基于deepLab-V3+的钢铁分割实验记录

负责人：高圣君

实验日期：2018年10月21日

实验环境：

GPU： TITAN Xp (12G) x4

python3.6.1

tensorflow-gpu 1.3.0

tensorflow 项目链接: <https://github.com/tensorflow/models.git>

reference: <https://blog.csdn.net/qq_32799915/article/details/80070711>

数据集：IronData\_CamVid

路径：新服务器下 /home/liushixin/AI/chendali/shengjunGao/Dataset/IronData\_Camvid

操作I：（源码默认参数）

learning\_policy: poly

base\_learning\_rate:0.0001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:80000

momentum:0.9

train\_batch\_size:2

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

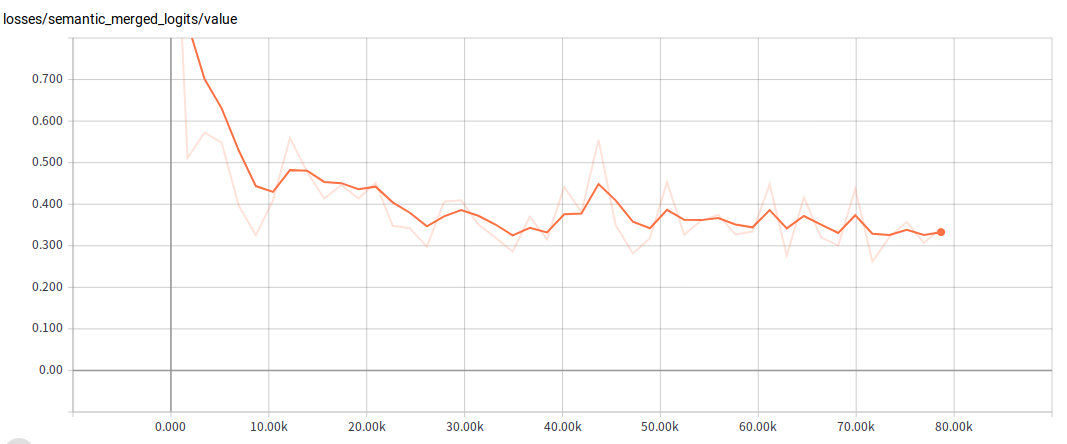
output\_stride:16

decoder\_output\_stride:4

model\_variant:"xception\_65"

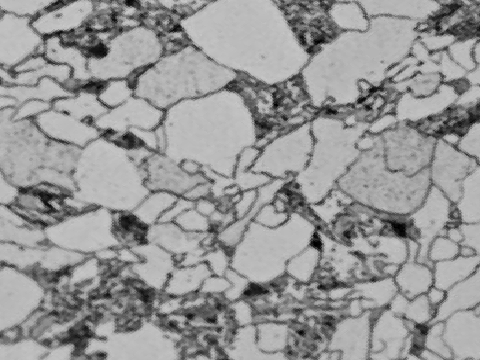
训练结果：

训练集loss曲线

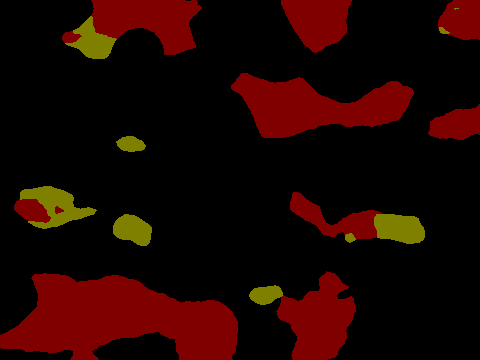


80000次迭代测试结果：

原图 Label

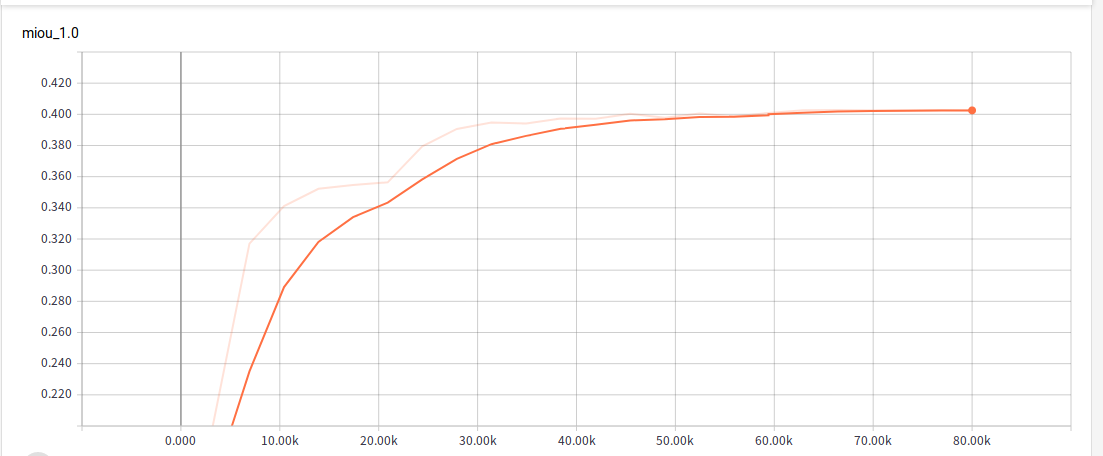


predict



Note: 黄色-珠光体 红色-贝氏体 黑色-铁素体

测试集mIou曲线



经过实验比较发现，在迭代至50000步以后在测试集上的mIOU基本不变。

操作II：（batch 加倍）

learning\_policy: poly

base\_learning\_rate:0.0001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:50000

momentum:0.9

train\_batch\_size:4

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

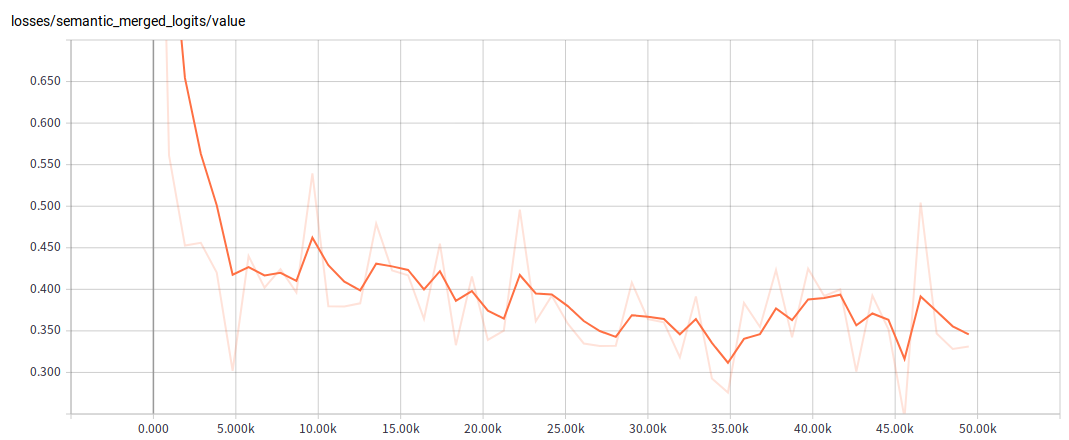
output\_stride:16

decoder\_output\_stride:4

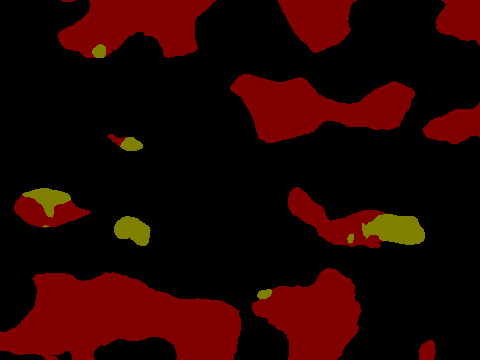
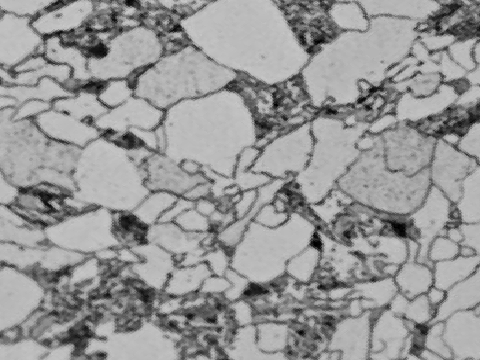
model\_variant:"xception\_65"

训练结果：

训练集loss曲线

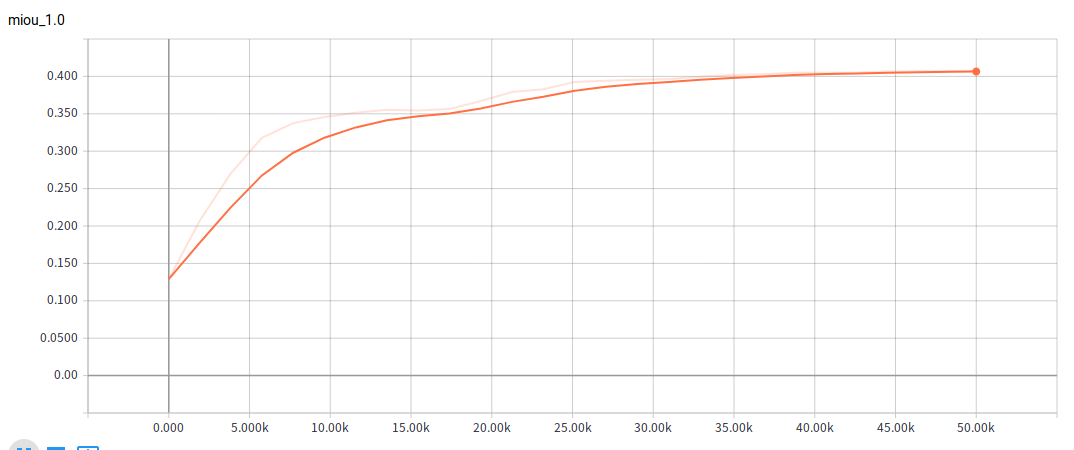


50000次迭代测试结果：

 原图 Label

predict

测试集mIou曲线



通过观察，调整batch大小对模型分割效果的提升作用不大（mIou为0.4），而且增大了计算时间。

操作III：（base\_learning\_rate 加倍）

learning\_policy: poly

base\_learning\_rate:0.001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:50000

momentum:0.9

train\_batch\_size:2

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

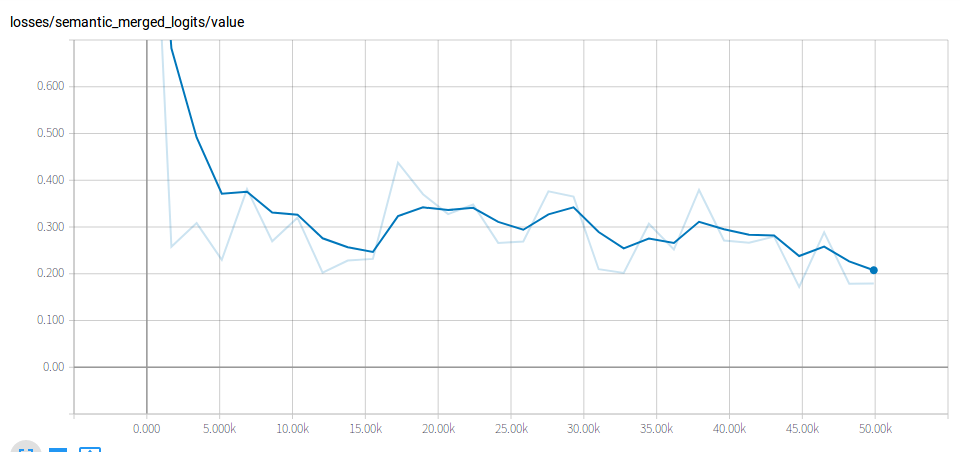
output\_stride:16

decoder\_output\_stride:4

model\_variant:"xception\_65"

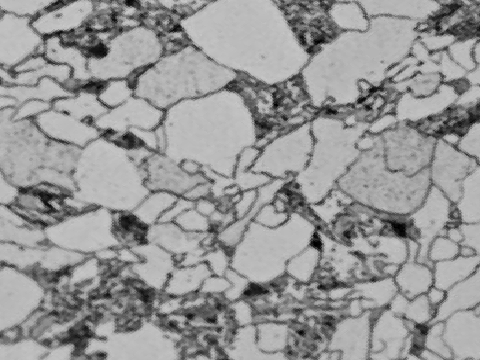
训练结果：

训练集loss曲线

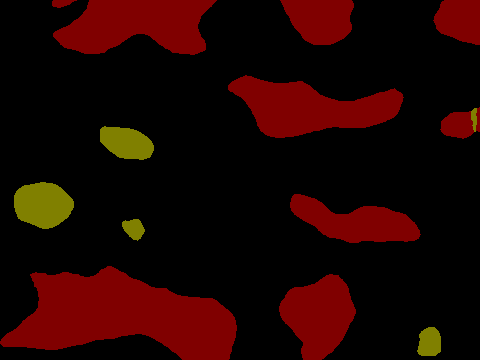


50000次迭代测试结果：

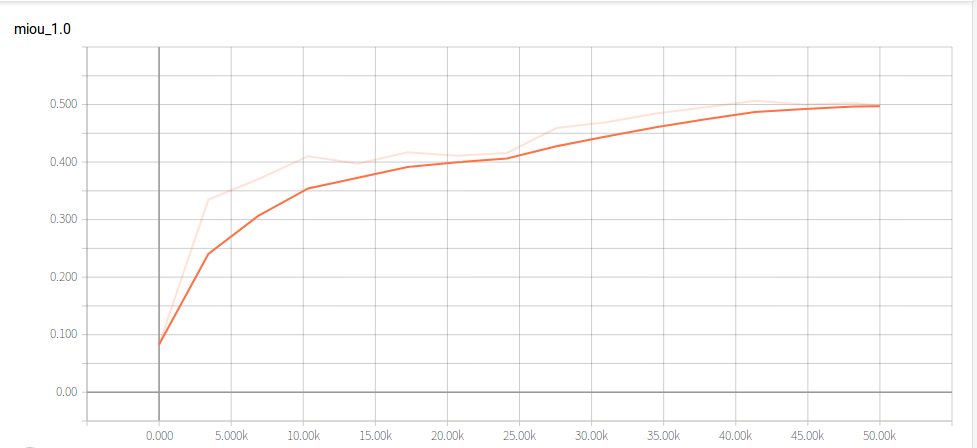
原图 Label

Predict



测试集mIou曲线



经过实验发现将base\_learning\_rate调大后，训练速度提升，同时，mIou提升幅度较大，提升20%左右。

操作IV：（更换学习策略：step）

learning\_policy: step

base\_learning\_rate:0.001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:50000

momentum:0.9

train\_batch\_size:2

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

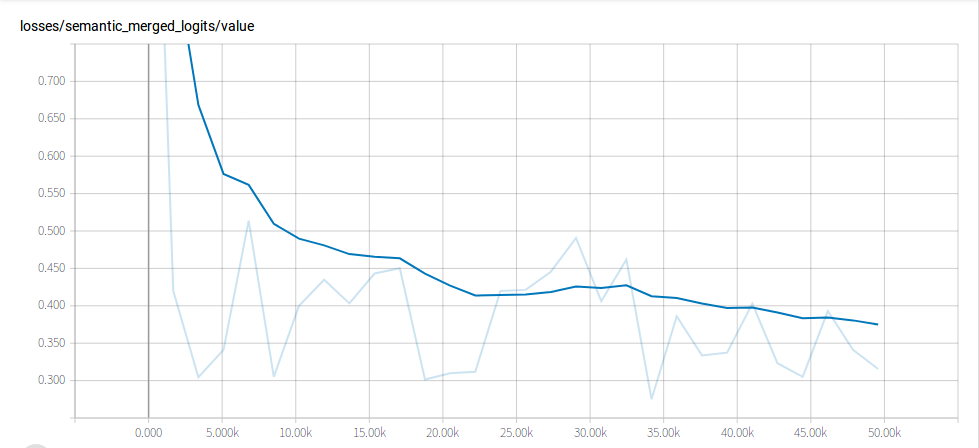
output\_stride:16

decoder\_output\_stride:4

model\_variant:"xception\_65"

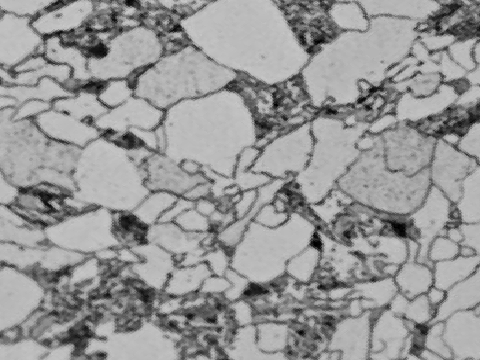
训练结果：

训练集loss曲线

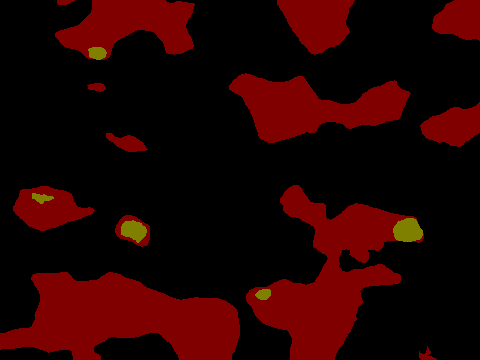


50000次迭代测试结果

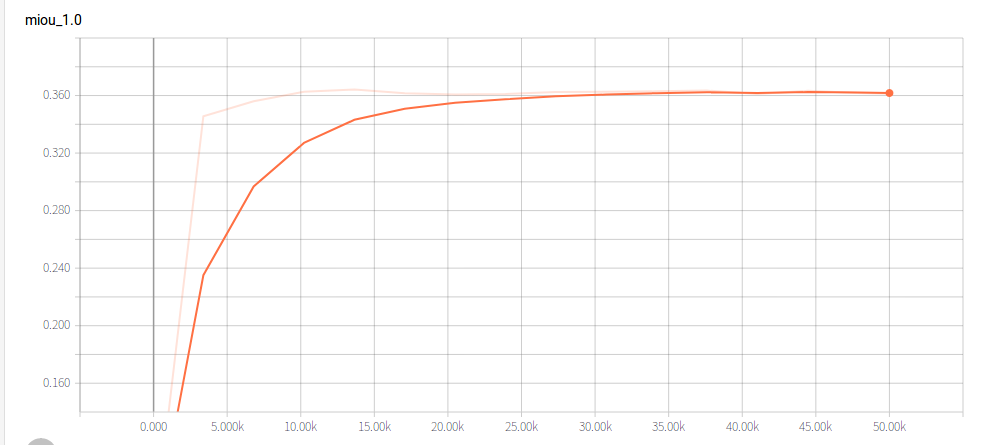
原图 Label

Predict



测试集mIou曲线



经过实验发现将学习策略更为step后，mIou损失很严重，为0.36左右。

操作V：（加深网络层数，将Xception有65层换为71层）

learning\_policy: poly

base\_learning\_rate:0.001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:50000

momentum:0.9

train\_batch\_size:2

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

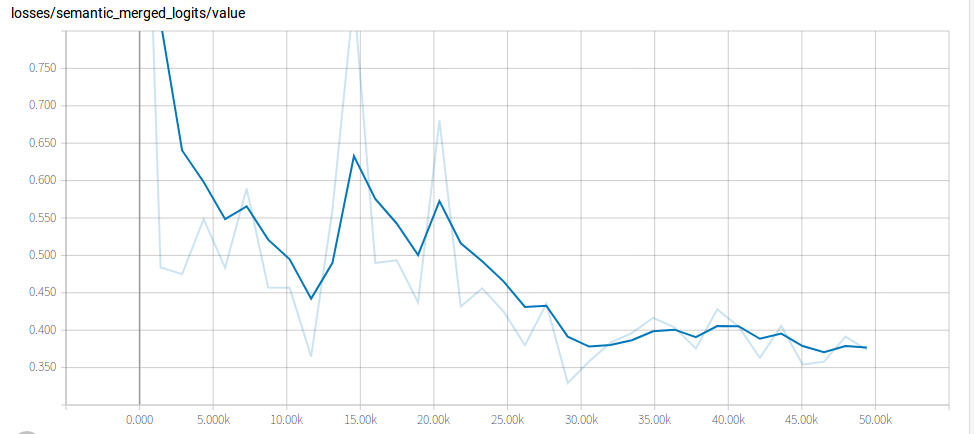
output\_stride:16

decoder\_output\_stride:4

model\_variant:"xception\_71"

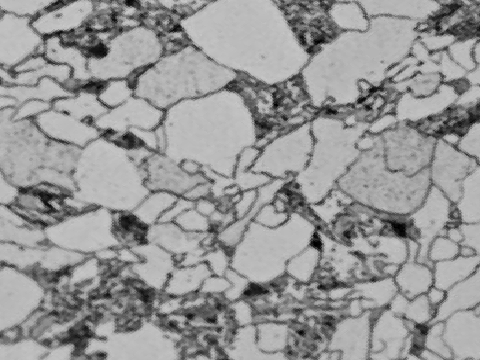
训练结果：

训练集loss曲线

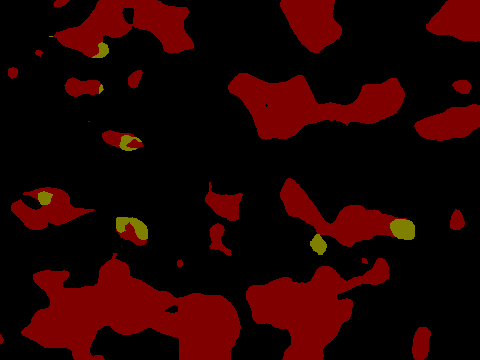


50000次迭代测试结果

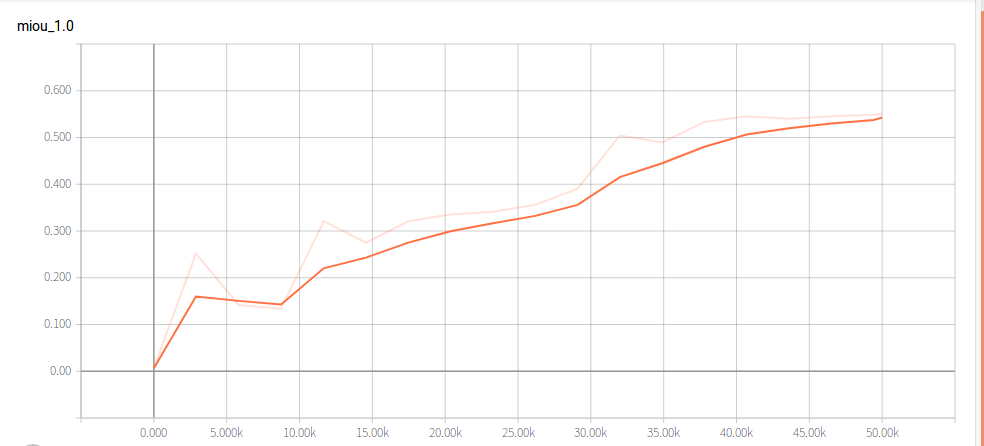
原图 Label

Predict



测试集mIou曲线



实验结果显示，将Xception加深后，mIOU有较大提升，达到0.54左右，相比提升8%左右。

操作VI：（减少网络层数，将Xception有65层换为41层）

learning\_policy: poly

base\_learning\_rate:0.001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:50000

momentum:0.9

train\_batch\_size:2

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

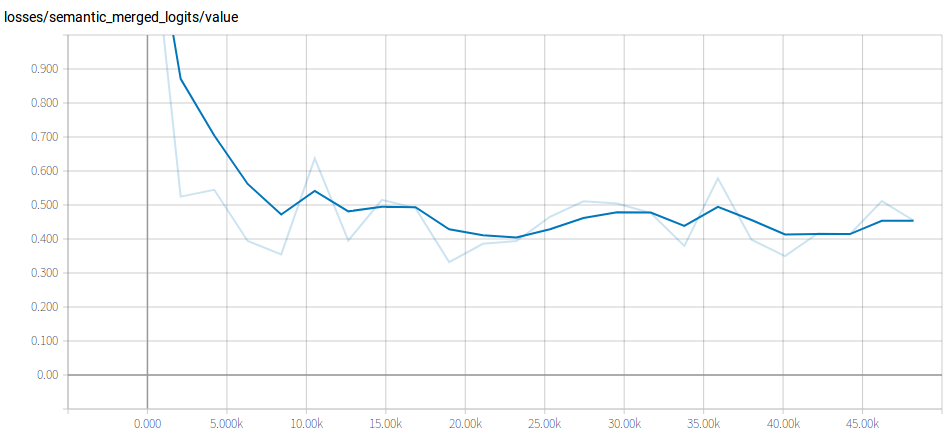
output\_stride:16

decoder\_output\_stride:4

model\_variant:"xception\_41"

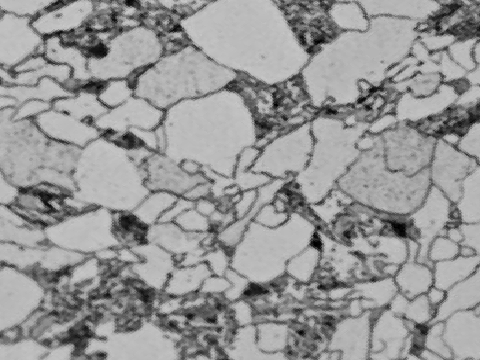
训练结果：

训练集loss曲线

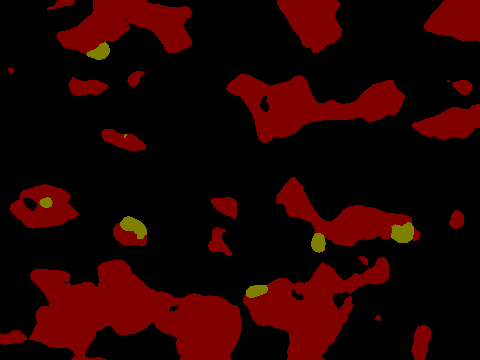


50000次迭代测试结果

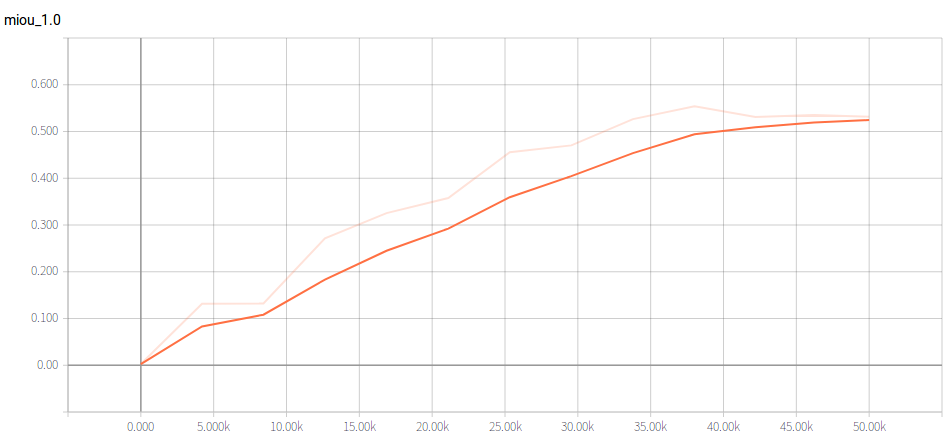
原图 Label

Predoct



测试集mIou曲线



测试结果显示mIou为0.53左右，提升效果不大。

操作VII：（更换为随机梯度下降法进行优化）

learning\_policy: poly

base\_learning\_rate:0.001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:50000

momentum:0.9

train\_batch\_size:3

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

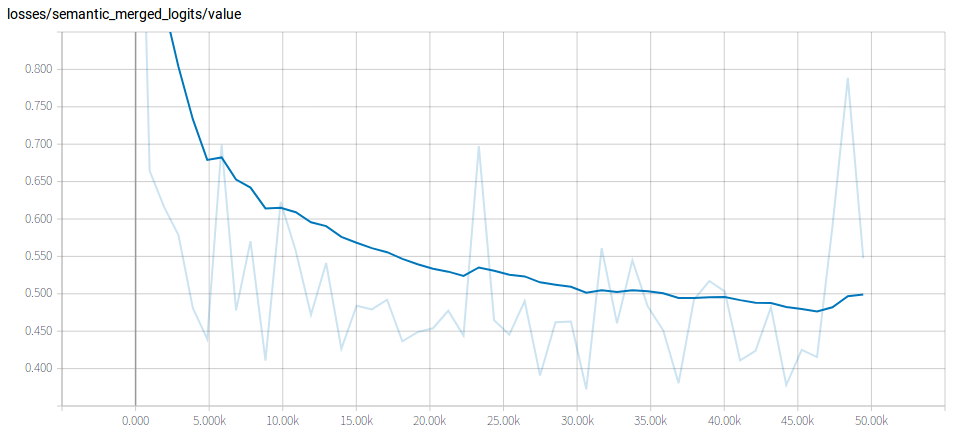
output\_stride:16

decoder\_output\_stride:4

model\_variant:"xception\_71"

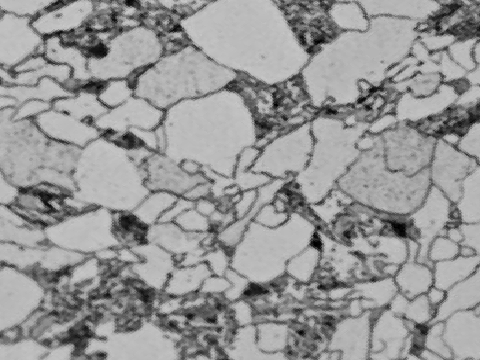
训练结果：

训练集loss曲线

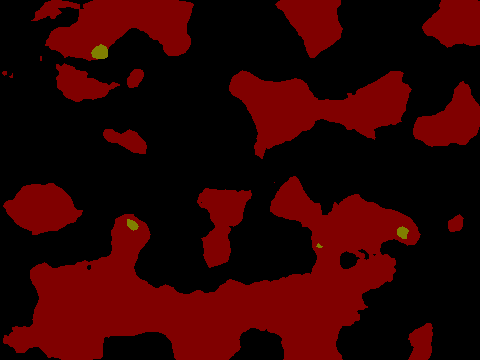


50000次迭代测试结果

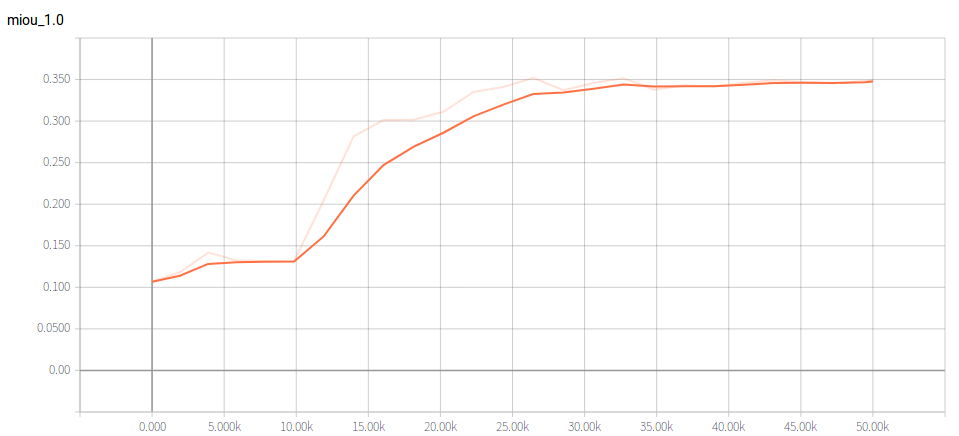
原图 Label

Predict



测试集mIou曲线



将优化器换为SGD后mIou下降较多，0.35左右。

操作VIII：（用output\_stride=16进行训练，然后用output\_stride=8进行测试）

learning\_policy: poly

base\_learning\_rate:0.001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:50000

momentum:0.9

train\_batch\_size:2

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

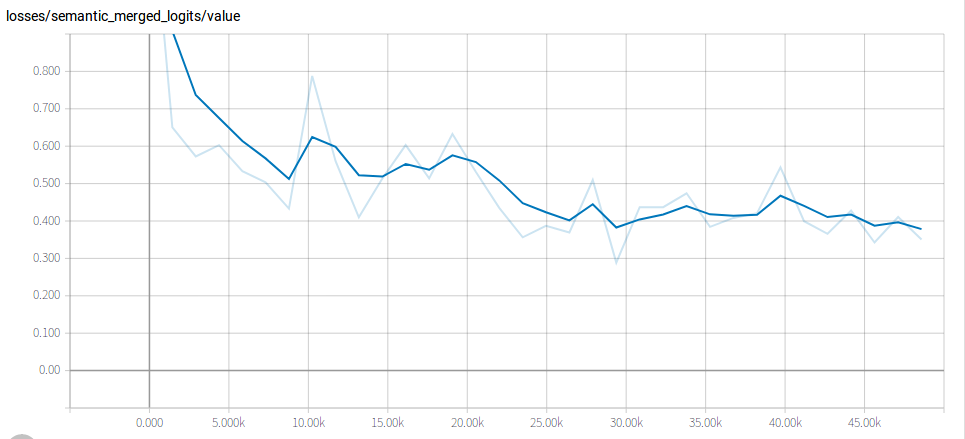
output\_stride:16

decoder\_output\_stride:4

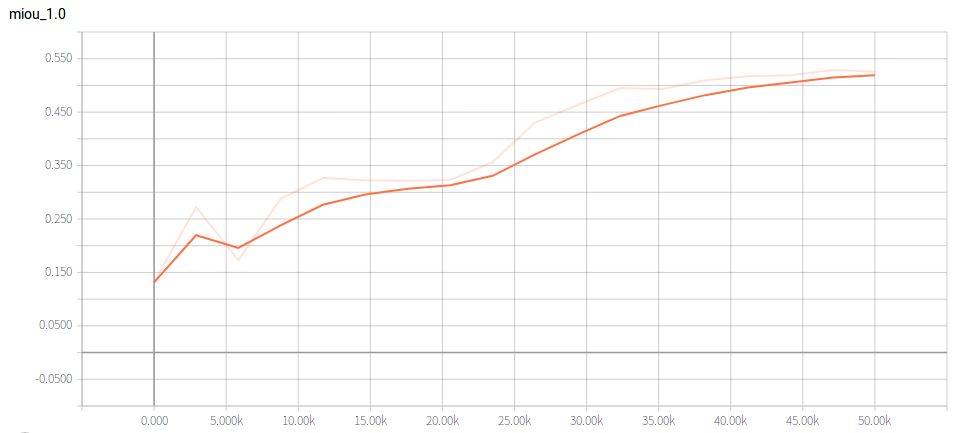
model\_variant:"xception\_71"

训练结果：

训练集loss曲线



测试集mIou曲线



经过实验发现在训练时，提取outstride为16的feature map进行上采样，然后用outstride为8的featuremap进行测试，发现mIou为0.528左右，对分割结果影响不大。

操作IX：（用output\_stride=8）

learning\_policy: poly

base\_learning\_rate:0.001

learning\_rate\_decay\_factor:0.1

learning\_rate\_decay\_step:2000

learning\_power:0.9

training\_number\_of\_steps:50000

momentum:0.9

train\_batch\_size:2

weight\_decay: 0.00004

train\_crop\_size:[360,480]

atrous\_rates:[6,12,8]

output\_stride:8

decoder\_output\_stride:4

model\_variant:"xception\_71"

此实验失败，原因是，由于基于tensorflow的框架，随着程序的运行，内存逐渐增大，由于硬件条件的有限，造成内存泄露。将outstride设为8，相当于图片的尺寸增大，同时，需要消耗更多的计算资源。