

MICCAI 2023 Challenges : STS-基于 3D CBCT 的 牙齿分割任务技术报告

王春实¹, 王一凡¹, 赵彬^{1,2*}

¹ 桂林电子科技大学人工智能学院, 广西桂林, 541004, 中国

² 广西高校人工智能算法工程重点实验室, 广西桂林, 541004, 中国

引言

该技术报告分别对初赛和复赛的开发过程进行了详细介绍, 并分别总结了各自的创新点。总体而言, 该任务具有一定的挑战性, 但通过查阅相关文献以及团队前期积累, 在一定程度上对该任务进行了解决。感谢组委会和相关负责老师给予的机会。

1 初赛

1.1 相关工作

针对半监督语义分割问题, 团队主要调研和研究了 Match 家族的相关工作: MixMatch[1]、ReMixMatch[2]、FixMatch[3]、FreeMatch[4]和 UniMatch[5]。这些工作为 Self-Training 类型, 集伪标签和一致性正则化思想, 符合该任务的实现思路。通过复现和研究这些文献, 团队最后选定 UniMatch 作为本次 3D 牙齿分割的 Baseline, 同时借鉴 FreeMatch 中自适应调整置信度阈值的思想优化模型在分割任务中参数学习的过程。

1.2 开发过程

1.2.1 改进 UniMatch 中的特征扰动

(本节代码位于: [model/my_smp/base/model.py](#) 第 37 行)

半监督学习(Semi-supervised Learning, SSL)领域有诸多相关工作都是基于 Mean-Teacher (MT)模型, 其本质是通过指数移动平均(Exponential Moving Average, EMA)来获得模型之间的差异, 从而实现同一张图像的略微不同的预测结果, 提高模型学习的稳定性。在 UniMatch 中并没有使用 MT 模型, 通过思考 MT 模型的学习过程, 团队提出以下观点: 对网络权重的 EMA 实际上也是扰动的一种。

上述的想法可以这样理解: 在 UniMatch 模型中, 其提出的在 Bottleneck 部分的特征扰动 \mathcal{P} 与 MT 模型的效果其实是类似的, 其本质都是通过加入不同的扰动来实现不同的输出。只不过 MT 模型使用的是整个网络权重下的「扰动」, 而 UniMatch 的特征扰动只在 Bottleneck 部分的高维特征进行 `nn.Dropout(0.5)`。

UniMatch 提出的目的就是获得更加广泛的扰动空间，扰动的本质目的是改变数据的分布和其表示的信息的丰富程度，EMA 的作用范围是网络的权重，在前向传播的过程中对同一个数据的改变较为明显，所以显然其等效扰动更为剧烈。

上述分析可以体现在团队训练 UniMatch 过程中观察到的现象：计算 Pre 和 p^{fp} 损失项 L_{fp} 的时候，其在训练后期容易变为 0。这说明普通的 `nn.Dropout(0.5)` 在牙齿分割任务中的扰动强度受限，需要进一步加强扰动，以使模型进行更好的一致性学习。

基于以上分析，团队首先提出使用以下两种方法改进 UniMatch 中的特征扰动：

- 1) 增强 `nn.Dropout()` 的丢失概率，如设为 `nn.Dropout(0.75)`。
- 2) 更换 `nn.Dropout(0.5)` 为其他的特征扰动方法，如随机噪声，或者其他类型的 Dropout 方法。

由于 UniMatch 中的 Bottleneck 部分存储了大量的多维语义特征，所以丢失太多的信息会造成模型精度的下降，团队在本竞赛中采用第二种方法，即更换 `nn.Dropout(0.5)` 为其他类型的 Dropout（在网络训练过程中添加随机噪声会严重影响训练性能，故放弃）。具体而言，在代码实施过程中，将 `nn.Dropout(0.5)` 更换成 `nn.AlphaDropout(0.5)`。Alpha Dropout [6] 是一种保持自归一化属性的 Dropout，对于均值和单位标准差为零的输入，Alpha Dropout 的输出保持输入的原始均值和标准差，即相比于传统 Dropout 只将某像素置 0，Alpha Dropout 还在此基础上加入了归一化操作，这使得数据在训练过程中更加平滑，不容易出现信号跳变，且能够对一个 Batch 中所有在 Bottleneck 处的值进行修改，从而获得更强的特征扰动。

1.2.2 改进 UniMatch 的固定阈值

(本节代码位于：`util/thresh_helper.py`，`unimatch_fourier.py` 第 145 行)

阈值在 SSL 训练过程中非常重要，决定了网络的自信程度，阈值过高或过低都会影响模型参数的学习。UniMatch 采用固定阈值输出预测结果，必然导致伪注释的错误标记在训练过程中的累积，影响分割精度。为此，团队将 FreeMatch 中的自适应调整置信度阈值的思想引入到当前的分割任务中，使得 UniMatch 在训练过程中自适应调节阈值，提高分割精度[15]。

1.2.3 UniMatch 的 Encoder-Decoder 选择

基于当前任务，团队对 UniMatch 中的编码器(Encoder)和解码器(Decoder)进行了实验探讨。

对于 Encoder，模型的大小并不与分割性能成正相关。通过横向对比在同一种 Decoder 下，不同 Encoder 的表现，ResNeSt-14d[7]比 ResNeSt-50d 分割效果更好；ResNeSt-14d（参数量 8M）和 ResNet-18（参数量 11M）相比，前者更好。

对于 Decoder，DeepLabV3+[13]能够获得更好的边缘信息，但对语义信息的解码能力稍弱；UNet[14]网络较为均衡，能够发挥较好的语义信息解码能力，但分割的边缘较差。从分割指标上来看：使用 DeepLabV3+的网络能够获得更好的豪斯多夫距离(Hausdorff Distance, HD)，而 UNet 则能够提供更好的 IoU 和 Dice 指标。由于本任务评价指标中 IoU 和 Dice 的占比较大，所以团队最终选择 UNet 作为 Decoder。而 UNet++[12]虽然能够提供更好的语义信息解码能力，但其模型参数量太大，训练成本较高，在任务中并未采用。

通过上述的一些实验探讨，选定 UniMatch 中的 Encoder 和 Decoder 分别为 ResNeSt-14d 和 UNet。

1.2.4 提出新的即插即用注意力模块 SCA

(本节代码位于：[model/my_smp/encoders/resnesc.py](#))

根据 1.2.5 的实验探讨，UNet 作为解码器时，获得的 HD 指标相比于其他两个指标表现较差，为了解码器能够更好地解码空间位置信息，得到更好的 HD，团队提出一种新的即插即用(plug-and-play)注意力模块 Split Coordinate Attention(SCA)，并将其嵌入到 1.2.5 中选择的 Encoder 的 ResNeSt-14d 中获得 SCANet，以使编码器保留较强的空间信息。SCA 的结构如图 1(b)所示。

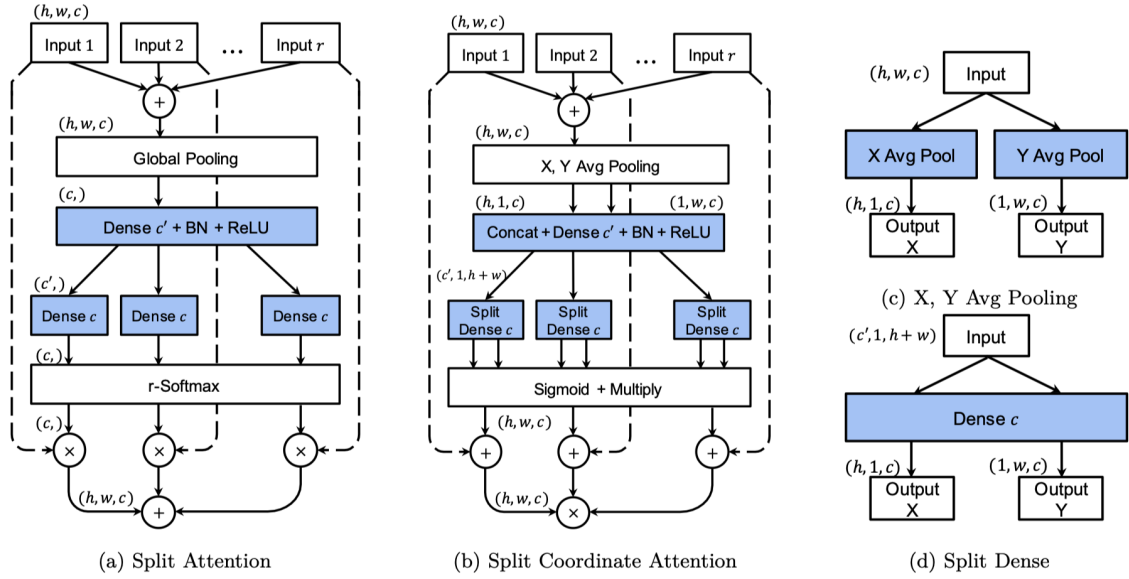


图 1 SCA 与 SA 的结构比较.(a) SA (b) SCA (c) X,Y 平均池化模块 (d) Split Dense 模块.

为了更好地说明 SCA 的提出过程，在此先引入注意力 Split Attention (SA)[7]，其结构如图 1(a)所示。SA 是 ResNeSt 中提出的一种新型注意力，它借鉴了 GoogleNet[8]、ResNeXt[9]和 SE-Net[10]等工作中的思想，融合 Feature-map Attention、Multi Path 和 Channel Attention 为一体，获得了较为不错的性能。

通过观察上述结构，发现实际上 ResNeSt[7]中的 Split Attention 还存在改进空间：Split Attention 提出的前提是标准卷积对通道关系很难进行建模，所以其使用了跨空间维度的全局平均池化来收集具有嵌入式通道统计信息的上下文信息。但是全局平均池化显然会破坏网络的空间语义特征，直接将空间维度挤压为 1×1 ，牺牲了一部分的位置信息，而位置信息在语义分割中捕获空间结构至关重要。

基于 SA 的不足，团队考虑将特征图平面按照不同的维度方向分别进行建模，即使用 X、Y 方向的平均池化代替原来的池化操作，并分裂成不同的前馈流，从而实现语义信息的跨维度提取[11]，最后在输出前进行矩阵乘法操作，形成一个与原始输入尺寸相同的注意力 Map。这样改进即能够有效地对跨通道信息进行建模又能够最大程度地保留空间信息。

接下来分小节对 SCA 进行详细说明。

1.2.4.1 定义 Featuremap Group

与 ResNeXt Block 和 ResNeSt Block 类似，同样引入两个超参数 K 和 R ，分别控制特征图组的数量和基数组 (Cardinal Group) 内的分裂数。因此特征组的总数为 $G = KR$ 。接着对每个单独的特征组应用一系列变换 $\{F_1, F_2, \dots, F_G\}$ ，其中每

个组的中间表示为 $U_i = F_i(X)$ ，其中 $i \in \{1, 2, \dots, G\}$ 。

1.2.1.2 Split Attention 中的问题

第 k 个基数组的表示为： $\hat{U}^k = \sum_{j=R(k-1)+1}^{Rk} U_j$ ，其中 $\hat{U}^k \in \mathbb{R}^{H \times W \times C/K}$ ， $k \in$

$\{1, 2, \dots, K\}$ 。那么 Split Attention 中的第 c 个分量计算如下：

$$s_c^k = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W \hat{U}_c^k(i, j). \quad (1)$$

但全局平均池化将全局空间信息压缩到通道描述符中，因此难以保留位置信息，需要对其进行改进。

1.2.1.3 Split Coordinate Attention

为了鼓励注意力块使用精确的位置信息在空间上捕捉远程交互，优化上述位置信息保留这一问题，团队分解了方程式中制定的全局池化，使用两个空间范围内的池化 Kernels，分别为 $(H, 1)$ 和 $(1, W)$ ，即分别沿水平和垂直坐标进行编码，因此 h, w 方向的第 c 个分量可以分别表示为：

$$s_c^{h,k} = \frac{1}{W} \sum_{0 \leq i < W} \hat{U}_c^k(h, i) \quad (2)$$

$$s_c^{w,k} = \frac{1}{H} \sum_{0 \leq j < H} \hat{U}_c^k(j, w) \quad (3)$$

接下来对上述分量进行合并，首先对其在通道为度进行 Concat 操作，再经过一次卷积和非线性层，可以得到：

$$s_c^k = \mathcal{G}^c([s_c^{h,k}, s_c^{w,k}]). \quad (4)$$

接下来对 s_c^k 进行拆分操作，记为 $t_c^{h,k}$ 和 $t_c^{w,k}$ ，得到：

$$\begin{cases} u_c^{h,k} = \sigma(\mathcal{S}_c^h(t_c^{h,k})) \\ u_c^{w,k} = \sigma(\mathcal{S}_c^w(t_c^{w,k})) \end{cases} \quad (5)$$

其中 σ 表示 Sigmoid。那么接下来需要对基数组的注意力进行聚合操作，那么第 c 个通道的特征图计算公式为：

$$V_c^k = \hat{U}_c^k \sum_{i=1}^R u_c^{h,k} \times u_c^{w,k} \quad (6)$$

1.2.1.4 SCANet

将提出的 SCA 模块嵌入到 ResNeSt-14d 中获得 SCANet，且参数量未发生变化。

1.2.5 模型总参数量

除了使用参数量最小版本的 SCANet 之外，团队还进一步对 UNet 解码器的参数量进行了限制，以更好地进行模型参数学习，即 UNet 每个 Stage 输出的 Channel 数量由原来的 256, 128, 64, 32, 16 分别修改为 64, 32, 16, 8, 4。整个模型总参数量为：**11.1M**

1.2.6 数据预处理

为了在后续模型学习中利用无标签数据生成的像素级伪注释，对数据进行切片处理。为了获得丰富的牙齿形态，对数据进行 3 轴切片，即将 3D 的牙齿数据按轴向面、冠状面和矢状面切片，例如一个 $640 \times 640 \times 400$ 的 CT 图像，按照三个轴切片，则可以得到 $640 + 640 + 400 = 1680$ 张切片图像。使用文件后缀作为切片轴的划分，为了简单起见，直接以 $_x$, $_y$ 和 $_z$ 作为切片文件的后缀 [16]。

此外，将切片统一进行归一化处理，图像数值映射到 $[0,1]$ 区间 [5]。由于切片的数量较多，将训练集(Train)切片中的 10% 作为验证集(val)。

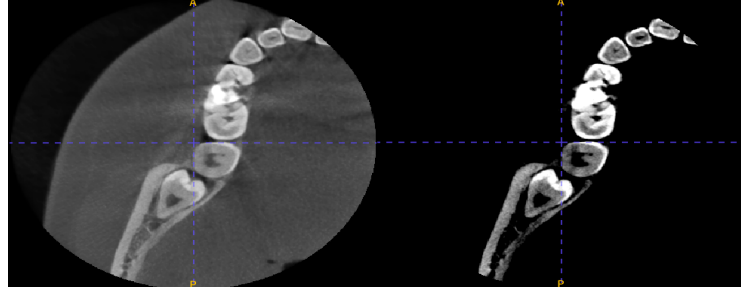
为了探索输入图像不同阈值对网络模型性能的影响，团队进行了阈值实验，实验结果如表 1 所示。

表 1 不同阈值的输入图像在验证集上的分割结果

Bottom	Top	Val1	Val2	Val3
0	1500	0.7738	0.8179	0.8621
500	1500	0.8167	0.8258	0.8762
500	1800	0.8386	0.8036	0.8849
550	1950	0.7253	0.8503	0.8732
200	2000	0.7577	0.883	\
500	2000	0.8162	0.848	0.8844
750	2000	0.8269	0.8321	\
300	2000	0.8131	0.854	0.8796
450	2050	0.829	0.8201	0.8671
400	2100	0.747	0.8128	0.8454
0	2500	0.8247	0.8301	0.8603
500	2500	0.8597	0.8602	0.8783

从表 1 可以看到，当阈值选择 500 和 2000 这个组合时模型在验证集上的分割结果总体是最好的，并能够获得更快的收敛速度，故在实验中选择此组合作为阈值。假设图像 x 为 CT 读取后的原始数据图像，其取值范围为 $[-1000, 3095]$ ，则 x 的每个像素点 $x_{x,y,z}$ 处理过程为：

$$x_{x,y,z} = \begin{cases} 500, & x_{x,y,z} \leq 500 \\ 2000, & x_{x,y,z} \geq 2000 \\ x_{x,y,z}, & \text{Others} \end{cases}$$



(a) 原始图像切片

(b) 阈值处理后图像切片

图 2 原始输入图像切片与进行阈值[500,2000]处理后的图像切片

如图 2 所示，将原始 CT 图像按照 500 和 2000 进行阈值处理后，可以显著增强目标信息，有利于模型更好地对语义信息进行学习，提高分割精度。

1.2.7 模型训练与预测

（预测代码在 [infer.py](#) 中）

将 1.2.6 处理后的切片数据送入改进后的 Improved-UniMatch 进行训练，并按照不同的轴分别训练 3 次，得到 3 个模型，分别定义 $f_x(w_x; x)$, $f_y(w_y; y)$, $f_z(w_z; z)$ ，其中 w_x , w_y , 和 w_z 分别为 3 个模型对应的参数； x , y , 和 z 分别为 3 个模型对应的输入。

对于一个输入图像的 3 个不同面的切片集 I_x, I_y, I_z ，分别得到 3 个预测结果记为 $p_x = f_x(w_x; I_x)$, $p_y = f_y(w_y; I_y)$, $p_z = f_z(w_z; I_z)$ ，接着使用 argmax 获得最终的预测结果：

$$Pre = \text{argmax}(p_x + p_y + p_z) \quad (7)$$

1.3 训练技巧

1.3.1 训练超参数设置

模型训练的优化器使用 AdamW，学习率 lr 初始值选择为 0.0001， weight_decay 为 0.0001。

学习率使用如下的公式进行更新：

$$lr = lr_b \times (1 - \frac{i}{N})^{0.9} \quad (8)$$

其中 lr_b 指的是基础学习率，此处为 0.0001； i 指的是当前迭代次数， N 指的是总迭代次数。

在使用阈值自适应方法的时候，momentum 参数设置为 0.9998 效果较好。如果动量过小，可能会造成阈值为 1.00，严重影响模型的收敛。

1.3.2 数据增强

团队使用 UniMatch 中自带的数据增强范式，但进行了一些改动，以更好的适应本任务。具体所使用的数据增强、参数以及应用增强的概率展示于表 2。

表 2 数据增强信息表

增强类型	参数	应用概率
ColorJitter	0.75, 0.75, 0.75, 0.35	0.8
CutMix	\	0.5
GaussianBlur	$\sigma \in (0.1, 0.5)$	0.5
RandomFlip	\	0.75
RandomRotate	\	0.5

1.3.3 Loss 函数设置

Loss 函数团队采用的是普通的 CrossEntropy Loss 和 Dice Loss，对于不同的损失项计算，采用了不同的 Loss 计算方式。

具体而言，每个未标记图像 x_u 同时受到两个算子的扰动，即弱扰动 \mathcal{A}^w （例如裁剪）和强扰动 \mathcal{A}^s （例如颜色抖动）。然后，整体目标函数是监督损失 \mathcal{L}_s 和无监督损失 \mathcal{L}_u 的组合：

$$\mathcal{L} = \frac{1}{2}(\mathcal{L}_s + \mathcal{L}_u) \quad (9)$$

通常，监督项 \mathcal{L}_s 是模型预测和真实标签之间的 CrossEntropy 和 Dice 损失。而无监督损失 \mathcal{L}_u 正则化了样本在强扰动下的预测与弱扰动下的预测相同，可以表示为：

$$\mathcal{L}_u = \frac{1}{B_u} \sum \mathbb{1}(\max(p^w) \geq \tau) H(p^w, p^s), \quad (10)$$

其中 B_u 是未标记数据的批量大小， τ 是过滤噪声标签的预定义置信度阈值。 H 最小化两个概率分布之间的熵：

$$p^w = \hat{F}(\mathcal{A}^w(x^u)); \quad p^s = F(\mathcal{A}^s(\mathcal{A}^w(x^u))) \quad (11)$$

为了简单起见，我们按照 FixMatch 将 \hat{F} 设置为与 F 完全相同。记特征扰动预测为 p^{fp} ，那么无监督损失 \mathcal{L}_u 为：

$$\mathcal{L}_u = \frac{1}{B_u} \sum \mathbb{1}(\max(p^w) \geq \tau) (H(p^w, p^s) + H(p^w, p^{fp})) \quad (12)$$

由于存在双流扰动，采用了对比学习的思想，与 FixMatch 相比，维护了两

个辅助前馈流，那么最终的无监督损失 \mathcal{L}_u 为：

$$\mathcal{L}_u = \frac{1}{B_u} \sum \mathbb{1}(\max(p^w) \geq \tau) (\lambda H(p^w, p^{fp}) + \frac{\mu}{2} (H(p^w, p^{s_1}) + H(p^w, p^{s_2}))) \quad (13)$$

1.4 初赛创新点汇总

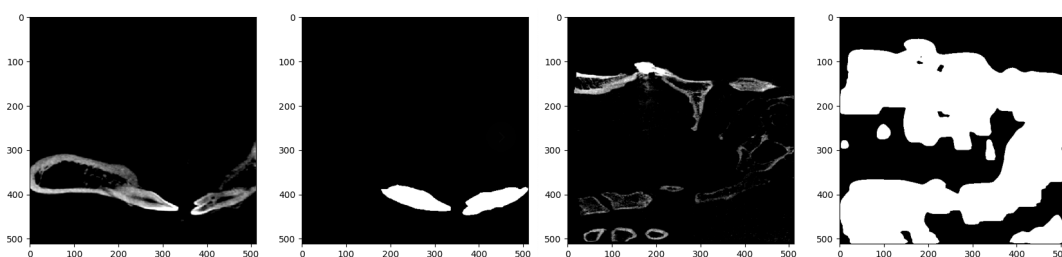
- 1) 改进了 UniMatch 中的特征扰动，使得模型获得更加广泛的扰动效果。
- 2) 改进了 UniMatch 中生成伪标签时的阈值，使其可以自适应调整，显著改善模型分割边缘的性能。
- 3) 提出了一种新的即插即用的注意力模块 SCA，嵌入到编码器中获得 SCANet，能够在不增加参数量的情况下改善分割的豪斯多夫距离指标，获得更好的分割边缘。
- 4) 所提出的牙齿分割算法实施简单，训练成本低，模型大小仅 11.1M。
- 5) 探讨了阈值对于分割性能的影响，并选取了最为合适的阈值进行数据预处理。

2 复赛

2.1 难点分析

2.1.1 域偏移问题

团队在复赛开始之后第一时间尝试使用初赛模型进行了半监督训练，在训练结束后，观察其在测试集(Test)上的预测结果，发现效果极差，如图 3 所示，推测有标签的数据和无标签的数据存在严重的域偏移 (Domain shift) 问题。



(a)验证集图像切片 (b) 验证集图像切片分割结果 (c) 测试集图像切片 (d) 测试集图像切片分割结果

图 3 验证集与测试集分割结果

为了较好地解决域偏移问题，团队对 Labeled 集、Unlabeled 集和 Test 集进行了数据统计分析。统计结果显示，Labeled 集合上的图像分辨率明显小于 Unlabeled 集和 Test 集，并且 CT 图像的像素强度范围在 Labeled 集合上范围为 $[-1000, 3095]$ ，而其他两个集合的像素强度范围则在 -1000 到 10000 多不等，这

导致 Labeled 集很难引导 Unlabeled 集进行半监督学习，进一步阻碍了模型在 Test 集上的泛化性能。

为了尽可能消除域偏移带来的影响，团队设计了一种 2-Stage 算法，通过两个 Stage 使得模型能够学到目标域的特征，并使用 Fourier Transform Augment (FTA) 模块使模型学到两个域之间的差异。

2.1.2 伪影问题

在 CT 图像中会存在各种伪影，而伪影对分割模型的影响是巨大的，如图 4 所示。

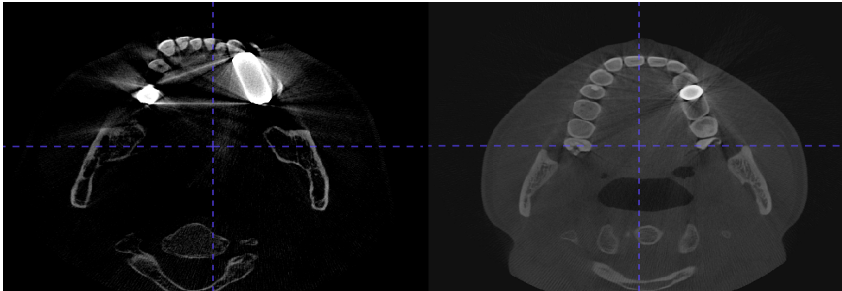


图 4 图像切片伪影实例

在半监督学习过程中，牙齿的伪影会降低伪标签的质量，最终影响模型的分割准确率。在后续生成伪注释的训练阶段 1 中，相较于 3D nnUNet，采用 2D nnUNet 生成的伪注释较为准确，受伪影的影响较小一些。

2.2 开发过程

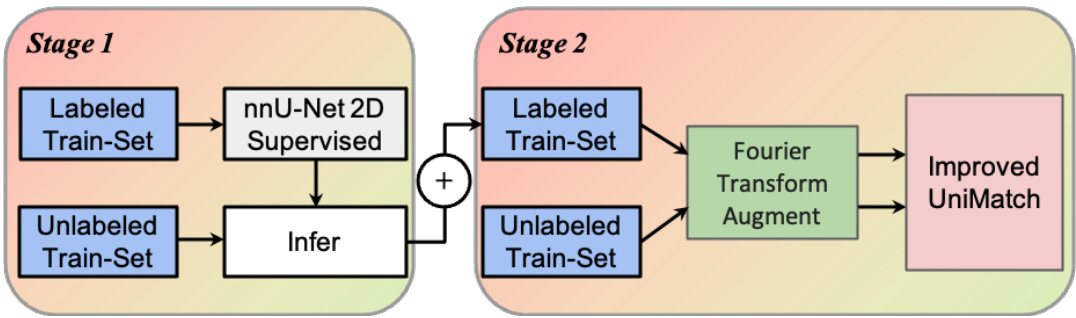


图 5 复赛 3D 牙齿分割总体流程图

基于初赛提出的 Improved-UniMatch 模型，复赛中，团队采用多阶段训练策略减少 Labeled 集和 Unlabeled 集的域偏移问题，以提高模型在 Test 集上的泛化性能。如图 5 为复赛 3D 牙齿分割的总体流程图，在第一阶段使用 2D nnU-Net 模型进行监督学习，为训练集中随机采样的少量 Unlabeled 样本生成像素级伪注释，然后将这些伪注释样本和训练集中的有标记样本合并成训练集中的有标记样本，协同训练集中的无标记样本进行域适应后送入 Improved-UniMatch 模型进

行参数学习。

接下来具体介绍相关阶段的训练过程。

2.2.1 第一阶段

在 Stage1 的训练过程中，利用 2D nnU-Net 作为分割网络。首先采用有标签的数据，训练 2D nnU-Net 20 个 Epochs，然后用训练好的 2D nnU-Net 网络为 Unlabeled 集中随机选取的 10 个无标签的数据生成低质量的像素级伪标签。低质量伪标签能定位到牙齿的位置，但会分割出较多无关的区域，如上颌的骨头、CT 的底座等，如图 6 所示。

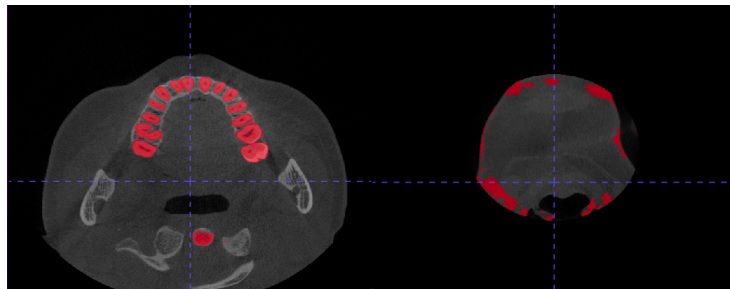


图 6 低质量的像素级伪注释

然后将这 10 张伪注释的样本与 Labeled 集进行合并，共同构成 Stage 2 训练集中有标记的样本集。

2.2.2 第二阶段

(本节代码位于：[dataset/micca3d.py](#) 第 198 行)

Stage 2 的训练建立在第一阶段处理过后的数据的基础上。首先按照初赛相同的标准划分好数据集 (Train、Val)，再经过 FTA 模块进行数据增强，使得模型能够学到目标域上的信息并使用初赛中提出的 Improved-UniMatch 模型进行训练。

2.2.2.1 FTA 模块

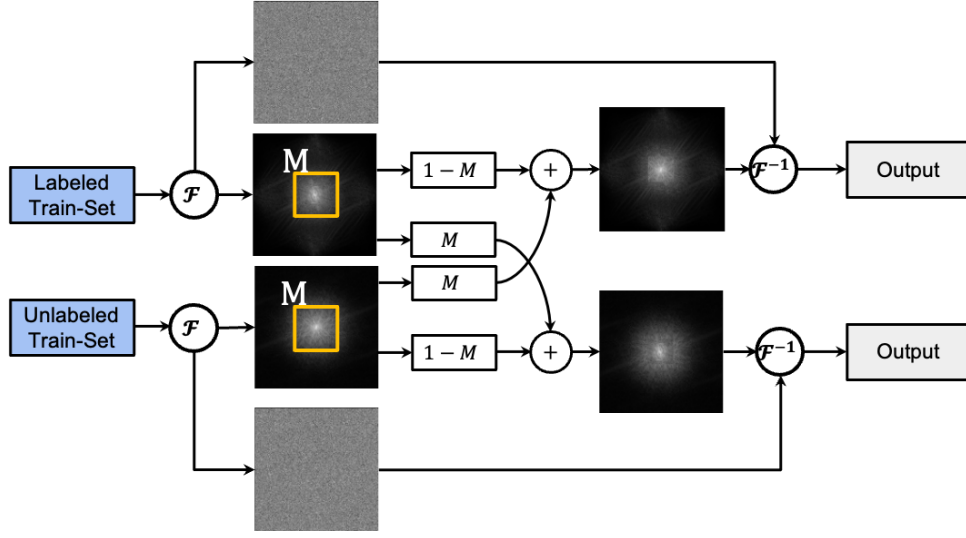


图 7 FTA 流程图

在训练过程中，随机抽取 1 张 Labeled 图像 x^w 和 1 张 Unlabeled 图像 x^u ，然后进行傅里叶变换 \mathcal{F} 将图像转移到频域并获得幅度谱 A^w, A^u 和相位图像 P^w, P^u ，其中幅度谱包含低级统计量，而相位图像包括原始信号的高级语义[17]。

然后，通过合并图像 x^u 的幅度信息来增强图像 x^w ：

$$A_{\text{new}}^w = (1 - \lambda)A^w * (1 - M) + \lambda A^u * M \quad (9)$$

其中 A_{new}^w 是新生成的相位图； λ 为调整 x^w 与 x^u 相位信息比例的参数； M 用于控制要交换的幅度谱的空间范围，此处将 M 设置为包含低频信息的幅度谱的中心区域。之后，将合并样本通过 \mathcal{F}^{-1} 从频域变换到图像域，得到经过傅里叶变换增强并融合了另一个样本的低层信息的图像样本 Z^w ：

$$Z^w = \mathcal{F}^{-1}(A_{\text{new}}^w, P^w) \quad (10)$$

同理交换 x^w 和 x^u ，可以得到 Z^u ：

$$Z^u = \mathcal{F}^{-1}(A_{\text{new}}^u, P^u) \quad (11)$$

值得一提的是 Z^w 和 Z^u 可以合并到一起进行计算，这将大大减少运算开销，即在同一次增强中需要同时对 x^u 和 x^w 进行增强，并将二者相互增强后的结果送入后续训练流程。

2.3 复赛创新点汇总

- 1) 提出了一种多阶段训练框架，能够减少域偏移带来的性能下降问题。
- 2) 使用 2D nnU-Net 生成低质量像素级伪注释，提升模型在目标域上的效果。

3) 采用了一种 FTA 数据增强模块，实现双流数据增强，提高模型在测试集上的泛化性能。

2.4 复赛结果

2.4.1 定量结果

表 3 模型在验证集和测试集上的量化结果				
数据集	Dice	IoU	HD	Score
验证集	0.8786	0.8413	0.0648	0.8844
测试集	0.8058	0.8376	0.1599	0.8256

2.4.2 定性结果

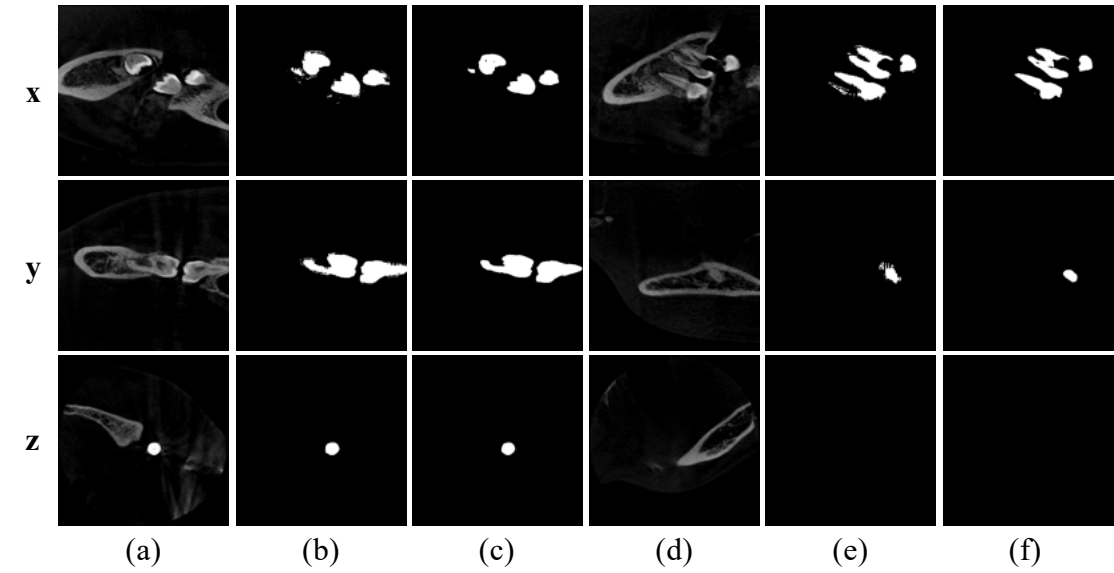


图 8 验证集源域数据分割结果示例

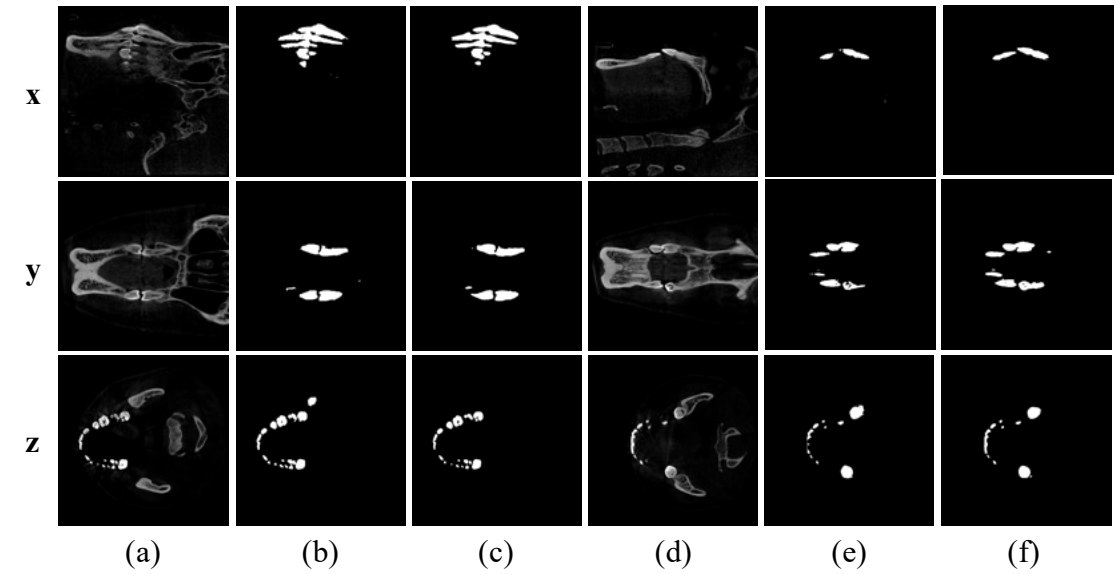


图 8 验证集目标域数据分割结果示例 (Label 为 nnUNet 生成)

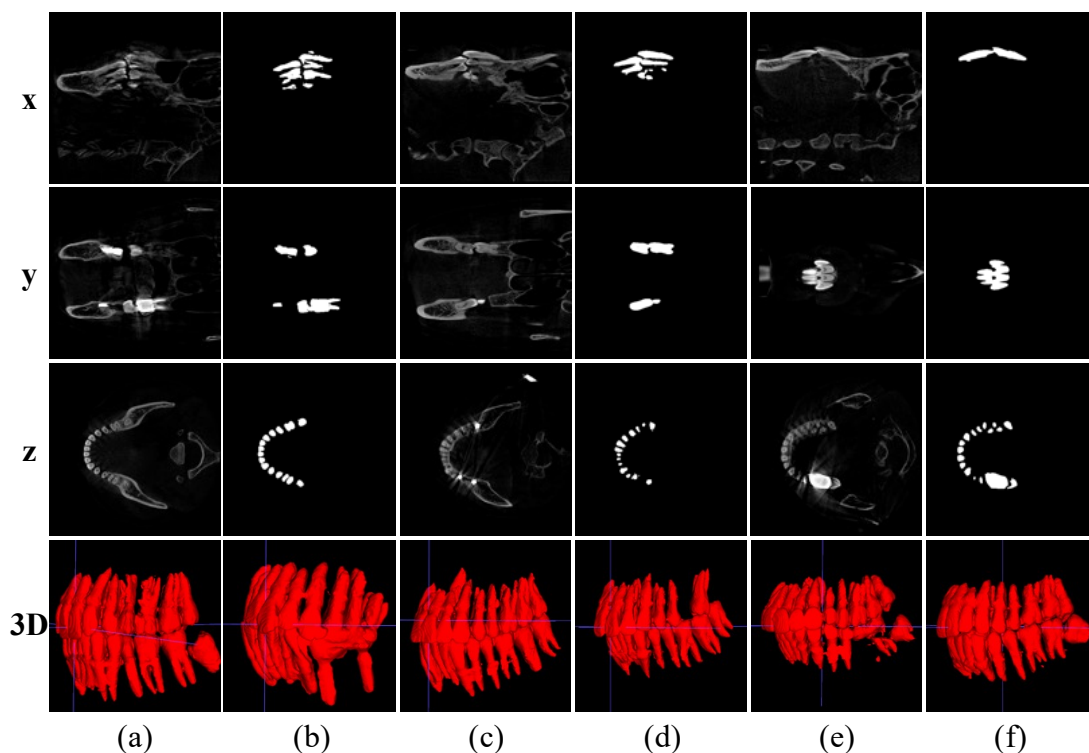


图 9 测试集目标域数据分割结果示例

参考文献

- [1] Berthelot D, Carlini N, Goodfellow I, et al. Mixmatch: A holistic approach to semi-supervised learning[J]. Advances in neural information processing systems, 2019, 32.
- [2] Berthelot D, Carlini N, Cubuk E D, et al. Remixmatch: Semi-supervised learning with distribution alignment and augmentation anchoring[J]. arXiv preprint arXiv:1911.09785, 2019.
- [3] Sohn K, Berthelot D, Carlini N, et al. Fixmatch: Simplifying semi-supervised learning with consistency and confidence[J]. Advances in neural information processing systems, 2020, 33: 596-608.
- [4] Wang Y, Chen H, Heng Q, et al. Freematch: Self-adaptive thresholding for semi-supervised learning[J]. arXiv preprint arXiv:2205.07246, 2022.
- [5] Yang L, Qi L, Feng L, et al. Revisiting weak-to-strong consistency in semi-supervised semantic segmentation[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 7236-7246.

- [6] Klambauer G, Unterthiner T, Mayr A, et al. Self-normalizing neural networks[J]. Advances in neural information processing systems, 2017, 30.
- [7] Zhang H, Wu C, Zhang Z, et al. Resnest: Split-attention networks[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022: 2736-2746.
- [8] Szegedy C, Liu W, Jia Y, et al. Going deeper with convolutions[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2015: 1-9.
- [9] Xie S, Girshick R, Dollár P, et al. Aggregated residual transformations for deep neural networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 1492-1500.
- [10] Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.
- [11] Hou Q, Zhou D, Feng J. Coordinate attention for efficient mobile network design[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 13713-13722.
- [12] Zhou Z, Siddiquee M M R, Tajbakhsh N, et al. Unet++: Redesigning skip connections to exploit multiscale features in image segmentation[J]. IEEE transactions on medical imaging, 2019, 39(6): 1856-1867.
- [13] Chen L C, Zhu Y, Papandreou G, et al. Encoder-decoder with atrous separable convolution for semantic image segmentation[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 801-818.
- [14] Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation[C]//Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9,
- [15] Sun B, Yang Y, Zhang L, et al. CorrMatch: Label Propagation via Correlation Matching for Semi-Supervised Semantic Segmentation[J]. arXiv preprint arXiv:2306.04300, 2023.
- [16] Liu Z, Cao C, Ding S, et al. Towards clinical diagnosis: Automated stroke lesion segmentation on multi-spectral MR image using convolutional neural network[J]. IEEE Access, 2018, 6: 57006-57016.

- [17] Yao H, Hu X, Li X. Enhancing pseudo label quality for semi-supervised domain-generalized medical image segmentation[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2022, 36(3): 3099-3107.