Exercise:

Create a breed identifier, choose an animal, collect images, adjust the network, so that the identifier can distinguish between different breeds (at least 8 breeds).

Submission requirements:

1. source code transferlearning.py
2. PDF documents
    1. Explaining the steps.
        1. Image collecting.
        2. Image processing.
        3. Build/test the model.
    2. Show the outputs.
3. Upload to e-learning before 4/26 14:10

Image collection

Ten kinds of monkey: 山魈、金絲猴、長鼻猴、長臂猿、捲尾猴、眼鏡猴、絨猴、獼猴

Collect each kind of images on Google.

Image processing

```
train_transforms=transforms.Compose([

    # Randomly crop the image to size 224x224
    transforms.RandomResizedCrop(224),

    # Randomly flip the image horizontally
    transforms.RandomHorizontalFlip(),

    # Convert the image to a PyTorch tensor
    transforms.ToTensor(),

    # Normalize the image with mean [0.485, 0.456, 0.406] and standard deviation [0.229, 0.224, 0.225]
    transforms.Normalize(
        [0.485,0.456,0.406],
        [0.229,0.224,0.225]
    )
])
```

Step1: Randomly crop the original image into size of 224x224.
Step2: Randomly flip the image horizontal.
Step3: Convert the image into Tensor.
Step4: Normalize the image.

Build/ test the model

```
model=models.resnet18(weights='ResNet18_Weights.DEFAULT')
```

Use resnet18's default bias as initial bias of our model.

Result

```
epoch: 0, Train loss: 0.5434, acc:0.2125,     Val loss: 0.3287, acc:0.6500
epoch: 1, Train loss: 0.3747, acc:0.4625,     Val loss: 0.1651, acc:0.8000
epoch: 2, Train loss: 0.3151, acc:0.5500,     Val loss: 0.0459, acc:0.9625
epoch: 3, Train loss: 0.2053, acc:0.7750,     Val loss: 0.0218, acc:1.0000
epoch: 4, Train loss: 0.1915, acc:0.7375,     Val loss: 0.0109, acc:0.9875
```

The accuracy and loss during training.

Retrieve part of the data to display the result.