Homework 3: Medical Image Segmentation

***GPUs may be needed for speeding up this training process. You are not allowed to use pre-defined U-Net**

Description

In this homework you will practice how to write U-Net in Python with TensorFlow or PyTorch. You need to understand how CNN works in order to implement this homework successfully. The goals of this homework are:

- To understand the steps to train/test the segmentation for medical image.
- Understand architecture of U-Net and how to connect each layer together by using TensorFlow or PyTorch.
- Understand how to compute the Dice score and IoU score.

Data preview:

The goal of this project/challenge is to predict the segmentation of Retina Blood Vessel**.**

The dataset can be downloaded from canvas homework 3.

The Retina Blood Vessel Segmentation dataset is a valuable resource for advancing the field of medical image analysis and enhancing the diagnosis of retinal vascular diseases. This dataset contains a comprehensive collection of retinal fundus images, meticulously annotated for blood vessel segmentation. Accurate segmentation of blood vessels is a critical task in ophthalmology as it aids in the early detection and management of various retinal pathologies, such as diabetic retinopathy and macular degeneration.

Content:

The dataset comprises a total of *N* high-resolution retinal fundus images captured using state-of-the-art imaging equipment. Each image comes with corresponding pixel-level ground truth annotations indicating the exact location of blood vessels. These annotations facilitate the development and evaluation of advanced segmentation algorithms.

Key Features:

Image Size: The images in the dataset are of varying dimensions mimicking the real-world diversity of retinal images.

Annotations: For each image, corresponding pixel-wise annotations in a *binary mask* (might need a little process to make it binary) format are provided. Blood vessel pixels are marked as 1, while background pixels are labeled as 0.

Pathological Variation: The dataset encompasses a spectrum of retinal conditions, including varying vessel widths, branching patterns, and presence of anomalies, making it suitable for evaluating the robustness of segmentation models.

Step 1: Data

Before starting to train a model, please get familiar with the dataset. When you look at the dataset, please answer the following questions: 1) How many data samples are included in the dataset? 2) Which problem will this dataset try to address? 3) What is the dimension ranging in the dataset? 4) Does the dataset have any missing information? E.g., missing features. 5) What is the label of this dataset? 6) How many percent of data will you use for training, validation and testing? 7) What kind of data pre-processing will you use for your training dataset?

Hint: You should use the same test dataset to compare the models' performance. Opencv or PILLOW are nice package to load and process images.

Step 2: Model

Here I selected U-Net model, you will experience different hyperparameters (include model architecture). Please change the following hyperparameters and report the model performance in the testing dataset.

Step 3: Objective

Which loss function are you going to use to train your model?

Step 4: Optimization

You are going to select your optimization function to train the models. Report the optimization you selected in this section and explain the reason for using the optimizer.

Step 5: Model selection

Based on your training experience, which model gives the best performance (accuracy). Have you experienced different learning rates?

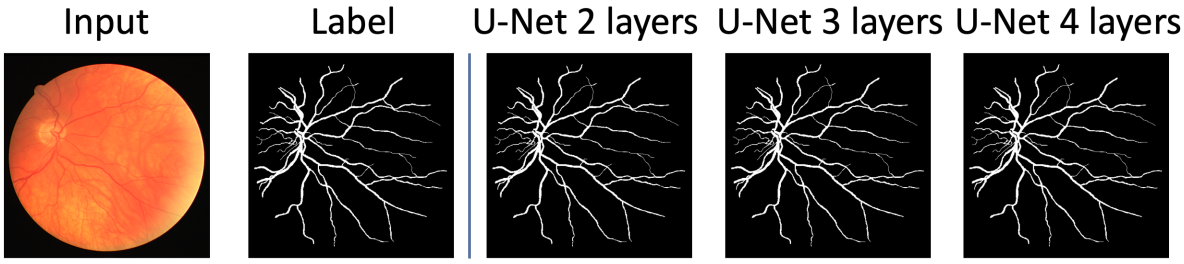
Model	Dice and IoU with normalization	Dice and IoU without normalization
U-Net 2 blocks	e.g dice 0.8, IoU 0.4	
U-Net 3 blocks		
Any Other?		

How do you avoid overfit your model and underfit your model?

Step 6: Model demo

In this step you should report your model prediction demo, which you did in the previous steps.

A. Show 4 text images with label and models prediction as example showing below:



Note: this is not real prediction

What should you submit?

You should submit a zip file (named as firstname_lastname_hw3.zip – e.g. jun_bai_hw2.zip) containing:

1. Your homework report from step 1- 6. You will answer all the questions in each step and fill the tables in step 2, step 5 and performance plot in step 6. Missing any part will lose some points. Please double check you have addressed all the questions.

2. Your code of all models and a README file explaining how to run the model. Your code should be well commented. In your code, you should have a function called *test_model*. The *test_model* function will load the trained model weights and load test dataset to predict.

3. Your highest performed U-Net model weights.

4. A folder contains screenshots of iteration of models' training and testing.