Final Project Report

Siyuan Chen [siyuanch@kth.se](mailto:siyuanch@kth.se)

**Preface**

According to the requirements of the final project for Ph.D. students, I conducted all my projects by myself, and my final report is consist of two projects. Proj\_A is the ‘Comparison Between PCA and VAE on the Medical Data Reduction’, which partly related to my research field. Proj\_B is the ‘Segmentation of Brain Tumors, which comes from the open data source. Here I appreciate this course because this course leads me into the world of image processing from blank and greatly benefits my research.

**Introduction**

Proj\_A: Data from medical images generally are highly dimensional, while sometimes researchers can not focus on all the features. As the linear unsupervised dimension reduction method, PCA has been used to reduce the dimension in much medical research. From this course, Variational Autoencoder (VAE) can also be used to reduce the dimension and generate new results but with the framework of machine learning, which brings out this project – to compare the different two methods on some small medical data.

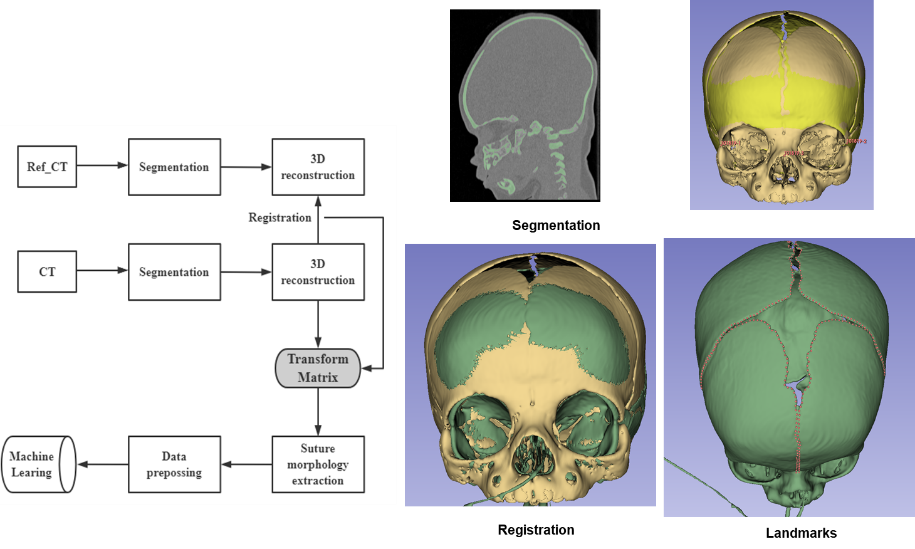
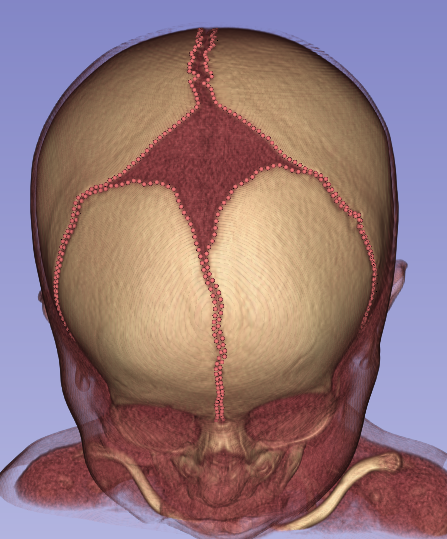
Proj\_B: This project is more relevant to this course. The dataset comes from the BraTS, which is an open-access dataset with 494 subjects’ MRI images with brain tumors. The main idea of this project is to build the code framework to achieve a multi-label segmentation task combined with all the knowledge I learned from the previous lab works. The data were divided into a training set and a validation set. The segmentation is based on the U-NET, the fitting process is combined with the callback tech, and various metrics are used in this project to evaluate the model performance.

**Main methods**

Proj\_A: The project is divided into 4 parts (Read the data, PCA Analysis, VAE Analysis, and Result comparison).

Data preprocess is based on the 3D slicer, where the morphology of all the subjects’ sutures (n = 69) are represented as 3d 800 landmarks. To overcome the limitation of the small number of data groups, the idea of ten-folder cross-validation is employed. Every method is used ten times with exclusively 1 group as test data, and the final mean values of ten results are compared.

Here the detail of the PCA method is omitted. For the VAE method, the encoder is consist of 5 convolution layers with the base number double each time, followed by 5 maxpooling layers, and 1 flatten layer and dense layer. The decoder is set as asymmetric with the encoder. Besides that, for the VAE, the latent space is the combination of the last dense layer and the sampling layer. The function of the sampling layer is to regularize the latent space. It is also noted that for VAE, the loss function consists of two parts, one is the regular loss function, and another is called the KL term, which is also used to regularize the model to improve the generalize ability. The metrics I used for this project are location square deviation, LSD and location mean deviation, LMD.

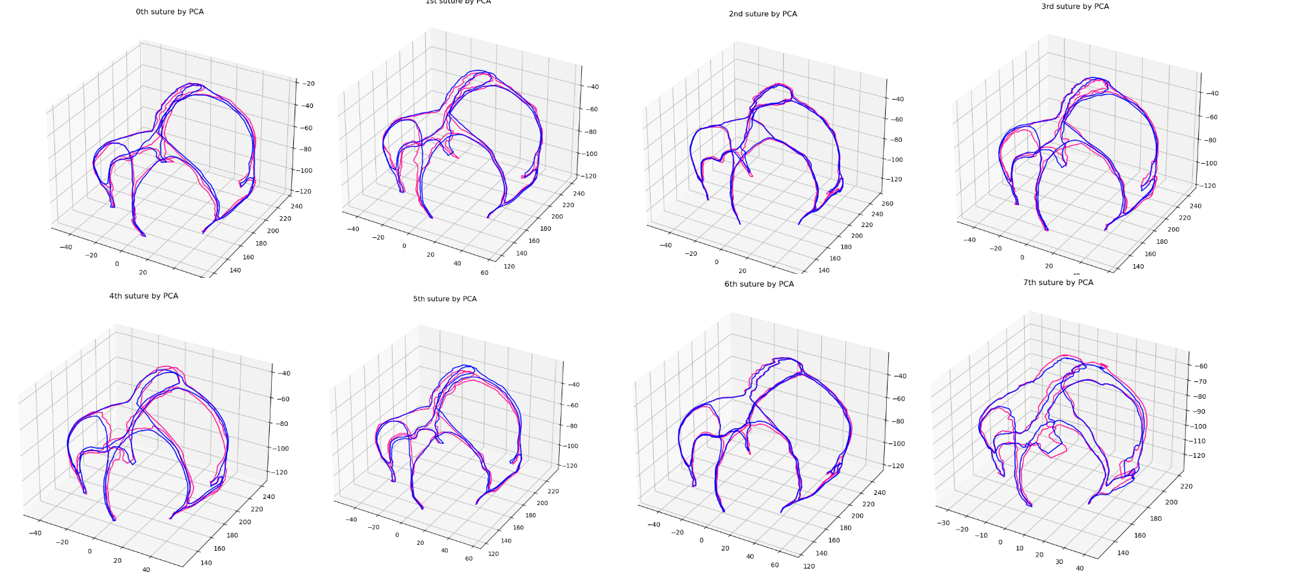
 

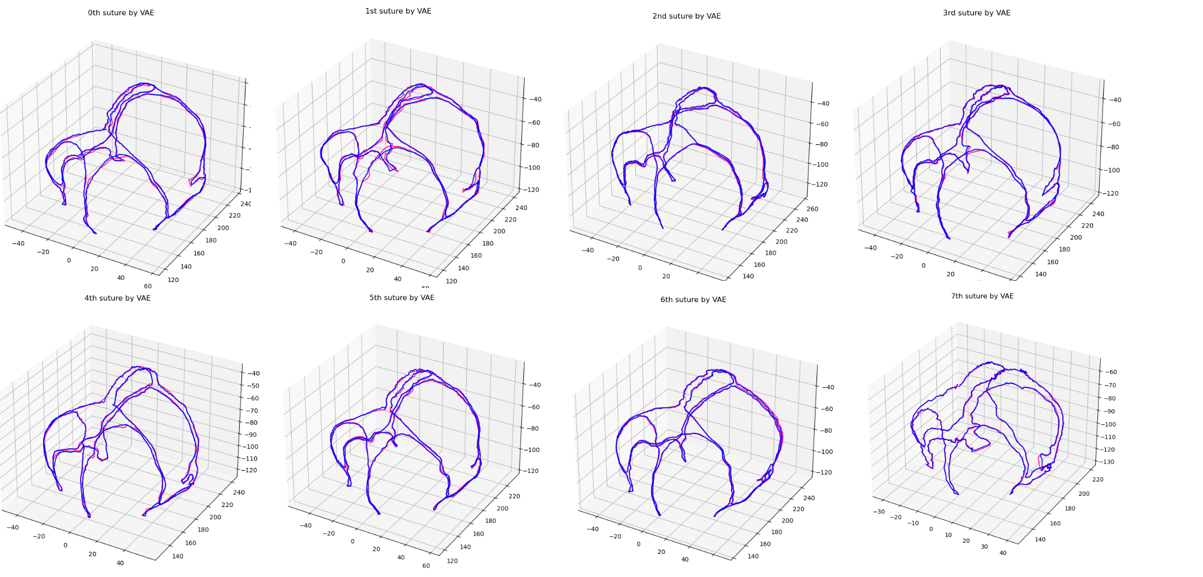
Proj B:

1. **Read the data.** Read the list of all the folders and convert the list to id. The final data are made up of train, test, and validation. The image size is set as 128\*128 with two channels. For each subject, there are five different kinds of file (Flair, t1, tice, t2, mask). The flair file and the t1ce file are inputed into two channels, and there are 4 labels in the Mask, where 0 represents background, 1 means core, 2 means edema, 3 means enhancing.
2. **Use the U-Net Architecture.** In total 4 concatentate layer which is same with what we built in Labwork.
3. **Add callback for training process.** I use CSVLogger to save my history, and use ModelCheckpoint to direct my training model develop towards the direction of making good validation result.
4. **Train the model.** training\_generator = DataGenerator(train\_ids, batch\_size=BATCH\_SIZE \* strategy.num\_replicas\_in\_sync)
   1. **Generator**
   2. **Compile** Use the categorical\_crossentropy as loss function, and use Aam as optimizer, and use various metrics.
   3. **Fit** Epochs is set as 30 combined with the callbacks setting
5. **Loading the history** Read the log file that I saved before
6. **Predict the Results** Define some functions to visualize the predicted tumor and compare with the real.

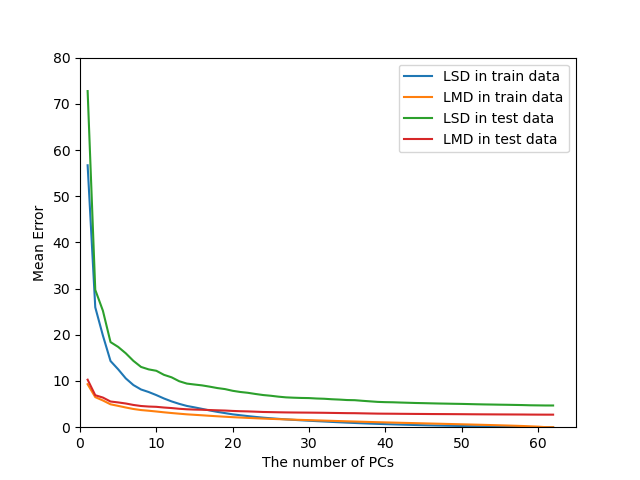
**Results**

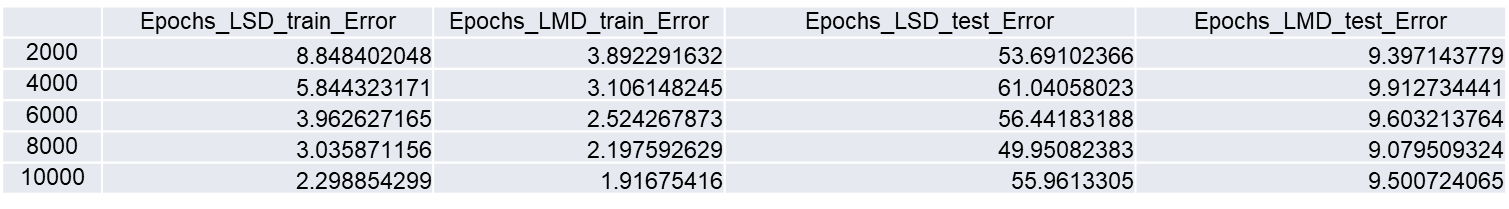
Proj\_A:

With only five features, PCA and VAE’s results on training data are shown as below.

****

The influence of the number of PCs on the mean error is shown as

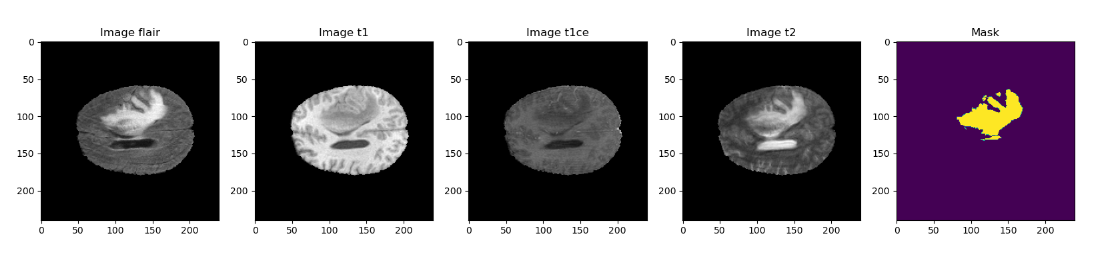


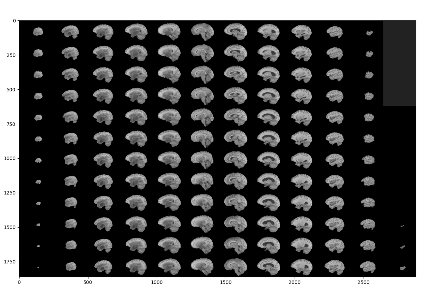
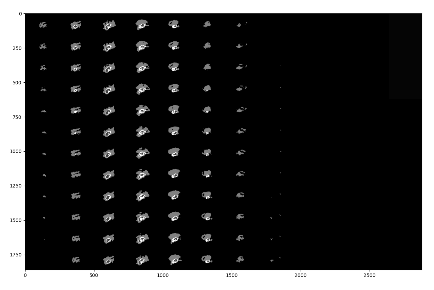


We can also see the LSD and LMD in VAE is smaller than PCA for training data, but VAE result on the test data is very bad, which means the VAE has a very severe overfitting.

Proj\_B:

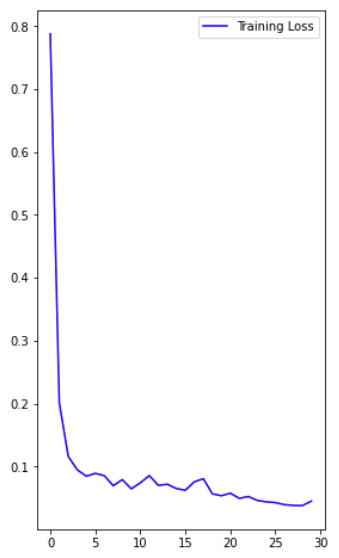
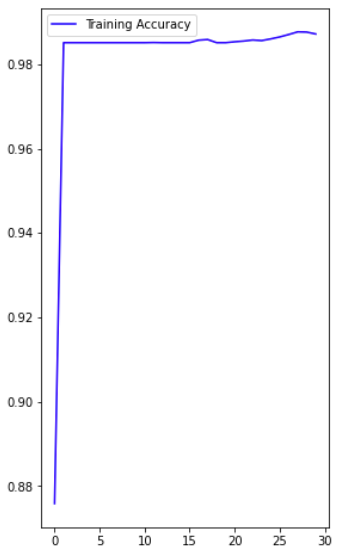
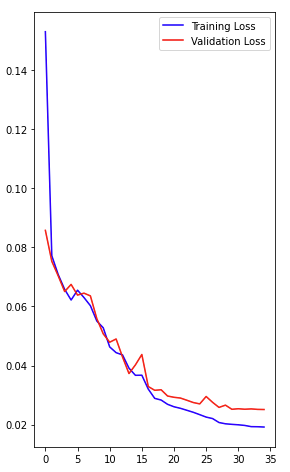
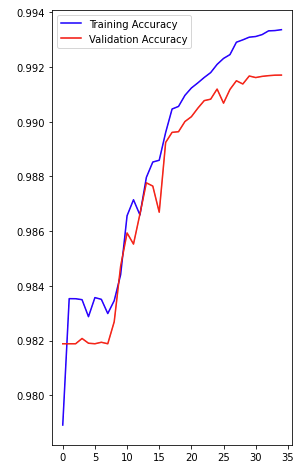
For each training data, there are five files. Below I show the first sample as the example.

Different slices of this subject’s T1 image and mask image are shown as below.

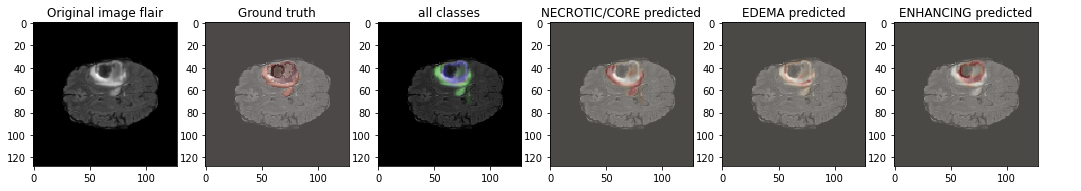
 

The ratio of Train data, valid data, and test data are 0.68, 0.2,0.12.

Below shows the accuracy result and the loss results with the callback. Compared with the accuracy result and loss results without a callback, we can see the result with callback has better performance, and the accuracy can increase with the increase of epoch.



Below you can see the predicted tumors result and different types of tumor in brain. (multi-lable)



Below compare the difference between the real tumor and the predicted tumor of one training data.

