**Deep Learning for Medical Image Segmentation (MedISeg)**

Medical Image Segmentation (MedISeg) is the most representative and comprehensive research topic in computer vision and medical image analysis. It can not only recognize the object category but also locate the pixel-level positions.

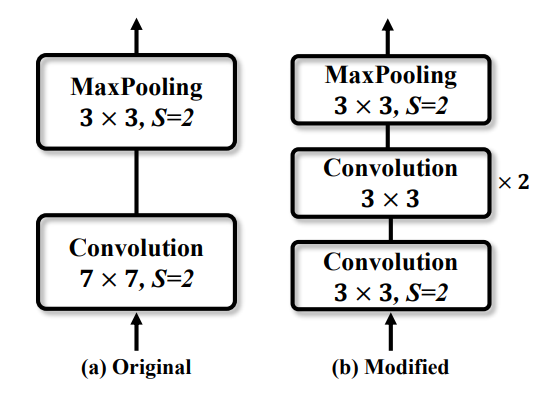


Fig. 1. Two implementation schemes of the input stem in ResNet, where (a) is the original implementation as claimed in its paper, and (b) is the modified implementation to reduce the computational costs. “S” denotes the stride size. “×2” denotes that this block is repeated twice.

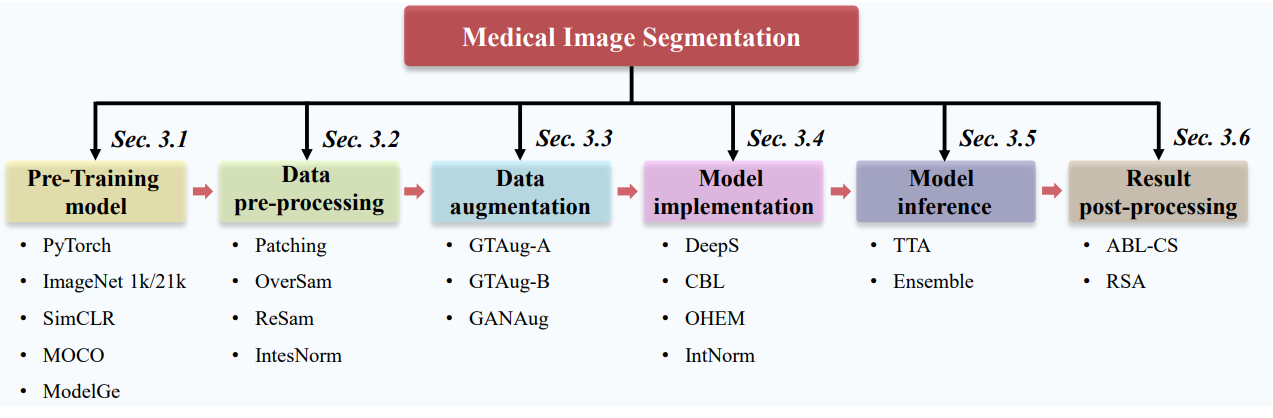


Fig. 2. An illustration of the surveyed MedISeg tricks and their latent relations. We separate the MedISeg model into six implementation phases, which are pre-training model, data pre-processing, data augmentation, model implementation, model inference, and result post-processing.

**Baselines**: **1)** **2D-UNet**: It consists of an encoder network and a decoder network. The encoder network follows classic fully convolutional architecture in that it has four spatial-tapering stages. Each stage consists of two 3\*3 convolutional layers followed by a rectified linear unit (ReLU) activation function and a global max pooling layer with stride size = 2.

The decoder network which takes output of encoder network as the input. It also has 4 stages that correspond to the same spatial encoder stages. Within each decoder stage, a 2D transposed convolutional operator first upsamples feature maps 2\*via bilinear interpolation operation and then two 3\*3 convolutional layers and a ReLU activation function are deployed in sequence. In the last decoder network layer, the channel size of the output feature maps is assigned to the class size of dataset used via a 2\*2 convolutional layer.

**2)** **3D-UNet**: It contains 3D encoder network and 3D decoder network. 3D-UNet is used to handle the 3D image dataset for segmentation. The implementation differences between 3D UNet and 2D UNet:

1. 2D convolutional layer is replaced by 3D convolutional layer.

2. A later connection is added between the same level of the encoder stage and a decoder stage which has the same spatial and channel size.

3. An intensity normalization layer is implemented on the input images.

**Platform**: The experiments are implemented on the PyTorch deep learning platform with NVIDIA GeForce RTX 2080 GPUs.

**Backbone**: ResNet-50 is used as the default backbone network which is the classic implementation of the fully convolutional network.

**Implementation for 2D-UNet**: 2D training images used for the lesion boundary segmentation and for the colon nuclei segmentation are uniformly resized into the fixed input size of 200\*200. The adaptive moment estimation is used as the optimizer where the weight decay is set to 0.0005.

**Implementation for 3D-UNet**: The stochastic gradient descent is used as the optimizer where the momentum is set to 0.9 and the weight decay is set to 0.0001. The cross-entropy loss is used as the unique loss function. In the inference phase, we perform patch-wise overlap with 50% region overlap.

**Methods And Experiments**:

**Pre-Training Model**: This method provides the favorable parameters such that training convergence can be easily accelerated, and the potential method can obtain a strong generalization ability. The network is pretrained on ImageNet consists of at least two basic forms (1K and 21K versions).

**PyTorch official weights**: These are backbone pre-trained weights provided by torchvision.models. These pre-trained weights are obtained by training the corresponding backbone network for the single-label image classification task.

**Model oriented ImageNet 1K weights**: As the backbone network that we use, the model-oriented ImageNet 1K weights can be obtained by training ResNet-50 for image classification.

**Model oriented ImageNet 21K weights:** Trained weights on the ImageNet 21K are more conducive to improving recognition performance of the down-stream computer vision models.

**SlimCLR weights**: It demonstrates introducing a learnable nonlinear transformation between feature representations and the contrastive learning loss can improve model representation quality.

SimCLR mainly consists of three implementation steps:

1) The input image is first grouped into some image patches.

2) Different data augmentation strategies are then implemented on image patches for different batches.

3) The model is finally trained to obtain the similar results for the same image patches with different augmentations, and mutually exclude other results.

**Model genesis (ModelGe) weights**: ModelGe is an advanced self-supervised model pre-training technology which consists of 4 transformation operations for single image restorations on computed tomography and magnetic resonance imaging (MRI) images.

**Data Pre-processing**: Four commonly used image pre-processing.

1. Patching: Training patching size is set to 96\*96\*96 without overlap and the patching size is set to 96\*96\*96 with 50% region overlap in the inference phase.
2. Oversampling: It is proposed to address the problem of class imbalance between positive and negative samples.
3. Resampling: It is proposed to improve the representational ability of the used dataset via machine learning model. It consists of four steps: 1. Spacing interpolation, 2. Window transform 3. Acquisition of mask effective range 4. Generation of sub-images.
4. Intensity Normalization: It is a specific normalization strategy used for medical images. There are two commonly used methods are z-scoring for all modalities and another one for computed tomography images.

**Experimental Results**: The larger the patching size, the better the model’s overall performance where the baseline model is based on patching96.

With the help of the Oversampling strategy, the model performance is overall improved.

Under the absence of resampling, the model performance is reduced on both KiTS19 and LiTS.

**Data Augmentation**: It can be used for MedISeg is divided into two categories: geometric transformation-based data augmentation (GTAug), and generative adversarial network (GAN)-based data augmentation (GANAug).

**Model Implementation**: There are 3 categories commonly used for the implementation:

1. Deep Supervision (DeepS),
2. Class Balance Loss (CBL),
3. Online hard example mining (OHEM)
4. Instance normalization (IntNorm)

**Deep Supervision**: It is an auxiliary learning trick and used for common scene image classification. This trick is implemented to supervise the backbone network by adding an auxiliary classifier or segmentation on some intermediate hidden layers in direct or indirect manner. It can be used to solve problems of training gradient disappearance or slow convergence speed.

Here extract feature maps from the last three decoder layers and use a 1 × 1 convolutional layer to project the lesion mask into the same channel size. Then, output feature maps from different layers are upsampled into the same spatial size as the input image for the segmentation head network by bilinear interpolations.

**Class Balance Loss**: CBL is usually used to learn a general class weight, i.e., the weight for each class is only related to the object category. CBL can improve model representational ability.

**Online hard example mining**: It filters out hard learning samples (i.e., images, objects, and pixels) via the loss function and these selected hard examples all have a high impact on the recognition tasks. These samples are applied to gradient descent in the model training process.

**Instance normalization**: IntNorm is a popular normalization algorithm that is suitable for recognition tasks with higher requirements on a single pixel. IntNorm is used in medical image because the batch size is set to small value during the training process.

**Model Inference**: The two kinds of inference tricks. Test Time Augmentation (TTA) and the model ensemble.

1. **TTA**: TTA can be used to improve recognition performance without training. So, it has potential to be a plug-and-play. At the same time, it can improve the ability of model calibration.
2. **Ensemble**: The model ensemble strategy aims to unite multiple trained models and achieve a multi-model fusion results on the test set based on a certain ensemble mechanism.

**Experimentation results**: The effectiveness of data augmentation scheme itself needs to be considered in a MedISeg model implementation.

**Results post-processing**: The purpose of the post-processing operations is mainly to improve model performance via non-learnable approaches. The two commonly used result post-processing schemes in the domain of medical image analysis are all-but-largest-component-suppression (ABL- CS), and removal of small area (RSA).

1. ABL- CS: It aims to remove some wrong areas in the segmentation results based on the knowledge of the organism’s physical properties.
2. RSA: For MedISeg, the imaging protocol is generally unchanged, such that the area of each instance segmentation mask remains unvaried as well. Based on this physical property, we can set a pixel-level threshold to remove some instance masks that are too small in the obtained segmentation masks.