

模式识别与机器学习 大作业

2017011954 吾尔开希 自 76

2020 年 6 月 17 日

摘要

在这次大作业中，dataset1 的细胞分割任务我们主要采用了语义分割与后处理结合的方式来完成。语义分割部分采用深度学习的方法来解决，将输入图片分成前景和背景，得到二值图。后处理部分采用分水岭算法和数学形态学的方法，进行比较精细的处理，将二值图像实例化，分成单独的细胞。我们尝试了多种数据增广方法，包括随机裁剪，随机缩放，随机翻转等。此外，我们还尝试了多种训练参数与模型结构。为避免过拟合，在提交结果时采用了模型集成的方法。解决 dataset2 任务时，我们与其他七位同学一起标注了训练数据集，从而扩充数据。然而 dataset2 数据的特点是细胞粘连严重，这给实例化带来了巨大的问题。为解决这个问题，我们在 dataset1 方法的基础上，在语义分割部分添加了对每个细胞边界的预测，利用边界在实例化时提取每个细胞的核，从而将不同的细胞更好的分开。由于手动标注的数据集较粗糙，所以在后期也尝试了使用网上的数据集（去掉网站测试集），来证明我们方法的有效性。针对 dataset1 的任务我们在不同的参数下进行了五十多次训练，尝试了多种后处理方法，在网站上提交最高的成绩是 0.840388。dataset2 我们进行了十余次训练，使用手动标注数据集训练的模型得到最高的成绩是 0.830856，使用网上的 DIC-HeLa 数据集训练得到最高的成绩约是 0.87。

目录

1 文献调研	2
1.1 实例分割	2
1.2 语义分割 + 后处理	3
1.2.1 网络结构	3
1.2.2 数据增广	3
1.2.3 模型平衡	4
1.2.4 后处理：分水岭算法与数学形态学	4
2 数据处理流程	5
2.1 dataset1 任务	5
2.1.1 数据集分割	5
2.1.2 数据增广	5
2.1.3 标准化	6
2.2 dataset2 任务	6

3 算法原理	7
3.1 dataset1 任务	7
3.2 dataset2 任务	8
4 实现过程	8
4.1 dataset1 任务算法实现	8
4.2 dataset2 任务算法实现	10
5 实验结果与模型性能分析	10
5.1 dataset1 的实验	11
5.1.1 模型训练收敛实验	11
5.1.2 模型结构实验	12
5.1.3 数据增广实验	12
5.1.4 损失函数实验	13
5.1.5 后处理参数实验	13
5.1.6 模型集成	13
5.1.7 总结	14
5.2 dataset2 的实验	14
5.2.1 边界生成方式	14
5.2.2 边界预测方式	14
5.2.3 损失函数	14
5.2.4 总结	14
6 总结	15
7 致谢	15

1 文献调研

实例分割问题一般有两种思路，一种是端到端的实例分割训练方法，另一种是语义分割 + 后处理的方法。无论是 dataset1 任务还是 dataset2 任务，算法的基本思路都是分成语义分割和后处理两部分，因此主要关注这两大方面。语义分割部分采用深度学习的方法，我们调研了深度学习中语义分割相关网络的结构，数据增广方法和模型平衡方法。后处理部分与分水岭算法以及数学形态学有关，因此调研这两部分相关文献。

1.1 实例分割

实例分割任务最直观的做法是先进进行目标检测，再进行语义分割，两个任务端到端连接实现共同训练。Kaiming He 团队在 2017 年提出的 Mask R-CNN 模型通过在目标检测任务的基础上添加语义分割的分支的方法，既简化了模型，又大幅提高了模型性能 [5]。

然而细胞分割任务使用实例分割是否合适？我们通过调研发现答案是否定的，原因有两点。首先，一般的实例分割任务中，不同实例的之间有本质的差距，比如图片中杯子的实例分割任

务，数据集中会有多种杯子，不同杯子之间的形状、颜色、摆放方式都是不同，因此先进行语义分割效果往往较差。然而在我们的细胞分割任务中，不同细胞之间只是形状大小的区别，本质是相同的，且图片的拍摄视角都是一致，因此先进行语义分割更合适。其次，实例分割方法中的目标检测往往是使用方形框，这与细胞形状不符，对有粘连细胞的分割会带来准确率的损失。因此，我们认为使用语义分割 + 后处理的方式是更为简单有效的。

1.2 语义分割 + 后处理

1.2.1 网络结构

深度学习模型受到广泛应用以来，研究者们也将眼光投向了语义分割问题。Ronneberger 等人在 2015 年创造性的提出 U-Net 模型结构，使用深度卷积神经网络进行医疗图像分割，与已有方法相比在效率和准确率方面都得到了巨大的提升 [7]。U-Net 的模型结构如图 1。U-Net 结构由特征提取器和拓展器构成，两部分结构对称。特征提取器通过池化操作得到多个不同尺寸的 feature map，这些 feature map 同时被送入拓展器的对应位置，从而实现不同尺寸特征的提取和感知。

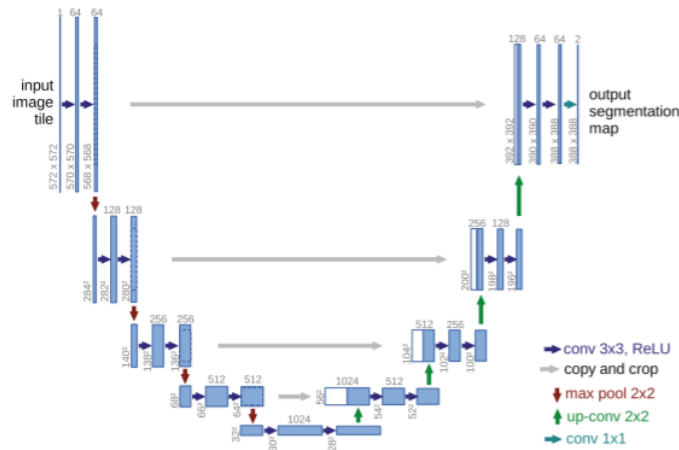


图 1: U-Net 模型结构

研究者们也在 U-Net 的基础上进行了多种改进，Oktay 等人提出的 Attention U-Net 在 U-Net 的基础上添加了注意门机制，从而使网络显式地关注与任务目标相关的区域特征 [6]。Attention U-Net 结构如图 2。此外，Alom 等人提出 R2U-Net，在 U-Net 的基础上添加循环卷积结构，从而在保持参数量不变的情况下提高模型的性能 [1]。也可以将 Attention U-Net 和 R2U-Net 相结合，得到更好的效果。另外，还可以将 U-Net 的特征提取部分换成 resnet，提高模型的表达能力。

1.2.2 数据增广

由于训练集和测试集的原图色调没有明显的变化，因此我们主要关注针对特征尺寸的数据增广。常用的数据增广方式包括随机裁剪，随机水平翻转，随机竖直翻转，随机缩放，随机旋转等 [4]。需要注意的是，数据增广操作不仅要原图做，也要对 ground truth 实施完全一样的操作。

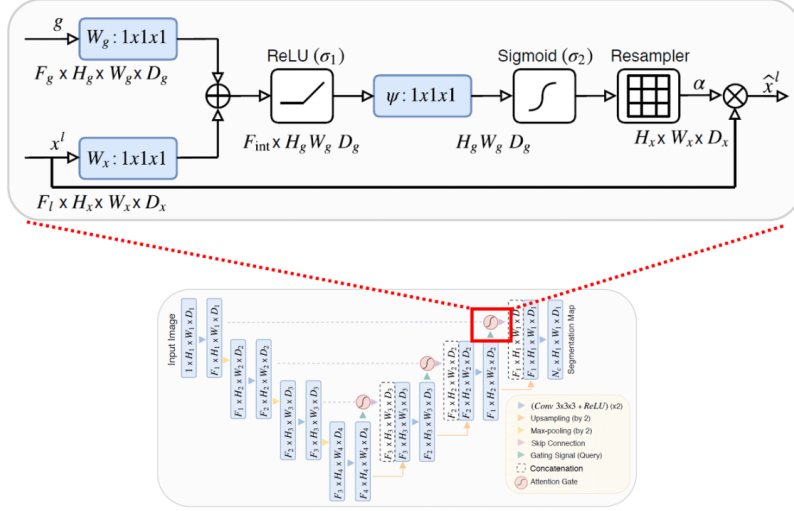


图 2: Attention U-Net 模型结构

1.2.3 模型平衡

语义分割模型的训练一般是将 binary cross-entropy 作为损失函数，而将 IoU 作为评价指标，IoU 的计算方法如公式 1

$$IoU(A, B) = \frac{A \cap B}{A \cup B} \quad (1)$$

binary cross-entropy 关注每个像素点的预测值，其与 IoU 的评价标准不同，比如一个 4 像素大小的细胞和 400 像素大小的细胞，两者在 IoU 计算中重要性是等同的，但在 binary cross-entropy 中差别却很大。一般解决这个问题的方法是使用 dice loss 来训练，dice loss 的表达式如公式 2。但如果直接用 dice loss 作为损失函数来训练，训练过程极不稳定。为了解决这个不平衡问题，Berman 等人提出了 Lovasz-Softmax loss [2]。

$$diceloss(pred, target) = 1 - \frac{A \cap B}{A + B} \quad (2)$$

此外，还有正负样本不平衡的问题等，可以通过给正负样本的损失函数加权值的方式来调节 [4]。

1.2.4 后处理：分水岭算法与数学形态学

分水岭算法广泛应用于图像分割，属于传统的无监督算法。其主要思路是从图片的若干种子点出发，根据地势高度（图像灰度）不断拓展，与水位不断上升的情形类似 [8]，最后在水域相交点设置分水岭，作为分割边界，算法部分过程如图 3。

数字形态学也常在各种图像处理任务中有所应用，主要用于改变图形的几何构型，其基本操作包括腐蚀、膨胀、开闭运算等 [3]。

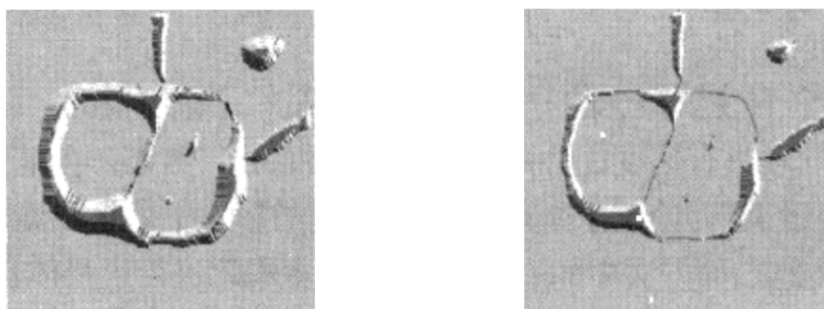


图 3: 分水岭分割算法实例

2 数据处理流程

2.1 dataset1 任务

2.1.1 数据集分割

训练集一共有 175 张图片，我们将这些图片五等分或十等分，取其中一份作为验证集，剩下的作为训练集。一方面是为了方便比较不同参数下的模型性能，节省网站提交次数；另一方面也是为了在不同训练集下训练得到多个模型，用于模型集成。

2.1.2 数据增广

我们用到的数据增广方式包括随机裁剪，随机水平翻转，随机竖直翻转，随机缩放，随机旋转，对输入数据随机处理，并训练多个 epoch，就相当于数据增广。在进行随机增广时，要注意对输入图片和 ground truth 进行完全一致的处理。使用全部 5 种增广方式的流程如图 4。

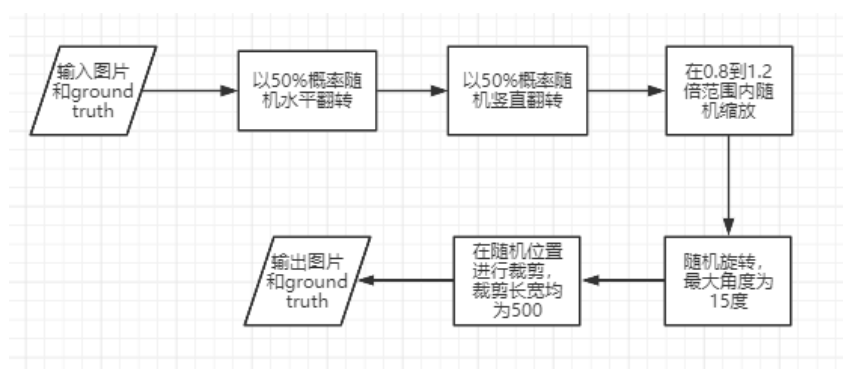


图 4: 数据增广流程

如果训练集有随机裁剪，那么训练出的模型在预测时的输入也应该是同样大小的裁剪结果，否则感受野范围的不同会严重影响模型性能。为了输出完整的分割结果，预测时需要使用 crop tile 方法，即先将图片有重叠的按一定大小分割（与训练时的随机分割大小相等），再将这些分割结果送入模型，得到多个预测结果，分别加到预测输出对应的区域，最后在预测输出的分割重叠区域除以重叠次数。

由于图片的背景不是单值的 0，而是带有噪声的灰色，所以在随机旋转的数据处理方式

时，处理结果会出现明显的原图方框，即使用背景的平均值来填充旋转后的空余部分，边界方框还是无法消除，如图 5。这样的边界方框可能会影响模型训练，我们用实验证实了这一结论，因此在最终版本中没有使用随机旋转。

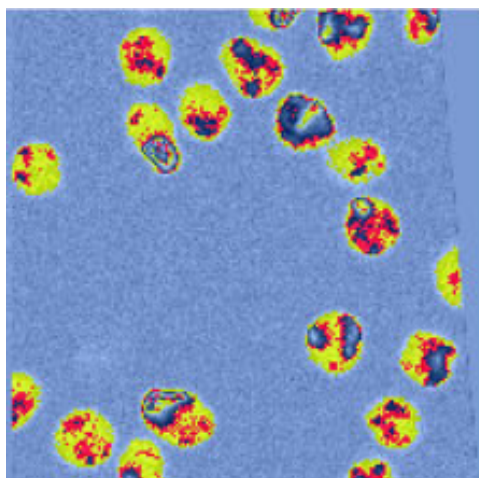


图 5: 随机旋转时图片出现边界方框

2.1.3 标准化

我们将一通道的图片复制成三通道，再压缩成 0 到 1 的小数，并用均值和方差进行标准化。

2.2 dataset2 任务

dataset2 数据集只有 8 张，数据量不足，因此我们与其他 7 位同学合作采用手动标注的方式标注了剩下的训练集图片，每人标注 20 张。dataset2 任务中的数据集分割以及数据增广方法与 dataset1 任务一致。由于 dataset2 任务的模型需要额外预测细胞边界，因此我们需要制作细胞边界的 ground truth。我们利用腐蚀操作从实例分割 ground truth 中提取出边界，边界的宽度与细胞大小的平方正成正比，其计算流程如图 6。其中，边界宽度 = $\sqrt{\frac{\text{细胞像素个数}}{18000}} \times 12$ 。

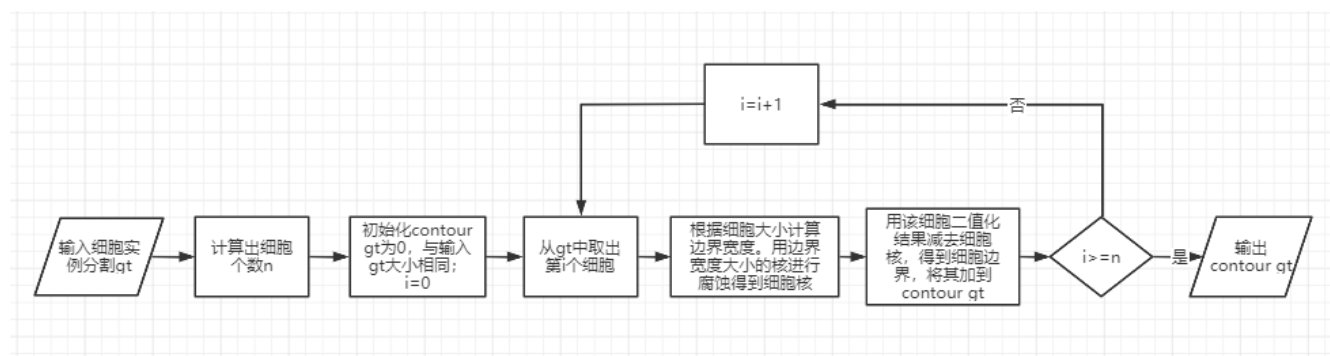


图 6: 边界提取流程

得到的边界 ground truth 是一幅二值图像，训练集中一幅图片的处理结果如图 7。



图 7: 细胞实例以及生成的边界二值图

3 算法原理

文献调研部分已经指出，语义分割 + 后处理的方法更适合细胞分割问题的求解，原因有两点，第一，一般的实例分割任务中，不同实例的之间有本质的区别，然而在我们的细胞分割任务中，不同细胞之间只是形状大小的区别，本质是相同的，且图片的拍摄视角都是一致，因此先进行语义分割更合适。第二，实例分割方法中的目标检测往往是使用方形框，这与细胞形状不符，对有粘连细胞的分割会带来准确率的损失。因此，我们认为使用语义分割 + 后处理的方式是更为简单有效的。

下面，我们来分别介绍一下 dataset1 任务中语义分割和后处理的算法原理，以及 dataset2 任务中添加边界预测的原因和相应的后处理算法原理。

3.1 dataset1 任务

细胞分割任务分成语义分割和后处理两部分，语义分割模型负责将输入图片的前后景分开，后处理部分通过处理二值图像，将其实例化，分割成一个个单独的细胞。

语义分割模块使用文献调研部分介绍的深度神经网络，尝试了 U-Net、Attention U-Net、R2U-Net 等，它们的输入为一定大小的灰度图，输出为与输入同样大小的二值图，输出为 1 的部分表示前景（细胞），为 0 的部分表示背景。这些深度神经网络由特征提取器和拓展器构成，特征提取器通过卷积和池化得到多个不同尺寸的 feature map，这些 feature map 同时被送入拓展器的对应位置，从而实现不同尺寸特征的提取和感知。神经网络的参数都是通过梯度后传算法学习出来的，比较依赖数据集大小和质量。

后处理的核心为分水岭算法，分水岭算法需要若干种子点，作为每个细胞的种子。我们在二值图像上计算每个前景像素点到最近的背景像素点的距离，找出距离的局部极值点，作为分水岭种子点。为了防止一个细胞内出现多个种子点，我们用距离阈值筛选掉一部分种子点，并用数字形态学的膨胀操作将距离较近的种子点连接起来。我们给分离的种子点赋予不同的 label，用距离相反数作为分水岭算法高度，运行分水岭算法得到实例分割初步结果。但由于之前用距离阈值筛选掉了一部分种子点，导致一些小细胞被遗漏，所以我们从语义分割二值图像中减去实例分割初步结果包含的前景，得到遗漏的前景，使用简单的连通域分析将遗漏的前景实例化，加到实例分割初步结果上，得到最终的实例分割结果。这种做法既保证了同一细胞不会被分裂，又确保较小的细胞不被遗漏。我们用测试集中的一张图片展示后处理过程，如图 8，可以看出后

处理比较有效完整的分割了不同细胞。

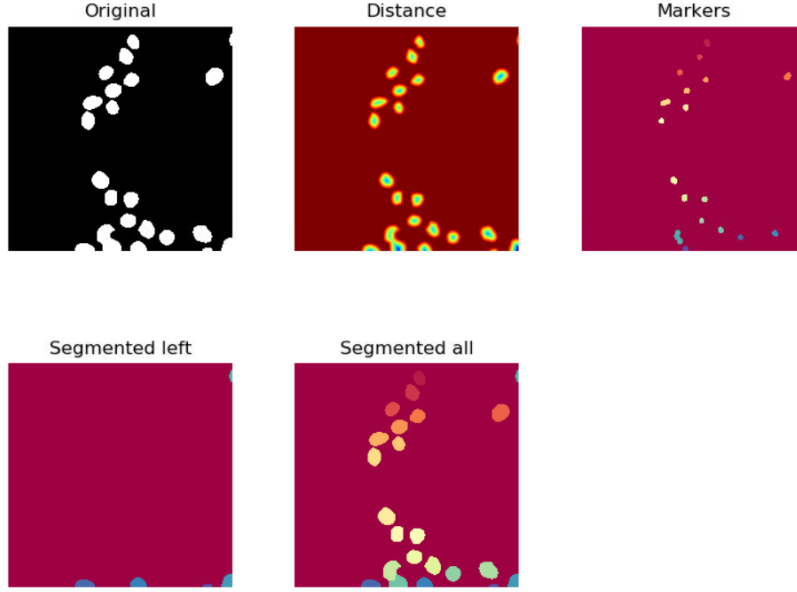


图 8: dataset1 中图片后处理示例，图中依次是语义分割二值图；前景像素到背景最短距离；初步筛选出的种子点；遗漏的细胞；实例分割最终结果

3.2 dataset2 任务

dataset2 的细胞分割任务与 dataset1 不同的是，由于 dataset2 的细胞连接较为紧密，语义分割模型需要额外输出对细胞边界的预测，因此深度神经网络的输出维度从 1 变成 2。另一种方案是训练一个单独的模型预测细胞边界。但实际上语义分割任务和细胞边界分割任务是相互辅助的，实验发现，从同一个模型预测出的细胞边界也可以与前景预测结果更好得契合。

后处理过程与 dataset1 任务有一定区别，输入是语义分割模型得到的前景预测和边界预测结果。用前景预测结果减去边界预测结果，得到细胞核，由于边界预测有些遗漏，所以对细胞核实施开运算，断开不同细胞相连的部分。使用连通域分析方法给不同细胞核赋予不同的 label，再用分水岭算法分出结果，分水岭算法的高度依然是前景像素点到背景像素点的最短距离。我们用测试集中的一张图片展示后处理过程，如图 9，可以看出预测的前景中细胞粘连比较严重，如果使用与 dataset1 同样的方法，那么很多细胞很难分开。

边界预测的训练实际上用到了数据集中的实例分割信息，因此能大幅提高实例化的性能。

4 实现过程

4.1 dataset1 任务算法实现

dataset1 的总体流程如图 10，即先对输入进行预处理，然后通过语义分割模型得到前景预测，再利用后处理将不同的细胞分开。

其中，后处理流程如图 11所示，基本原理已在算法原理部分详述。

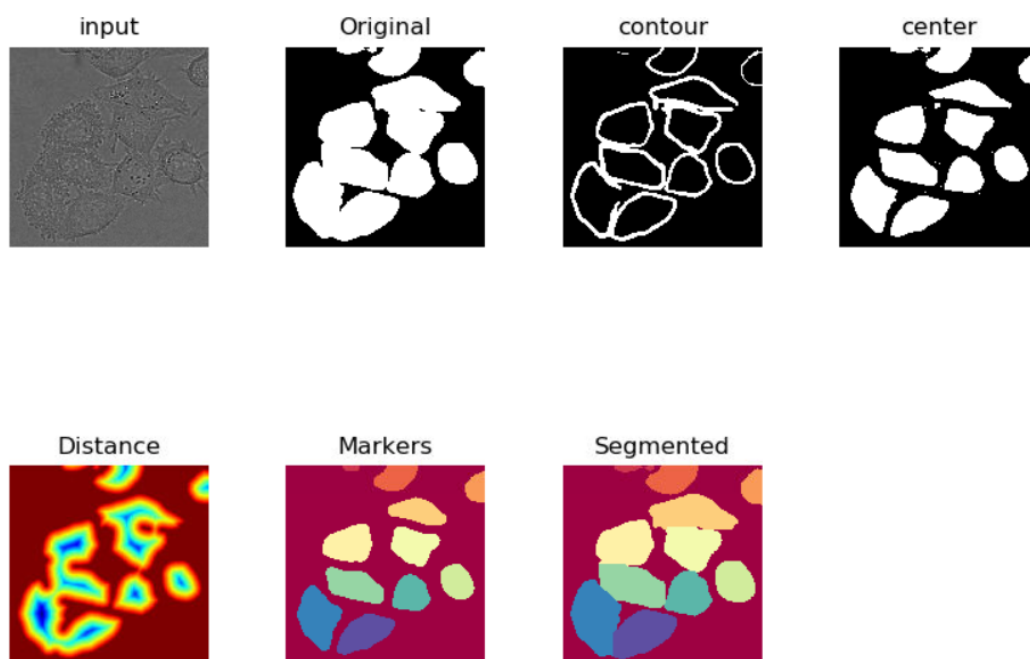


图 9: dataset2 中图片后处理示例，图中依次是前景预测结果；边界预测结果；前景减边界得到细胞核；前景像素点到背景像素点的最短距离；细胞核实例化得到分水岭种子；实例分割最终结果

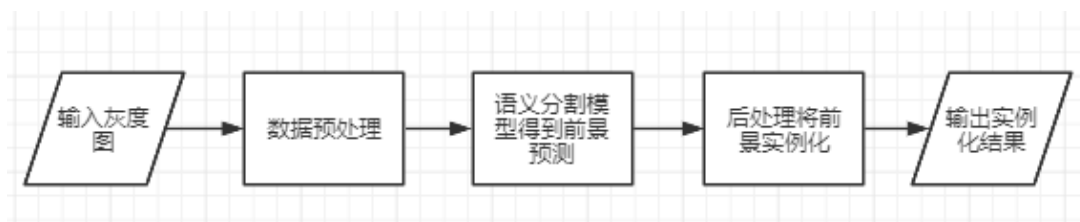


图 10: 数据集 1 总体处理流程

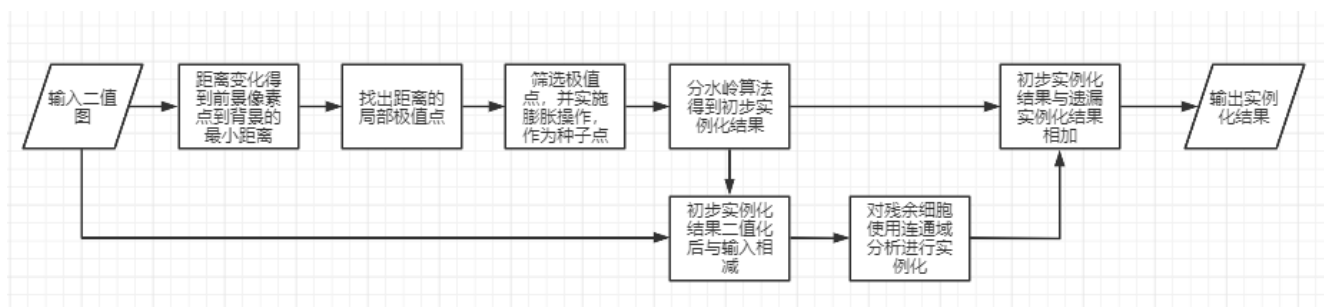


图 11: 数据集 1 后处理流程

4.2 dataset2 任务算法实现

dataset2 的总体流程如图 12，即先对输入进行预处理，然后通过语义分割模型得到前景预测和边界预测，利用这两张二值图通过后处理将不同的细胞分开。

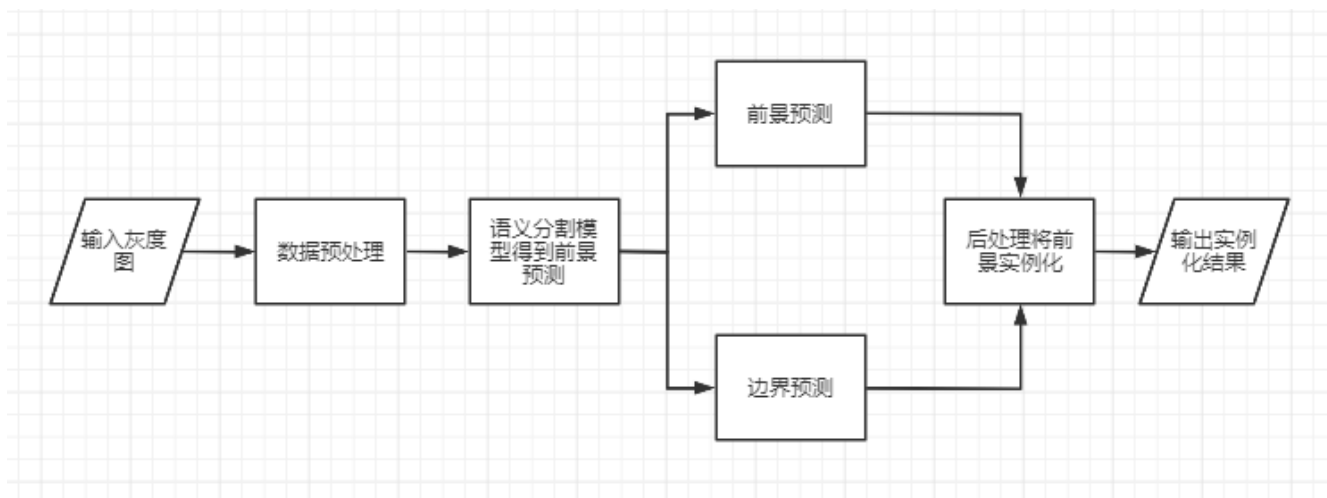


图 12: 数据集 2 总体处理流程

其中，后处理流程如图 13，基本原理已在算法原理部分详述。

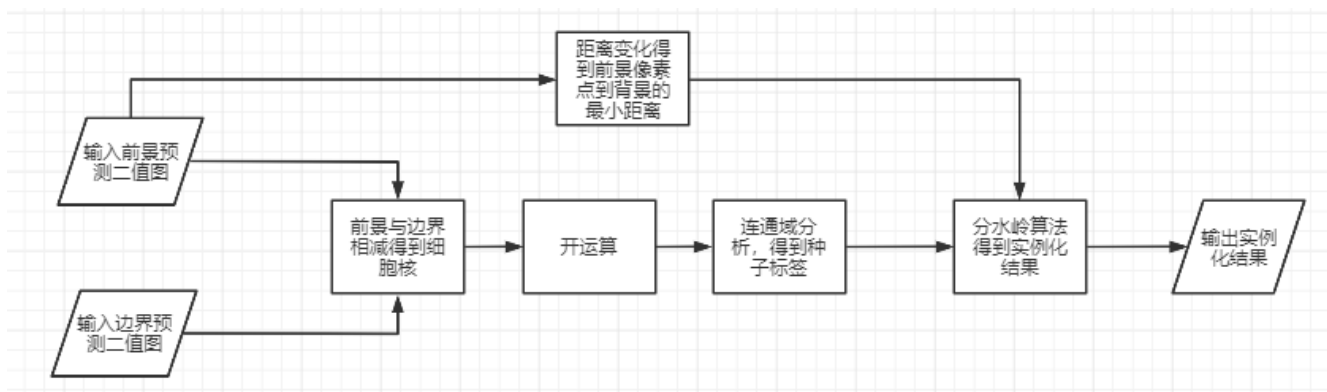


图 13: 数据集 2 后处理流程

5 实验结果与模型性能分析

前期在摸索阶段进行了一些实验，在后处理方面也在一直改进，下面主要呈现方法稳定后有效的比较结果。为了节省网站提交次数，实验之间性能的对比主要依据验证集上的 jaccard 分数。

训练的损失函数有多种尝试，默认情况下为 binary cross-entropy，使用 Adam 优化器进行训练。训练过程中，在验证集上计算 IoU 来评价性能指标。一般取验证集 IoU 最高的模型来计算 jaccard 分数。

Epoch 数	jaccard 分数
40	0.7617
120	0.8095
200	0.8211
250	0.8184

表 1: 不同 epoch 数训练的结果

学习率调整方式	jaccard 分数
0.01	0.8211
0.01->0.001	0.8335
0.01->0.0001	0.8199

表 2: 不同学习率调整方式下训练的结果

5.1 dataset1 的实验

5.1.1 模型训练收敛实验

由于神经网络参数量较大，局部极值点较多，所以需要较多的迭代次数才能达到最优性能，这需从模型性能和损失函数变化曲线来分析。

下面是控制其他条件相同的情况下，不同 epoch 下的训练效果如表 1。可以看出，在 epoch 数达到 200 时，模型的训练基本饱和。

此外，模型的训练程度也与学习率有关，训练过程中学习率的调整使用指数衰减模型，在不同的学习率调整下训练得到的模型性能如表 2。可以看出，学习率指数衰减十倍时，训练出的模型性能较优。

通过观察训练集损失函数的变化以及验证集 IoU 的变化，可以看出模型训练基本收敛，如图 14



图 14: 训练集损失函数和验证集 IoU 的变化

模型结构	jaccard 分数
U-Net 64c	0.8335
U-Net 32c	0.8329
attention U-Net	0.8328
attention R2U-Net	0.8222
resnet50 U-Net	0.8196
U-Net 32c 深度减 1	0.8067

表 3: 不同模型结构下训练的结果

数据增广方式	jaccard 分数
无	0.8329
随机缩放	0.8113
随机水平竖直翻转	0.8310
随机裁剪	0.8356
随机旋转	0.8087

表 4: 不同数据增广方式下训练的结果

5.1.2 模型结构实验

由于不同的任务对模型表达能力的要求不同，所以我们需要尝试不同的模型结构来比较性能。不同模型的实验结果如表 3所示，其中 resnet50 U-Net 是指将 U-Net 的特征提取器改为 resnet50。从实验结果可以看出，参数过多的网络，比如 resnet50 U-Net，反而容易过拟合导致性能下降，同时，深度减少也会使性能降低。此外，加入 attention 结构和 R2U-Net 结构并没有有效提升网络性能。虽然 U-Net 32c 的准确率比 U-Net 64c 的准确率略低一些，但其网络参数少了很多，且训练速度也快了不少，因此我们决定采用 U-Net 32c 结构。

5.1.3 数据增广实验

在多个 epoch 训练时，数据增广操作通过随机改变输入数据来起到增广的效果，不同的数据增广方式会有不同的性能。我们在数据处理流程部分已经分析了一些数据增广方式的缺点，下面我们通过实验进一步验证。

如表 4所示，随机旋转的实验性能较差，印证了我们的假设。此外，随机缩放也表现出较差的性能，可能是由于细胞的大小尺寸只有大致相同的几类。在多种数据增广方式中，随机裁剪的增广方式提升性能最大，这可能是由于在预测时采用 crop tile 方式，重叠部分的结果是多次预测取平均，起到了一定的模型集成作用。

在采取随机裁剪的增广方式时，裁剪的尺寸决定输入网络的图片大小，以及预测 crop tile 时重叠区域的大小，所以也会对结果产生一定的影响。如表 5所示，当裁剪尺寸为 500 时，模型训练结果最佳。

随机裁剪尺寸	jaccard 分数
550	0.8356
500	0.8443
400	0.8343

表 5: 不同裁剪尺寸下训练的结果

损失函数组合方式	jaccard 分数
交叉熵	0.8443
dice loss	0.8130
交叉熵 +dice loss	0.8441
交叉熵 +10×dice loss	0.8440
Lovasz-Softmax loss	0.8443

表 6: 不同损失函数下训练的结果

5.1.4 损失函数实验

在文献调研部分我们已经指出，损失函数 binary cross-entropy 与评价指标 IoU 的优化目标并不完全相同，因此我们尝试了多种损伤函数的组合和不同的权重，如表 6。可以看出损失函数的改变并没有带来性能提升，但与我们的分析一致的是，单独使用 dice loss 来训练时，训练过程不稳定，结果也较差。

5.1.5 后处理参数实验

后处理过程主要有两个参数，一是距离阈值，即得到距离极值点后用这个阈值筛选掉一部分极值点；二是使用膨胀操作将距离较近的极值点连接起来时，所使用膨胀算子的大小。由于不同后处理参数的性能与验证集特性相关性较强，所以我們是在 cross validation 的基础上进行实验，即将训练数据分成 5 部分，取每一部分作为验证集来训练出 5 个模型，用这 5 个验证集的 jaccard 平均值作为平均标准，结果如表 7所示，参数取 13，8 时性能较优。

5.1.6 模型集成

为了减少过拟合对性能的影响，在提交结果时采用模型集成的方法，即将训练数据分成 5 部分，取每一部分作为验证集来训练出 5 个模型，每张图片取这 5 个模型预测结果的 sigmoid 平均值，再用 0.5 阈值二值化，在网站上提交的结果达到 0.840388。

距离阈值	膨胀算子大小	平均 jaccard 分数
10	8	0.8198
13	8	0.8236
15	8	0.8210
13	6	0.8215

表 7: 不同后处理参数下训练的结果

边界生成方式	网站 jaccard 分数
边界宽度相等	0.8210
边界宽度不等	0.825514

表 8: 不同边界生成方式下训练的结果

边界预测方式	网站 jaccard 分数
U-Net 输出维度增加 1	0.825514
使用两个 U-Net 网络	0.7443

表 9: 不同边界预测方式下训练的结果

5.1.7 总结

经过实验，最佳的参数为：训练 200 个 epoch；模型使用 U-Net 32c；采用 500×500 大小的随机裁剪，预测时使用 crop tile；损失函数使用交叉熵；后处理使用距离阈值 13，膨胀算子大小为 8；使用模型集成。

5.2 dataset2 的实验

在进行 dataset2 的实验时，默认 dataset1 的实验结果仍然适用，由于时间关系不再重复进行实验。

5.2.1 边界生成方式

由于 dataset2 任务的语义分割模型除了生成前景的预测，还要生成边界的预测，因此为了模型训练，需要从前景 ground truth 中生成边界 ground truth。这有两种方式，一是使所有细胞的边界宽度相等，二是使细胞边界的宽度与细胞面积的开方成正比。结果如表 8，边界宽度不等时模型性能较好。

5.2.2 边界预测方式

边界预测有两种方式，一是将 U-Net 的输出维度从 1 改为 2，二是使用单独的 U-Net 网络训练边界预测模型。两种方式的模型性能如表 9

5.2.3 损失函数

在模型训练过程中，前景预测的损失函数和边界预测的损失函数是分开计算的，为了在学习过程中更关注边界预测性能，可以将两个损失函数结果加权后进行梯度后传。两种不同损失函数训练的模型性能如表 10，可以看出强调边界预测时模型性能较优。

5.2.4 总结

经过实验，最佳的参数为：生成的边界宽度与细胞大小有关；使用同一个 U-Net 预测边界；对边界预测的损失函数加权后梯度后传。在手动标注数据集上训练，提交的最高分数为 0.830856；

损失函数	网站 jaccard 分数
前景预测损失 + 边界预测损失 1	0.825514
前景预测损失 +10× 边界预测损失	0.830856

表 10: 不同损失函数下训练的结果

在去除测试集的 DIC-Hela 数据集上训练，提交的最高分数为 0.87。在 DIC-Hela 数据集上训练时，为防止过拟合，epoch 减少为 120

6 总结

这次大作业中，我们的主要思路是语义分割加后处理，使用深度学习的语义分割模型预测出输入图片的前景和后景，再使用较为精细的后处理将语义分割结果实例化。为了扩充数据集，我们与另外 7 位同学合作手工标注了训练集。此外，为了解决 dataset2 中细胞粘连程度严重的问题，我们额外增加了对细胞边界的预测，使用前景预测和边界预测共同进行后处理，得到实例化结果。

在这篇报告中，我们首先陈述了文献调研过程，主要调研了深度学习中的实例分割问题和语义分割问题，涉及到网络结构、数据增广和模型平衡等。同时也调研了后处理中用到的传统算法，即分水岭算法和数学形态学。接下来，我们详述了数据处理流程，包括数据集分割、不同数据增广操作的优缺点和数据标准化方法。我们在算法原理部分中，解释了语义分割算法的输入输出以及具体过程，同时论证了其合理性。紧接着，我们在实现过程部分中用流程图的方式更为清晰的展示了算法的处理流程。最后，我们展示了 dataset1 和 dataset2 中一些重要实验的结果，并用表格的方式呈现，方便读者进行相互比对。

在这篇大作业的完成过程中，我们对机器学习有了更深入的理解，同时提高了工程实践的能力。学会通过观察结果发现，并解决问题，是我们最大的收获。虽然在 dataset1 上做了比较多的实验，但遗憾的是，由于时间原因，模型结构探索的实验还是不充分，dataset2 的实验也略少一些，打算在网站重新开放后继续做一些探索。针对 dataset1 我们在网站上提交最高的成绩是 0.840388。dataset2 我们使用手动标注数据集训练的模型得到最高的成绩是 0.830856，使用网上的 DIC-HeLa 数据集训练得到最高的成绩约是 0.87。

7 致谢

为了扩充数据集 2 的数据，我们 8 个人共同手动标注了训练集，标注者分别为：

冯立恒 (2017011954)
 关雄峻 (2017011517)
 郭昊洵 (2017011548)
 牛衍昌 (2017011598)
 吾尔开西 (2017011589)
 吴紫屹 (2017011527)
 殷昊男 (2017011513)

张博睿 (2017011537)

每人标注 20 张总计 160 张。

此外，在完成项目的过程中与冯立恒、关雄峻、吴紫屹同学多次交流思路，受到了很大的启发，在此致谢。

参考文献

- [1] Md Zahangir Alom, Mahmudul Hasan, Chris Yakopcic, Tarek M Taha, and Vijayan K Asari. Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation. *arXiv preprint arXiv:1802.06955*, 2018.
- [2] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. The lovász-softmax loss: a tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4413–4421, 2018.
- [3] Edward R Dougherty. An introduction to morphological image processing. *SPIE*, 1992, 1992.
- [4] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and Jose Garcia-Rodriguez. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint arXiv:1704.06857*, 2017.
- [5] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.
- [6] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, et al. Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*, 2018.
- [7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [8] Leila Shafarenko, Maria Petrou, and Josef Kittler. Automatic watershed segmentation of randomly textured color images. *IEEE transactions on Image Processing*, 6(11):1530–1544, 1997.