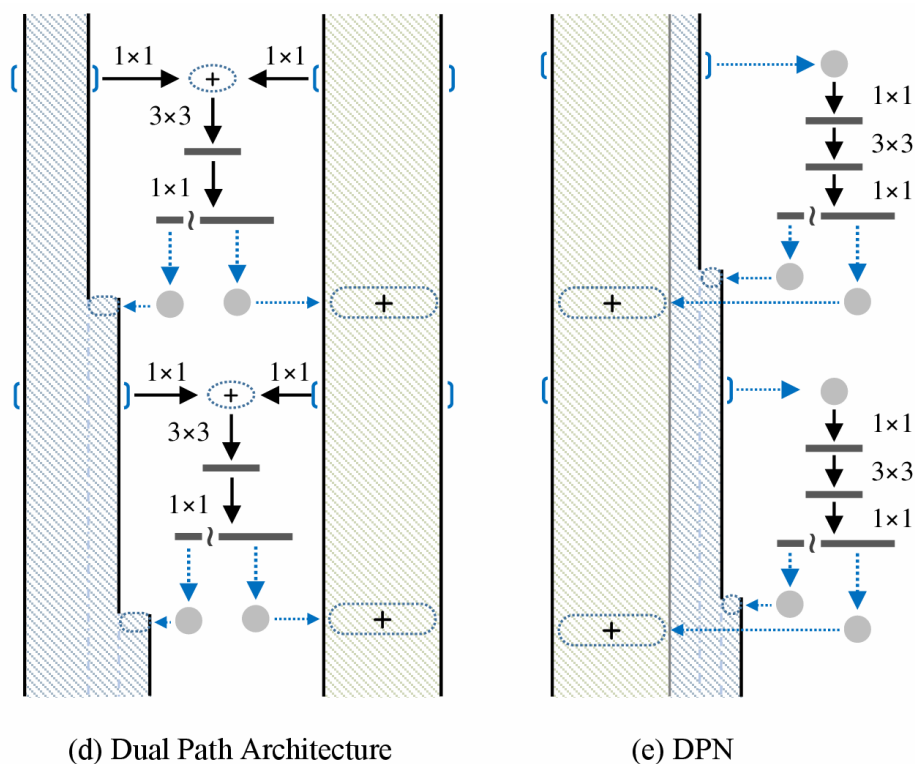


- 战队：业余玩家
- 队员：Derrors（队长）、心态巨烂、zzzlx
- 最终分数：0.99930
- 运行环境：Pytorch 1.1 + 1080Ti GPU
- 使用的模型及其基本结构：Dual Path Network



- 参考文献：<https://arxiv.org/abs/1707.01629>
- 解题思路：
  1. 数据预处理：首先将训练集图片目录及其对应的标签进行封装，同时对图片数据进行预处理（缩放为  $64 \times 64$ ，之后进行中心裁剪为  $32 \times 32$ ，最后归一化处理）；
  2. 加载模型：采用 131 层的深度神经网络模型对训练集进行训练；
  3. 模型训练：，相关参数设置见源代码文件，在训练过程中，训练集准确率达到 0.999 时便对测试集进行一次预测，并保存此时的模型参数；

4. 根据结果对模型进行分析调优，我们调优的思路主要在于以下几个方面：

(1) 模型的创新性，我们尝试了 ResNet、ResNext、DenseNet、SE\_ResNet 等

深度模型，最终选择了比较新的 DPN 模型；

(2) 选择了合适的模型后，在模型深度方面进行优化，根据经验，加深模型可学习

到更深层次的特征，在多次试验中，在算力允许的情况下选择了 131 层的 DPN

网络；

(3) 数据增强，在对模型调参优化提升不明显时，我们选择在数据预处理上进行优

化，对输入模型的数据进行增强，选择将不同大小的图片进行缩放、中心裁剪

为相同大小的尺寸进行处理，并对图片对比度、亮度等进行调整试验，从而选

择效果最好的处理方式；

• 文件目录及代码说明：

|                                 |                         |
|---------------------------------|-------------------------|
| <code>./train/</code>           | # 训练集图片目录               |
| <code>./test/</code>            | # 测试集图片目录               |
| <code>./output/</code>          | # 结果输出目录                |
| <code>    dpn1024_50.csv</code> | # 最终预测结果的 csv 文件        |
| <code>    dpn1024_50.pth</code> | # 最终预测结果的模型参数           |
| <code>./data.csv</code>         | # 训练集文件名及其对应的标签         |
| <code>./train.txt</code>        | # 生成的训练集图片路径及对应标签       |
| <code>./test.txt</code>         | # 生成的测试集图片路径（标签全部初始为 0） |
| <code>./read_data.py</code>     | # 生成训练集/测试集图片路径和标签的代码   |
| <code>./dpn.py</code>           | # DPN 网络结构定义            |
| <code>./train.py</code>         | # 模型训练源代码               |
| <code>./valid.py</code>         | # 加载保存的参数复现预测结果源代码      |

# 对模型进行训练：python train.py

# 复现预测结果：python valid.py