

# GOALS 挑战赛-青光眼检测

队伍名：嘟嘟的团队

队长：郎督（东北大学-信息科学与工程学院）

队员：刘佳新（北京航空航天大学-前沿科学技术创新研究院）

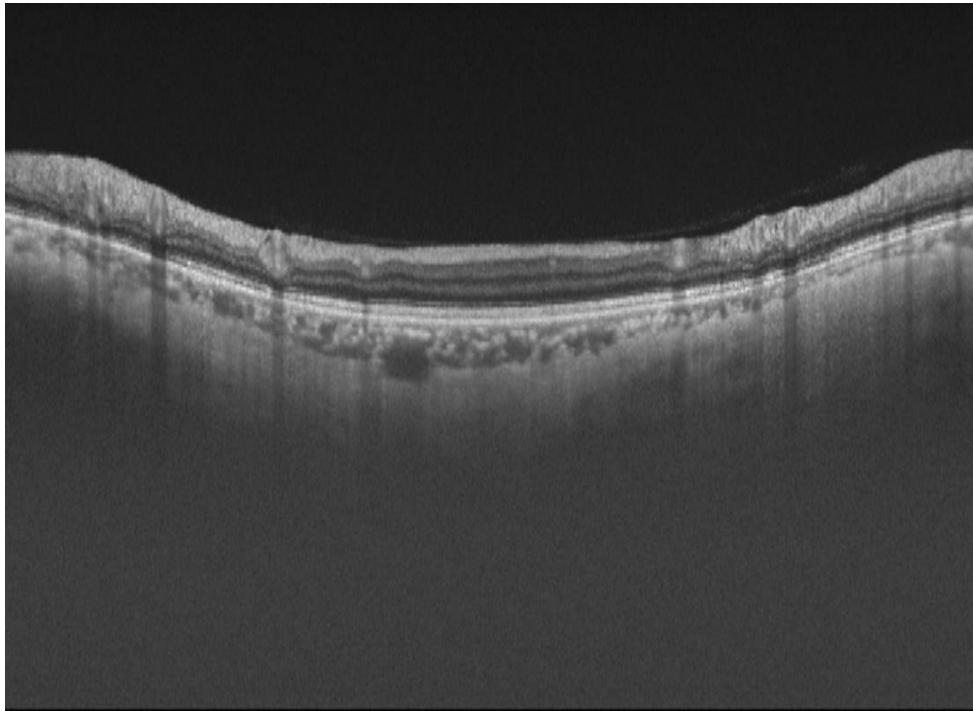
白昊东（东北大学-信息科学与工程学院）

## 一、概述

青光眼是一种慢性神经退行性疾病, 是世界上导致不可逆转但可预防的失明的主要原因之一。它是一种多因素的视神经病变, 其特征是视网膜神经节细胞及其轴突进行性神经变性, 导致视网膜神经纤维层衰减, 是视神经头损伤和视野丧失的一种特殊形式。光谱域 OCT 在临床应用中得到了广泛的应用, 可提供视神经头、视神经纤维层和黄斑的定量评估, 并能检测青光眼随时间的纵向结构丧失。本报告是基于 OCT 图像的青光眼诊断任务报告。

## 二、数据集介绍

GOALS 提供的数据集由中国广州中山大学中山眼科中心提供, 数据集中包含 300 个视盘旁环扫 OCT 数据 (Circumpapillary Optical Coherence Tomography, Circumpapillary OCT)。我们将样本三等分, 分别为训练过程、预赛过程和决赛过程提供 100 个数据。



图一、数据示例

## 三、方案设计

经测试，分类任务的 baseline 可以得到非常好的结果，我们修改 baseline 的网络部分，直接采用 CNN 网络 ResNet50 完成分类预测任务，并对预测的概率进行阈值化处理，获得了很好的效果。

### 1、数据划分

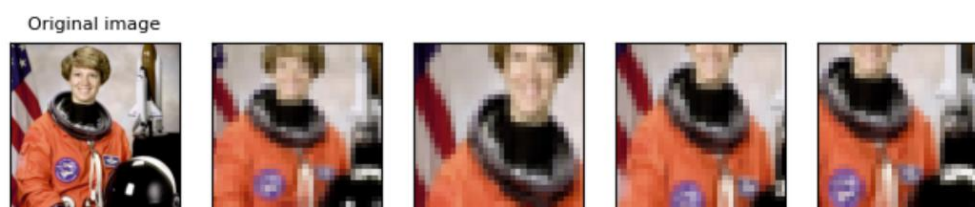
对官方提供的 100 张数据进行划分，训练集验证集划分比例为 8：2，80 张图片作为训练集，20 张图片作为验证集。

### 2、数据预处理

数据增强策略采用如下策略：RandomResizedCrop，RandomHorizontalFlip，RandomVerticalFlip，RandomRotation，各函数具体参数见实现细节章节。

#### （一）RandomResizedCrop

RandomResizedCrop 在图像的随机位置裁剪指定大小的图像，然后将裁剪的图片 resize 到指定大小。该方法包含 3 个参数 size, scale, ratio, size 为输出的图片大小，scale 为裁剪的图像和原图的比例，ratio 为裁剪后的图像宽高比范围，下图为示例：



图二、RandomResizedCrop

#### （二）RandomHorizontalFlip

RandomHorizontalFlip 是随机对图像进行水平翻转，该函数只有一个参数 p，表示翻转的概率，示例如下：



图三、RandomHorizontalFlip

(三) RandomVerticalFlip

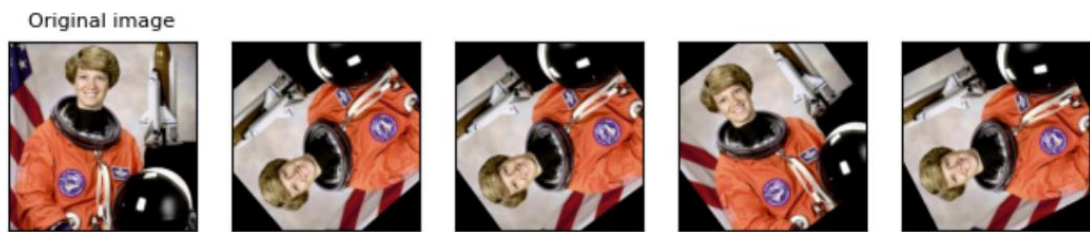
RandomVerticalFlip 是随机对图像进行水平翻转，该函数只有一个参数 p，表示翻转的概率，示例如下：



图四、RandomVerticalFlip

(四) RandomRotation

RandomRotation 是随机对图像随机旋转，可以设置旋转的角度范围，示例如下：



图五、RandomRotation

3、模型选择

选择 ResNet50 作为分类模型, 输出类别设置为 2(与 baseline 的不同在于移除了 baseline 中后加的 2 个 Linear 层)，并加载 ImageNet-1k 的预训练权重，未对网络结构进行修改，resnet50 网络结构如下：

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112			7×7, 64, stride 2		
				3×3 max pool, stride 2		
conv2_x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1			average pool, 1000-d fc, softmax		
FLOPs		$1.8 \times 10^9$	$3.6 \times 10^9$	$3.8 \times 10^9$	$7.6 \times 10^9$	$11.3 \times 10^9$

图六、网络结构

## 4、后处理

网络预测的结果是类别对应的概率，为了提高指标，对预测的概率进行阈值化处理，公式如下：

$$P(x) = \begin{cases} 0 & \text{if } x \leq 0.5 \\ 1 & \text{else} \end{cases}$$

## 5、实现细节

使用 python 语言，paddlepaddle 深度学习框架完成训练-预测，相关配置如下表：

PaddlePaddle	2.3.0
GPU	Tesla V100 32GB
CUDA	10.1

表一、配置

数据增强函数及参数设置：

函数	参数
paddle.vision.transform.RandomResizedCrop	size=512 scale=(0.90, 1.1) ratio=(0.90, 1.1)
paddle.vision.transform. RandomHorizontalFlip	prob=0.5
paddle.vision.transform. RandomVerticalFlip	prob=0.5
paddle.vision.transform. RandomRotation	degrees=30

模型函数及参数：

函数	参数
resnet50	pretrained=True num_classes=2

其他设置：

batch size	8
iters	1000
Learning rate	1e-6
optimizer	Adam

## 四、参考文献

[1]Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. "Deep Residual Learning for Image Recognition." (2015).

[2]Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, Liang-Chieh Chen. "Mobilenetv2: Inverted Residuals and Linear Bottlenecks." (2019).

[3]Zhuang Liu, Hanzi Mao, Chao-Yuan Wu,Christoph Feichtenhofer,Trevor Darrell,Saining Xie. "A

Convnet for the 2020s." (2022).

# GOALS 挑战赛-环扫 OCT 图像的层分割

队伍名：嘟嘟的团队

队长：郎督（东北大学-信息科学与工程学院）

队员：刘佳新（北京航空航天大学-前沿科学技术创新研究院）

白昊东（东北大学-信息科学与工程学院）

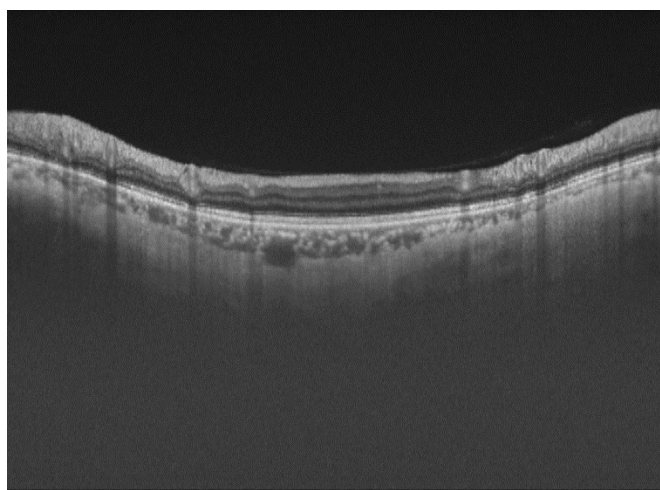
## 一、概述

青光眼是一种慢性神经退行性疾病, 是世界上导致不可逆转但可预防的失明的主要原因之一。它是一种多因素的视神经病变, 其特征是视网膜神经节细胞及其轴突进行性神经变性, 导致视网膜神经纤维层衰减, 是视神经头损伤和视野丧失的一种特殊形式。光谱域 OCT 在临床应用中得到了广泛的应用, 可提供视神经头、视神经纤维层和黄斑的定量评估, 并能检测青光眼随时间的纵向结构丧失。本报告是基于 OCT 图像的青青光眼图像分割报告。

## 二、数据集介绍

GOALS 提供的数据集由中国广州中山大学中山眼科中心提供, 数据集中包含 300 个视盘旁环扫 OCT 数据 (Circumpapillary Optical Coherence Tomography, Circumpapillary OCT)。我们将样本三等分, 分别为训练过程、预赛过程和决赛过程提供 100 个数据。

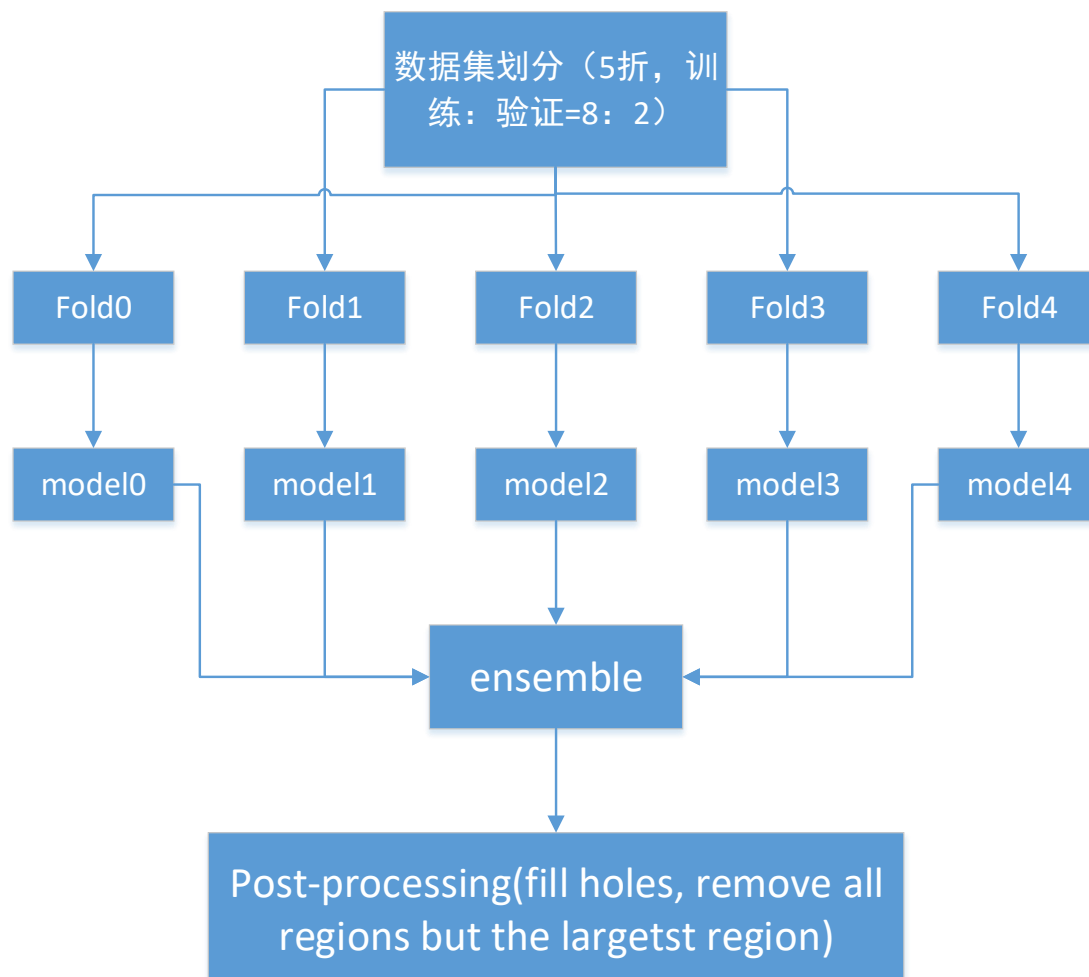
各数据的初始 RNFL, GCIPL 和脉络膜层分割区域标注由中国中山大学中山眼科中心 5 名临床眼科医生手动完成。这 5 名医生在没有获得任何患者信息及图像对应疾病信息的情况下, 独立对图像中 RNFL, GCIPL 和脉络膜层区域的边缘线进行勾勒。随后, 5 个初始标注结果汇总给更高级医生进行融合。标注结果的融合采用取平均边缘线的方式, 融合医生检查初始的边缘线标注结果, 去掉标注误差大的初始边缘线, 将剩余的初始边缘线取平均, 即可得到目标区域边缘线金标准。随后将 RNFL, GCIPL 和脉络膜层区域边缘线内像素点赋不同像素值以获得最终的层分割金标准。



图一、数据示例

### 三、方案设计

数据集一共仅包含 100 张图片，数据非常的少，为了提高分割模型的鲁棒性，我们使用 5 折交叉验证训练 5 个模型，在预测时融合模型，并使用空洞填充、保留最大连通区域等后处理方法来提高分割结果。流程图如下所示：



图二、流程图

#### 1、数据划分

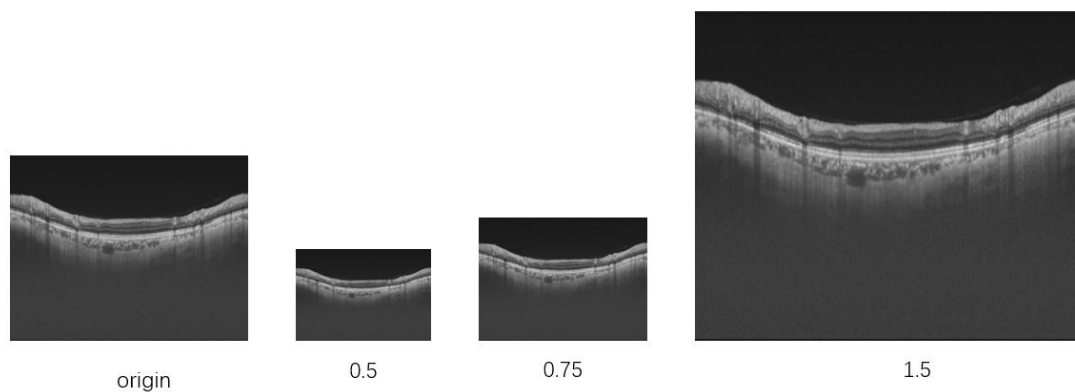
使用了五折交叉验证的方法，对官方提供的 100 张数据进行划分，训练集验证集划分比例为 8: 2，80 张图片作为训练集，20 张图片作为验证集。每折的验证集无重复。

#### 2、数据预处理

数据增强策略采用如下策略：ResizeStepScaling，RandomPaddingCrop，RandomHorizontalFlip，RandomDistort，各函数具体参数见实现细节章节。

## (一) ResizeStepScaling

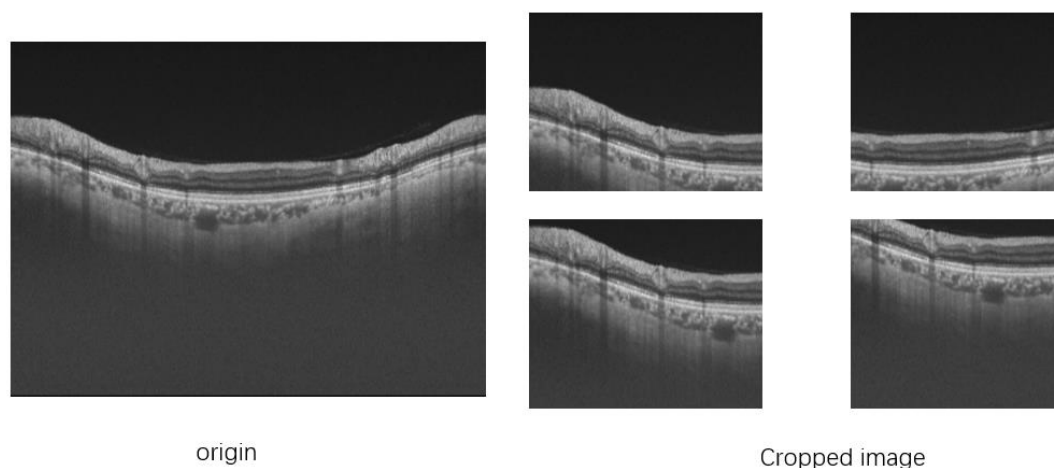
ResizeStepScaling 是对图像进行随机缩放，该方法包含 min\_scale\_factor, max\_scale\_factor, scale\_step\_size 3 个参数，min\_scale\_factor 是最小缩放比例，max\_scale\_factor 是最大缩放比例，scale\_step\_size 为缩放比例步长，下图为示例：



图三、ResizeStepScaling

## (二) RandomPaddingCrop

RandomPaddingCrop 是随机对图像进行裁剪，并对大小小于指定裁剪尺寸的图片进行填充，该方法包含 crop\_size, im\_padding\_value, label\_padding\_value 共 3 个参数，crop\_size 为裁剪后的图片尺寸，im\_padding\_value 为填充区域的像素值，label\_padding\_value 为填充区域对应 mask 的标签值，示例如下：



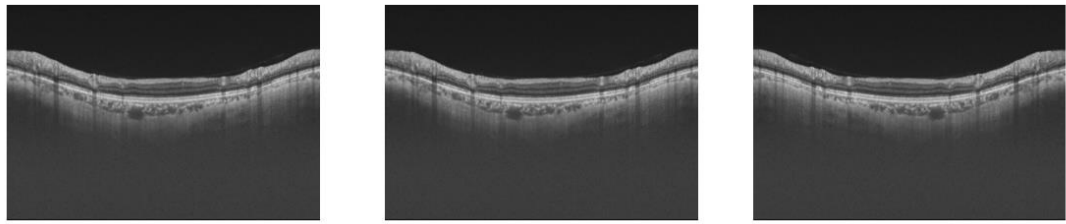
图四、RandomPaddingCrop

## (三) RandomHorizontalFlip

RandomHorizontalFlip 是随机对图像进行水平翻转，该方法只有一个参数 p，表示翻转



的概率，示例如下：



图四、RandomHorizontalFlip

#### (四) RandomDistort

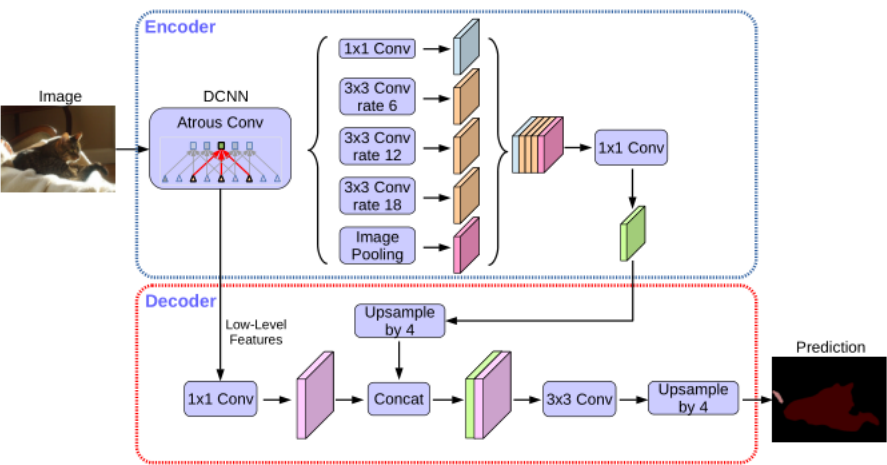
RandomDistort 是对图像的亮度、对比度、饱和度、色调、锐度进行一定范围的随机调整，该方法主要参数有 brightness\_range, contrast\_range, saturation\_range, hue\_range, sharpness\_range, brightness\_range 为亮度调整范围，contrast\_range 是对比度调整范围，saturation\_range 为饱和度调整范围，hue\_range 为色调调整范围，sharpness\_range 为锐度调整范围，示例如下：



图五、RandomDistort

### 3、模型选择

我们选择 deeplabv3p 作为分割模型，output stride 为 8，backbone 选择 resnet101，并加载在 ImageNet-1k 上的预训练参数，模型结构如下(使用 PaddleSeg 提供的 Deeplabv3p，未对结构进行修改)：



图六、网络结构

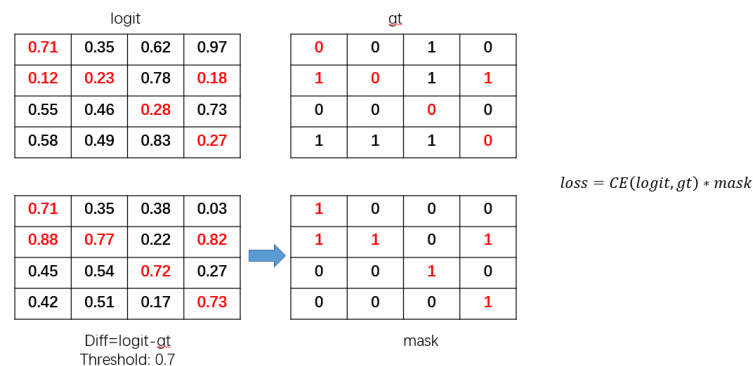
## 4、损失函数

损失函数选择 OhemCrossEntropyLoss 和 RelaxBoundaryLoss 2 个损失函数结合，损失函数公式如下：

$$L_{seg} = L_{ohem} + L_{relaxboundary}$$

### 1、OhemCrossEntropyLoss

OhemCrossEntropyLoss 设定一个阈值，如果预测概率与真实值的差低于该阈值，则认为该样本为难分类样本，如果预测概率和真实值的差高于该阈值，则认为该样本是易分类样本。在计算损失函数时，只计算难分类样本的损失。其计算原理如下：

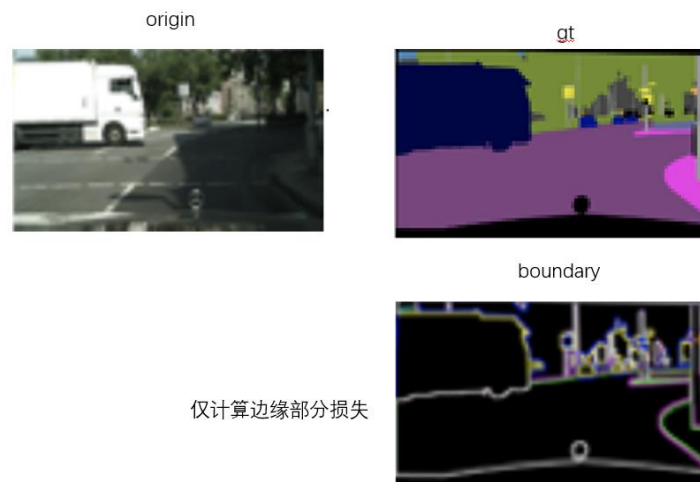


图七、OhemCrossEntropyLoss

我们在 OhemCrossEntropyLoss 中添加了 label smooth。在传统的分类任务计算损失的过程中，是将真实的标签做成 one-hot 的形式，然后使用交叉熵来计算损失。而 label smoothing 是将真实的 one hot 标签做一个标签平滑处理，使得标签变成又概率值的 soft label.其中，在真实 label 处的概率值最大，其他位置的概率值是个非常小的数。在 label smoothing 中有个参数 epsilon，描述了将标签软化的程度，该值越大，经过 label smoothing 后的标签向量的标签概率值越小，标签越平滑

### 2、RelaxBoundaryLoss

为了更好地分割边缘，引入 RelaxBoundaryLoss，该损失函数仅对 GT 的边缘区域计算 loss，可以提高边缘位置的分割结果。



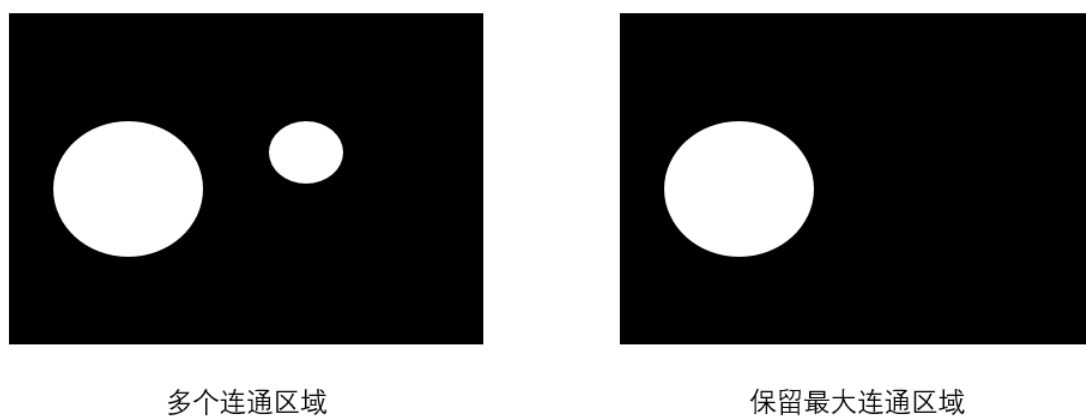
图八、RelaxBoundaryLoss

## 5、后处理

因为数据集中每个类别都仅包含一个连通区域, 使用孔洞填充和保留最大连通区域的方法来优化分割结果, 示例图如下:



图九、孔洞填充



图十、保留最大连通区域

## 6、实现细节

使用 python 语言，paddlepaddle 深度学习框架，使用 PaddleSeg 语义分割套件完成分割任务，相关配置如下表：

PaddlePaddle	2.3.0
GPU	Tesla V100 32GB
CUDA	10.1
PaddleSeg	Release/2.6

表一、配置

为了提升模型的鲁棒性，我们采用五折交叉验证的方法，划分了 5 个数据集，对五个数据集分别训练一个模型，所有模型的训练参数一致。在预测时，将五个模型的预测结果取平均作为预测结果，同时使用了预测时数据增强策略（左右翻转，上下翻转，比例缩放等，数据增强策略与上述相同）。

模型及参数设置：

函数	参数
paddleseg.models.DeepLabV3P	backbone: type: ResNet101_vd output_stride: 8 multi_grid: [1, 2, 4] pretrained: <a href="https://bj.bcebos.com/paddleseg/dygraph/resnet101_vd_ssld.tar.gz">https://bj.bcebos.com/paddleseg/dygraph/resnet101_vd_ssld.tar.gz</a> num_classes: 4 backbone_indices: [0, 3] aspp_ratios: [1, 12, 24, 36] aspp_out_channels: 256 align_corners: False

数据增强函数及参数设置：

函数	参数
ResizeStepScaling	min_scale_factor: 0.75 max_scale_factor: 2.0 scale_step_size: 0.25
RandomPaddingCrop	crop_size: [800, 600]
RandomHorizontalFlip	prob=0.5
RandomDistort	brightness_range: 0.4 contrast_range: 0.4 saturation_range: 0.4

损失函数及参数：

函数	参数
OhemCrossEntropyLoss	label_smooth: True thresh=0.7 min_kept=10000, ignore_index=255
RelaxBoundaryLoss	border=1 calculate_weights=False upper_bound=1.0 ignore_index=255

学习率优化器：

Learning rate	type: PolynomialDecay learning_rate: 0.01 end_lr: 0 power: 0.9
optimizer	type: sgd momentum: 0.9 weight_decay: 4.0e-3

其他设置：

batch size	4
iters	4000
Normalize	mean:[0.22619521,0.22619521,0.22619521] std: [0.16060336, 0.16060336, 0.16060336]

## 四、参考文献

- [1]Chen, Liang-Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, Alan L. Yuille. "DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected Crfs." 40, no. 4 (2017): 834-48.
- [2]Chen, Liang-Chieh, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. "Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation." Paper presented at the Proceedings of the European conference on computer vision (ECCV), 2018.
- [3]He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep Residual Learning for Image Recognition." Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.
- [4]Isensee, Fabian, Jens Petersen, Andre Klein, David Zimmerer, Paul F Jaeger, Simon Kohl, Jakob Wasserthal, *et al.* "Nnu-Net: Self-Adapting Framework for U-Net-Based Medical Image

Segmentation." (2018).

- [5]Li, Xiangtai, Xia Li, Li Zhang, Guangliang Cheng, Jianping Shi, Zhouchen Lin, Shaohua Tan, and Yunhai Tong. "Improving Semantic Segmentation Via Decoupled Body and Edge Supervision." Paper presented at the European Conference on Computer Vision, 2020.
- [6]Liu, Yi, Lutao Chu, Guowei Chen, Zewu Wu, Zeyu Chen, Baohua Lai, and Yuying %J arXiv preprint arXiv:06175 Hao. "Paddleseg: A High-Efficient Development Toolkit for Image Segmentation." (2021).
- [7]Shrivastava, Abhinav, Abhinav Gupta, and Ross Girshick. "Training Region-Based Object Detectors with Online Hard Example Mining." Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.