LightM-UNet：基于Mamba模型的医学图像分割模型轻量化的研究与改进

摘要

UNet及其变体在医学图像分割中应用十分广泛。然而这些模型不少有着大量的计算量和模型参数，这很大程度上限制了在他们在临床医疗的应用。最近基于状态空间模型的Mamba架构在性能上和轻量化上都有着优秀的表现，成为CNN和Transformer架构的强力对手。这项工作使用Mamba替代UNet中的CNN和Transformer，旨在解决医学图像分割中模型轻量化的问题。以此作者引入了LightM-UNet，是一个集成了Mamba和UNet轻量级的框架。具体来说，本项工作使用Mamba残差层来提取输入的深层语义特征以建模长距离空间依赖关系，使得模型具有线性的计算复杂度，此外还取得了优秀的性能。本文在基于复现LightM-UNet的工作上，对其中损失函数设计进行了优化和设计并进行了实验，提升了模型性能，且在轻量化工作上分析了Mamba参数量原因并设计一个基于Mamba的轻量级模块和系统框架。望在模型轻量化的同时保证模型性能。

关键词：医学图像分割；模型轻量化；Mamba；损失函数

1 引言

医学图像分割可以减轻医生工作负担，以及是智能医疗发展的有用工具。但现如今医学图像分割任务仍面临着诸多挑战，首先在医学图像的复杂性和多样性上，这使得分割任务变得困难。其次医学图像往往存在噪声、模糊和伪影等问题，这些因素会影响分割算法的性能。以及获取高质量的标注数据成本高昂且耗时。这些问题统称为性能问题，有很多方法，如域迁移方法 [1] 、扩散模型 [2] 和弱监督方法 [3] ，不断改进模型，性能不断提高。另一个是模型轻量化难题，尤其是3D医学图像的分割，有着模型训练和推理时间长以及临床设备性能限制的痛点，因此如何保证性能也能够保证轻量化是医学图像分割研究中的一个重要课题。

为应对医学图像分割中的轻量化问题也提出了多种解决方案。一个方向是基于卷积的方向。主要包括卷积的U-Net [4] 结构及其变体，因其易于扩展结构和良好的性能而广泛应用于医学图像分割一定程度上降低了参数量。后续大部分研究注重于性能效果，倾向于添加许多新的结构，虽然性能有提升，但损失了轻量化。另一个方向是基于ViT [5] ，自注意力机制引入视觉任务，以学习图像的全局关系。而目前主流方向是基于卷积和Transformer混合分割，弥补卷积网络全局特征提取，缓和了Transformer计算量大的缺点，这种方法在保持了高性能的同时又保证了一定程度的模型轻量化。

最近状态空间模型（SSM）提出。而在经典SSM研究 [6] 的基础上，提出的Mamba [7] 框架不仅可以用来建立远程依赖，而且它随着输入大小有着线性计算复杂度。因此作者引入了一个新的基于Mamba的轻量级分割模型LightM-UNet [8] ，并提出了的RVM（残差视觉Mamba）

层，通过纯Mamba的方式来提出深层特征，LightM-UNet参数和计算量得到大大下降。本人在复现代码的时候，察觉到作者使用的是分割领域常用的BCE和Dice的损失函数，但是由于医学图像的特殊性，这可能不是最好的选择，这里我采用了更为合适的自适应区域特定损失函数，通过实验对比取得了相比于作者更好的效果，此外还对Mamba模型参数来源进行分析，并设计了一个新的轻量化模块SPM（分段平行Mamba），用来替换作者的RVM，理论上能够进一步降低参数和计算量。

2 相关工作

2.1 CNN和Transformer

医学图像分割目前有两种常用的网络架构，分别是CNN和Transformer，二者已经在医学图像分割中占据了主要工作。CNN的工作包括有U-Net和DeepLab [9] 等，它能高效率的分层提取图像特征，同时比一般的全连接网络的参数更低，而计算效率更高。它们共享的权重信息能够使它们擅长于捕获平移不变性和局部特征。后续大部分研究注重于性能效果，倾向于添加许多新的结构，虽然性能够有提升，但损失了轻量化。直到后续提出的nnUNet [10] ，以“去掉许多复杂的网络设计，而将重点放在其他方面”为理念，提出将注意力集中在数据集适配的模型配置上，就能做到显著的模型性能提升。该工作在许多数据集上取得了当时的最好的性能，且网络参数大大减少。

Transformer最初是使用在自然语言处理这种序列任务上的架构，核心是注意力机制的计算，目前已经成功地用于视觉任务，例如用于图像识别的ViT和用于各种视觉任务的SwinTransformer [11] 。它与CNN相比，Transformer本质上不处理图像空间特征，而是将图像视为一系列小块作为序列输入。它具有更好的捕获全局信息的能力。由于这和CNN的互补的特点，目前许多研究探索了通过混合网络架构将Transformer引入CNN中，例如TransUNet [12] 、UNETR [13] 、nnFormer [14] 和SwinUNETR [15] 。尽管Transformer有很强的长距离特征处理能力，但由于自注意力机制的计算随着输入大小的增加而按平方规模增长，特别是对于通常具有高分辨率的医学图像来说依然是一个难题。因此，如何在提升CNN的长距离特征处理能力仍然是一个等待解决的问题。

2.2 基于SSM的Mamba

最近状态空间模型的提出，它以被设计为一种构建深度网络的高效且有效的模块而出现，它在连续长序列数据分析中取得了优秀性能，且有着线性的计算复杂度。Mamba则进一步通过选择性机制改进了SSM，允许模型以输入依赖的方式选择相关信息。通过与硬件感知结合，使得Mamba在密集工作如语言上超越了Transformer。此外SSM在视觉任务也表现优异，如图像和视频分类。而图像块和图像特征也可以被视为序列，使用Mamba块来增强CNN长距离建模能力成为了后续发展方向，随着Mamba2的发布，相关的将其应用在视觉任务的工作提出，如U-Mamba [16] 。其中提出了一个混合的CNN-SSM块，结合了CNN局部特征和Mamba全局特征提取的优点，但是U-Mamba仍引入了不小的参数和计算。

因此作者基于U-Mamba引入了基于Mamba的轻量级U型分割模型LightM-UNet，其中提出的视觉Mamba层（RVM），能以纯Mamba的方式从图像中提取深度语义特征，通过实验验证，LightM-UNet仅有1M的参数，并且超越了现有的最先进的模型，并且相比于U-Mamba其参数和计算量分别降低了116倍和224倍，因此它能在显著降低参数和计算成本的同时，实现了最先进的性能。

3 本文方法

虽然目前医学图像分割包括有2D和3D的任务，但是本文工作主要集中于轻量化，因此主要介绍的是3D这一更具有复杂计算代表性的任务讲解。

LightM-UNet的总体架构如下图所示，总体架构是U形结构是基于nnUNet作为基础模型，其中编码器中包含RVM层，中间是RVM层的结构，主要是一个VSS模块，通过和Mamba-2对比不难发现，VSS模块就是基于Mamba-2改版。首先给定一个输入图像。首先使用DWConv（深度分离卷积层） [17] 对输入图像进行浅层的特征提取得到浅层特征图，随后经过三个连续的编码块，每次经过一个编码块，特征图通道数目加倍，分辨率减半，之后特征图再经过瓶颈层进一步的特征提取，形状大小不变，该结构为三个连续的RVM层，之后经过三个连续的解码层，特征图通道数减半分辨率翻倍，并进行了跳跃连接，最后利用DWConv层将通道数映射到分割目标的数量，归一化之后并应用SoftMax激活函数生成最终图像掩码。

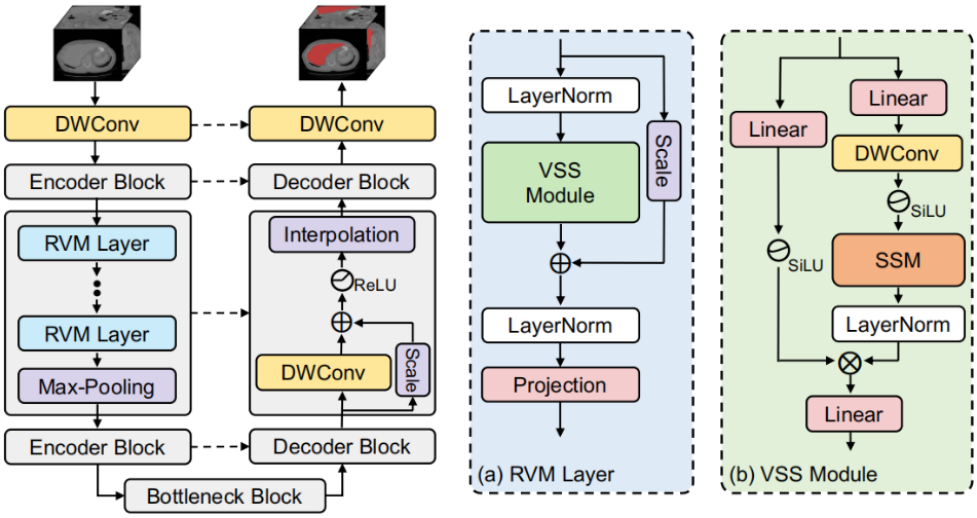


图 1. LightM-UNet框架

3.1 残差视觉Mamba层（Residual Vision Mamba Layer，RVM层）

LightM-UNet提出了RVM层，该模块用来提取图像深度语义特征，具体来说，LightM-UNet利用残差连接和残差调整因子，可以进一步的增强了SSM的远程空间建模能力的优点下，还几乎没有引入新的参数和计算复杂度。如图1 (a)所示，给定输入的深度特征$M_{in}^{l} \in R^{\check{L}\times\check{c}}$，RVM层首先对输入特征使用LayerNorm进行归一化处理，然后使用VSS来捕获空间的长期依赖关系。随后，它在之后再使用一个调整因子$s \in R^{\check{c}}$，它作为一个可以学习的参数，后续能更好的进行后续的残差连接，这个过程可以用公式表示如下：

$$\widetilde{M}^{l} = VSSM\left(LayerNorm\left(M_{in}^{l}\right)\right) + s \tag{1}$$

接下来，RVM层使用另一个LayerNorm对$\widetilde{M}^{l}$进行规范化，然后利用一个投影层将M^{l}转换为更深层次的特征。上述过程可表述为：

$$M_{out}^{l} = Projection\left(LayerNorm\left(\widetilde{M}^{l}\right)\right) \tag{2}$$

3.2 视觉状态空间模块（Vision State-Space Module，VSS模块）

LightM-UNet引入了VSS模块，如图1 (b)所示，用于远程空间建模。VSS模块$W_{in}^{l} \in R^{\check{L}\times\check{c}}$特征图作为输入，并将其引导到两个并行分支中。在第一个分支中，VSS模块首先使用线性层将特

征通道扩展到$\lambda \times \check{C}$，其中λ表示一个预定义的信道扩展因子，一般为2。随后，再进行一个DWConv和SiLU激活函数，之后再然后是SSM和LayerNorm。在第二个分支中，VSS模块使用一个线性层将特征通道扩展到$\lambda \times \check{C}$，然后再通过一个SiLU激活函数。随后，VSS模块两个分支的特征图进行Hadmard乘积，并将通道数恢复为$\check{C}$，以生成与输入W_{in}具有相同形状的输出W_{out}。上述过程可表述为：

$$\begin{aligned} W_1 &= LayerNorm(SSM(SiLU(DWConv(Linear(w_{in}))))) \\ W_2 &= \text{SiLU}\big(Linear(W_{in})\big) \\ W_{out} &= \text{Linear}(W_1 \odot W_2) \end{aligned} \tag{3}$$

其中，$\odot$表示Hadmard乘积。

3.3 瓶颈块（Bottleneck Block，瓶颈块）

与Transformer类似，当网络深度变得过度时 [18] ，Mamba会遇到收敛的问题。所以作者对于LightM-UNet通过合并四个连续的RVM层来网络的瓶颈结构，可以进一步建模空间长期依赖关系。在这些瓶颈区域内，特征通道的数量和分辨率保持不变。

3.4 训练阶段

作者为了评估模型的性能，选择了两个公开可用的医学图像数据集：LiTs数据集 [19] ，包含3D CT图像，以及Montgomery&Shenzhen数据集 [20] ，包含2D X射线图像。数据被随机分为训练集、验证集和测试集，比例为7:1:2。LightM-UNet的三个编码器块中的RVM层数量分别设置为1、2和2。并使用SGD作为优化器，初始学习率为1e-4。使用PolyLRScheduler作为调度器，训练了100轮。损失函数被设计为交叉熵损失和Dice损失的简单组合。作者对于LiTs数据集，图像被归一化并调整大小为$128 \times 128 \times 128$，批量大小为2。对于Montgomery&Shenzhen数据集，图像被归一化并调整大小为512×512，批量大小为12。

4 复现细节

4.1 与已有开源代码对比

本轮在GitHub上发布了源代码，本次复现工作正是基于源代码进行了修改和添加完成的。使用的数据集除了文中提到的LiTs数据集之外，还使用了私有的Lung79数据集，这是一个79个3D的人体肺部CT扫描图像，需要对肺部的动脉、静脉和气道进行分割。此外我修改了原文中所使用的损失函数以及替换了原文的RVM架构。复现流程如下所示：

4.2 实验环境的搭建

本次复现任务所使用的编程语言为Python3.10，基于anaconda虚拟环境，并使用深度学习框架Pytorch。在Windows 10上基于Pycharm进行编程，所有实验都是在两块NVIDIA A100上进行的。

- 在Linux系统上安装anaconda并创建一个python3.10版本的虚拟环境
- 激活虚拟环境并进入测试是否创建成功
- 根据显卡驱动版本安装相应的pytorch并判断是否安装成功
- 进入到项目相应目录并安装Mamba和nnUNet框架
- 安装项目预配置，能够执行项目中测试实例，此时环境搭建成功

4.3 创新点

包括损失函数创新以及模块创新，分别为模型提高性能以及减轻模型参数量和计算量方面的工作，具体细节如下所示：

自适应区域特定损失函数（Adaptive Region-Specific Loss，ARSL）

损失函数通常用于测量网络预测与真实标注之间的差异，并指导网络参数的优化。Dice损失和交叉熵损失是使用最广泛的两种损失函数。但是这两种损失函数都对所有像素一视同仁。但是对于医学图像，有许多像素例如组织对比度低的器官边界附近的像素，通常更难被分割。这些像素通常是限制当前深度学习网络性能的因素。所以需要更多地关注这些像素可能会提高学习效率和自动分割性能。

通常，在Dice损失中，FP（假阳性）和FN（假阴性）误差相同权重。但在多数医学图像上，图像分割中两种误差类型一般存在严重的不平衡。因为分割目标远小于背景时，网络通常更容易出现FN误差而不是FP误差。为解决此问题Dice拓展为Tversky [21] 损失。对不同的误差加以权重。然而，这种损失函数的性能对控制两种类型误差权重超参数非常敏感。为解决此问题，设计在训练过程中逐步调整损失参数，以获得更好的性能。ARSL [22] 采用了类似的策略，通过在网络训练过程中根据局部预测结果自适应调整FP和FN误差之间的权衡，来优化区域特定的损失。

如图2所示，ARSL与计算整个图像体积的预测和真实标注之间重叠的传统Dice损失不同，通过将体积划分为子区域，以分别优化每个子区域的网络预测。这种区域特定的损失允许自动调整每个子区域的权重，以便更多地强调难以实现高预测准确性的子区域。

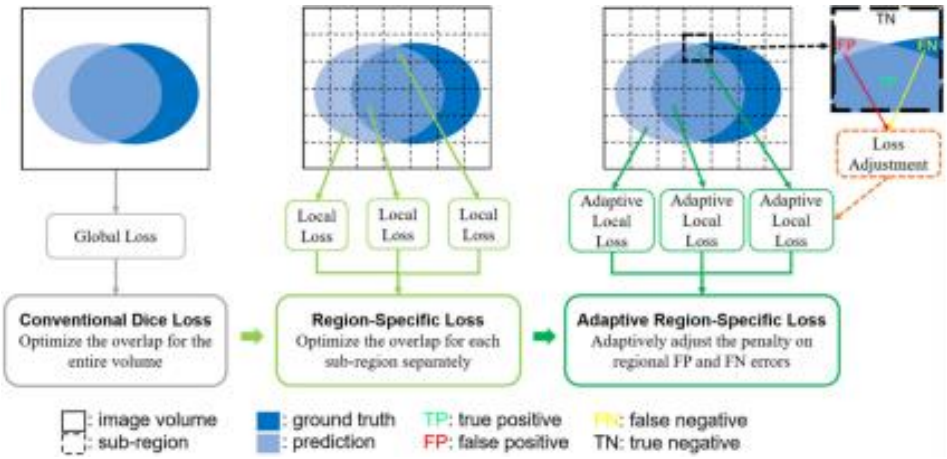


图 2. 一些损失函数对比

在像素级分割学习中，网络尝试最大化真正例（TP）预测并最小化假正例（FP）和假负例（FN）误差。DSC可以被视为精确度$P = TP/(TP + FP)$和召回率$R = TP/(TP + FN)$的调和平均，因此在Dice损失函数中，FP和FN被同等加权。但对于医学图像分割，标签和背景数据不平衡，使网络预测偏向背景，FN误差在网络训练期间往往比FP误差更为主导。为了在精确度和召回率性能之间实现更好的权衡，Tversky提出了Tversky Loss，定义如下：

$$L_{\text{Tversky}} = 1 - \frac{\sum_{i\in V} p_i \cdot g_i + \varepsilon}{\sum_{i\in V} p_i\, g_i + \alpha \sum_{i\in V} p_i(1 - g_i) + \beta \sum_{i\in V}(1 - p_i) g_i + \varepsilon} \tag{4}$$

当$\alpha = \beta = 0.5$时，它退化为$1 - DSC$。当β大于α时，Tversky损失更多地强调FN误差以提高召回率。原始Tversky损失中的α和β参数在训练前预定义。由于这些参数的选择严重影响最终学

习性能，它们的值通常通过手动试错进行微调，十分繁琐。所以可以进一步引入基于所提出的区域特定损失的每个子区域的自适应误差惩罚。这里Tversky被用作自适应区域特定损失计算的基础：

$$L_{\text{Adaptive-region-specific}} = \sum_{k}\left(1 - \frac{\sum_{i\in V_k} p_i \cdot g_i + \varepsilon}{\sum_{i\in V_k} p_i\, g_i + \alpha_{\text{Adaptive}} \sum_{i\in V_k} p_i(1-g_i) + \beta_{\text{Adaptive}} \sum_{i\in V_k}(1-p_i)g_i + \varepsilon}\right) \tag{5}$$

其中参数α_{Adaptive}和$\beta_{Adaptive}$在训练过程中进行微调，以强调不同类型的区域误差并优化网络学习。根据Tversky损失，α和β分别是FP和FN误差权重。在特定子区域中，如果FP误差大于FN误差，则应增加α的值以增加对FP误差的惩罚并提升精确度。同样，β的值将增加以惩罚FN误差并提升召回率。进一步来说，设计了以下算法根据给定子区域V_k中FP和FN误差的比例调整$\alpha_{Adaptive}$和$\beta_{Adaptive}$参数：

$$\begin{aligned}
\alpha_{Adaptive} &= A + B \cdot \frac{FP}{FP + FN} \\
&= A + B \cdot \frac{\sum_{i\in V_k} p_i(1-g_i) + \varepsilon}{\sum_{i\in V_k} p_i(1-g_i) + \sum_{i\in V_k}(1-p_i)g_i + \varepsilon} \\
\beta_{\text{Adaptive}} &= \text{A} + \text{B} \cdot \frac{\text{FN}}{\text{FP} + \text{FN}} \\
&= \text{A} + \text{B} \cdot \frac{\sum_{i\in V_k}(1-p_i)g_i + \varepsilon}{\sum_{i\in V_k} p_i(1-g_i) + \sum_{i\in V_k}(1-p_i)g_i + \varepsilon}
\end{aligned} \tag{6}$$

$\alpha_{Adaptive}$和$\beta_{Adaptive}$的值在[A, A+B]范围内变化，并分别随着FP和FN误差的比例线性增加。因此，根据训练过程中子区域内网络预测结果，损失函数对FP和FN误差的强调会自适应地进行微调。A和B经过多次测试，选择设置为0.3和0.4。

本损失函数不仅可以更多地关注难以预测的不同子区域，而且还实现了在训练过程中根据自动调整的函数参数对每个子区域进行自适应误差权重，可以实现更有效的自动分割学习。

分段平行Mamba层（Segmented Parallel Mamba Layers，SPM）

在基于SSM的Mamba中，通道数、SSM状态维数的大小、内部一维卷积核的大小、投影膨胀乘数和步长的秩都会影响模型参数。而在这些情况下，通道数的影响是最大的，其主要影响来自以下多个方向：首先，Mamba内部扩展投影通道的d_{inner}由投影扩展乘数和输入通道数的乘积决定。这可以用下面的等式来具体表示 [23]

$$d_{inner} = expand * d_{model} \tag{7}$$

其中，d_{inner}为内部展开投影通道，展开为投影展开乘数（默认固定为2），d_{model}为输入通道数。

其次，在Mamba内的输入投影层（两个分支都使用相同的输入线性层）和输出投影层的参数将与输入通道的数量直接相关。输入投影层和输出投影层的操作方式如下：

$$\begin{aligned}
in_{proj}&: nn.Linear(d_{model}, d_{inner} * 2, bias = False) \\
out_{proj}&: nn.Linear(d_{inner}, d_{model}, bias = False)
\end{aligned} \tag{8}$$

其中，输入投影（in_{proj}）层参数为$d_model * d_inner * 2$，输出投影层（out_{proj}）参数为$d_{inner} * d_{model}$。所以可以得出结论，输入通道的数量d_{model}是控制参数的关键因素，其中内部扩展投影通道d_{inner}也由d_{model}控制。

此外，SSM的中间线性投影层也是影响参数影响的关键。有关详情如下：

$$\begin{aligned}
x_{proj} &= nn.Linear(d_{inner}, dt_{rank} + d_{state} * 2), bias = False) \\
dt_{proj} &= nn.Linear(dt_{rank}, d_{inner}, bias = True)
\end{aligned} \tag{9}$$

其中dt_{rank}是步骤的秩（$dt_{rank} = d_{model}/16$），d_state是状态维数的大小（固定为16），其中参数可以推导出为$d_{inner} * (dt_rank + d_state * 2)$。$dt_proj$是步长的线性投影层，具有参数$(dt_rank * d_{inner}) + d_{inner}$，主要用于步长（$dt$）的线性投影。因此，我们可以得出结论，所有参数仍然主要由输入通道数d_{model}控制。

此外，内部卷积$\left(nn。Conv1d(d_{inner}, d_{inner}, d_{conv}, bias = True)\right)$还提供了$d_{inner} * d_{inner} * d_{conv} + d_{inner}$的参数影响。在本文中，$d_{conv}$固定为4，因此卷积提供了一个参数为$4 * d_{inner}2 + d_{inner}$的参数，该参数也由$d_{model}$控制。

基于以上输入通道数对于Mamba参数的关键因素讨论，所以如何减少通道数目时间少参数的关键步骤，所以提出并行的Mamba层的方法，称为SPM层。能够在保持处理通道总数不变的同时，在计算复杂度最低的情况下获得了良好的性能。

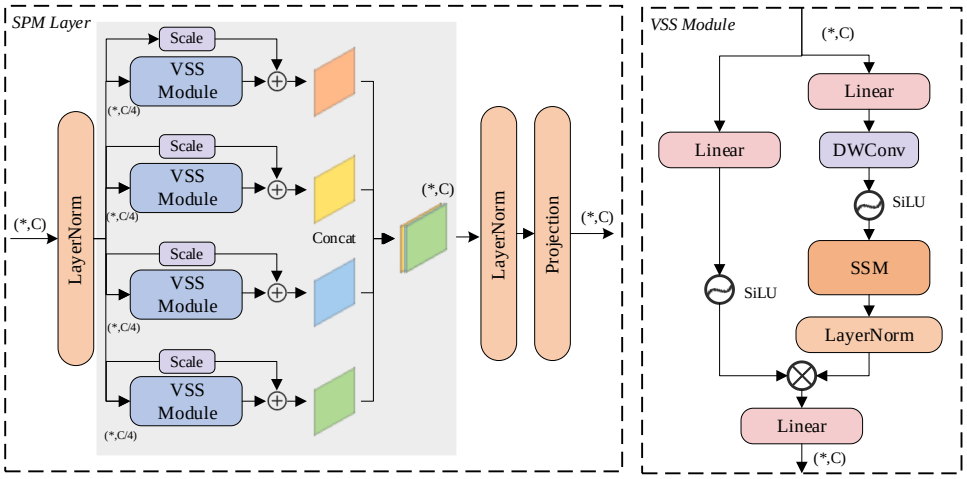


图 3. SPM模块架构

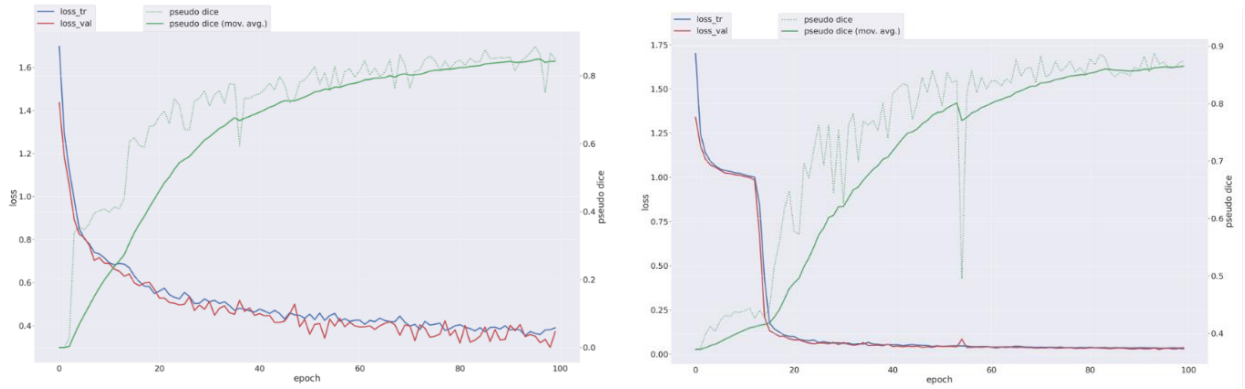
使用SPM来处理深层特征。通道号为C的特征X首先通过LayeNorm层，然后划分为$Y1 = C/4$、$Y2 = C/4$、$Y3 = C/4$和$Y4 = C/4$特征图，每个通道号为C/4。然后，将每个特征输入Mamba，然后对输出进行残余连接和调整因子，以优化提取远程空间信息能力。最后，将这四个特征组合成通道号为C的特征$Xout$，然后分别通过LayerNorm和投影运算输出。

5 实验结果分析

实验结果部分主要包括两个方面，一个ARSL损失函数对比实验部分，另一个是结合了SPM模块的总体框架部分，前者已经编码实现，而后者还没有编码实现。

ARSL损失函数对比试验结果部分

编写代码实现了ARSL损失函数，并使用原生的Dice与交叉熵损失函数，和ARSL函数与交叉熵损失函数分辨训练模型，分别得到下面两个训练过程结果（其余所有设置均一样）：

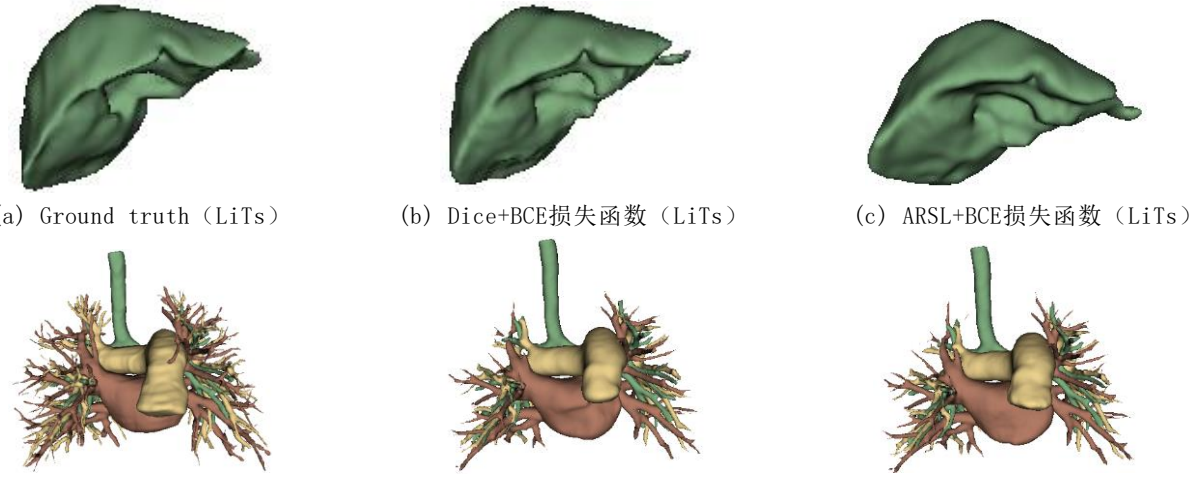


(a) Dice+BCE损失函数（LiTs）　(b) ARSL+BCE损失函数（LiTs）

(c) Dice+BCE损失函数（Lung79）　(d) ARSL+BCE损失函数（Lung79）

图 4. 不同损失函数在不同数据集上训练图

通过对比可以看出使用ARSL+BCE的损失函数相比于Dice+BCE的损失函数训练，能够更快的收敛，并且训练过程中其最终的Dice分数相比于前者更加优秀，性能更为突出。但为什么ARSL+BCE损失下降效果图如此奇怪，此时损失函数下降的斜率变化明显，这是因为随着训练的进行，损失函数中的随FP和FN误差权重α和β随训练进行达到一个合适的数值，从而加速了训练过程，初期阶段是误差权重是否合适和损失下降是一个正向反馈，但在后续随着损失难以继续下降，误差权重也逐渐稳定，则损失下降的速率逐渐降低，趋于平稳。这也是使用本损失函数其损失下降快的原因。



(a) Ground truth（LiTs）　(b) Dice+BCE损失函数（LiTs）　(c) ARSL+BCE损失函数（LiTs）

(d) Ground truth（Lung79）　(e) Dice+BCE损失函数（Lung79）　(f) ARSL+BCE损失函数（Lung79）

图 5. 不同损失函数在不同数据集上真实标签和预测标签对比

通过在这两个数据集上的训练效果图，不难看出，相对于单纯基于Dice和BCE的损失函数的模型的预测结果虽然大体上和真实值相同，但对于一些细节标记上不难看出，在图5的b上右上角的末端分割效果不是很好，而这在ARSL和BCE的损失函数上面来看相对更好。在Lung79上来看，虽然二者对于细粒度分割都不是很好，但相比于前者损失函数，后者在中间部分分割效果更好。综上可以发现，使用ARSL+BCE的模型相对于使用Dice+BCE损失函数的模型效果更好，尤其是在对于一些图像细节部分，这些往往会不太被模型注意的小部分上。而且训练过程中前者相对于后者的损失函数平稳得更快，能更快的得到所需模型。

结合了SPM模块的框架设计

将设计好的SPM层加入到LightM-UNet中，得到下面完整的架构图，显然这样的修改还是不够的，还需要验证具体训练效果如何，作为后续工作展望。

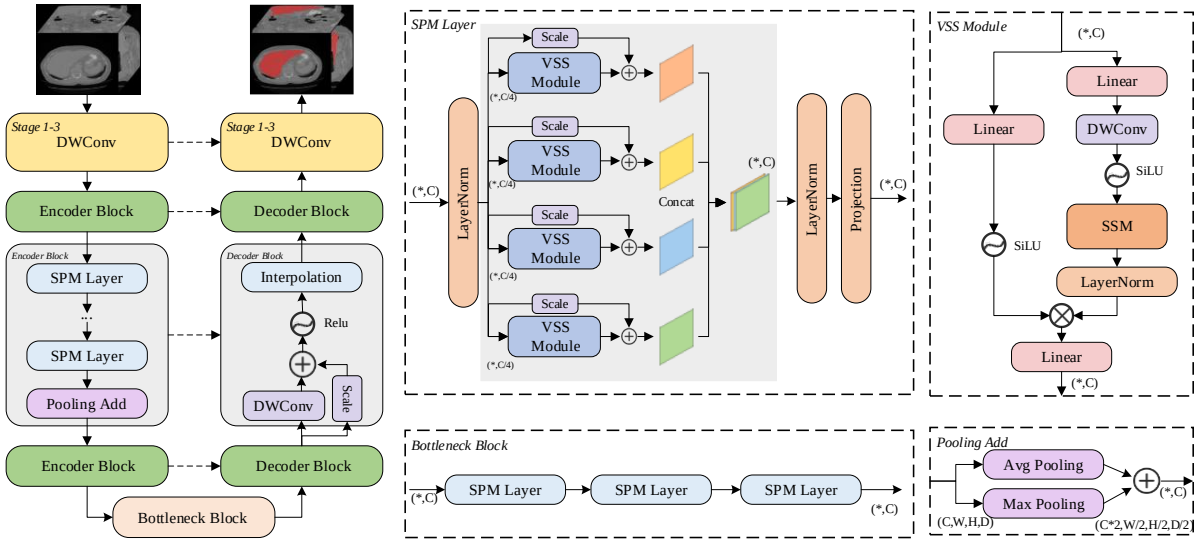


图 6. 融合SPM模块总体框架图

相比于作者工作替换了RVM为SPM模块之外，在编码层的池化模块将原始的池化操作修改为了池化和的方式，通过这样能够保证在缩小分辨率的同时不会丢失太多的图像特征。

6 总结与展望

复现一篇优秀的工作才能使得本人投身于该领域，是前沿工作的一次交互，通过本次复现，我对于领域相关的一些工作更为熟悉以及一些做法的意义所在。通过本次复现，不仅仅提高了我的代码能力，对于我本身文献阅读理解能力更是一次大的提升。本次复现过程中当然会遇到不少的困难，包括但不局限于，找到的最新工作是否符合自身的领域需求，找到的工作是否使用的是最新的方法，是否能够很好的执行论文工作的方法步骤，是否能够理解每一步的做法意义以及思考本次工作是否还有能够提升的地方，提升的方式，以及具体实现。

在阅读一些前沿文献之后，找到如上述的一些创新方法，有想法便实践。目前已经成功将损失函数这一块的工作实现并取得了较好的效果，遗憾的是重要的框架创新由于时间关系还没有具体实现，我相信通过这样的设计一定能达到后续的目标。实现该框架也正是后续的工作之一。

参考文献

[1] X. Bian, X. Luo, C. Wang, W. Liu, and X. Lin, “DDA-Net: Unsupervised cross-modality medical image segmentation via dual domain adaptation,” Comput. Methods Programs Biomed., 2022, 213:106531.

[2] W. Ji and A.C.S. Chung, “Diffusion-based domain adaptation for medical image segmentation using stochastic step alignment,” in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2024, MICCAI 2024. Lecture Notes in Computer Science, vol. 15008. Cham: Springer, 2024.

[3] H.R. Roth, D. Yang, Z. Xu, X. Wang, and D. Xu, “Going to extremes: Weakly supervised medical image segmentation,” Mach. Learn. Knowl. Extr., 2021, 3(2):507-524.

[4] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. Cham: Springer, 2015, pp. 234–241.

[5] A. Dosovitskiy, “An image is worth 16x16 words: Transformers for image recognition at scale,” arXiv preprint arXiv:2010.11929, 2020.

[6] T. Dao and A. Gu, “Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality,” arXiv preprint arXiv:2405.21060, 2024.

[7] A. Gu and T. Dao, “Mamba: Linear-time sequence modeling with selective state spaces,” arXiv preprint arXiv:2312.00752, 2023.

[8] W. Liao, Y. Zhu, X. Wang, et al., “Lightm-unet: Mamba assists in lightweight U-Net for medical image segmentation,” arXiv preprint arXiv:2403.05246, 2024.

[9] Chen L C, Papandreou G, Kokkinos I, et al. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 40(4): 834-848.

[10] Isensee F, Petersen J, Klein A, et al. nnu-net: Self-adapting framework for u-net-based medical image segmentation[J]. arXiv preprint arXiv:1809.10486, 2018.

[11] Liu Z, Lin Y, Cao Y, et al. Swin transformer: Hierarchical vision transformer using shifted windows[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 10012-10022.

[12] Chen J, Lu Y, Yu Q, et al. Transunet: Transformers make strong encoders for medical image segmentation[J]. arXiv preprint arXiv:2102.04306, 2021.

[13] Hatamizadeh A, Tang Y, Nath V, et al. Unetr: Transformers for 3d medical image segmentation[C]//Proceedings of the IEEE/CVF winter conference on applications of computer vision. 2022: 574-584.

[14] Zhou H Y, Guo J, Zhang Y, et al. nnformer: Interleaved transformer for volumetric segmentation[J]. arXiv preprint arXiv:2109.03201, 2021.

[15] Hatamizadeh A, Nath V, Tang Y, et al. Swin unetr: Swin transformers for semantic segmentation of brain tumors in mri images[C]//International MICCAI brainlesion workshop. Cham: Springer International Publishing, 2021: 272-284.

[16] J. Ma, F. Li, and B. Wang, “U-mamba: Enhancing long-range dependency for biomedical image segmentation,” arXiv preprint arXiv:2401.04722, 2024.

[17] F. Chollet, “Xception: Deep learning with depthwise separable convolutions,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.

[18] Touvron H, Cord M, Sablayrolles A, et al. Going deeper with image transformers[C]//Proceedings of the IEEE/CVF international conference on computer vision. 2021: 32-42.

[19] Bilic P, Christ P, Li H B, et al. The liver tumor segmentation benchmark (lits)[J]. Medical Image Analysis, 2023, 84: 102680.

[20] Jaeger S, Candemir S, Antani S, et al. Two public chest X-ray datasets for computer-aided screening of pulmonary diseases[J]. Quantitative imaging in medicine and surgery, 2014, 4(6): 475.

[21] Salehi S S M, Erdogmus D, Gholipour A. Tversky loss function for image segmentation using 3D fully convolutional deep networks[C]//International workshop on machine learning in medical imaging. Cham: Springer International Publishing, 2017: 379-387.

[22] Chen Y, Yu L, Wang J Y, et al. Adaptive Region-Specific Loss for Improved Medical Image Segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(11): 13408-13421.

[23] Wu R, Liu Y, Liang P, et al. Ultralight vm-unet: Parallel vision mamba significantly reduces parameters for skin lesion segmentation[J]. arXiv preprint arXiv:2403.20035, 2024.