
  [-0.0100, -0.0164, -0.0009],
  [-0.0262,  0.0113,  0.0157]],

 [[ 0.0199, -0.0205, -0.0036],
  [ 0.0237, -0.0182, -0.0003],
  [-0.0407,  0.0078, -0.0439]],

 ...,

 [[ 0.0248,  0.0196, -0.0160],
  [-0.0012, -0.0146, -0.0178],
  [-0.0108, -0.0349,  0.0039]],

 [[-0.0087, -0.0125, -0.0323],
  [ 0.0093, -0.0328,  0.0069],
  [-0.0275, -0.0363, -0.0478]],

 [[-0.0204,  0.0004, -0.0137],
  [ 0.0243, -0.0342,  0.0004],
  [-0.0058, -0.0375,  0.0063]]],


[[[-0.0228,  0.0075,  0.0112],
  [-0.0161, -0.0136,  0.0156],
  [-0.0378, -0.0358, -0.0395]],

 [[ 0.0096,  0.0185,  0.0040],
  [ 0.0278,  0.0242, -0.0101],
  [ 0.0216,  0.0065, -0.0079]],

 [[ 0.0174, -0.0136, -0.0126],
  [-0.0213,  0.0114, -0.0162],
  [-0.0230, -0.0004,  0.0227]],
```

```
          ...,

          [[ 0.0344,  0.0110,  0.0198],
           [ 0.0073, -0.0067,  0.0169],
           [ 0.0265,  0.0142,  0.0097]],

          [[-0.0513, -0.0330, -0.0374],
           [-0.0324, -0.0530, -0.0385],
           [-0.0509, -0.0173, -0.0300]],

          [[ 0.0174,  0.0141, -0.0252],
           [ 0.0258, -0.0136,  0.0083],
           [-0.0224, -0.0196, -0.0366]]]], device='cuda:0')), ('conv8.bias', tensor([ 0.0075,  0.003
4, -0.0129,  0.0079, -0.0112, -0.0290, -0.0216,  0.0214,
         -0.0083, -0.0163,  0.0157, -0.0204, -0.0105, -0.0051,  0.0185, -0.0223,
          0.0043, -0.0092, -0.0244, -0.0180,  0.0096, -0.0240, -0.0023,  0.0197,
          0.0100,  0.0282,  0.0152, -0.0174, -0.0125,  0.0278, -0.0037,  0.0015,
          0.0023,  0.0027,  0.0215, -0.0133, -0.0191,  0.0025,  0.0193,  0.0048,
         -0.0212,  0.0191,  0.0147, -0.0164,  0.0117,  0.0193,  0.0124, -0.0158,
          0.0023, -0.0098, -0.0121, -0.0177, -0.0042,  0.0107,  0.0124,  0.0123,
         -0.0120, -0.0279, -0.0104, -0.0027,  0.0089,  0.0106,  0.0065,  0.0219,
          0.0260,  0.0219,  0.0229,  0.0240, -0.0257, -0.0124, -0.0080,  0.0059,
         -0.0120,  0.0013, -0.0137, -0.0110,  0.0071, -0.0114, -0.0238,  0.0264,
         -0.0036, -0.0134,  0.0232,  0.0275, -0.0291,  0.0166,  0.0107, -0.0207,
          0.0257,  0.0035, -0.0169,  0.0124,  0.0183, -0.0164, -0.0259,  0.0029,
          0.0103, -0.0079, -0.0053, -0.0081, -0.0158, -0.0069, -0.0274,  0.0075,
         -0.0239, -0.0246, -0.0112, -0.0102,  0.0292,  0.0085,  0.0193, -0.0031,
          0.0190, -0.0219,  0.0274, -0.0238, -0.0066,  0.0220, -0.0057,  0.0096,
         -0.0235,  0.0213,  0.0292, -0.0109, -0.0223, -0.0155, -0.0220,  0.0200,
          0.0231,  0.0068, -0.0064, -0.0150,  0.0201, -0.0086,  0.0031, -0.0116,
         -0.0208,  0.0022, -0.0274, -0.0084,  0.0135,  0.0101,  0.0257, -0.0142,
          0.0100,  0.0249, -0.0232, -0.0178,  0.0149, -0.0182,  0.0064,  0.0074,
          0.0102, -0.0263, -0.0151,  0.0059, -0.0004, -0.0057, -0.0006, -0.0006,
          0.0256,  0.0021,  0.0080,  0.0082,  0.0279, -0.0271,  0.0144, -0.0234,
         -0.0266, -0.0181,  0.0095,  0.0165, -0.0079,  0.0219,  0.0036,  0.0045,
          0.0159,  0.0086,  0.0120,  0.0017, -0.0001, -0.0268,  0.0168, -0.0016,
         -0.0064, -0.0151,  0.0195,  0.0130, -0.0062,  0.0231,  0.0229, -0.0135,
          0.0054,  0.0049,  0.0040, -0.0205,  0.0149,  0.0230,  0.0269, -0.0106,
          0.0174,  0.0290,  0.0095,  0.0174, -0.0104,  0.0212,  0.0085,  0.0275,
         -0.0112, -0.0155,  0.0060,  0.0155, -0.0269,  0.0198,  0.0157,  0.0016,
         -0.0130,  0.0136, -0.0124,  0.0114,  0.0162,  0.0183, -0.0047,  0.0141,
         -0.0268,  0.0083, -0.0028,  0.0073,  0.0246,  0.0193, -0.0087,  0.0261,
          0.0219, -0.0275, -0.0006,  0.0072,  0.0281, -0.0098, -0.0135,  0.0182,
         -0.0142, -0.0079,  0.0007, -0.0262,  0.0242,  0.0079,  0.0290,  0.0050,
          0.0146, -0.0118,  0.0129,  0.0231,  0.0054, -0.0024,  0.0078, -0.0046],
        device='cuda:0')), ('conv8_bn.weight', tensor([0.3950, 0.7576, 0.5905, 0.6799, 0.6796, 0.353
5, 0.7547, 0.6946, 0.9597,
         0.3164, 0.8478, 0.7108, 0.3047, 0.2725, 0.9336, 0.9314, 0.2175, 0.9146,
         0.5696, 0.5498, 0.4104, 0.9328, 0.1461, 0.2531, 0.9410, 0.9415, 0.7040,
         0.7412, 0.5626, 0.3544, 0.8529, 0.6380, 0.5560, 0.8623, 0.4069, 0.3309,
         0.6652, 0.1803, 0.6954, 0.0005, 0.7678, 0.3999, 0.9586, 0.1532, 0.1982,
         0.0972, 0.0673, 0.6495, 0.2478, 0.7578, 0.6789, 0.2671, 0.3943, 0.7894,
         0.1086, 0.8967, 0.6110, 0.0806, 0.9573, 0.9392, 0.0667, 0.3357, 0.0936,
         0.4446, 0.1424, 0.4442, 0.4998, 0.7893, 0.7255, 0.1741, 0.8295, 0.5483,
         0.3853, 0.7085, 0.6361, 0.6862, 0.1408, 0.2989, 0.3931, 0.7460, 0.2605,
         0.2528, 0.2128, 0.4557, 0.2234, 0.8609, 0.3460, 0.5216, 0.6349, 0.6334,
         0.8032, 0.3445, 0.0460, 0.6077, 0.4955, 0.4262, 0.6490, 0.0284, 0.6413,
         0.3912, 0.8455, 0.0381, 0.5338, 0.4125, 0.8561, 0.7806, 0.6132, 0.8318,
         0.6492, 0.5468, 0.8148, 0.4559, 0.7535, 0.4234, 0.3766, 0.4038, 0.3629,
         0.6638, 0.1063, 0.1898, 0.5951, 0.3464, 0.7569, 0.5719, 0.5711, 0.5137,
         0.6547, 0.3385, 0.3655, 0.7504, 0.4886, 0.3398, 0.2553, 0.0799, 0.4842,
         0.6521, 0.1403, 0.2975, 0.5716, 0.2254, 0.5100, 0.4427, 0.0926, 0.2179,
         0.9044, 0.7377, 0.6296, 0.2649, 0.3159, 0.9664, 0.2183, 0.3049, 0.9575,
         0.7525, 0.8259, 0.0490, 0.8557, 0.9373, 0.9111, 0.7795, 0.5140, 0.7007,
         0.8222, 0.0964, 0.7161, 0.3359, 0.0626, 0.6447, 0.6315, 0.6860, 0.1301,
         0.3627, 0.3577, 0.2619, 0.7785, 0.4801, 0.7705, 0.4820, 0.0514, 0.0343,
         0.9097, 0.6223, 0.3970, 0.3110, 0.3068, 0.7010, 0.9594, 0.3991, 0.8051,
         0.2488, 0.4667, 0.1273, 0.9426, 0.8612, 0.8200, 0.7114, 0.9940, 0.4016,
         0.0915, 0.7168, 0.0395, 0.8521, 0.5641, 0.1923, 0.7882, 0.2036, 0.0707,
         0.5435, 0.4907, 0.3884, 0.4210, 0.7085, 0.2531, 0.3926, 0.9163, 0.3457,
         0.2511, 0.6767, 0.6143, 0.3079, 0.8796, 0.5486, 0.9796, 0.0995, 0.7908,
         0.0601, 0.8239, 0.8751, 0.3608, 0.1236, 0.7855, 0.6883, 0.3833, 0.2078,
         0.6624, 0.6915, 0.8788, 0.6599, 0.4155, 0.2652, 0.9215, 0.5686, 0.6340,
         0.8535, 0.7445, 0.0882, 0.8090, 0.4102, 0.6402, 0.3177, 0.8554, 0.8581,
         0.1113, 0.9505, 0.5283, 0.7962], device='cuda:0')), ('conv8_bn.bias', tensor([-0.0373, -0.0
275, -0.0398, -0.0246, -0.0438, -0.0305, -0.0402, -0.0150,
         -0.0322, -0.0293, -0.0286, -0.0086, -0.0249, -0.0591, -0.0184, -0.0251,
          0.0116, -0.0603, -0.0518, -0.0207, -0.0289, -0.0292, -0.0097, -0.0311,
```

```
        -0.0116, -0.0602, -0.0518, -0.0207, -0.0299, -0.0292, -0.0097, -0.0311,
        -0.0322, -0.0317, -0.0300, -0.0213, -0.0384, -0.0292, -0.0339, -0.0346,
        -0.0426, -0.0053, -0.0283, -0.0402, -0.0037, -0.0382, -0.0325, -0.0028,
        -0.0343, -0.0462, -0.0340, -0.0285, -0.0168, -0.0276, -0.0226, -0.0253,
        -0.0362, -0.0311, -0.0184, -0.0375, -0.0419, -0.0308, -0.0050, -0.0298,
        -0.0391, -0.0184, -0.0263, -0.0456, -0.0234, -0.0357, -0.0277, -0.0061,
        -0.0385, -0.0387, -0.0675, -0.0284, -0.0258, -0.0433, -0.0367, -0.0509,
        -0.0268, -0.0155, -0.0470, -0.0559, -0.0269, -0.0357, -0.0223, -0.0291,
        -0.0153, -0.0526, -0.0375, -0.0196, -0.0378, -0.0199, -0.0451, -0.0334,
        -0.0248, -0.0362, -0.0793, -0.0023, -0.0321, -0.0486, -0.0179, -0.0146,
        -0.0397, -0.0111, -0.0759, -0.0270, -0.0254, -0.0253, -0.0153, -0.0138,
        -0.0239, -0.0625, -0.0109, -0.0315, -0.0320, -0.0514, -0.0357, -0.0467,
        -0.0421, -0.0231, -0.0422, -0.0244, -0.0386, -0.0543, -0.0396, -0.0358,
        -0.0164, -0.0241, -0.0214, -0.0168, -0.0397, -0.0481, -0.0334, -0.0333,
        -0.0176, -0.0409, -0.0203, -0.0394, -0.0141, -0.0125, -0.0300, -0.0558,
        -0.0147, -0.0235, -0.0477, -0.0122, -0.0202, -0.0446, -0.0357, -0.0325,
        -0.0456, -0.0451, -0.0129, -0.0139, -0.0158, -0.0310, -0.0299, -0.0263,
        -0.0332, -0.0381, -0.0279, -0.0137, -0.0406, -0.0254, -0.0500, -0.0476,
        -0.0335, -0.0307, -0.0261, -0.0369, -0.0373, -0.0343, -0.0168, -0.0281,
        -0.0313, -0.0367, -0.0076, -0.0210, -0.0385, -0.0306, -0.0371, -0.0383,
        -0.0405, -0.0348, -0.0325, -0.0295, -0.0421, -0.0142, -0.0567, -0.0187,
        -0.0498, -0.0425, -0.0137, -0.0017, -0.0123, -0.0705, -0.0437, -0.0184,
        -0.0279, -0.0320, -0.0278, -0.0103, -0.0255, -0.0254, -0.0406, -0.0155,
        -0.0128, -0.0371, -0.0522, -0.0365, -0.0241, -0.0143, -0.0236, -0.0522,
        -0.0348, -0.0176, -0.0228, -0.0491, -0.0241, -0.0209, -0.0403, -0.0093,
        -0.0262, -0.0180, -0.0414, -0.0541, -0.0325, -0.0101, -0.0154, -0.0374,
        -0.0568, -0.0328, -0.0172, -0.0312, -0.0232, -0.0345, -0.0649, -0.0202,
        -0.0372, -0.0480, -0.0566, -0.0340, -0.0208, -0.0409, -0.0141, -0.0282,
        -0.0344, -0.0406, -0.0414, -0.0328, -0.0357, -0.0188, -0.0316, -0.0206,
        -0.0583, -0.0326, -0.0309, -0.0295, -0.0262, -0.0494, -0.0296, -0.0172],
       device='cuda:0')), ('conv8_bn.running_mean', tensor([-0.2786, -0.2291, -0.6259, -1.6281, -0
.6431,  0.6817, -0.1511, -0.1967,
        -1.1781, -1.1682,  0.5955,  0.2445,  0.0404,  0.8720, -0.4565,  0.2917,
         0.3788,  0.3880, -0.9341, -0.2953,  0.3771, -1.1873, -0.2454, -1.2068,
        -0.2289, -0.4326,  0.1549, -1.7095, -0.6786,  0.1394, -0.3512, -1.4212,
         0.5891, -0.4728,  0.1216,  0.0235, -0.1428, -0.1836, -1.4761, -0.6676,
        -0.6855, -0.4030, -0.4837, -0.2035,  0.0713,  1.1983, -0.2757,  0.1920,
        -0.7023, -1.1963, -0.5880, -0.3182, -0.6456,  0.1024, -1.1766, -0.7197,
        -1.0457, -1.1110, -0.9935, -0.9785,  0.3215, -0.6740, -0.9832, -0.8526,
         0.2992, -0.5884,  0.0598, -0.8956, -1.0264, -1.0765, -0.2176, -0.9508,
        -0.0053, -0.9000,  0.4957,  0.1841, -0.0029, -0.0559, -0.0604, -1.3111,
         0.0795,  0.8656,  0.5257,  0.4005,  0.2891, -0.5099,  0.5881,  0.8331,
        -0.5051, -1.0429,  0.1695, -0.6792,  1.0827, -0.7057, -1.3561, -0.2218,
        -0.2010,  1.4802,  0.3477, -0.4864, -0.5040,  0.8379, -1.1692,  1.0380,
        -0.6810,  1.3512, -0.4677, -0.2695, -1.2693,  0.1313, -0.1570,  0.1815,
        -0.1112, -0.8690, -0.7141,  0.1445,  0.0872,  0.1551,  0.9804,  0.1957,
        -0.9433, -0.8856, -0.6447, -0.2293, -0.5418, -0.4199, -0.8266,  0.4708,
        -0.1017,  0.2325, -1.1559,  0.3117, -0.2733,  1.7489, -1.1239,  0.2769,
        -0.6714, -0.7702, -1.1205, -0.6732, -1.6440,  0.1518, -0.3583,  0.7342,
         0.2554, -0.2487, -0.5838, -1.4014, -0.9580, -0.7061, -0.5410, -0.1319,
        -0.9977, -0.0822, -1.6955,  0.2826, -1.1282,  0.2483, -0.6010, -1.9393,
         0.7010, -0.2225, -0.4329, -0.1255,  0.3968, -0.3858, -0.1927,  0.5291,
         1.4053,  1.5808,  0.5931, -0.6130,  0.6280,  0.1280, -0.8753,  0.4241,
         0.4034,  1.1461,  1.5751,  1.2834, -0.1860,  1.2223, -0.2076,  0.4301,
        -0.0984,  0.1689, -0.9701,  0.2258, -1.5617,  0.8372,  0.0823,  0.3151,
        -0.5727,  1.2629, -1.4451, -1.0767, -1.9437,  1.4698, -0.0143, -0.5366,
        -0.1786,  0.2032,  0.1978, -0.2331, -1.3101, -1.1159,  0.6279,  0.5065,
         0.3075, -0.6971, -0.9403,  0.0429, -0.1682, -0.8233,  0.1109,  0.0057,
         0.0613, -1.5974,  1.4596, -0.1636, -1.1752, -0.3845, -1.3340,  0.3755,
         0.2048,  1.5776, -0.4800, -0.6243,  0.4022, -0.0944,  0.6003, -0.9538,
         1.6581,  1.4565,  0.0450,  0.0072, -0.9844,  0.8634,  1.2377,  1.3242,
        -1.3760, -0.1383,  0.4128,  1.4198,  0.7797,  0.0584,  0.7847, -0.1725,
         0.0274, -0.7781,  1.0917, -0.2637, -0.2097,  0.1816, -1.5176, -0.7542],
       device='cuda:0')), ('conv8_bn.running_var', tensor([0.2567, 0.2866, 0.3114, 0.5239, 0.2021,
0.1691, 0.1638, 0.2081, 0.2614,
        0.2973, 0.3053, 0.2265, 0.2416, 0.3292, 0.3777, 0.2065, 0.2491, 0.2028,
        0.1946, 0.1810, 0.1820, 0.2881, 0.1944, 0.2220, 0.1959, 0.2369, 0.1748,
        0.4579, 0.2436, 0.2487, 0.2967, 0.2509, 0.1828, 0.2481, 0.1869, 0.3096,
        0.2931, 0.2049, 0.2561, 0.1826, 0.2820, 0.1726, 0.1355, 0.2463, 0.1408,
        0.3650, 0.2250, 0.2127, 0.3287, 0.2001, 0.2178, 0.2632, 0.3061, 0.1660,
        0.2996, 0.2464, 0.2455, 0.3266, 0.2838, 0.2278, 0.2491, 0.2976, 0.1738,
        0.3436, 0.3170, 0.2894, 0.2247, 0.3117, 0.3183, 0.2029, 0.2661, 0.1893,
        0.1915, 0.3010, 0.2944, 0.1835, 0.2033, 0.1870, 0.4224, 0.2538, 0.1865,
        0.1857, 0.2078, 0.2453, 0.2796, 0.2744, 0.2105, 0.1926, 0.2526, 0.2512,
        0.1781, 0.2534, 0.2072, 0.2977, 0.3508, 0.2030, 0.2107, 0.2639, 0.2113,
        0.2407, 0.2784, 0.2490, 0.2612, 0.2072, 0.2821, 0.1798, 0.3152, 0.2416,
        0.3017, 0.2331, 0.1939, 0.2194, 0.2872, 0.3713, 0.2201, 0.2070, 0.1939,
```

```
        0.1923, 0.1890, 0.2425, 0.2248, 0.2332, 0.4479, 0.2414, 0.2172, 0.2001,
        0.2797, 0.2225, 0.3804, 0.2393, 0.4375, 0.1920, 0.2549, 0.3148, 0.2508,
        0.1863, 0.2441, 0.2206, 0.3231, 0.3194, 0.4850, 0.1583, 0.3514, 0.1553,
        0.2346, 0.2182, 0.3194, 0.2702, 0.2727, 0.2792, 0.2450, 0.2427, 0.3802,
        0.1942, 0.3072, 0.1770, 0.4579, 0.3582, 0.1991, 0.2690, 0.2242, 0.4209,
        0.3136, 0.3175, 0.2270, 0.2332, 0.2192, 0.2105, 0.3151, 0.3112, 0.2677,
        0.2336, 0.2169, 0.3124, 0.2725, 0.2502, 0.3575, 0.2156, 0.2029, 0.2172,
        0.2569, 0.2674, 0.1748, 0.2235, 0.2437, 0.1891, 0.2241, 0.2717, 0.3845,
        0.2355, 0.1589, 0.2486, 0.2446, 0.3681, 0.3576, 0.2536, 0.5154, 0.3108,
        0.1819, 0.2497, 0.1984, 0.2301, 0.2575, 0.1513, 0.2752, 0.3419, 0.2243,
        0.2921, 0.2752, 0.4325, 0.3503, 0.1965, 0.1935, 0.2966, 0.1834, 0.2072,
        0.2628, 0.2744, 0.3078, 0.2389, 0.3189, 0.2200, 0.3851, 0.2340, 0.1568,
        0.2364, 0.3176, 0.1779, 0.2211, 0.1959, 0.2359, 0.3222, 0.2832, 0.2645,
        0.2168, 0.1656, 0.4528, 0.1617, 0.4045, 0.2693, 0.2749, 0.2466, 0.2440,
        0.2146, 0.1769, 0.2884, 0.2793, 0.2205, 0.1360, 0.2479, 0.2121, 0.2848,
        0.1637, 0.1758, 0.2797, 0.4029], device='cuda:0')), ('conv8_bn.num_batches_tracked', tensor
(8160, device='cuda:0')), ('conv9.weight', tensor([[[[-0.0401, -0.0194, -0.0011],
          [ 0.0097, -0.0193,  0.0081],
          [-0.0172, -0.0227,  0.0106]],

         [[ 0.0026,  0.0003,  0.0095],
          [-0.0113,  0.0149,  0.0049],
          [-0.0149,  0.0120,  0.0112]],

         [[-0.0102,  0.0189,  0.0208],
          [ 0.0076, -0.0213, -0.0096],
          [-0.0281, -0.0213, -0.0189]],

         ...,

         [[ 0.0154, -0.0037, -0.0047],
          [-0.0013,  0.0191,  0.0037],
          [-0.0142, -0.0172, -0.0171]],

         [[-0.0287, -0.0153, -0.0402],
          [-0.0142, -0.0154, -0.0282],
          [ 0.0217, -0.0006, -0.0058]],

         [[ 0.0016, -0.0071,  0.0138],
          [ 0.0217, -0.0158,  0.0015],
          [-0.0081, -0.0028,  0.0036]]],


        [[[-0.0012,  0.0249,  0.0297],
          [ 0.0047, -0.0085, -0.0010],
          [ 0.0052,  0.0067,  0.0202]],

         [[-0.0124,  0.0119, -0.0158],
          [-0.0120,  0.0083, -0.0125],
          [ 0.0169,  0.0049,  0.0141]],

         [[-0.0241,  0.0065,  0.0161],
          [ 0.0043, -0.0112,  0.0018],
          [ 0.0049,  0.0063, -0.0210]],

         ...,

         [[ 0.0320,  0.0158,  0.0151],
          [-0.0024,  0.0190, -0.0012],
          [ 0.0207,  0.0021,  0.0022]],

         [[ 0.0079,  0.0030, -0.0143],
          [-0.0214,  0.0091, -0.0075],
          [-0.0127, -0.0101, -0.0144]],

         [[-0.0013,  0.0111, -0.0129],
          [-0.0179, -0.0194,  0.0121],
          [ 0.0140,  0.0189,  0.0223]]],


        [[[ 0.0097,  0.0252, -0.0019],
          [ 0.0109, -0.0014,  0.0171],
          [ 0.0061, -0.0121, -0.0122]],

         [[ 0.0186, -0.0139, -0.0000],
          [ 0.0167, -0.0015,  0.0207],
          [-0.0058,  0.0140,  0.0248]],
```

```
 [[ 0.0356,  0.0157,  0.0082],
  [ 0.0179,  0.0175,  0.0118],
  [-0.0076, -0.0069,  0.0098]],

 ...,

 [[ 0.0267,  0.0024,  0.0216],
  [ 0.0213,  0.0343,  0.0308],
  [ 0.0138, -0.0057,  0.0027]],

 [[-0.0039, -0.0052, -0.0126],
  [-0.0359, -0.0170, -0.0075],
  [-0.0028, -0.0124, -0.0313]],

 [[ 0.0134,  0.0002,  0.0033],
  [ 0.0223, -0.0090,  0.0195],
  [-0.0019,  0.0214,  0.0153]]],


...,


[[[-0.0128, -0.0082, -0.0381],
  [-0.0372, -0.0076, -0.0376],
  [-0.0062, -0.0236, -0.0123]],

 [[-0.0194, -0.0242, -0.0032],
  [-0.0158, -0.0183, -0.0374],
  [-0.0057, -0.0131,  0.0080]],

 [[-0.0135,  0.0099, -0.0157],
  [-0.0173, -0.0052, -0.0138],
  [ 0.0002, -0.0313, -0.0176]],

 ...,

 [[ 0.0062,  0.0182,  0.0213],
  [ 0.0212,  0.0326,  0.0154],
  [ 0.0183,  0.0039,  0.0023]],

 [[ 0.0011, -0.0079, -0.0181],
  [-0.0248,  0.0062,  0.0001],
  [-0.0129, -0.0107,  0.0041]],

 [[ 0.0170,  0.0189,  0.0094],
  [ 0.0089,  0.0232,  0.0090],
  [-0.0034, -0.0084,  0.0129]]],


[[[ 0.0115, -0.0006,  0.0012],
  [ 0.0265,  0.0191,  0.0043],
  [ 0.0130,  0.0054,  0.0293]],

 [[ 0.0110, -0.0044, -0.0103],
  [-0.0148, -0.0093,  0.0105],
  [-0.0110, -0.0151,  0.0195]],

 [[ 0.0184,  0.0146,  0.0103],
  [ 0.0062,  0.0178, -0.0070],
  [ 0.0169, -0.0107, -0.0043]],

 ...,

 [[-0.0250,  0.0034, -0.0072],
  [-0.0097,  0.0119,  0.0121],
  [ 0.0003,  0.0056, -0.0099]],

 [[ 0.0387,  0.0051,  0.0300],
  [ 0.0154,  0.0040,  0.0176],
  [ 0.0008,  0.0031, -0.0111]],

 [[ 0.0313,  0.0109,  0.0220],
  [ 0.0218,  0.0058,  0.0234],
  [ 0.0001,  0.0416,  0.0178]]],


[[[ 0.0101, -0.0001, -0.0150],
```

```
         [[[-0.0191, -0.0091, -0.0159],
           [-0.0128,  0.0126, -0.0204],
           [ 0.0005,  0.0193, -0.0001]],

          [[ 0.0047, -0.0101,  0.0116],
           [-0.0153, -0.0071,  0.0164],
           [-0.0018, -0.0066, -0.0108]],

          [[-0.0234, -0.0241, -0.0007],
           [-0.0190, -0.0119,  0.0013],
           [ 0.0078, -0.0122, -0.0186]],

          ...,

          [[ 0.0188,  0.0216, -0.0032],
           [ 0.0251,  0.0061,  0.0206],
           [-0.0125,  0.0106, -0.0143]],

          [[-0.0118, -0.0157,  0.0010],
           [ 0.0266, -0.0072,  0.0200],
           [ 0.0254,  0.0188,  0.0349]],

          [[ 0.0070,  0.0021, -0.0023],
           [ 0.0094,  0.0079,  0.0278],
           [ 0.0335,  0.0038, -0.0033]]]], device='cuda:0')), ('conv9.bias', tensor([-0.0125,  0.015
0, -0.0092,  0.0062,  0.0186,  0.0138,  0.0153,  0.0149,
        -0.0107,  0.0170, -0.0070, -0.0021, -0.0139, -0.0018,  0.0198, -0.0163,
         0.0137, -0.0125, -0.0132, -0.0150,  0.0132, -0.0026, -0.0081,  0.0065,
         0.0155, -0.0184, -0.0047,  0.0173, -0.0088,  0.0191, -0.0096, -0.0052,
        -0.0094, -0.0076,  0.0182,  0.0156, -0.0020, -0.0102,  0.0033,  0.0045,
         0.0019, -0.0035, -0.0184, -0.0061,  0.0102, -0.0160,  0.0000, -0.0088,
        -0.0175,  0.0165, -0.0076,  0.0055,  0.0045,  0.0104,  0.0109,  0.0020,
         0.0102,  0.0035,  0.0049,  0.0182,  0.0013, -0.0173,  0.0022,  0.0205,
        -0.0056,  0.0037,  0.0143, -0.0101,  0.0127, -0.0081, -0.0120,  0.0140,
         0.0151,  0.0063,  0.0174,  0.0103,  0.0046,  0.0173,  0.0082, -0.0065,
        -0.0056,  0.0088,  0.0113,  0.0073,  0.0048,  0.0094, -0.0063,  0.0009,
         0.0083,  0.0189, -0.0010,  0.0023,  0.0173,  0.0168,  0.0180,  0.0015,
        -0.0115,  0.0150, -0.0013,  0.0042, -0.0074,  0.0079,  0.0165, -0.0038,
        -0.0061, -0.0041,  0.0175,  0.0094,  0.0126, -0.0122,  0.0163,  0.0071,
        -0.0007, -0.0017, -0.0168,  0.0072, -0.0207, -0.0137, -0.0060, -0.0135,
         0.0160,  0.0045,  0.0050, -0.0050, -0.0142,  0.0007, -0.0181,  0.0201],
       device='cuda:0')), ('conv9_bn.weight', tensor([0.9555, 0.8036, 0.6342, 1.0029, 0.0349, 0.196
7, 0.2373, 0.7224, 0.9667,
        0.4167, 0.6038, 0.2548, 0.1591, 0.3788, 0.8739, 0.3875, 0.6714, 0.5794,
        0.1169, 0.1191, 0.9435, 0.2906, 0.1307, 0.1851, 0.5292, 0.3881, 0.1194,
        0.2878, 0.0641, 0.7973, 0.7154, 0.8538, 0.4938, 0.3061, 0.5594, 0.8879,
        0.5988, 0.6477, 0.8888, 0.8864, 0.1811, 0.2092, 0.2502, 0.8261, 0.6549,
        0.3445, 0.6368, 0.3962, 0.2461, 0.7130, 0.3038, 0.5263, 0.9602, 0.1788,
        0.8796, 0.2116, 0.6428, 0.6522, 0.4647, 0.6414, 0.3103, 0.6630, 0.3744,
        0.7710, 0.6298, 0.1504, 0.2689, 0.6347, 0.5884, 0.5801, 0.1769, 0.2783,
        0.5778, 0.7068, 0.5478, 0.7073, 0.0884, 0.3276, 0.8710, 1.0468, 0.6319,
        0.2372, 0.9321, 0.2039, 0.0864, 0.8710, 0.9018, 0.6732, 0.1412, 0.9734,
        0.6231, 1.0060, 0.9666, 0.1108, 0.0419, 0.2593, 0.0002, 0.2119, 0.4209,
        0.0986, 0.6918, 0.0310, 0.8055, 0.5234, 0.4369, 0.2455, 0.1856, 0.1326,
        0.4730, 0.3047, 0.3640, 0.8552, 0.6728, 0.7603, 0.9446, 0.4601, 0.7376,
        0.8073, 0.1829, 0.5514, 0.5910, 0.3350, 0.4541, 0.6063, 0.1666, 0.1322,
        0.6027, 0.5793], device='cuda:0')), ('conv9_bn.bias', tensor([-0.0330, -0.0217, -0.0147, -
0.0072, -0.0100, -0.0019, -0.0135,  0.0007,
        -0.0207, -0.0532, -0.0157,  0.0160, -0.0010,  0.0157, -0.0227, -0.0716,
        -0.0481, -0.0299, -0.0190, -0.0230, -0.0411, -0.0044, -0.0004, -0.0336,
        -0.0299,  0.0161, -0.0190,  0.0067, -0.0061, -0.0289, -0.0299, -0.0086,
        -0.0107, -0.0334, -0.0154, -0.0254, -0.0321, -0.0197, -0.0289, -0.0539,
        -0.0030, -0.0230, -0.0201, -0.0348, -0.0341, -0.0245, -0.0547, -0.0327,
         0.0044, -0.0634, -0.0234, -0.0072, -0.0321, -0.0364, -0.0232, -0.0466,
        -0.0506, -0.0075, -0.0050,  0.0102, -0.0519, -0.0285, -0.0260, -0.0918,
        -0.0065, -0.0047, -0.0445, -0.0695, -0.0190, -0.0145, -0.0155,  0.0054,
        -0.0252, -0.0307, -0.0316, -0.0098,  0.0139, -0.0247, -0.0336, -0.0344,
        -0.0194, -0.0255, -0.0105, -0.0235,  0.0038, -0.0146, -0.0573, -0.0321,
         0.0057, -0.0061, -0.0358, -0.0277, -0.0380,  0.0124, -0.0134, -0.0171,
        -0.0013, -0.0208, -0.0375, -0.0032, -0.0356, -0.0110, -0.0354, -0.0407,
        -0.0345, -0.0313, -0.0627, -0.0095, -0.0316,  0.0029, -0.0320, -0.0463,
        -0.0633, -0.0150, -0.0353, -0.0380, -0.0539, -0.0108, -0.0063, -0.0261,
         0.0011, -0.0244, -0.0176, -0.0240,  0.0143,  0.0076, -0.0116, -0.0293],
       device='cuda:0')), ('conv9_bn.running_mean', tensor([-1.4488, -0.9198, -0.3364, -1.4885, -0
.1806,  0.7628,  0.3698, -0.0677,
        -0.6278, -1.2140, -0.4533,  0.3260, -0.3548,  0.3000, -1.2680, -1.2761,
        -1.1771, -0.7269, -0.7231, -0.4227, -1.1017,  0.7726,  0.1648, -0.3491,
```

```
         0.1652,  1.0855, -0.6164,  0.4493,  0.2435, -0.8959, -0.9240, -0.3270,
        -0.6001, -1.2640, -0.6702, -0.4411, -1.2835, -0.4226, -0.7347, -1.3891,
         0.4648, -0.6786, -1.2675, -0.6004, -1.1999, -0.3019, -0.9863,  0.3593,
        -0.6253, -1.4053, -0.5465, -0.6613, -1.2323, -0.5325, -0.8033, -1.6272,
        -0.2264, -0.8643,  0.5638, -0.4411, -0.8801, -1.0330, -1.3543, -1.4196,
         0.1522,  0.0509, -0.6559, -1.5132, -0.6381,  0.7517, -0.2386,  0.1857,
         0.2266, -1.3088, -0.6967, -0.4713,  0.5824, -0.6058, -0.7043, -1.1956,
        -0.5701, -0.4119, -0.7284, -0.4845,  1.5532, -1.3070, -1.4329, -0.1819,
         0.2533, -0.6663, -0.8096, -0.8734, -0.1629,  1.0471, -0.7049, -0.3940,
        -0.2009,  0.0820, -0.4741, -0.3431, -1.3215,  0.5855, -0.7092, -0.7302,
        -1.1733, -1.0559, -1.0396, -1.1679, -0.8339,  0.2914, -0.9168, -1.7544,
        -1.1213, -0.4270, -0.9166, -0.8978, -1.2313, -1.4646, -0.1293, -0.8109,
         0.0777, -0.7977, -0.8342, -0.9931,  0.3679, -0.7027,  1.1193, -0.9060],
       device='cuda:0')), ('conv9_bn.running_var', tensor([0.4056, 0.4762, 0.4347, 0.6082, 0.4911,
0.4422, 0.5272, 0.6404, 0.3950,
        0.4098, 0.4670, 0.4691, 0.4935, 0.7356, 0.4385, 0.4664, 0.4502, 0.2893,
        0.3233, 0.2608, 0.4067, 0.4913, 0.3471, 0.2962, 0.3366, 1.0227, 0.3260,
        0.4108, 0.5270, 0.4152, 0.4575, 0.9025, 0.2806, 0.4022, 0.3487, 0.4147,
        0.4070, 0.4515, 0.4187, 0.5419, 0.8264, 0.2962, 0.4435, 0.4672, 0.4567,
        0.3622, 0.3702, 0.5099, 0.4445, 0.4370, 0.3143, 0.4017, 0.4839, 0.3744,
        0.5743, 0.5213, 0.3362, 0.3607, 0.7559, 0.5496, 0.3487, 0.4628, 0.4460,
        0.4262, 0.4325, 0.4792, 0.4332, 0.5383, 0.3607, 0.5862, 0.2680, 0.5038,
        0.4258, 0.5698, 0.4475, 0.4227, 0.7879, 0.5015, 0.5533, 0.3881, 0.3829,
        0.3430, 0.9030, 0.3271, 0.7198, 0.4824, 0.4817, 0.3615, 0.4698, 0.3949,
        0.3541, 0.4614, 0.2986, 0.7659, 0.3836, 0.3340, 0.0274, 0.3387, 0.2958,
        0.3042, 0.5174, 0.3807, 0.5543, 0.2983, 0.4206, 0.4200, 0.4862, 0.5097,
        0.3359, 0.5183, 0.3370, 0.6362, 0.5448, 0.3888, 0.4872, 0.4145, 0.5067,
        0.4492, 0.3343, 0.4029, 0.3499, 0.3669, 0.5901, 0.4351, 0.6436, 0.3294,
        0.8052, 0.3305], device='cuda:0')), ('conv9_bn.num_batches_tracked', tensor(8160, device='c
uda:0')), ('conv10.weight', tensor([[[[ 6.2817e-03,  2.5143e-02,  8.7423e-03,  3.3386e-02,
2.4129e-02],
          [ 2.5889e-02,  1.4497e-02,  2.1209e-02,  1.7558e-02,  3.1274e-02],
          [-8.7551e-03,  1.5731e-02,  7.1584e-03,  3.3049e-02,  2.2491e-02],
          [ 7.9594e-03,  3.4122e-02,  1.2075e-02,  2.8851e-02,  3.7112e-02],
          [ 2.3186e-02,  1.8593e-02,  3.0815e-02,  3.0024e-02,  4.6403e-02]],

         [[ 1.0036e-04, -9.7199e-03, -2.1399e-03,  7.0495e-04,  2.2064e-03],
          [ 2.2012e-02, -7.7031e-04, -4.8839e-03,  5.4042e-03, -9.6653e-03],
          [-9.6374e-04,  2.4844e-02, -5.9766e-03,  1.1392e-02,  8.8397e-03],
          [ 2.9631e-02,  2.7097e-03,  4.4058e-03, -1.3415e-03, -3.9816e-03],
          [-3.2529e-03,  1.3657e-02,  1.2269e-03,  5.9423e-03,  1.9842e-02]],

         [[ 2.7150e-02,  8.3621e-03,  1.2276e-02, -1.0126e-02, -6.3498e-03],
          [ 6.0066e-03,  5.8464e-03,  5.0765e-03,  7.7181e-03,  1.4957e-02],
          [ 2.2404e-03, -2.6807e-03, -3.2920e-03,  1.6401e-02,  1.4059e-02],
          [ 1.2069e-02,  1.6975e-02, -1.6175e-02,  6.4811e-03,  7.7313e-03],
          [ 1.9768e-02,  8.1700e-03,  9.6380e-03,  1.7907e-02, -1.3491e-02]],

         ...,

         [[-7.4683e-03, -1.5679e-02, -4.1912e-03,  4.2822e-03,  4.8738e-03],
          [ 7.0736e-03,  1.5370e-02, -1.4390e-02,  2.5831e-03, -8.3306e-03],
          [-2.5004e-03,  2.6691e-03, -1.5682e-02,  2.3549e-03,  6.2510e-03],
          [ 7.4019e-03,  1.4122e-02,  1.7132e-03,  1.1815e-02,  1.4722e-02],
          [ 1.8804e-03, -1.1930e-02,  3.4842e-03, -1.1307e-03, -1.4298e-02]],

         [[ 1.6620e-02, -6.2950e-04, -1.3199e-02,  1.5334e-02,  6.7752e-03],
          [ 1.3706e-03,  4.1506e-03, -6.6545e-03, -1.2566e-02,  1.5795e-02],
          [ 1.1379e-02,  4.5135e-03, -1.5394e-02,  9.8528e-03, -1.3778e-02],
          [ 1.5918e-02,  3.7658e-03, -7.1476e-03,  9.6585e-04, -9.9840e-04],
          [ 3.3181e-04, -5.8062e-03,  1.0069e-02, -3.7352e-03,  9.9981e-03]],

         [[ 5.1911e-02,  1.5034e-02,  4.4266e-02,  3.4878e-02,  4.3233e-02],
          [ 1.9979e-02,  2.0631e-02,  1.4610e-02,  4.1412e-02,  3.4059e-02],
          [ 1.6502e-02,  1.1631e-02,  3.7704e-03,  3.0818e-02,  2.1846e-02],
          [ 3.6874e-02,  5.2203e-02,  4.4949e-02,  3.7796e-02,  3.7080e-02],
          [ 5.2271e-02,  4.1201e-02,  5.8012e-02,  3.1806e-02,  7.0491e-02]]],


        [[[ 9.0660e-04, -5.5239e-03, -5.3901e-03,  1.2152e-02, -1.2698e-03],
          [ 6.8816e-03,  8.4212e-03, -7.6905e-03, -7.3564e-03, -5.3874e-03],
          [ 2.0416e-02,  2.1263e-03,  1.0248e-03, -7.3100e-03,  2.0624e-02],
          [ 1.3736e-05, -1.4891e-02, -6.5024e-03, -4.1613e-03,  5.4207e-03],
          [ 1.4280e-03,  1.6876e-03,  5.2963e-03,  1.5912e-02,  7.3181e-03]],

         [[-1.1306e-02,  6.8395e-03,  8.1663e-03,  7.4754e-03, -1.3935e-02],
          [ 7.9261e-03, -1.2083e-02, -6.9919e-03, -4.1007e-03, -1.5086e-02],
```

```
  [-5.0021e-03, -4.6177e-03, -1.3157e-02,  7.6331e-03, -2.7226e-03],
  [-1.5563e-03, -3.5798e-03, -6.4431e-03, -4.9914e-03, -1.4618e-02],
  [ 1.5442e-03, -1.7719e-02, -1.1445e-02, -6.8486e-03,  1.0528e-02]],

 [[-2.1388e-02, -1.6696e-02, -1.8334e-02, -2.1127e-02, -6.7749e-03],
  [-2.0000e-02,  8.2940e-03,  1.7176e-02, -1.3214e-03,  1.3368e-02],
  [ 1.0055e-02, -1.0544e-02, -1.9012e-02, -1.5313e-02, -6.5585e-03],
  [ 5.8509e-03, -1.9243e-02,  1.2421e-02, -1.2133e-02,  7.9977e-03],
  [ 8.6964e-03,  3.5466e-03,  5.0442e-03,  7.2810e-03, -2.1878e-02]],

 ...,

 [[ 2.1700e-02, -8.5230e-03,  9.7824e-03,  1.0841e-02, -7.2078e-03],
  [ 1.2201e-02, -5.4953e-03,  9.2548e-03,  4.3451e-03,  1.2369e-02],
  [-1.8761e-03,  1.3105e-02,  1.4603e-02, -1.7434e-02,  7.4526e-03],
  [-9.9920e-03,  1.2499e-02, -1.8197e-02, -2.2013e-02, -6.3962e-03],
  [ 5.5968e-03, -1.1989e-02, -2.3007e-02, -5.2499e-03,  3.4135e-03]],

 [[-1.1615e-02, -1.7479e-03, -4.1460e-03, -4.7127e-03, -4.6278e-03],
  [ 2.5352e-03, -8.3968e-03,  1.0542e-02, -1.1208e-02, -2.3326e-02],
  [-1.2367e-02, -1.5806e-02, -6.3132e-03, -7.6110e-03,  1.7243e-02],
  [ 4.7616e-03,  8.0359e-03, -1.8228e-02,  1.6028e-02,  1.5921e-02],
  [-8.7055e-03,  7.9780e-03,  3.6597e-03,  1.0811e-02, -1.0552e-03]],

 [[-1.2990e-02,  1.3498e-02, -5.7866e-03, -1.6272e-02, -2.2664e-02],
  [ 6.3222e-04,  2.0750e-04,  6.6520e-03, -1.1556e-02,  8.9396e-03],
  [-2.0644e-02, -8.2705e-03, -1.1117e-02, -3.9411e-03, -1.9174e-02],
  [ 1.9128e-02,  1.6238e-02,  5.5350e-03, -1.2630e-02,  8.8629e-04],
  [-1.5338e-02, -1.4060e-02, -7.5573e-04,  7.8372e-03, -8.4056e-03]]],


[[[-6.4391e-03, -1.2288e-02,  1.3873e-02,  2.3890e-02, -1.0849e-02],
  [ 9.1011e-04,  1.1254e-02,  1.5716e-02,  2.4479e-03, -1.0851e-03],
  [-1.7425e-02, -2.1292e-02,  1.7722e-03,  1.8640e-02, -6.1471e-04],
  [-1.1880e-03, -3.7107e-03, -1.1255e-03,  1.1904e-02, -7.6102e-03],
  [ 1.2491e-02,  3.6620e-03,  7.0221e-03,  1.1694e-02,  1.0532e-02]],

 [[ 5.2100e-04, -3.2642e-03, -1.8836e-03,  1.3809e-02,  1.1093e-02],
  [-3.4928e-03, -6.5890e-04, -1.6311e-02, -8.1511e-03,  2.4231e-02],
  [ 4.6134e-03, -6.4812e-03,  7.4410e-04, -1.0016e-02,  2.3921e-02],
  [-4.5907e-03, -6.9681e-03,  1.0715e-02,  1.2151e-03,  2.4222e-02],
  [ 3.2502e-03, -1.8632e-02, -1.3671e-03, -4.9245e-03, -1.1169e-02]],

 [[-1.6995e-02, -1.7350e-02, -2.4596e-02,  1.9573e-03, -1.2844e-02],
  [-2.6142e-02,  3.3801e-03,  4.9879e-03,  4.9124e-03, -9.3385e-03],
  [ 4.4237e-03, -1.1478e-02,  7.9675e-03, -2.4284e-02,  8.4047e-03],
  [ 2.6575e-03,  3.5621e-03, -9.0434e-03, -2.0448e-02,  2.1632e-02],
  [-3.5245e-03, -1.3018e-02,  5.6025e-03, -7.1030e-03, -2.0186e-02]],

 ...,

 [[ 1.1383e-02,  9.8666e-03, -4.3057e-04,  1.0291e-02, -1.4961e-02],
  [-8.6517e-04,  1.1636e-03, -1.6600e-02, -1.8523e-03,  2.0067e-03],
  [-9.3379e-03,  1.0600e-02, -4.9824e-03, -1.7779e-03,  1.0564e-02],
  [ 2.2081e-02, -1.3461e-03,  4.9044e-03,  2.0448e-02, -6.8512e-03],
  [ 8.2259e-04,  4.6622e-03,  2.1639e-03,  9.8008e-03, -6.7556e-03]],

 [[ 1.5116e-02,  1.0629e-02,  1.7687e-02,  9.2752e-03,  1.7564e-02],
  [ 8.8541e-03,  6.9460e-03,  1.8375e-02, -2.8814e-03, -4.2814e-03],
  [-3.4342e-03, -1.5777e-02, -1.3742e-02,  1.7195e-02,  1.2290e-02],
  [ 3.0843e-02, -7.9244e-03, -1.1607e-02,  1.4275e-02, -2.6463e-03],
  [ 1.4703e-03, -7.5077e-03,  8.9896e-03, -2.7904e-03,  6.0768e-03]],

 [[ 8.1764e-03,  1.4338e-02,  4.8840e-03, -4.7651e-04, -9.4566e-03],
  [-8.9902e-03, -1.0034e-02, -1.5520e-03,  1.2285e-02, -3.6393e-03],
  [-1.6264e-02, -1.8572e-02,  9.0786e-03, -2.5912e-04, -2.7739e-02],
  [ 1.7164e-02, -1.9889e-02, -7.1579e-03, -9.8151e-03, -1.1928e-02],
  [ 1.4842e-02, -5.0136e-03, -2.1927e-02, -2.2982e-02,  5.9691e-03]]],


...,


[[[-4.0407e-03, -1.7055e-02,  1.2488e-02, -1.0893e-02, -3.5260e-03],
  [-8.2272e-03,  2.0869e-02, -1.8419e-03,  1.0806e-02,  2.3531e-02],
  [ 1.6485e-02, -8.4018e-03, -2.0627e-02,  1.1980e-02,  4.6660e-03],
  [-1.4722e-02, -3.1579e-02, -3.5618e-02, -8.5948e-03, -8.1120e-03],
```

```
  [-1.8397e-03,  1.0367e-02,  1.1438e-02, -2.2401e-03,  1.1933e-02]],

 [[ 2.1221e-02,  1.2398e-03,  8.0700e-03,  2.3848e-02,  1.2156e-02],
  [ 9.7801e-03, -9.6468e-03,  1.6699e-02,  9.7144e-03,  6.6385e-03],
  [ 4.1920e-03, -3.6109e-03, -6.1492e-03,  5.5989e-03, -5.8736e-03],
  [ 1.2736e-02,  7.2818e-03,  5.9356e-03,  2.0813e-02,  6.0958e-03],
  [ 2.6413e-02,  6.3364e-03, -7.7088e-03, -5.6029e-03,  2.2887e-03]],

 [[ 9.6161e-03,  1.4236e-03,  1.7008e-02,  6.5027e-03,  7.1462e-03],
  [ 1.9916e-02,  1.0014e-02,  2.5384e-02,  2.4355e-02,  2.3933e-02],
  [ 1.7077e-02,  5.1447e-03,  1.9581e-02,  8.9286e-03,  6.3330e-03],
  [ 2.4073e-02,  3.2320e-03, -4.2897e-03, -9.0105e-04,  8.8822e-03],
  [ 3.3874e-02,  1.3968e-02,  2.6710e-02,  2.5727e-02, -2.4015e-04]],

 ...,

 [[ 1.5144e-02,  1.8668e-03, -1.9271e-03,  1.8720e-02,  9.1018e-03],
  [ 1.5717e-04, -8.2171e-03,  1.5232e-02, -6.9244e-03,  1.4357e-02],
  [-1.1292e-02,  1.2073e-02,  1.5913e-02, -6.5381e-03, -1.9603e-02],
  [ 1.1821e-02,  3.3182e-03, -2.9378e-02, -5.4761e-03, -5.7880e-03],
  [ 1.1557e-02, -2.0137e-02, -1.6162e-02,  9.1129e-03,  4.3666e-03]],

 [[ 1.7277e-03, -3.7495e-03, -2.4699e-03, -1.2913e-02, -1.6962e-02],
  [-1.3968e-02, -3.0976e-04, -6.5925e-03, -4.5576e-03, -4.0008e-03],
  [-3.8653e-03, -2.2035e-03, -2.0639e-02,  5.1328e-03,  3.6771e-04],
  [ 1.9690e-02,  4.2472e-03, -6.9020e-03, -8.8428e-03, -1.8126e-02],
  [-1.0002e-02, -3.2088e-02, -2.2446e-03, -4.0045e-03, -3.0222e-02]],

 [[-7.1385e-03,  3.1681e-03, -1.2550e-02, -1.1957e-02, -1.2196e-02],
  [-3.2258e-02, -1.4882e-02,  2.2261e-04, -3.0388e-03, -2.0408e-02],
  [ 5.0202e-03, -1.7944e-02, -1.0502e-02, -1.0178e-04, -1.3811e-02],
  [-2.6895e-02, -1.4065e-02, -1.3834e-02, -1.7540e-02, -2.5900e-02],
  [-2.2223e-02, -2.3315e-03, -1.3593e-02, -2.6698e-02, -1.9708e-02]]],


[[[-1.1781e-02,  4.1746e-03,  1.4163e-02,  9.4522e-03, -7.5726e-03],
  [-1.0093e-02,  1.5063e-02,  2.0926e-02, -3.8348e-03,  2.5349e-02],
  [-1.5846e-02, -1.1250e-02,  1.0588e-02, -9.0862e-03,  1.5927e-02],
  [ 5.0993e-06,  4.5551e-04, -1.5244e-02, -4.6126e-03,  1.9844e-03],
  [-2.4223e-03, -7.0788e-03,  1.8101e-03, -1.5193e-02,  9.8894e-03]],

 [[ 1.6617e-02, -1.6427e-02,  3.5825e-03, -1.5904e-02, -1.0060e-03],
  [ 1.8803e-02,  1.1881e-02,  6.6634e-03, -4.7187e-03,  1.3022e-02],
  [-1.4052e-02, -2.1464e-03,  1.3423e-02,  9.5046e-03, -4.7602e-03],
  [ 1.8051e-02, -7.3048e-03,  8.8566e-03, -1.4041e-02, -1.0648e-02],
  [ 1.3987e-02,  1.8426e-02, -4.8157e-03,  1.9206e-02,  2.2493e-02]],

 [[ 9.1035e-03, -1.1392e-02, -1.2621e-02,  9.4668e-03, -1.0205e-02],
  [ 3.0639e-02,  1.8167e-02,  1.4796e-02, -7.5542e-03,  1.4638e-02],
  [ 9.4661e-03,  4.4124e-03, -2.8535e-03, -1.2286e-02,  1.1636e-02],
  [-2.0588e-02,  1.0104e-02, -1.2560e-02, -1.2694e-02, -1.1556e-02],
  [-1.8768e-02, -2.1475e-02, -2.0362e-02,  3.1343e-03,  6.0909e-03]],

 ...,

 [[ 1.4073e-02, -6.0821e-03,  1.7512e-02, -1.3615e-02, -5.4036e-03],
  [-3.8498e-04, -6.6526e-03, -1.5266e-02, -2.1393e-02,  2.5966e-03],
  [-1.2326e-02,  4.2485e-03, -2.6620e-02, -6.6252e-04, -3.0855e-03],
  [-1.0657e-02, -1.6765e-02, -2.0545e-02, -9.5367e-03, -2.0177e-02],
  [-1.9821e-02,  1.6636e-02, -2.7639e-02, -4.5542e-03,  4.4798e-03]],

 [[-2.9205e-02, -2.0951e-02, -3.5421e-02, -2.9085e-02, -1.3787e-02],
  [-3.5595e-02, -4.8519e-03, -5.5804e-03, -1.6756e-02, -2.3922e-02],
  [-5.2737e-03, -1.3162e-02, -1.6218e-02, -2.3166e-03, -1.6009e-02],
  [ 1.5160e-03, -1.9140e-02, -8.5934e-04, -9.8416e-03, -1.5094e-02],
  [ 8.2726e-03, -6.4025e-03,  1.0129e-02,  1.2699e-03,  1.7972e-03]],

 [[ 8.1005e-03,  9.8343e-03,  1.8122e-03,  6.0979e-03, -1.2084e-02],
  [ 3.3947e-03,  1.9967e-02,  1.4185e-02,  6.9954e-03,  7.3445e-03],
  [ 6.8567e-03, -3.6700e-03, -9.2069e-03, -2.2691e-02, -1.7507e-02],
  [-4.8226e-03,  2.3077e-03,  2.5738e-03, -2.0911e-02,  1.6658e-03],
  [ 5.8589e-03,  1.7845e-02, -3.8995e-03,  1.5010e-02, -5.4048e-04]]],


[[[ 1.2495e-02, -1.4015e-02, -4.1450e-03,  1.6495e-03, -5.3045e-03],
  [ 1.6592e-02, -1.9763e-02, -1.9743e-02, -7.1962e-04, -1.0993e-02],
  [-1.3166e-02, -1.8426e-02, -1.3412e-02, -8.9752e-03, -1.3867e-02],
```

```
          [-6.2768e-03, -7.3924e-03,  2.6375e-03, -9.0399e-04, -4.3208e-03],
          [ 1.3636e-02, -7.3613e-04, -1.0328e-02,  5.0506e-03, -1.2893e-02]],

         [[-2.2387e-02, -1.9720e-03,  1.1089e-02, -3.5800e-03,  1.2648e-02],
          [-8.6791e-03, -2.7445e-04,  1.8458e-02,  1.2399e-02,  2.6576e-02],
          [-1.8987e-02, -8.8241e-04,  7.8329e-03, -1.0461e-02,  1.0262e-02],
          [ 1.1772e-03, -2.8573e-03,  1.4854e-02, -8.7120e-03, -2.1889e-03],
          [-5.3367e-03, -1.9026e-02,  8.9665e-03, -1.5674e-02, -4.0684e-03]],

         [[ 4.7416e-03, -1.6987e-02, -1.6506e-02, -1.0267e-03, -4.9018e-03],
          [-1.8722e-02, -3.0369e-02, -3.0841e-03,  3.4460e-03, -1.3079e-02],
          [-2.4567e-02, -1.7048e-03, -2.3560e-02, -2.5161e-02, -1.1067e-02],
          [-2.6448e-02, -3.1467e-02, -2.9685e-03, -2.2518e-02, -9.4634e-03],
          [-1.8959e-02, -2.3437e-02, -1.1180e-02, -2.6427e-02, -2.2744e-02]],

         ...,

         [[-1.7248e-02, -1.9013e-02, -1.3405e-02, -1.9886e-02,  4.4358e-03],
          [-2.6379e-02, -2.4503e-02,  8.9724e-03,  1.4088e-03,  1.3317e-02],
          [-2.2889e-02,  2.0173e-03,  4.9050e-03,  1.7160e-02,  1.0769e-02],
          [ 8.8967e-03, -6.3310e-03, -7.6759e-03, -2.1770e-02,  4.8739e-03],
          [-2.0693e-02, -2.3299e-02,  2.5247e-03, -8.6531e-03,  1.3161e-02]],

         [[ 1.5187e-02,  3.1471e-02,  3.4822e-02,  1.1718e-02,  1.4803e-02],
          [ 9.6417e-04, -1.1098e-03,  5.9184e-03,  1.7997e-02,  2.2002e-02],
          [-1.0870e-02, -4.5265e-03, -1.0564e-02, -2.2765e-03,  1.2922e-02],
          [-8.1979e-03, -5.1426e-03,  1.6086e-02, -1.0861e-02, -6.3098e-03],
          [ 1.7080e-02,  1.7287e-02,  1.4841e-02,  1.8287e-02,  1.7492e-02]],

         [[ 2.3493e-03, -4.0591e-03, -2.7293e-03,  2.0478e-02,  8.1347e-03],
          [ 5.2078e-03,  1.4693e-02,  1.7772e-02, -4.0212e-03,  1.1600e-02],
          [ 6.1098e-03,  7.1264e-03, -2.3186e-02, -2.8058e-02,  2.6067e-03],
          [-2.0817e-02, -2.1825e-03, -2.6617e-02, -2.7943e-02, -2.9732e-03],
          [-1.8214e-02, -1.4334e-02,  4.1074e-03,  5.0859e-03, -2.5356e-03]]]],
       device='cuda:0')), ('conv10.bias', tensor([-0.0026,  0.0112,  0.0158,  0.0080,  0.0046, -0.0
015,  0.0064,  0.0010,
         0.0023, -0.0170, -0.0030, -0.0053, -0.0053, -0.0076, -0.0041,  0.0116,
        -0.0080, -0.0069,  0.0117, -0.0007,  0.0004, -0.0075, -0.0020, -0.0079,
         0.0079, -0.0067,  0.0023,  0.0072, -0.0081,  0.0145, -0.0073, -0.0067,
         0.0082, -0.0151,  0.0112,  0.0057,  0.0103,  0.0058, -0.0089, -0.0069,
        -0.0124, -0.0169,  0.0058,  0.0124, -0.0042,  0.0095, -0.0063, -0.0111,
        -0.0095, -0.0023,  0.0099, -0.0076, -0.0009, -0.0094,  0.0150, -0.0088,
        -0.0130,  0.0013,  0.0048, -0.0009, -0.0167, -0.0027,  0.0038,  0.0046,
         0.0109, -0.0160, -0.0074, -0.0041, -0.0141, -0.0003,  0.0134,  0.0149,
         0.0147, -0.0106,  0.0004, -0.0010, -0.0105,  0.0058,  0.0042, -0.0033,
        -0.0120, -0.0014, -0.0164,  0.0135, -0.0097, -0.0087, -0.0022, -0.0079,
         0.0164,  0.0136, -0.0096,  0.0053,  0.0024, -0.0003,  0.0143, -0.0017,
         0.0021, -0.0073, -0.0025,  0.0165, -0.0053,  0.0134, -0.0137,  0.0126,
         0.0135,  0.0120,  0.0051,  0.0025, -0.0160,  0.0016, -0.0067,  0.0088,
         0.0089,  0.0048, -0.0117, -0.0081,  0.0088,  0.0145, -0.0158,  0.0059,
        -0.0106,  0.0038,  0.0165, -0.0000, -0.0169,  0.0124,  0.0157,  0.0038,
        -0.0081,  0.0087, -0.0111, -0.0107,  0.0089, -0.0044,  0.0136, -0.0071,
        -0.0027,  0.0046, -0.0107, -0.0176, -0.0092,  0.0175, -0.0129, -0.0021,
        -0.0023, -0.0143, -0.0043, -0.0161,  0.0060,  0.0125,  0.0096, -0.0057,
         0.0138,  0.0017, -0.0147,  0.0081,  0.0021,  0.0068,  0.0127, -0.0117,
        -0.0103,  0.0050, -0.0173, -0.0173, -0.0020,  0.0176,  0.0163, -0.0156,
         0.0011,  0.0080,  0.0073, -0.0010,  0.0144,  0.0021, -0.0102,  0.0115,
         0.0045, -0.0002, -0.0176, -0.0065, -0.0085, -0.0154, -0.0168,  0.0088,
        -0.0125,  0.0171,  0.0173, -0.0144,  0.0022, -0.0042,  0.0068, -0.0115,
         0.0069, -0.0028, -0.0166,  0.0104,  0.0114,  0.0130,  0.0050, -0.0172,
         0.0135, -0.0075,  0.0168, -0.0037, -0.0120, -0.0090, -0.0133, -0.0007,
         0.0173,  0.0016, -0.0001, -0.0024,  0.0156, -0.0173, -0.0077, -0.0021,
        -0.0095, -0.0156,  0.0105, -0.0029,  0.0115, -0.0045, -0.0119,  0.0173,
        -0.0114,  0.0100, -0.0131, -0.0116,  0.0046, -0.0133, -0.0125,  0.0113,
         0.0112, -0.0025, -0.0088,  0.0129,  0.0059, -0.0136, -0.0140, -0.0114,
        -0.0008,  0.0045,  0.0056,  0.0092, -0.0153, -0.0091,  0.0152, -0.0106,
        -0.0063,  0.0091,  0.0166, -0.0149, -0.0019,  0.0010, -0.0016, -0.0043],
       device='cuda:0')), ('conv10_bn.weight', tensor([0.9574, 0.8719, 0.3094, 0.7103, 0.8708, 0.59
07, 0.6962, 0.7781, 0.1493,
        0.4711, 0.8632, 0.6209, 0.1200, 0.8243, 0.9585, 0.3394, 0.6713, 0.6208,
        0.0222, 0.3748, 0.4149, 0.6597, 0.3182, 0.4524, 0.2472, 0.2529, 0.8372,
        0.7255, 0.9873, 0.8260, 0.3953, 0.1170, 0.3200, 0.5600, 0.9800, 0.7089,
        0.5446, 0.6121, 0.7686, 0.9634, 0.8814, 0.1783, 0.4021, 0.3195, 0.3848,
        0.2476, 0.8067, 0.5497, 0.2707, 0.8030, 0.9018, 0.1819, 0.7895, 0.6092,
        0.3522, 0.3897, 0.3359, 0.4925, 0.1721, 0.4661, 0.9531, 0.4119, 0.7910,
        0.4223, 0.9804, 0.2906, 0.5077, 0.6424, 0.1324, 0.1814, 0.2428, 0.3561,
        0.0506, 0.4073, 0.4773, 0.6870, 0.4744, 0.2148, 0.0539, 0.4735, 0.8406,
```

```
        0.4248, 0.6628, 0.4446, 1.0221, 0.9567, 0.6187, 0.7335, 0.3946, 1.0390,
        0.4776, 0.5432, 0.2249, 1.0221, 0.1935, 0.2909, 0.2147, 0.0746, 0.9491,
        0.3848, 0.0318, 0.8840, 0.3697, 0.8595, 0.8487, 0.1669, 0.7233, 0.2523,
        0.3459, 0.3933, 0.4453, 0.6323, 0.0321, 0.4803, 0.4205, 0.5458, 0.7055,
        0.8605, 0.1558, 0.5016, 0.6649, 0.5341, 0.0747, 0.8564, 1.0188, 0.2629,
        0.8731, 0.2867, 0.4853, 0.9289, 0.0506, 0.3897, 0.2679, 0.3383, 0.9186,
        0.3008, 0.0473, 0.3732, 1.0197, 0.6788, 0.4746, 0.6426, 0.4678, 0.2702,
        0.0378, 0.5563, 0.9461, 0.6690, 0.4905, 0.7035, 0.1770, 0.3713, 0.1543,
        0.4482, 0.3511, 0.4065, 0.1740, 0.4262, 0.2528, 0.4396, 0.3111, 0.9822,
        0.5936, 0.8067, 0.5692, 0.4263, 0.9733, 0.6934, 0.5364, 0.2938, 0.7474,
        0.5016, 0.8436, 0.5660, 0.5814, 0.0490, 0.2389, 0.1536, 0.2901, 0.6612,
        0.4956, 0.1756, 0.3754, 0.1256, 0.9962, 0.1471, 0.5431, 0.5135, 0.9680,
        0.8064, 0.7039, 0.5978, 0.6716, 0.5332, 0.4770, 0.1516, 0.4874, 0.7137,
        0.5287, 0.3056, 0.2949, 0.0528, 0.9863, 0.9929, 0.7107, 0.1523, 0.0512,
        0.2712, 1.0012, 0.1256, 0.8235, 0.3838, 0.9719, 0.0439, 0.8478, 0.5519,
        0.0794, 0.1829, 0.6937, 0.7900, 0.3939, 0.8686, 0.8622, 0.9774, 0.4437,
        0.0777, 0.4172, 0.5816, 1.0111, 0.5333, 0.5666, 0.4171, 1.0165, 0.1031,
        0.4004, 0.7593, 0.4795, 0.7619, 0.8269, 0.9809, 0.3164, 0.1144, 0.5569,
        0.0634, 0.5086, 0.6928, 1.0104, 0.7181, 0.4280, 0.0710, 0.5843, 0.6272,
        0.5930, 1.0062, 0.8267, 0.8267], device='cuda:0')), ('conv10_bn.bias', tensor([ 0.0036,  0.
0025,  0.0091,  0.0056,  0.0151, -0.0066,  0.0034,  0.0065,
         0.0064,  0.0083,  0.0086,  0.0119,  0.0091,  0.0038, -0.0023, -0.0011,
         0.0032,  0.0166,  0.0011,  0.0093,  0.0073,  0.0053,  0.0138,  0.0111,
         0.0079,  0.0031,  0.0045,  0.0053, -0.0005,  0.0074,  0.0035,  0.0031,
        -0.0004,  0.0049,  0.0008, -0.0015,  0.0063,  0.0061,  0.0042,  0.0058,
         0.0059,  0.0053,  0.0077,  0.0002,  0.0067, -0.0006,  0.0015,  0.0138,
         0.0035, -0.0073,  0.0105, -0.0013,  0.0077,  0.0067,  0.0022,  0.0013,
         0.0112,  0.0075,  0.0060,  0.0100,  0.0033,  0.0065,  0.0096,  0.0062,
         0.0048,  0.0073,  0.0039,  0.0077,  0.0084,  0.0035,  0.0056, -0.0012,
         0.0084,  0.0060,  0.0110,  0.0023, -0.0022,  0.0026,  0.0008,  0.0115,
         0.0050,  0.0133, -0.0078,  0.0077,  0.0091,  0.0115,  0.0110,  0.0096,
         0.0017,  0.0057,  0.0055,  0.0009,  0.0053, -0.0032, -0.0023,  0.0033,
         0.0015,  0.0015,  0.0040, -0.0055,  0.0060, -0.0053,  0.0056,  0.0059,
         0.0067, -0.0039,  0.0078,  0.0077,  0.0071,  0.0107,  0.0110,  0.0127,
         0.0035,  0.0067,  0.0084,  0.0051,  0.0046,  0.0100,  0.0093,  0.0069,
        -0.0062,  0.0033,  0.0089,  0.0028,  0.0051, -0.0041,  0.0057, -0.0066,
        -0.0011,  0.0067,  0.0077, -0.0029, -0.0024, -0.0056,  0.0032,  0.0109,
         0.0066,  0.0066,  0.0026, -0.0001,  0.0049,  0.0007,  0.0060, -0.0042,
         0.0064,  0.0027,  0.0038, -0.0008,  0.0095,  0.0074,  0.0068,  0.0127,
         0.0084, -0.0027,  0.0057,  0.0037, -0.0021,  0.0087,  0.0045,  0.0045,
         0.0064,  0.0151,  0.0089, -0.0070,  0.0129,  0.0116,  0.0003,  0.0118,
         0.0045,  0.0078,  0.0003,  0.0065,  0.0028,  0.0091,  0.0000,  0.0111,
         0.0120,  0.0061,  0.0053,  0.0056,  0.0009,  0.0010,  0.0075,  0.0045,
         0.0072,  0.0094,  0.0051,  0.0067,  0.0034,  0.0019,  0.0033,  0.0087,
         0.0074,  0.0051,  0.0107,  0.0041,  0.0022,  0.0064, -0.0039,  0.0019,
         0.0012,  0.0066,  0.0069,  0.0007,  0.0002,  0.0065,  0.0053,  0.0025,
         0.0051,  0.0060,  0.0078,  0.0106,  0.0051,  0.0046,  0.0062,  0.0033,
         0.0073,  0.0074,  0.0082,  0.0049,  0.0095,  0.0057,  0.0042, -0.0005,
         0.0099, -0.0065,  0.0077, -0.0012,  0.0085, -0.0038, -0.0058,  0.0060,
        -0.0042,  0.0082,  0.0088,  0.0038,  0.0074, -0.0005,  0.0058,  0.0088,
         0.0174,  0.0102,  0.0066,  0.0066,  0.0056,  0.0041, -0.0071, -0.0028,
         0.0049,  0.0084,  0.0009, -0.0019,  0.0093, -0.0015,  0.0053,  0.0121],
       device='cuda:0')), ('conv10_bn.running_mean', tensor([ 3.4322, -0.8694,  2.5208,  1.2851, -1
.8463,  2.2079,  1.3602, -2.4892,
        -0.1411,  0.7107, -0.9358,  0.1974,  4.6023, -0.0319,  2.6550,  0.8385,
        -0.8820, -2.3434, -2.8002,  1.0875,  0.1949,  2.3251,  0.4589,  1.0728,
         1.5150, -1.3917, -1.4775,  0.9675,  1.0826,  0.4977,  0.2614,  0.4543,
         0.1593,  2.4504, -0.8753,  2.3893,  2.6153,  3.1015,  0.8897, -0.8526,
         1.8590,  1.8974, -1.7497,  3.1004,  0.6619,  1.7284,  0.0554, -2.1859,
         2.2896,  1.9430,  0.7314,  3.2056,  1.1122,  1.9716,  2.3952, -1.4129,
         3.3785,  0.2931,  1.2752,  1.3913, -2.9420,  1.8616,  1.8105,  1.3099,
        -0.3559, -1.1717, -0.6288,  1.7929,  1.5785, -0.6149,  3.2558,  1.6201,
         2.6523,  0.2927,  0.6006,  2.0009,  1.0797,  0.6670,  0.9359,  3.4739,
         0.6134,  1.2079,  1.4402, -2.0682, -0.5147, -0.4147, -2.2960,  0.4261,
         1.2840,  0.0017,  0.1030,  1.7542, -0.0850,  2.7894, -1.2011,  0.0821,
         2.5657, -2.3723,  2.1242,  2.3043, -4.8459, -0.2784,  0.1935,  0.8453,
         2.0271,  2.2936, -3.0891, -1.4100,  0.5058, -1.6206, -1.9826, -2.0027,
        -3.9635, -0.6922,  2.0322, -0.6969,  2.4267, -0.3047, -0.1974,  1.6449,
         2.4011,  0.7743,  2.1788, -0.6351,  1.3837,  1.8342, -1.0124,  1.2034,
         0.4926,  0.4029,  1.8296,  2.6585,  0.3078,  1.6934,  0.7658, -1.7725,
         0.0328,  1.1336,  0.6725,  0.7488, -0.4208,  2.2988,  2.1251,  2.0033,
        -2.6019,  1.4190,  2.8916,  0.8437, -1.4039,  0.9395,  1.9834, -2.1811,
         2.5302,  0.3850,  0.7542, -0.9638,  2.2244,  0.2410,  2.2819,  2.0724,
         0.9122, -0.4040, -2.2876,  2.3891,  0.3559, -0.4704,  1.6183,  2.8387,
         2.1983, -2.1311,  2.6400,  2.5387,  1.6406,  0.2450, -0.1057, -2.1470,
        -0.1121,  1.5487,  2.2483, -0.5352,  1.5947,  0.5930, -1.0905,  1.6834,
        -0.1186, -1.0189,  0.5662,  2.5166,  2.0842,  2.0928,  0.6249, -1.6960,
```

```
         0.3875,  1.1282, -1.4510,  0.4012,  0.6299, -2.9096,  1.3106,  0.8814,
         1.9797,  1.0785, -1.7682,  0.3918,  2.9527,  1.9150, -2.1760,  1.7450,
         1.6179,  2.4539,  2.5868, -1.4049,  0.3867, -0.5036, -1.8864,  2.7301,
        -2.5968,  4.3148, -1.2206,  3.6887,  0.4248, -1.3366, -0.8190,  2.1200,
        -2.2672,  2.0990,  0.2541,  1.8991,  1.3408,  1.7693,  1.7496, -0.5887,
         2.8004,  0.1366,  1.5990,  3.0427, -0.1943, -1.3743, -2.9037,  2.1872,
        -1.6152,  2.3233,  0.3255,  1.4665,  2.0599,  2.6696,  1.3689,  0.3300,
        -0.9188,  1.8829,  2.0241,  2.7289,  3.8754,  2.3483, -1.2952, -0.0457],
       device='cuda:0')), ('conv10_bn.running_var', tensor([1.5743, 0.9858, 1.1790, 0.9226, 1.0977,
1.4519, 1.0857, 0.9442, 0.9591,
        1.0015, 1.1653, 0.9930, 1.7820, 1.1414, 1.2468, 1.4529, 1.1170, 0.8161,
        1.2931, 1.1458, 1.0629, 1.5305, 0.9668, 1.1938, 1.0788, 1.4397, 1.0447,
        1.3210, 1.2400, 1.2297, 1.4338, 1.0968, 1.1956, 0.9647, 1.1379, 1.3762,
        1.1042, 1.1069, 0.9107, 0.9582, 1.3220, 1.2239, 1.0242, 1.2866, 0.8187,
        1.2076, 1.2896, 1.2960, 0.9841, 1.7076, 1.0334, 1.4528, 1.2513, 1.1532,
        1.4862, 1.0786, 1.6170, 1.1578, 0.9629, 0.6877, 1.1407, 1.5325, 1.0445,
        1.0795, 1.2758, 0.8552, 0.8673, 1.3360, 1.3203, 1.0897, 1.1103, 1.4319,
        1.5586, 1.1282, 1.4560, 1.2513, 1.0493, 1.5836, 1.4642, 1.0819, 1.0885,
        1.2572, 1.3653, 1.0686, 0.9978, 1.2645, 1.4546, 1.1011, 1.1458, 1.2378,
        0.9934, 1.4495, 0.8324, 1.5057, 1.3501, 1.2225, 1.1273, 1.1452, 1.2017,
        1.6655, 1.7075, 1.2866, 1.1609, 1.0325, 1.3373, 1.7345, 1.0403, 1.0576,
        1.0330, 0.9363, 1.3330, 0.8286, 1.0074, 1.1996, 1.0864, 0.9712, 1.0027,
        0.9470, 1.1494, 1.2603, 1.5358, 1.0559, 1.2103, 1.1398, 1.3250, 1.5552,
        1.4122, 1.6717, 1.5053, 1.2454, 1.4459, 1.5942, 1.1914, 1.7158, 1.2774,
        1.2664, 1.3205, 1.0258, 1.3510, 1.4146, 1.2977, 1.2540, 1.2409, 1.4807,
        1.2549, 1.6740, 1.5980, 1.1636, 1.2593, 1.2593, 0.9933, 0.9663, 1.3337,
        1.4483, 1.2067, 1.2403, 1.4244, 0.9873, 1.2008, 1.1438, 1.1660, 1.0038,
        1.5695, 1.7022, 1.0299, 1.6018, 1.4194, 1.2718, 1.5852, 1.0956, 1.3453,
        1.6196, 1.3532, 1.0184, 1.3210, 1.7928, 0.9330, 1.2070, 1.1585, 1.0563,
        1.3560, 1.1584, 0.8802, 0.9189, 1.0884, 1.5750, 1.1342, 1.0477, 0.9706,
        0.9177, 1.1745, 1.1757, 1.1751, 1.2095, 1.3599, 1.3636, 1.2212, 1.1839,
        1.3990, 1.1138, 1.1733, 1.2042, 0.9836, 1.3273, 1.4003, 1.0727, 0.9773,
        0.9470, 0.9258, 1.3979, 1.2116, 1.5332, 1.4542, 1.3231, 0.8584, 1.2207,
        1.5548, 1.3387, 0.8971, 1.4064, 1.3890, 1.0418, 0.9298, 1.5528, 0.7671,
        1.9038, 1.3876, 1.5301, 1.2482, 1.4987, 1.5911, 0.9695, 1.5308, 0.9527,
        1.2256, 1.3200, 1.0004, 1.3141, 1.2098, 1.2739, 1.0112, 1.3561, 1.4298,
        1.0023, 1.2974, 1.1323, 1.4836, 1.3612, 1.0628, 1.5690, 1.0692, 1.3971,
        1.1786, 1.5271, 0.9096, 1.1054], device='cuda:0')), ('conv10_bn.num_batches_tracked', tensc
r(8160, device='cuda:0')), ('fc1.weight', tensor([[-0.0014, -0.0012, -0.0032,  ..., -0.0059,
0.0019,  0.0055],
        [-0.0060,  0.0037, -0.0055,  ..., -0.0058, -0.0038, -0.0064],
        [-0.0046,  0.0016,  0.0005,  ..., -0.0029, -0.0044, -0.0043],
        ...,
        [ 0.0037, -0.0030, -0.0002,  ..., -0.0044,  0.0032, -0.0020],
        [-0.0062, -0.0013, -0.0018,  ..., -0.0008,  0.0001, -0.0027],
        [-0.0039,  0.0027,  0.0018,  ..., -0.0043, -0.0006, -0.0065]],
       device='cuda:0')), ('fc1.bias', tensor([ 0.0039, -0.0014, -0.0017, -0.0018,  0.0095, -
0.0065,  0.0042,  0.0049,
         0.0002,  0.0048,  0.0032, -0.0054,  0.0084, -0.0061,  0.0026,  0.0037,
        -0.0044, -0.0021,  0.0048,  0.0150, -0.0063,  0.0025, -0.0021,  0.0000,
         0.0052, -0.0005, -0.0058,  0.0026,  0.0100, -0.0057, -0.0049,  0.0046,
        -0.0011, -0.0013,  0.0040,  0.0040,  0.0006,  0.0052, -0.0019, -0.0002,
        -0.0024,  0.0124,  0.0019,  0.0022, -0.0026,  0.0042, -0.0066, -0.0020,
         0.0017, -0.0050,  0.0010, -0.0061,  0.0046,  0.0023,  0.0013, -0.0029,
        -0.0040, -0.0048,  0.0072,  0.0003, -0.0027,  0.0056, -0.0029, -0.0039,
        -0.0016,  0.0088, -0.0010,  0.0011, -0.0032, -0.0048, -0.0062, -0.0038,
        -0.0035, -0.0049,  0.0019,  0.0015, -0.0018, -0.0040,  0.0018,  0.0083,
         0.0036, -0.0017,  0.0012, -0.0060,  0.0053,  0.0029,  0.0130,  0.0059,
         0.0123,  0.0001, -0.0010, -0.0005, -0.0016,  0.0018,  0.0015, -0.0052,
        -0.0068,  0.0027,  0.0122,  0.0095, -0.0059,  0.0055, -0.0026, -0.0063,
        -0.0066,  0.0108,  0.0008, -0.0028, -0.0029,  0.0005,  0.0055,  0.0096,
         0.0097,  0.0013,  0.0055,  0.0032,  0.0042, -0.0065, -0.0031, -0.0007,
         0.0044,  0.0023,  0.0056, -0.0014,  0.0044,  0.0078, -0.0065, -0.0059,
         0.0058,  0.0006, -0.0052,  0.0026, -0.0046, -0.0031, -0.0017,  0.0060,
        -0.0052,  0.0080, -0.0005, -0.0028,  0.0036, -0.0006, -0.0054, -0.0032,
        -0.0049, -0.0064,  0.0049, -0.0044,  0.0011,  0.0004,  0.0042,  0.0037,
         0.0004, -0.0021,  0.0071,  0.0014, -0.0037, -0.0016, -0.0021, -0.0041,
        -0.0018,  0.0006,  0.0045, -0.0052,  0.0027,  0.0004, -0.0064,  0.0054,
         0.0038,  0.0056, -0.0064, -0.0006,  0.0014, -0.0022,  0.0018,  0.0020,
        -0.0038, -0.0043,  0.0102,  0.0096, -0.0039, -0.0034,  0.0037,  0.0102,
         0.0030,  0.0062,  0.0015, -0.0045,  0.0009,  0.0051,  0.0008, -0.0037,
        -0.0039,  0.0020,  0.0040, -0.0046,  0.0047, -0.0011,  0.0055,  0.0039,
        -0.0028,  0.0045, -0.0055, -0.0041, -0.0002,  0.0034, -0.0044, -0.0011,
        -0.0023,  0.0111, -0.0005, -0.0038, -0.0009, -0.0006,  0.0028,  0.0063,
         0.0036, -0.0053,  0.0026, -0.0049,  0.0045, -0.0052, -0.0038,  0.0030,
        -0.0041,  0.0070, -0.0047, -0.0034,  0.0069, -0.0022, -0.0064,  0.0058,
        -0.0024,  0.0056,  0.0002,  0.0042,  0.0038, -0.0008, -0.0057, -0.0039,
```

```
         0.0096,  0.0000,  0.0082, -0.0033,  0.0017,  0.0052,  0.0063,  0.0089,
         0.0046,  0.0002, -0.0010,  0.0006, -0.0011,  0.0036,  0.0017,  0.0013,
        -0.0058, -0.0006,  0.0027,  0.0043,  0.0017, -0.0031, -0.0018,  0.0011,
         0.0010,  0.0027, -0.0044, -0.0037,  0.0005,  0.0082,  0.0051, -0.0061,
        -0.0034,  0.0048,  0.0056,  0.0072,  0.0010,  0.0043, -0.0048, -0.0035,
         0.0026,  0.0052, -0.0013,  0.0009,  0.0034, -0.0064,  0.0097, -0.0050,
         0.0047,  0.0056,  0.0073, -0.0002, -0.0004, -0.0019,  0.0032, -0.0009,
         0.0016, -0.0024,  0.0049,  0.0053, -0.0001, -0.0015,  0.0047,  0.0113,
         0.0035,  0.0055,  0.0043, -0.0036,  0.0010, -0.0044, -0.0034, -0.0022,
        -0.0060,  0.0032,  0.0007, -0.0016,  0.0052, -0.0020,  0.0137, -0.0036,
         0.0027,  0.0051, -0.0055,  0.0052,  0.0056, -0.0058,  0.0044,  0.0005,
        -0.0067, -0.0003,  0.0022, -0.0055,  0.0010,  0.0009,  0.0050, -0.0023,
        -0.0029, -0.0038, -0.0040,  0.0094, -0.0013, -0.0029,  0.0023,  0.0016,
        -0.0020,  0.0019, -0.0045,  0.0044, -0.0008,  0.0057, -0.0061,  0.0094,
        -0.0028, -0.0021,  0.0014,  0.0061, -0.0019,  0.0004,  0.0064,  0.0051,
         0.0056,  0.0013,  0.0035,  0.0129, -0.0040,  0.0020,  0.0048,  0.0031,
        -0.0031,  0.0002, -0.0060, -0.0006,  0.0037,  0.0051, -0.0029,  0.0044,
         0.0048,  0.0003,  0.0044,  0.0001, -0.0027, -0.0013, -0.0045, -0.0056,
         0.0021,  0.0034,  0.0053,  0.0030,  0.0027, -0.0036,  0.0048, -0.0052,
        -0.0042, -0.0002, -0.0013, -0.0022,  0.0109, -0.0030, -0.0008,  0.0113,
         0.0024, -0.0005,  0.0001, -0.0001, -0.0044,  0.0053, -0.0019, -0.0005,
        -0.0049, -0.0040,  0.0033,  0.0010,  0.0035,  0.0025, -0.0024,  0.0001,
         0.0129,  0.0022, -0.0010,  0.0106, -0.0005,  0.0018,  0.0097,  0.0013,
        -0.0015, -0.0056, -0.0011, -0.0034,  0.0022, -0.0027, -0.0052,  0.0043,
        -0.0061, -0.0062,  0.0003, -0.0052,  0.0028, -0.0012,  0.0037, -0.0043,
        -0.0025,  0.0019,  0.0075, -0.0038, -0.0068,  0.0048,  0.0102, -0.0007,
         0.0008,  0.0017,  0.0044, -0.0001,  0.0089, -0.0057, -0.0066,  0.0061,
        -0.0037,  0.0045, -0.0046, -0.0051, -0.0001, -0.0019,  0.0033,  0.0076,
         0.0049,  0.0040, -0.0013,  0.0070, -0.0004,  0.0003, -0.0050,  0.0033,
        -0.0007,  0.0024,  0.0058,  0.0017, -0.0057, -0.0045, -0.0020, -0.0002,
         0.0040,  0.0097, -0.0007,  0.0031,  0.0011,  0.0037, -0.0062,  0.0047,
         0.0012,  0.0056,  0.0062,  0.0023,  0.0050, -0.0037,  0.0019,  0.0009,
        -0.0007, -0.0009,  0.0019,  0.0001, -0.0065,  0.0005,  0.0049,  0.0036,
        -0.0006,  0.0007,  0.0007, -0.0000,  0.0014, -0.0007,  0.0001,  0.0028],
       device='cuda:0')), ('fc2.weight', tensor([[ 0.0002,  0.0227,  0.0370,  ...,  0.0170,  0.0276
, -0.0272],
        [-0.0281, -0.0263,  0.0078,  ..., -0.0260, -0.0428, -0.0196],
        [ 0.0111, -0.0243, -0.0267,  ...,  0.0145, -0.0112, -0.0366],
        ...,
        [ 0.0020, -0.0330, -0.0411,  ..., -0.0363,  0.0249, -0.0144],
        [ 0.0152, -0.0018, -0.0316,  ...,  0.0114, -0.0036,  0.0256],
        [ 0.0183, -0.0189, -0.0335,  ...,  0.0152,  0.0124,  0.0029]],
       device='cuda:0')), ('fc2.bias', tensor([-0.0277,  0.0015, -0.0240,  0.0406, -0.0146, -0.0343
,  0.0339,  0.0166,
         0.0474,  0.0490,  0.0013,  0.0383,  0.0142, -0.0102,  0.0066, -0.0287,
        -0.0171,  0.0166, -0.0271, -0.0080, -0.0085, -0.0115,  0.0190, -0.0228,
         0.0297,  0.0106, -0.0193,  0.0249,  0.0004,  0.0444, -0.0161, -0.0266,
        -0.0259, -0.0244,  0.0146, -0.0209,  0.0474,  0.0255,  0.0440, -0.0407,
        -0.0317,  0.0047, -0.0120, -0.0090, -0.0372,  0.0299,  0.0155, -0.0204,
         0.0303, -0.0128, -0.0262, -0.0181, -0.0144, -0.0440, -0.0281, -0.0190,
         0.0014, -0.0140, -0.0184, -0.0073,  0.0402,  0.0357, -0.0295,  0.0157,
        -0.0039,  0.0286, -0.0333, -0.0107, -0.0272, -0.0002, -0.0266,  0.0305,
        -0.0038, -0.0256, -0.0243, -0.0022,  0.0361,  0.0152,  0.0069, -0.0114,
        -0.0187,  0.0285, -0.0099, -0.0251, -0.0434, -0.0304,  0.0401, -0.0159,
         0.0089,  0.0187, -0.0430, -0.0330,  0.0210,  0.0182,  0.0326,  0.0208,
        -0.0396,  0.0404,  0.0077,  0.0235, -0.0230,  0.0037,  0.0259,  0.0219,
         0.0053, -0.0283,  0.0117, -0.0163,  0.0030,  0.0446, -0.0074,  0.0384,
         0.0010,  0.0603, -0.0124,  0.0194,  0.0394, -0.0482,  0.0376, -0.0083,
        -0.0348,  0.0186, -0.0440, -0.0326,  0.0093, -0.0207,  0.0351, -0.0210,
        -0.0388, -0.0012, -0.0374,  0.0304,  0.0407, -0.0063, -0.0028, -0.0147,
        -0.0147,  0.0360, -0.0147,  0.0204,  0.0417, -0.0166,  0.0352, -0.0031,
         0.0115,  0.0109,  0.0422,  0.0353,  0.0000, -0.0015, -0.0459,  0.0380,
        -0.0034,  0.0107,  0.0348,  0.0140,  0.0281,  0.0068,  0.0148, -0.0015,
         0.0326, -0.0246, -0.0198, -0.0160, -0.0293,  0.0521,  0.0345,  0.0126,
         0.0389,  0.0230,  0.0448,  0.0220,  0.0032, -0.0439, -0.0166, -0.0206,
        -0.0285, -0.0472, -0.0264, -0.0297, -0.0355,  0.0237,  0.0360, -0.0155,
         0.0381,  0.0001,  0.0302, -0.0460,  0.0359, -0.0049,  0.0215,  0.0005,
         0.0524, -0.0230,  0.0012,  0.0004, -0.0418,  0.0447, -0.0354, -0.0050,
        -0.0194,  0.0386,  0.0352,  0.0298, -0.0026,  0.0003,  0.0137, -0.0228,
         0.0508, -0.0303,  0.0196, -0.0357, -0.0273, -0.0265,  0.0376, -0.0246,
        -0.0292, -0.0369, -0.0043,  0.0191,  0.0135,  0.0193, -0.0057, -0.0378,
        -0.0108, -0.0274,  0.0171, -0.0135, -0.0047, -0.0166,  0.0020,  0.0235,
        -0.0267,  0.0132, -0.0357, -0.0153, -0.0034,  0.0139, -0.0110, -0.0440,
         0.0313, -0.0083,  0.0058, -0.0107,  0.0267,  0.0501, -0.0344,  0.0200,
        -0.0320, -0.0226,  0.0408,  0.0310,  0.0032, -0.0022,  0.0206, -0.0262],
       device='cuda:0')), ('fc3.weight', tensor([[ 0.0589, -0.0480, -0.0001,  ...,  0.0276, -0.0456
,  0.0393],
```

```
        [-0.0331, -0.0007, -0.0290,  ..., -0.0039, -0.0172, -0.0226],
        [ 0.0237, -0.0216, -0.0094,  ...,  0.0204,  0.0042,  0.0030],
        ...,
        [-0.0124,  0.0459, -0.0652,  ..., -0.0627,  0.0389, -0.0150],
        [ 0.0567, -0.0199,  0.0350,  ...,  0.0588, -0.0322, -0.0356],
        [ 0.0549, -0.0369, -0.0619,  ..., -0.0372, -0.0175,  0.0435]],
       device='cuda:0')), ('fc3.bias', tensor([-0.0323,  0.0570, -0.0589, -0.0096,  0.0286,  0.0466
,  0.0133,  0.0247,
        -0.0095,  0.0171, -0.0199,  0.0123,  0.0033,  0.0032,  0.0590,  0.0153,
         0.0094, -0.0296, -0.0488, -0.0480, -0.0674, -0.0109, -0.0566, -0.0512,
         0.0559, -0.0098, -0.0222,  0.0330,  0.0486, -0.0623, -0.0445,  0.0489,
         0.0353, -0.0198, -0.0405,  0.0638, -0.0426, -0.0529,  0.0026, -0.0379,
         0.0522,  0.0189,  0.0134, -0.0013, -0.0166,  0.0534, -0.0354, -0.0425,
         0.0324, -0.0064,  0.0028,  0.0355,  0.0198, -0.0269,  0.0074,  0.0178,
         0.0320,  0.0403, -0.0613, -0.0251, -0.0275, -0.0520, -0.0208, -0.0217,
         0.0629,  0.0330,  0.0648,  0.0319,  0.0589, -0.0423, -0.0131,  0.0059,
         0.0056,  0.0377, -0.0377,  0.0703,  0.0457, -0.0237,  0.0464,  0.0292,
         0.0226,  0.0377,  0.0525,  0.0455, -0.0313, -0.0063,  0.0527,  0.0436,
        -0.0420, -0.0195, -0.0069,  0.0017, -0.0606, -0.0160,  0.0428, -0.0140,
         0.0057,  0.0129, -0.0146,  0.0244, -0.0483, -0.0197, -0.0684,  0.0072,
         0.0470,  0.0297, -0.0535, -0.0157, -0.0499, -0.0263,  0.0359, -0.0614,
        -0.0574,  0.0115, -0.0606,  0.0374, -0.0616, -0.0295, -0.0186, -0.0423],
       device='cuda:0')), ('fc4.weight', tensor([[ 0.0716, -0.0167, -0.0153,  ..., -0.0480, -
0.0229, -0.0882],
        [ 0.0693,  0.0740, -0.0413,  ...,  0.0466, -0.0232,  0.0098],
        [-0.0867, -0.0048,  0.0696,  ...,  0.0053, -0.0931, -0.0362],
        ...,
        [ 0.0328, -0.0253, -0.0669,  ..., -0.0085, -0.0262,  0.0517],
        [-0.0804, -0.0090, -0.0710,  ..., -0.0607, -0.0046, -0.0525],
        [-0.0549, -0.0477, -0.0443,  ...,  0.0757,  0.0267,  0.0233]],
       device='cuda:0')), ('fc4.bias', tensor([ 0.0453,  0.0293, -0.0254, -0.0443,  0.0712,  0.0572
,  0.0652,  0.0519,
        -0.0531,  0.0123, -0.0122, -0.0401,  0.0184, -0.0144,  0.0712, -0.0003,
        -0.0179,  0.0853,  0.0131,  0.0837, -0.0631,  0.0350,  0.0670, -0.0248,
        -0.0489,  0.0447,  0.0031, -0.0088,  0.0898,  0.0306,  0.0860, -0.0529,
        -0.0134, -0.0049,  0.0675,  0.0358,  0.0821,  0.0961, -0.0686, -0.0165,
        -0.0271,  0.0062,  0.0653, -0.0306,  0.0059,  0.0647,  0.0805, -0.0680,
        -0.0607, -0.0212, -0.0180, -0.0168, -0.0297,  0.0823,  0.0131, -0.0655,
         0.0460,  0.0195, -0.0742, -0.0152, -0.0064, -0.0360, -0.0451,  0.0728,
        -0.0458,  0.0608, -0.0084, -0.0087, -0.0727, -0.0411,  0.0259,  0.0475,
         0.0051,  0.0654, -0.0499,  0.0257, -0.0570,  0.0557, -0.0762,  0.0056,
         0.0316,  0.0955, -0.0079,  0.0254], device='cuda:0')), ('fc5.weight', tensor([[ 0.0142, -0
.1379, -0.4771,  ...,  0.0908, -0.1196,  0.0444],
        [-0.0626, -0.1112,  0.0393,  ..., -0.0272,  0.0505, -0.0236],
        [-0.0648, -0.1147, -0.1023,  ..., -0.0781, -0.1307, -0.0795],
        ...,
        [ 0.0402,  0.0536, -0.0618,  ...,  0.0473,  0.0031,  0.0872],
        [-0.0700, -0.0236, -0.0545,  ...,  0.0560,  0.0286, -0.1046],
        [ 0.0123,  0.0761, -0.0776,  ..., -0.1179, -0.0396,  0.0058]],
       device='cuda:0')), ('fc5.bias', tensor([-0.0446,  0.0917,  0.0640, -0.1023,  0.0632,  0.0883
, -0.0751,  0.0145,
         0.0404, -0.0932, -0.0266,  0.0905,  0.0017,  0.0793,  0.0597,  0.0854,
        -0.0934,  0.0540,  0.0681,  0.0248, -0.0811], device='cuda:0'))])
```

Evaluate on test set

In [0]:

```
ds_test = VocDataset('VOCdevkit_2007/VOC2007test/','test', test_transform)

test_loader = torch.utils.data.DataLoader(dataset=ds_test,
                                          batch_size=50,
                                          shuffle=False,
                                          num_workers=1)

mAP_test, test_loss, test_aps = test_classifier(test_loader, classifier, criterion)
```

```
annotation ind  0
annotation ind  10
annotation ind  20
annotation ind  30
annotation ind  40
annotation ind  50
annotation ind  60
```

```
annotation ind  60
annotation ind  70
annotation ind  80
annotation ind  90
annotation ind  100
annotation ind  110
annotation ind  120
annotation ind  130
annotation ind  140
annotation ind  150
annotation ind  160
annotation ind  170
annotation ind  180
annotation ind  190
annotation ind  200
annotation ind  210
annotation ind  220
annotation ind  230
annotation ind  240
annotation ind  250
annotation ind  260
annotation ind  270
annotation ind  280
annotation ind  290
annotation ind  300
annotation ind  310
annotation ind  320
annotation ind  330
annotation ind  340
annotation ind  350
annotation ind  360
annotation ind  370
annotation ind  380
annotation ind  390
annotation ind  400
annotation ind  410
annotation ind  420
annotation ind  430
annotation ind  440
annotation ind  450
annotation ind  460
annotation ind  470
annotation ind  480
annotation ind  490
annotation ind  500
annotation ind  510
annotation ind  520
annotation ind  530
annotation ind  540
annotation ind  550
annotation ind  560
annotation ind  570
annotation ind  580
annotation ind  590
annotation ind  600
annotation ind  610
annotation ind  620
annotation ind  630
annotation ind  640
annotation ind  650
annotation ind  660
annotation ind  670
annotation ind  680
annotation ind  690
annotation ind  700
annotation ind  710
annotation ind  720
annotation ind  730
annotation ind  740
annotation ind  750
annotation ind  760
annotation ind  770
annotation ind  780
annotation ind  790
annotation ind  800
annotation ind  810
annotation ind  820
```

```
annotation ind  830
annotation ind  840
annotation ind  850
annotation ind  860
annotation ind  870
annotation ind  880
annotation ind  890
annotation ind  900
annotation ind  910
annotation ind  920
annotation ind  930
annotation ind  940
annotation ind  950
annotation ind  960
annotation ind  970
annotation ind  980
annotation ind  990
annotation ind  1000
annotation ind  1010
annotation ind  1020
annotation ind  1030
annotation ind  1040
annotation ind  1050
annotation ind  1060
annotation ind  1070
annotation ind  1080
annotation ind  1090
annotation ind  1100
annotation ind  1110
annotation ind  1120
annotation ind  1130
annotation ind  1140
annotation ind  1150
annotation ind  1160
annotation ind  1170
annotation ind  1180
annotation ind  1190
annotation ind  1200
annotation ind  1210
annotation ind  1220
annotation ind  1230
annotation ind  1240
annotation ind  1250
annotation ind  1260
annotation ind  1270
annotation ind  1280
annotation ind  1290
annotation ind  1300
annotation ind  1310
annotation ind  1320
annotation ind  1330
annotation ind  1340
annotation ind  1350
annotation ind  1360
annotation ind  1370
annotation ind  1380
annotation ind  1390
annotation ind  1400
annotation ind  1410
annotation ind  1420
annotation ind  1430
annotation ind  1440
annotation ind  1450
annotation ind  1460
annotation ind  1470
annotation ind  1480
annotation ind  1490
annotation ind  1500
annotation ind  1510
annotation ind  1520
annotation ind  1530
annotation ind  1540
annotation ind  1550
annotation ind  1560
annotation ind  1570
annotation ind  1580
annotation ind  1590
```

```
annotation ind  1600
annotation ind  1610
annotation ind  1620
annotation ind  1630
annotation ind  1640
annotation ind  1650
annotation ind  1660
annotation ind  1670
annotation ind  1680
annotation ind  1690
annotation ind  1700
annotation ind  1710
annotation ind  1720
annotation ind  1730
annotation ind  1740
annotation ind  1750
annotation ind  1760
annotation ind  1770
annotation ind  1780
annotation ind  1790
annotation ind  1800
annotation ind  1810
annotation ind  1820
annotation ind  1830
annotation ind  1840
annotation ind  1850
annotation ind  1860
annotation ind  1870
annotation ind  1880
annotation ind  1890
annotation ind  1900
annotation ind  1910
annotation ind  1920
annotation ind  1930
annotation ind  1940
annotation ind  1950
annotation ind  1960
annotation ind  1970
annotation ind  1980
annotation ind  1990
annotation ind  2000
annotation ind  2010
annotation ind  2020
annotation ind  2030
annotation ind  2040
annotation ind  2050
annotation ind  2060
annotation ind  2070
annotation ind  2080
annotation ind  2090
annotation ind  2100
annotation ind  2110
annotation ind  2120
annotation ind  2130
annotation ind  2140
annotation ind  2150
annotation ind  2160
annotation ind  2170
annotation ind  2180
annotation ind  2190
annotation ind  2200
annotation ind  2210
annotation ind  2220
annotation ind  2230
annotation ind  2240
annotation ind  2250
annotation ind  2260
annotation ind  2270
annotation ind  2280
annotation ind  2290
annotation ind  2300
annotation ind  2310
annotation ind  2320
annotation ind  2330
annotation ind  2340
annotation ind  2350
annotation ind  2360
```

```
annotation ind  2370
annotation ind  2380
annotation ind  2390
annotation ind  2400
annotation ind  2410
annotation ind  2420
annotation ind  2430
annotation ind  2440
annotation ind  2450
annotation ind  2460
annotation ind  2470
annotation ind  2480
annotation ind  2490
annotation ind  2500
annotation ind  2510
annotation ind  2520
annotation ind  2530
annotation ind  2540
annotation ind  2550
annotation ind  2560
annotation ind  2570
annotation ind  2580
annotation ind  2590
annotation ind  2600
annotation ind  2610
annotation ind  2620
annotation ind  2630
annotation ind  2640
annotation ind  2650
annotation ind  2660
annotation ind  2670
annotation ind  2680
annotation ind  2690
annotation ind  2700
annotation ind  2710
annotation ind  2720
annotation ind  2730
annotation ind  2740
annotation ind  2750
annotation ind  2760
annotation ind  2770
annotation ind  2780
annotation ind  2790
annotation ind  2800
annotation ind  2810
annotation ind  2820
annotation ind  2830
annotation ind  2840
annotation ind  2850
annotation ind  2860
annotation ind  2870
annotation ind  2880
annotation ind  2890
annotation ind  2900
annotation ind  2910
annotation ind  2920
annotation ind  2930
annotation ind  2940
annotation ind  2950
annotation ind  2960
annotation ind  2970
annotation ind  2980
annotation ind  2990
annotation ind  3000
annotation ind  3010
annotation ind  3020
annotation ind  3030
annotation ind  3040
annotation ind  3050
annotation ind  3060
annotation ind  3070
annotation ind  3080
annotation ind  3090
annotation ind  3100
annotation ind  3110
annotation ind  3120
annotation ind  3130
```

```
annotation ind  3140
annotation ind  3150
annotation ind  3160
annotation ind  3170
annotation ind  3180
annotation ind  3190
annotation ind  3200
annotation ind  3210
annotation ind  3220
annotation ind  3230
annotation ind  3240
annotation ind  3250
annotation ind  3260
annotation ind  3270
annotation ind  3280
annotation ind  3290
annotation ind  3300
annotation ind  3310
annotation ind  3320
annotation ind  3330
annotation ind  3340
annotation ind  3350
annotation ind  3360
annotation ind  3370
annotation ind  3380
annotation ind  3390
annotation ind  3400
annotation ind  3410
annotation ind  3420
annotation ind  3430
annotation ind  3440
annotation ind  3450
annotation ind  3460
annotation ind  3470
annotation ind  3480
annotation ind  3490
annotation ind  3500
annotation ind  3510
annotation ind  3520
annotation ind  3530
annotation ind  3540
annotation ind  3550
annotation ind  3560
annotation ind  3570
annotation ind  3580
annotation ind  3590
annotation ind  3600
annotation ind  3610
annotation ind  3620
annotation ind  3630
annotation ind  3640
annotation ind  3650
annotation ind  3660
annotation ind  3670
annotation ind  3680
annotation ind  3690
annotation ind  3700
annotation ind  3710
annotation ind  3720
annotation ind  3730
annotation ind  3740
annotation ind  3750
annotation ind  3760
annotation ind  3770
annotation ind  3780
annotation ind  3790
annotation ind  3800
annotation ind  3810
annotation ind  3820
annotation ind  3830
annotation ind  3840
annotation ind  3850
annotation ind  3860
annotation ind  3870
annotation ind  3880
annotation ind  3890
annotation ind  3900
```

```
annotation ind  3910
annotation ind  3920
annotation ind  3930
annotation ind  3940
annotation ind  3950
annotation ind  3960
annotation ind  3970
annotation ind  3980
annotation ind  3990
annotation ind  4000
annotation ind  4010
annotation ind  4020
annotation ind  4030
annotation ind  4040
annotation ind  4050
annotation ind  4060
annotation ind  4070
annotation ind  4080
annotation ind  4090
annotation ind  4100
annotation ind  4110
annotation ind  4120
annotation ind  4130
annotation ind  4140
annotation ind  4150
annotation ind  4160
annotation ind  4170
annotation ind  4180
annotation ind  4190
annotation ind  4200
annotation ind  4210
annotation ind  4220
annotation ind  4230
annotation ind  4240
annotation ind  4250
annotation ind  4260
annotation ind  4270
annotation ind  4280
annotation ind  4290
annotation ind  4300
annotation ind  4310
annotation ind  4320
annotation ind  4330
annotation ind  4340
annotation ind  4350
annotation ind  4360
annotation ind  4370
annotation ind  4380
annotation ind  4390
annotation ind  4400
annotation ind  4410
annotation ind  4420
annotation ind  4430
annotation ind  4440
annotation ind  4450
annotation ind  4460
annotation ind  4470
annotation ind  4480
annotation ind  4490
annotation ind  4500
annotation ind  4510
annotation ind  4520
annotation ind  4530
annotation ind  4540
annotation ind  4550
annotation ind  4560
annotation ind  4570
annotation ind  4580
annotation ind  4590
annotation ind  4600
annotation ind  4610
annotation ind  4620
annotation ind  4630
annotation ind  4640
annotation ind  4650
annotation ind  4660
annotation ind  4670
```

```
annotation ind  4680
annotation ind  4690
annotation ind  4700
annotation ind  4710
annotation ind  4720
annotation ind  4730
annotation ind  4740
annotation ind  4750
annotation ind  4760
annotation ind  4770
annotation ind  4780
annotation ind  4790
annotation ind  4800
annotation ind  4810
annotation ind  4820
annotation ind  4830
annotation ind  4840
annotation ind  4850
annotation ind  4860
annotation ind  4870
annotation ind  4880
annotation ind  4890
annotation ind  4900
annotation ind  4910
annotation ind  4920
annotation ind  4930
annotation ind  4940
annotation ind  4950
Begin loading images
loaded image  0
loaded image  10
loaded image  20
loaded image  30
loaded image  40
loaded image  50
loaded image  60
loaded image  70
loaded image  80
loaded image  90
loaded image  100
loaded image  110
loaded image  120
loaded image  130
loaded image  140
loaded image  150
loaded image  160
loaded image  170
loaded image  180
loaded image  190
loaded image  200
loaded image  210
loaded image  220
loaded image  230
loaded image  240
loaded image  250
loaded image  260
loaded image  270
loaded image  280
loaded image  290
loaded image  300
loaded image  310
loaded image  320
loaded image  330
loaded image  340
loaded image  350
loaded image  360
loaded image  370
loaded image  380
loaded image  390
loaded image  400
loaded image  410
loaded image  420
loaded image  430
loaded image  440
loaded image  450
loaded image  460
loaded image  470
```

```
loaded image  480
loaded image  490
loaded image  500
loaded image  510
loaded image  520
loaded image  530
loaded image  540
loaded image  550
loaded image  560
loaded image  570
loaded image  580
loaded image  590
loaded image  600
loaded image  610
loaded image  620
loaded image  630
loaded image  640
loaded image  650
loaded image  660
loaded image  670
loaded image  680
loaded image  690
loaded image  700
loaded image  710
loaded image  720
loaded image  730
loaded image  740
loaded image  750
loaded image  760
loaded image  770
loaded image  780
loaded image  790
loaded image  800
loaded image  810
loaded image  820
loaded image  830
loaded image  840
loaded image  850
loaded image  860
loaded image  870
loaded image  880
loaded image  890
loaded image  900
loaded image  910
loaded image  920
loaded image  930
loaded image  940
loaded image  950
loaded image  960
loaded image  970
loaded image  980
loaded image  990
loaded image  1000
loaded image  1010
loaded image  1020
loaded image  1030
loaded image  1040
loaded image  1050
loaded image  1060
loaded image  1070
loaded image  1080
loaded image  1090
loaded image  1100
loaded image  1110
loaded image  1120
loaded image  1130
loaded image  1140
loaded image  1150
loaded image  1160
loaded image  1170
loaded image  1180
loaded image  1190
loaded image  1200
loaded image  1210
loaded image  1220
loaded image  1230
loaded image  1240
```

```
loaded image  1250
loaded image  1260
loaded image  1270
loaded image  1280
loaded image  1290
loaded image  1300
loaded image  1310
loaded image  1320
loaded image  1330
loaded image  1340
loaded image  1350
loaded image  1360
loaded image  1370
loaded image  1380
loaded image  1390
loaded image  1400
loaded image  1410
loaded image  1420
loaded image  1430
loaded image  1440
loaded image  1450
loaded image  1460
loaded image  1470
loaded image  1480
loaded image  1490
loaded image  1500
loaded image  1510
loaded image  1520
loaded image  1530
loaded image  1540
loaded image  1550
loaded image  1560
loaded image  1570
loaded image  1580
loaded image  1590
loaded image  1600
loaded image  1610
loaded image  1620
loaded image  1630
loaded image  1640
loaded image  1650
loaded image  1660
loaded image  1670
loaded image  1680
loaded image  1690
loaded image  1700
loaded image  1710
loaded image  1720
loaded image  1730
loaded image  1740
loaded image  1750
loaded image  1760
loaded image  1770
loaded image  1780
loaded image  1790
loaded image  1800
loaded image  1810
loaded image  1820
loaded image  1830
loaded image  1840
loaded image  1850
loaded image  1860
loaded image  1870
loaded image  1880
loaded image  1890
loaded image  1900
loaded image  1910
loaded image  1920
loaded image  1930
loaded image  1940
loaded image  1950
loaded image  1960
loaded image  1970
loaded image  1980
loaded image  1990
loaded image  2000
loaded image  2010
```

```
loaded image  2020
loaded image  2030
loaded image  2040
loaded image  2050
loaded image  2060
loaded image  2070
loaded image  2080
loaded image  2090
loaded image  2100
loaded image  2110
loaded image  2120
loaded image  2130
loaded image  2140
loaded image  2150
loaded image  2160
loaded image  2170
loaded image  2180
loaded image  2190
loaded image  2200
loaded image  2210
loaded image  2220
loaded image  2230
loaded image  2240
loaded image  2250
loaded image  2260
loaded image  2270
loaded image  2280
loaded image  2290
loaded image  2300
loaded image  2310
loaded image  2320
loaded image  2330
loaded image  2340
loaded image  2350
loaded image  2360
loaded image  2370
loaded image  2380
loaded image  2390
loaded image  2400
loaded image  2410
loaded image  2420
loaded image  2430
loaded image  2440
loaded image  2450
loaded image  2460
loaded image  2470
loaded image  2480
loaded image  2490
loaded image  2500
loaded image  2510
loaded image  2520
loaded image  2530
loaded image  2540
loaded image  2550
loaded image  2560
loaded image  2570
loaded image  2580
loaded image  2590
loaded image  2600
loaded image  2610
loaded image  2620
loaded image  2630
loaded image  2640
loaded image  2650
loaded image  2660
loaded image  2670
loaded image  2680
loaded image  2690
loaded image  2700
loaded image  2710
loaded image  2720
loaded image  2730
loaded image  2740
loaded image  2750
loaded image  2760
loaded image  2770
loaded image  2780
```

```
loaded image  2790
loaded image  2800
loaded image  2810
loaded image  2820
loaded image  2830
loaded image  2840
loaded image  2850
loaded image  2860
loaded image  2870
loaded image  2880
loaded image  2890
loaded image  2900
loaded image  2910
loaded image  2920
loaded image  2930
loaded image  2940
loaded image  2950
loaded image  2960
loaded image  2970
loaded image  2980
loaded image  2990
loaded image  3000
loaded image  3010
loaded image  3020
loaded image  3030
loaded image  3040
loaded image  3050
loaded image  3060
loaded image  3070
loaded image  3080
loaded image  3090
loaded image  3100
loaded image  3110
loaded image  3120
loaded image  3130
loaded image  3140
loaded image  3150
loaded image  3160
loaded image  3170
loaded image  3180
loaded image  3190
loaded image  3200
loaded image  3210
loaded image  3220
loaded image  3230
loaded image  3240
loaded image  3250
loaded image  3260
loaded image  3270
loaded image  3280
loaded image  3290
loaded image  3300
loaded image  3310
loaded image  3320
loaded image  3330
loaded image  3340
loaded image  3350
loaded image  3360
loaded image  3370
loaded image  3380
loaded image  3390
loaded image  3400
loaded image  3410
loaded image  3420
loaded image  3430
loaded image  3440
loaded image  3450
loaded image  3460
loaded image  3470
loaded image  3480
loaded image  3490
loaded image  3500
loaded image  3510
loaded image  3520
loaded image  3530
loaded image  3540
loaded image  3550
```

```
loaded image  3560
loaded image  3570
loaded image  3580
loaded image  3590
loaded image  3600
loaded image  3610
loaded image  3620
loaded image  3630
loaded image  3640
loaded image  3650
loaded image  3660
loaded image  3670
loaded image  3680
loaded image  3690
loaded image  3700
loaded image  3710
loaded image  3720
loaded image  3730
loaded image  3740
loaded image  3750
loaded image  3760
loaded image  3770
loaded image  3780
loaded image  3790
loaded image  3800
loaded image  3810
loaded image  3820
loaded image  3830
loaded image  3840
loaded image  3850
loaded image  3860
loaded image  3870
loaded image  3880
loaded image  3890
loaded image  3900
loaded image  3910
loaded image  3920
loaded image  3930
loaded image  3940
loaded image  3950
loaded image  3960
loaded image  3970
loaded image  3980
loaded image  3990
loaded image  4000
loaded image  4010
loaded image  4020
loaded image  4030
loaded image  4040
loaded image  4050
loaded image  4060
loaded image  4070
loaded image  4080
loaded image  4090
loaded image  4100
loaded image  4110
loaded image  4120
loaded image  4130
loaded image  4140
loaded image  4150
loaded image  4160
loaded image  4170
loaded image  4180
loaded image  4190
loaded image  4200
loaded image  4210
loaded image  4220
loaded image  4230
loaded image  4240
loaded image  4250
loaded image  4260
loaded image  4270
loaded image  4280
loaded image  4290
loaded image  4300
loaded image  4310
loaded image  4320
```

```
loaded image  4330
loaded image  4340
loaded image  4350
loaded image  4360
loaded image  4370
loaded image  4380
loaded image  4390
loaded image  4400
loaded image  4410
loaded image  4420
loaded image  4430
loaded image  4440
loaded image  4450
loaded image  4460
loaded image  4470
loaded image  4480
loaded image  4490
loaded image  4500
loaded image  4510
loaded image  4520
loaded image  4530
loaded image  4540
loaded image  4550
loaded image  4560
loaded image  4570
loaded image  4580
loaded image  4590
loaded image  4600
loaded image  4610
loaded image  4620
loaded image  4630
loaded image  4640
loaded image  4650
loaded image  4660
loaded image  4670
loaded image  4680
loaded image  4690
loaded image  4700
loaded image  4710
loaded image  4720
loaded image  4730
loaded image  4740
loaded image  4750
loaded image  4760
loaded image  4770
loaded image  4780
loaded image  4790
loaded image  4800
loaded image  4810
loaded image  4820
loaded image  4830
loaded image  4840
loaded image  4850
loaded image  4860
loaded image  4870
loaded image  4880
loaded image  4890
loaded image  4900
loaded image  4910
loaded image  4920
loaded image  4930
loaded image  4940
loaded image  4950
-------  Class: aeroplane        AP:   0.6648  -------
-------  Class: bicycle          AP:   0.5027  -------
-------  Class: bird             AP:   0.3829  -------
-------  Class: boat             AP:   0.4632  -------
-------  Class: bottle           AP:   0.1938  -------
-------  Class: bus              AP:   0.3465  -------
-------  Class: car              AP:   0.6809  -------
-------  Class: cat              AP:   0.4489  -------
-------  Class: chair            AP:   0.4754  -------
-------  Class: cow              AP:   0.2402  -------
-------  Class: diningtable      AP:   0.3058  -------
-------  Class: dog              AP:   0.3658  -------
-------  Class: horse            AP:   0.6997  -------
-------  Class: motorbike        AP:   0.5377  -------
```

```
-------  Class: person           AP:   0.8331  -------
-------  Class: pottedplant      AP:   0.2448  -------
-------  Class: sheep            AP:   0.3362  -------
-------  Class: sofa             AP:   0.3518  -------
-------  Class: train            AP:   0.6272  -------
-------  Class: tvmonitor        AP:   0.3815  -------
mAP: 0.4542
Avg loss: 0.17731280624866486
```

In [0]:

```
output_submission_csv('my_solution.csv', test_aps)
```