

Project 1 (alexnet_v1)

August 5, 2019

1 Painter classification

1.0.1 Author: Fan Wenxiao

1.0.2 School: Beijing Institute of Technology

First, we classify the problem. The problem of classification of painters is a process of supervising learning and a classification problem. So for such a problem, we used the more established model alexnet on the network for training. The final accuracy is stable at 85.41%.

Secondly, in this problem, there is a problem that the data set is too small, so we enhance the data set by means of random rotation and mirror flipping, and increase the size of the data set from 400 to 4000. This is conducive to the generalization of the model.

```
[0]: import torch
import torch.nn as nn
import torch.nn.functional as F
import torchvision.transforms as transforms
from torch.utils.data import DataLoader
from torchvision.datasets import ImageFolder
import torch.optim as optim
import numpy as np
```

Load data that has been data-enhanced and convert it to a tensor.

```
[0]: data_transform = transforms.Compose([
    #transforms.Resize(299),
    #transforms.CenterCrop(299),
    transforms.ToTensor(),
    transforms.Normalize(mean=[0.5, 0.5, 0.5],
                          std=[0.5, 0.5, 0.5])
])
train_dataset = ImageFolder(root='/content/drive/My Drive/Colab Notebooks/
→artist_distribute',transform=data_transform)
trainloader = torch.utils.data.DataLoader(train_dataset,batch_size=128,
→shuffle=True)

print(train_dataset.class_to_idx)
#print(len(train_dataset[0]))
```

```
{'canaletto': 0, 'claudmonet': 1, 'george_romney': 2, 'jmw_turner': 3,
'john_robert_cozens': 4, 'paul_cezanne': 5, 'paul_gauguin': 6, 'paul_sandby': 7,
'peter_paul_rubens': 8, 'rembrandt': 9, 'richard_wilson': 10}
```

Define the network structure. Here we have modified alexnet to add batch normalization in front of the activation function for each layer in the convolutional layer. At the same time, two attention mechanisms are added to each layer of the convolutional layer. These two attention mechanisms come from Resnet-18.

```
[0]: #
class ChannelAttention(nn.Module):
    def __init__(self, in_planes, ratio=16):
        super(ChannelAttention, self).__init__()
        self.avg_pool = nn.AdaptiveAvgPool2d(1)
        self.max_pool = nn.AdaptiveMaxPool2d(1)

        self.fc1 = nn.Conv2d(in_planes, in_planes // 16, 1, bias=False)
        self.relu1 = nn.ReLU()
        self.fc2 = nn.Conv2d(in_planes // 16, in_planes, 1, bias=False)

        self.sigmoid = nn.Sigmoid()

    def forward(self, x):
        avg_out = self.fc2(self.relu1(self.fc1(self.avg_pool(x))))
        max_out = self.fc2(self.relu1(self.fc1(self.max_pool(x))))
        out = avg_out + max_out
        return self.sigmoid(out)

class SpatialAttention(nn.Module):
    def __init__(self, kernel_size=7):
        super(SpatialAttention, self).__init__()

        assert kernel_size in (3, 7), 'kernel size must be 3 or 7'
        padding = 3 if kernel_size == 7 else 1

        self.conv1 = nn.Conv2d(2, 1, kernel_size, padding=padding, bias=False)
        self.sigmoid = nn.Sigmoid()

    def forward(self, x):
        avg_out = torch.mean(x, dim=1, keepdim=True)
        max_out, _ = torch.max(x, dim=1, keepdim=True)
        x = torch.cat([avg_out, max_out], dim=1)
        x = self.conv1(x)
        return self.sigmoid(x)

[0]: class Net(torch.nn.Module):
    def __init__(self):
        super(Net, self).__init__()
```

```

self.conv1 = torch.nn.Sequential(
    torch.nn.Conv2d(3, 96, 11, 4, 0),
    torch.nn.BatchNorm2d(96),
    torch.nn.ReLU(),
    torch.nn.MaxPool2d(3,2)
)
self.conv2 = torch.nn.Sequential(
    torch.nn.Conv2d(96, 256, 5, 1, 2),
    torch.nn.BatchNorm2d(256),
    torch.nn.ReLU(),
    torch.nn.MaxPool2d(3,2)
)
self.conv3 = torch.nn.Sequential(
    torch.nn.Conv2d(256,384, 3, 1, 1),
    torch.nn.BatchNorm2d(384),
    torch.nn.ReLU()

)
self.conv4 = torch.nn.Sequential(
    torch.nn.Conv2d(384,384, 3, 1, 1),
    torch.nn.BatchNorm2d(384),
    torch.nn.ReLU()

)
self.conv5 = torch.nn.Sequential(
    torch.nn.Conv2d(384,256, 3, 1, 1),
    torch.nn.BatchNorm2d(256),
    torch.nn.ReLU(),
    torch.nn.MaxPool2d(3,2)

)
self.dense = torch.nn.Sequential(
    torch.nn.Linear(9216, 4096),
    #torch.nn.BatchNorm2d(4096),
    torch.nn.ReLU(),
    torch.nn.Dropout(0.5),
    torch.nn.Linear(4096, 512),
    #torch.nn.BatchNorm2d(512),
    torch.nn.ReLU(),
    torch.nn.Dropout(0.5),
    #torch.nn.BatchNorm2d(512),
    torch.nn.Linear(512, 11)
)
self.ca1 = ChannelAttention(24 * 4)
self.ca2 = ChannelAttention(256)
self.ca4 = ChannelAttention(256)
self.ca3 = ChannelAttention(384)

```

```

        self.sa = SpatialAttention()

    def forward(self, x):
        conv1_out = self.conv1(x)
        conv1_out = self.ca1(conv1_out) * conv1_out
        conv1_out = self.sa(conv1_out) * conv1_out

        conv2_out = self.conv2(conv1_out)
        conv2_out = self.ca2(conv2_out) * conv2_out
        conv2_out = self.sa(conv2_out) * conv2_out

        conv3_out = self.conv3(conv2_out)
        conv3_out = self.ca3(conv3_out) * conv3_out
        conv3_out = self.sa(conv3_out) * conv3_out

        conv4_out = self.conv4(conv3_out)
        conv4_out = self.ca3(conv4_out) * conv4_out
        conv4_out = self.sa(conv4_out) * conv4_out

        conv5_out = self.conv5(conv4_out)
        conv5_out = self.ca4(conv5_out) * conv5_out
        conv5_out = self.sa(conv5_out) * conv5_out
        #conv5_out = self.ca(conv5_out)
        res = conv5_out.view(conv5_out.size(0), -1)
        out = self.dense(res)
        return out

```

```

[0]: na = Net()
      print(na)

```

```

Net(
  (conv1): Sequential(
    (0): Conv2d(3, 96, kernel_size=(11, 11), stride=(4, 4))
    (1): BatchNorm2d(96, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
    (2): ReLU()
    (3): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,
ceil_mode=False)
  )
  (conv2): Sequential(
    (0): Conv2d(96, 256, kernel_size=(5, 5), stride=(1, 1), padding=(2, 2))
    (1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
    (2): ReLU()
    (3): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,
ceil_mode=False)
  )
)

```

```

(conv3): Sequential(
  (0): Conv2d(256, 384, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(384, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  (2): ReLU()
)
(conv4): Sequential(
  (0): Conv2d(384, 384, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(384, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  (2): ReLU()
)
(conv5): Sequential(
  (0): Conv2d(384, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
  (1): BatchNorm2d(256, eps=1e-05, momentum=0.1, affine=True,
track_running_stats=True)
  (2): ReLU()
  (3): MaxPool2d(kernel_size=3, stride=2, padding=0, dilation=1,
ceil_mode=False)
)
(dense): Sequential(
  (0): Linear(in_features=9216, out_features=4096, bias=True)
  (1): ReLU()
  (2): Dropout(p=0.5)
  (3): Linear(in_features=4096, out_features=512, bias=True)
  (4): ReLU()
  (5): Dropout(p=0.5)
  (6): Linear(in_features=512, out_features=11, bias=True)
)
(ca1): ChannelAttention(
  (avg_pool): AdaptiveAvgPool2d(output_size=1)
  (max_pool): AdaptiveMaxPool2d(output_size=1)
  (fc1): Conv2d(96, 6, kernel_size=(1, 1), stride=(1, 1), bias=False)
  (relu1): ReLU()
  (fc2): Conv2d(6, 96, kernel_size=(1, 1), stride=(1, 1), bias=False)
  (sigmoid): Sigmoid()
)
(ca2): ChannelAttention(
  (avg_pool): AdaptiveAvgPool2d(output_size=1)
  (max_pool): AdaptiveMaxPool2d(output_size=1)
  (fc1): Conv2d(256, 16, kernel_size=(1, 1), stride=(1, 1), bias=False)
  (relu1): ReLU()
  (fc2): Conv2d(16, 256, kernel_size=(1, 1), stride=(1, 1), bias=False)
  (sigmoid): Sigmoid()
)
(ca4): ChannelAttention(
  (avg_pool): AdaptiveAvgPool2d(output_size=1)
  (max_pool): AdaptiveMaxPool2d(output_size=1)

```

```

        (fc1): Conv2d(256, 16, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (relu1): ReLU()
        (fc2): Conv2d(16, 256, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (sigmoid): Sigmoid()
    )
    (ca3): ChannelAttention(
        (avg_pool): AdaptiveAvgPool2d(output_size=1)
        (max_pool): AdaptiveMaxPool2d(output_size=1)
        (fc1): Conv2d(384, 24, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (relu1): ReLU()
        (fc2): Conv2d(24, 384, kernel_size=(1, 1), stride=(1, 1), bias=False)
        (sigmoid): Sigmoid()
    )
    (sa): SpatialAttention(
        (conv1): Conv2d(2, 1, kernel_size=(7, 7), stride=(1, 1), padding=(3, 3),
bias=False)
        (sigmoid): Sigmoid()
    )
)

```

Define the loss function and optimizer, and start training.

```

[7]: net = Net()
#net = torch.load('/content/drive/My Drive/Colab Notebooks/alexnet23.pkl')
net = net.cuda()
criterion = nn.CrossEntropyLoss()
criterion = criterion.cuda()
optimizer = optim.Adam(net.parameters(), lr=0.001, weight_decay=0.00005)
#ExponentialLR
optim.lr_scheduler.StepLR(optimizer, 10, gamma=0.1, last_epoch=-1)
print("Start Training...")
LOSS = []
num_epoch = 30
for epoch in range(num_epoch):
    # 10batchloss
    loss10 = 0.0
    # dataloader
    for i, data in enumerate(trainloader):
        inputs, labels = data
        inputs = inputs.cuda()
        labels = labels.cuda()
        #print(labels)
        optimizer.zero_grad()
        outputs = net(inputs)
        loss = criterion(outputs, labels)
        loss.backward()
        optimizer.step()
        #print('a')

```

```

    #print(torch.cuda.current_device())
    loss10 += loss.item()
    if i % 10 == 9:
        print('[Epoch %d, Batch %5d] loss: %.3f' %
              (epoch + 1, i + 1, loss10 / 10))
        LOSS.append(loss10/10)
        loss10 = 0.0

print("Done Training!")

```

Start Training...

```

[Epoch 1, Batch 10] loss: 2.494
[Epoch 1, Batch 20] loss: 1.802
[Epoch 1, Batch 30] loss: 1.603
[Epoch 2, Batch 10] loss: 1.368
[Epoch 2, Batch 20] loss: 1.214
[Epoch 2, Batch 30] loss: 1.113
[Epoch 3, Batch 10] loss: 0.958
[Epoch 3, Batch 20] loss: 0.932
[Epoch 3, Batch 30] loss: 0.803
[Epoch 4, Batch 10] loss: 0.742
[Epoch 4, Batch 20] loss: 0.719
[Epoch 4, Batch 30] loss: 0.715
[Epoch 5, Batch 10] loss: 0.657
[Epoch 5, Batch 20] loss: 0.623
[Epoch 5, Batch 30] loss: 0.659
[Epoch 6, Batch 10] loss: 0.568
[Epoch 6, Batch 20] loss: 0.570
[Epoch 6, Batch 30] loss: 0.522
[Epoch 7, Batch 10] loss: 0.485
[Epoch 7, Batch 20] loss: 0.516
[Epoch 7, Batch 30] loss: 0.477
[Epoch 8, Batch 10] loss: 0.425
[Epoch 8, Batch 20] loss: 0.388
[Epoch 8, Batch 30] loss: 0.375
[Epoch 9, Batch 10] loss: 0.281
[Epoch 9, Batch 20] loss: 0.336
[Epoch 9, Batch 30] loss: 0.292
[Epoch 10, Batch 10] loss: 0.249
[Epoch 10, Batch 20] loss: 0.307
[Epoch 10, Batch 30] loss: 0.303
[Epoch 11, Batch 10] loss: 0.274
[Epoch 11, Batch 20] loss: 0.175
[Epoch 11, Batch 30] loss: 0.174
[Epoch 12, Batch 10] loss: 0.178
[Epoch 12, Batch 20] loss: 0.155
[Epoch 12, Batch 30] loss: 0.137

```

| | | |
|------------------|-----|-------------|
| [Epoch 13, Batch | 10] | loss: 0.130 |
| [Epoch 13, Batch | 20] | loss: 0.120 |
| [Epoch 13, Batch | 30] | loss: 0.193 |
| [Epoch 14, Batch | 10] | loss: 0.147 |
| [Epoch 14, Batch | 20] | loss: 0.138 |
| [Epoch 14, Batch | 30] | loss: 0.125 |
| [Epoch 15, Batch | 10] | loss: 0.108 |
| [Epoch 15, Batch | 20] | loss: 0.101 |
| [Epoch 15, Batch | 30] | loss: 0.101 |
| [Epoch 16, Batch | 10] | loss: 0.159 |
| [Epoch 16, Batch | 20] | loss: 0.127 |
| [Epoch 16, Batch | 30] | loss: 0.125 |
| [Epoch 17, Batch | 10] | loss: 0.125 |
| [Epoch 17, Batch | 20] | loss: 0.122 |
| [Epoch 17, Batch | 30] | loss: 0.108 |
| [Epoch 18, Batch | 10] | loss: 0.072 |
| [Epoch 18, Batch | 20] | loss: 0.087 |
| [Epoch 18, Batch | 30] | loss: 0.059 |
| [Epoch 19, Batch | 10] | loss: 0.062 |
| [Epoch 19, Batch | 20] | loss: 0.080 |
| [Epoch 19, Batch | 30] | loss: 0.111 |
| [Epoch 20, Batch | 10] | loss: 0.123 |
| [Epoch 20, Batch | 20] | loss: 0.067 |
| [Epoch 20, Batch | 30] | loss: 0.051 |
| [Epoch 21, Batch | 10] | loss: 0.046 |
| [Epoch 21, Batch | 20] | loss: 0.044 |
| [Epoch 21, Batch | 30] | loss: 0.064 |
| [Epoch 22, Batch | 10] | loss: 0.083 |
| [Epoch 22, Batch | 20] | loss: 0.047 |
| [Epoch 22, Batch | 30] | loss: 0.081 |
| [Epoch 23, Batch | 10] | loss: 0.089 |
| [Epoch 23, Batch | 20] | loss: 0.069 |
| [Epoch 23, Batch | 30] | loss: 0.092 |
| [Epoch 24, Batch | 10] | loss: 0.054 |
| [Epoch 24, Batch | 20] | loss: 0.055 |
| [Epoch 24, Batch | 30] | loss: 0.033 |
| [Epoch 25, Batch | 10] | loss: 0.022 |
| [Epoch 25, Batch | 20] | loss: 0.039 |
| [Epoch 25, Batch | 30] | loss: 0.031 |
| [Epoch 26, Batch | 10] | loss: 0.023 |
| [Epoch 26, Batch | 20] | loss: 0.044 |
| [Epoch 26, Batch | 30] | loss: 0.084 |
| [Epoch 27, Batch | 10] | loss: 0.080 |
| [Epoch 27, Batch | 20] | loss: 0.075 |
| [Epoch 27, Batch | 30] | loss: 0.039 |
| [Epoch 28, Batch | 10] | loss: 0.079 |
| [Epoch 28, Batch | 20] | loss: 0.060 |
| [Epoch 28, Batch | 30] | loss: 0.076 |


```

[Epoch 29, Batch    10] loss: 0.044
[Epoch 29, Batch    20] loss: 0.034
[Epoch 29, Batch    30] loss: 0.036
[Epoch 30, Batch    10] loss: 0.024
[Epoch 30, Batch    20] loss: 0.044
[Epoch 30, Batch    30] loss: 0.016
Done Training!

```

Evaluation

```

[8]: # In[eval]
test_dataset = ImageFolder(root='/content/drive/My Drive/Colab Notebooks/
    ↳test_set',transform=data_transform)
testloader = torch.utils.data.DataLoader(test_dataset,batch_size=46,
    ↳shuffle=True)
eval_loss = 0
correct = 0
total = 0
for i, data in enumerate(testloader):
    inputs, labels = data
    inputs = inputs.cuda()
    labels = labels.cuda()
    #print(labels)
    outputs = net(inputs)
    loss = criterion(outputs, labels)
    eval_loss += loss.item()
    total += labels.size(0)
    pred = torch.max(outputs, 1)[1]
    print('pred: ',pred,' labels: ',labels)
    correct += (pred == labels).sum().item()
print('eval_loss: ',eval_loss)
print('correct: ',correct)
print('Accuracy of the network on the test images: %d %%' % (100 * correct /
    ↳total))

```

```

pred: tensor([ 3,  4,  4, 10,  7,  4,  2,  5,  6,  6,  8,  1,  2,  5,  3,  5,
 9,  2,
          1,  1,  2,  4,  8,  3,  1, 10,  3,  4, 10,  0,  1,  6,  0,  3,  9,  1,
 9,  9,  7,  3,  9,  2,  5,  2,  2, 10], device='cuda:0') labels:
tensor([ 3,  4,  3,  3,  7,  4,  2,  1,  6,  6,  5,  1,  2,  5,  3,  5,  8,  2,
 1,  1,  2,  4,  8,  3,  1, 10,  3,  4,  3,  0,  7,  6,  0,  3,  9,  1,
 9,  9,  7,  3,  9,  2,  5,  2,  2, 10], device='cuda:0')
eval_loss: 1.051384449005127
correct: 39
Accuracy of the network on the test images: 84 %

```

```

[9]: summ = 0
for ii in range(100):

```

```

test_dataset = ImageFolder(root='/content/drive/My Drive/Colab Notebooks/
→test_set',transform=data_transform)
testloader = torch.utils.data.DataLoader(test_dataset,batch_size=46,
→shuffle=True)
eval_loss = 0
correct = 0
total = 0
for i, data in enumerate(testloader):
    inputs, labels = data
    #print(labels)

    inputs = inputs.cuda()
    labels = labels.cuda()
    outputs = net(inputs)
    loss = criterion(outputs, labels)
    eval_loss += loss.item()
    total += labels.size(0)
    pred = torch.max(outputs, 1)[1]
    print('pred: ',pred,' labels: ',labels)
    correct += (pred == labels).sum().item()
    summ += (100 * correct / total)
print('eval_loss: ',eval_loss)
print('correct: ',correct)
print('Accuracy of the network on the test images: %d %%' % (100 * correct /
→total))
print("total_acc: ", summ/100)

```

```

pred: tensor([ 1,  3,  5,  5,  9,  2,  5,  0,  1,  3,  0,  4,  6,  9,  4,  9,
 3,  2,
           3,  6,  2,  2,  1,  7,  7,  3, 10,  1, 10,  6,  8,  5, 10,  4,  6,  1,
           8,  4,  9,  2,  1,  3, 10,  2,  9,  2], device='cuda:0') labels:
tensor([ 7,  3,  5,  5,  8,  2,  5,  0,  1,  3,  0,  4,  3,  9,  4,  9,  3,  2,
 3,  6,  2,  2,  1,  7,  7,  3,  3,  1, 10,  6,  5,  1,  3,  4,  6,  1,
           8,  4,  9,  2,  1,  3, 10,  2,  9,  2], device='cuda:0')
eval_loss: 1.3486568927764893
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 7,  2,  1,  2,  3,  2,  5,  8,  1,  2,  9,  4, 10,  4,  9,  1,
 2,  0,
           7,  6,  4,  1,  2, 10,  8,  2,  5,  3,  3,  9,  9,  3, 10,  3,  6,  9,
           3,  4, 10,  1,  0,  1,  1,  6,  4,  5], device='cuda:0') labels:
tensor([ 7,  2,  1,  2,  3,  2,  5,  8,  7,  2,  9,  4,  3,  3,  9,  1,  2,  0,
 7,  6,  4,  1,  2,  3,  5,  2,  5,  3,  3,  8,  9,  3, 10,  3,  6,  9,
           3,  4, 10,  1,  0,  1,  1,  6,  4,  5], device='cuda:0')
eval_loss: 0.9233337044715881
correct: 40
Accuracy of the network on the test images: 86 %

```

```

pred: tensor([ 3,  5, 10,  3, 10,  6,  5,  4,  1,  5,  3,  4,  2,  9,  2,  4,
 2,  2,
           2,  2,  1,  4,  8, 10,  0,  2,  7,  1,  3,  0,  1, 10,  6,  9,  3,  9,
 5,  5,  9,  9,  4,  1,  3,  1,  6,  7], device='cuda:0') labels:
tensor([ 3,  5,  3,  3, 10,  6,  5,  4,  1,  5,  3,  4,  2,  8,  2,  4,  2,  2,
 2,  2,  1,  4,  8, 10,  0,  2,  7,  1,  3,  0,  7,  3,  6,  9,  3,  9,
 1,  5,  9,  9,  3,  1,  3,  1,  6,  7], device='cuda:0')
eval_loss: 0.7947221994400024
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 7,  3,  6,  9,  2,  3,  4, 10,  1,  3,  0,  9,  5,  8,  4,  1,
 1,  4,
           8,  5,  2,  5, 10,  9,  0,  9,  3,  6,  7,  3,  1,  4,  5,  2,  3,  1,
 4,  2,  3,  2,  2, 10,  1,  6,  9,  2], device='cuda:0') labels:
tensor([ 7,  3,  6,  9,  2,  3,  3, 10,  1,  3,  0,  9,  1,  8,  4,  1,  1,  4,
 5,  5,  2,  5, 10,  9,  0,  8,  3,  6,  7,  3,  7,  4,  5,  2,  3,  1,
 4,  2,  3,  2,  2,  3,  1,  6,  9,  2], device='cuda:0')
eval_loss: 1.043348789215088
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 9,  8,  7,  9,  2,  6,  5,  3,  6,  7, 10,  4,  9,  0,  1,  9,
10,  6,
           3,  1,  2,  5,  4,  9, 10,  2,  2,  4,  1,  8,  3,  2,  3,  1,  3,  5,
 3,  2, 10,  5,  0,  2,  6,  1,  4,  1], device='cuda:0') labels:
tensor([ 9,  5,  7,  9,  2,  6,  5,  3,  6,  7,  3,  4,  9,  0,  1,  9, 10,  6,
 3,  1,  2,  1,  4,  8, 10,  2,  2,  4,  7,  8,  3,  2,  3,  1,  3,  5,
 3,  2,  3,  5,  0,  2,  3,  1,  4,  1], device='cuda:0')
eval_loss: 1.0928888320922852
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 1,  7, 10,  9,  4,  2,  4,  4,  5,  8,  7, 10,  2, 10, 10,  9,
 4,  1,
           8,  0,  3,  2,  5,  2,  9,  9,  1,  1,  2,  8,  2,  5,  5,  1,  3,  6,
 2,  3,  4,  3,  3,  1,  6,  9, 10,  0], device='cuda:0') labels:
tensor([ 1,  7,  3,  9,  4,  2,  4,  4,  5,  8,  7, 10,  2,  3,  3,  9,  3,  1,
 5,  0,  3,  2,  5,  2,  9,  8,  1,  7,  2,  6,  2,  1,  5,  1,  3,  6,
 2,  3,  4,  3,  3,  1,  6,  9, 10,  0], device='cuda:0')
eval_loss: 0.970335066318512
correct: 37
Accuracy of the network on the test images: 80 %
pred: tensor([ 2,  2,  2,  1,  3, 10,  1,  2, 10,  6,  2,  4,  4,  2,  9,  7,
 7,  3,
           5,  2, 10,  3,  5,  4,  3,  6,  1,  1,  3,  1,  0,  4,  8,  1,  9,  4,
 9,  5,  9,  9,  8,  3,  3,  3,  0,  1], device='cuda:0') labels:
tensor([ 2,  2,  2,  1,  3,  3,  1,  2, 10,  6,  2,  3,  4,  2,  9,  7,  7,  3,
 5,  2, 10,  3,  5,  4,  3,  6,  1,  1,  6,  7,  0,  4,  8,  1,  9,  4,
 9,  5,  9,  8,  5,  3,  3,  3,  0,  1], device='cuda:0')
eval_loss: 0.983558177947998

```

```

correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 9,  8,  4,  2,  1,  3, 10,  1,  2,  3, 10,  3,  9,  1,  4,  3,
 4,  1,
           2,  7,  1,  8,  6,  9, 10,  6,  0,  9,  4,  1,  1,  2,  3,  5,  0,  9,
           2,  5,  2,  4,  5,  2,  7,  3,  6,  3], device='cuda:0') labels:
tensor([ 9,  5,  4,  2,  7,  3, 10,  1,  2,  3, 10,  3,  9,  1,  4,  3,  4,  1,
 2,  7,  1,  8,  6,  9,  3,  6,  0,  9,  3,  1,  1,  2,  3,  5,  0,  8,
 2,  5,  2,  4,  5,  2,  7,  3,  6,  3], device='cuda:0')
eval_loss: 0.6441054940223694
correct: 41
Accuracy of the network on the test images: 89 %
pred: tensor([ 9, 10,  3,  2,  4,  1,  5,  4,  9,  1,  9,  7,  3,  2,  4,  8,
 8, 10,
           1,  6,  1,  4,  5,  3,  2,  9,  1,  7, 10,  6,  3,  2,  6,  4,  0,  2,
           3,  9,  2,  2,  1,  3, 10,  5,  0,  5], device='cuda:0') labels:
tensor([ 9, 10,  3,  2,  4,  1,  5,  3,  8,  7,  9,  7,  3,  2,  4,  8,  5, 10,
 1,  6,  1,  4,  1,  3,  2,  9,  1,  7,  3,  6,  3,  2,  6,  4,  0,  2,
 3,  9,  2,  2,  1,  3,  3,  5,  0,  5], device='cuda:0')
eval_loss: 0.8409097790718079
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 9,  8,  1,  2,  9,  2,  7,  2,  7,  3,  2,  3,  3, 10,  1,  6,
 2,  2,
           10,  6,  5,  8, 10,  4,  5,  4,  4,  3,  0,  5,  0,  6,  3,  5,  2,  3,
           1,  9,  4,  1,  8,  1,  0,  1,  9,  9], device='cuda:0') labels:
tensor([ 9,  6,  7,  2,  8,  2,  7,  2,  7,  3,  2,  3,  3, 10,  1,  6,  2,  2,
 3,  3,  1,  8, 10,  4,  5,  4,  4,  3,  3,  5,  0,  6,  3,  5,  2,  3,
 1,  9,  4,  1,  5,  1,  0,  1,  9,  9], device='cuda:0')
eval_loss: 0.9790526628494263
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([ 3,  3,  0,  7,  6,  9,  1,  8,  0,  5,  3,  2,  4,  2,  2,  2,
 1,  4,
           2,  4, 10,  9,  6,  5,  6,  5,  9,  1,  3, 10, 10,  7,  0,  5,  9,  4,
           1,  9,  2,  1,  3, 10,  6,  2,  1,  1], device='cuda:0') labels:
tensor([ 3,  3,  0,  7,  6,  8,  1,  8,  3,  5,  3,  2,  4,  2,  2,  2,  7,  4,
 2,  4,  3,  9,  3,  5,  6,  5,  9,  1,  3,  3, 10,  7,  0,  5,  9,  4,
 1,  9,  2,  1,  3, 10,  6,  2,  1,  1], device='cuda:0')
eval_loss: 0.8305867910385132
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1,  8,  9,  3, 10,  2,  7,  9,  2,  3,  5,  1,  4,  4,  6, 10,
 9,  6,
           4,  3, 10,  6,  3,  5,  0,  2,  3,  1,  1,  6, 10,  9,  2,  9,  3,  8,
           2,  1,  5,  4,  1,  0,  1,  7,  2,  2], device='cuda:0') labels:
tensor([ 1,  8,  9,  3,  3,  2,  7,  9,  2,  3,  5,  1,  4,  4,  6,  3,  9,  3,
 4,  3, 10,  6,  3,  5,  0,  2,  3,  1,  7,  6, 10,  8,  2,  9,  3,  5,

```

```

        2, 1, 5, 4, 1, 0, 1, 7, 2, 2], device='cuda:0')
eval_loss: 0.7832142114639282
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 3, 4, 6, 4, 1, 3, 3, 10, 8, 5, 3, 2, 4, 1, 1, 10,
2, 0,
        3, 2, 8, 10, 2, 4, 6, 0, 9, 10, 6, 5, 5, 5, 2, 9, 4, 1,
        1, 7, 2, 9, 7, 1, 3, 9, 9, 2], device='cuda:0') labels:
tensor([ 3, 3, 6, 4, 1, 3, 3, 3, 5, 5, 3, 2, 4, 1, 7, 3, 2, 0,
        3, 2, 8, 10, 2, 4, 6, 0, 9, 10, 6, 5, 5, 1, 2, 9, 4, 1,
        1, 7, 2, 9, 7, 1, 3, 8, 9, 2], device='cuda:0')
eval_loss: 1.10773503780365
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 3, 1, 4, 5, 4, 2, 3, 7, 3, 1, 9, 0, 1, 2, 10, 8,
9, 10,
        2, 5, 6, 8, 1, 4, 3, 3, 9, 2, 9, 9, 4, 6, 6, 1, 2, 5,
        0, 3, 7, 5, 1, 10, 4, 2, 2, 10], device='cuda:0') labels:
tensor([ 3, 1, 4, 1, 4, 2, 3, 7, 3, 1, 9, 0, 1, 2, 3, 5, 9, 10,
        2, 5, 6, 8, 1, 4, 3, 3, 9, 2, 9, 8, 4, 6, 6, 1, 2, 5,
        0, 3, 7, 5, 7, 10, 3, 2, 2, 3], device='cuda:0')
eval_loss: 1.379602313041687
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 4, 2, 8, 1, 10, 9, 10, 2, 10, 5, 10, 1, 3, 7, 8, 5,
9, 1,
        9, 3, 6, 5, 3, 9, 4, 2, 0, 3, 6, 1, 5, 6, 2, 2, 0, 2,
        1, 4, 3, 9, 3, 7, 2, 4, 1, 4], device='cuda:0') labels:
tensor([ 3, 2, 5, 1, 10, 9, 3, 2, 10, 5, 3, 7, 3, 7, 8, 5, 9, 1,
        9, 3, 6, 1, 3, 8, 4, 2, 0, 3, 6, 1, 5, 6, 2, 2, 0, 2,
        1, 4, 3, 9, 3, 7, 2, 4, 1, 4], device='cuda:0')
eval_loss: 0.9635856747627258
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 2, 6, 3, 3, 2, 6, 2, 5, 4, 10, 3, 5, 1, 1, 9, 2,
4, 0,
        1, 1, 3, 2, 4, 1, 5, 10, 10, 4, 7, 3, 1, 6, 3, 5, 9, 9,
        8, 6, 8, 9, 2, 10, 9, 0, 2, 7], device='cuda:0') labels:
tensor([ 2, 6, 3, 3, 2, 6, 2, 5, 4, 3, 3, 5, 7, 1, 9, 2, 4, 0,
        1, 1, 3, 2, 4, 1, 1, 3, 10, 4, 7, 3, 1, 6, 3, 5, 8, 9,
        5, 3, 8, 9, 2, 10, 9, 0, 2, 7], device='cuda:0')
eval_loss: 0.8988186120986938
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 1, 9, 3, 1, 1, 3, 0, 4, 10, 2, 5, 9, 1, 10, 8, 3,
3, 1,
        5, 6, 6, 7, 2, 4, 7, 9, 9, 6, 4, 2, 3, 2, 4, 5, 8, 1,
        10, 2, 9, 2, 0, 2, 5, 4, 3, 10], device='cuda:0') labels:

```

```

tensor([ 1,  9,  3,  1,  1,  3,  0,  4,  3,  2,  1,  9,  7, 10,  5,  3,  3,  1,
         5,  6,  6,  7,  2,  4,  7,  8,  9,  6,  4,  2,  3,  2,  4,  5,  8,  1,
        10,  2,  9,  2,  0,  2,  5,  3,  3,  3], device='cuda:0')
eval_loss: 1.1470890045166016
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 5,  8,  4,  9,  3,  3,  1,  2,  9,  2,  9, 10,  3,  4,  1,  0,
               2,  7,
               10,  9,  2,  4,  6, 10,  2,  5,  9,  6,  5,  6,  4,  3,  0,  1,  3,  4,
               3,  1,  8,  1,  1,  2,  5,  2, 10,  7], device='cuda:0') labels:
tensor([ 5,  8,  4,  9,  3,  3,  1,  2,  9,  2,  9, 10,  3,  4,  1,  0,  2,  7,
        10,  9,  2,  4,  6,  3,  2,  5,  8,  6,  5,  6,  3,  3,  0,  1,  3,  4,
         3,  1,  5,  7,  1,  2,  1,  2,  3,  7], device='cuda:0')
eval_loss: 0.9138121604919434
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 4,  9, 10,  5,  8,  0,  6,  4,  7,  1,  4,  1,  3,  3, 10,  2,
               5,  2,
               0,  1,  0,  2,  9,  5,  6,  5,  2,  3,  9,  7,  6,  3,  8,  2,  2,  1,
               9,  2,  1,  4,  1,  9, 10,  3, 10,  4], device='cuda:0') labels:
tensor([ 4,  9, 10,  5,  5,  0,  6,  4,  7,  1,  3,  1,  3,  3,  3,  2,  5,  2,
         0,  1,  3,  2,  9,  5,  6,  1,  2,  3,  9,  7,  6,  3,  8,  2,  2,  7,
         8,  2,  1,  4,  1,  9, 10,  3,  3,  4], device='cuda:0')
eval_loss: 1.0906049013137817
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([ 2,  3,  3,  5,  1,  5,  5,  2,  4, 10,  2,  6,  1,  8,  1,  6,
               0,  1,
               9,  3,  9,  3,  7,  3,  9,  1,  8,  3,  3,  2,  0, 10,  6,  2,  4,  7,
               2,  6,  2,  9,  4,  9,  0,  1,  4,  5], device='cuda:0') labels:
tensor([ 2,  3,  3,  5,  1,  5,  1,  2,  4, 10,  2,  6,  1,  8,  1,  6,  0,  1,
         9,  3,  8,  3,  7,  3,  9,  1,  5,  3,  3,  2,  3, 10,  3,  2,  4,  7,
         2,  6,  2,  9,  4,  9,  0,  7,  4,  5], device='cuda:0')
eval_loss: 1.013755202293396
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1,  6,  2,  1,  1,  5,  2,  0,  9,  3,  2,  3,  1,  9,  1,  4,
               10,  7,
               3,  5,  9, 10,  1,  4,  2,  4,  8,  9, 10,  4,  7,  2,  2, 10,  3,  5,
               1,  8, 10,  0,  6,  6,  0,  2,  9,  4], device='cuda:0') labels:
tensor([ 1,  6,  2,  1,  1,  5,  2,  0,  9,  3,  2,  3,  7,  8,  1,  4, 10,  7,
         3,  5,  9,  3,  1,  4,  2,  3,  5,  9,  3,  4,  7,  2,  2, 10,  3,  5,
         1,  8,  3,  3,  6,  6,  0,  2,  9,  4], device='cuda:0')
eval_loss: 1.1647247076034546
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([ 1,  2,  7,  2,  3,  1,  4,  4,  9,  2,  2,  6,  6,  1, 10,  1,
               3,  4,

```

```

      8, 2, 5, 3, 1, 10, 3, 10, 2, 9, 3, 5, 4, 9, 9, 2, 9, 3,
      5, 7, 10, 6, 4, 5, 6, 0, 1, 0], device='cuda:0') labels:
tensor([ 7, 2, 7, 2, 3, 1, 4, 4, 8, 2, 2, 6, 5, 1, 10, 1, 3, 4,
      8, 2, 5, 3, 1, 3, 3, 3, 2, 9, 3, 5, 4, 9, 9, 2, 9, 3,
      1, 7, 10, 6, 3, 5, 6, 0, 1, 0], device='cuda:0')
eval_loss: 0.8112536668777466
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 1, 0, 9, 1, 9, 7, 7, 10, 8, 1, 9, 5, 4, 1, 5, 6,
      4, 3,
      4, 4, 3, 1, 3, 6, 2, 5, 0, 3, 1, 5, 2, 2, 3, 0, 9, 9,
      4, 3, 8, 2, 2, 10, 2, 10, 6, 2], device='cuda:0') labels:
tensor([ 1, 0, 9, 7, 8, 7, 7, 10, 5, 1, 9, 5, 4, 1, 1, 6, 4, 3,
      4, 4, 3, 1, 3, 6, 2, 5, 0, 3, 1, 5, 2, 2, 3, 3, 9, 9,
      3, 3, 8, 2, 2, 10, 2, 3, 6, 2], device='cuda:0')
eval_loss: 1.1525132656097412
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 4, 4, 7, 9, 5, 3, 3, 2, 9, 6, 10, 0, 9, 2, 2, 2,
      2, 8,
      0, 1, 3, 1, 1, 5, 4, 9, 1, 2, 10, 10, 1, 4, 4, 5, 2, 3,
      6, 8, 3, 3, 7, 9, 1, 8, 1, 0], device='cuda:0') labels:
tensor([ 4, 3, 7, 9, 5, 3, 3, 2, 9, 6, 10, 0, 8, 2, 2, 2, 2, 5,
      3, 1, 3, 1, 1, 5, 4, 9, 7, 2, 3, 10, 1, 4, 4, 5, 2, 3,
      6, 6, 3, 3, 7, 9, 1, 8, 1, 0], device='cuda:0')
eval_loss: 0.773604691028595
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 9, 1, 10, 1, 3, 6, 4, 6, 1, 9, 1, 2, 9, 2, 7, 1,
      3, 3,
      2, 10, 4, 1, 8, 3, 7, 4, 2, 2, 0, 3, 9, 3, 4, 6, 1, 2,
      5, 9, 0, 10, 5, 10, 2, 8, 4, 5], device='cuda:0') labels:
tensor([ 9, 1, 10, 1, 3, 6, 4, 6, 1, 9, 1, 2, 8, 2, 7, 1, 3, 3,
      2, 3, 3, 1, 8, 3, 7, 4, 2, 2, 0, 3, 9, 3, 4, 6, 7, 2,
      5, 9, 0, 3, 5, 10, 2, 5, 4, 5], device='cuda:0')
eval_loss: 0.8754656314849854
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 3, 10, 9, 9, 6, 0, 5, 3, 9, 2, 3, 4, 7, 8, 4, 1,
      3, 4,
      2, 1, 9, 1, 6, 10, 2, 2, 2, 1, 3, 7, 6, 10, 2, 1, 3, 10,
      5, 2, 0, 9, 5, 4, 5, 4, 1, 8], device='cuda:0') labels:
tensor([ 3, 3, 9, 9, 6, 0, 1, 3, 8, 2, 3, 4, 7, 5, 4, 1, 3, 3,
      2, 1, 9, 1, 6, 10, 2, 2, 2, 7, 3, 7, 6, 10, 2, 1, 3, 3,
      5, 2, 0, 9, 5, 4, 5, 4, 1, 8], device='cuda:0')
eval_loss: 1.0695720911026
correct: 39
Accuracy of the network on the test images: 84 %

```

```

pred: tensor([ 2,  2,  3,  4,  2, 10,  1,  2,  0,  4,  3,  6,  9,  2,  6,  8,
10,  3,
           4,  1,  9,  3, 10,  0,  1,  6,  6,  1,  7,  3,  5,  9,  2,  3,  5,  2,
           5,  9,  1, 10,  1,  9,  4,  7,  4,  5], device='cuda:0') labels:
tensor([ 2,  2,  3,  3,  2,  3,  1,  2,  0,  4,  3,  6,  9,  2,  6,  8,  3,  3,
           4,  7,  9,  3, 10,  0,  1,  6,  5,  1,  7,  3,  5,  9,  2,  3,  1,  2,
           5,  9,  1, 10,  1,  8,  4,  7,  4,  5], device='cuda:0')
eval_loss: 0.989715576171875
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([10,  9,  5,  2,  8,  3,  4,  2,  6,  4,  3,  2,  2,  8,  4,  1,
1,  9,
           9,  1,  1,  1,  7,  3,  7, 10,  5,  0,  2,  2,  0,  2,  5,  5,  9,  6,
           10,  4,  9, 10,  6,  6,  1,  3,  3,  3], device='cuda:0') labels:
tensor([10,  9,  5,  2,  5,  3,  4,  2,  6,  4,  3,  2,  2,  8,  4,  1,  1,  9,
           9,  1,  1,  7,  7,  3,  7,  3,  5,  0,  2,  2,  0,  2,  1,  5,  8,  6,
           10,  4,  9,  3,  6,  3,  1,  3,  3,  3], device='cuda:0')
eval_loss: 0.9693500399589539
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 6,  7, 10,  4,  4,  8,  5,  9, 10,  9,  1,  4,  1,  9,  3,  3,
10,  6,
           4,  2,  2,  3,  1,  2, 10,  0,  2,  5,  1,  1,  2,  8,  6,  0,  2,  2,
           1,  9,  5,  9,  3,  3,  3,  7,  4,  5], device='cuda:0') labels:
tensor([ 6,  7,  3,  4,  4,  5,  1,  9,  3,  8,  1,  4,  1,  9,  3,  3, 10,  6,
           4,  2,  2,  3,  7,  2, 10,  0,  2,  5,  1,  1,  2,  8,  6,  0,  2,  2,
           1,  9,  5,  9,  3,  3,  3,  7,  3,  5], device='cuda:0')
eval_loss: 1.2273080348968506
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 3,  5,  4,  2,  2,  2,  7,  2,  2,  5,  2,  6,  0,  5,  9,  8,
1,  1,
           6,  3,  3,  3,  4,  1,  9,  8,  5,  9,  4,  3, 10,  3,  9,  1,  6,  7,
           10,  9,  3, 10,  0,  2,  1,  4,  1,  4], device='cuda:0') labels:
tensor([ 3,  5,  4,  2,  2,  2,  7,  2,  2,  5,  2,  6,  0,  5,  9,  8,  1,  1,
           6,  3,  3,  3,  4,  1,  9,  5,  1,  9,  4,  3, 10,  3,  9,  1,  6,  7,
           10,  8,  3,  3,  0,  2,  1,  4,  7,  3], device='cuda:0')
eval_loss: 0.6455234289169312
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 9,  6,  1,  1,  2,  9,  1,  2,  8,  1,  8,  0,  2,  1,  6,  9,
7,  4,
           10,  3,  2, 10,  9,  6,  2,  5,  1,  3,  5,  7,  4,  1, 10,  0,  2,  9,
           4,  4,  5,  1,  3, 10,  3,  2,  3,  6], device='cuda:0') labels:
tensor([ 9,  6,  1,  3,  2,  9,  1,  2,  5,  1,  8,  0,  2,  1,  3,  9,  7,  4,
           10,  3,  2,  3,  8,  6,  2,  5,  7,  3,  5,  7,  4,  1,  3,  0,  2,  9,
           4,  4,  5,  1,  3, 10,  3,  2,  3,  6], device='cuda:0')
eval_loss: 1.2703564167022705

```



```

correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 4,  9,  8,  6, 10,  1,  5,  3,  3,  1,  9,  6,  5,  1, 10,  5,
  9,  3,
            2, 10,  6,  7,  5,  3,  1,  2,  9,  4,  2,  2,  3,  0,  9,  4,  7,  4,
            8,  2,  1,  4,  2, 10,  2,  1,  3,  0], device='cuda:0') labels:
tensor([ 4,  9,  8,  6, 10,  7,  5,  3,  3,  1,  9,  6,  5,  1, 10,  1,  9,  3,
  2,  3,  6,  7,  5,  3,  1,  2,  8,  4,  2,  2,  3,  0,  9,  3,  7,  4,
  5,  2,  1,  4,  2,  3,  2,  1,  3,  0], device='cuda:0')
eval_loss: 1.1064441204071045
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 3,  6,  3,  6,  1,  9, 10,  4,  9,  2, 10,  2,  7,  2,  1,  2,
  9,  2,
            8,  3,  4,  1,  1,  1,  3,  2,  6,  3, 10,  4,  6,  0,  8,  2,  5,  1,
            6,  0,  3,  9, 10,  9,  5,  7,  5,  4], device='cuda:0') labels:
tensor([ 3,  6,  3,  3,  1,  9, 10,  4,  9,  2,  3,  2,  7,  2,  1,  2,  9,  2,
  8,  3,  4,  1,  1,  7,  3,  2,  6,  3, 10,  4,  6,  0,  5,  2,  5,  1,
  5,  0,  3,  8,  3,  9,  1,  7,  5,  4], device='cuda:0')
eval_loss: 1.0042834281921387
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([ 9,  3,  0, 10,  2,  9,  2,  5,  6,  4,  5,  1,  3,  9,  2,  0,
 10, 10,
            1,  7,  7,  9,  4,  5,  2,  4,  3,  9,  8,  4, 10,  3,  3,  6,  1,  1,
            8,  2,  6,  1,  1,  3,  2,  2,  4,  6], device='cuda:0') labels:
tensor([ 9,  3,  0,  3,  2,  9,  2,  5,  6,  3,  5,  7,  3,  9,  2,  0,  3, 10,
  1,  7,  7,  8,  4,  5,  2,  4,  3,  9,  5,  4, 10,  3,  3,  6,  1,  1,
  8,  2,  1,  1,  1,  3,  2,  2,  4,  6], device='cuda:0')
eval_loss: 0.9873523116111755
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 0,  3,  3,  9,  1,  5,  1,  8, 10,  5,  9,  1,  0,  9,  4,  4,
  2,  3,
            4, 10,  1,  2,  6,  7,  5,  3,  4,  3,  6,  3,  2,  9, 10,  6,  5,  2,
            1,  7,  4, 10,  8,  2,  9,  1,  2,  2], device='cuda:0') labels:
tensor([ 0,  3,  3,  9,  7,  1,  1,  8, 10,  5,  9,  1,  0,  9,  3,  4,  2,  3,
  4,  3,  1,  2,  6,  7,  5,  3,  4,  3,  6,  3,  2,  8, 10,  6,  5,  2,
  1,  7,  4,  3,  5,  2,  9,  1,  2,  2], device='cuda:0')
eval_loss: 0.9499732255935669
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 1,  9,  9,  3,  6,  3,  4,  3,  2,  2,  1,  3,  0,  8,  4,  1,
  1,  1,
            6,  6,  7,  4,  5,  2,  9,  9,  6, 10,  3,  4,  7,  0,  2,  9,  4,  2,
            2, 10,  8,  6, 10,  7,  2,  1,  3,  5], device='cuda:0') labels:
tensor([ 1,  9,  9,  3,  1,  3,  4,  3,  2,  2,  7,  3,  0,  5,  3,  1,  1,  1,
  6,  6,  7,  4,  5,  2,  9,  8,  6, 10,  3,  4,  7,  0,  2,  9,  4,  2,

```

```

        2, 3, 8, 5, 3, 10, 2, 1, 3, 5], device='cuda:0')
eval_loss: 0.9587622284889221
correct: 37
Accuracy of the network on the test images: 80 %
pred: tensor([ 2, 10, 9, 8, 2, 1, 9, 7, 2, 1, 3, 2, 6, 5, 6, 4,
1, 3,
        2, 8, 6, 3, 10, 3, 1, 9, 2, 9, 4, 1, 4, 0, 1, 9, 7, 3,
        0, 3, 5, 10, 4, 4, 0, 5, 2, 5], device='cuda:0') labels:
tensor([ 2, 10, 9, 8, 2, 1, 9, 7, 2, 1, 3, 2, 6, 5, 6, 4, 7, 3,
        2, 5, 6, 3, 3, 3, 1, 9, 2, 8, 4, 1, 3, 0, 1, 9, 7, 3,
        3, 3, 5, 10, 4, 4, 0, 1, 2, 5], device='cuda:0')
eval_loss: 0.9683318138122559
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 4, 1, 1, 6, 3, 8, 6, 10, 10, 2, 9, 3, 3, 1, 2, 4,
9, 0,
        1, 2, 2, 2, 1, 3, 5, 9, 10, 6, 4, 3, 2, 5, 3, 7, 10, 5,
        9, 2, 9, 4, 7, 5, 0, 4, 1, 1], device='cuda:0') labels:
tensor([ 4, 1, 1, 6, 3, 8, 6, 10, 3, 2, 8, 3, 3, 1, 2, 3, 9, 0,
        1, 2, 2, 2, 7, 3, 5, 9, 10, 6, 4, 3, 2, 5, 3, 7, 3, 5,
        9, 2, 9, 4, 7, 5, 0, 4, 1, 1], device='cuda:0')
eval_loss: 0.7767335772514343
correct: 41
Accuracy of the network on the test images: 89 %
pred: tensor([ 4, 4, 6, 9, 1, 9, 0, 9, 5, 6, 1, 3, 2, 10, 0, 6,
10, 5,
        2, 2, 2, 3, 4, 1, 9, 2, 0, 8, 1, 9, 3, 5, 3, 3, 7, 5,
        4, 3, 3, 1, 7, 2, 2, 5, 6, 1], device='cuda:0') labels:
tensor([ 4, 4, 3, 8, 1, 9, 0, 9, 5, 6, 1, 3, 2, 10, 3, 6, 10, 1,
        2, 2, 2, 3, 4, 1, 9, 2, 0, 8, 1, 9, 3, 5, 3, 3, 7, 5,
        4, 3, 3, 1, 7, 2, 2, 5, 6, 7], device='cuda:0')
eval_loss: 0.7791065573692322
correct: 41
Accuracy of the network on the test images: 89 %
pred: tensor([ 7, 8, 0, 4, 1, 6, 2, 2, 2, 4, 8, 1, 2, 9, 3, 7,
4, 3,
        1, 6, 2, 5, 4, 10, 5, 4, 10, 9, 3, 3, 5, 1, 1, 6, 1, 0,
        2, 9, 9, 10, 3, 3, 10, 9, 2, 5], device='cuda:0') labels:
tensor([ 7, 5, 0, 4, 1, 6, 2, 2, 2, 3, 8, 1, 2, 8, 3, 7, 4, 3,
        1, 6, 2, 5, 4, 10, 5, 4, 3, 9, 3, 3, 1, 1, 7, 6, 1, 0,
        2, 9, 9, 10, 3, 3, 3, 9, 2, 5], device='cuda:0')
eval_loss: 1.2866805791854858
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 4, 3, 5, 1, 3, 10, 10, 2, 2, 10, 4, 0, 5, 8, 6, 8,
0, 4,
        10, 5, 2, 9, 2, 7, 9, 2, 1, 1, 1, 2, 4, 1, 7, 1, 1, 3,
        6, 3, 3, 6, 9, 9, 4, 2, 3, 9], device='cuda:0') labels:

```

```

tensor([ 4,  3,  5,  1,  3,  3, 10,  2,  2,  3,  4,  0,  5,  5,  6,  8,  0,  3,
         10,  5,  2,  9,  2,  7,  9,  2,  1,  1,  1,  2,  4,  1,  7,  7,  1,  3,
         6,  3,  3,  6,  8,  9,  4,  2,  3,  9], device='cuda:0')
eval_loss: 0.5835800170898438
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 3,  5,  1,  5,  9,  7,  7,  4,  3,  3,  1,  6,  9,  4, 10,  1,
               3,  0,
               2,  6,  0,  9,  2,  2,  3,  5,  4,  2,  1, 10,  2,  8,  4,  9,  2,  6,
               1,  9,  2,  1,  3, 10,  8,  1,  4, 10], device='cuda:0') labels:
tensor([ 3,  5,  1,  5,  9,  7,  7,  3,  3,  3,  1,  6,  9,  4,  3,  1,  3,  0,
         2,  6,  0,  9,  2,  2,  3,  5,  4,  2,  1,  3,  2,  5,  4,  8,  2,  6,
         1,  9,  2,  1,  3, 10,  8,  7,  4, 10], device='cuda:0')
eval_loss: 1.0550540685653687
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1,  8,  8,  7,  2,  9,  2,  5,  2,  3,  1,  2,  3,  3,  2,  0,
               4, 10,
               3,  1,  5,  3,  6,  1,  2,  1,  7,  1,  3,  9,  3, 10,  5, 10,  2,  3,
               6,  6,  9,  4,  9,  4,  4,  5,  9,  0], device='cuda:0') labels:
tensor([ 1,  8,  5,  7,  2,  9,  2,  5,  2,  3,  1,  2,  6,  3,  2,  0,  4, 10,
         3,  7,  1,  3,  6,  1,  2,  1,  7,  1,  3,  9,  3,  3,  5, 10,  2,  3,
         6,  3,  8,  4,  9,  4,  4,  5,  9,  0], device='cuda:0')
eval_loss: 1.2115460634231567
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 6,  5,  6,  9,  1,  9,  1,  6,  8,  2,  2, 10,  4, 10,  2, 10,
               2,  4,
               1,  2,  0,  4,  3,  3,  1,  9, 10,  2,  7,  4,  5,  3,  1,  9,  0,  3,
               5,  1,  8,  9,  4,  3,  1,  7,  2,  3], device='cuda:0') labels:
tensor([ 6,  5,  6,  9,  1,  9,  1,  6,  8,  2,  2, 10,  3,  3,  2,  3,  2,  4,
         1,  2,  0,  4,  3,  3,  1,  8, 10,  2,  7,  4,  5,  3,  7,  9,  0,  3,
         5,  1,  5,  9,  4,  3,  1,  7,  2,  3], device='cuda:0')
eval_loss: 0.9052700996398926
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1,  1,  7,  8,  9,  0, 10,  9,  0,  3,  9,  5,  4,  5,  2,  6,
               6,  3,
               7, 10,  4,  9,  4,  3,  3,  2,  2,  3,  3,  1,  4,  9,  2,  8,  2,  5,
               1,  1,  2,  1,  2,  4, 10,  6, 10,  1], device='cuda:0') labels:
tensor([ 1,  1,  7,  5,  9,  0,  3,  9,  0,  3,  9,  5,  4,  5,  2,  6,  6,  3,
         7, 10,  4,  9,  4,  3,  3,  2,  2,  3,  3,  1,  3,  8,  2,  8,  2,  5,
         1,  1,  2,  1,  2,  4, 10,  6,  3,  7], device='cuda:0')
eval_loss: 0.9134114980697632
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1, 10,  1,  2,  8,  3,  6,  8,  3,  5,  5,  6,  5,  9, 10,  4,
               3,  2,

```

```

        7, 4, 1, 5, 2, 6, 0, 9, 10, 9, 6, 6, 1, 0, 3, 2, 10, 1,
        9, 3, 2, 2, 4, 4, 3, 7, 9, 2], device='cuda:0') labels:
tensor([ 1, 10, 1, 2, 5, 3, 1, 8, 3, 1, 5, 6, 5, 9, 10, 4, 3, 2,
        7, 4, 7, 5, 2, 6, 0, 9, 3, 9, 3, 6, 1, 0, 3, 2, 3, 1,
        9, 3, 2, 2, 4, 4, 3, 7, 8, 2], device='cuda:0')
eval_loss: 0.9890806674957275
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([ 2, 1, 7, 2, 1, 4, 3, 4, 0, 7, 6, 5, 9, 2, 5, 10,
3, 3,
        4, 4, 9, 9, 10, 4, 9, 6, 5, 1, 0, 3, 0, 8, 6, 2, 1, 1,
        10, 9, 1, 5, 10, 2, 3, 2, 1, 2], device='cuda:0') labels:
tensor([ 2, 7, 7, 2, 1, 3, 3, 4, 3, 7, 6, 5, 9, 2, 5, 3, 3, 3,
        4, 4, 9, 8, 10, 4, 9, 6, 5, 1, 0, 3, 0, 8, 6, 2, 1, 1,
        3, 9, 1, 5, 10, 2, 3, 2, 1, 2], device='cuda:0')
eval_loss: 0.709557831287384
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1, 4, 10, 2, 5, 4, 4, 1, 0, 3, 9, 2, 1, 10, 9, 0,
10, 5,
        6, 2, 9, 5, 10, 8, 7, 3, 9, 1, 2, 6, 2, 4, 3, 2, 2, 1,
        6, 3, 3, 6, 1, 7, 9, 5, 3, 1], device='cuda:0') labels:
tensor([ 1, 4, 10, 2, 5, 4, 4, 1, 0, 3, 9, 2, 1, 3, 9, 0, 3, 5,
        6, 2, 9, 5, 10, 8, 7, 3, 8, 1, 2, 6, 2, 4, 3, 2, 2, 1,
        3, 3, 3, 6, 1, 7, 9, 5, 3, 7], device='cuda:0')
eval_loss: 0.7428016066551208
correct: 41
Accuracy of the network on the test images: 89 %
pred: tensor([ 4, 4, 5, 1, 2, 0, 1, 1, 9, 6, 10, 6, 3, 2, 5, 9,
0, 1,
        8, 9, 9, 9, 7, 2, 5, 1, 2, 3, 4, 4, 3, 2, 10, 10, 2, 8,
        6, 1, 7, 3, 2, 10, 3, 3, 5, 6], device='cuda:0') labels:
tensor([ 4, 4, 5, 1, 2, 0, 1, 7, 8, 6, 10, 6, 3, 2, 5, 9, 0, 1,
        5, 9, 9, 9, 7, 2, 5, 1, 2, 3, 4, 4, 3, 2, 10, 3, 2, 8,
        6, 1, 7, 3, 2, 3, 3, 3, 1, 3], device='cuda:0')
eval_loss: 1.166865348815918
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 8, 2, 3, 3, 9, 2, 4, 4, 1, 4, 2, 2, 10, 2, 7, 5,
9, 1,
        1, 0, 5, 10, 1, 6, 3, 5, 0, 9, 1, 3, 2, 1, 9, 10, 8, 9,
        10, 4, 7, 0, 6, 5, 6, 4, 2, 3], device='cuda:0') labels:
tensor([ 5, 2, 3, 3, 9, 2, 4, 3, 1, 4, 2, 2, 3, 2, 7, 1, 9, 1,
        1, 0, 5, 10, 7, 6, 3, 5, 3, 8, 1, 3, 2, 1, 9, 3, 8, 9,
        10, 4, 7, 0, 6, 5, 6, 4, 2, 3], device='cuda:0')
eval_loss: 1.1852713823318481
correct: 38
Accuracy of the network on the test images: 82 %

```

```

pred: tensor([ 3, 1, 1, 4, 0, 5, 0, 3, 6, 10, 6, 9, 2, 1, 8, 8,
4, 9,
          7, 3, 2, 3, 2, 3, 0, 9, 2, 1, 10, 5, 5, 10, 7, 4, 3, 6,
9, 1, 4, 5, 4, 2, 2, 9, 1, 2], device='cuda:0') labels:
tensor([ 3, 1, 7, 4, 0, 5, 3, 3, 6, 3, 6, 9, 2, 1, 8, 5, 4, 9,
7, 3, 2, 3, 2, 3, 0, 8, 2, 1, 10, 1, 5, 10, 7, 4, 3, 6,
9, 1, 3, 5, 4, 2, 2, 9, 1, 2], device='cuda:0')
eval_loss: 0.9950085282325745
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 5, 3, 4, 0, 2, 5, 2, 1, 4, 1, 9, 5, 8, 6, 3, 9,
3, 9,
          1, 8, 3, 2, 4, 0, 2, 2, 6, 1, 8, 7, 3, 6, 2, 10, 10, 9,
4, 7, 3, 10, 9, 5, 1, 4, 0, 2], device='cuda:0') labels:
tensor([ 1, 3, 3, 3, 2, 5, 2, 1, 4, 7, 9, 5, 8, 6, 3, 9, 3, 9,
1, 5, 3, 2, 4, 0, 2, 2, 6, 1, 6, 7, 3, 1, 2, 10, 3, 8,
4, 7, 3, 10, 9, 5, 1, 4, 0, 2], device='cuda:0')
eval_loss: 1.6609522104263306
correct: 37
Accuracy of the network on the test images: 80 %
pred: tensor([ 7, 8, 1, 9, 5, 9, 2, 3, 3, 1, 2, 0, 2, 2, 6, 9,
3, 4,
          3, 1, 4, 3, 1, 4, 8, 2, 1, 0, 2, 6, 0, 4, 5, 10, 2, 3,
4, 10, 5, 1, 10, 7, 6, 1, 9, 9], device='cuda:0') labels:
tensor([ 7, 5, 1, 9, 5, 9, 2, 3, 3, 1, 2, 0, 2, 2, 6, 9, 3, 4,
3, 7, 4, 3, 1, 4, 8, 2, 1, 3, 2, 6, 0, 3, 5, 10, 2, 3,
4, 3, 5, 1, 10, 7, 6, 1, 8, 9], device='cuda:0')
eval_loss: 0.9517163634300232
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 9, 3, 4, 1, 9, 3, 4, 10, 8, 6, 4, 3, 3, 3, 6, 2,
2, 1,
          5, 0, 8, 9, 2, 1, 1, 10, 5, 9, 7, 2, 4, 10, 3, 2, 3, 1,
5, 1, 2, 7, 1, 6, 4, 9, 0, 2], device='cuda:0') labels:
tensor([ 9, 3, 4, 1, 8, 3, 4, 10, 8, 6, 4, 3, 3, 3, 6, 2, 2, 7,
5, 0, 5, 9, 2, 1, 1, 3, 5, 9, 7, 2, 3, 10, 3, 2, 3, 1,
5, 1, 2, 7, 1, 6, 4, 9, 0, 2], device='cuda:0')
eval_loss: 0.8818397521972656
correct: 41
Accuracy of the network on the test images: 89 %
pred: tensor([ 3, 4, 2, 7, 4, 5, 2, 6, 2, 0, 7, 1, 2, 1, 3, 2,
6, 4,
          9, 8, 4, 4, 9, 3, 3, 5, 10, 0, 3, 2, 5, 3, 3, 8, 10, 6,
9, 10, 1, 1, 9, 9, 2, 1, 5, 1], device='cuda:0') labels:
tensor([ 3, 3, 2, 7, 4, 5, 2, 6, 2, 0, 7, 1, 2, 7, 3, 2, 6, 4,
9, 8, 4, 4, 9, 3, 3, 1, 3, 0, 3, 2, 5, 3, 3, 5, 10, 6,
8, 10, 1, 1, 9, 9, 2, 1, 5, 1], device='cuda:0')
eval_loss: 1.0131107568740845

```

```

correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 2, 10, 5, 7, 0, 3, 6, 2, 4, 8, 3, 7, 9, 10, 4, 9,
1, 4,
          2, 3, 4, 6, 6, 3, 10, 0, 5, 2, 1, 4, 2, 9, 9, 1, 3, 9,
          5, 1, 8, 2, 10, 1, 10, 2, 1, 6], device='cuda:0') labels:
tensor([ 2, 3, 5, 7, 0, 3, 6, 2, 4, 5, 3, 7, 9, 3, 4, 9, 1, 3,
          2, 3, 4, 6, 6, 3, 10, 0, 5, 2, 7, 4, 2, 8, 9, 1, 3, 9,
          1, 1, 8, 2, 10, 1, 3, 2, 1, 5], device='cuda:0')
eval_loss: 0.7497124075889587
correct: 37
Accuracy of the network on the test images: 80 %
pred: tensor([ 9, 2, 3, 1, 8, 2, 4, 8, 10, 9, 9, 2, 6, 4, 1, 3,
10, 0,
          9, 6, 0, 3, 3, 5, 0, 3, 4, 2, 3, 1, 1, 2, 1, 1, 5, 5,
          5, 9, 6, 2, 10, 4, 2, 6, 7, 7], device='cuda:0') labels:
tensor([ 9, 2, 3, 1, 5, 2, 4, 8, 10, 8, 9, 2, 3, 4, 1, 3, 3, 0,
          9, 6, 3, 3, 3, 5, 0, 3, 4, 2, 3, 1, 1, 2, 7, 1, 5, 5,
          1, 9, 6, 2, 10, 4, 2, 6, 7, 7], device='cuda:0')
eval_loss: 0.977796733379364
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 8, 1, 0, 3, 1, 6, 1, 3, 5, 3, 4, 10, 2, 4, 5, 6,
3, 9,
          9, 9, 2, 10, 2, 2, 3, 4, 10, 2, 8, 6, 7, 4, 6, 7, 5, 9,
          2, 1, 1, 1, 2, 6, 3, 3, 9, 0], device='cuda:0') labels:
tensor([ 5, 7, 0, 3, 1, 6, 1, 3, 5, 3, 4, 3, 2, 4, 5, 6, 3, 9,
          9, 9, 2, 10, 2, 2, 3, 4, 10, 2, 8, 6, 7, 4, 5, 7, 1, 8,
          2, 1, 1, 1, 2, 3, 3, 3, 9, 0], device='cuda:0')
eval_loss: 1.1122941970825195
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 0, 2, 9, 3, 6, 5, 1, 4, 2, 4, 1, 10, 1, 4, 9, 1,
8, 10,
          3, 6, 7, 2, 4, 2, 9, 3, 6, 2, 10, 2, 1, 9, 1, 8, 0, 3,
          10, 5, 3, 2, 4, 5, 3, 1, 9, 7], device='cuda:0') labels:
tensor([ 0, 2, 9, 3, 6, 5, 1, 4, 2, 4, 1, 10, 7, 3, 8, 1, 8, 3,
          3, 6, 7, 2, 4, 2, 9, 3, 6, 2, 3, 2, 1, 9, 1, 5, 0, 3,
          10, 5, 3, 2, 4, 5, 3, 1, 9, 7], device='cuda:0')
eval_loss: 0.904826283454895
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 9, 2, 3, 2, 1, 2, 3, 6, 8, 1, 2, 2, 1, 4, 9, 3,
1, 0,
          5, 2, 10, 6, 7, 6, 7, 3, 5, 9, 4, 0, 6, 2, 5, 1, 4, 10,
          9, 8, 1, 10, 3, 4, 5, 9, 3, 3], device='cuda:0') labels:
tensor([ 9, 2, 3, 2, 1, 2, 3, 6, 8, 1, 2, 2, 1, 4, 9, 3, 1, 0,
          5, 2, 10, 3, 7, 6, 7, 3, 1, 8, 4, 0, 6, 2, 5, 1, 4, 10,

```

```

    9, 5, 7, 3, 3, 4, 5, 9, 3, 3], device='cuda:0')
eval_loss: 0.6667746901512146
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 4, 3, 10, 2, 3, 10, 9, 9, 1, 2, 2, 2, 2, 7, 4, 1,
1, 10,
            0, 0, 9, 3, 1, 6, 10, 2, 3, 5, 4, 1, 6, 4, 3, 4, 8, 6,
            5, 5, 1, 8, 5, 9, 2, 7, 3, 9], device='cuda:0') labels:
tensor([ 4, 3, 3, 2, 3, 10, 9, 9, 1, 2, 2, 2, 2, 7, 4, 7, 1, 3,
0, 0, 9, 3, 1, 6, 10, 2, 3, 5, 3, 1, 6, 4, 3, 4, 5, 6,
            5, 5, 1, 8, 1, 8, 2, 7, 3, 9], device='cuda:0')
eval_loss: 1.144700288772583
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 3, 1, 10, 2, 5, 4, 2, 1, 3, 3, 5, 2, 1, 0, 4, 3,
2, 0,
            4, 10, 7, 6, 7, 10, 2, 4, 5, 6, 9, 9, 9, 6, 1, 8, 5, 3,
            5, 1, 4, 9, 9, 2, 10, 3, 1, 2], device='cuda:0') labels:
tensor([ 3, 1, 10, 2, 1, 4, 2, 1, 3, 3, 5, 2, 1, 0, 4, 3, 2, 0,
4, 10, 7, 6, 7, 3, 2, 3, 5, 6, 9, 9, 9, 6, 1, 8, 5, 3,
            5, 7, 4, 9, 8, 2, 3, 3, 1, 2], device='cuda:0')
eval_loss: 0.994540810585022
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 7, 4, 2, 9, 2, 3, 6, 5, 6, 10, 3, 8, 9, 5, 9, 10,
4, 9,
            0, 1, 4, 1, 1, 2, 3, 7, 2, 5, 2, 9, 2, 4, 3, 10, 2, 10,
            1, 4, 6, 1, 5, 5, 1, 0, 3, 3], device='cuda:0') labels:
tensor([ 7, 4, 2, 8, 2, 3, 6, 5, 6, 10, 3, 8, 9, 5, 9, 3, 4, 9,
0, 7, 4, 1, 1, 2, 3, 7, 2, 5, 2, 9, 2, 4, 3, 10, 2, 3,
            1, 3, 6, 1, 5, 1, 1, 0, 3, 3], device='cuda:0')
eval_loss: 0.7889291644096375
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 7, 10, 2, 4, 9, 1, 1, 7, 4, 2, 3, 0, 0, 4, 2, 2,
5, 3,
            3, 1, 6, 5, 6, 6, 10, 10, 4, 9, 2, 9, 10, 3, 1, 1, 3, 1,
            3, 6, 2, 5, 9, 8, 2, 4, 9, 5], device='cuda:0') labels:
tensor([ 7, 3, 2, 4, 8, 7, 1, 7, 3, 2, 3, 0, 0, 4, 2, 2, 5, 3,
3, 1, 6, 5, 6, 1, 10, 3, 4, 9, 2, 9, 10, 3, 1, 1, 3, 1,
            3, 6, 2, 5, 9, 8, 2, 4, 9, 5], device='cuda:0')
eval_loss: 0.886530339717865
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 6, 4, 5, 4, 3, 10, 10, 3, 2, 1, 9, 7, 10, 2, 6, 2,
5, 4,
            3, 5, 9, 2, 3, 10, 2, 9, 9, 7, 5, 1, 1, 3, 6, 4, 4, 3,
            0, 2, 8, 5, 1, 0, 2, 1, 9, 1], device='cuda:0') labels:

```

```

tensor([ 6,  4,  5,  4,  3, 10, 10,  3,  2,  1,  9,  7,  3,  2,  6,  2,  5,  4,
         3,  5,  9,  2,  3,  3,  2,  9,  8,  7,  1,  7,  1,  3,  6,  3,  4,  3,
         0,  2,  8,  5,  1,  0,  2,  1,  9,  1], device='cuda:0')
eval_loss: 0.8952721953392029
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1,  2,  0,  3,  4,  9,  2,  6,  3,  3,  2,  4,  9,  8,  9,  1,
               2,  1,
               10,  4,  5,  2,  3,  7,  5,  2,  5,  6,  3,  8, 10,  3,  7, 10,  7,  1,
               1,  4,  9,  0,  9,  5,  4,  2,  3, 10], device='cuda:0') labels:
tensor([ 1,  2,  0,  6,  4,  8,  2,  6,  3,  3,  2,  3,  9,  5,  9,  1,  2,  1,
         3,  4,  1,  2,  3,  7,  5,  2,  5,  6,  3,  8, 10,  3,  7,  3,  7,  1,
         1,  4,  9,  0,  9,  5,  4,  2,  3, 10], device='cuda:0')
eval_loss: 0.9720498919487
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 0,  3,  7,  5,  2,  2,  0,  5,  3,  1,  9,  8,  4,  5,  3,  2,
               1,  2,
               1, 10,  7,  3,  1,  6,  5,  1,  3,  6,  2, 10,  3,  9,  6,  9,  4, 10,
               7,  9,  2,  6,  4,  4,  4,  9,  2,  0], device='cuda:0') labels:
tensor([ 0,  3,  7,  5,  2,  2,  0,  1,  3,  1,  9,  8,  4,  5,  3,  2,  1,  2,
         1, 10,  7,  3,  1,  6,  5,  1,  3,  6,  2,  3,  3,  9,  5,  9,  4, 10,
         7,  9,  2,  6,  3,  4,  4,  8,  2,  3], device='cuda:0')
eval_loss: 0.6598302721977234
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1,  5,  3,  1,  5,  3, 10,  7,  2, 10,  0,  7,  9,  2,  2,  1,
               2,  3,
               1,  4,  4,  6,  9,  4,  3,  8,  9,  2,  3,  6,  0,  3, 10,  9,  0,  1,
               9,  1,  4,  5,  2,  2,  6,  8,  5,  4], device='cuda:0') labels:
tensor([ 1,  5,  3,  1,  5,  3, 10,  7,  2,  3,  0,  7,  9,  2,  2,  1,  2,  3,
         1,  4,  4,  6,  9,  4,  3,  8,  9,  2,  3,  6,  3,  3, 10,  9,  0,  7,
         8,  1,  4,  1,  2,  2,  6,  5,  5,  3], device='cuda:0')
eval_loss: 1.1582030057907104
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([10,  8,  9,  4,  6,  3,  3,  3,  1,  5,  0,  2, 10,  4,  9,  4,
               3,  2,
               2,  2,  6,  2,  3,  2,  3,  1,  3, 10,  6,  1,  1,  4,  1,  4,  9,  9,
               0,  2,  9,  8,  1,  5,  7,  7,  5,  1], device='cuda:0') labels:
tensor([10,  5,  9,  4,  6,  3,  3,  3,  1,  5,  0,  2,  3,  4,  9,  3,  3,  2,
         2,  2,  6,  2,  3,  2,  3,  1,  3, 10,  6,  1,  1,  4,  1,  4,  9,  8,
         0,  2,  9,  8,  7,  5,  7,  7,  5,  1], device='cuda:0')
eval_loss: 1.2473064661026
correct: 41
Accuracy of the network on the test images: 89 %
pred: tensor([ 1,  8,  0,  2,  3,  9,  6,  1,  6,  5,  8,  1,  6,  1,  6, 10,
               5,  4,

```



```

    0, 3, 2, 4, 9, 4, 2, 10, 5, 9, 4, 5, 3, 2, 3, 2, 1, 7,
    7, 3, 2, 0, 2, 1, 9, 9, 3, 3], device='cuda:0') labels:
tensor([ 1, 8, 3, 2, 3, 9, 6, 1, 6, 5, 5, 1, 6, 1, 3, 10, 5, 4,
        0, 3, 2, 4, 9, 4, 2, 10, 5, 9, 4, 1, 3, 2, 3, 2, 1, 7,
        7, 3, 2, 0, 2, 7, 9, 8, 3, 3], device='cuda:0')
eval_loss: 1.0966007709503174
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 6, 4, 3, 2, 8, 9, 9, 7, 2, 3, 1, 3, 0, 3, 2, 10,
        3, 2,
        7, 2, 1, 0, 1, 6, 6, 5, 8, 3, 4, 9, 9, 2, 10, 6, 5, 5,
        9, 3, 4, 2, 4, 1, 1, 5, 10, 5], device='cuda:0') labels:
tensor([ 3, 4, 3, 2, 8, 9, 9, 7, 2, 3, 1, 3, 0, 3, 2, 3, 3, 2,
        7, 2, 1, 0, 1, 6, 6, 1, 5, 3, 4, 9, 9, 2, 10, 6, 5, 1,
        8, 3, 4, 2, 4, 1, 7, 5, 10, 5], device='cuda:0')
eval_loss: 1.0029431581497192
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 4, 4, 1, 10, 2, 1, 4, 2, 4, 1, 10, 5, 2, 2, 3, 2,
        2, 3,
        10, 9, 9, 0, 4, 10, 7, 8, 6, 3, 5, 3, 1, 9, 5, 2, 7, 6,
        9, 0, 9, 3, 10, 8, 1, 6, 1, 1], device='cuda:0') labels:
tensor([ 4, 4, 1, 3, 2, 1, 4, 2, 4, 1, 3, 5, 2, 2, 3, 2, 2, 3,
        10, 9, 8, 0, 3, 10, 7, 8, 6, 3, 5, 3, 7, 9, 5, 2, 7, 6,
        9, 0, 9, 3, 3, 5, 1, 6, 1, 1], device='cuda:0')
eval_loss: 0.8158286809921265
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 1, 4, 1, 1, 9, 4, 6, 6, 3, 6, 2, 5, 9, 3, 10, 8,
        3, 0,
        3, 5, 10, 4, 7, 9, 2, 2, 2, 10, 8, 4, 1, 0, 10, 1, 4, 2,
        0, 5, 9, 5, 1, 2, 2, 7, 3, 9], device='cuda:0') labels:
tensor([ 1, 4, 1, 1, 9, 4, 6, 6, 3, 6, 2, 5, 9, 3, 3, 8, 3, 3,
        3, 5, 10, 3, 7, 9, 2, 2, 2, 10, 5, 4, 1, 0, 3, 7, 4, 2,
        0, 1, 8, 5, 1, 2, 2, 7, 3, 9], device='cuda:0')
eval_loss: 1.0669814348220825
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([ 9, 4, 5, 0, 2, 6, 3, 4, 5, 9, 3, 2, 2, 7, 2, 7,
        3, 1,
        6, 8, 3, 1, 4, 3, 10, 1, 1, 4, 2, 1, 10, 6, 5, 9, 2, 0,
        4, 9, 8, 6, 2, 9, 3, 3, 1, 10], device='cuda:0') labels:
tensor([ 9, 4, 1, 0, 2, 6, 3, 3, 5, 9, 3, 2, 2, 7, 2, 7, 3, 1,
        5, 8, 3, 1, 4, 3, 3, 1, 7, 4, 2, 1, 10, 6, 5, 8, 2, 0,
        4, 9, 5, 6, 2, 9, 3, 3, 1, 10], device='cuda:0')
eval_loss: 1.0782134532928467
correct: 39
Accuracy of the network on the test images: 84 %

```

```

pred: tensor([ 3, 0, 1, 6, 2, 4, 10, 7, 1, 3, 1, 9, 3, 1, 7, 4,
9, 4,
          5, 4, 2, 3, 4, 10, 8, 9, 6, 2, 8, 2, 5, 1, 2, 8, 3, 9,
          0, 3, 10, 0, 9, 2, 1, 1, 2, 5], device='cuda:0') labels:
tensor([ 3, 3, 1, 6, 2, 4, 10, 7, 1, 3, 1, 9, 3, 1, 7, 4, 9, 4,
          5, 3, 2, 3, 4, 3, 8, 9, 6, 2, 6, 2, 5, 1, 2, 5, 3, 8,
          0, 3, 10, 0, 9, 2, 7, 1, 2, 5], device='cuda:0')
eval_loss: 1.204832911491394
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([10, 2, 4, 2, 10, 1, 9, 6, 6, 3, 2, 4, 4, 4, 8, 5,
5, 1,
          1, 2, 2, 10, 9, 1, 3, 6, 5, 0, 3, 7, 0, 3, 5, 7, 8, 1,
          9, 0, 9, 1, 3, 9, 2, 2, 3, 4], device='cuda:0') labels:
tensor([10, 2, 3, 2, 3, 1, 9, 6, 6, 3, 2, 4, 4, 4, 5, 5, 5, 1,
          1, 2, 2, 10, 9, 7, 3, 6, 1, 0, 3, 7, 0, 3, 5, 7, 8, 1,
          8, 3, 9, 1, 3, 9, 2, 2, 3, 4], device='cuda:0')
eval_loss: 0.8469946384429932
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 2, 8, 3, 2, 6, 0, 4, 9, 7, 10, 3, 6, 10, 4, 0, 4,
5, 1,
          9, 2, 4, 8, 2, 5, 9, 1, 1, 2, 5, 3, 3, 7, 10, 3, 9, 1,
          1, 3, 0, 6, 2, 2, 1, 1, 4, 9], device='cuda:0') labels:
tensor([ 2, 8, 3, 2, 6, 0, 3, 8, 7, 3, 3, 6, 10, 4, 0, 4, 5, 1,
          9, 2, 4, 5, 2, 5, 9, 1, 1, 2, 5, 3, 3, 7, 10, 3, 9, 7,
          1, 3, 3, 6, 2, 2, 1, 1, 4, 9], device='cuda:0')
eval_loss: 0.6873779892921448
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 0, 2, 4, 6, 3, 2, 9, 3, 9, 6, 1, 7, 5, 10, 7, 1,
1, 4,
          5, 3, 6, 10, 3, 5, 2, 1, 2, 0, 9, 4, 3, 9, 4, 10, 9, 5,
          2, 3, 1, 8, 3, 2, 2, 8, 1, 4], device='cuda:0') labels:
tensor([ 0, 2, 4, 6, 3, 2, 9, 3, 9, 6, 7, 7, 5, 3, 7, 1, 1, 3,
          1, 3, 6, 10, 3, 5, 2, 1, 2, 0, 9, 4, 3, 8, 4, 10, 9, 5,
          2, 3, 1, 8, 3, 2, 2, 5, 1, 4], device='cuda:0')
eval_loss: 0.8458244204521179
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 8, 7, 1, 8, 1, 9, 3, 3, 1, 6, 10, 0, 4, 2, 10, 2,
4, 2,
          6, 6, 9, 3, 2, 3, 1, 0, 9, 10, 3, 6, 4, 5, 7, 4, 2, 9,
          3, 9, 10, 1, 5, 2, 2, 1, 1, 5], device='cuda:0') labels:
tensor([ 5, 7, 1, 8, 7, 8, 3, 3, 1, 6, 10, 0, 4, 2, 10, 2, 4, 2,
          3, 6, 9, 3, 2, 3, 1, 0, 9, 3, 3, 6, 4, 5, 7, 4, 2, 9,
          3, 9, 3, 1, 5, 2, 2, 1, 1, 5], device='cuda:0')
eval_loss: 0.8719325065612793

```

```

correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([10, 8, 0, 6, 4, 6, 9, 4, 1, 2, 4, 6, 10, 1, 5, 3,
9, 1,
          2, 7, 3, 10, 5, 2, 3, 2, 8, 9, 3, 0, 3, 1, 6, 7, 1, 0,
          4, 1, 9, 4, 3, 2, 2, 5, 2, 9], device='cuda:0') labels:
tensor([ 3, 5, 0, 6, 4, 5, 9, 3, 1, 2, 4, 6, 10, 1, 1, 3, 8, 1,
          2, 7, 3, 10, 5, 2, 3, 2, 8, 9, 3, 3, 3, 1, 6, 7, 1, 0,
          4, 7, 9, 4, 3, 2, 2, 5, 2, 9], device='cuda:0')
eval_loss: 1.162832260131836
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([10, 3, 7, 10, 6, 1, 2, 4, 0, 2, 8, 0, 9, 8, 2, 9,
1, 9,
          5, 1, 4, 4, 3, 2, 10, 2, 3, 1, 5, 2, 2, 10, 9, 1, 1, 6,
          3, 7, 1, 3, 4, 6, 9, 6, 4, 3], device='cuda:0') labels:
tensor([10, 3, 7, 3, 5, 1, 2, 4, 0, 2, 5, 0, 9, 8, 2, 9, 7, 8,
          5, 1, 4, 3, 3, 2, 10, 2, 3, 1, 5, 2, 2, 3, 9, 1, 1, 6,
          3, 7, 1, 3, 4, 6, 9, 6, 4, 3], device='cuda:0')
eval_loss: 1.0650361776351929
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 0, 2, 9, 1, 4, 2, 5, 6, 3, 3, 1, 6, 2, 1, 1, 7,
0, 7,
          4, 1, 2, 3, 2, 8, 9, 9, 1, 2, 5, 4, 2, 5, 10, 3, 3, 6,
          4, 9, 4, 10, 10, 3, 1, 5, 3, 9], device='cuda:0') labels:
tensor([ 0, 2, 9, 1, 4, 2, 5, 6, 3, 3, 1, 6, 2, 1, 1, 7, 0, 7,
          4, 1, 2, 3, 2, 8, 8, 9, 1, 2, 5, 3, 2, 5, 10, 3, 3, 6,
          4, 9, 4, 3, 10, 3, 7, 5, 3, 9], device='cuda:0')
eval_loss: 0.9702419638633728
correct: 42
Accuracy of the network on the test images: 91 %
pred: tensor([10, 2, 7, 9, 2, 4, 1, 10, 1, 3, 2, 5, 4, 1, 3, 3,
6, 0,
          6, 4, 9, 1, 1, 8, 10, 1, 4, 5, 2, 5, 3, 0, 2, 2, 5, 3,
          4, 6, 7, 10, 9, 3, 9, 2, 8, 9], device='cuda:0') labels:
tensor([ 3, 2, 7, 9, 2, 4, 1, 10, 1, 3, 2, 5, 4, 7, 3, 3, 6, 0,
          6, 3, 8, 1, 1, 5, 10, 1, 4, 1, 2, 5, 3, 0, 2, 2, 5, 3,
          4, 6, 7, 3, 9, 3, 9, 2, 8, 9], device='cuda:0')
eval_loss: 0.9041318297386169
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 7, 9, 2, 0, 10, 2, 1, 0, 5, 1, 2, 6, 10, 3, 9, 7,
8, 9,
          10, 3, 3, 5, 1, 3, 4, 8, 5, 6, 6, 4, 6, 2, 3, 1, 9, 4,
          4, 2, 1, 2, 10, 3, 6, 5, 9, 2], device='cuda:0') labels:
tensor([ 7, 8, 2, 0, 3, 2, 1, 0, 5, 1, 2, 6, 3, 3, 9, 7, 5, 9,
          10, 3, 3, 1, 1, 3, 4, 8, 5, 3, 6, 4, 1, 2, 3, 1, 9, 4,

```

```

    4, 2, 7, 2, 10, 3, 6, 5, 9, 2], device='cuda:0')
eval_loss: 0.9390644431114197
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([ 3, 10, 9, 1, 3, 5, 1, 7, 2, 3, 6, 2, 4, 3, 1, 2,
 2, 0,
           9, 5, 9, 9, 4, 10, 8, 4, 0, 6, 4, 7, 1, 1, 2, 3, 5, 9,
           0, 8, 2, 3, 6, 2, 4, 5, 1, 3], device='cuda:0') labels:
tensor([ 3, 10, 8, 1, 3, 1, 7, 7, 2, 3, 6, 2, 4, 3, 1, 2, 2, 3,
 9, 5, 9, 9, 4, 10, 5, 3, 0, 6, 4, 7, 1, 1, 2, 3, 5, 9,
 0, 8, 2, 3, 6, 2, 4, 5, 1, 3], device='cuda:0')
eval_loss: 0.8579078316688538
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 6, 3, 2, 1, 9, 7, 10, 9, 10, 8, 3, 6, 1, 2, 2, 4,
 1, 3,
           5, 10, 9, 9, 7, 4, 6, 0, 5, 3, 1, 2, 3, 4, 1, 0, 5, 8,
          10, 2, 2, 5, 2, 3, 1, 9, 4, 6], device='cuda:0') labels:
tensor([ 6, 3, 2, 1, 8, 7, 3, 9, 3, 8, 3, 6, 7, 2, 2, 4, 1, 3,
 5, 10, 9, 9, 7, 4, 3, 0, 5, 3, 1, 2, 3, 4, 1, 0, 1, 5,
 10, 2, 2, 5, 2, 3, 1, 9, 4, 6], device='cuda:0')
eval_loss: 1.0151355266571045
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 6, 2, 3, 6, 4, 0, 2, 5, 10, 3, 1, 1, 2, 9, 5, 5,
 2, 2,
           1, 3, 10, 8, 7, 3, 8, 4, 6, 10, 9, 2, 1, 6, 5, 3, 9, 3,
           9, 4, 2, 10, 4, 0, 7, 1, 9, 4], device='cuda:0') labels:
tensor([ 6, 2, 3, 1, 4, 0, 2, 5, 3, 3, 7, 1, 2, 9, 1, 5, 2, 2,
 1, 3, 3, 8, 7, 3, 5, 4, 6, 10, 9, 2, 1, 6, 5, 3, 9, 3,
 8, 3, 2, 10, 4, 0, 7, 1, 9, 4], device='cuda:0')
eval_loss: 1.0614252090454102
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([ 4, 2, 4, 9, 1, 0, 6, 10, 3, 7, 9, 6, 1, 6, 3, 2,
 9, 6,
           5, 2, 8, 3, 5, 9, 3, 2, 8, 2, 3, 10, 4, 1, 1, 7, 4, 1,
           5, 10, 2, 9, 1, 3, 10, 2, 0, 5], device='cuda:0') labels:
tensor([ 4, 2, 4, 8, 1, 0, 6, 3, 3, 7, 9, 6, 1, 3, 3, 2, 9, 6,
 1, 2, 5, 3, 5, 9, 3, 2, 8, 2, 3, 3, 4, 1, 1, 7, 4, 7,
 5, 10, 2, 9, 1, 3, 10, 2, 0, 5], device='cuda:0')
eval_loss: 0.9536502957344055
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 2, 1, 2, 3, 3, 4, 1, 4, 3, 2, 2, 9, 3, 6, 1, 9,
 5, 4,
           7, 0, 9, 0, 2, 5, 6, 10, 4, 5, 5, 8, 10, 9, 7, 3, 10, 8,
           2, 1, 6, 2, 1, 9, 8, 10, 1, 3], device='cuda:0') labels:

```

```

tensor([ 2,  1,  2,  3,  3,  4,  1,  4,  3,  2,  2,  9,  3,  6,  1,  9,  1,  4,
         7,  0,  9,  0,  2,  5,  3, 10,  4,  5,  5,  6,  3,  9,  7,  3, 10,  5,
         2,  7,  6,  2,  1,  8,  8,  3,  1,  3], device='cuda:0')
eval_loss: 1.0167291164398193
correct: 38
Accuracy of the network on the test images: 82 %
pred: tensor([10,  9, 10,  2,  6,  4,  0,  9,  4,  9,  7,  2,  4,  7, 10,  5,
              1,  1,
              6,  3,  2,  4,  0,  1,  1,  2,  1,  6,  2,  3,  8,  2,  5,  2,  5,  1,
              3,  4,  5,  9, 10,  5,  3,  3,  9,  3], device='cuda:0') labels:
tensor([ 3,  9, 10,  2,  6,  4,  0,  9,  3,  9,  7,  2,  4,  7,  3,  5,  1,  1,
         6,  3,  2,  4,  0,  1,  7,  2,  1,  6,  2,  3,  8,  2,  5,  2,  5,  1,
         3,  4,  1,  9, 10,  5,  3,  3,  8,  3], device='cuda:0')
eval_loss: 0.7982006669044495
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 9,  2,  1,  2,  9,  6,  5,  4,  0,  5,  1,  3,  2,  7,  1,  4,
              4,  2,
              6,  7,  8,  1,  2,  3, 10,  6, 10,  5,  4,  9,  1,  9,  0,  5,  8,  3,
              10,  4,  9,  3,  3,  0,  2,  1,  2,  3], device='cuda:0') labels:
tensor([ 9,  2,  1,  2,  9,  6,  1,  4,  0,  5,  1,  3,  2,  7,  7,  4,  4,  2,
         6,  7,  8,  1,  2,  3, 10,  6,  3,  5,  4,  9,  1,  8,  3,  5,  5,  3,
         10,  3,  9,  3,  3,  0,  2,  1,  2,  3], device='cuda:0')
eval_loss: 1.003605604171753
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 2,  3, 10,  5,  5,  9,  0,  6,  0,  3,  6,  1,  9,  1,  2,  7,
              6,  4,
              3,  1,  1,  4,  2,  9,  8,  6,  5,  2,  8,  3, 10,  3,  3,  2,  7,  4,
              1,  1,  9,  9,  5,  2,  4, 10,  2, 10], device='cuda:0') labels:
tensor([ 2,  3,  3,  1,  5,  9,  0,  6,  0,  3,  6,  1,  9,  1,  2,  7,  3,  4,
         3,  1,  7,  4,  2,  8,  8,  6,  5,  2,  5,  3, 10,  3,  3,  2,  7,  4,
         1,  1,  9,  9,  5,  2,  4,  3,  2, 10], device='cuda:0')
eval_loss: 0.8461123704910278
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 9,  9,  5,  6,  1,  9,  9,  3,  5, 10,  4,  9,  4,  1,  2,  5,
              7,  2,
              2, 10,  3,  4,  1,  4,  3,  7,  3,  0,  2,  1,  1, 10,  1,  6,  6,  4,
              3,  3,  2,  8,  8, 10,  2,  1,  0,  2], device='cuda:0') labels:
tensor([ 9,  9,  5,  6,  7,  9,  8,  3,  5, 10,  4,  9,  3,  1,  2,  5,  7,  2,
         2,  3,  3,  4,  1,  4,  3,  7,  3,  0,  2,  1,  1,  3,  1,  6,  6,  4,
         3,  3,  2,  8,  5, 10,  2,  1,  0,  2], device='cuda:0')
eval_loss: 0.8817638754844666
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 1,  1,  2,  4,  5,  2,  3,  2,  4,  3, 10,  2,  9,  3,  6,  5,
              4,  0,

```

```

      8, 5, 4, 10, 1, 1, 10, 9, 9, 2, 3, 0, 6, 2, 3, 4, 9, 1,
      2, 10, 7, 8, 1, 1, 6, 9, 3, 7], device='cuda:0') labels:
tensor([ 1, 1, 2, 4, 5, 2, 3, 2, 4, 3, 10, 2, 9, 3, 6, 5, 4, 0,
        8, 5, 3, 3, 1, 1, 10, 9, 9, 2, 3, 0, 6, 2, 3, 4, 9, 1,
        2, 3, 7, 5, 1, 7, 6, 8, 3, 7], device='cuda:0')
eval_loss: 0.7112213373184204
correct: 40
Accuracy of the network on the test images: 86 %
pred: tensor([ 2, 9, 4, 3, 4, 0, 2, 7, 5, 6, 1, 5, 1, 6, 1, 1,
        4, 3,
        2, 3, 3, 6, 2, 9, 8, 4, 2, 3, 4, 1, 2, 9, 0, 5, 7, 9,
        1, 1, 0, 2, 10, 8, 10, 10, 10, 9], device='cuda:0') labels:
tensor([ 2, 8, 4, 3, 4, 0, 2, 7, 5, 6, 7, 5, 1, 6, 1, 1, 4, 3,
        2, 3, 3, 6, 2, 9, 8, 3, 2, 3, 4, 1, 2, 9, 0, 5, 7, 9,
        1, 1, 3, 2, 10, 5, 3, 3, 10, 9], device='cuda:0')
eval_loss: 0.9118428230285645
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 6, 2, 4, 1, 0, 3, 1, 5, 9, 0, 3, 0, 10, 9, 8, 5,
        2, 2,
        3, 10, 5, 1, 6, 2, 2, 5, 7, 9, 1, 1, 5, 7, 3, 2, 10, 9,
        1, 3, 4, 6, 2, 4, 6, 9, 10, 4], device='cuda:0') labels:
tensor([ 6, 2, 4, 1, 3, 3, 7, 5, 9, 0, 3, 0, 3, 9, 8, 5, 2, 2,
        3, 3, 5, 1, 6, 2, 2, 1, 7, 8, 1, 1, 5, 7, 3, 2, 10, 9,
        1, 3, 4, 3, 2, 4, 6, 9, 10, 4], device='cuda:0')
eval_loss: 0.8130225539207458
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 9, 4, 1, 6, 3, 5, 8, 10, 2, 2, 1, 2, 9, 8, 3, 1,
        6, 10,
        1, 3, 9, 10, 6, 2, 0, 0, 4, 4, 0, 2, 3, 5, 7, 5, 2, 6,
        7, 3, 9, 3, 9, 1, 4, 1, 5, 2], device='cuda:0') labels:
tensor([ 9, 4, 1, 6, 3, 5, 5, 3, 2, 2, 1, 2, 9, 8, 3, 7, 6, 10,
        1, 3, 8, 10, 6, 2, 3, 0, 4, 4, 0, 2, 3, 1, 7, 5, 2, 3,
        7, 3, 9, 3, 9, 1, 4, 1, 5, 2], device='cuda:0')
eval_loss: 0.8862463235855103
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 1, 9, 2, 10, 3, 9, 6, 5, 1, 1, 8, 4, 5, 4, 3, 2,
        10, 6,
        1, 3, 8, 3, 9, 2, 2, 3, 2, 9, 1, 1, 4, 5, 3, 9, 4, 2,
        0, 7, 2, 1, 7, 0, 10, 3, 4, 3], device='cuda:0') labels:
tensor([ 1, 9, 2, 3, 6, 9, 6, 5, 1, 1, 5, 4, 5, 4, 3, 2, 10, 6,
        7, 3, 8, 3, 9, 2, 2, 3, 2, 9, 1, 1, 3, 5, 3, 8, 4, 2,
        0, 7, 2, 1, 7, 0, 10, 3, 4, 3], device='cuda:0')
eval_loss: 0.6350084543228149
correct: 40
Accuracy of the network on the test images: 86 %

```

```

pred: tensor([ 1,  2,  9,  6,  8,  1,  2,  4,  2, 10,  8,  1,  0,  0,  9,  5,
  7,  6,
           1,  7,  1,  4,  3,  5,  3,  2,  3,  6,  1,  3,  4,  2,  5,  3,  4,  9,
  6, 10,  3,  5, 10,  2,  9,  9, 10,  2], device='cuda:0') labels:
tensor([ 1,  2,  9,  6,  8,  1,  2,  4,  2,  3,  5,  1,  0,  0,  8,  1,  7,  3,
  1,  7,  7,  4,  3,  5,  3,  2,  3,  6,  1,  3,  4,  2,  5,  3,  4,  9,
  6, 10,  3,  5,  3,  2,  9,  9, 10,  2], device='cuda:0')
eval_loss: 0.8534974455833435
correct: 39
Accuracy of the network on the test images: 84 %
pred: tensor([ 2,  1,  4,  9,  5,  3,  9, 10,  3,  1,  2,  3,  2, 10,  1,  7,
  6,  2,
           2,  7,  2, 10,  5,  9,  6,  9,  1,  4,  1,  1,  5, 10,  3,  4,  6,  3,
  0,  4,  6,  5,  0,  3,  2,  5,  9,  8], device='cuda:0') labels:
tensor([ 2,  1,  4,  9,  1,  3,  9, 10,  3,  1,  2,  3,  2, 10,  1,  7,  6,  2,
  2,  7,  2,  3,  5,  9,  6,  9,  1,  4,  7,  1,  5,  3,  3,  4,  6,  3,
  0,  4,  3,  5,  0,  3,  2,  5,  8,  8], device='cuda:0')
eval_loss: 1.0181236267089844
correct: 40
Accuracy of the network on the test images: 86 %
total_acc: 85.41304347826078

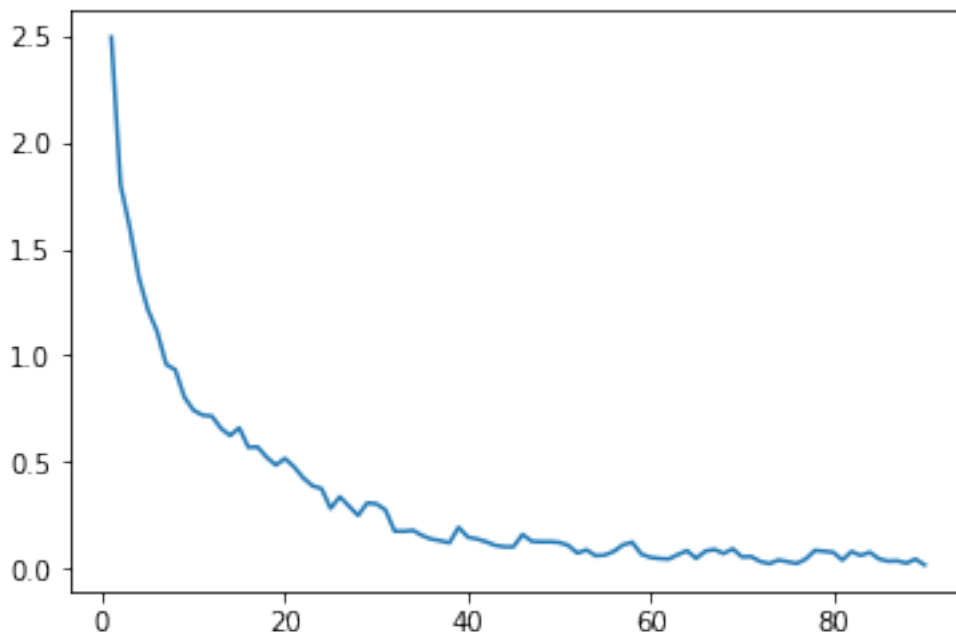
```

Visualization of training error.

```

[10]: # In[]
from matplotlib import pyplot as plt
x = np.linspace(1,len(LOSS),len(LOSS))
plt.plot(x,LOSS)
plt.show()

```



It can be seen that the exact interest rate of our model is stable at 85.41%, which is equivalent to only 6 pictures in the 46 pictures of the test set.

Advantages of the model: less convolution layer and less training.