DEEP LEARNING FOR SEGMENTATION OF INTRACRANIAL VESSEL WALL PATHOLOGIES

Anitha Bhat Talagini Ashoka, 223736
Pavan Kumar Kandapagari, 221281

Under Supervision of
Dr.-Ing. Sylvia Saalfeld, Msc. Annika Niemann
Otto-von-Guericke Universität Magdeburg
Department of Computer Science

November 2020

Contents

Abstract:

Reliable semantic segmentation for histological images of intracranial aneurysm wall is essential for both scientific research and clinical practice. The deep learning method for intracranial aneurysm wall segmentation is a new approach for the given histological image dataset. Identification of different tissue textures from these images and verification with subject experts was the first prominent step. We recognized 9 unique tissue textures namely Organizing Thrombus, Intact Wall, My-ointimal Hyperplasia, Degenerated Wall, White Thrombus, Decellularized OT(/MH) (OT= Organized Thrombus, MH = Myointimal Hyperplasia), Red Thrombus, Inflammatory Cells and Infiltrating Blood. We hex colour coded each tissue texture and extracted the ground truths. We implemented a U-net[7] architecture for this multi-class semantic segmentation problem. The developed model is trained for 50 epochs and converges at the 39th epoch, giving the best results for the selected hyper-parameters. We chose accuracy, Tversky loss, and Dice co-efficient as evaluation metrics. We achieved 76.73% test accuracy. Our developed model segmentation result was better than the ground truth segmentation. Further improvements are possible with more class examples for training, hyper-parameter tuning, and transfer learning techniques.

Introduction

Semantic segmentation of medical pathologies is one of the key challenges in medical image processing. Semantic segmentation is a high-level task that paves the way towards complete image understanding. The importance of image understanding is a core computer vision problem which is highlighted by the fact that an increasing number of applications nourish from inferring knowledge from imagery.

Histological image analysis is part of a computer-assisted diagnosis. A brain aneurysm[4] is an abnormal bulge or "ballooning" in the wall of an artery in the brain. They are sometimes called berry aneurysms because they are often the size of a small berry. Most brain aneurysms produce no symptoms until they become large, begin to leak blood, or burst. The medical term for an aneurysm that develops inside the brain is an intracranial or cerebral aneurysm. Most brain aneurysms only cause noticeable symptoms if they burst (rupture). This leads to an extremely serious condition known as a subarachnoid hemorrhage, where bleeding caused by the ruptured aneurysm can cause extensive brain damage and symptoms. If a brain aneurysm is detected before it ruptures, treatment may be recommended to prevent it from rupturing in the future. Most aneurysms do not rupture, so treatment is only carried out if the risk of a rupture is particularly high[1]. This Project is to help to identify these aneurysms early by providing semantics for the images so that required treatment can be carried out.

For the given histological images of a brain aneurysm, identification of the different tissue textures is a crucial first step which in turn can be helpful for the histopathologist to analyze the image and identify the aneurysm. Manual segmentation of these different tissues is a reliable method, but the process is time-consuming and laborious. Another problem is when images are stored in their raw format they occupy a very large size (approx. 200MB per image) on disk, these in turn cause issues for processing the image

[1] https://www.nhs.uk/conditions/brain-aneurysm/

by any software or script.

Digitized tissue histopathology has now become amenable to the application of com-

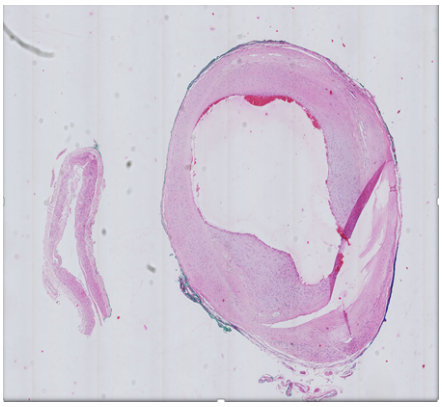

Figure 1.1: An example of input image received in .tiff(raw) format

puterized image analysis and machine learning techniques. The dramatic increase in computational power and advancement in deep learning techniques make an optimistic view towards these approaches. In digital image processing and computer vision, image segmentation is the process of partitioning a digital image into multiple segments. Semantic segmentation adds information to the image which is helpful and easier for further automatic analysis.

Convolutional Neural Networks (CNNs) is surpassing other approaches in terms of accuracy and efficiency[10]. Among these, the U-net[7] architecture is state of the art for bio-medical image segmentation which can be trained end-to-end. Hence we propose a network with U-net architecture that segment the nine different tissue textures and also differentiates the background which is identified in the given dataset of histological images.

Related Work

There has been several research in biomedical image segmentation for better predictive model performances, but semantic segmentation of intracranial aneurysm wall using deep learning technique is a new research problem. The following literature and published papers were used as a reference to understand the pre-existing work, to design and implement our model for aneurysm wall pathologies.

Histologic segmentation has Large complexity and variability of appearances and shapes of anatomical structures make medical image segmentation one of the most challenging and essential tasks in any CAD system. Due to diversity of objects-of-interest, image modalities, and CAD problems no universal feature set and general segmentation technique exist. Some popular techniques are filter and threshold based, rule-based, texture-based clustering, statistical, atlas-based, and deformable models-based techniques[2].

Alberto Garcia-Garcia, et al [3] attempt to summarise different semantic segmentation methods based on deep learning techniques. They used PASCAL VOC2012, PASCAL Context, PASCAL, Person-Part, CamVid, CityScapes, Stanford Background and Sift-Flow datasets to evaluate most popular deep learning methods for semantic segmentation. DeepLab [1] is the most solid method which outperforms the rest in all the 7 mentioned RGB images dataset by a significant margin. The most popular dataset PASCAL VOC2012 having 2618 images containing 4754 annotated objects and 10 different classes gave result of 79.70 IOU.

Jonathan Long et al [6] adapted contemporary classification networks(AlexNet [5],the VGG net [11], and GoogLeNet [12]) into fully convolutional networks and transfered their learned representations by fine-tuning to the segmentation task. They then define a novel architecture that combines semantic information from a deep, coarse layer with appearance information from a shallow, fine layer to produce accurate and detailed seg-

mentation. This fully convolutional network achieves state-of-the-art segmentation on PASCAL VOC dataset (20% relative improvement to 62.2% mean IU on 2012). For this particular network patchwise training with sampling do not yield faster or better convergence for dense prediction. Whole image training was effective and efficient.

Olaf Ronneberger et al [7] presented a network and training strategy that relies on the strong use of data augmentation to use the available annotated samples more efficiently. The architecture consists of a contracting path to capture context and a symmetric expanding path that enables precise localization. Due to it's shape it is also called as U-net. It showed that such a network can be trained end-to-end from very few images and outperformed the prior best method (a sliding-window convolutional network) on the ISBI challenge for segmentation of neuronal structures in electron microscopic stacks. For PhC-U373 and PhC-U373 datasets they achieved 0.9203, 0.7756 average IOU respectively.

Wang et al [13] proposed a deep learning framework for cancer metastasis detection. 400 whole slide histopathological images further divided into 270 for training and 130 for testing. While pre-processing, first they identified tissue within the histopathological images and exclude background white space. They used patchwise training of size 256*256 for this binary classification problem. Performance is evaluated by four well-known deep learning network AlexNet [5], VGG16 [11], GoogLeNet [12] and FaceNet [9]. GoogLeNet and VGG16 achieved the best patch-based classification performance giving accuracy of 98.4% and 97.9%. Combining with deep learning system's predictions with the human pathologist's diagnoses produced a major reduction in pathologist error rate. These results suggest that integrating deep learning-based approaches into the work-flow of the diagnostic pathologist could drive improvements in the reproducibility, accuracy and clinical value of pathological diagnoses.

Dataset & Methods

3.1 Dataset

The given dataset had 177 images divided into 3 different aneurysm samples namely Com, S-340, S-385. All the images are in Tagged Image Format(.tiff) with large dimensions of at least 8000×8000 pixels. The initial step was to identify the different tissue textures present in these histological images and distinguish the differences between them. After the careful analysis of the different tissue textures and discussion with our supervisor also verification with the subject experts, we classified a total of 9 different tissue textures from the given set of images. To extract the ground-truths, we hex colour coded each tissue texture. The tissue textures and the hex colour code used of the segmentation model and classes for the respective textures are shown in Figure 3.1 also the medical explanations for all the tissue textures are as follows:

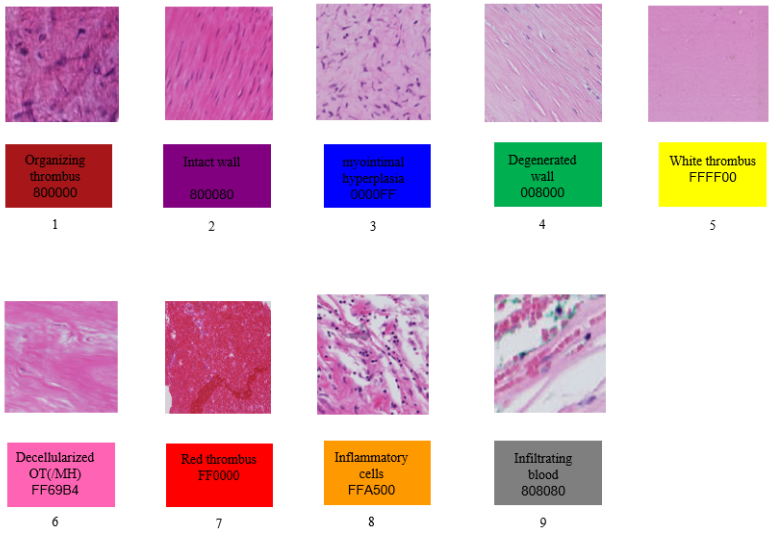

Figure 3.1: Tissue textures observed and extracted from input images along with their respective color mappings and hex codes

1. **Organizing Thrombus**: A thrombus containing fibrous tissue that may be structured into layers. Tiny blood vessels may course through the clot
2. **Intact Wall**: Linearly organized smooth muscle cells
3. **Myointimal Hyperplasia**: Mixed distribution of spicular nuclei
4. **Degenerated Wall**: Smooth muscle cells
5. **White Thrombus**: This is thrombus of most likely solely fibrin with very few red blood cells
6. **Decellularized OT(/MH) (OT=Organized Thrombus, MH=Myointimal Hyperplasia)**: By histology it is really difficult to tell whether the origin of this tissue is wall or thrombus, because after decellularization (even if the origin was organized thrombus) they both look the same. However based on the relation of this area with the dark red areas, in this particular samples, the pinks are most likely OT
7. **Red Thrombus**: Fresh thrombus with a lot of red blood cells
8. **Inflammatory Cells**: Most likely inflammatory cell infiltration but should always be confirmed in immunohistochemical methods (e.g. staining for CD45)
9. **Infiltrating Blood**: Mixed structures/undefined Red blood cells infiltrated between the connective tissue bundles

From the analysis and number of samples in the given dataset, we divided these tissue texture classes into minority and majority example classes. Myointimal Hyperplasia, Degenerated Wall, White Thrombus, Red Thrombus tissue texture are considered as majority class examples due to their large availability present in the given dataset, and remaining tissue texture classes are considered as minority class examples.

3.2 Methods

In the following subsection, we explain the pre-processing steps taken, the method developed for multi-class semantic segmentation. We explain U-net architecture which predicts each pixel's class for the derived 9 tissue structures and also differentiates the background of the image.

3.2.1 Data pre-processing

The essential pre-processing task was the creation of ground truths for the histological image dataset. We used Adobe Photoshop[1] software to create segmentation masks. After examining the individual images from the dataset, it was clear that the dimension of each image was very huge to fit in the memory of any moderate machine.
We designed a function that makes the patches of the required size and includes an overlapping parameter in percentage. We made patches of dimension 256×256 pixels from the images and masks. Then these patches are converted to '.png' format to reduce the storage size. We also took into consideration the problem with the alpha channel i.e., RGBA images in the original '.tiff' patches while converting the file format to '.png'.
Both image and mask patches are renamed the same to avoid mismatching in the nomenclature. Then they are stored in the image and mask folders. The extracted patches from images and masks had a large bias towards the background, so we filtered out the patches which having only background pixels from the dataset. Figure 3.2 and Figure 3.3 show an example image and for which the ground truth was created.

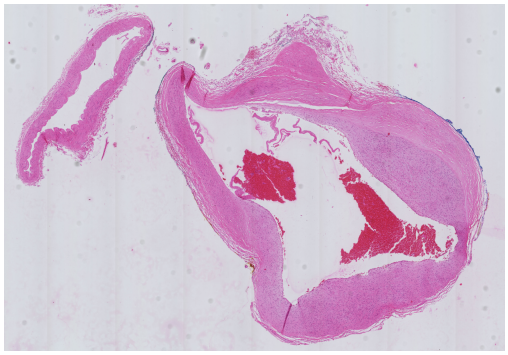

Figure 3.2: An example image of dataset that shows the textures associated in a cross section of blood vessel which are used for segmentation

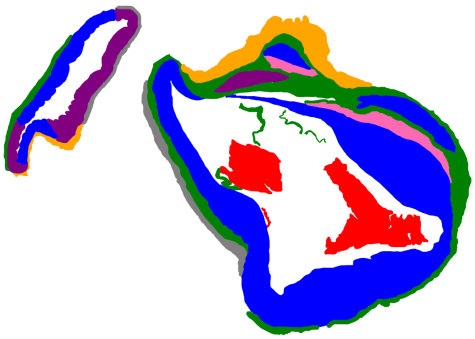

Figure 3.3: Corresponding 3.2 Semantic ground-truth represented in RGB domain

K-Means clustering is used for the pre-processing of the masks in order to convert the RGB image from 3 channel image to 1 channel label. This also helps us to avoid color imbalance and transparency artifacts caused by Photoshop as shown in Figures 3.4 and 3.5
The same color pallet used for making the ground truth was used for the various clusters in the K-means as can be seen in Figure 3.6. Methods and functions used for K-mean

[1]https://www.adobe.com/products/photoshop.html

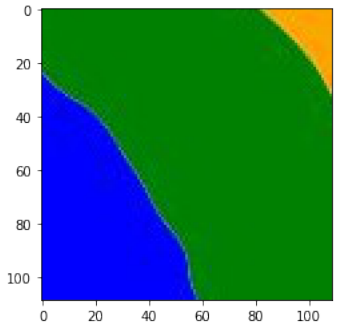

Figure 3.4: Mask of an image before passing through K-means model as it shows blurs , smudges and other artifacts on it causing it to represent different numpy value in those pixels

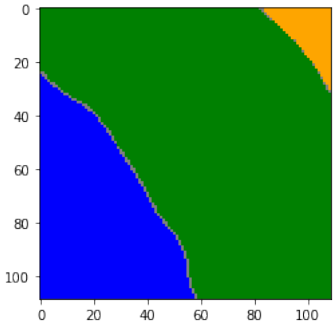

Figure 3.5: Mask after running it through a K-means model shows no artifacts and clearly defined borders between classes

clustering is included in the Figure 3.7. After this we convert the sparse array into a one-hot encoding with depth equal to number of classes. Finally we show the full masks without and with k-means implementation in Figures 3.8 and 3.9

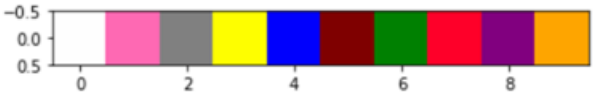

Figure 3.6: Colors represented in the K-means model and their classes

```
t0 = time()
color_palette_2, w, h = preprossesing_image(r'/data/student/github/DLforWallCharacteristics/color_palette_
2.jpg')
kmeans_color_palette_2 = KMeans(n_clusters=n_colors, random_state=42).fit(color_palette_2)
print('done in %0.3fs. ' % (time() - t0))
print(kmeans_color_palette_2.cluster_centers_)
```

Figure 3.7: code for K-means

Images and masks patches were sorted and stored into respective folders after the K-mean clustering which are further divided and stored in train, test and validation folders respectively according to train, test and validation split ratio. Custom data generators were designed for creating batches of images and masks for model input. Before training the model we normalize the data by scaling them to have a values between 0 and 1.

3.2.2 U-net architecture

The main idea behind CNN is to learn the feature mapping of an image and exploit it to make more nuanced feature mapping. In image segmentation, we need to convert feature

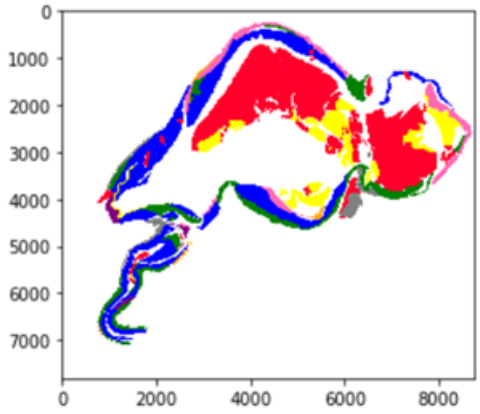

Figure 3.8: Example for an image that is read directly from disk, artifacts cannot be seen easily at this scale

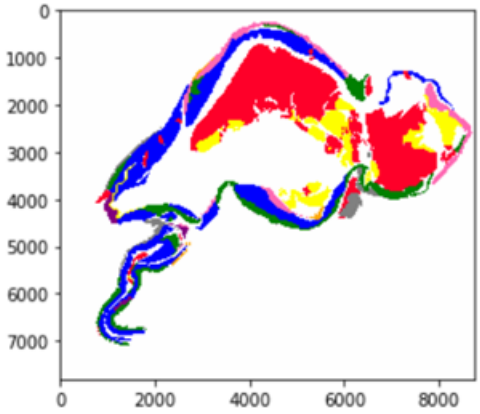

Figure 3.9: Example of the same mask after passing through K-means model, all artifacts removed

map into a vector as well as to reconstruct an image from this vector. This is a mammoth task because it's a lot tougher to convert a vector into an image than vice versa.

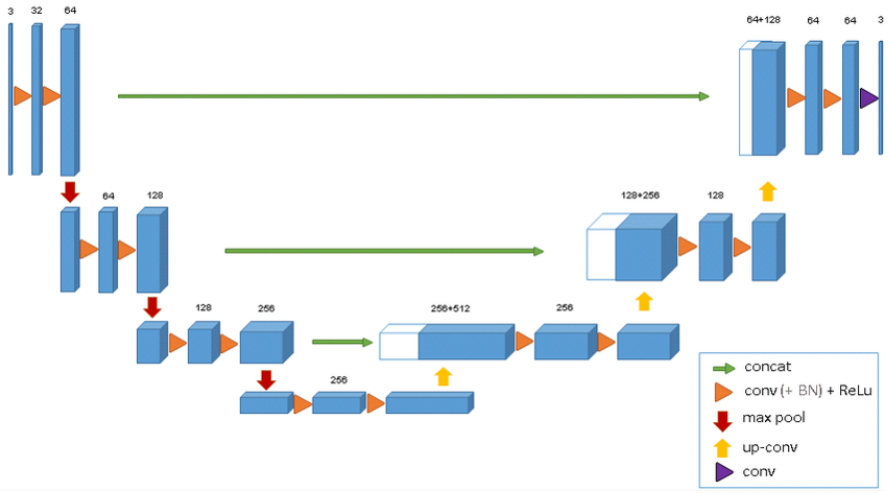

Figure 3.10: A representation of U-Net architecture use in the model

U-Net[7] uses the same feature maps that are used for the contraction to expand a vector to a segmented image. This would preserve the structural integrity of the image which would reduce distortion enormously.

Justifying its name U-net architecture looks like a 'U', consisting of three sections: The contraction, the bottleneck and the expansion section. The contraction section consists of many contraction blocks. Each block takes an input, applies two 3×3 convolution layers followed by a 2×2 max pooling. The number of kernels or feature maps after

each block doubles so that architecture can learn the complex structures effectively. The bottom most layer mediates between the contraction layer and the expansion layer. It uses two 3×3 CNN layers followed by 2×2 up convolution layer and batch normalize all the convolution layers.

The Expansion section similar to the contraction layer, it consists of several expansion blocks. Each block passes the input to two 3×3 CNN layers followed by a 2×2 up-sampling layer. After each block, number of feature maps used by the convolutional layer becomes half to maintain symmetry. However, every time the input gets appended by feature maps of the corresponding contraction layer. This action would ensure that the features that are learned while contracting the image will be used to reconstruct it. The number of expansion blocks is the same as the number of contraction blocks. After that, the resultant mapping passes through another 3×3 CNN layer with the number of feature maps equal to the number of segments desired.

We retained the same U-net[7] architecture with 18 convolutional layers(contraction and expansion path) with ReLu activation function and last output convolution layer with Softmax activation function for classification. The lesser convolutions layers compared to the original U-net architecture is justified by the smaller patch sizes we used i.e 256×256 pixels. Whereas the original U-net architecture is designed and trained for 512×512 pixels patches[7]. The U-net architecture of our developed model is given in Figure3.10

3.2.3 Parameter Initialization

We divided the dataset into train, test, and validation sets on the ratio 60:20:20 respectively. To handle data imbalance, we assured our training and validation dataset have moderately equal majority class examples and the presence of all minority class examples. We manually picked the patches of minority class examples and added them into each of these sub-divided datasets. The Dataset consists of 7314 patches from which 4876 patches were used for training, 1219 patches for validation, and 1219 patches for testing. The model trained with Adam optimizer for default learning rate(0.0001). We used categorical cross-entropy loss for this multi-class segmentation problem. Evaluation metrics are accuracy, Tversky loss[12] and Dice co-efficient.

Our developed model trained for 50 epochs. We also added callbacks while training the model, weights were saved when the model improves, training weights were saved in '.h5' file format. For early stopping criteria, validation Dice co-efficient was used. The model stops training where there were no improvements from the last 10 epochs. Our model

training stopped at 39 epochs. In other words, the model converged at this point of training.

Evaluation & Discussion

4.1 Results

For the multi-class semantic segmentation problem for brain aneurysm histological images, we developed a U-net model which trained for 50 epochs considering accuracy, Tversky loss [8] and Dice co-efficient as evaluation metrics. The Tversky index is to address the issue of data imbalance and achieve a much better trade-off between precision and recall in training which is specially used in fully convolutional deep neural networks[12]. Based on experimental results, the Tversky loss function act as a generalized framework to effectively train deep neural networks. The Tversky loss score is computed for each class separately and then summed up.
We used Adam optimizer initialized with its default learning rate(0.0001), ReLu activation function for non-classifier layers, and Softmax activation function for the final classifying layer. We used the kernel shape of 3×3. Implemented early stopping which monitors validation Dice co-efficient. Model checkpoints along with best model and weights were saved in .h5 file format.
Among the different combinations of hyper-parameters we tried, the table below represents the best results.

Dataset	Accuracy(%)	Tversky loss	Dice coefficient
Training	72.14	9.2681	0.8401
Test	76.73	9.2783	0.8111

Table 4.1: Results for train, validation and test data

Evaluation plots for training and validation loss, training and validation accuracy are shown in Figure 4.1, and training and validation Dice coefficient plot is shown in Figure 4.2
From the overall results, we notice that our developed model segmentation results were

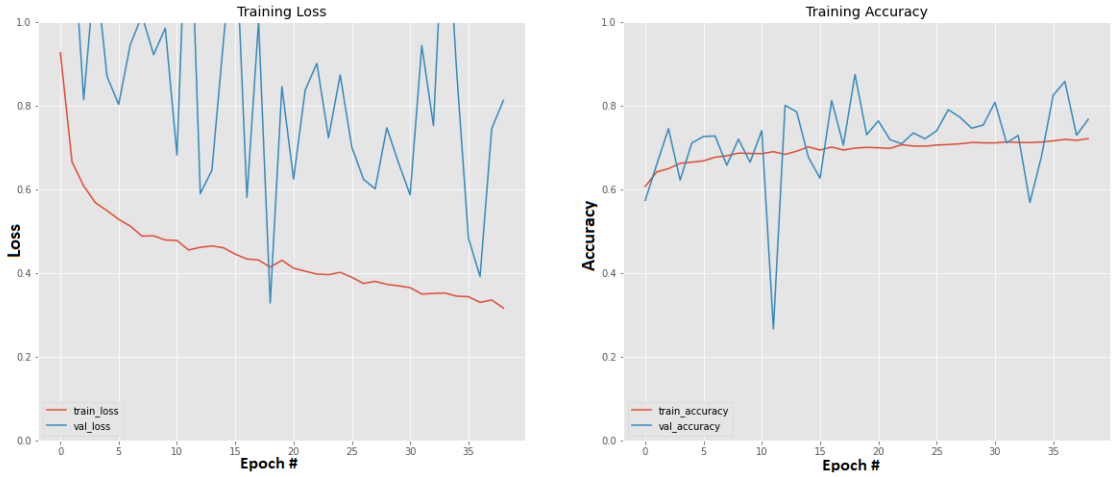

Figure 4.1: Training and validation Accuracy and loss plots

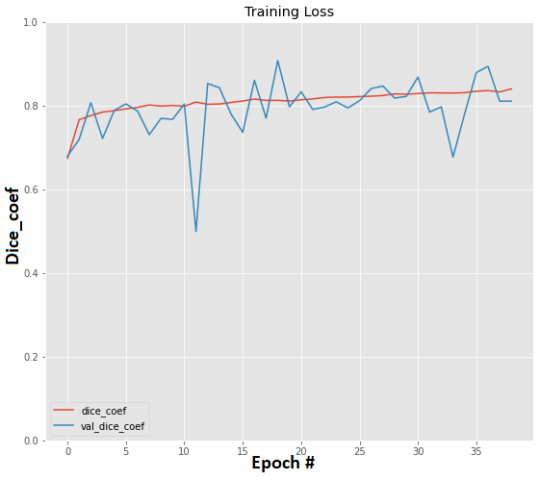

Figure 4.2: Training and validation Dice coefficient value plot

better than that of segmentation ground-truths. For Red Thrombus, Myointimal Hyperplasia Cells were accurately identified in the majority of example cases compared to all other classes. This is justified as it has a higher number of available examples and less overlapping cell structure. White Thrombus, Intact Wall, Infiltrating Blood tissue textures gave moderate classification results. Organizing Thrombus, Intact Wall, Decellularized OT classes are still not identified by the network. which is suspected to be due to lower samples present in the dataset. From observation on the resultant images for some of the patches, we notice that Decellularized OT is mostly identified as white thrombus by our trained model.

The following Figures 4.3 4.4 4.5 depict good and bad segmentation results.

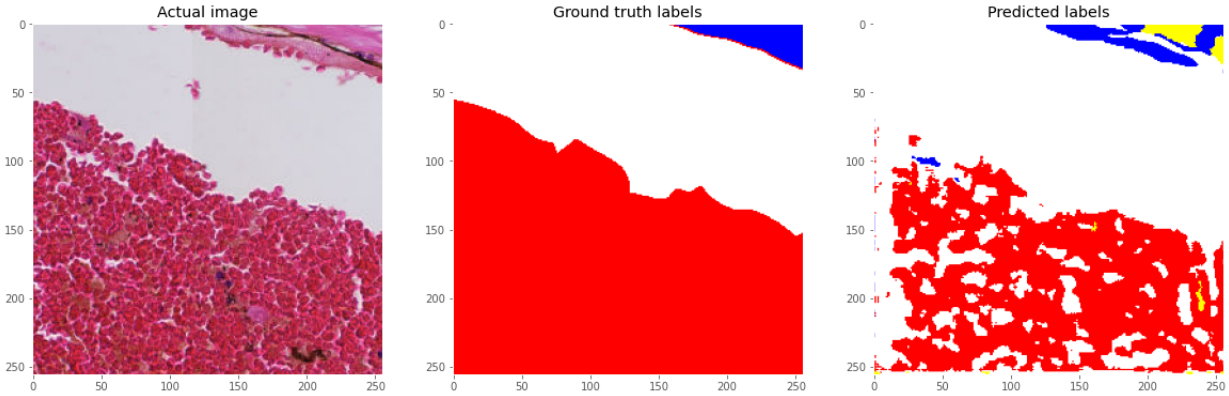

Figure 4.3: Example for reasonably good visual prediction

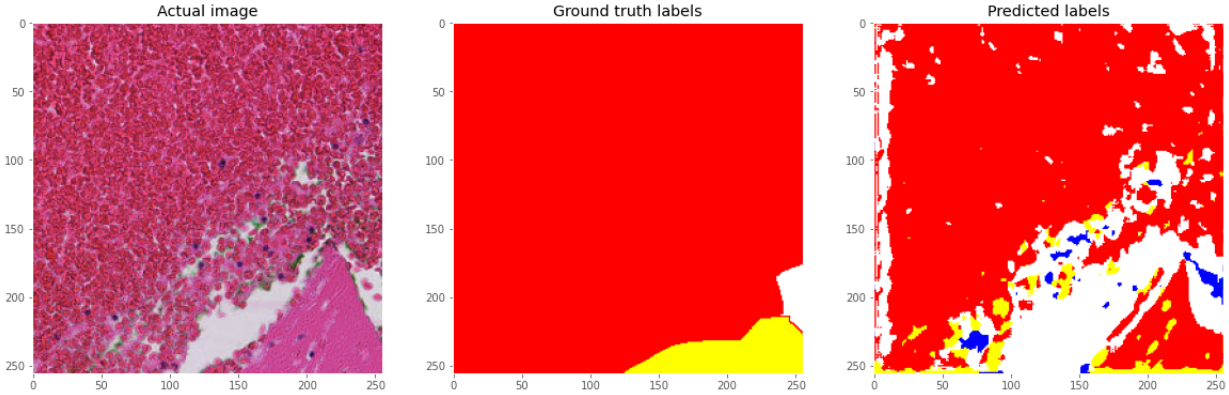

Figure 4.4: Example for another good prediction

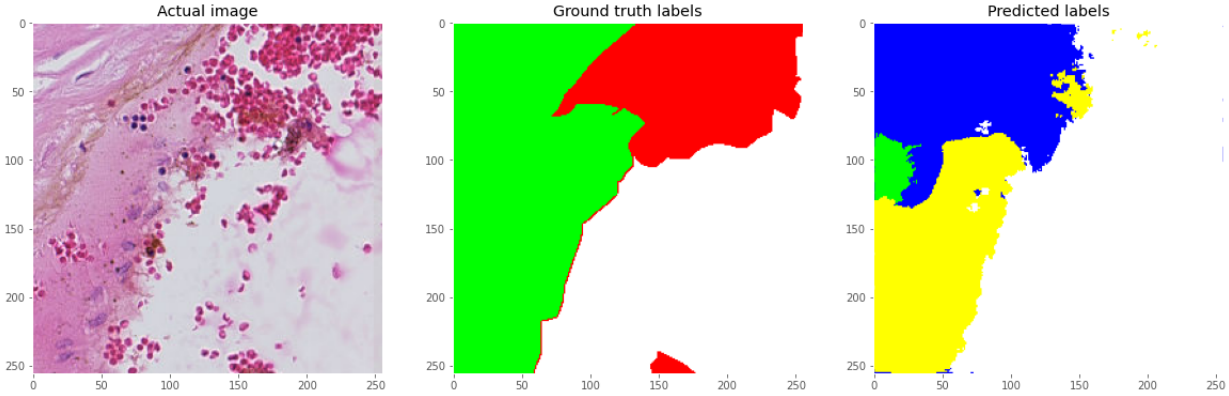

Figure 4.5: In this example visually almost all the classes are misrepresented

Summary & Future Work

5.1 Summary

This project involved the tasks of analysis of brain aneurysm histological images for the identification of different tissue structures. The first step was the creation of ground truth labels for the given dataset. Data pre-processing involved the generation of patches, with the required overlapping configuration from images and masks respectively. And eliminating background patches along with K-mean smoothing for the model input. The designed custom generators for images and masks were then used to load the data into the model. We used U-net architecture to design our model for the semantic segmentation task, forwarded with improvising the evaluation metrics with the help of hyper-parameters. With the help of our model, now it's possible to segment brain aneurysm histological images and identify the different tissue textures in images with less effort.

5.2 Limitation

One of the major limitations were ground-truth labeling for the given dataset. Manually drawing the ground truths was time-consuming and constant supervision and feedback is necessary from a subject expert to avoid errors. Due to limited knowledge in the domain, we might have miss labeled a few tissue textures. The given dataset was highly imbalanced, which led to poor learning of the model. We were able to design a model for the semantic segmentation problem statement handling the class imbalance problem. However, the results have a huge scope for improvements. Higher computational hardware is always helpful to achieve higher performance with a short time span.

5.3 Future Work

Ground truths of the dataset have to be acquired in close cooperation with medical experts. The proposed model can be further explored for optimum performance and better evaluation parameter results. Also, the run-time of the program could be reduced by employing GPU-based techniques. With good computational support. This developed model performance can be further explored with higher patch size input data. Transfer learning can be a better approach, that allows us to use the knowledge gained from other tasks to tackle new but similar problems quickly and effectively. Also building an individual segmentation model for each class would be promising future work. This could help specialize the model in segmenting each tissue texture from the dataset, also simplifies the model complexity.

Bibliography

[1] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR*, abs/1606.00915, 2016.

[2] Ahmed Elnakib, Georgy Gimel'farb, Jasjit Suri, and Ayman El-Baz. *Medical Image Segmentation: A Brief Survey*, pages 1–39. 04 2011.

[3] Alberto Garcia-Garcia, Sergio Orts-Escolano, Sergiu Oprea, Victor Villena-Martinez, and José García Rodríguez. A review on deep learning techniques applied to semantic segmentation. *CoRR*, abs/1704.06857, 2017.

[4] Foster DM. Jersey AM. Cerebral aneurysms. a service of the national library of medicine, national institute of health (ncbi bookshelf).treasure island: Stat pearls. 2019.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[6] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. *CoRR*, abs/1411.4038, 2014.

[7] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]).

[8] Seyed Sadegh Mohseni Salehi, Deniz Erdogmus, and Ali Gholipour. Tversky loss function for image segmentation using 3d fully convolutional deep networks. *CoRR*, abs/1706.05721, 2017.

[9] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832, 2015.

[10] F. Shi, Q. Yang, X. Guo, T. A. Qureshi, Z. Tian, H. Miao, D. Dey, D. Li, and Z. Fan. Intracranial vessel wall segmentation using convolutional neural networks. *IEEE Transactions on Biomedical Engineering*, 66(10):2840–2847, 2019.

[11] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[12] C. Szegedy, Wei Liu, Yangqing Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–9, 2015.

[13] Dayong Wang, Aditya Khosla, Rishab Gargeya, Humayun Irshad, and Andrew H. Beck. Deep learning for identifying metastatic breast cancer, 2016.