任务二

关于训练方法：

到目前为止我们组的训练只使用了最基本的SGD优化器，默认网络和MSE。 在此任务里我们会按照项目要求使用其他的优化器，网络。

从任务二开始，我们只使用进行了图片增广之后的image，加入了经过随机小角度旋转的图片。我们在之前的训练时除了图片随机旋转之外还加入了图片随机翻转，但这样会大大增加训练时间，所以我们在接下来的训练中只加入了图片随机旋转。

1. 如果换用Adam呢？请找出Adam的lr

相对于SGD，ADAM在一开始便可以使用较大的学习率，以达到让loss 实现快速收敛的目的，下面的训练中我们会用ADAM来实验，学习率我们用的是0.01，beta分别是0.9 和0.999

2. 如果一开始用Adam，之后换成sgd呢?

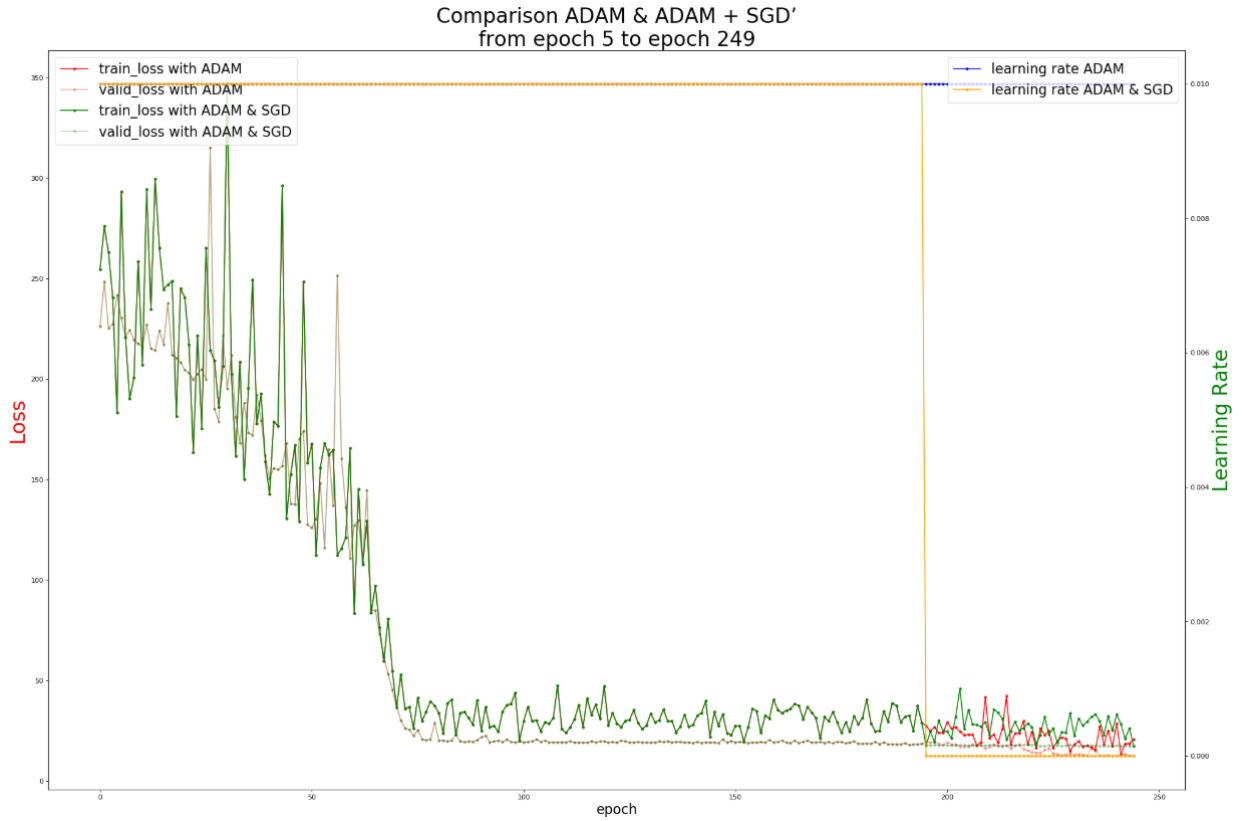
我们在下列两种情况下分别对模型进行了250轮的训练，并画出了ADAM和ADAM+SGD的训练折线。

→ 250轮都用ADAM

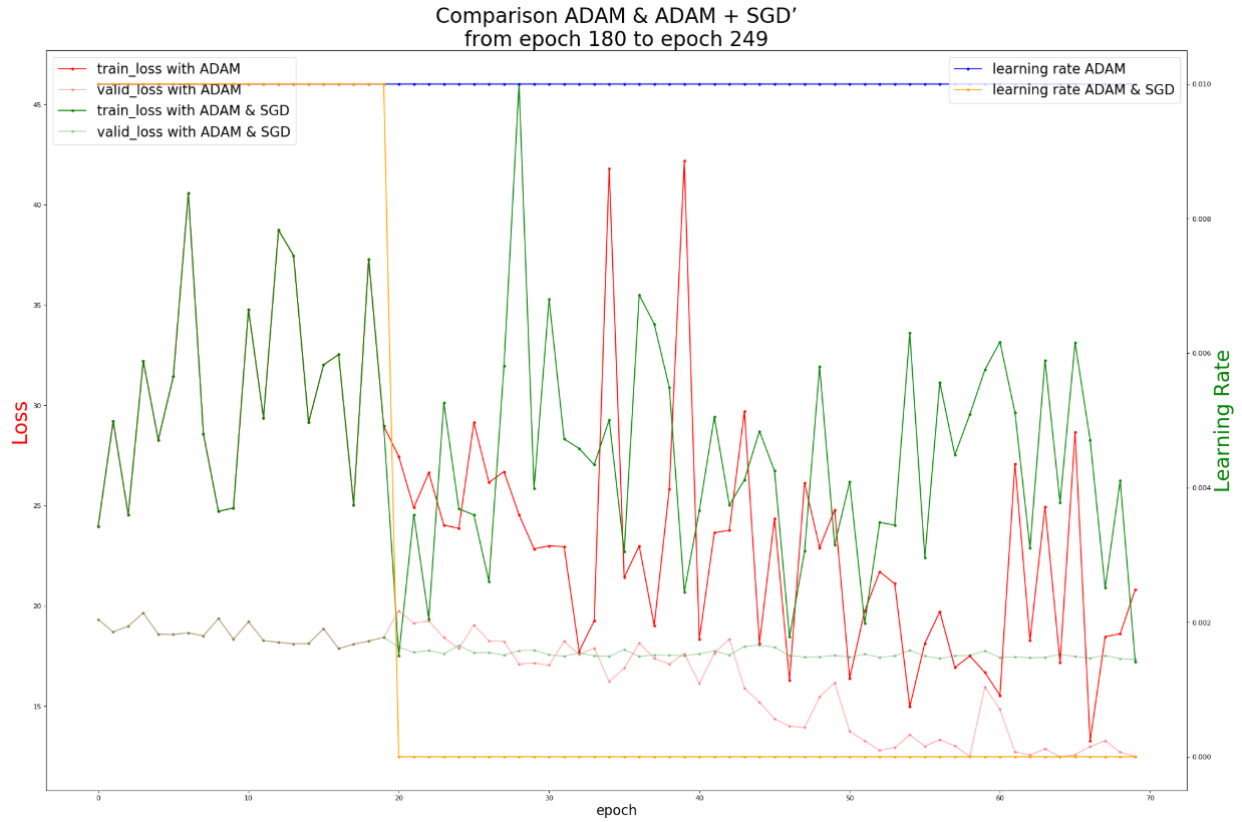
→ 前200轮用ADAM，后50轮用SGD

我们训练的模型参数分别如下:

```
Batch:64
beta:0.9, 0.999
Epoch1: 0-249 (ADAM lr =0.01)
Epoch2: 0-199 (ADAM lr =0.01) + 200-249 (SGD lr = 0.00001)
criterion: MSE
```



→ 可以看到，相对于SGD，使用ADAM作为优化器可以使模型的loss 在70轮训练左右就能达到一个比较小的值



→ 从第201轮训练开始，分别用SGD和ADAM做了50轮的训练

→ 对比两种情况下的valid_loss，虽然在使用SGD的情况下，valid loss比较大，但是它的表现相对于ADAM要平稳的多，应该是之前的训练次数还不足够的关系(loss还未完全收敛)

→ 所以我们推测，当训练的次数加大，比如进行400轮训练，前350轮用ADAM，后50轮用SGD，相比400轮训练都使用ADAM的情况，应该会得到更好更稳定的valid loss

3. 如果用step改变lr呢?

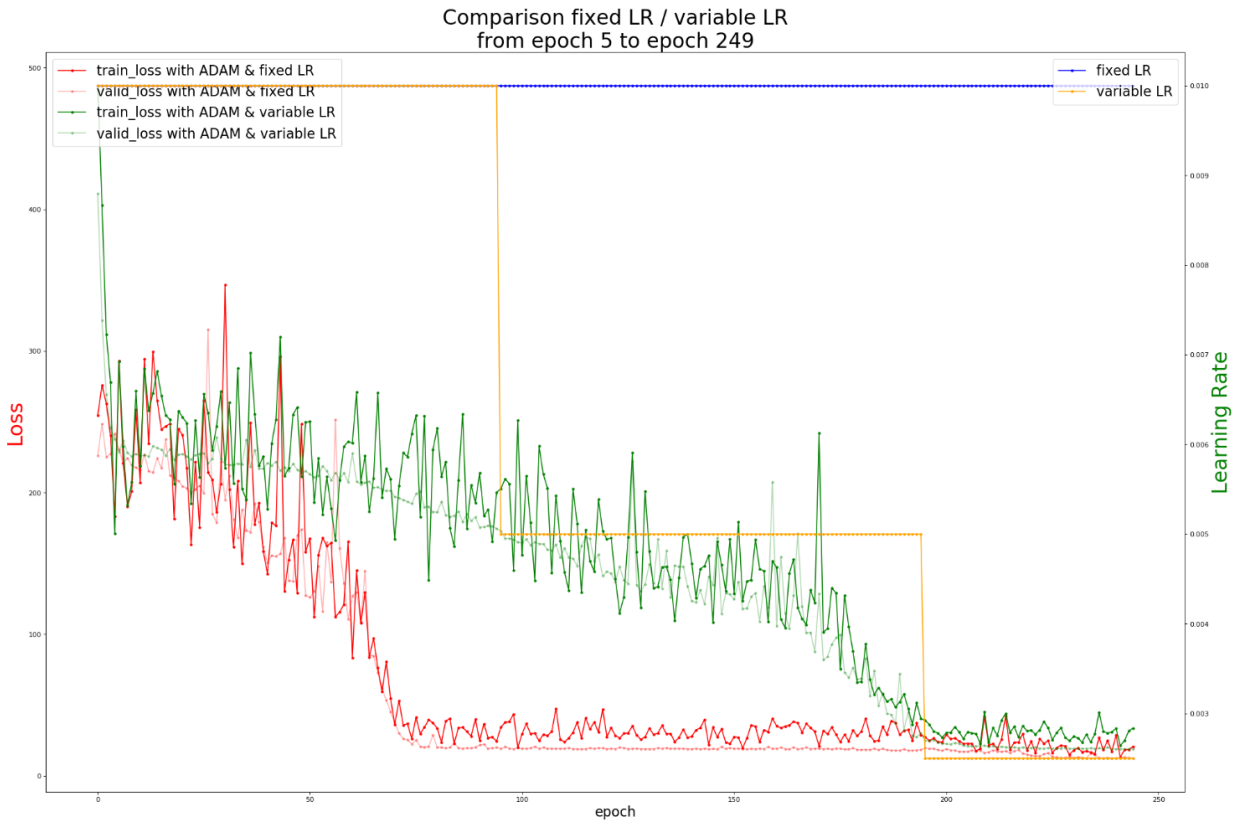
我们修改了我们的模型，除了可变学习率之外，其他模型参数和前面以ADAM为优化器的模型的参数相同

→ 初始学习率 0.01

→ 每训练100轮，学习率下降到之前的0.5倍

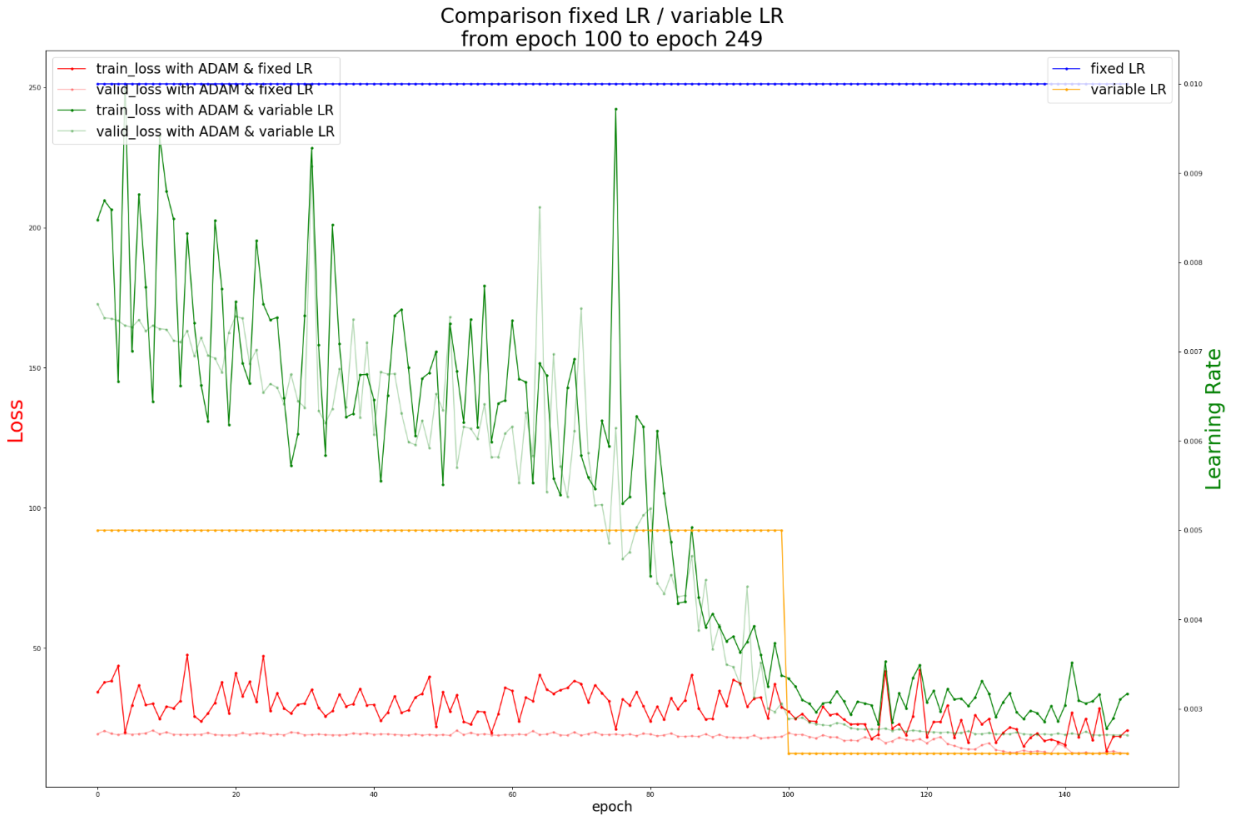
```
# Scheduler Step
scheduler= optim.lr_scheduler.StepLR(optimizer, step_size=100, gamma=0.5, last_epoch=-1)
```

下图为我们分别对固定学习率的模型和可变学习率的模型做的训练



→ 可变学习率由于学习率每隔100轮训练降低的关系，loss下降比较慢

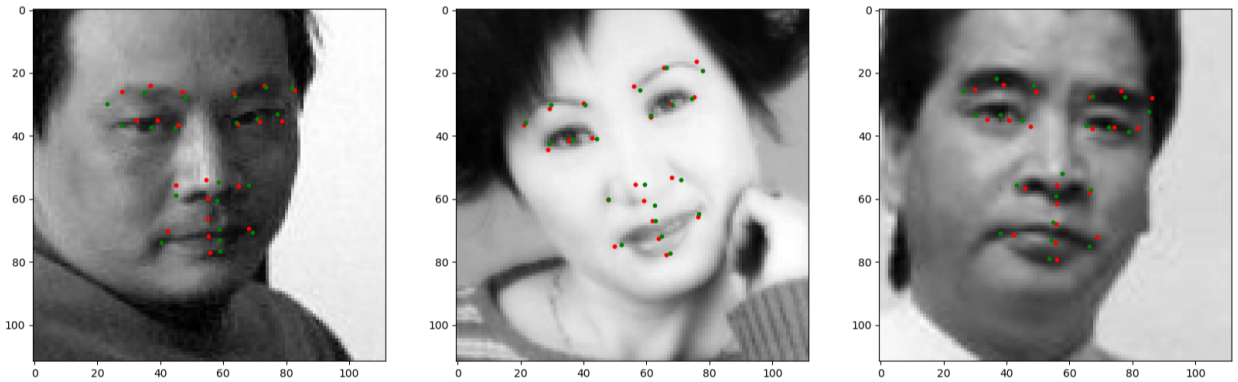
→ 最终基本能达到和固定学习率下的模型一样的loss



→ 但是可边学习率下的loss 震荡情况比较严重

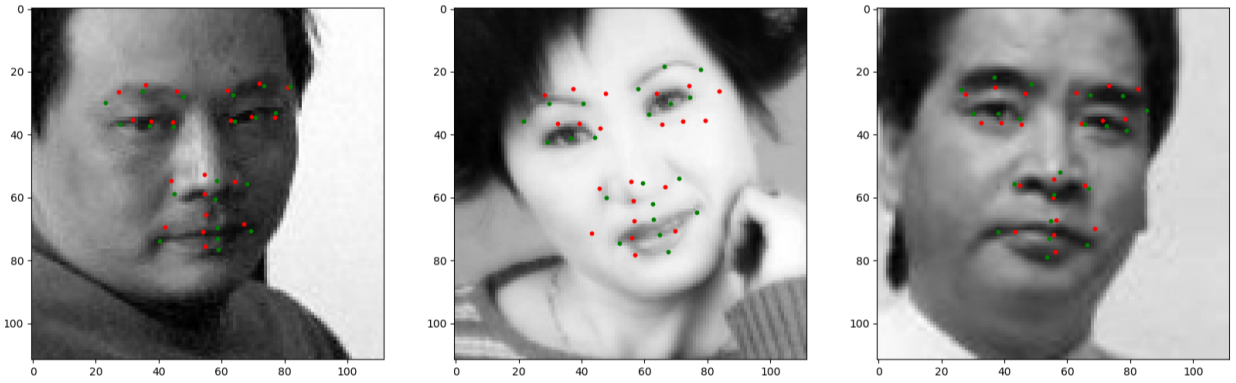
以下为Test结果(红点为测试坐标，绿色点为ground truth)

ADAM(fixed LR)



→ 有误差但是误差并不大

ADAM(variable LR)



→ 对于识别旋转了一定角度的人脸，误差较大(中间的图)

→ 推测误差比较大的原因应该是过早的降低了学习率，导致模型训练不足

4. 如果加上batch normalization呢?

为了对比训练和测试效果我们分别用了以下3种方法来进行训练:

→ ADAM

→ ADAM + BN

→ ADAM + BN + Dropout

ADAM

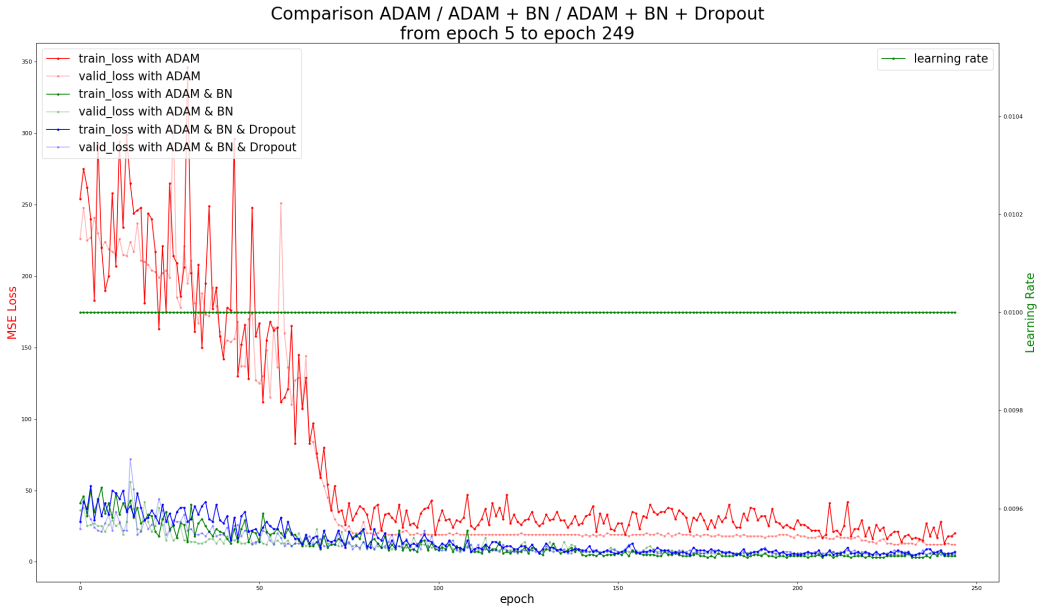
```
Learning rate: 0.01 beta:0.9, 0.999
Batch_Size:64
Epoch: 250
criterion: MSE
```

ADAM + BN

```
Learning rate: 0.01 beta:0.9, 0.999
Batch_Size:64
Epoch: 250
criterion: MSE
BN: 分别在conv1_1,conv2_2 和conv3_3 后面加了BN层
```

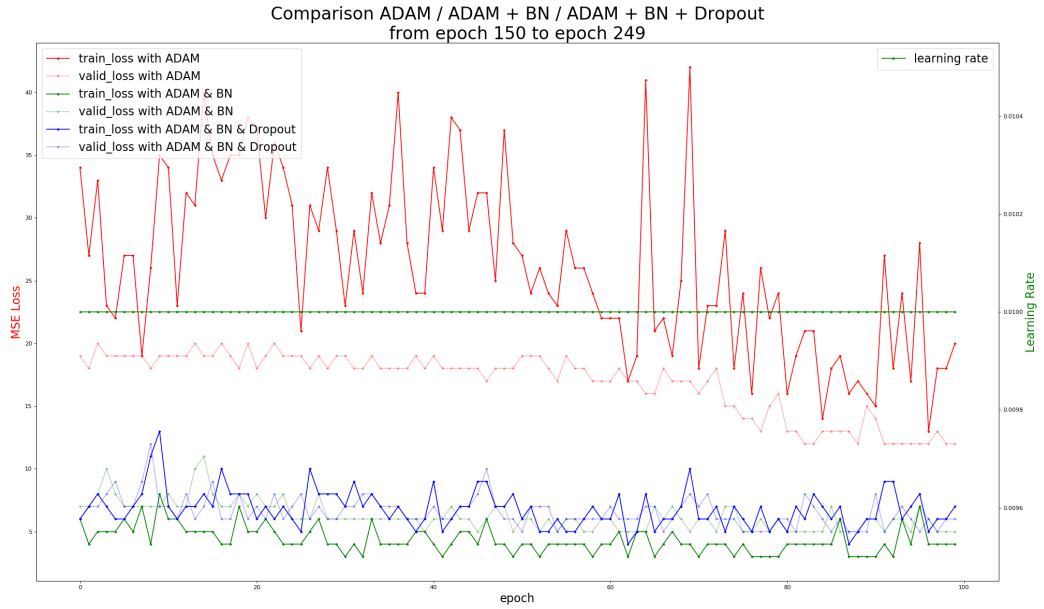
ADAM + BN + Dropout

```
Learning rate: 0.01 beta:0.9, 0.999
Batch_Size:64
Epoch: 250
criterion: MSE
BN: 分别在conv1_1,conv2_2 和conv3_3 后面加了BN层
Dropout:在flatten后面加了dropout层，随机激活70%的神经元
```



→ 加了BN层的网络可以更快的收敛(蓝色以及绿色折线)

→ 加了BN层网络的Loss能在相同的训练epoch下达到更低(对比红色和绿色折线)



对于这两种有BN的网络:

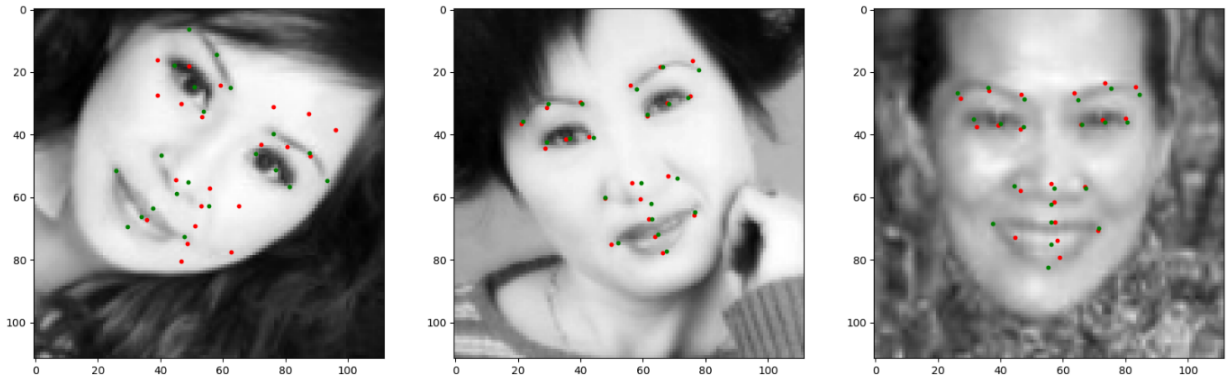
→ 验证集的Loss 都能下降到6左右

→ 只有BN的网络有一些过拟合现象(深绿色的训练Loss低于浅绿色的验证Loss)

→ 而加了Dropout的网络很好的抑制了过拟合现象(训练LOSS ≈ 验证LOSS)

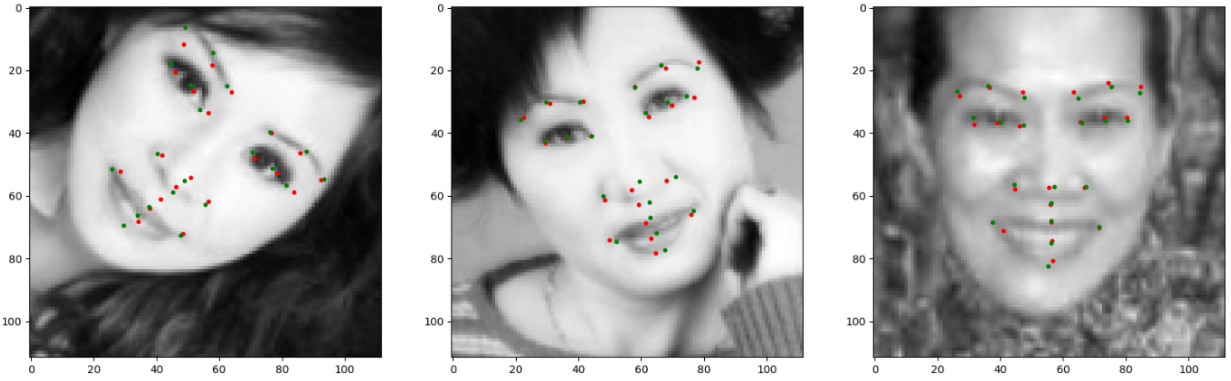
以下为Test结果(红点为测试坐标，绿色点为ground truth)

ADAM



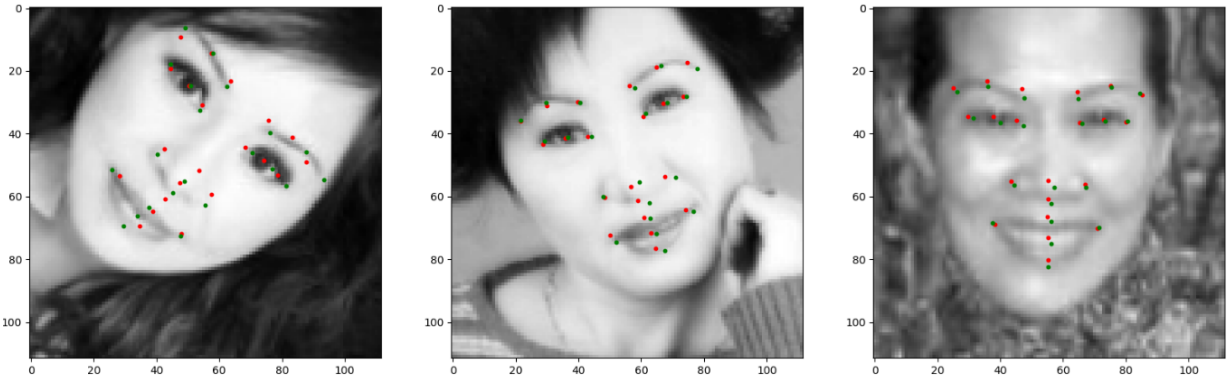
→ 在左图的人脸大角度情况下，训练明显不足，不能很好的识别旋转的图片中的关键点

ADAM+BN



→ 测试结果比较准确，个别点有误差

ADAM + BN + Dropout



→ 测试结果比较准确，个别点有误差

结论: ADAM作为优化器能让loss快速收敛，在网络中加入BN层对于训练是非常必要的，Dropout层能改善过拟合。