Project 1: Medical Image Analysis (Score 20)

Project 1 for Deep Learning, Spring 2025

Deadline: 2025.04.23 22:00

Medical image segmentation is an important and meaningful task in computer vision. In this project, you are required to apply deep learning techniques to perform the image segmentation of the MRI (magnetic resonance imaging) images of the brain. For a processed MRI image, you need to use a neural network (*e.g.*, CNN) to segment **the region of the lesion**. The dataset is from Kaggle [1], containing MRI images of 110 patients. Each patient corresponds to multiple images in `.tif` format with 3 channels. The corresponding segmentation annotations are single-channel binary masks (white for the lesion area). Figure 1 shows some examples. More details can be found in [2].

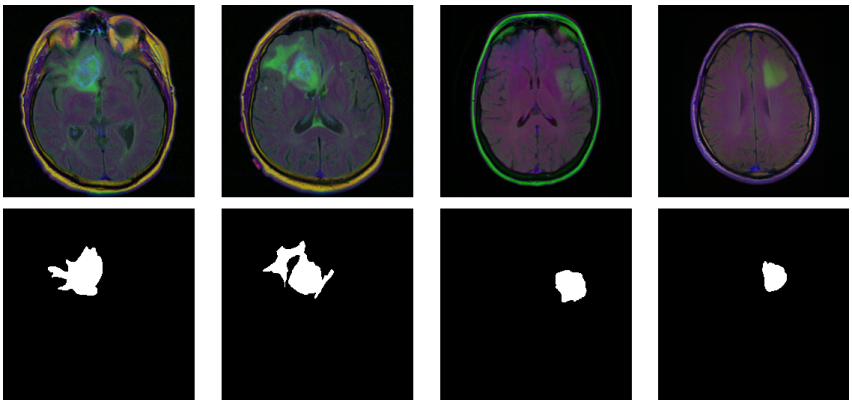

Figure 1: Brain MRI Images (first row) and their corresponding ground-truth masks (second row).

The dataset is also available at `https://cloud.tsinghua.edu.cn/f/098e06a8b7b74063a790/?dl=1`

In this homework, you are required to implement a **U-Net** [3] to perform the Brain MRI image segmentation task first. After that, you are required to **try some improvements** on the baseline to achieve better results.

1 Files Description

We provide a baseline to guide you finishing this project (both `MindSpore` and `Pytorch` version).

Our code framework is stored in the `brain-seg` folder. The framework contains the following files:

- `train.ipynb` describes the training process of the model;
- `inference.ipynb` is used to inference the trained model on the test set;
- `unet.py` defines the structure of U-Net [3]. **Please complete this file first**;
- `loss.py` defines the loss function (*i.e.*, Dice Loss [5]);
- `dataset.py` implements the class to read the dataset;
- `transform.py` is used to pre-process the data;

- `utils.py` defines some helper functions;
- `logger.py` is used to record the training curves (*e.g.*, loss curves) with TensorBoard [4].

Please configure your own environment first to run the baseline properly. You can refer to the installation commands at the top of `train.ipynb`.

2 Requirements

You are required to complete and run the baseline first and then try to improve the segmentation performance of the model on the MRI images of the brain. **You need to submit all codes and a short report** (pdf format) with the following requirements:

- Completing and successfully running the baseline (see the **# TODO** parts in `unet.py`).
- Trying some improvements on the baseline to achieve better results.
- Including visualization results (*e.g.*, some images and their corresponding segmentation results) and accuracy in your report (evaluated by the validation mean DSC).

Hint: You can improve the network structure, loss function, training process (e.g., optimizer), etc., and also refer to some tricks for image segmentation [6]

3 Attention

- You need to submit all codes and a report (at least two pages **in PDF format**). Delete the dataset and checkpoint file before submit.
- The report should detail all improvement attempts (whether successful or not) and corresponding results (DSC and several **segmentation examples**) and provide an analysis of the results.
- **Plagiarism of is not permitted**. If your implementation is based on any open source code, be sure to properly cite the source and include a description (*e.g.*, which parts are referenced).

References

[1] Brain MRI dataset: `https://www.kaggle.com/mateuszbuda/lgg-mri-segmentation`

[2] Buda et al. Association of genomic subtypes of lower-grade gliomas with shape features automatically extracted by a deep learning algorithm. Computers in Biology and Medicine 2019.

[3] Ronneberger et al. U-Net: Convolutional Networks for Biomedical Image Segmentation. MICCAI 2015.

[4] TensorBoard: `https://www.tensorflow.org/tensorboard`.

[5] Milletari F, Navab N, Ahmadi S A. V-net: Fully convolutional neural networks for volumetric medical image segmentation[C]//2016 fourth international conference on 3D vision (3DV). IEEE, 2016: 565-571..

[6] Minaee et al. Image Segmentation Using Deep Learning: A Survey. arXiv 2020.