```
import torch
import torchvision
from torchvision.models.detection import FasterRCNN
from torchvision.models.detection.rpn import AnchorGenerator
from torchvision.transforms import functional as F
from torch.utils.data import DataLoader
from torchvision import transforms
from torch.utils.data import Dataset
from os import listdir
from os.path import isfile, join
import json
from PIL import Image
import numpy as np
from torchvision.io import read_image
from torchvision.ops.boxes import masks_to_boxes
from torchvision import tv_tensors
from torchvision.transforms.v2 import functional as F
from torchvision.models.detection import fasterrcnn_resnet50_fpn
from torchvision.models.detection import keypointrcnn_resnet50_fpn
from torchvision.models.detection.faster_rcnn import FastRCNNPredictor
from torchvision.models.detection.faster_rcnn import TwoMLPHead
from torchvision.models.detection.faster_rcnn import RPNHead
from torchvision.models.detection.backbone_utils import resnet_fpn_backbone
from torchvision.models.detection.keypoint_rcnn import KeypointRCNNHeads
from torchvision.models.detection.keypoint_rcnn import KeypointRCNNPredictor
# from torchvision.ops import MultiScaleRoIAlign
import torch.nn.functional as FF
import torch.nn as nn
from torch import Tensor
from torchvision.models.detection.backbone_utils import BackboneWithFPN



class CustomDataset(Dataset):
    def __init__(self, image_folder, annotation_file, transform=None):
        self.imageFolderLen = 0
        self.image_folder = image_folder
        self.annotations = self.load_annotations(annotation_file)
        self.transform = transform

    def load_annotations(self, annotation_file):
        categoryLabels = []
        bboxLabels = []
        targets = []
        mypath = annotation_file
        onlyfiles = [f for f in listdir(mypath) if isfile(join(mypath, f))]
        i = 0
        for fileName in sorted(onlyfiles):
            f = open(mypath+fileName)
```

```
            f = open(mypath+fileName)
            dict1 = json.load(f)
            tmpBoxArr = []
            tmpKeypointsArr = []
            tmpCategoryArr = []
            for instance in dict1['annotations']:
                if instance["category_id"]==0:
                    tmpCategoryArr.append(2)
                else:
                    tmpCategoryArr.append(instance["category_id"])


                xmin = instance["bbox"][0] # image width is 1008 in the dataSet
                ymin = instance["bbox"][1] # image height is 756 in the dataSet
                xmax = instance["bbox"][2] # image width is 1008 in the dataSet
                ymax = instance["bbox"][3] # image height is 756 in the dataSet
                resizedbbox = [xmin,ymin,xmax,ymax]
                tmpBoxArr.append(resizedbbox)

                tmpKeypointsArr.append(np.array(instance["keypoints"]).reshape(-1,3))
            tmpTargets = {"boxes":tmpBoxArr,"labels":tmpCategoryArr,"image_id":[],"area":
            targets.append(tmpTargets)

        return targets

    def __len__(self):
        return len(self.annotations)

    def __getitem__(self, idx):
        # Load image and annotations
#         image_path = os.path.join(self.image_folder, self.annotations[idx]['image_path'
        onlyfiles = [f for f in listdir(self.image_folder) if isfile(join(self.image_fold
        onlyfiles = [ff for ff in sorted(onlyfiles)]
        self.imageFolderLen = len(onlyfiles)
        # image = Image.open().convert("RGB")

        img = read_image(self.image_folder+(onlyfiles[idx]))
        image = tv_tensors.Image(img)
        target = self.annotations[idx]
#         print(target["boxes"])
        target["labels"] = torch.Tensor(target["labels"]).to(dtype=torch.int64)
        target["boxes"] = tv_tensors.BoundingBoxes(target["boxes"], format="XYXY", canvas
        target["keypoints"] = torch.Tensor(np.array(target["keypoints"])).to(dtype=torch.
        target["image_id"] = torch.tensor([idx])
        target["area"] = (target["boxes"][:, 3] - target["boxes"][:, 1]) * (target["boxes
        target["iscrowd"] = torch.zeros(len(target["boxes"]), dtype=torch.int64)
#         # Apply transformations if available
        if self.transform is not None:
            image, target = self.transform(image,target)
#         bboxLabelsTarget
```

```
        return image, target


from torchvision.transforms import v2 as T
from google.colab import drive
import os

drive.mount('/content/drive')
os.system("wget https://raw.githubusercontent.com/pytorch/vision/main/references/detectio
os.system("wget https://raw.githubusercontent.com/pytorch/vision/main/references/detectio
os.system("wget https://raw.githubusercontent.com/pytorch/vision/main/references/detectio
os.system("wget https://raw.githubusercontent.com/pytorch/vision/main/references/detectio
os.system("wget https://raw.githubusercontent.com/pytorch/vision/main/references/detectio


def get_transform(train):
    transforms = []
    if train:
        transforms.append(T.RandomHorizontalFlip(0.5))
    transforms.append(T.ToDtype(torch.float, scale=True))
    transforms.append(T.ToPureTensor())
    return T.Compose(transforms)
```

```
 Mounted at /content/drive
```

```
from typing import Dict, List, Optional, Tuple,Union
from torchvision.ops.poolers import LevelMapper,_filter_input,_setup_scales,_onnx_merge_l
from torchvision.utils import _log_api_usage_once
from torchvision.ops.roi_align import roi_align

def _convert_to_roi_format(boxes: List[Tensor]) -> Tensor:
    concat_boxes = torch.cat(boxes, dim=0)
    device, dtype = concat_boxes.device, concat_boxes.dtype
    ids = torch.cat(
        [torch.full_like(b[:, :1], i, dtype=dtype, layout=torch.strided, device=device) f
        dim=0,
    )
    rois = torch.cat([ids, concat_boxes], dim=1)
    return rois

def _multiscale_roi_align(
    x_filtered: List[Tensor],
    boxes: List[Tensor],
    output_size: List[int],
    sampling_ratio: int,
    scales: Optional[List[float]],
    mapper: Optional[LevelMapper],
) -> Tensor:
    """
    Args:
```

```
    x_filtered (List[Tensor]): List of input tensors.
    boxes (List[Tensor[N, 4]]): boxes to be used to perform the pooling operation, in
        (x1, y1, x2, y2) format and in the image reference size, not the feature map
        reference. The coordinate must satisfy ``0 <= x1 < x2`` and ``0 <= y1 < y2``.
    output_size (Union[List[Tuple[int, int]], List[int]]): size of the output
    sampling_ratio (int): sampling ratio for ROIAlign
    scales (Optional[List[float]]): If None, scales will be automatically inferred. D
    mapper (Optional[LevelMapper]): If none, mapper will be automatically inferred. D
Returns:
    result (Tensor)
"""
if scales is None or mapper is None:
    raise ValueError("scales and mapper should not be None")

num_levels = len(x_filtered)
rois = _convert_to_roi_format(boxes)


if num_levels == 1:
    return roi_align(
        x_filtered[0],
        rois,
        output_size=output_size,
        spatial_scale=scales[0],
        sampling_ratio=sampling_ratio,
    )

levels = mapper(boxes)

num_rois = len(rois)


num_channels = x_filtered[0].shape[1]

dtype, device = x_filtered[0].dtype, x_filtered[0].device
result = torch.zeros(
    (
        num_rois,
        num_channels,
    )
    + output_size,
    dtype=dtype,
    device=device,
)
tracing_results = []
for level, (per_level_feature, scale) in enumerate(zip(x_filtered, scales)):
    idx_in_level = torch.where(levels == level)[0]
    rois_per_level = rois[idx_in_level]

    result_idx_in_level = roi_align(
        per_level_feature,
```

```
            rois_per_level,
            output_size=output_size,
            spatial_scale=scale,
            sampling_ratio=sampling_ratio,
        )

        if torchvision._is_tracing():
            tracing_results.append(result_idx_in_level.to(dtype))
        else:
            # result and result_idx_in_level's dtypes are based on dtypes of different
            # elements in x_filtered.  x_filtered contains tensors output by different
            # layers.  When autocast is active, it may choose different dtypes for
            # different layers' outputs.  Therefore, we defensively match result's dtype
            # before copying elements from result_idx_in_level in the following op.
            # We need to cast manually (can't rely on autocast to cast for us) because
            # the op acts on result in-place, and autocast only affects out-of-place ops.
            result[idx_in_level] = result_idx_in_level.to(result.dtype)

    if torchvision._is_tracing():
        result = _onnx_merge_levels(levels, tracing_results)

    return result


class MultiScaleRoIAlign(nn.Module):
    """
    Multi-scale RoIAlign pooling, which is useful for detection with or without FPN.

    It infers the scale of the pooling via the heuristics specified in eq. 1
    of the `Feature Pyramid Network paper <https://arxiv.org/abs/1612.03144>`_.
    They keyword-only parameters ``canonical_scale`` and ``canonical_level``
    correspond respectively to ``224`` and ``k0=4`` in eq. 1, and
    have the following meaning: ``canonical_level`` is the target level of the pyramid fr
    which to pool a region of interest with ``w x h = canonical_scale x canonical_scale``

    Args:
        featmap_names (List[str]): the names of the feature maps that will be used
            for the pooling.
        output_size (List[Tuple[int, int]] or List[int]): output size for the pooled regi
        sampling_ratio (int): sampling ratio for ROIAlign
        canonical_scale (int, optional): canonical_scale for LevelMapper
        canonical_level (int, optional): canonical_level for LevelMapper

    Examples::

        >>> m = torchvision.ops.MultiScaleRoIAlign(['feat1', 'feat3'], 3, 2)
        >>> i = OrderedDict()
        >>> i['feat1'] = torch.rand(1, 5, 64, 64)
        >>> i['feat2'] = torch.rand(1, 5, 32, 32)  # this feature won't be used in the po
        >>> i['feat3'] = torch.rand(1, 5, 16, 16)
```

```
    >>> # create some random bounding boxes
    >>> boxes = torch.rand(6, 4) * 256; boxes[:, 2:] += boxes[:, :2]
    >>> # original image size, before computing the feature maps
    >>> image_sizes = [(512, 512)]
    >>> output = m(i, [boxes], image_sizes)
    >>> print(output.shape)
    >>> torch.Size([6, 5, 3, 3])

"""

__annotations__ = {"scales": Optional[List[float]], "map_levels": Optional[LevelMappe

def __init__(
    self,
    featmap_names: List[str],
    output_size: Union[int, Tuple[int], List[int]],
    sampling_ratio: int,
    *,
    canonical_scale: int = 224,
    canonical_level: int = 4,
):
    super().__init__()
    _log_api_usage_once(self)
    if isinstance(output_size, int):
        output_size = (output_size, output_size)
    self.featmap_names = featmap_names
    self.sampling_ratio = sampling_ratio
    self.output_size = tuple(output_size)
    self.scales = None
    self.map_levels = None
    self.canonical_scale = canonical_scale
    self.canonical_level = canonical_level

def forward(
    self,
    x: Dict[str, Tensor],
    boxes: List[Tensor],
    image_shapes: List[Tuple[int, int]],
) -> Tensor:
    """
    Args:
        x (OrderedDict[Tensor]): feature maps for each level. They are assumed to hav
            all the same number of channels, but they can have different sizes.
        boxes (List[Tensor[N, 4]]): boxes to be used to perform the pooling operation
            (x1, y1, x2, y2) format and in the image reference size, not the feature
            reference. The coordinate must satisfy ``0 <= x1 < x2`` and ``0 <= y1 < y
        image_shapes (List[Tuple[height, width]]): the sizes of each image before the
            have been fed to a CNN to obtain feature maps. This allows us to infer th
            scale factor for each one of the levels to be pooled.
    Returns:
        result (Tensor)
```

```
        """
        x_filtered = _filter_input(x, self.featmap_names)
        if self.scales is None or self.map_levels is None:
            self.scales, self.map_levels = _setup_scales(
                x_filtered, image_shapes, self.canonical_scale, self.canonical_level
            )

        return _multiscale_roi_align(
            x_filtered,
            boxes,
            self.output_size,
            self.sampling_ratio,
            self.scales,
            self.map_levels,
        )

    def __repr__(self) -> str:
        return (
            f"{self.__class__.__name__}(featmap_names={self.featmap_names}, "
            f"output_size={self.output_size}, sampling_ratio={self.sampling_ratio})"
        )



class Balancer(nn.Module):
    def __init__(self):
        super().__init__()

        self.layer_1 = nn.Sequential(
            nn.Conv2d(1, 8, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1)),
            nn.ReLU(inplace=True),
            nn.MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
            nn.Conv2d(8, 8, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1)),
            nn.ReLU(inplace=True),
            nn.MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
            nn.Conv2d(8, 8, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1)),
            nn.ReLU(inplace=True),
            nn.MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
            nn.Conv2d(8, 8, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1)),
            nn.ReLU(inplace=True),
            nn.MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
        )

        # self.layer_2 = nn.Sequential(
        #     nn.Conv2d(64, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1)),
        #     nn.ReLU(inplace=True),
        #     nn.Conv2d(64, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1)),
        #     nn.ReLU(inplace=True),
        #     nn.Conv2d(32, 16, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1)),
        #     nn.ReLU(inplace=True),
        #     nn.MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=Fals
```

```
        #     nn.MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=Fals

        # )

    def forward(self,x):
        x = x.float()
        x = self.layer_1(x)
        # x = self.layer_2(x)
        return x



from drive.MyDrive.roi_heads import RoIHeads
from drive.MyDrive.roi_heads import fastrcnn_loss,maskrcnn_loss,maskrcnn_inference,keypoi
# from torchvision.models.detection.roi_heads import *
from torchvision.ops import boxes as box_ops
import matplotlib.pyplot as plt


def keypointrcnn_loss(keypoint_logits, proposals, gt_keypoints, keypoint_matched_idxs):
    # type: (Tensor, List[Tensor], List[Tensor], List[Tensor]) -> Tensor
    N, K, H, W = keypoint_logits.shape
    if H != W:
        raise ValueError(
            f"keypoint_logits height and width (last two elements of shape) should be equ
        )
    discretization_size = H
    heatmaps = []
    valid = []
    for proposals_per_image, gt_kp_in_image, midx in zip(proposals, gt_keypoints, keypoin
        kp = gt_kp_in_image[midx]
        heatmaps_per_image, valid_per_image = keypoints_to_heatmap(kp, proposals_per_imag
        # print(valid_per_image,len(valid_per_image))
        # print(proposals_per_image,"\n\n")
        heatmaps.append(heatmaps_per_image.view(-1))
        valid.append(valid_per_image.view(-1))

    keypoint_targets = torch.cat(heatmaps, dim=0)
    valid = torch.cat(valid, dim=0).to(dtype=torch.uint8)
    valid = torch.where(valid)[0]

    # torch.mean (in binary_cross_entropy_with_logits) doesn't
    # accept empty tensors, so handle it sepaartely
    if keypoint_targets.numel() == 0 or len(valid) == 0:
        return keypoint_logits.sum() * 0

    keypoint_logits = keypoint_logits.reshape(N * K, H * W)
    # print(keypoint_logits.shape)
    keypoint_loss = FF.cross_entropy(keypoint_logits[valid], keypoint_targets[valid])
    # print(torch.argmax(keypoint_logits[valid],axis=1))
    return keypoint_loss
```

```
        return keypoint_loss

    class CustomRoiHead(RoIHeads):
        # def calculate_distances(self,point1, point2, grid_points):
        #     x1, y1 = point1
        #     x2, y2 = point2
        #     x_grid, y_grid = grid_points
        #     distance1 = torch.sqrt((x_grid - x1)**2 + (y_grid - y1)**2).to(device)
        #     distance2 = torch.sqrt((x_grid - x2)**2 + (y_grid - y2)**2).to(device)
        #     return distance1, distance2

        # def create_feature_map(self,point1, point2, resolution=(10, 10)):
        #     # Generate grid of points
        #     x_vals = torch.linspace(min(point1[0], point2[0]), max(point1[0], point2[0]), r
        #     y_vals = torch.linspace(min(point1[1], point2[1]), max(point1[1], point2[1]), r
        #     X, Y = torch.meshgrid(x_vals, y_vals)

        #     # Calculate distances of each point from both given points
        #     distances1, distances2 = self.calculate_distances(point1, point2, (X, Y))

        #     # Normalize distances based on the difference between them
        #     total_distances = distances1 + distances2
        #     normalized_distances = distances1 / total_distances

        #     return normalized_distances

        # def guidingFeatureMapProducer(self,point1,point2,resolution):
        #     # Create feature map
        #     feature_map = self.create_feature_map(point1, point2, resolution=resolution)
        #     feature_map[feature_map<feature_map[int(point1[1])][int(point1[0])]] = 0
        #     feature_map = feature_map*5
        #     minExcludingZeros = feature_map[feature_map>0].min()
        #     print()
        #     feature_map-=minExcludingZeros
        #     feature_map[feature_map<0] = 0
        #     # maxPixel = torch.max(feature_map[feature_map>0]).to(device)
        #     # feature_map[feature_map>0]=maxPixel-feature_map[feature_map>0]+torch.min(feat

        #     # Plot heatmap
        #     plt.imshow(feature_map.cpu(), cmap='gray')
        #     plt.scatter(point1[1].cpu(), point1[0].cpu(), color='blue')  # Plot points
        #     plt.scatter(point2[1].cpu(), point2[0].cpu(), color='red')  # Plot points


        #     plt.title('Feature Map Heatmap')
        #     plt.show()
        #     return feature_map


        def calculate_distances(self,point1, point2, grid_points):
```

```
        x1, y1 = point1
        x2, y2 = point2
        x_grid, y_grid = grid_points
        distance1 = np.sqrt((x_grid - x1)**2 + (y_grid - y1)**2)
        distance2 = np.sqrt((x_grid - x2)**2 + (y_grid - y2)**2)
        return distance1, distance2

    def create_feature_map(self,point1, point2, resolution=(10, 10)):

        # Generate grid of points
        x_vals = np.linspace(min(point1[0], point2[0]), max(point1[0], point2[0]), resolu
        y_vals = np.linspace(min(point1[1], point2[1]), max(point1[1], point2[1]), resolu
        X, Y = np.meshgrid(x_vals, y_vals)

        # Calculate distances of each point from both given points
        distances1, distances2 = self.calculate_distances(point1, point2, (X, Y))

        # Normalize distances based on the difference between them
        total_distances = distances1 + distances2
        normalized_distances = distances1 / total_distances

        return normalized_distances

    def guidingFeatureMapProducer(self,point1,point2,point3,offsetToAdd,resolution,show):
        point1 = point1.cpu().numpy()
        point2 = point2.cpu().numpy()
        point3 = point3.cpu().numpy()
        # Create feature map
        feature_map = self.create_feature_map(point1, point2, resolution=resolution)
        feature_map[feature_map<feature_map[int(point2[1])][int(point2[0])]]=0
        feature_map = feature_map*5
        if len(feature_map[feature_map>0])>0:
          minExcludingZeros = feature_map[feature_map>0].min()
        else:
          minExcludingZeros = 0
        feature_map-=minExcludingZeros
        feature_map[feature_map<0]= 0

        mn = feature_map[feature_map>0].mean()/2
        if len(feature_map[feature_map>0])>0:
            maxPixel = np.max(feature_map[feature_map>0])
            feature_map[feature_map>0]=maxPixel-feature_map[feature_map>0]+np.min(feature

        if len(feature_map[feature_map!=0])>0:
          feature_map[feature_map!=0] = 1


        if point3[0]-offsetToAdd<0:
            point3[0] = offsetToAdd
        if point3[1]-offsetToAdd<0:
            point3[1]=offsetToAdd
```

```
    point3[1]=offsetToAdd

if point3[0]+offsetToAdd>resolution[0]-1:
    point3[0]=resolution[0]-1
if point3[1]-offsetToAdd>resolution[1]-1:
    point3[1]=resolution[0]-1

topleftX = point3[0]-offsetToAdd
topleftY = point3[1]-offsetToAdd
flag = True

if point3[0]-1<0:
  point3[0]=1

if point3[0]+1>resolution[0]-1:
  point3[0]=resolution[0]-2

if point3[1]-1<0:
  point3[1]=1
if point3[1]+1>resolution[1]-1:
  point3[1]=resolution[1]-2


if feature_map[int(point3[0]-1)][int(point3[1]-1)] == 0 or feature_map[int(point3
    flag = False


if flag:
    feature_map[True] = 0
for i in range(int(topleftX),int(topleftX+offsetToAdd)):
    for j in range(int(topleftY),int(topleftY+offsetToAdd)):
        if flag:
            feature_map[i][j] = 1



# if len(feature_map[feature_map==0])>0:
#   feature_map[feature_map==0] = 1

# feature_map[:,0:5]=0
# feature_map[:,-5:]=0
# feature_map[0:5,:]=0
# feature_map[-5:,:]=0
# zeromasks = feature_map==0
# nonzeromasks = feature_map==1

# if len(feature_map[zeromasks])>0:
#   feature_map[zeromasks]=1
# if len(feature_map[nonzeromasks])>0:
#   feature_map[nonzeromasks]=0
```

```
        # if show==1:
        #   # # Plot heatmap
        #   plt.imshow(feature_map, cmap='gray')
        #   plt.scatter(point1[0], point1[1], color='blue')  # Plot points
        #   plt.scatter(point2[0], point2[1], color='red')  # Plot points

        #   plt.title('Feature Map Heatmap')
        #   plt.show()
        #   print("\n\n")


        return torch.tensor(feature_map).to(device)



    def guidingOffsetProducer(self,point1,point2,initialBox,offsetToAdd,resolution,show):
        condition = 5
      # point1 = tip
      # point2 = back

      # x and y are for tip
        point1 = point1.cpu().numpy()
        point2 = point2.cpu().numpy()
        x = point1[0]
        y = point1[1]

        # xp and yp are for back
        xp = point2[0]
        yp = point2[1]

#xmin ymin xmax ymax
        # if x-xp>=condition and np.absolute(y-yp)<=condition:
        #   initialBox = [initialBox[0]-offsetToAdd,initialBox[1],x,initialBox[3]]
        # elif x-xp<=-1*condition and np.absolute(y-yp)<=condition:
        #   initialBox = [x,initialBox[1],initialBox[2]+offsetToAdd,initialBox[3]]
        # elif np.absolute(x-xp)<=condition and y-yp>=condition:
        #   initialBox = [initialBox[0],initialBox[1]-offsetToAdd,initialBox[2],y]
        # elif np.absolute(x-xp)<=condition and y-yp<=-1*condition:
        #   initialBox = [initialBox[0],y,initialBox[2],initialBox[3]+offsetToAdd]
        # elif x-xp<=-1*condition and y-yp<=-1*condition:
        #   initialBox = [x,y,initialBox[2]+offsetToAdd,initialBox[3]+offsetToAdd]
        # elif x-xp>=condition and y-yp>=condition:
        #   initialBox = [initialBox[0]-offsetToAdd,initialBox[1]-offsetToAdd,x,y]
        # elif x-xp<=-1*condition and y-yp>=condition:
        #   initialBox = [x,initialBox[1]-offsetToAdd,initialBox[2]+offsetToAdd,y]
        # elif x-xp>=condition and y-yp<=-1*condition:
```

```
    # elif x-xp>=condition and y-yp<=-1*condition:
    #   initialBox = [initialBox[0]-offsetToAdd,y,x,initialBox[3]+offsetToAdd]

    if x-xp>=condition and np.absolute(y-yp)<=condition:
      initialBox = [initialBox[0]-offsetToAdd,initialBox[1],initialBox[2],initialBox[
    elif x-xp<=-1*condition and np.absolute(y-yp)<=condition:
      initialBox = [initialBox[0],initialBox[1],initialBox[2]+offsetToAdd,initialBox[
    elif np.absolute(x-xp)<=condition and y-yp>=condition:
      initialBox = [initialBox[0],initialBox[1]-offsetToAdd,initialBox[2],initialBox[
    elif np.absolute(x-xp)<=condition and y-yp<=-1*condition:
      initialBox = [initialBox[0],initialBox[1],initialBox[2],initialBox[3]+offsetToA
    elif x-xp<=-1*condition and y-yp<=-1*condition:
      initialBox = [initialBox[0],initialBox[1],initialBox[2]+offsetToAdd,initialBox[
    elif x-xp>=condition and y-yp>=condition:
      initialBox = [initialBox[0]-offsetToAdd,initialBox[1]-offsetToAdd,initialBox[2]
    elif x-xp<=-1*condition and y-yp>=condition:
      initialBox = [initialBox[0],initialBox[1]-offsetToAdd,initialBox[2]+offsetToAdd
    elif x-xp>=condition and y-yp<=-1*condition:
      initialBox = [initialBox[0]-offsetToAdd,initialBox[1],initialBox[2],initialBox[
    return initialBox



def subsample_array(self,arr, chunk_sizes):
  subsamples = []
  start_index = 0
  for size in chunk_sizes:
      end_index = start_index + size
      subsamples.append(arr[start_index:end_index, :, :,:])
      start_index = end_index
  return subsamples


def postprocess_detections(
    self,
    class_logits,  # type: Tensor
    box_regression,  # type: Tensor
    proposals,  # type: List[Tensor]
    image_shapes,  # type: List[Tuple[int, int]]
):
    # type: (...) -> Tuple[List[Tensor], List[Tensor], List[Tensor]]
    device = class_logits.device
    num_classes = class_logits.shape[-1]

    boxes_per_image = [boxes_in_image.shape[0] for boxes_in_image in proposals]
    pred_boxes = self.box_coder.decode(box_regression, proposals)

    pred_scores = FF.softmax(class_logits, -1)

    pred_boxes_list = pred_boxes.split(boxes_per_image, 0)
    pred_scores_list = pred_scores.split(boxes_per_image, 0)
```

```
        all_boxes = []
        all_scores = []
        all_labels = []

        for boxes, scores, image_shape in zip(pred_boxes_list, pred_scores_list, image_sh
            boxes = box_ops.clip_boxes_to_image(boxes, image_shape)

            # create labels for each prediction
            labels = torch.arange(num_classes, device=device)
            labels = labels.view(1, -1).expand_as(scores)
            # remove predictions with the background label
            boxes = boxes[:, 1:]
            scores = scores[:, 1:]
            labels = labels[:, 1:]



            # batch everything, by making every class prediction be a separate instance
            boxes = boxes.reshape(-1, 4)
            scores = scores.reshape(-1)
            labels = labels.reshape(-1)


            # remove low scoring boxes
            inds = torch.where(scores > self.score_thresh)[0]
            boxes, scores, labels = boxes[inds], scores[inds], labels[inds]

            # remove empty boxes
            keep = box_ops.remove_small_boxes(boxes, min_size=1e-2)
            boxes, scores, labels = boxes[keep], scores[keep], labels[keep]

            # non-maximum suppression, independently done per class
            keep = box_ops.batched_nms(boxes, scores, labels, self.nms_thresh)
            # keep only topk scoring predictions
            keep = keep[: self.detections_per_img]
            boxes, scores, labels = boxes[keep], scores[keep], labels[keep]

            all_boxes.append(boxes)
            all_scores.append(scores)
            all_labels.append(labels)

        return all_boxes, all_scores, all_labels

    def forward(self, features, proposals, image_shapes,targets=None):
        # print(proposals,"\n")
        if targets is not None:
                    for t in targets:
                        # TODO: https://github.com/pytorch/pytorch/issues/26731
                        floating_point_types = (torch.float, torch.double, torch.half)
                        if not t["boxes"].dtype in floating_point_types:
```

```
                    if not t["boxes"].dtype in floating_point_types:
                        raise TypeError(f"target boxes must of float type, instead go
                    if not t["labels"].dtype == torch.int64:
                        raise TypeError(f"target labels must of int64 type, instead g
                    if self.has_keypoint():
                        if not t["keypoints"].dtype == torch.float32:
                            raise TypeError(f"target keypoints must of float type, in

        if self.training:
            proposals, matched_idxs, labels, regression_targets = self.select_training_sa
            # print(labels[0].shape,len(labels))
        else:
            labels = None
            regression_targets = None
            matched_idxs = None
        # print([t["boxes"] for t in targets],"\n\n")
        box_features = self.box_roi_pool(features, proposals, image_shapes)
        box_features = self.box_head(box_features)
        class_logits, box_regression = self.box_predictor(box_features)

        result: List[Dict[str, torch.Tensor]] = []
        losses = {}
        if self.training:
            if labels is None:
                raise ValueError("labels cannot be None")
            if regression_targets is None:
                raise ValueError("regression_targets cannot be None")
            loss_classifier, loss_box_reg = fastrcnn_loss(class_logits, box_regression, l
            losses = {"loss_classifier": loss_classifier, "loss_box_reg": loss_box_reg}
        else:
            boxes, scores, labels = self.postprocess_detections(class_logits, box_regress
            num_images = len(boxes)
            for i in range(num_images):
                result.append(
                    {
                        "boxes": boxes[i],
                        "labels": labels[i],
                        "scores": scores[i],
                    }
                )

        if self.has_mask():
            mask_proposals = [p["boxes"] for p in result]
            if self.training:
                if matched_idxs is None:
                    raise ValueError("if in training, matched_idxs should not be None")

                # during training, only focus on positive boxes
                num_images = len(proposals)
                mask_proposals = []
                pos_matched_idxs = []
```

```
            for img_id in range(num_images):
                pos = torch.where(labels[img_id] > 0)[0]
                mask_proposals.append(proposals[img_id][pos])
                pos_matched_idxs.append(matched_idxs[img_id][pos])
        else:
            pos_matched_idxs = None

        if self.mask_roi_pool is not None:
            mask_features = self.mask_roi_pool(features, mask_proposals, image_shapes
            mask_features = self.mask_head(mask_features)
            mask_logits = self.mask_predictor(mask_features)
        else:
            raise Exception("Expected mask_roi_pool to be not None")

        loss_mask = {}
        if self.training:
            if targets is None or pos_matched_idxs is None or mask_logits is None:
                raise ValueError("targets, pos_matched_idxs, mask_logits cannot be No

            gt_masks = [t["masks"] for t in targets]
            gt_labels = [t["labels"] for t in targets]
            rcnn_loss_mask = maskrcnn_loss(mask_logits, mask_proposals, gt_masks, gt_
            loss_mask = {"loss_mask": rcnn_loss_mask}
        else:
            labels = [r["labels"] for r in result]
            masks_probs = maskrcnn_inference(mask_logits, labels)
            for mask_prob, r in zip(masks_probs, result):
                r["masks"] = mask_prob

        losses.update(loss_mask)

    # keep none checks in if conditional so torchscript will conditionally
    # compile each branch
    if (
        self.keypoint_roi_pool is not None
        and self.keypoint_head is not None
        and self.keypoint_predictor is not None
    ):
        keypoint_proposals = [p["boxes"] for p in result]
        if self.training:
            # during training, only focus on positive boxes
            num_images = len(proposals)
            keypoint_proposals = []
            pos_matched_idxs = []
            if matched_idxs is None:
                raise ValueError("if in trainning, matched_idxs should not be None")

            for img_id in range(num_images):
                pos = torch.where(labels[img_id] > 0)[0]
                keypoint_proposals.append(proposals[img_id][pos])
                pos_matched_idxs.append(matched_idxs[img_id][pos])
```

```
        pos_matched_idxs.append(matched_idxs[img_id][pos])
else:
    pos_matched_idxs = None

offset = torch.tensor([-0,-0,0,0]).to(device)
keypoint_proposals_withNoOffest = []
for someTensor in keypoint_proposals:
    keypoint_proposals_withNoOffest.append(someTensor+offset)


keypoint_features = self.keypoint_roi_pool(features, keypoint_proposals_withN
keypoint_features_copied = keypoint_features
# print(keypoint_features_copied.shape)
keypoint_features = self.keypoint_head(keypoint_features)
keypoint_logits = self.keypoint_predictor(keypoint_features)
keypoint_logits_OldHead = keypoint_logits


offset = torch.tensor([-0,-0,0,0]).to(device)
keypoint_proposals_withOffest = []
for someTensor in keypoint_proposals:
    keypoint_proposals_withOffest.append(someTensor+offset)

# # print(keypoint_logits.shape,[i.shape for i in keypoint_proposals_withOffe
# chunk_sizes = [i.shape[0] for i in keypoint_proposals_withOffest]
# # Subsample the array into chunks

# subsamples = self.subsample_array(keypoint_logits, chunk_sizes)
# tensor_list = []
# for i in range(len(subsamples)):
#     keypoints_probs, kp_scores = keypointrcnn_inference(subsamples[i], [key
#     for j,counter in zip(keypoints_probs[0][:,0:3,0:2],range(len(keypoint_p
#         xmin,ymin,xmax,ymax = keypoint_proposals_withOffest[i][counter]
#         resizedBBxIntoFrame = keypoint_proposals_withOffest[i][counter]-tor
#         pickingPointXY = j[0,:]
#         backPointXY = j[1,:]
#         tipPointXY = j[2,:]
#         # scalingFactor1 = int(xmax-xmin)
#         # scalingFactor2 = int(ymax-ymin)
#         scalingFactor1=224
#         scalingFactor2=224
#         # print(scalingFactor1,scalingFactor2)
#         # print(backPointXY,tipPointXY)
#         backPointXYResizedIntoFrame = ((backPointXY-torch.tensor([xmin,ymin
#         tipPointXYResizedIntoFrame = ((tipPointXY-torch.tensor([xmin,ymin])
#         pickingPointXYResizedIntoFrame = ((pickingPointXY-torch.tensor([xmi

#         # print(backPointXYResizedIntoFrame,tipPointXYResizedIntoFrame,resi
#         backPointXYFloor = torch.floor(backPointXYResizedIntoFrame+torch.te
#         tipPointXYFloor = torch.floor(tipPointXYResizedIntoFrame+torch.tens
#         pickingPointXYFloor = torch.floor(pickingPointXYResizedIntoFrame+to
```

```
#         guideFeature = self.guidingFeatureMapProducer(tipPointXYFloor,backP
#         guideFeature = guideFeature.reshape(-1,scalingFactor1,scalingFactor
#         balancedGuide = self.guideBalancer(guideFeature)
#         tensor_list.append(torch.cat((keypoint_features_copied[i], balanced
#         # try:
#         #   self.guidingFeatureMapProducer(tipPointXYFloor,backPointXYFloor
#         # except:
#         #   pass


# guidedFeatureMap = torch.stack(tensor_list, dim=0)
# guidedFeatureMap = guidedFeatureMap.to(dtype=torch.float32)
# guidedFeatureMap = self.keypoint_head_guided(guidedFeatureMap)
# keypoint_logits = self.keypoint_predictor_guided(guidedFeatureMap)







# print(keypoint_logits.shape,[i.shape for i in keypoint_proposals_withOffest
chunk_sizes = [i.shape[0] for i in keypoint_proposals_withNoOffest]
# Subsample the array into chunks

subsamples = self.subsample_array(keypoint_logits_OldHead, chunk_sizes)
tensor_list = []
for i in range(len(subsamples)):
    keypoints_probs, kp_scores = keypointrcnn_inference(subsamples[i], [keypo
    for j,counter in zip(keypoints_probs[0][:,0:3,0:2],range(len(keypoint_pro
        # xmin,ymin,xmax,ymax = keypoint_proposals_withNoOffest[i][counter]
        # resizedBBxIntoFrame = keypoint_proposals_withNoOffest[i][counter]-t
        # pickingPointXY = j[0,:]
        backPointXY = j[1,:]
        tipPointXY = j[2,:]
        # scalingFactor1 = int(xmax-xmin)
        # scalingFactor2 = int(ymax-ymin)
        scalingFactor1=224
        scalingFactor2=224
#         # print(scalingFactor1,scalingFactor2)
#         # print(backPointXY,tipPointXY)
        # backPointXYResizedIntoFrame = ((backPointXY-torch.tensor([xmin,ymin
        # tipPointXYResizedIntoFrame = ((tipPointXY-torch.tensor([xmin,ymin])
        # pickingPointXYResizedIntoFrame = ((pickingPointXY-torch.tensor([xmi

#         # print(backPointXYResizedIntoFrame,tipPointXYResizedIntoFrame,resi
        # backPointXYFloor = torch.floor(backPointXYResizedIntoFrame+torch.te
        # tipPointXYFloor = torch.floor(tipPointXYResizedIntoFrame+torch.tens
#         pickingPointXYFloor = torch.floor(pickingPointXYResizedIntoFrame+to
        # print(keypoint_proposals_withNoOffest[i][counter])
```

```
            updatedBox = self.guidingOffsetProducer(tipPointXY,backPointXY,keypoi
            keypoint_proposals_withOffest[i][counter] = torch.tensor(updatedBox)
            # print(keypoint_proposals_withNoOffest[i][counter])
            # print(keypoint_proposals_withOffest[i][counter])
#           guideFeature = guideFeature.reshape(-1,scalingFactor1,scalingFactor
#           balancedGuide = self.guideBalancer(guideFeature)
#           tensor_list.append(torch.cat((keypoint_features_copied[i], balanced
#           # try:
#           #   self.guidingFeatureMapProducer(tipPointXYFloor,backPointXYFloor
#           # except:
#           #   pass


# guidedFeatureMap = torch.stack(tensor_list, dim=0)
# guidedFeatureMap = guidedFeatureMap.to(dtype=torch.float32)
# guidedFeatureMap = self.keypoint_head_guided(guidedFeatureMap)
# keypoint_logits = self.keypoint_predictor_guided(guidedFeatureMap)




keypoint_features_guided = self.keypoint_roi_pool_guided(features, keypoint_p
guidedFeatureMap = self.keypoint_head_guided(keypoint_features_guided)
keypoint_logits = self.keypoint_predictor_guided(guidedFeatureMap)


if self.training:
  print("-- -- -- -- -- --")
loss_keypoint = {}
if self.training:
    if targets is None or pos_matched_idxs is None:
        raise ValueError("both targets and pos_matched_idxs should not be Non

    gt_keypoints = [t["keypoints"] for t in targets]

    gt_keypoints_old = []
    for i in range(len(gt_keypoints)):
        gt_keypoints_old.append(gt_keypoints[i][:,1:,:])

    gt_keypoints_guided = []
    for i in range(len(gt_keypoints)):
        gt_keypoints_guided.append(gt_keypoints[i][:,0,:].reshape(-1,1,3))


    rcnn_loss_keypoint_old = keypointrcnn_loss(
        keypoint_logits_OldHead[:,1:], keypoint_proposals_withNoOffest, gt_ke
    )

    rcnn_loss_keypoint_guided = keypointrcnn_loss(
```

```
                    keypoint_logits[:,0].reshape(-1,1,56,56), keypoint_proposals_withOffe
                )

                # loss_keypoint = {"loss_keypoint": (rcnn_loss_keypoint_guided*1+rcnn_los
                loss_keypoint = {"loss_keypoint": rcnn_loss_keypoint_guided}



                # rcnn_loss_keypoint = keypointrcnn_loss(
                #     keypoint_logits, keypoint_proposals_withNoOffest, gt_keypoints, pos
                # )
                # loss_keypoint = {"loss_keypoint": rcnn_loss_keypoint}


            else:
                if keypoint_logits is None or keypoint_proposals_withNoOffest is None:
                    raise ValueError(
                        "both keypoint_logits and keypoint_proposals_withOffest should no
                    )

                # keypoints_probs, kp_scores = keypointrcnn_inference(keypoint_logits, ke
                # keypoints_probs, kp_scores = keypointrcnn_inference(torch.cat((keypoint



                keypoints_probs_OldHead, kp_scores_OldHead = keypointrcnn_inference(keypo
                keypoints_probs_GuidedHead, kp_scores_GuidedHead= keypointrcnn_inference(

                keypoints_probs = []
                keypoints_probs.append(torch.cat((keypoints_probs_GuidedHead[0][:,0,:].re

                kp_scores = []
                kp_scores.append(torch.cat((kp_scores_GuidedHead[0][:,0].reshape(-1,1), k







                for keypoint_prob, kps, r in zip(keypoints_probs, kp_scores, result):
                    r["keypoints"] = keypoint_prob
                    r["keypoints_scores"] = kps
            losses.update(loss_keypoint)

        return result, losses


import utils
# # Create a dataset instance and a data loader
dataset = CustomDataset(image_folder='/content/drive/MyDrive/StrawDI_Db1/train/img/', ann
```

```
data_loader = DataLoader(dataset, batch_size=8, shuffle=True,collate_fn=utils.collate_fn)



import matplotlib.pyplot as plt
out = next(iter(data_loader))


i = 0
image = out[0][i]
pred = out[1][i]
print(pred)
print()
import matplotlib.pyplot as plt
# !pip install PyGame
from torchvision.utils import draw_bounding_boxes, draw_segmentation_masks
from PIL import Image
import torchvision.transforms as transforms
# take picture
from torchvision.ops import nms

import cv2

transform = transforms.Compose([
    transforms.ToTensor(),
    # Add more transforms if needed
])


image = (255.0 * (image - image.min()) / (image.max() - image.min())).to(torch.uint8)

image = image[:3, ...]


pred_labels = [f"pedestrian: {0:.3f}" for label in pred["labels"]]
pred_boxes = (pred["boxes"]*(torch.Tensor([1,1,1,1])).to(torch.device('cpu'))).long()


finalBoxes = pred_boxes
print(pred["labels"])
print(finalBoxes)

finalLabels = np.array(pred_labels)

output_image = draw_bounding_boxes(image,finalBoxes , finalLabels, colors="red")

plt.figure(figsize=(10, 10))
plt.imshow(image.permute(1, 2, 0))

# masks = (pred["masks"] > 0.7).squeeze(1)
# output_image = draw_segmentation_masks(output_image, masks, alpha=0.5, colors="blue")
```

```
plt.figure(figsize=(10, 10))
plt.imshow(output_image.permute(1, 2, 0))
```

```
import gc

model.cpu()
gc.collect()
torch.cuda.empty_cache()
```

```
from torchvision.models.detection import KeypointRCNN
# backbone = resnet_fpn_backbone(backbone_name = "resnet18", pretrained = True)
# backbone = torchvision.models.mobilenet_v3_small(pretrained=True).features
# backbone[-1][0] = nn.Conv2d(in_channels=96, out_channels=256, kernel_size=1, stride=1,
# backbone[-1][1] = nn.BatchNorm2d(256,eps=0.001, momentum=0.01)


# for shufflenet_v2_x0_5
backbone = torchvision.models.shufflenet_v2_x0_5(pretrained=True)
backbone = nn.Sequential(*list(backbone.children())[:-1])
backbone[-1][0] = nn.Conv2d(in_channels=192, out_channels=256, kernel_size=1, stride=1, p
backbone[-1][1] = nn.BatchNorm2d(256)
backbone[-1][2] = nn.ReLU(inplace=True)


# for shufflenet_v2_x1_0
# backbone = torchvision.models.shufflenet_v2_x1_0(pretrained=True)
# backbone = nn.Sequential(*list(backbone.children())[:-1])
# backbone[-1][0] = nn.Conv2d(in_channels=464, out_channels=256, kernel_size=1, stride=1,
# backbone[-1][1] = nn.BatchNorm2d(256)
# backbone[-1][2] = nn.ReLU(inplace=True)
```

```
# model = torchvision.models.detection.keypointrcnn_resnet50_fpn(pretrained=False,
#                                                                 pretrained_backbone=
#                                                                 num_keypoints=5,num_
```

```
anchor_generator = AnchorGenerator(
        sizes=((128,256,512),),
        aspect_ratios=((0.5, 1.0, 2.0),),
`
```

```
)
backbone.out_channels = 256
model = KeypointRCNN(
      backbone,
      num_classes=3,
      num_keypoints=5,
      rpn_anchor_generator=anchor_generator,
      box_roi_pool=torchvision.ops.MultiScaleRoIAlign(featmap_names=["0"], output_size=7,
      keypoint_roi_pool=torchvision.ops.MultiScaleRoIAlign(featmap_names=["0"], output_si
  )
```

```
# get number of input features for the classifier
in_features = model.roi_heads.box_predictor.cls_score.in_features
# replace the pre-trained head with a new one
model.roi_heads.box_predictor = FastRCNNPredictor(in_features,3)
```

```
backbone
```

```
     Sequential(
       (0): Sequential(
         (0): Conv2d(3, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
     bias=False)
         (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
     track_running_stats=True)
         (2): ReLU(inplace=True)
       )
       (1): MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1, ceil_mode=False)
       (2): Sequential(
         (0): InvertedResidual(
           (branch1): Sequential(
             (0): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
     groups=24, bias=False)
             (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
     track_running_stats=True)
             (2): Conv2d(24, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
             (3): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
     track_running_stats=True)
             (4): ReLU(inplace=True)
           )
           (branch2): Sequential(
             (0): Conv2d(24, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
             (1): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
     track_running_stats=True)
             (2): ReLU(inplace=True)
             (3): Conv2d(58, 58, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
     groups=58, bias=False)
             (4): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
     track_running_stats=True)
             (5): Conv2d(58, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
```

```
        (6): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
        (7): ReLU(inplace=True)
      )
    )
    (1): InvertedResidual(
      (branch1): Sequential()
      (branch2): Sequential(
        (0): Conv2d(58, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (1): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
        (2): ReLU(inplace=True)
        (3): Conv2d(58, 58, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
groups=58, bias=False)
        (4): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
        (5): Conv2d(58, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (6): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
        (7): ReLU(inplace=True)
      )
    )
    (2): InvertedResidual(
      (branch1): Sequential()
      (branch2): Sequential(
        (0): Conv2d(58, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (1): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
```

```
sum(p.numel() for p in backbone.parameters())
```

```
895972
```

```
model
```

```
KeypointRCNN(
  (transform): GeneralizedRCNNTransform(
      Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
      Resize(min_size=(640, 672, 704, 736, 768, 800), max_size=1333,
mode='bilinear')
  )
  (backbone): Sequential(
    (0): Sequential(
      (0): Conv2d(3, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
bias=False)
      (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (2): ReLU(inplace=True)
    )
    (1): MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1, ceil_mode=False)
    (2): Sequential(
      (0): InvertedResidual(
        (branch1): Sequential(
          (0): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
```

```
          (0): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
    groups=24, bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
          (2): Conv2d(24, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (3): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
          (4): ReLU(inplace=True)
        )
        (branch2): Sequential(
          (0): Conv2d(24, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (1): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
          (2): ReLU(inplace=True)
          (3): Conv2d(58, 58, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
    groups=58, bias=False)
          (4): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
          (5): Conv2d(58, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (6): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
          (7): ReLU(inplace=True)
        )
      )
      (1): InvertedResidual(
        (branch1): Sequential()
        (branch2): Sequential(
          (0): Conv2d(58, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (1): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
          (2): ReLU(inplace=True)
          (3): Conv2d(58, 58, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
    groups=58, bias=False)
          (4): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
          (5): Conv2d(58, 58, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (6): BatchNorm2d(58, eps=1e-05, momentum=0.1, affine=True,
    track_running_stats=True)
          (7): ReLU(inplace=True)
        )
```

```
# model.backbone = BackboneWithFPN(backbone,return_layers={"3":"0","6":"1","8":"2","12":"
```

```
# model.backbone = BackboneWithFPN(backbone,return_layers={"5":"0"},in_channels_list=[102
```

```
# model.backbone = backbone
```

```
model
```

```
    KeypointRCNN(
      (transform): GeneralizedRCNNTransform(
          Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
          Resize(min_size=(640, 672, 704, 736, 768, 800), max_size=1333,
```

```
mode='bilinear')
  )
  (backbone): BackboneWithFPN(
    (body): IntermediateLayerGetter(
      (0): Conv2dNormActivation(
        (0): Conv2d(3, 16, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
bias=False)
        (1): BatchNorm2d(16, eps=0.001, momentum=0.01, affine=True,
track_running_stats=True)
        (2): Hardswish()
      )
      (1): InvertedResidual(
        (block): Sequential(
          (0): Conv2dNormActivation(
            (0): Conv2d(16, 16, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
groups=16, bias=False)
            (1): BatchNorm2d(16, eps=0.001, momentum=0.01, affine=True,
track_running_stats=True)
            (2): ReLU(inplace=True)
          )
          (1): SqueezeExcitation(
            (avgpool): AdaptiveAvgPool2d(output_size=1)
            (fc1): Conv2d(16, 8, kernel_size=(1, 1), stride=(1, 1))
            (fc2): Conv2d(8, 16, kernel_size=(1, 1), stride=(1, 1))
            (activation): ReLU()
            (scale_activation): Hardsigmoid()
          )
          (2): Conv2dNormActivation(
            (0): Conv2d(16, 16, kernel_size=(1, 1), stride=(1, 1), bias=False)
            (1): BatchNorm2d(16, eps=0.001, momentum=0.01, affine=True,
track_running_stats=True)
          )
        )
      )
      (2): InvertedResidual(
        (block): Sequential(
          (0): Conv2dNormActivation(
            (0): Conv2d(16, 72, kernel_size=(1, 1), stride=(1, 1), bias=False)
            (1): BatchNorm2d(72, eps=0.001, momentum=0.01, affine=True,
track_running_stats=True)
            (2): ReLU(inplace=True)
          )
          (1): Conv2dNormActivation(
            (0): Conv2d(72, 72, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
groups=72, bias=False)
            (1): BatchNorm2d(72, eps=0.001, momentum=0.01, affine=True,
track_running_stats=True)
            (2): ReLU(inplace=True)
          )
          (2): Conv2dNormActivation(
            (0): Conv2d(72, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
            (1): BatchNorm2d(24, eps=0.001, momentum=0.01, affine=True,
track_running_stats=True)
          )
```

```
# model.rpn.head = RPNHead(256,3)
# anchor_generator = AnchorGenerator(
#         sizes=((128,256,512),),
#         aspect_ratios=((0.5, 1.0, 2.0),),
# )
# model.rpn.anchor_generator = anchor_generator


# for param in model.parameters():
#     param.requires_grad = False


# box_roi_pool = torchvision.ops.MultiScaleRoIAlign(featmap_names=["0", "1", "2", "3"], o
box_roi_pool = torchvision.ops.MultiScaleRoIAlign(featmap_names=["0"], output_size=7, sam

resolution = box_roi_pool.output_size[0]
representation_size = 1024
# box_head = TwoMLPHead(256 * resolution**2, representation_size)
box_head = TwoMLPHead(256 * resolution**2, representation_size)

representation_size = 1024
box_predictor = FastRCNNPredictor(representation_size, 3)

model.roi_heads = CustomRoiHead(
            # Box
            box_roi_pool,
            box_head,
            box_predictor,
            0.5,
            0.5,
            512,
            0.25,
            None,
            0.05,
            0.5,
            100,
        )

# model.roi_heads.keypoint_roi_pool = MultiScaleRoIAlign(featmap_names=["0", "1", "2", "3
model.roi_heads.keypoint_roi_pool = MultiScaleRoIAlign(featmap_names=["0"], output_size=1




keypoint_layers = tuple(512 for _ in range(2))
# out_channels = backbone.out_channels
out_channels=256

model.roi_heads.keypoint_head = KeypointRCNNHeads(out_channels, keypoint_layers)
```

```
keypoint_dim_reduced = 512
model.roi_heads.keypoint_predictor = KeypointRCNNPredictor(keypoint_dim_reduced, 5)


for param in model.parameters():
    param.requires_grad = False



model.roi_heads.keypoint_roi_pool_guided = MultiScaleRoIAlign(featmap_names=["0"], output
# model.roi_heads.guideBalancer = Balancer()
model.roi_heads.keypoint_head_guided = KeypointRCNNHeads(out_channels, keypoint_layers)
keypoint_dim_reduced = 512
model.roi_heads.keypoint_predictor_guided = KeypointRCNNPredictor(keypoint_dim_reduced, 5

# # train on the GPU or on the CPU, if a GPU is not available
device = torch.device('cuda') if torch.cuda.is_available() else torch.device('cpu')

model.to(device)
```

```
KeypointRCNN(
  (transform): GeneralizedRCNNTransform(
      Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
      Resize(min_size=(640, 672, 704, 736, 768, 800), max_size=1333,
mode='bilinear')
  )
  (backbone): Sequential(
    (0): Sequential(
      (0): Conv2d(3, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
bias=False)
      (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (2): ReLU(inplace=True)
    )
    (1): MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1, ceil_mode=False)
    (2): Sequential(
      (0): InvertedResidual(
        (branch1): Sequential(
          (0): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
groups=24, bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (2): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (3): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (4): ReLU(inplace=True)
        )
        (branch2): Sequential(
          (0): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (2): ReLU(inplace=True)
          (3): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
groups=24, bias=False)
```

```
      groups=24, bias=False)
                (4): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
                (5): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
                (6): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
                (7): ReLU(inplace=True)
              )
            )
            (1): InvertedResidual(
              (branch1): Sequential()
              (branch2): Sequential(
                (0): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
                (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
                (2): ReLU(inplace=True)
                (3): Conv2d(24, 24, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
      groups=24, bias=False)
                (4): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
                (5): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
                (6): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
      track_running_stats=True)
                (7): ReLU(inplace=True)
              )
```

```
sum(p.numel() for p in model.parameters())
```

```
21867622
```

```
for param in model.parameters():
    print(param.requires_grad)
```

```
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
```

```
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
False
```

```
# anchor_generator = AnchorGenerator(
#         sizes=((32,64,128),(32,64,128),(32,64,128),(32,64,128),),
#         aspect_ratios=((0.5, 1.0, 2.0),(0.5, 1.0, 2.0),(0.5, 1.0, 2.0),(0.5, 1.0, 2.0),
# )
# model.rpn.anchor_generator = anchor_generator
# model.roi_heads.box_roi_pool = torchvision.ops.MultiScaleRoIAlign(
#     featmap_names=['0'],
#     output_size=7,
#     sampling_ratio=2
# )


from engine import train_one_epoch, evaluate
import utils

num_classes = 3
```

```
_
# use our dataset and defined transformations

dataset = CustomDataset(image_folder='/content/drive/MyDrive/StrawDI_Db1/train/img/', ann
dataset_test = CustomDataset(image_folder='/content/drive/MyDrive/StrawDI_Db1/test/img/',


dataset = CustomDataset(image_folder='/content/drive/MyDrive/StrawDI_Db1/val/img/', annot


import math
import sys
import time
def train_one_epoch(model, optimizer, data_loader, device, epoch,print_freq,scheduler=Non
    model.train()
    metric_logger = utils.MetricLogger(delimiter="  ")
    metric_logger.add_meter("lr", utils.SmoothedValue(window_size=1, fmt="{value:.6f}"))
    header = f"Epoch: [{epoch}]"

    lr_scheduler = None
    # if epoch == 0:
    #     warmup_factor = 1.0 / 1000
    #     warmup_iters = min(1000, len(data_loader) - 1)
    #     lr_scheduler = torch.optim.lr_scheduler.LinearLR(
    #         optimizer, start_factor=warmup_factor, total_iters=warmup_iters
    #     )
    #     # lr_scheduler = scheduler


    for images, targets in metric_logger.log_every(data_loader, print_freq, header):
        images = list(image.to(device) for image in images)
        targets = [{k: v.to(device) if isinstance(v, torch.Tensor) else v for k, v in t.i
        with torch.cuda.amp.autocast(enabled=scaler is not None):
            loss_dict = model(images, targets)
            losses = sum(loss for loss in loss_dict.values())

        # reduce losses over all GPUs for logging purposes
        loss_dict_reduced = utils.reduce_dict(loss_dict)
        losses_reduced = sum(loss for loss in loss_dict_reduced.values())

        loss_value = losses_reduced.item()
        if not math.isfinite(loss_value):
            print(f"Loss is {loss_value}, stopping training")
            print(loss_dict_reduced)
            sys.exit(1)

        optimizer.zero_grad()
        if scaler is not None:
            scaler.scale(losses).backward()
            scaler.step(optimizer)
            scaler.update()
        else:
```

```
            losses.backward()
            optimizer.step()

        if lr_scheduler is not None:
            lr_scheduler.step()

        metric_logger.update(loss=losses_reduced, **loss_dict_reduced)
        metric_logger.update(lr=optimizer.param_groups[0]["lr"])

    return metric_logger


from torch.optim.lr_scheduler import StepLR
# define training and validation data loaders

data_loader = torch.utils.data.DataLoader(
    dataset,
    batch_size=12,
    shuffle=True,
    num_workers=4,
    collate_fn=utils.collate_fn
)

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

# get the model using our helper function
# model = get_model_instance_segmentation(num_classes)

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

# # and a learning rate scheduler
# lr_scheduler = torch.optim.lr_scheduler.StepLR(
#     optimizer,
#     step_size=3,
#     gamma=0.1
# )

     KeypointRCNN(
       (transform): GeneralizedRCNNTransform(
           Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
           Resize(min_size=(640, 672, 704, 736, 768, 800), max_size=1333,
     mode='bilinear')
       )
       (backbone): Sequential(
         (0): Sequential(
```

```
    (0): Sequential(
      (0): Conv2d(3, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
bias=False)
      (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (2): ReLU(inplace=True)
    )
    (1): MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1, ceil_mode=False)
    (2): Sequential(
      (0): InvertedResidual(
        (branch1): Sequential(
          (0): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
groups=24, bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (2): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (3): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (4): ReLU(inplace=True)
        )
        (branch2): Sequential(
          (0): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (2): ReLU(inplace=True)
          (3): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
groups=24, bias=False)
          (4): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (5): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (6): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (7): ReLU(inplace=True)
        )
      )
      (1): InvertedResidual(
        (branch1): Sequential()
        (branch2): Sequential(
          (0): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (2): ReLU(inplace=True)
          (3): Conv2d(24, 24, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
groups=24, bias=False)
          (4): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (5): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (6): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (7): ReLU(inplace=True)
        )
```

```
# i = 0
# image = out[0][i]
import matplotlib.pyplot as plt
```

```
# !pip install PyGame
from torchvision.utils import draw_bounding_boxes, draw_segmentation_masks
from PIL import Image
import torchvision.transforms as transforms
# take picture
from torchvision.ops import nms

import cv2


def box_iou(box1, box2):

    # Calculate intersection area
    x1 = torch.max(box1[:, None, 0], box2[:, 0])
    y1 = torch.max(box1[:, None, 1], box2[:, 1])
    x2 = torch.min(box1[:, None, 2], box2[:, 2])
    y2 = torch.min(box1[:, None, 3], box2[:, 3])

    inter = (x2 - x1).clamp(0) * (y2 - y1).clamp(0)

    # Calculate area of both boxes
    area1 = (box1[:, 2] - box1[:, 0]) * (box1[:, 3] - box1[:, 1])
    area2 = (box2[:, 2] - box2[:, 0]) * (box2[:, 3] - box2[:, 1])

    # Calculate union area
    union = area1[:, None] + area2 - inter

    # Compute IoU
    iou = inter / union
    return iou

def testFN(batchInxShowStart=0,batchInxShowEnd=1,ShowTestImage=True,ShowLog=True):
    print("Test Started")
    testSetTruePreds = 0
    acceptablePredictionsInDataset = 0.00001
    sumOfPxlwiseError = 0

    # try:
    for batchInx,(image,label) in enumerate(data_loader_test):
        image = image[0]
        transform = transforms.Compose([
            transforms.ToTensor(),
            # Add more transforms if needed
        ])

        eval_transform = get_transform(train=False)

        model.eval()
        with torch.no_grad():
            x = eval_transform(image)
            # convert RGBA -> RGB and move to device
```

```
    # convert RGBA -> RGB and move to device
    x = x[:3, ...].to(device)
    predictions = model([x, ])
    pred = predictions[0]


image = (255.0 * (image - image.min()) / (image.max() - image.min())).to(torch.ui

image = image[:3, ...]




scoreMask = pred["scores"]>0.5

pred_labels = [f"pedestrian: {score:.3f}" for label, score in zip(pred["labels"][
pred_boxes = (pred["boxes"][scoreMask]*(torch.Tensor([1,1,1,1])).to(device)).long


keep = nms(pred_boxes.float(), pred["scores"][scoreMask], iou_threshold=0.5)

finalBoxes = pred_boxes[keep]+torch.Tensor([-0,-0,0,0]).to(device)

# print(batchInx)
# print(finalBoxes)
# print(label[0]["boxes"])

iou_threshold = 0.5
iou_matrix = box_iou(finalBoxes.cpu(),label[0]["boxes"])
masked_iou_matrix = torch.where(iou_matrix >= iou_threshold, iou_matrix, torch.te

try:
    label2PredsBoxMatching = [int(torch.argmax(col)) if torch.max(col) != float('
except:
  continue
# print(label2PredsBoxMatching)


try:
  finalLabels = np.array(pred_labels)[np.array(keep.tolist())]
except:
  continue
# print(finalBoxes[:,2]-finalBoxes[:,0],finalBoxes[:,3]-finalBoxes[:,1])
# print(finalBoxes[:,0],"\n",finalBoxes[:,1],"\n",finalBoxes[:,2],"\n",finalBoxes
output_image = draw_bounding_boxes(image,finalBoxes , finalLabels, colors="red")

keypointsToPlot = np.array([point[:2].cpu() for keypoints in predictions[0]["keyp
# print(keypointsToPlot.reshape(-1,5,2)[:,0])
# plt.figure(figsize=(10, 10))
# plt.imshow(output_image.permute(1, 2, 0))
```

```
# evaluator
    allPredictedKeypoints = keypointsToPlot.reshape(-1,5,2)[:,0]
    allLabelKeypoints = label[0]["keypoints"][:,0,:2]
    perImagelabelkeycounter = 0
    perImagetruePreds = 0
    flag = 0
    for labelPoint in enumerate(allLabelKeypoints):
      if label2PredsBoxMatching[labelPoint[0]] != float('-inf'):
        if labelPoint[1][0]!=0 or labelPoint[1][1]!=0:
          KeypointPredictedMatched = allPredictedKeypoints[label2PredsBoxMatching[lab
          pxlwiseDistanceForStraberry = np.sqrt((KeypointPredictedMatched[0]-labelPoi
          sumOfPxlwiseError+=pxlwiseDistanceForStraberry
          acceptablePredictionsInDataset+=1
          if pxlwiseDistanceForStraberry < 25:
              testSetTruePreds+=1
          else:
            flag=1
            continue
          if ShowLog:
            print(pxlwiseDistanceForStraberry)


    # batchInx>=55 and batchInx<=65
    # batchInx>=65 and batchInx<=75
    # batchInx>=75 and batchInx<=85
    # batchInx>=85 and batchInx<=95
    # and batchInx>=5 and batchInx<=7
    if(flag==1 and batchInx>= batchInxShowStart and batchInx<batchInxShowEnd and Show
        plt.figure(figsize=(10, 10))
        plt.imshow(output_image.permute(1, 2, 0))
        plt.scatter(keypointsToPlot.reshape(-1,5,2)[:,3:, 0], keypointsToPlot.reshape
        plt.scatter(keypointsToPlot.reshape(-1,5,2)[:,0, 0], keypointsToPlot.reshape(
        plt.scatter(keypointsToPlot.reshape(-1,5,2)[:,1, 0], keypointsToPlot.reshape(
        plt.scatter(keypointsToPlot.reshape(-1,5,2)[:,2, 0], keypointsToPlot.reshape(
        plt.scatter(label[0]["keypoints"][:,0,0], label[0]["keypoints"][:,0,1], s=10,
        # print(batchInx)



# evaluator end

    # print(allPredictedKeypoints)
    # print(allLabelKeypoints)

    # plt.scatter(keypointsToPlot[:, 0], keypointsToPlot[:, 1], s=10, c='r', marker='
    # plt.scatter(label[0]["keypoints"][:,0,0], label[0]["keypoints"][:,0,1], s=10, c
    # break
    if batchInx>142:
      break
# except:
print("Test Ended")
```

```
    print( Test Ended )
    return testSetTruePreds,acceptablePredictionsInDataset,sumOfPxlwiseError



# construct an optimizer
params = [p for p in model.parameters() if p.requires_grad]
# Set up the Adam optimizer
optimizer = torch.optim.Adam(
    params,
    lr=0.0002,
    weight_decay=0.0001
)
scheduler = StepLR(optimizer, step_size=1, gamma=0.98)


# let's train it for 5 epochs
num_epochs = 20
averagePxlError = True
listOfAveragePxlErrors = []
listOfAccuracy = []
try:
  for epoch in range(num_epochs):


        train_one_epoch(model, optimizer, data_loader, device, epoch,print_freq=5)
        testSetTruePreds,acceptablePredictionsInDataset,sumOfPxlwiseError = testFN(0,1,Fa
        print(testSetTruePreds/acceptablePredictionsInDataset)
        listOfAveragePxlErrors.append(sumOfPxlwiseError/acceptablePredictionsInDataset)
        listOfAccuracy.append(testSetTruePreds/acceptablePredictionsInDataset)
        print("This Epochs AveragePlxError Test",sumOfPxlwiseError/acceptablePredictionsI

        if averagePxlError==True:
          averagePxlError = sumOfPxlwiseError/acceptablePredictionsInDataset
        else:
          if (sumOfPxlwiseError/acceptablePredictionsInDataset)<averagePxlError:
            averagePxlError = sumOfPxlwiseError/acceptablePredictionsInDataset
            torch.save(model.state_dict(), "/content/drive/MyDrive/shufflenet_v2_x0_5NotG

        print("Best Epochs AveragePlxError Test",averagePxlError)
        print(sumOfPxlwiseError,acceptablePredictionsInDataset)

        scheduler.step()

        #   # evaluate on the test dataset
        #   evaluate(model, data_loader_test, device=device)
except:
  print("That's it!")
finally:
  plt.figure(figsize=(10, 6))
  plt.plot(range(len(listOfAveragePxlErrors)),listOfAveragePxlErrors, marker='o', linesty
  plt.xlabel('Epoch')
```

```
plt.xlabel('Epoch')
plt.ylabel('Average Pixel Error')
plt.title('Average Pixel Error per Epoch')
plt.grid(True)
plt.show()


plt.figure(figsize=(10, 6))
plt.plot(range(len(listOfAccuracy)),listOfAccuracy, marker='o', linestyle='-', color='b
plt.xlabel('Epoch')
plt.ylabel('Accuracy')
plt.title('Accuracy per Epoch')
plt.grid(True)
plt.show()
print(listOfAveragePxlErrors)
print(listOfAccuracy)
```

```
<ipython-input-16-865744b71ab2>:23: FutureWarning: `torch.cuda.amp.autocast(args...)`
  with torch.cuda.amp.autocast(enabled=scaler is not None):
<ipython-input-21-3ac5d076514b>:546: UserWarning: To copy construct from a tensor, it
  keypoint_proposals_withOffest[i][counter] = torch.tensor(updatedBox)
Streaming output truncated to the last 5000 lines.
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [170/234]  eta: 0:04:04  lr: 0.000192  loss: 6.0692 (6.1491)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [175/234]  eta: 0:03:43  lr: 0.000192  loss: 6.1642 (6.1552)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [180/234]  eta: 0:03:30  lr: 0.000192  loss: 6.2644 (6.1571)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [185/234]  eta: 0:03:09  lr: 0.000192  loss: 6.1445 (6.1508)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [190/234]  eta: 0:02:49  lr: 0.000192  loss: 6.0980 (6.1474)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [195/234]  eta: 0:02:29  lr: 0.000192  loss: 6.0806 (6.1479)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [200/234]  eta: 0:02:10  lr: 0.000192  loss: 5.9424 (6.1447)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [205/234]  eta: 0:01:50  lr: 0.000192  loss: 6.0271 (6.1440)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [210/234]  eta: 0:01:30  lr: 0.000192  loss: 6.1121 (6.1502)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [215/234]  eta: 0:01:13  lr: 0.000192  loss: 6.0271 (6.1459)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [220/234]  eta: 0:00:54  lr: 0.000192  loss: 6.0875 (6.1452)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [225/234]  eta: 0:00:34  lr: 0.000192  loss: 6.0025 (6.1447)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [230/234]  eta: 0:00:15  lr: 0.000192  loss: 5.9989 (6.1429)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [2]  [233/234]  eta: 0:00:03  lr: 0.000192  loss: 5.9910 (6.1349)  loss_classi
Epoch: [2] Total time: 0:14:48 (3.7979 s / it)
Test Started
Test Ended
0.8345454393719012
This Epochs AveragePlxError Test tensor(16.8157)
Best Epochs AveragePlxError Test tensor(16.7072)
```

```
tensor(9248.6348) 550.00001
-- -- -- -- -- --
Epoch: [3]  [  0/234]  eta: 0:53:37  lr: 0.000188  loss: 6.2561 (6.2561)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [  5/234]  eta: 0:17:48  lr: 0.000188  loss: 6.0608 (6.0963)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 10/234]  eta: 0:20:11  lr: 0.000188  loss: 6.0608 (6.0902)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 15/234]  eta: 0:16:37  lr: 0.000188  loss: 6.0411 (6.0955)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 20/234]  eta: 0:16:23  lr: 0.000188  loss: 6.0411 (6.0837)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 25/234]  eta: 0:14:46  lr: 0.000188  loss: 6.0792 (6.0810)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 30/234]  eta: 0:13:30  lr: 0.000188  loss: 6.0792 (6.0733)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 35/234]  eta: 0:12:37  lr: 0.000188  loss: 6.1790 (6.1072)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 40/234]  eta: 0:12:32  lr: 0.000188  loss: 6.1065 (6.1022)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 45/234]  eta: 0:13:09  lr: 0.000188  loss: 6.1219 (6.1198)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 50/234]  eta: 0:12:28  lr: 0.000188  loss: 6.0911 (6.1015)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 55/234]  eta: 0:11:50  lr: 0.000188  loss: 6.0408 (6.1040)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 60/234]  eta: 0:11:35  lr: 0.000188  loss: 6.0850 (6.1177)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 65/234]  eta: 0:10:58  lr: 0.000188  loss: 6.0187 (6.1110)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 70/234]  eta: 0:10:26  lr: 0.000188  loss: 6.1121 (6.1254)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 75/234]  eta: 0:09:56  lr: 0.000188  loss: 6.0907 (6.1059)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 80/234]  eta: 0:10:23  lr: 0.000188  loss: 6.0187 (6.0992)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 85/234]  eta: 0:09:52  lr: 0.000188  loss: 6.0187 (6.0948)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 90/234]  eta: 0:09:24  lr: 0.000188  loss: 5.9464 (6.0975)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [ 95/234]  eta: 0:08:57  lr: 0.000188  loss: 5.9543 (6.0931)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [100/234]  eta: 0:08:42  lr: 0.000188  loss: 6.0105 (6.0887)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [105/234]  eta: 0:08:15  lr: 0.000188  loss: 6.0105 (6.0836)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [110/234]  eta: 0:08:07  lr: 0.000188  loss: 5.9648 (6.0781)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [115/234]  eta: 0:07:44  lr: 0.000188  loss: 5.9239 (6.0708)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [120/234]  eta: 0:07:26  lr: 0.000188  loss: 5.9239 (6.0718)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [125/234]  eta: 0:07:01  lr: 0.000188  loss: 5.9792 (6.0785)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [130/234]  eta: 0:06:36  lr: 0.000188  loss: 5.9792 (6.0677)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [135/234]  eta: 0:06:13  lr: 0.000188  loss: 6.0982 (6.0761)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [140/234]  eta: 0:05:57  lr: 0.000188  loss: 6.0982 (6.0766)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [145/234]  eta: 0:05:45  lr: 0.000188  loss: 5.9857 (6.0742)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [150/234]  eta: 0:05:24  lr: 0.000188  loss: 6.1280 (6.0839)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [155/234]  eta: 0:05:02  lr: 0.000188  loss: 5.9543 (6.0787)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [160/234]  eta: 0:04:44  lr: 0.000188  loss: 5.9573 (6.0925)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [165/234]  eta: 0:04:23  lr: 0.000188  loss: 5.9783 (6.0933)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [170/234]  eta: 0:04:02  lr: 0.000188  loss: 5.9783 (6.0927)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [175/234]  eta: 0:03:41  lr: 0.000188  loss: 6.1105 (6.0915)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [3]  [180/234]  eta: 0:03:29  lr: 0.000188  loss: 6.1185 (6.0927)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [185/234]  eta: 0:03:08  lr: 0.000188  loss: 6.0571 (6.0882)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [190/234]  eta: 0:02:48  lr: 0.000188  loss: 5.9908 (6.0843)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [195/234]  eta: 0:02:27  lr: 0.000188  loss: 5.8945 (6.0791)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [200/234]  eta: 0:02:09  lr: 0.000188  loss: 5.8898 (6.0808)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [205/234]  eta: 0:01:49  lr: 0.000188  loss: 5.8784 (6.0759)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [210/234]  eta: 0:01:32  lr: 0.000188  loss: 5.8945 (6.0738)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [215/234]  eta: 0:01:12  lr: 0.000188  loss: 6.1359 (6.0744)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [220/234]  eta: 0:00:53  lr: 0.000188  loss: 6.0867 (6.0732)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [3]  [225/234]  eta: 0:00:34  lr: 0.000188  loss: 6.1359 (6.0751)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [230/234]  eta: 0:00:15  lr: 0.000188  loss: 6.1119 (6.0820)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [3]  [233/234]  eta: 0:00:03  lr: 0.000188  loss: 6.0961 (6.0807)  loss_classi
Epoch: [3] Total time: 0:14:38 (3.7543 s / it)
Test Started
Test Ended
0.8135283215077089
This Epochs AveragePlxError Test tensor(17.6373)
Best Epochs AveragePlxError Test tensor(16.7072)
tensor(9647.6016) 547.00001
-- -- -- -- -- --
Epoch: [4]  [  0/234]  eta: 0:54:04  lr: 0.000184  loss: 5.8528 (5.8528)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [  5/234]  eta: 0:28:34  lr: 0.000184  loss: 5.8528 (5.9889)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 10/234]  eta: 0:20:57  lr: 0.000184  loss: 6.1790 (6.0767)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 15/234]  eta: 0:17:04  lr: 0.000184  loss: 6.0565 (6.0910)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 20/234]  eta: 0:16:21  lr: 0.000184  loss: 6.0273 (6.0353)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 25/234]  eta: 0:14:43  lr: 0.000184  loss: 6.0264 (6.0462)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 30/234]  eta: 0:13:34  lr: 0.000184  loss: 6.0242 (6.0685)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 35/234]  eta: 0:12:39  lr: 0.000184  loss: 6.0242 (6.0663)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 40/234]  eta: 0:14:09  lr: 0.000184  loss: 6.0019 (6.0616)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 45/234]  eta: 0:13:15  lr: 0.000184  loss: 6.0019 (6.0683)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 50/234]  eta: 0:12:28  lr: 0.000184  loss: 6.1335 (6.0816)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 55/234]  eta: 0:11:51  lr: 0.000184  loss: 6.1572 (6.0904)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 60/234]  eta: 0:11:40  lr: 0.000184  loss: 6.1850 (6.0827)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 65/234]  eta: 0:11:09  lr: 0.000184  loss: 6.1572 (6.0729)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 70/234]  eta: 0:11:10  lr: 0.000184  loss: 5.9866 (6.0826)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 75/234]  eta: 0:10:41  lr: 0.000184  loss: 6.0332 (6.0794)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 80/234]  eta: 0:10:24  lr: 0.000184  loss: 6.0525 (6.0921)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 85/234]  eta: 0:09:54  lr: 0.000184  loss: 6.0525 (6.0742)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 90/234]  eta: 0:09:27  lr: 0.000184  loss: 6.0525 (6.0833)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [ 95/234]  eta: 0:08:59  lr: 0.000184  loss: 6.0293 (6.0702)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [100/234]  eta: 0:09:01  lr: 0.000184  loss: 5.8333 (6.0649)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [105/234]  eta: 0:08:37  lr: 0.000184  loss: 5.9729 (6.0713)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [110/234]  eta: 0:08:10  lr: 0.000184  loss: 5.9386 (6.0597)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [115/234]  eta: 0:07:44  lr: 0.000184  loss: 5.9386 (6.0477)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [4]  [120/234]  eta: 0:07:28  lr: 0.000184  loss: 5.9922 (6.0531)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [125/234]  eta: 0:07:02  lr: 0.000184  loss: 5.9989 (6.0604)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [130/234]  eta: 0:06:39  lr: 0.000184  loss: 6.0316 (6.0613)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [135/234]  eta: 0:06:27  lr: 0.000184  loss: 6.0678 (6.0680)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [140/234]  eta: 0:06:11  lr: 0.000184  loss: 6.1725 (6.0697)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [145/234]  eta: 0:05:48  lr: 0.000184  loss: 6.0678 (6.0718)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [150/234]  eta: 0:05:25  lr: 0.000184  loss: 6.0025 (6.0582)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [155/234]  eta: 0:05:03  lr: 0.000184  loss: 5.9720 (6.0550)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [160/234]  eta: 0:04:45  lr: 0.000184  loss: 5.7287 (6.0440)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [4]  [165/234]  eta: 0:04:29  lr: 0.000184  loss: 5.7035 (6.0482)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [170/234]  eta: 0:04:09  lr: 0.000184  loss: 5.9202 (6.0490)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [175/234]  eta: 0:03:48  lr: 0.000184  loss: 5.8536 (6.0406)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [180/234]  eta: 0:03:29  lr: 0.000184  loss: 5.8713 (6.0349)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [185/234]  eta: 0:03:09  lr: 0.000184  loss: 5.8536 (6.0368)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [190/234]  eta: 0:02:48  lr: 0.000184  loss: 5.8251 (6.0342)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [195/234]  eta: 0:02:28  lr: 0.000184  loss: 5.8713 (6.0289)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [200/234]  eta: 0:02:12  lr: 0.000184  loss: 6.0673 (6.0337)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [205/234]  eta: 0:01:52  lr: 0.000184  loss: 5.9937 (6.0293)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [4]  [210/234]  eta: 0:01:32  lr: 0.000184  loss: 5.9937 (6.0318)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [215/234]  eta: 0:01:12  lr: 0.000184  loss: 6.0673 (6.0348)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [220/234]  eta: 0:00:53  lr: 0.000184  loss: 5.9944 (6.0336)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [225/234]  eta: 0:00:34  lr: 0.000184  loss: 5.9887 (6.0317)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [230/234]  eta: 0:00:15  lr: 0.000184  loss: 5.9191 (6.0257)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [4]  [233/234]  eta: 0:00:03  lr: 0.000184  loss: 5.9191 (6.0225)  loss_classi
Epoch: [4] Total time: 0:14:56 (3.8310 s / it)
Test Started
Test Ended
0.8051001674845143
This Epochs AveragePlxError Test tensor(16.9324)
Best Epochs AveragePlxError Test tensor(16.7072)
tensor(9295.8770) 549.00001
-- -- -- -- -- --
Epoch: [5]  [  0/234]  eta: 1:11:49  lr: 0.000181  loss: 6.1798 (6.1798)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [  5/234]  eta: 0:20:28  lr: 0.000181  loss: 5.9560 (6.0742)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 10/234]  eta: 0:15:24  lr: 0.000181  loss: 6.1798 (6.1766)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 15/234]  eta: 0:13:24  lr: 0.000181  loss: 6.0932 (6.0709)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 20/234]  eta: 0:13:45  lr: 0.000181  loss: 6.0130 (6.0264)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 25/234]  eta: 0:14:28  lr: 0.000181  loss: 6.0152 (6.0223)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 30/234]  eta: 0:13:21  lr: 0.000181  loss: 6.0152 (6.0670)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 35/234]  eta: 0:12:28  lr: 0.000181  loss: 6.0152 (6.0575)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 40/234]  eta: 0:12:40  lr: 0.000181  loss: 6.0738 (6.0711)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 45/234]  eta: 0:11:55  lr: 0.000181  loss: 6.0388 (6.0698)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 50/234]  eta: 0:11:16  lr: 0.000181  loss: 6.0332 (6.0690)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 55/234]  eta: 0:10:43  lr: 0.000181  loss: 6.0332 (6.0843)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [5]  [ 60/234]  eta: 0:11:12  lr: 0.000181  loss: 6.0332 (6.0586)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 65/234]  eta: 0:10:41  lr: 0.000181  loss: 6.0432 (6.0623)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 70/234]  eta: 0:10:10  lr: 0.000181  loss: 6.0004 (6.0599)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 75/234]  eta: 0:09:50  lr: 0.000181  loss: 6.0004 (6.0632)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 80/234]  eta: 0:09:37  lr: 0.000181  loss: 5.9584 (6.0454)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 85/234]  eta: 0:09:11  lr: 0.000181  loss: 5.9360 (6.0435)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 90/234]  eta: 0:08:45  lr: 0.000181  loss: 5.9014 (6.0271)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [ 95/234]  eta: 0:08:20  lr: 0.000181  loss: 5.8807 (6.0180)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [100/234]  eta: 0:08:25  lr: 0.000181  loss: 5.8807 (6.0210)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [5]  [105/234]  eta: 0:08:05  lr: 0.000181  loss: 5.8609 (6.0149)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [110/234]  eta: 0:07:40  lr: 0.000181  loss: 5.9503 (6.0200)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [115/234]  eta: 0:07:17  lr: 0.000181  loss: 5.8609 (6.0026)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [120/234]  eta: 0:07:00  lr: 0.000181  loss: 5.8893 (6.0124)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [125/234]  eta: 0:06:39  lr: 0.000181  loss: 5.9503 (6.0118)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [130/234]  eta: 0:06:26  lr: 0.000181  loss: 5.8459 (5.9957)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [135/234]  eta: 0:06:04  lr: 0.000181  loss: 5.9289 (5.9993)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [140/234]  eta: 0:05:50  lr: 0.000181  loss: 5.9289 (6.0001)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [145/234]  eta: 0:05:29  lr: 0.000181  loss: 5.7102 (5.9855)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [5]  [150/234]  eta: 0:05:08  lr: 0.000181  loss: 5.8171 (5.9747)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [155/234]  eta: 0:04:48  lr: 0.000181  loss: 5.8171 (5.9740)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [160/234]  eta: 0:04:30  lr: 0.000181  loss: 5.6642 (5.9735)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [165/234]  eta: 0:04:15  lr: 0.000181  loss: 5.8430 (5.9782)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [170/234]  eta: 0:03:55  lr: 0.000181  loss: 5.9404 (5.9751)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [175/234]  eta: 0:03:36  lr: 0.000181  loss: 6.0141 (5.9765)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [180/234]  eta: 0:03:19  lr: 0.000181  loss: 6.0016 (5.9816)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [185/234]  eta: 0:02:59  lr: 0.000181  loss: 6.0016 (5.9841)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [190/234]  eta: 0:02:40  lr: 0.000181  loss: 5.9967 (5.9806)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [5]  [195/234]  eta: 0:02:21  lr: 0.000181  loss: 5.9967 (5.9817)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [200/234]  eta: 0:02:06  lr: 0.000181  loss: 6.0073 (5.9913)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [205/234]  eta: 0:01:47  lr: 0.000181  loss: 5.8965 (5.9846)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [210/234]  eta: 0:01:28  lr: 0.000181  loss: 5.9221 (5.9830)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [215/234]  eta: 0:01:09  lr: 0.000181  loss: 6.0073 (5.9919)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [220/234]  eta: 0:00:51  lr: 0.000181  loss: 5.9221 (5.9873)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [225/234]  eta: 0:00:33  lr: 0.000181  loss: 5.9573 (5.9878)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [230/234]  eta: 0:00:14  lr: 0.000181  loss: 5.9573 (5.9877)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [5]  [233/234]  eta: 0:00:03  lr: 0.000181  loss: 5.9350 (5.9854)  loss_classi
Epoch: [5] Total time: 0:14:06 (3.6155 s / it)
Test Started
Test Ended
0.8500913921372689
This Epochs AveragePlxError Test tensor(15.6015)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(8534.0381) 547.00001
```

```
tensor(8554.0381) 547.00001
-- -- -- -- -- --
Epoch: [6]  [  0/234]  eta: 1:49:39  lr: 0.000177  loss: 5.6674 (5.6674)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [  5/234]  eta: 0:28:42  lr: 0.000177  loss: 5.7531 (5.9917)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 10/234]  eta: 0:19:52  lr: 0.000177  loss: 5.8440 (5.9576)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 15/234]  eta: 0:16:26  lr: 0.000177  loss: 5.7943 (5.8423)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 20/234]  eta: 0:15:52  lr: 0.000177  loss: 5.7943 (5.8484)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 25/234]  eta: 0:14:26  lr: 0.000177  loss: 5.7515 (5.8531)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 30/234]  eta: 0:13:19  lr: 0.000177  loss: 5.6947 (5.8450)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 35/234]  eta: 0:12:28  lr: 0.000177  loss: 5.7515 (5.8799)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 40/234]  eta: 0:13:43  lr: 0.000177  loss: 5.7515 (5.8684)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [6]  [ 45/234]  eta: 0:12:50  lr: 0.000177  loss: 5.7650 (5.8557)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 50/234]  eta: 0:12:06  lr: 0.000177  loss: 5.8013 (5.8406)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 55/234]  eta: 0:11:28  lr: 0.000177  loss: 5.8692 (5.8774)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 60/234]  eta: 0:11:18  lr: 0.000177  loss: 5.9292 (5.8977)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 65/234]  eta: 0:10:44  lr: 0.000177  loss: 5.8720 (5.9129)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 70/234]  eta: 0:10:45  lr: 0.000177  loss: 6.0093 (5.9173)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 75/234]  eta: 0:10:22  lr: 0.000177  loss: 6.0271 (5.9371)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 80/234]  eta: 0:10:07  lr: 0.000177  loss: 6.0093 (5.9465)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 85/234]  eta: 0:09:37  lr: 0.000177  loss: 6.0093 (5.9441)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [6]  [ 90/234]  eta: 0:09:10  lr: 0.000177  loss: 6.0147 (5.9474)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [ 95/234]  eta: 0:08:42  lr: 0.000177  loss: 6.0291 (5.9658)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [100/234]  eta: 0:08:28  lr: 0.000177  loss: 6.0291 (5.9630)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [105/234]  eta: 0:08:22  lr: 0.000177  loss: 6.0291 (5.9652)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [110/234]  eta: 0:07:56  lr: 0.000177  loss: 6.1235 (5.9723)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [115/234]  eta: 0:07:32  lr: 0.000177  loss: 5.9435 (5.9825)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [120/234]  eta: 0:07:15  lr: 0.000177  loss: 5.9452 (5.9789)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [125/234]  eta: 0:06:52  lr: 0.000177  loss: 5.9452 (5.9761)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [130/234]  eta: 0:06:29  lr: 0.000177  loss: 5.9520 (5.9742)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [6]  [135/234]  eta: 0:06:07  lr: 0.000177  loss: 5.9912 (5.9773)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [140/234]  eta: 0:06:02  lr: 0.000177  loss: 5.9971 (5.9854)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [145/234]  eta: 0:05:40  lr: 0.000177  loss: 6.0289 (5.9923)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [150/234]  eta: 0:05:19  lr: 0.000177  loss: 6.0356 (5.9896)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [155/234]  eta: 0:04:58  lr: 0.000177  loss: 6.0289 (5.9902)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [160/234]  eta: 0:04:40  lr: 0.000177  loss: 5.9826 (5.9918)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [165/234]  eta: 0:04:20  lr: 0.000177  loss: 5.8717 (5.9824)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [170/234]  eta: 0:04:03  lr: 0.000177  loss: 5.9235 (5.9860)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [175/234]  eta: 0:03:44  lr: 0.000177  loss: 5.9235 (5.9818)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [6]  [180/234]  eta: 0:03:26  lr: 0.000177  loss: 5.9235 (5.9757)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [185/234]  eta: 0:03:05  lr: 0.000177  loss: 5.9578 (5.9737)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [190/234]  eta: 0:02:45  lr: 0.000177  loss: 5.9578 (5.9785)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [195/234]  eta: 0:02:25  lr: 0.000177  loss: 6.0072 (5.9808)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [200/234]  eta: 0:02:09  lr: 0.000177  loss: 5.8728 (5.9770)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [205/234]  eta: 0:01:50  lr: 0.000177  loss: 5.9120 (5.9803)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [210/234]  eta: 0:01:30  lr: 0.000177  loss: 5.8728 (5.9785)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [215/234]  eta: 0:01:11  lr: 0.000177  loss: 5.8537 (5.9744)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [220/234]  eta: 0:00:52  lr: 0.000177  loss: 5.9193 (5.9870)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [6]  [225/234]  eta: 0:00:33  lr: 0.000177  loss: 5.9384 (5.9896)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [230/234]  eta: 0:00:14  lr: 0.000177  loss: 6.0518 (5.9900)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [6]  [233/234]  eta: 0:00:03  lr: 0.000177  loss: 6.0518 (5.9842)  loss_classi
Epoch: [6] Total time: 0:14:28 (3.7096 s / it)
Test Started
Test Ended
0.8302919556516067
This Epochs AveragePlxError Test tensor(16.3136)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(8939.8721) 548.00001
-- -- -- -- -- --
Epoch: [7]  [  0/234]  eta: 1:40:16  lr: 0.000174  loss: 5.8863 (5.8863)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [  5/234]  eta: 0:28:43  lr: 0.000174  loss: 5.6963 (5.8106)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 10/234]  eta: 0:19:39  lr: 0.000174  loss: 5.8863 (5.8620)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 15/234]  eta: 0:16:16  lr: 0.000174  loss: 5.8863 (5.9034)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 20/234]  eta: 0:16:03  lr: 0.000174  loss: 5.9186 (5.9211)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 25/234]  eta: 0:14:33  lr: 0.000174  loss: 5.9186 (5.8629)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [7]  [ 30/234]  eta: 0:13:22  lr: 0.000174  loss: 5.9106 (5.8849)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 35/234]  eta: 0:13:33  lr: 0.000174  loss: 5.8955 (5.9000)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 40/234]  eta: 0:13:44  lr: 0.000174  loss: 5.8306 (5.9113)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 45/234]  eta: 0:12:51  lr: 0.000174  loss: 5.8955 (5.9195)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 50/234]  eta: 0:12:10  lr: 0.000174  loss: 5.8955 (5.9454)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 55/234]  eta: 0:11:30  lr: 0.000174  loss: 5.8675 (5.9479)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 60/234]  eta: 0:11:19  lr: 0.000174  loss: 5.8307 (5.9398)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 65/234]  eta: 0:10:44  lr: 0.000174  loss: 5.8300 (5.9102)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 70/234]  eta: 0:10:49  lr: 0.000174  loss: 5.7320 (5.9147)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [7]  [ 75/234]  eta: 0:10:18  lr: 0.000174  loss: 5.8300 (5.9286)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 80/234]  eta: 0:10:02  lr: 0.000174  loss: 5.8910 (5.9279)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 85/234]  eta: 0:09:32  lr: 0.000174  loss: 5.8895 (5.9190)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 90/234]  eta: 0:09:05  lr: 0.000174  loss: 5.9602 (5.9307)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [ 95/234]  eta: 0:08:39  lr: 0.000174  loss: 5.8910 (5.9263)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [100/234]  eta: 0:08:50  lr: 0.000174  loss: 5.7665 (5.9121)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [105/234]  eta: 0:08:23  lr: 0.000174  loss: 5.8293 (5.9127)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [110/234]  eta: 0:07:57  lr: 0.000174  loss: 5.7975 (5.9064)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [115/234]  eta: 0:07:32  lr: 0.000174  loss: 5.8363 (5.9340)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [7]  [120/234]  eta: 0:07:15  lr: 0.000174  loss: 5.9074 (5.9198)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [125/234]  eta: 0:06:52  lr: 0.000174  loss: 5.9674 (5.9322)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [130/234]  eta: 0:06:29  lr: 0.000174  loss: 5.9933 (5.9336)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [135/234]  eta: 0:06:16  lr: 0.000174  loss: 5.8259 (5.9286)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [140/234]  eta: 0:06:02  lr: 0.000174  loss: 5.9419 (5.9425)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [145/234]  eta: 0:05:40  lr: 0.000174  loss: 5.8259 (5.9391)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [150/234]  eta: 0:05:18  lr: 0.000174  loss: 5.9087 (5.9424)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [155/234]  eta: 0:04:57  lr: 0.000174  loss: 5.9802 (5.9419)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [160/234]  eta: 0:04:39  lr: 0.000174  loss: 5.9087 (5.9448)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [7]  [165/234]  eta: 0:04:23  lr: 0.000174  loss: 5.9087 (5.9428)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [170/234]  eta: 0:04:01  lr: 0.000174  loss: 5.8828 (5.9385)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [175/234]  eta: 0:03:43  lr: 0.000174  loss: 5.8828 (5.9385)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [180/234]  eta: 0:03:25  lr: 0.000174  loss: 5.7675 (5.9291)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [185/234]  eta: 0:03:05  lr: 0.000174  loss: 5.7759 (5.9292)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [190/234]  eta: 0:02:45  lr: 0.000174  loss: 5.8481 (5.9352)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [195/234]  eta: 0:02:25  lr: 0.000174  loss: 5.8876 (5.9414)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [200/234]  eta: 0:02:09  lr: 0.000174  loss: 6.1495 (5.9440)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [205/234]  eta: 0:01:49  lr: 0.000174  loss: 6.1876 (5.9508)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [7]  [210/234]  eta: 0:01:30  lr: 0.000174  loss: 6.1412 (5.9480)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [215/234]  eta: 0:01:11  lr: 0.000174  loss: 6.0450 (5.9578)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [220/234]  eta: 0:00:52  lr: 0.000174  loss: 6.0450 (5.9607)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [225/234]  eta: 0:00:33  lr: 0.000174  loss: 6.0137 (5.9569)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [230/234]  eta: 0:00:14  lr: 0.000174  loss: 5.9771 (5.9523)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [7]  [233/234]  eta: 0:00:03  lr: 0.000174  loss: 5.9057 (5.9555)  loss_classi
Epoch: [7] Total time: 0:14:34 (3.7388 s / it)
Test Started
Test Ended
0.8241758090810292
This Epochs AveragePlxError Test tensor(16.5253)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9022.8193) 546.00001
-- -- -- -- -- --
Epoch: [8]  [  0/234]  eta: 1:18:25  lr: 0.000170  loss: 5.8502 (5.8502)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [  5/234]  eta: 0:22:00  lr: 0.000170  loss: 5.8502 (5.8688)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 10/234]  eta: 0:16:27  lr: 0.000170  loss: 5.8502 (5.7667)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [8]  [ 15/234]  eta: 0:14:06  lr: 0.000170  loss: 5.9765 (5.8789)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 20/234]  eta: 0:14:16  lr: 0.000170  loss: 6.0251 (5.9347)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 25/234]  eta: 0:13:07  lr: 0.000170  loss: 6.0522 (5.9768)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 30/234]  eta: 0:13:31  lr: 0.000170  loss: 6.0478 (5.9344)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 35/234]  eta: 0:13:06  lr: 0.000170  loss: 5.8972 (5.9123)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 40/234]  eta: 0:12:58  lr: 0.000170  loss: 5.8703 (5.9256)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 45/234]  eta: 0:12:11  lr: 0.000170  loss: 5.7835 (5.9188)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 50/234]  eta: 0:11:34  lr: 0.000170  loss: 5.8151 (5.9094)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 55/234]  eta: 0:10:57  lr: 0.000170  loss: 5.8944 (5.9300)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
Epoch: [8]  [ 60/234]  eta: 0:11:15  lr: 0.000170  loss: 5.8172 (5.9108)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 65/234]  eta: 0:10:41  lr: 0.000170  loss: 5.8151 (5.9032)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 70/234]  eta: 0:10:26  lr: 0.000170  loss: 5.9718 (5.9228)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 75/234]  eta: 0:09:57  lr: 0.000170  loss: 5.6926 (5.8885)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 80/234]  eta: 0:09:43  lr: 0.000170  loss: 5.7540 (5.8857)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 85/234]  eta: 0:09:15  lr: 0.000170  loss: 5.7637 (5.8765)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 90/234]  eta: 0:08:50  lr: 0.000170  loss: 5.7637 (5.8789)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [ 95/234]  eta: 0:08:24  lr: 0.000170  loss: 5.8445 (5.8792)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [100/234]  eta: 0:08:24  lr: 0.000170  loss: 5.8898 (5.8878)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [8]  [105/234]  eta: 0:08:05  lr: 0.000170  loss: 5.9676 (5.9009)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [110/234]  eta: 0:07:41  lr: 0.000170  loss: 5.8465 (5.8907)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [115/234]  eta: 0:07:17  lr: 0.000170  loss: 5.7843 (5.8860)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [120/234]  eta: 0:07:02  lr: 0.000170  loss: 5.8448 (5.8909)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [125/234]  eta: 0:06:40  lr: 0.000170  loss: 5.9387 (5.9165)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [130/234]  eta: 0:06:26  lr: 0.000170  loss: 5.9752 (5.9145)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [135/234]  eta: 0:06:04  lr: 0.000170  loss: 6.0708 (5.9194)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [140/234]  eta: 0:05:51  lr: 0.000170  loss: 6.1162 (5.9248)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [145/234]  eta: 0:05:30  lr: 0.000170  loss: 6.0302 (5.9233)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [8]  [150/234]  eta: 0:05:08  lr: 0.000170  loss: 6.0383 (5.9303)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [155/234]  eta: 0:04:48  lr: 0.000170  loss: 6.0302 (5.9276)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [160/234]  eta: 0:04:31  lr: 0.000170  loss: 5.8921 (5.9171)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [165/234]  eta: 0:04:15  lr: 0.000170  loss: 5.8921 (5.9110)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [170/234]  eta: 0:03:55  lr: 0.000170  loss: 5.8921 (5.9132)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [175/234]  eta: 0:03:37  lr: 0.000170  loss: 5.8921 (5.9150)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [180/234]  eta: 0:03:19  lr: 0.000170  loss: 5.8795 (5.9080)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [185/234]  eta: 0:02:59  lr: 0.000170  loss: 5.8224 (5.9025)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [190/234]  eta: 0:02:40  lr: 0.000170  loss: 5.7441 (5.8962)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [8]  [195/234]  eta: 0:02:21  lr: 0.000170  loss: 5.6291 (5.8961)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [200/234]  eta: 0:02:05  lr: 0.000170  loss: 5.6852 (5.8918)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [205/234]  eta: 0:01:46  lr: 0.000170  loss: 5.8538 (5.8919)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [210/234]  eta: 0:01:27  lr: 0.000170  loss: 5.8783 (5.8994)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [215/234]  eta: 0:01:09  lr: 0.000170  loss: 5.8783 (5.9014)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [220/234]  eta: 0:00:51  lr: 0.000170  loss: 6.0089 (5.9016)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [225/234]  eta: 0:00:32  lr: 0.000170  loss: 6.0089 (5.8984)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [230/234]  eta: 0:00:14  lr: 0.000170  loss: 6.0089 (5.9007)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [8]  [233/234]  eta: 0:00:03  lr: 0.000170  loss: 5.7750 (5.8982)  loss_classi
Epoch: [8] Total time: 0:14:00 (3.5935 s / it)
Test Started
Test Ended
0.8083941458322237
This Epochs AveragePlxError Test tensor(17.1961)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9423.4658) 548.00001
```

```
-- -- -- -- -- --
Epoch: [9]  [  0/234]  eta: 1:33:52  lr: 0.000167  loss: 5.7887 (5.7887)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [  5/234]  eta: 0:26:54  lr: 0.000167  loss: 5.7832 (5.8858)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 10/234]  eta: 0:19:05  lr: 0.000167  loss: 5.8313 (5.9438)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 15/234]  eta: 0:15:58  lr: 0.000167  loss: 5.7887 (5.8737)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 20/234]  eta: 0:15:32  lr: 0.000167  loss: 5.8277 (5.8800)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 25/234]  eta: 0:13:57  lr: 0.000167  loss: 5.8277 (5.8849)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 30/234]  eta: 0:12:55  lr: 0.000167  loss: 5.6876 (5.8549)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 35/234]  eta: 0:12:05  lr: 0.000167  loss: 5.8052 (5.8923)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 40/234]  eta: 0:13:10  lr: 0.000167  loss: 5.7691 (5.8784)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [9]  [ 45/234]  eta: 0:12:23  lr: 0.000167  loss: 5.7691 (5.8772)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 50/234]  eta: 0:11:42  lr: 0.000167  loss: 5.8333 (5.8651)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 55/234]  eta: 0:11:06  lr: 0.000167  loss: 5.7604 (5.8533)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 60/234]  eta: 0:10:56  lr: 0.000167  loss: 5.7604 (5.8526)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 65/234]  eta: 0:10:25  lr: 0.000167  loss: 5.8803 (5.8837)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 70/234]  eta: 0:10:18  lr: 0.000167  loss: 5.9177 (5.8918)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 75/234]  eta: 0:09:48  lr: 0.000167  loss: 6.0228 (5.9017)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 80/234]  eta: 0:09:45  lr: 0.000167  loss: 6.0970 (5.9057)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 85/234]  eta: 0:09:16  lr: 0.000167  loss: 5.9743 (5.9060)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [9]  [ 90/234]  eta: 0:08:50  lr: 0.000167  loss: 6.0228 (5.9132)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [ 95/234]  eta: 0:08:27  lr: 0.000167  loss: 5.9396 (5.9155)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [100/234]  eta: 0:08:14  lr: 0.000167  loss: 5.9034 (5.9150)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [105/234]  eta: 0:08:02  lr: 0.000167  loss: 5.9034 (5.9116)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [110/234]  eta: 0:07:44  lr: 0.000167  loss: 5.8734 (5.9145)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [115/234]  eta: 0:07:21  lr: 0.000167  loss: 5.8455 (5.9083)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [120/234]  eta: 0:07:06  lr: 0.000167  loss: 5.8455 (5.9068)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [125/234]  eta: 0:06:43  lr: 0.000167  loss: 5.7720 (5.9004)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [130/234]  eta: 0:06:21  lr: 0.000167  loss: 5.7194 (5.8979)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [9]  [135/234]  eta: 0:05:59  lr: 0.000167  loss: 5.7194 (5.8915)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [140/234]  eta: 0:05:51  lr: 0.000167  loss: 5.6836 (5.8875)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [145/234]  eta: 0:05:33  lr: 0.000167  loss: 5.7400 (5.8896)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [150/234]  eta: 0:05:12  lr: 0.000167  loss: 5.8231 (5.8908)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [155/234]  eta: 0:04:51  lr: 0.000167  loss: 5.9194 (5.8985)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [160/234]  eta: 0:04:35  lr: 0.000167  loss: 5.9194 (5.8986)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [165/234]  eta: 0:04:14  lr: 0.000167  loss: 5.9194 (5.8966)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [170/234]  eta: 0:03:54  lr: 0.000167  loss: 5.9002 (5.8907)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [175/234]  eta: 0:03:37  lr: 0.000167  loss: 5.8632 (5.8920)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [9]  [180/234]  eta: 0:03:21  lr: 0.000167  loss: 5.9165 (5.9024)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [185/234]  eta: 0:03:01  lr: 0.000167  loss: 5.8632 (5.8944)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [190/234]  eta: 0:02:42  lr: 0.000167  loss: 5.8632 (5.8949)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [195/234]  eta: 0:02:22  lr: 0.000167  loss: 5.7472 (5.8935)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [200/234]  eta: 0:02:05  lr: 0.000167  loss: 5.7472 (5.8942)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [205/234]  eta: 0:01:46  lr: 0.000167  loss: 5.9493 (5.8992)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [210/234]  eta: 0:01:28  lr: 0.000167  loss: 5.8964 (5.8965)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [215/234]  eta: 0:01:09  lr: 0.000167  loss: 5.8964 (5.8904)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [220/234]  eta: 0:00:51  lr: 0.000167  loss: 5.8511 (5.8933)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [225/234]  eta: 0:00:33  lr: 0.000167  loss: 5.7356 (5.8906)  loss_classi
```

```
Epoch: [9]  [229/234]  eta: 0:00:33  lr: 0.000167  loss: 5.7990 (5.8960)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [230/234]  eta: 0:00:14  lr: 0.000167  loss: 5.7844 (5.8918)  loss_classi
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [9]  [233/234]  eta: 0:00:03  lr: 0.000167  loss: 5.7877 (5.8917)  loss_classi
Epoch: [9] Total time: 0:14:09 (3.6310 s / it)
Test Started
Test Ended
0.84374998448989
This Epochs AveragePlxError Test tensor(15.8520)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(8623.5117) 544.00001
-- -- -- -- -- --
Epoch: [10]  [  0/234]  eta: 0:53:32  lr: 0.000163  loss: 5.2118 (5.2118)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [  5/234]  eta: 0:24:15  lr: 0.000163  loss: 6.0360 (5.8848)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 10/234]  eta: 0:19:46  lr: 0.000163  loss: 6.0360 (5.9286)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 15/234]  eta: 0:16:31  lr: 0.000163  loss: 5.8938 (5.8874)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 20/234]  eta: 0:16:00  lr: 0.000163  loss: 5.8938 (5.8251)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 25/234]  eta: 0:14:29  lr: 0.000163  loss: 5.8938 (5.8626)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [10]  [ 30/234]  eta: 0:13:18  lr: 0.000163  loss: 5.9348 (5.9385)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 35/234]  eta: 0:12:29  lr: 0.000163  loss: 5.8545 (5.9392)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 40/234]  eta: 0:13:07  lr: 0.000163  loss: 5.8545 (5.9283)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 45/234]  eta: 0:12:41  lr: 0.000163  loss: 5.7783 (5.9084)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 50/234]  eta: 0:12:01  lr: 0.000163  loss: 5.7890 (5.9076)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 55/234]  eta: 0:11:23  lr: 0.000163  loss: 5.8021 (5.9239)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 60/234]  eta: 0:11:15  lr: 0.000163  loss: 5.8021 (5.9128)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 65/234]  eta: 0:10:43  lr: 0.000163  loss: 6.0281 (5.9162)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 70/234]  eta: 0:10:11  lr: 0.000163  loss: 6.0397 (5.9181)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [10]  [ 75/234]  eta: 0:10:01  lr: 0.000163  loss: 6.0397 (5.9273)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 80/234]  eta: 0:09:56  lr: 0.000163  loss: 6.0397 (5.9255)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 85/234]  eta: 0:09:27  lr: 0.000163  loss: 6.0434 (5.9347)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 90/234]  eta: 0:09:00  lr: 0.000163  loss: 5.8263 (5.9126)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [ 95/234]  eta: 0:08:34  lr: 0.000163  loss: 5.8168 (5.9156)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [100/234]  eta: 0:08:21  lr: 0.000163  loss: 5.8085 (5.9112)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [105/234]  eta: 0:07:57  lr: 0.000163  loss: 5.7680 (5.9089)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [110/234]  eta: 0:07:43  lr: 0.000163  loss: 5.8168 (5.9101)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [115/234]  eta: 0:07:19  lr: 0.000163  loss: 5.8566 (5.9043)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [10]  [120/234]  eta: 0:07:07  lr: 0.000163  loss: 6.0486 (5.9212)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [125/234]  eta: 0:06:44  lr: 0.000163  loss: 6.0486 (5.9161)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [130/234]  eta: 0:06:22  lr: 0.000163  loss: 5.8566 (5.9172)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [135/234]  eta: 0:06:01  lr: 0.000163  loss: 5.7834 (5.9108)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [140/234]  eta: 0:05:50  lr: 0.000163  loss: 5.7510 (5.9084)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [145/234]  eta: 0:05:29  lr: 0.000163  loss: 5.7507 (5.8966)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [150/234]  eta: 0:05:08  lr: 0.000163  loss: 5.7510 (5.9059)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [155/234]  eta: 0:04:50  lr: 0.000163  loss: 5.7731 (5.9010)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [160/234]  eta: 0:04:33  lr: 0.000163  loss: 5.7731 (5.9100)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [165/234]  eta: 0:04:13  lr: 0.000163  loss: 5.7731 (5.9081)  loss_class
```

```
Epoch: [10]  [165/234]  eta: 0:04:13  lr: 0.000163  loss: 5.7731 (5.9081)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [170/234]  eta: 0:03:53  lr: 0.000163  loss: 5.7584 (5.8953)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [175/234]  eta: 0:03:34  lr: 0.000163  loss: 5.7995 (5.8966)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [180/234]  eta: 0:03:19  lr: 0.000163  loss: 5.7715 (5.8956)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [185/234]  eta: 0:03:01  lr: 0.000163  loss: 5.5999 (5.8866)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [190/234]  eta: 0:02:41  lr: 0.000163  loss: 5.5999 (5.8855)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [195/234]  eta: 0:02:22  lr: 0.000163  loss: 5.5895 (5.8786)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [200/234]  eta: 0:02:04  lr: 0.000163  loss: 5.5999 (5.8772)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [205/234]  eta: 0:01:45  lr: 0.000163  loss: 5.6356 (5.8714)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [210/234]  eta: 0:01:28  lr: 0.000163  loss: 5.6786 (5.8768)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [215/234]  eta: 0:01:09  lr: 0.000163  loss: 5.9071 (5.8801)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [220/234]  eta: 0:00:51  lr: 0.000163  loss: 5.9071 (5.8788)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [225/234]  eta: 0:00:33  lr: 0.000163  loss: 5.9486 (5.8827)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [230/234]  eta: 0:00:14  lr: 0.000163  loss: 5.9740 (5.8862)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [10]  [233/234]  eta: 0:00:03  lr: 0.000163  loss: 5.9740 (5.8886)  loss_class
Epoch: [10] Total time: 0:14:09 (3.6294 s / it)
Test Started
Test Ended
0.8062157073817969
This Epochs AveragePlxError Test tensor(17.6048)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9629.8105) 547.00001
-- -- -- -- -- --
Epoch: [11]  [  0/234]  eta: 0:55:06  lr: 0.000160  loss: 5.4737 (5.4737)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [  5/234]  eta: 0:17:55  lr: 0.000160  loss: 5.7667 (5.8822)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 10/234]  eta: 0:17:07  lr: 0.000160  loss: 5.5621 (5.7276)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [11]  [ 15/234]  eta: 0:15:34  lr: 0.000160  loss: 5.5621 (5.7082)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 20/234]  eta: 0:15:20  lr: 0.000160  loss: 5.7005 (5.7454)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 25/234]  eta: 0:13:56  lr: 0.000160  loss: 5.6110 (5.7834)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 30/234]  eta: 0:12:53  lr: 0.000160  loss: 5.7005 (5.7792)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 35/234]  eta: 0:12:03  lr: 0.000160  loss: 5.7843 (5.7726)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 40/234]  eta: 0:12:06  lr: 0.000160  loss: 5.8568 (5.7821)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 45/234]  eta: 0:11:59  lr: 0.000160  loss: 5.7898 (5.7676)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 50/234]  eta: 0:11:35  lr: 0.000160  loss: 5.7898 (5.7511)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 55/234]  eta: 0:11:03  lr: 0.000160  loss: 5.8568 (5.7883)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
```

```
Epoch: [11]  [ 60/234]  eta: 0:10:52  lr: 0.000160  loss: 5.6471 (5.7787)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 65/234]  eta: 0:10:20  lr: 0.000160  loss: 5.8654 (5.7938)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 70/234]  eta: 0:09:50  lr: 0.000160  loss: 5.8654 (5.7901)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 75/234]  eta: 0:09:22  lr: 0.000160  loss: 5.8654 (5.8040)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 80/234]  eta: 0:09:26  lr: 0.000160  loss: 5.8654 (5.7919)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 85/234]  eta: 0:09:05  lr: 0.000160  loss: 5.8112 (5.7886)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 90/234]  eta: 0:08:41  lr: 0.000160  loss: 5.7380 (5.7938)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [ 95/234]  eta: 0:08:16  lr: 0.000160  loss: 5.7191 (5.7944)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [100/234]  eta: 0:08:05  lr: 0.000160  loss: 5.7191 (5.7886)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [105/234]  eta: 0:07:41  lr: 0.000160  loss: 5.7191 (5.7910)  loss_class
```

```
Epoch: [11]  [105/234]  eta: 0:07:41  lr: 0.000160  loss: 5.7191 (5.7910)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [110/234]  eta: 0:07:18  lr: 0.000160  loss: 5.6726 (5.7860)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [115/234]  eta: 0:07:04  lr: 0.000160  loss: 5.6983 (5.7975)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [120/234]  eta: 0:06:53  lr: 0.000160  loss: 5.6983 (5.7964)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [125/234]  eta: 0:06:32  lr: 0.000160  loss: 5.6272 (5.7771)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [130/234]  eta: 0:06:11  lr: 0.000160  loss: 5.5950 (5.7662)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [135/234]  eta: 0:05:50  lr: 0.000160  loss: 5.5180 (5.7554)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [140/234]  eta: 0:05:36  lr: 0.000160  loss: 5.4921 (5.7491)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [145/234]  eta: 0:05:15  lr: 0.000160  loss: 5.6101 (5.7684)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [150/234]  eta: 0:04:59  lr: 0.000160  loss: 5.8634 (5.7755)  loss_class
```

-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [155/234] eta: 0:04:40 lr: 0.000160 loss: 5.9080 (5.7805) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [160/234] eta: 0:04:27 lr: 0.000160 loss: 6.0430 (5.7861) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [165/234] eta: 0:04:07 lr: 0.000160 loss: 5.8397 (5.7843) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [170/234] eta: 0:03:47 lr: 0.000160 loss: 5.8194 (5.7897) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [175/234] eta: 0:03:28 lr: 0.000160 loss: 5.8443 (5.7887) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [180/234] eta: 0:03:12 lr: 0.000160 loss: 5.8443 (5.7840) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [185/234] eta: 0:02:55 lr: 0.000160 loss: 5.8443 (5.7846) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [190/234] eta: 0:02:37 lr: 0.000160 loss: 5.7930 (5.7866) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11] [195/234] eta: 0:02:18 lr: 0.000160 loss: 5.7930 (5.7900) loss_class

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [200/234]  eta: 0:02:01  lr: 0.000160  loss: 5.8441 (5.7949)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [205/234]  eta: 0:01:43  lr: 0.000160  loss: 5.9591 (5.7978)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [210/234]  eta: 0:01:25  lr: 0.000160  loss: 5.8628 (5.7941)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [215/234]  eta: 0:01:06  lr: 0.000160  loss: 6.0165 (5.8053)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [220/234]  eta: 0:00:50  lr: 0.000160  loss: 5.9617 (5.8065)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [225/234]  eta: 0:00:32  lr: 0.000160  loss: 5.8628 (5.8038)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [230/234]  eta: 0:00:14  lr: 0.000160  loss: 5.8814 (5.8051)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [11]  [233/234]  eta: 0:00:03  lr: 0.000160  loss: 5.7113 (5.8051)  loss_class
Epoch: [11] Total time: 0:13:47 (3.5377 s / it)
Test Started
Test Ended
0.8201834711892941
This Epochs AveragePlxError Test tensor(16.9315)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9227.6777) 545.00001
-- -- -- -- -- --
```

```
Epoch: [12]  [  0/234]  eta: 0:55:00  lr: 0.000157  loss: 6.0518 (6.0518)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [  5/234]  eta: 0:17:52  lr: 0.000157  loss: 5.6164 (5.7537)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 10/234]  eta: 0:14:25  lr: 0.000157  loss: 5.6462 (5.7345)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 15/234]  eta: 0:14:01  lr: 0.000157  loss: 5.6462 (5.6975)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 20/234]  eta: 0:15:05  lr: 0.000157  loss: 5.5977 (5.7228)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 25/234]  eta: 0:13:50  lr: 0.000157  loss: 5.6462 (5.7299)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 30/234]  eta: 0:12:48  lr: 0.000157  loss: 5.7576 (5.7579)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 35/234]  eta: 0:11:59  lr: 0.000157  loss: 5.7717 (5.7726)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 40/234]  eta: 0:12:07  lr: 0.000157  loss: 5.7834 (5.7644)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 45/234]  eta: 0:11:21  lr: 0.000157  loss: 5.7717 (5.7619)  loss_class
```

```
Epoch: [12]  [ 45/234]  eta: 0:11:31  lr: 0.000157  loss: 5.7717 (5.7610)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 50/234]  eta: 0:11:24  lr: 0.000157  loss: 5.7717 (5.7645)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 55/234]  eta: 0:10:48  lr: 0.000157  loss: 5.6259 (5.7592)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 60/234]  eta: 0:10:51  lr: 0.000157  loss: 5.6259 (5.7670)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 65/234]  eta: 0:10:21  lr: 0.000157  loss: 5.6563 (5.7634)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 70/234]  eta: 0:09:51  lr: 0.000157  loss: 5.6259 (5.7529)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 75/234]  eta: 0:09:23  lr: 0.000157  loss: 5.5157 (5.7309)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 80/234]  eta: 0:09:12  lr: 0.000157  loss: 5.5128 (5.7321)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 85/234]  eta: 0:09:00  lr: 0.000157  loss: 5.5094 (5.7253)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [ 90/234]  eta: 0:08:34  lr: 0.000157  loss: 5.5625 (5.7225)  loss_class
```

Epoch: [12] [90/234] eta: 0:08:34 lr: 0.000157 loss: 5.9029 (5.7229) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [95/234] eta: 0:08:15 lr: 0.000157 loss: 5.6219 (5.7308) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [100/234] eta: 0:08:03 lr: 0.000157 loss: 5.7414 (5.7283) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [105/234] eta: 0:07:40 lr: 0.000157 loss: 5.7614 (5.7253) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [110/234] eta: 0:07:18 lr: 0.000157 loss: 5.8178 (5.7348) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [115/234] eta: 0:06:57 lr: 0.000157 loss: 5.8178 (5.7353) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [120/234] eta: 0:06:49 lr: 0.000157 loss: 5.7614 (5.7362) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [125/234] eta: 0:06:28 lr: 0.000157 loss: 5.7235 (5.7309) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [130/234] eta: 0:06:07 lr: 0.000157 loss: 5.7235 (5.7371) loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12] [135/234] eta: 0:05:50 lr: 0.000157 loss: 5.7444 (5.7391) loss_class

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [140/234]  eta: 0:05:35  lr: 0.000157  loss: 5.6836 (5.7341)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [145/234]  eta: 0:05:14  lr: 0.000157  loss: 5.7444 (5.7378)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [150/234]  eta: 0:04:55  lr: 0.000157  loss: 5.7037 (5.7382)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [155/234]  eta: 0:04:36  lr: 0.000157  loss: 5.6759 (5.7349)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [160/234]  eta: 0:04:24  lr: 0.000157  loss: 5.7037 (5.7447)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [165/234]  eta: 0:04:06  lr: 0.000157  loss: 5.7037 (5.7496)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [170/234]  eta: 0:03:47  lr: 0.000157  loss: 5.7203 (5.7557)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [175/234]  eta: 0:03:28  lr: 0.000157  loss: 5.7796 (5.7507)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [180/234]  eta: 0:03:11  lr: 0.000157  loss: 5.8077 (5.7625)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [185/234]  eta: 0:02:53  lr: 0.000157  loss: 5.8093 (5.7686)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [190/234]  eta: 0:02:36  lr: 0.000157  loss: 5.8093 (5.7683)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [195/234]  eta: 0:02:17  lr: 0.000157  loss: 5.8117 (5.7674)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [200/234]  eta: 0:02:01  lr: 0.000157  loss: 5.6654 (5.7659)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [205/234]  eta: 0:01:43  lr: 0.000157  loss: 5.5965 (5.7683)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [210/234]  eta: 0:01:24  lr: 0.000157  loss: 5.5733 (5.7658)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [215/234]  eta: 0:01:06  lr: 0.000157  loss: 5.5965 (5.7688)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [220/234]  eta: 0:00:49  lr: 0.000157  loss: 5.8325 (5.7730)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [225/234]  eta: 0:00:31  lr: 0.000157  loss: 5.8715 (5.7784)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [230/234]  eta: 0:00:14  lr: 0.000157  loss: 5.9031 (5.7765)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [12]  [233/234]  eta: 0:00:03  lr: 0.000157  loss: 5.8715 (5.7751)  loss_class
Epoch: [12] Total time: 0:13:40 (3.5085 s / it)
Test Started
Test Ended
0.8142076354424839
This Epochs AveragePlxError Test tensor(17.3846)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9544.1504) 549.00001
-- -- -- -- -- --
Epoch: [13]  [  0/234]  eta: 1:16:33  lr: 0.000154  loss: 5.7981 (5.7981)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [  5/234]  eta: 0:21:53  lr: 0.000154  loss: 5.7875 (5.8180)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 10/234]  eta: 0:16:15  lr: 0.000154  loss: 5.7875 (5.6734)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 15/234]  eta: 0:14:08  lr: 0.000154  loss: 5.7981 (5.7240)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 20/234]  eta: 0:15:24  lr: 0.000154  loss: 5.8255 (5.7738)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 25/234]  eta: 0:13:59  lr: 0.000154  loss: 5.6984 (5.7349)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 30/234]  eta: 0:12:51  lr: 0.000154  loss: 5.8603 (5.7757)  loss_class
```

```
Epoch: [13]  [ 30/234]  eta: 0:12:51  lr: 0.000154  loss: 5.8669 (5.7797)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 35/234]  eta: 0:12:00  lr: 0.000154  loss: 5.6984 (5.7636)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 40/234]  eta: 0:12:35  lr: 0.000154  loss: 5.6755 (5.7729)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 45/234]  eta: 0:11:53  lr: 0.000154  loss: 5.8210 (5.7700)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 50/234]  eta: 0:11:15  lr: 0.000154  loss: 5.6355 (5.7723)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 55/234]  eta: 0:10:41  lr: 0.000154  loss: 5.7725 (5.7873)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 60/234]  eta: 0:10:54  lr: 0.000154  loss: 5.8271 (5.8070)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 65/234]  eta: 0:10:24  lr: 0.000154  loss: 5.8442 (5.8018)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 70/234]  eta: 0:10:10  lr: 0.000154  loss: 5.8687 (5.8127)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 75/234]  eta: 0:09:41  lr: 0.000154  loss: 5.8692 (5.8195)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 80/234]  eta: 0:09:31  lr: 0.000154  loss: 5.7968 (5.7991)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 85/234]  eta: 0:09:03  lr: 0.000154  loss: 5.7620 (5.7823)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 90/234]  eta: 0:08:40  lr: 0.000154  loss: 5.6408 (5.7897)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [ 95/234]  eta: 0:08:24  lr: 0.000154  loss: 5.5182 (5.7872)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [100/234]  eta: 0:08:10  lr: 0.000154  loss: 5.5064 (5.7699)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [105/234]  eta: 0:07:55  lr: 0.000154  loss: 5.5422 (5.7651)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [110/234]  eta: 0:07:33  lr: 0.000154  loss: 5.5617 (5.7816)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [115/234]  eta: 0:07:10  lr: 0.000154  loss: 5.5930 (5.7818)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [120/234]  eta: 0:06:56  lr: 0.000154  loss: 5.7258 (5.7810)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [125/234]  eta: 0:06:34  lr: 0.000154  loss: 5.8302 (5.7873)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [130/234]  eta: 0:06:19  lr: 0.000154  loss: 5.7258 (5.7857)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [135/234]  eta: 0:05:58  lr: 0.000154  loss: 5.7092 (5.7799)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [140/234]  eta: 0:05:48  lr: 0.000154  loss: 5.5177 (5.7745)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [145/234]  eta: 0:05:26  lr: 0.000154  loss: 5.6197 (5.7822)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [150/234]  eta: 0:05:05  lr: 0.000154  loss: 5.7375 (5.7853)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [155/234]  eta: 0:04:45  lr: 0.000154  loss: 5.8489 (5.7879)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [160/234]  eta: 0:04:31  lr: 0.000154  loss: 5.9088 (5.7858)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [165/234]  eta: 0:04:11  lr: 0.000154  loss: 5.6877 (5.7739)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [170/234]  eta: 0:03:52  lr: 0.000154  loss: 5.8489 (5.7813)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [175/234]  eta: 0:03:35  lr: 0.000154  loss: 5.6877 (5.7805)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [180/234]  eta: 0:03:18  lr: 0.000154  loss: 5.6304 (5.7733)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [185/234]  eta: 0:02:59  lr: 0.000154  loss: 5.7869 (5.7852)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [190/234]  eta: 0:02:39  lr: 0.000154  loss: 5.7869 (5.7949)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [195/234]  eta: 0:02:21  lr: 0.000154  loss: 5.8619 (5.7943)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [200/234]  eta: 0:02:04  lr: 0.000154  loss: 5.8746 (5.7854)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [205/234]  eta: 0:01:46  lr: 0.000154  loss: 5.7375 (5.7829)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [210/234]  eta: 0:01:27  lr: 0.000154  loss: 5.6904 (5.7820)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [215/234]  eta: 0:01:08  lr: 0.000154  loss: 5.6858 (5.7851)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [220/234]  eta: 0:00:51  lr: 0.000154  loss: 5.6904 (5.7781)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [225/234]  eta: 0:00:32  lr: 0.000154  loss: 5.6674 (5.7737)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [230/234]  eta: 0:00:14  lr: 0.000154  loss: 5.6487 (5.7741)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [13]  [233/234]  eta: 0:00:03  lr: 0.000154  loss: 5.6674 (5.7705)  loss_class
Epoch: [13] Total time: 0:14:02 (3.6009 s / it)
Test Started
Test Ended
0.809872014444753
This Epochs AveragePlxError Test tensor(17.4846)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9564.0781) 547.00001
-- -- -- -- -- --
Epoch: [14]  [  0/234]  eta: 1:23:25  lr: 0.000151  loss: 5.4474 (5.4474)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [  5/234]  eta: 0:21:49  lr: 0.000151  loss: 5.4474 (5.6402)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 10/234]  eta: 0:16:36  lr: 0.000151  loss: 5.6228 (5.5949)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 15/234]  eta: 0:14:22  lr: 0.000151  loss: 5.6728 (5.6839)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 20/234]  eta: 0:14:18  lr: 0.000151  loss: 5.6728 (5.7145)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 25/234]  eta: 0:13:10  lr: 0.000151  loss: 5.6728 (5.7117)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 30/234]  eta: 0:12:52  lr: 0.000151  loss: 5.6845 (5.6990)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 35/234]  eta: 0:12:07  lr: 0.000151  loss: 5.6715 (5.7023)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 40/234]  eta: 0:12:36  lr: 0.000151  loss: 5.6845 (5.6914)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 45/234]  eta: 0:11:57  lr: 0.000151  loss: 5.9003 (5.7215)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 50/234]  eta: 0:11:19  lr: 0.000151  loss: 5.9095 (5.7369)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 55/234]  eta: 0:10:44  lr: 0.000151  loss: 5.8245 (5.7117)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 60/234]  eta: 0:10:52  lr: 0.000151  loss: 5.7187 (5.7046)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 65/234]  eta: 0:10:20  lr: 0.000151  loss: 5.6749 (5.7247)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 70/234]  eta: 0:09:53  lr: 0.000151  loss: 5.5954 (5.7123)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 75/234]  eta: 0:09:44  lr: 0.000151  loss: 5.6365 (5.7240)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 80/234]  eta: 0:09:36  lr: 0.000151  loss: 5.6365 (5.7249)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 85/234]  eta: 0:09:07  lr: 0.000151  loss: 5.6365 (5.7321)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 90/234]  eta: 0:08:42  lr: 0.000151  loss: 5.5732 (5.7176)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [ 95/234]  eta: 0:08:20  lr: 0.000151  loss: 5.7363 (5.7252)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [100/234]  eta: 0:08:12  lr: 0.000151  loss: 5.7909 (5.7416)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [105/234]  eta: 0:07:48  lr: 0.000151  loss: 5.7363 (5.7358)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [110/234]  eta: 0:07:33  lr: 0.000151  loss: 5.8207 (5.7442)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [115/234]  eta: 0:07:11  lr: 0.000151  loss: 5.8461 (5.7582)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [120/234]  eta: 0:06:56  lr: 0.000151  loss: 5.8293 (5.7654)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [125/234]  eta: 0:06:34  lr: 0.000151  loss: 5.8902 (5.7572)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [130/234]  eta: 0:06:13  lr: 0.000151  loss: 5.8293 (5.7504)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [135/234]  eta: 0:05:56  lr: 0.000151  loss: 5.7906 (5.7540)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [140/234]  eta: 0:05:41  lr: 0.000151  loss: 5.6424 (5.7556)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [145/234]  eta: 0:05:24  lr: 0.000151  loss: 5.6574 (5.7547)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [150/234]  eta: 0:05:03  lr: 0.000151  loss: 5.8313 (5.7645)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [155/234]  eta: 0:04:43  lr: 0.000151  loss: 5.8143 (5.7667)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [160/234]  eta: 0:04:27  lr: 0.000151  loss: 5.8313 (5.7602)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [165/234]  eta: 0:04:07  lr: 0.000151  loss: 5.8313 (5.7637)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [170/234]  eta: 0:03:49  lr: 0.000151  loss: 5.5145 (5.7544)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [175/234]  eta: 0:03:30  lr: 0.000151  loss: 5.5145 (5.7553)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [180/234]  eta: 0:03:15  lr: 0.000151  loss: 5.7119 (5.7595)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [185/234]  eta: 0:02:56  lr: 0.000151  loss: 5.7914 (5.7662)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [190/234]  eta: 0:02:37  lr: 0.000151  loss: 5.7637 (5.7609)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [195/234]  eta: 0:02:18  lr: 0.000151  loss: 5.7241 (5.7634)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [200/234]  eta: 0:02:02  lr: 0.000151  loss: 5.7107 (5.7549)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [205/234]  eta: 0:01:44  lr: 0.000151  loss: 5.7241 (5.7601)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [210/234]  eta: 0:01:25  lr: 0.000151  loss: 5.7671 (5.7593)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [215/234]  eta: 0:01:08  lr: 0.000151  loss: 5.6661 (5.7536)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [220/234]  eta: 0:00:50  lr: 0.000151  loss: 5.7770 (5.7554)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [225/234]  eta: 0:00:32  lr: 0.000151  loss: 5.5101 (5.7524)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [230/234]  eta: 0:00:14  lr: 0.000151  loss: 5.5055 (5.7477)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [14]  [233/234]  eta: 0:00:03  lr: 0.000151  loss: 5.4820 (5.7447)  loss_class
Epoch: [14] Total time: 0:13:47 (3.5349 s / it)
Test Started
Test Ended
0.8105646482592961
This Epochs AveragePlxError Test tensor(17.6504)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9690.0908) 549.00001
-- -- -- -- -- --
Epoch: [15]  [  0/234]  eta: 1:17:00  lr: 0.000148  loss: 5.3823 (5.3823)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [  5/234]  eta: 0:21:26  lr: 0.000148  loss: 5.6994 (5.8382)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 10/234]  eta: 0:18:29  lr: 0.000148  loss: 5.9076 (5.9162)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 15/234]  eta: 0:15:26  lr: 0.000148  loss: 5.7740 (5.8117)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 20/234]  eta: 0:15:10  lr: 0.000148  loss: 5.7740 (5.7583)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 25/234]  eta: 0:13:45  lr: 0.000148  loss: 5.7592 (5.7525)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 30/234]  eta: 0:12:41  lr: 0.000148  loss: 5.5313 (5.7053)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 35/234]  eta: 0:11:55  lr: 0.000148  loss: 5.5313 (5.7102)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 40/234]  eta: 0:12:20  lr: 0.000148  loss: 5.4411 (5.6600)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 45/234]  eta: 0:12:06  lr: 0.000148  loss: 5.5192 (5.6926)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 50/234]  eta: 0:11:27  lr: 0.000148  loss: 5.6504 (5.7151)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 55/234]  eta: 0:10:53  lr: 0.000148  loss: 5.6504 (5.7266)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 60/234]  eta: 0:10:46  lr: 0.000148  loss: 5.8102 (5.7311)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 65/234]  eta: 0:10:14  lr: 0.000148  loss: 5.7114 (5.7373)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 70/234]  eta: 0:09:43  lr: 0.000148  loss: 5.7114 (5.7455)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 75/234]  eta: 0:09:28  lr: 0.000148  loss: 5.6437 (5.7363)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 80/234]  eta: 0:09:31  lr: 0.000148  loss: 5.6437 (5.7323)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 85/234]  eta: 0:09:05  lr: 0.000148  loss: 5.6099 (5.7247)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 90/234]  eta: 0:08:40  lr: 0.000148  loss: 5.6129 (5.7363)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [ 95/234]  eta: 0:08:16  lr: 0.000148  loss: 5.7125 (5.7352)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [100/234]  eta: 0:08:04  lr: 0.000148  loss: 5.7117 (5.7281)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [105/234]  eta: 0:07:40  lr: 0.000148  loss: 5.7313 (5.7390)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [110/234]  eta: 0:07:23  lr: 0.000148  loss: 5.6233 (5.7308)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [115/234]  eta: 0:07:08  lr: 0.000148  loss: 5.7117 (5.7376)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [120/234]  eta: 0:06:54  lr: 0.000148  loss: 5.7313 (5.7429)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [125/234]  eta: 0:06:33  lr: 0.000148  loss: 5.6580 (5.7370)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [130/234]  eta: 0:06:11  lr: 0.000148  loss: 5.7350 (5.7493)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [135/234]  eta: 0:05:50  lr: 0.000148  loss: 5.7401 (5.7664)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [140/234]  eta: 0:05:38  lr: 0.000148  loss: 5.8537 (5.7737)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [145/234]  eta: 0:05:18  lr: 0.000148  loss: 5.9304 (5.7807)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [150/234]  eta: 0:05:01  lr: 0.000148  loss: 5.9166 (5.7829)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [155/234]  eta: 0:04:41  lr: 0.000148  loss: 5.9166 (5.7867)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [160/234]  eta: 0:04:25  lr: 0.000148  loss: 5.8107 (5.7890)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [165/234]  eta: 0:04:06  lr: 0.000148  loss: 5.7094 (5.7820)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [170/234]  eta: 0:03:46  lr: 0.000148  loss: 5.7072 (5.7824)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [175/234]  eta: 0:03:27  lr: 0.000148  loss: 5.7050 (5.7824)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [180/234]  eta: 0:03:12  lr: 0.000148  loss: 5.6800 (5.7787)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [185/234]  eta: 0:02:56  lr: 0.000148  loss: 5.6884 (5.7743)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [190/234]  eta: 0:02:37  lr: 0.000148  loss: 5.6800 (5.7725)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [195/234]  eta: 0:02:18  lr: 0.000148  loss: 5.6905 (5.7741)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [200/234]  eta: 0:02:01  lr: 0.000148  loss: 5.6905 (5.7730)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [205/234]  eta: 0:01:43  lr: 0.000148  loss: 5.7013 (5.7699)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [210/234]  eta: 0:01:24  lr: 0.000148  loss: 5.7013 (5.7683)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [215/234]  eta: 0:01:07  lr: 0.000148  loss: 5.5228 (5.7643)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [220/234]  eta: 0:00:50  lr: 0.000148  loss: 5.6450 (5.7702)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [225/234]  eta: 0:00:32  lr: 0.000148  loss: 5.6511 (5.7695)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [230/234]  eta: 0:00:14  lr: 0.000148  loss: 5.6450 (5.7675)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [15]  [233/234]  eta: 0:00:03  lr: 0.000148  loss: 5.6450 (5.7685)  loss_class
Epoch: [15] Total time: 0:13:43 (3.5213 s / it)
Test Started
Test Ended
0.8161764555849916
This Epochs AveragePlxError Test tensor(17.0316)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9265.1758) 544.00001
-- -- -- -- -- --
Epoch: [16]  [  0/234]  eta: 0:52:54  lr: 0.000145  loss: 5.1714 (5.1714)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [  5/234]  eta: 0:18:09  lr: 0.000145  loss: 5.6325 (5.7246)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 10/234]  eta: 0:15:52  lr: 0.000145  loss: 5.6346 (5.6907)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 15/234]  eta: 0:13:55  lr: 0.000145  loss: 5.6325 (5.6994)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 20/234]  eta: 0:15:11  lr: 0.000145  loss: 5.6325 (5.6713)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 25/234]  eta: 0:13:49  lr: 0.000145  loss: 5.7264 (5.7042)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 30/234]  eta: 0:12:49  lr: 0.000145  loss: 5.7468 (5.7302)  loss_class
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 35/234]  eta: 0:11:59  lr: 0.000145  loss: 5.7214 (5.7297)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 40/234]  eta: 0:12:01  lr: 0.000145  loss: 5.7214 (5.7307)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 45/234]  eta: 0:11:46  lr: 0.000145  loss: 5.7214 (5.7604)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 50/234]  eta: 0:11:12  lr: 0.000145  loss: 5.6639 (5.7353)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 55/234]  eta: 0:10:56  lr: 0.000145  loss: 5.6760 (5.7223)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 60/234]  eta: 0:10:47  lr: 0.000145  loss: 5.6571 (5.6766)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 65/234]  eta: 0:10:17  lr: 0.000145  loss: 5.6639 (5.6841)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 70/234]  eta: 0:09:49  lr: 0.000145  loss: 5.6873 (5.6902)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 75/234]  eta: 0:09:23  lr: 0.000145  loss: 5.6809 (5.6944)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 80/234]  eta: 0:09:19  lr: 0.000145  loss: 5.7465 (5.7018)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 85/234]  eta: 0:08:55  lr: 0.000145  loss: 5.7068 (5.7042)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 90/234]  eta: 0:08:40  lr: 0.000145  loss: 5.7505 (5.7132)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [ 95/234]  eta: 0:08:16  lr: 0.000145  loss: 5.8782 (5.7237)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [100/234]  eta: 0:08:03  lr: 0.000145  loss: 5.9164 (5.7292)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [105/234]  eta: 0:07:42  lr: 0.000145  loss: 5.8782 (5.7272)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [110/234]  eta: 0:07:19  lr: 0.000145  loss: 5.7348 (5.7210)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [115/234]  eta: 0:06:57  lr: 0.000145  loss: 5.6112 (5.7138)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [120/234]  eta: 0:06:46  lr: 0.000145  loss: 5.6038 (5.7134)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [125/234]  eta: 0:06:32  lr: 0.000145  loss: 5.6038 (5.7163)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [130/234]  eta: 0:06:10  lr: 0.000145  loss: 5.6250 (5.7218)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [135/234]  eta: 0:05:51  lr: 0.000145  loss: 5.6707 (5.7173)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [140/234]  eta: 0:05:34  lr: 0.000145  loss: 5.6707 (5.7124)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [145/234]  eta: 0:05:16  lr: 0.000145  loss: 5.6259 (5.7045)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [150/234]  eta: 0:04:59  lr: 0.000145  loss: 5.5541 (5.6975)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [155/234]  eta: 0:04:39  lr: 0.000145  loss: 5.6259 (5.6994)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [160/234]  eta: 0:04:25  lr: 0.000145  loss: 5.6259 (5.7017)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [165/234]  eta: 0:04:07  lr: 0.000145  loss: 5.6028 (5.6986)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [170/234]  eta: 0:03:48  lr: 0.000145  loss: 5.5956 (5.6937)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [175/234]  eta: 0:03:28  lr: 0.000145  loss: 5.5806 (5.6973)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [180/234]  eta: 0:03:11  lr: 0.000145  loss: 5.5806 (5.6972)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [185/234]  eta: 0:02:54  lr: 0.000145  loss: 5.5956 (5.6989)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [190/234]  eta: 0:02:36  lr: 0.000145  loss: 5.6407 (5.6939)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [195/234]  eta: 0:02:18  lr: 0.000145  loss: 5.5750 (5.6905)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [200/234]  eta: 0:02:01  lr: 0.000145  loss: 5.4866 (5.6857)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [205/234]  eta: 0:01:43  lr: 0.000145  loss: 5.4983 (5.6915)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [210/234]  eta: 0:01:24  lr: 0.000145  loss: 5.7227 (5.6928)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [215/234]  eta: 0:01:06  lr: 0.000145  loss: 5.7227 (5.6939)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [220/234]  eta: 0:00:49  lr: 0.000145  loss: 5.7028 (5.6867)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [225/234]  eta: 0:00:31  lr: 0.000145  loss: 5.7028 (5.6899)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [230/234]  eta: 0:00:14  lr: 0.000145  loss: 5.7028 (5.6895)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [16]  [233/234]  eta: 0:00:03  lr: 0.000145  loss: 5.6070 (5.6911)  loss_class
Epoch: [16] Total time: 0:13:45 (3.5261 s / it)
Test Started
Test Ended
0.7802197659300409
This Epochs AveragePlxError Test tensor(18.2328)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9955.0947) 546.00001
-- -- -- -- -- --
Epoch: [17]  [  0/234]  eta: 1:01:02  lr: 0.000142  loss: 5.8380 (5.8380)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [  5/234]  eta: 0:18:35  lr: 0.000142  loss: 5.5095 (5.4740)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 10/234]  eta: 0:14:36  lr: 0.000142  loss: 5.5753 (5.4922)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 15/234]  eta: 0:12:43  lr: 0.000142  loss: 5.5611 (5.4716)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 20/234]  eta: 0:13:50  lr: 0.000142  loss: 5.5598 (5.4779)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 25/234]  eta: 0:13:25  lr: 0.000142  loss: 5.5611 (5.5216)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 30/234]  eta: 0:12:29  lr: 0.000142  loss: 5.5611 (5.5198)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 35/234]  eta: 0:11:45  lr: 0.000142  loss: 5.6281 (5.5425)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 40/234]  eta: 0:11:54  lr: 0.000142  loss: 5.6904 (5.5682)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 45/234]  eta: 0:11:15  lr: 0.000142  loss: 5.6014 (5.5320)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 50/234]  eta: 0:10:41  lr: 0.000142  loss: 5.5489 (5.5315)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 55/234]  eta: 0:10:29  lr: 0.000142  loss: 5.6014 (5.5546)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 60/234]  eta: 0:10:43  lr: 0.000142  loss: 5.5489 (5.5608)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 65/234]  eta: 0:10:11  lr: 0.000142  loss: 5.6889 (5.5754)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 70/234]  eta: 0:09:42  lr: 0.000142  loss: 5.7397 (5.5855)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 75/234]  eta: 0:09:22  lr: 0.000142  loss: 5.5192 (5.5690)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 80/234]  eta: 0:09:15  lr: 0.000142  loss: 5.5192 (5.5706)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 85/234]  eta: 0:08:50  lr: 0.000142  loss: 5.5167 (5.5800)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 90/234]  eta: 0:08:32  lr: 0.000142  loss: 5.5192 (5.5939)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [ 95/234]  eta: 0:08:08  lr: 0.000142  loss: 5.7632 (5.6040)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [100/234]  eta: 0:08:01  lr: 0.000142  loss: 5.8019 (5.6048)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [105/234]  eta: 0:07:39  lr: 0.000142  loss: 5.6880 (5.6063)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [110/234]  eta: 0:07:20  lr: 0.000142  loss: 5.6133 (5.6128)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [115/234]  eta: 0:06:58  lr: 0.000142  loss: 5.5781 (5.6171)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [120/234]  eta: 0:06:44  lr: 0.000142  loss: 5.5585 (5.6176)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [125/234]  eta: 0:06:26  lr: 0.000142  loss: 5.7332 (5.6231)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [130/234]  eta: 0:06:05  lr: 0.000142  loss: 5.8076 (5.6363)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [135/234]  eta: 0:05:49  lr: 0.000142  loss: 5.6015 (5.6251)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [140/234]  eta: 0:05:34  lr: 0.000142  loss: 5.6045 (5.6236)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [145/234]  eta: 0:05:15  lr: 0.000142  loss: 5.6015 (5.6273)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [150/234]  eta: 0:04:55  lr: 0.000142  loss: 5.6045 (5.6372)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [155/234]  eta: 0:04:35  lr: 0.000142  loss: 5.7019 (5.6309)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [160/234]  eta: 0:04:21  lr: 0.000142  loss: 5.7739 (5.6384)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [165/234]  eta: 0:04:02  lr: 0.000142  loss: 5.7739 (5.6443)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [170/234]  eta: 0:03:45  lr: 0.000142  loss: 5.6963 (5.6424)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [175/234]  eta: 0:03:26  lr: 0.000142  loss: 5.7180 (5.6436)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [180/234]  eta: 0:03:10  lr: 0.000142  loss: 5.6802 (5.6396)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [185/234]  eta: 0:02:52  lr: 0.000142  loss: 5.6566 (5.6342)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [190/234]  eta: 0:02:33  lr: 0.000142  loss: 5.6802 (5.6433)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [195/234]  eta: 0:02:16  lr: 0.000142  loss: 5.6984 (5.6466)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [200/234]  eta: 0:01:59  lr: 0.000142  loss: 5.6984 (5.6431)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [205/234]  eta: 0:01:42  lr: 0.000142  loss: 5.6984 (5.6381)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [210/234]  eta: 0:01:24  lr: 0.000142  loss: 5.6766 (5.6441)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [215/234]  eta: 0:01:06  lr: 0.000142  loss: 5.6766 (5.6540)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [220/234]  eta: 0:00:49  lr: 0.000142  loss: 5.7928 (5.6599)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [225/234]  eta: 0:00:31  lr: 0.000142  loss: 5.7721 (5.6593)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [230/234]  eta: 0:00:13  lr: 0.000142  loss: 5.8116 (5.6658)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [17]  [233/234]  eta: 0:00:03  lr: 0.000142  loss: 5.7928 (5.6620)  loss_class
Epoch: [17] Total time: 0:13:31 (3.4692 s / it)
Test Started
Test Ended
0.7919707884676864
This Epochs AveragePlxError Test tensor(17.7857)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9746.5557) 548.00001
-- -- -- -- -- --
Epoch: [18]  [  0/234]  eta: 1:18:33  lr: 0.000139  loss: 6.1857 (6.1857)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [  5/234]  eta: 0:21:27  lr: 0.000139  loss: 5.6855 (5.7910)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 10/234]  eta: 0:16:06  lr: 0.000139  loss: 5.6671 (5.6365)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 15/234]  eta: 0:14:17  lr: 0.000139  loss: 5.5242 (5.5855)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 20/234]  eta: 0:14:22  lr: 0.000139  loss: 5.5133 (5.5710)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 25/234]  eta: 0:13:15  lr: 0.000139  loss: 5.5133 (5.6028)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 30/234]  eta: 0:12:43  lr: 0.000139  loss: 5.5133 (5.6101)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 35/234]  eta: 0:11:54  lr: 0.000139  loss: 5.5915 (5.6215)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 40/234]  eta: 0:12:26  lr: 0.000139  loss: 5.7989 (5.6445)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 45/234]  eta: 0:11:43  lr: 0.000139  loss: 5.5915 (5.6257)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 50/234]  eta: 0:11:12  lr: 0.000139  loss: 5.6578 (5.6366)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 55/234]  eta: 0:10:38  lr: 0.000139  loss: 5.6578 (5.6519)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 60/234]  eta: 0:10:31  lr: 0.000139  loss: 5.5669 (5.6453)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 65/234]  eta: 0:10:12  lr: 0.000139  loss: 5.5711 (5.6475)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 70/234]  eta: 0:09:43  lr: 0.000139  loss: 5.5164 (5.6254)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 75/234]  eta: 0:09:30  lr: 0.000139  loss: 5.5144 (5.6203)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 80/234]  eta: 0:09:18  lr: 0.000139  loss: 5.5144 (5.6412)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 85/234]  eta: 0:08:55  lr: 0.000139  loss: 5.5144 (5.6404)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 90/234]  eta: 0:08:31  lr: 0.000139  loss: 5.7898 (5.6634)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [ 95/234]  eta: 0:08:08  lr: 0.000139  loss: 5.8091 (5.6572)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [100/234]  eta: 0:07:59  lr: 0.000139  loss: 5.5639 (5.6509)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [105/234]  eta: 0:07:35  lr: 0.000139  loss: 5.4820 (5.6306)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [110/234]  eta: 0:07:22  lr: 0.000139  loss: 5.5005 (5.6355)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [115/234]  eta: 0:07:00  lr: 0.000139  loss: 5.4820 (5.6312)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [120/234]  eta: 0:06:47  lr: 0.000139  loss: 5.5005 (5.6382)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [125/234]  eta: 0:06:26  lr: 0.000139  loss: 5.5364 (5.6328)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [130/234]  eta: 0:06:06  lr: 0.000139  loss: 5.5983 (5.6354)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [135/234]  eta: 0:05:48  lr: 0.000139  loss: 5.7613 (5.6473)  loss_class
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [140/234]  eta: 0:05:37  lr: 0.000139  loss: 5.6882 (5.6417)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [145/234]  eta: 0:05:17  lr: 0.000139  loss: 5.6882 (5.6375)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [150/234]  eta: 0:04:57  lr: 0.000139  loss: 5.6882 (5.6385)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [155/234]  eta: 0:04:37  lr: 0.000139  loss: 5.6146 (5.6338)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [160/234]  eta: 0:04:22  lr: 0.000139  loss: 5.6438 (5.6399)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [165/234]  eta: 0:04:03  lr: 0.000139  loss: 5.6356 (5.6389)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [170/234]  eta: 0:03:45  lr: 0.000139  loss: 5.6356 (5.6415)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [175/234]  eta: 0:03:26  lr: 0.000139  loss: 5.7132 (5.6431)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [180/234]  eta: 0:03:12  lr: 0.000139  loss: 5.6340 (5.6401)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [185/234]  eta: 0:02:53  lr: 0.000139  loss: 5.7033 (5.6467)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [190/234]  eta: 0:02:34  lr: 0.000139  loss: 5.6948 (5.6453)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [195/234]  eta: 0:02:16  lr: 0.000139  loss: 5.6569 (5.6433)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [200/234]  eta: 0:01:59  lr: 0.000139  loss: 5.6569 (5.6393)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [205/234]  eta: 0:01:42  lr: 0.000139  loss: 5.6251 (5.6426)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [210/234]  eta: 0:01:24  lr: 0.000139  loss: 5.5718 (5.6417)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [215/234]  eta: 0:01:06  lr: 0.000139  loss: 5.6155 (5.6430)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [220/234]  eta: 0:00:49  lr: 0.000139  loss: 5.6463 (5.6442)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [225/234]  eta: 0:00:31  lr: 0.000139  loss: 5.6147 (5.6373)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [230/234]  eta: 0:00:14  lr: 0.000139  loss: 5.5828 (5.6387)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [18]  [233/234]  eta: 0:00:03  lr: 0.000139  loss: 5.5349 (5.6375)  loss_class
Epoch: [18] Total time: 0:13:33 (3.4765 s / it)
Test Started
Test Ended
0.819012782102143
This Epochs AveragePlxError Test tensor(17.4602)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9550.7070) 547.00001
-- -- -- -- -- --
Epoch: [19]  [  0/234]  eta: 1:09:29  lr: 0.000136  loss: 5.4474 (5.4474)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [  5/234]  eta: 0:19:42  lr: 0.000136  loss: 5.3933 (5.5141)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 10/234]  eta: 0:14:58  lr: 0.000136  loss: 5.5887 (5.6720)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 15/234]  eta: 0:14:21  lr: 0.000136  loss: 5.5338 (5.6250)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 20/234]  eta: 0:14:25  lr: 0.000136  loss: 5.5266 (5.5920)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 25/234]  eta: 0:13:31  lr: 0.000136  loss: 5.5622 (5.6040)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 30/234]  eta: 0:12:29  lr: 0.000136  loss: 5.5266 (5.6152)  loss_class
-- -- -- -- -- --
```

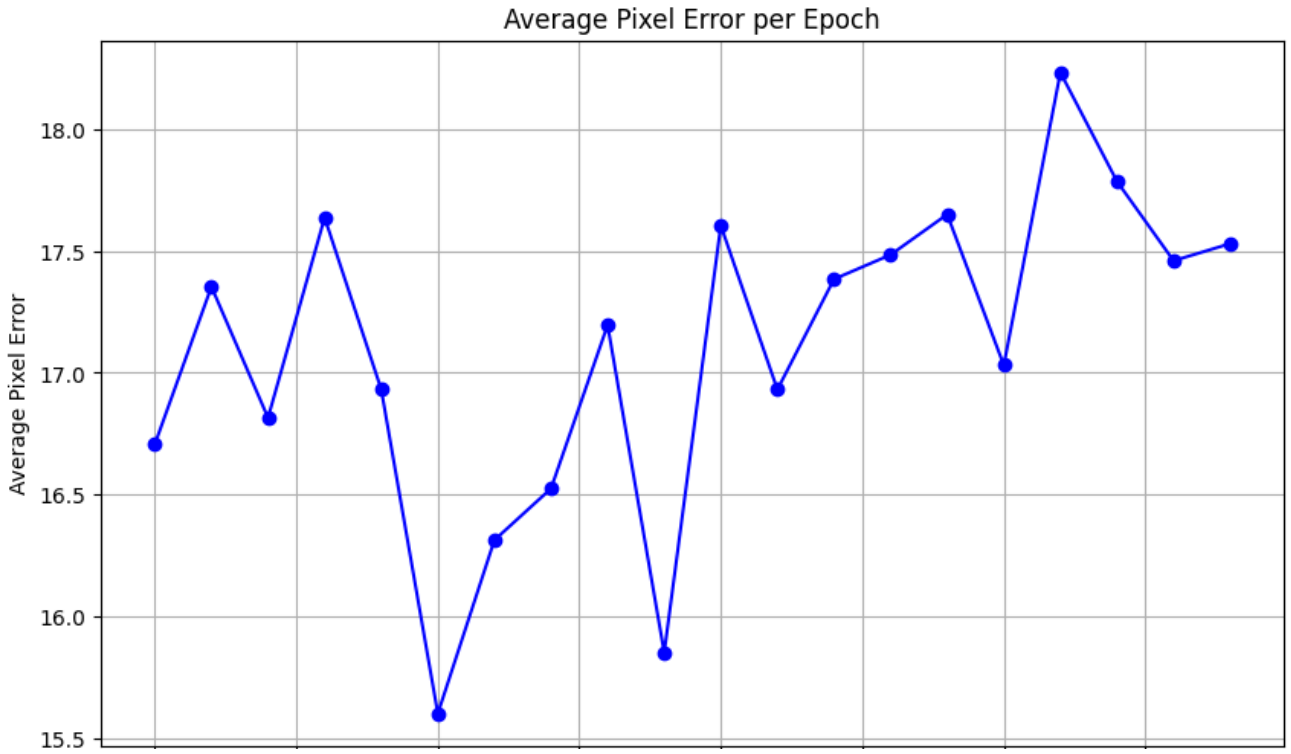
```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 35/234]  eta: 0:11:46  lr: 0.000136  loss: 5.5266 (5.6089)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 40/234]  eta: 0:12:01  lr: 0.000136  loss: 5.6024 (5.6368)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 45/234]  eta: 0:11:19  lr: 0.000136  loss: 5.6068 (5.6390)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 50/234]  eta: 0:11:08  lr: 0.000136  loss: 5.7185 (5.6513)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 55/234]  eta: 0:10:36  lr: 0.000136  loss: 5.7697 (5.6489)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 60/234]  eta: 0:10:35  lr: 0.000136  loss: 5.6660 (5.6495)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 65/234]  eta: 0:10:05  lr: 0.000136  loss: 5.4614 (5.6389)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 70/234]  eta: 0:09:37  lr: 0.000136  loss: 5.4614 (5.6592)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 75/234]  eta: 0:09:20  lr: 0.000136  loss: 5.6045 (5.6701)  loss_class
-- -- -- -- -- --
```

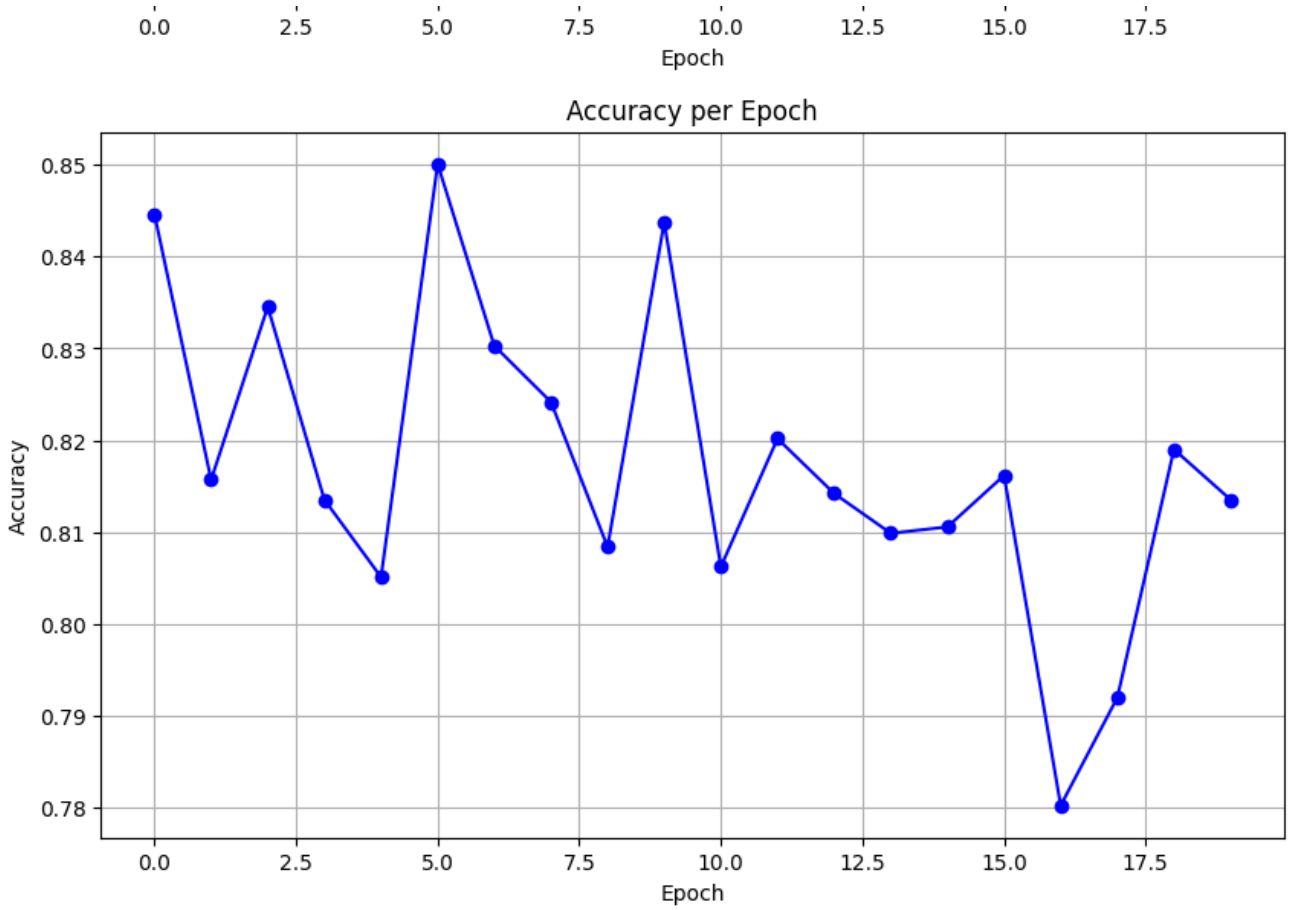
```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 80/234]  eta: 0:09:12  lr: 0.000136  loss: 5.7559 (5.6869)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 85/234]  eta: 0:08:57  lr: 0.000136  loss: 5.7676 (5.6921)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 90/234]  eta: 0:08:32  lr: 0.000136  loss: 5.7559 (5.6994)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [ 95/234]  eta: 0:08:08  lr: 0.000136  loss: 5.7449 (5.6913)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [100/234]  eta: 0:07:58  lr: 0.000136  loss: 5.7429 (5.6862)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [105/234]  eta: 0:07:34  lr: 0.000136  loss: 5.7429 (5.6785)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [110/234]  eta: 0:07:17  lr: 0.000136  loss: 5.7429 (5.6731)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [115/234]  eta: 0:06:56  lr: 0.000136  loss: 5.7453 (5.6760)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [120/234]  eta: 0:06:48  lr: 0.000136  loss: 5.7453 (5.6724)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [125/234]  eta: 0:06:27  lr: 0.000136  loss: 5.5803 (5.6601)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [130/234]  eta: 0:06:07  lr: 0.000136  loss: 5.4528 (5.6495)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [135/234]  eta: 0:05:46  lr: 0.000136  loss: 5.3643 (5.6375)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [140/234]  eta: 0:05:32  lr: 0.000136  loss: 5.3986 (5.6337)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [145/234]  eta: 0:05:14  lr: 0.000136  loss: 5.4044 (5.6312)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [150/234]  eta: 0:04:54  lr: 0.000136  loss: 5.4794 (5.6306)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [155/234]  eta: 0:04:35  lr: 0.000136  loss: 5.6059 (5.6391)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [160/234]  eta: 0:04:22  lr: 0.000136  loss: 5.6040 (5.6302)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [165/234]  eta: 0:04:04  lr: 0.000136  loss: 5.6059 (5.6386)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [170/234]  eta: 0:03:45  lr: 0.000136  loss: 5.6040 (5.6321)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [175/234]  eta: 0:03:26  lr: 0.000136  loss: 5.4483 (5.6303)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [180/234]  eta: 0:03:10  lr: 0.000136  loss: 5.5289 (5.6324)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [185/234]  eta: 0:02:52  lr: 0.000136  loss: 5.4483 (5.6261)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [190/234]  eta: 0:02:33  lr: 0.000136  loss: 5.5219 (5.6249)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [195/234]  eta: 0:02:16  lr: 0.000136  loss: 5.5395 (5.6240)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [200/234]  eta: 0:02:00  lr: 0.000136  loss: 5.5395 (5.6249)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [205/234]  eta: 0:01:41  lr: 0.000136  loss: 5.5395 (5.6192)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [210/234]  eta: 0:01:23  lr: 0.000136  loss: 5.6379 (5.6207)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
```

```
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [215/234]  eta: 0:01:06  lr: 0.000136  loss: 5.6809 (5.6241)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [220/234]  eta: 0:00:49  lr: 0.000136  loss: 5.6941 (5.6199)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [225/234]  eta: 0:00:31  lr: 0.000136  loss: 5.5794 (5.6183)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [230/234]  eta: 0:00:13  lr: 0.000136  loss: 5.5644 (5.6164)  loss_class
-- -- -- -- -- --
-- -- -- -- -- --
-- -- -- -- -- --
Epoch: [19]  [233/234]  eta: 0:00:03  lr: 0.000136  loss: 5.4644 (5.6178)  loss_class
Epoch: [19] Total time: 0:13:32 (3.4720 s / it)
Test Started
Test Ended
0.8135283215077089
This Epochs AveragePlxError Test tensor(17.5303)
Best Epochs AveragePlxError Test tensor(15.6015)
tensor(9589.0674) 547.00001
```






```
[tensor(16.7072), tensor(17.3536), tensor(16.8157), tensor(17.6373), tensor(16.9324),
[0.844606931542835, 0.815693415772018, 0.8345454393719012, 0.8135283215077089, 0.8051
```

```
# torch.save(model.state_dict(), "/content/drive/MyDrive/shufflenet_v2_x1_0NotGuided50Off


model.to(device)

     KeypointRCNN(
       (transform): GeneralizedRCNNTransform(
           Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
           Resize(min_size=(640, 672, 704, 736, 768, 800), max_size=1333,
     mode='bilinear')
       )
       (backbone): BackboneWithFPN(
         (body): IntermediateLayerGetter(
           (0): Conv2dNormActivation(
             (0): Conv2d(3, 16, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
     bias=False)
             (1): BatchNorm2d(16, eps=0.001, momentum=0.01, affine=True,
     track_running_stats=True)
             (2): Hardswish()
           )
           (1): InvertedResidual(
             (block): Sequential(
               (0): Conv2dNormActivation(
                 (0): Conv2d(16, 16, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
     groups=16, bias=False)
                 (1): BatchNorm2d(16, eps=0.001, momentum=0.01, affine=True,
     track_running_stats=True)
                 (2): ReLU(inplace=True)
               )
               (1): SqueezeExcitation(
                 (avgpool): AdaptiveAvgPool2d(output_size=1)
                 (fc1): Conv2d(16, 8, kernel_size=(1, 1), stride=(1, 1))
                 (fc2): Conv2d(8, 16, kernel_size=(1, 1), stride=(1, 1))
                 (activation): ReLU()
                 (scale_activation): Hardsigmoid()
               )
               (2): Conv2dNormActivation(
                 (0): Conv2d(16, 16, kernel_size=(1, 1), stride=(1, 1), bias=False)
                 (1): BatchNorm2d(16, eps=0.001, momentum=0.01, affine=True,
     track_running_stats=True)
               )
             )
           )
           (2): InvertedResidual(
             (block): Sequential(
               (0): Conv2dNormActivation(
                 (0): Conv2d(16, 72, kernel_size=(1, 1), stride=(1, 1), bias=False)
                 (1): BatchNorm2d(72, eps=0.001, momentum=0.01, affine=True,
     track_running_stats=True)
                 (2): ReLU(inplace=True)
               )
               (1): Conv2dNormActivation(
                 (0): Conv2d(72, 72, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
     groups=72, bias=False)
                 (1): BatchNorm2d(72, eps=0.001, momentum=0.01, affine=True
```

```
            (1): BatchNorm2d(72, eps=0.001, momentum=0.01, affine=True,
      track_running_stats=True)
            (2): ReLU(inplace=True)
          )
          (2): Conv2dNormActivation(
            (0): Conv2d(72, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
            (1): BatchNorm2d(24, eps=0.001, momentum=0.01, affine=True,
      track_running_stats=True)
          )
```

```
testSetTruePreds,acceptablePredictionsInDataset,sumOfPxlwiseError = testFN(5,20,True,True)

print(testSetTruePreds/acceptablePredictionsInDataset)
print(sumOfPxlwiseError/acceptablePredictionsInDataset)
print(sumOfPxlwiseError,acceptablePredictionsInDataset)
```

```
      Test Started
      tensor(18.1255)
      tensor(14.0706)
      tensor(3.7723)
      tensor(13.8408)
      tensor(9.4489)
      tensor(11.5952)
      tensor(11.0096)
      tensor(17.2903)<ipython-input-42-3ac5d076514b>:546: UserWarning: To copy construct fr
        keypoint_proposals_withOffest[i][counter] = torch.tensor(updatedBox)

      tensor(8.4680)
      tensor(21.6151)
      tensor(18.3199)
      tensor(2.9803)
      tensor(24.6749)
      tensor(10.9944)
      tensor(10.2718)
      tensor(10.7269)
      tensor(7.4761)
      tensor(2.4357)
      tensor(5.8351)
      tensor(5.6572)
      tensor(12.3330)
      tensor(9.6466)
      tensor(10.6876)
      tensor(17.0244)
      tensor(20.5403)
      tensor(9.4082)
      tensor(14.2317)
      tensor(7.4239)
      tensor(17.2630)
      tensor(22.9537)
      tensor(14.4341)
      tensor(17.9535)
      tensor(12.1284)
      tensor(18.6567)
      tensor(12.4949)
```

```
tensor(8.3522)
tensor(7.0461)
tensor(9.6981)
tensor(6.0733)
tensor(10.7294)
tensor(8.0262)
tensor(23.0807)
tensor(3.8448)
tensor(2.7747)
tensor(6.4510)
tensor(9.1537)
tensor(21.7351)
tensor(21.9570)
tensor(8.1769)
tensor(7.5369)
tensor(11.4025)
tensor(20.8165)
tensor(24.2826)
tensor(21.9340)
tensor(10.1053)
tensor(12.9731)
tensor(4.9849)
tensor(16.8031)
tensor(15.6963)
tensor(14.9608)
tensor(8.9110)
tensor(5.0034)
tensor(2.4129)
tensor(8.6495)
tensor(16.4146)
tensor(7.7290)
tensor(16.0545)
tensor(16.0774)
tensor(1.5074)
tensor(10.1153)
tensor(12.2717)
tensor(19.8943)
tensor(18.8704)
tensor(14.8972)
tensor(13.3270)
tensor(3.9079)
tensor(7.6911)
tensor(6.0889)
tensor(7.4466)
tensor(2.8204)
tensor(8.6872)
tensor(6.7245)
tensor(17.0036)
tensor(12.7906)
tensor(1.9180)
tensor(5.6042)
tensor(12.6611)
tensor(11.8518)
tensor(3.1920)
tensor(6.7016)
```

```
tensor(6.7016)
tensor(23.7375)
tensor(14.6564)
tensor(8.0361)
tensor(12.5379)
tensor(6.8156)
tensor(10.0589)
tensor(16.6863)
tensor(15.0268)
tensor(21.7770)
tensor(20.1991)
tensor(13.2446)
tensor(15.6040)
tensor(8.4883)
tensor(14.9512)
tensor(23.7274)
tensor(3.3975)
tensor(9.0097)
tensor(19.8957)
tensor(14.8276)
tensor(5.4795)
tensor(7.3648)
tensor(12.7232)
tensor(7.5027)
tensor(4.0021)
tensor(12.0351)
tensor(20.7787)
tensor(16.9563)
tensor(14.7568)
tensor(1.5842)
tensor(5.0468)
tensor(22.8125)
tensor(8.7173)
tensor(5.1556)
tensor(9.7050)
tensor(15.2894)
tensor(16.9752)
tensor(4.8109)
tensor(21.9009)
tensor(11.2848)
tensor(20.5469)
tensor(17.3049)
tensor(6.0050)
tensor(7.5782)
tensor(12.5482)
tensor(7.2380)
tensor(5.6076)
tensor(3.2165)
tensor(11.8286)
tensor(13.7569)
tensor(24.8587)
tensor(20.1619)
tensor(20.4816)
tensor(16.0867)
tensor(23.0560)
```

```
tensor(22.3773)
tensor(4.6299)
tensor(14.0105)
tensor(9.2580)
tensor(4.5421)
tensor(13.4676)
tensor(11.3822)
tensor(11.4138)
tensor(14.0954)
tensor(11.3826)
tensor(5.7399)
tensor(7.5857)
tensor(3.9166)
tensor(10.6398)
tensor(21.1840)
tensor(8.0180)
tensor(18.9943)
tensor(6.0561)
tensor(6.7656)
tensor(20.5909)
tensor(3.2881)
tensor(0.1309)
tensor(7.6835)
tensor(14.7030)
tensor(6.0896)
tensor(10.8756)
tensor(22.6700)
tensor(2.7375)
tensor(3.7075)
tensor(17.6478)
tensor(6.4571)
tensor(14.3261)
tensor(0.8434)
tensor(11.3387)
tensor(1.4114)
tensor(13.2806)
tensor(6.5882)
tensor(15.9104)
tensor(10.3782)
tensor(21.3105)
tensor(3.1959)
tensor(2.2797)
tensor(3.3459)
tensor(16.5713)
tensor(23.1617)
tensor(16.4578)
tensor(13.6735)
tensor(16.9417)
tensor(18.4404)
tensor(9.7226)
tensor(3.2860)
tensor(15.6066)
tensor(7.3685)
tensor(12.8907)
```

```
tensor(7.3227)
tensor(23.2624)
tensor(21.6186)
tensor(3.6676)
tensor(15.3215)
tensor(12.7728)
tensor(2.1426)
tensor(16.2659)
tensor(12.8507)
tensor(3.6844)
tensor(3.5467)
tensor(9.2762)
tensor(5.9080)
tensor(24.4336)
tensor(6.1547)
tensor(18.0235)
tensor(24.6986)
tensor(20.1373)
tensor(18.3385)
tensor(9.9580)
tensor(11.1661)
tensor(8.6470)
tensor(17.4822)
tensor(12.0212)
tensor(4.4368)
tensor(7.5721)
tensor(13.7749)
tensor(7.2027)
tensor(15.1578)
tensor(2.9336)
tensor(3.2002)
tensor(18.0793)
tensor(4.2332)
tensor(11.1696)
tensor(5.0080)
tensor(20.1982)
tensor(15.8953)
tensor(17.9963)
tensor(5.7850)
tensor(12.9623)
tensor(12.0613)
tensor(7.0673)
tensor(17.3726)
tensor(23.8194)
tensor(4.8494)
tensor(7.1113)
tensor(3.8198)
tensor(5.8833)
tensor(13.8720)
tensor(14.9412)
tensor(3.9325)
tensor(10.2096)
tensor(9.3979)
tensor(9.4401)
```

```
tensor(10.3353)
tensor(10.8946)
tensor(5.4024)
tensor(10.8237)
tensor(4.8635)
tensor(1.5734)
tensor(7.2217)
tensor(8.9931)
tensor(2.6632)
tensor(16.6793)
tensor(7.7907)
tensor(9.0985)
tensor(11.3185)
tensor(17.5198)
tensor(2.2985)
tensor(13.2736)
tensor(11.5831)
tensor(8.3170)
tensor(11.8888)
tensor(5.0240)
tensor(6.1980)
tensor(5.1847)
tensor(7.6868)
tensor(21.9215)
tensor(10.8235)
tensor(7.5327)
tensor(8.1592)
tensor(8.1099)
tensor(3.6468)
tensor(3.5444)
tensor(7.6384)
tensor(14.8598)
tensor(17.4046)
tensor(6.9634)
tensor(9.1821)
tensor(13.9145)
tensor(17.2182)
tensor(15.8389)
tensor(16.1610)
tensor(10.0617)
tensor(12.6536)
tensor(4.9175)
tensor(5.0040)
tensor(10.1820)
tensor(11.4646)
tensor(19.8035)
tensor(16.4801)
tensor(10.1660)
tensor(14.1038)
tensor(6.0742)
tensor(18.7621)
tensor(2.5943)
tensor(7.2077)
tensor(16.9814)
```

```
tensor(11.6931)
tensor(8.7988)
tensor(3.6219)
tensor(17.9260)
tensor(20.3748)
tensor(5.1040)
tensor(2.8531)
tensor(5.2431)
tensor(15.0581)
tensor(19.7587)
tensor(20.5209)
tensor(13.6625)
tensor(22.2875)
tensor(17.1638)
tensor(10.3446)
tensor(14.0678)
tensor(6.4083)
tensor(6.5841)
tensor(6.8603)
tensor(12.6677)
tensor(2.1041)
tensor(6.5347)
tensor(8.6729)
tensor(1.7817)
tensor(16.4769)
tensor(14.7221)
tensor(6.5141)
tensor(6.3447)
tensor(7.0322)
tensor(14.2472)
tensor(11.5224)
tensor(10.3240)
tensor(12.3260)
tensor(17.9632)
tensor(8.4760)
tensor(14.8222)
tensor(13.9592)
tensor(5.4896)
tensor(8.4407)
tensor(6.1763)
tensor(5.7058)
tensor(16.0879)
tensor(16.5219)
tensor(22.6181)
tensor(19.1038)
tensor(3.9959)
tensor(3.4588)
tensor(5.4450)
tensor(14.3530)
tensor(14.5432)
tensor(1.7675)
tensor(10.1312)
tensor(10.7604)
tensor(15.1152)
tensor(14.8797)
```

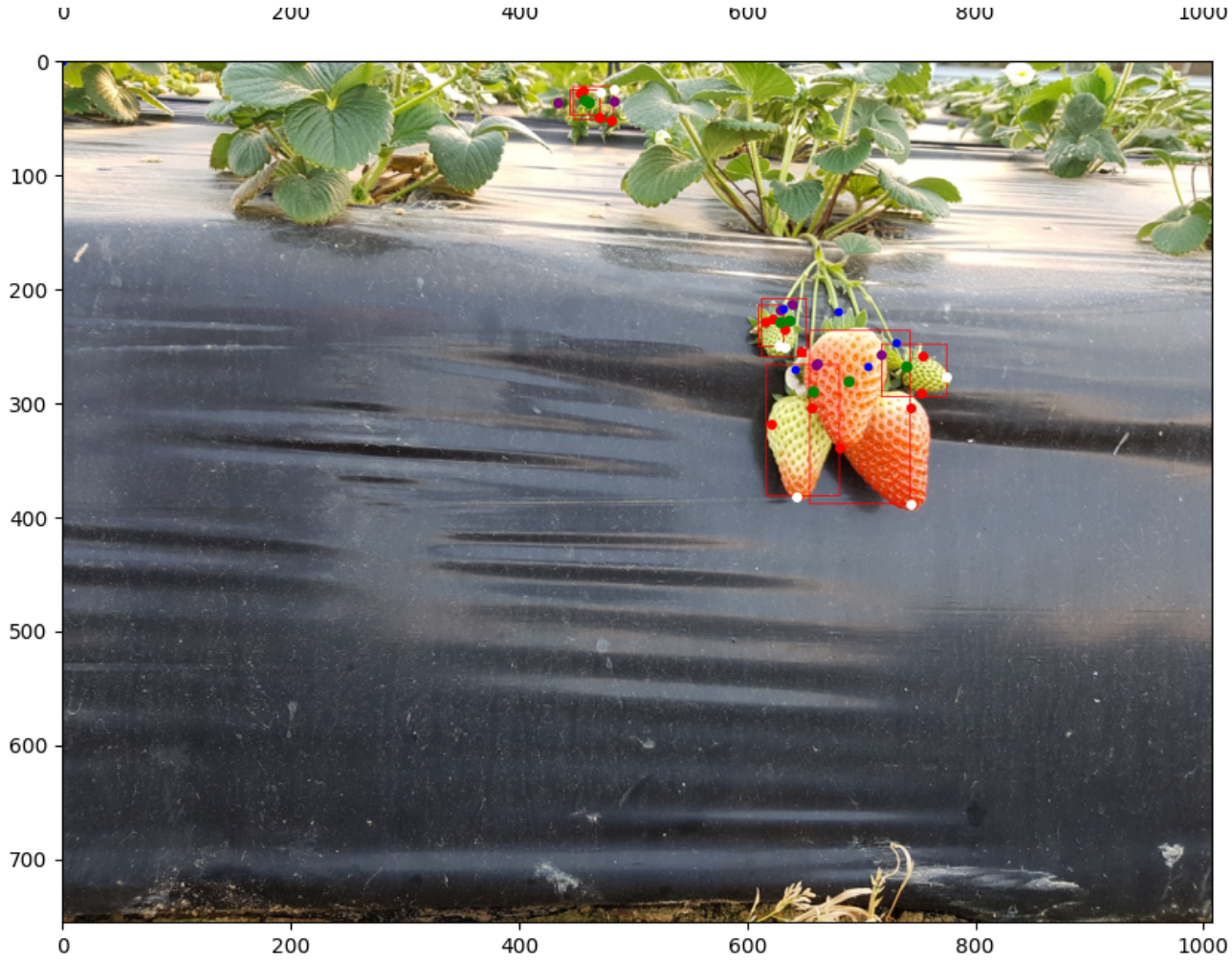
```
tensor(14.0797)
tensor(22.7334)
tensor(24.6863)
tensor(9.1984)
tensor(8.4497)
tensor(21.5309)
tensor(10.8093)
tensor(18.1220)
tensor(8.3147)
tensor(19.5754)
tensor(21.8474)
tensor(23.1368)
tensor(20.9907)
tensor(10.8603)
tensor(18.7404)
tensor(5.1030)
tensor(12.8244)
tensor(8.8633)
tensor(1.5354)
tensor(14.9594)
tensor(7.9999)
tensor(7.1370)
tensor(19.8049)
tensor(10.6723)
tensor(15.2550)
tensor(14.4845)
tensor(15.4634)
tensor(5.0348)
tensor(13.5181)
tensor(9.0302)
tensor(7.7794)
tensor(16.7877)
tensor(6.0747)
tensor(12.3692)
tensor(5.0899)
tensor(4.3808)
tensor(12.4295)
tensor(14.1619)
tensor(6.0139)
tensor(11.3541)
tensor(5.7688)
tensor(15.1985)
tensor(14.0999)
tensor(12.5942)
tensor(9.3750)
tensor(2.7344)
tensor(16.5431)
tensor(2.7655)
tensor(12.9373)
tensor(2.2141)
tensor(0.6844)
tensor(4.3189)
tensor(13.2530)
tensor(9.9175)
tensor(1.8947)
```

```
tensor(11.2650)
tensor(19.6020)
tensor(8.4450)
tensor(3.7888)
---------------------------------------------------------------------------
KeyboardInterrupt                         Traceback (most recent call last)
<ipython-input-54-ce2dd9a9ef24> in <cell line: 1>()
----> 1 testSetTruePreds,acceptablePredictionsInDataset,sumOfPxlwiseError =
testFN(60,67,True,True)
      2
      3 print(testSetTruePreds/acceptablePredictionsInDataset)
      4 print(sumOfPxlwiseError/acceptablePredictionsInDataset)
      5 print(sumOfPxlwiseError,acceptablePredictionsInDataset)
```

8 frames

```
/usr/local/lib/python3.10/dist-packages/torch/nn/modules/module.py in
_call_impl(self, *args, **kwargs)
   1554
   1555     def _call_impl(self, *args, **kwargs):
-> 1556         forward_call = (self._slow_forward if torch._C._get_tracing_state()
else self.forward)
   1557         # If we don't have any hooks, we want to skip the rest of the logic
in
   1558         # this function, and just call forward.

KeyboardInterrupt:
```


0
200
400
600
800
1000
0
100
200
300
400
500
600
700


0
100
200
300
400
500
600
700


```
# create the model from the code above and then run the code below
device = torch.device('cuda') if torch.cuda.is_available() else torch.device('cpu')
model.load_state_dict(torch.load("/content/drive/MyDrive/shufflenet_v2_x0_5NotGuidedNoOff
model.eval()
model.to(device)
```

```
<ipython-input-27-67976b4ff080>:3: FutureWarning: You are using `torch.load` with `we
  model.load_state_dict(torch.load("/content/drive/MyDrive/shufflenet_v2_x0_5NotGuide
KeypointRCNN(
  (transform): GeneralizedRCNNTransform(
      Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
      Resize(min_size=(640, 672, 704, 736, 768, 800), max_size=1333,
mode='bilinear')
  )
  (backbone): Sequential(
    (0): Sequential(
      (0): Conv2d(3, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
bias=False)
      (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
      (2): ReLU(inplace=True)
    )
    (1): MaxPool2d(kernel_size=3, stride=2, padding=1, dilation=1, ceil_mode=False)
    (2): Sequential(
      (0): InvertedResidual(
        (branch1): Sequential(
          (0): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
groups=24, bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True
```

```
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (2): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (3): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (4): ReLU(inplace=True)
        )
        (branch2): Sequential(
          (0): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (2): ReLU(inplace=True)
          (3): Conv2d(24, 24, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1),
groups=24, bias=False)
          (4): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (5): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (6): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (7): ReLU(inplace=True)
        )
      )
      (1): InvertedResidual(
        (branch1): Sequential()
        (branch2): Sequential(
          (0): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (1): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (2): ReLU(inplace=True)
          (3): Conv2d(24, 24, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1),
groups=24, bias=False)
          (4): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
          (5): Conv2d(24, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
          (6): BatchNorm2d(24, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
```