Diabetic Macular Edema Segmentation using Deep Learning.

Bruno Soares de Castro
up202309106@up.pt

Faculty of Engineering at the University of Porto - FEUP
International Master in Computer Vision - IMCV

Introduction

This report details the complete methodology for the automatic segmentation of pathological fluid regions using OCT scans. To achieve this goal, we have designed a basic pipeline with simple steps, which is represented in Figure 1.

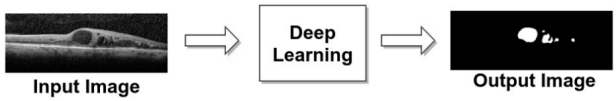

Figure 1: Main step of the proposed methodology

Methodology

The methodology for implementing the recognition system involved some key steps, including splitting the data between the train and test datasets, and subsequently carrying out cross-validation on the training data. The following section outlines the detailed approach taken to accomplish each of these tasks.

Image Dataset

The dataset consists of 50 images with corresponding ground truth labels. It was divided into a 90% training dataset and a 10% test dataset. For each architecture, a 6-fold cross-validation strategy was employed.

Deep Learning

In this project, we utilized two architectures: the Feature Pyramid Network (FPN) and the Pyramid Scene Parsing Network (PSPNet). Both architectures were employed with the same objective: segmentation. In Table 1, we present the parameters that were configured for training the model.

Table 1: Parameters of deep learning model architecture

	FPN	**PSPNet**
Encoder	resnet34	resnet34
Learning rate	0.0001	0.0001
Optimizer	Adam	Adam
Batch size	5	5
epoch	20	20
threshold	0.5	0.5

Cross-Validation

For each architecture, a 6-fold cross-validation strategy was employed, as illustrated in Figure 2.

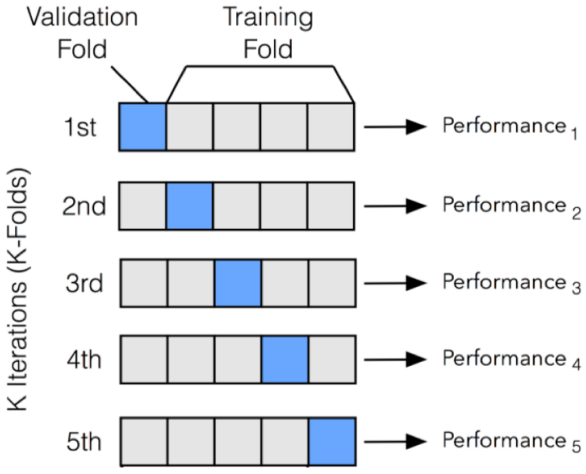

Figure 2: Cross Validation Methodology

During the process of splitting the dataset into training and validation sets, we normalized each training.

After performing cross-validation and verifying that the models did not overfit, we refit our model using the entire 90% training dataset and evaluated its performance using the test data as the final assessment.

Metrics

To properly evaluate the performance of the models, we calculate two measures during training. The first measure is the loss function, and the second measure is the mean Intersection over Union (IoU).

Results

The validation dataset loss function results for the FPN and PSPNet models during training are presented in Table 2. Across six cross-validation folds (CV), both models exhibit varying performance.

Table 2: Validation Dataset Loss Function Results During Training

	FPN		**PSPNet**	
	First	Last	First	Last
CV 1	0.7033	0.2103	0.9697	0.1957
CV 2	0.8089	0.2323	0.9519	0.1682
CV 3	0.6778	0.1697	0.9466	0.2160
CV 4	0.7290	0.1354	0.9533	0.3407
CV 5	0.7912	0.1539	0.9611	0.2614
CV 6	0.6271	0.1866	0.9718	0.1882

In the case of FPN, the initial loss (First) ranges from 0.6271 to 0.8089, with a final loss (Last) ranging from 0.1354 to 0.2323. Meanwhile, for PSPNet, the initial loss spans from 0.9466 to 0.9718, with final losses ranging from 0.1682 to 0.3407. Notably, FPN has shown great results from start to finish.

The validation results from the IoU metric, depicted in Table 3, offer insights into the performance of two segmentation models, FPN and PSPNet.

Table 3: Validation Dataset IoU Results During Training

	FPN		**PSPNet**	
	First	Last	First	Last
CV 1	0.2189	0.6656	0.07837	0.6821
CV 2	0.1290	0.5532	0.05832	0.5792
CV 3	0.2368	0.6856	0.03574	0.5548
CV 4	0.2006	0.7228	0.07084	0.5063
CV 5	0.1450	0.7107	0.02186	0.5779
CV 6	0.2671	0.6895	0.11450	0.6485

Throughout the training process, both FPN and PSPNet exhibit varying degrees of IoU across different cross-validation folds. Notably, FPN demonstrates relatively higher IoU scores compared to PSPNet across almost all folds.

In Figure 3, we present the validation dataset results for each epoch, considering the FPN architecture.

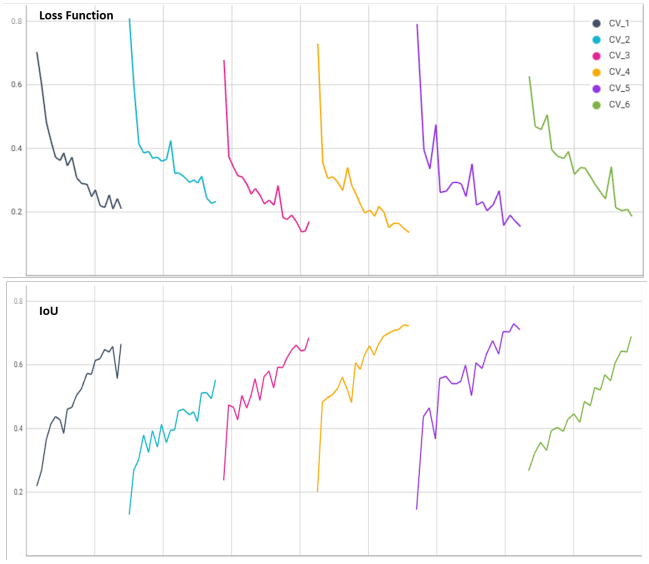

Figure 3: Loss Function and IoU Results for Each Cross-Validation Iteration for FPN Architecture

A similar result to the previous one is evident among the cross-validation folds, leading us to consider that although we do not have an optimal value for the loss function and IoU, our model is not overfitting.

In Figure 4, we present the validation dataset results for each epoch, considering the PSPNet architecture.

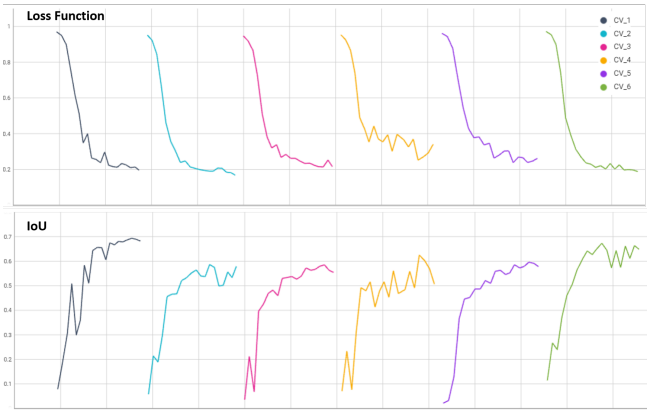

Figure 4: Loss Function and IoU Results for Each Cross-Validation Iteration for PSPNet Architecture

For the PSPNet architecture, it is evident that, unlike the FPN architecture, the values for the loss function and IoU stabilize before the 20th epoch. This demonstrates that for PSPNet, 20 epochs were sufficient, whereas with FPN, there might be room for improvement in metrics. However, PSPNet does not exhibit overfitting.

The results for the test dataset after refitting using the same parameters that were used for cross-validation for both models are presented in Table 3.

Table 4: Test Dataset Loss Function and IoU Results after refitting

	FPN	**PSPNet**
Loss Function	0.1850	0.2047
IoU	0.7179	0.6580

The results here only confirm that FPN yielded slightly better results than PSPNET.

In Figure 5 and 6, we have the predicted images for the FPN and PSPNet architectures, respectively.

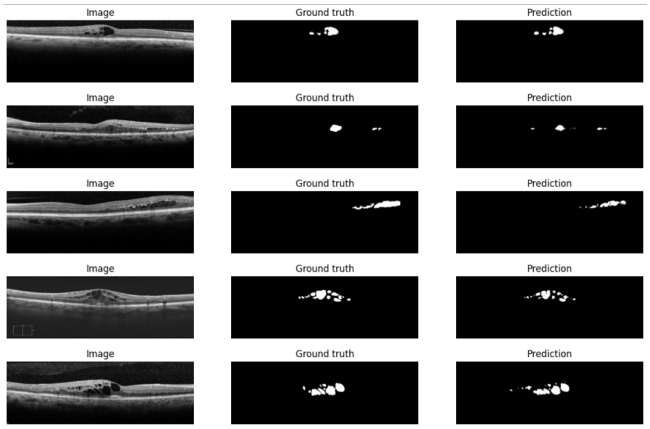

Figure 5: Predicted image from the test dataset using the FPN architecture.

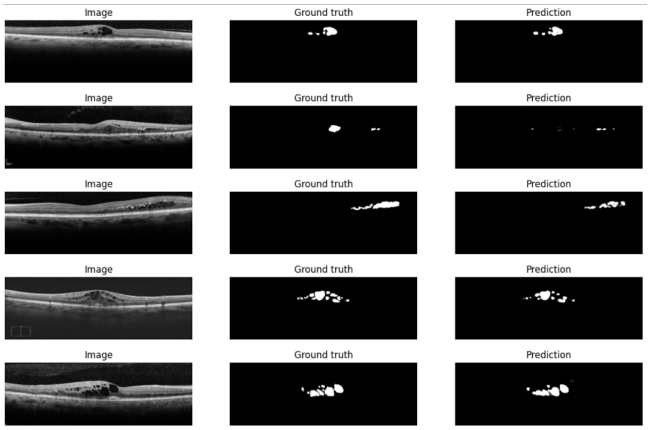

Figure 6: Predicted image from the test dataset using the PSPNet architecture.

Conclusion

Even with a small dataset, the results from cross-validation demonstrated optimal performance for both the FPN and PSPNet architectures.

In this experiment, both models exhibited excellent performance metrics. However, FPN yielded slightly higher metrics compared to PSPNet.