

有两个文件夹（train 和 test）包含图片，各包含五种类型的水稻图片，train 文件夹里每种 2 千张，test 文件夹里每种 858 张

D:\023\machine\123\train				
名称	修改日期	类型	大小	
Arborio	2023/12/14 20:26	文件夹		
Basmati	2023/12/14 20:26	文件夹		
Ipsala	2023/12/14 20:26	文件夹		
Jasmine	2023/12/14 20:26	文件夹		
Karacadag	2023/12/14 20:26	文件夹		

分别生成 train_data.csv 与 test_data.csv 文件，第一列是图片路径，第二列是标签（用的 0-4，分别代表五种），程序会分别从 csv 文件里遍历路径，去使用对应路径的图片进行训练与测试，

	A	B	C	D
1	0	1		
2	D:/023/machine/123/train/Arborio\Arborio (8199). jpg	0		
3	D:/023/machine/123/train/Arborio\Arborio (82). jpg	0		
4	D:/023/machine/123/train/Arborio\Arborio (820). jpg	0		
5	D:/023/machine/123/train/Arborio\Arborio (8200). jpg	0		
6	D:/023/machine/123/train/Arborio\Arborio (8201). jpg	0		
7	D:/023/machine/123/train/Arborio\Arborio (8202). jpg	0		
8	D:/023/machine/123/train/Arborio\Arborio (8203). jpg	0		
9	D:/023/machine/123/train/Arborio\Arborio (8204). jpg	0		
10	D:/023/machine/123/train/Arborio\Arborio (8205). jpg	0		
11	D:/023/machine/123/train/Arborio\Arborio (8206). jpg	0		
12	D:/023/machine/123/train/Arborio\Arborio (8207). jpg	0		
13	D:/023/machine/123/train/Arborio\Arborio (8208). jpg	0		
14	D:/023/machine/123/train/Arborio\Arborio (8209). jpg	0		
15	D:/023/machine/123/train/Arborio\Arborio (821). jpg	0		
16	D:/023/machine/123/train/Arborio\Arborio (8210). jpg	0		
17	D:/023/machine/123/train/Arborio\Arborio (8211). jpg	0		
18	D:/023/machine/123/train/Arborio\Arborio (8212). jpg	0		
19	D:/023/machine/123/train/Arborio\Arborio (8213). jpg	0		
20	D:/023/machine/123/train/Arborio\Arborio (8214). jpg	0		
21	D:/023/machine/123/train/Arborio\Arborio (8215). jpg	0		
22	D:/023/machine/123/train/Arborio\Arborio (8216). jpg	0		
23	D:/023/machine/123/train/Arborio\Arborio (8217). jpg	0		
24	D:/023/machine/123/train/Arborio\Arborio (8218). jpg	0		
25	D:/023/machine/123/train/Arborio\Arborio (8219). jpg	0		
26	D:/023/machine/123/train/Arborio\Arborio (822). jpg	0		
27	D:/023/machine/123/train/Arborio\Arborio (8220). jpg	0		

现在卡在了 train.py 输出训练模型这里，我请教的那位师兄最表示模型没问题，求训练损失，训练准确率，测试损失，测试准确率这部分有问题，比如 train_accuracy 以及 test_accuracy 这两个不可能小于 20%，

```
with torch.no_grad():
    model.eval()
    for x, y in test_iter:
        x = x.cuda()
        y = y.cpu()
        out = model(x).cpu()
        l = loss(out, y)

        test_acc += torch.sum(out.argmax(dim=1) == y.argmax(dim=1)) # 121212
        test_loss += l

train_loss /= len(train_iter.dataset) # 使用整个数据集的大小
train_accuracy = 100 * train_acc / len(train_iter.dataset)

# 计算测试损失和准确率
test_loss /= len(test_iter.dataset)
test_accuracy = 100 * test_acc / len(test_iter.dataset)

# 打印训练和测试的损失与准确率
print(
    f"Epoch {epoch}: "
    f"train_loss: {train_loss:.4f}, train_accuracy: {train_accuracy:.2f}%, "
    f"test_loss: {test_loss:.4f}, test_accuracy: {test_accuracy:.2f}%"
```