

APPENDIX

Anonymous authors

Paper under double-blind review

1 DATA PRE-PROCESS

BRATS2020 dataset contains fully-paired multi-modality MRI images with labels for 369 patients. We choose 150 patients for the training set, 60 patients for the validation set, and the rest for the testing set. We expand our training dataset by data augmentation including random horizontal and vertical flip, rotate, zoom and random crop.

2 IMPLEMENTATION DETAILS

The learning rates for the generator and discriminator are 10^{-4} at the first 50 epochs and then linearly decay to 10^{-6} at the last epoch. While the learning rate for the segmentation network is 2×10^{-4} at the first 50 epochs and similarly linearly decays to 2×10^{-6} at the last epoch. The batch size is set to 8 by default. The generator is trained every three epochs while the discriminator is trained every epoch.

3 NETWORK ARCHITECTURES

The details of the network architectures are demonstrated as follows. The global and local discriminators are shown in Table 1. The segmentation network is shown in Table 2. And the generator is shown in Table 3.

Table 1: The architecture of both the global and local discriminator.

Layer	Discriminator
1	Conv2d (kernel size=4, stride=2, padding=1), LRelu
2	Conv2d (kernel size=4, stride=2, padding=1), LRelu
3	Conv2d (kernel size=4, stride=2, padding=1), LRelu
4	Conv2d (kernel size=4, stride=2, padding=1), LRelu
5	Conv2d (kernel size=4, stride=2, padding=1), LRelu
6	Conv2d (kernel size=4, stride=2, padding=1), LRelu
7-1	Conv2d (kernel size=3, stride=1, padding=1)
7-2	Conv2d (kernel size=2, stride=1, padding=0), Reshape

Table 2: The architecture of the segmentation network.

Layer	Segmentation network
1-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
1-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
1-3	MaxPool2d (kernel size=2, stride=2)
2-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
2-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
2-3	MaxPool2d (kernel size=2, stride=2)
3-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
3-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
3-3	MaxPool2d (kernel size=2, stride=2)
4-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
4-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
4-3	MaxPool2d (kernel size=2, stride=2)
5-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
5-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
6-1	Upsample(scale_factor=2, mode='bilinear')
6-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
6-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
7-1	Upsample(scale_factor=2, mode='bilinear')
7-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
7-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
7-1	Upsample(scale_factor=2, mode='bilinear')
7-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
7-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
8-1	Upsample(scale_factor=2, mode='bilinear')
8-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
8-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
9-1	Upsample(scale_factor=2, mode='bilinear')
9-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
9-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
10	Conv2d (kernel size=3, stride=1, padding=0)

Table 3: The architecture of the generator. Note that we only demonstrate a pair of encoder-decoder here and the architectures of the two pairs are the same.

Layer	Generator
1-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
1-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
1-3	MaxPool2d (kernel size=2, stride=2)
2-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
2-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
2-3	MaxPool2d (kernel size=2, stride=2)
3-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
3-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
3-3	MaxPool2d (kernel size=2, stride=2)
4-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu, Dropout
4-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu, Dropout
4-3	MaxPool2d (kernel size=2, stride=2)
5-1	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
5-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
6-1	Upsample(scale_factor=2, mode='bilinear')
6-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
6-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
7-1	Upsample(scale_factor=2, mode='bilinear')
7-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
7-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
7-1	Upsample(scale_factor=2, mode='bilinear')
7-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
7-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
8-1	Upsample(scale_factor=2, mode='bilinear')
8-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
8-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
9-1	Upsample(scale_factor=2, mode='bilinear')
9-2	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
9-3	Conv2d (kernel size=3, stride=1, padding=1), IN, LRelu
10	Conv2d (kernel size=3, stride=1, padding=0), Tanh