

**REPUBLIC OF TURKEY
YILDIZ TECHNICAL UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING**



TISSUE LAYERS SECTION SEGMENTATION

17011901 – Muhammed Johar
18011113 – Giyat MOSA

SENIOR PROJECT

Advisor
Prof. Songül VARLI

May, 2023

ACKNOWLEDGEMENTS

We would like to extend our heartfelt appreciation and gratitude to all those who have contributed to the success of this project. Firstly, we would like to express our sincere thanks to our consultant, Prof. Songül VARLI, for her invaluable guidance and support throughout the project. Her expertise and insightful feedback have greatly contributed to the quality and accuracy of our work.

We would also like to acknowledge the significant contributions of Yasemin TOPUZ. His expertise in the field has been instrumental in our project's success, and we are grateful for his valuable insights and assistance.

Muhammed Johar
Giyat MOSA

TABLE OF CONTENTS

LIST OF SYMBOLS	v
LIST OF FIGURES	vi
LIST OF TABLES	viii
ABSTRACT	ix
1 Introduction	1
2 Preview	2
2.1 Hematoxylin and Eosin (H&E) stain	2
2.2 Histology (The Study of Tissues)	2
2.2.1 The skin	3
3 Feasibility	4
3.1 Technical feasibility	4
3.1.1 <i>Software feasibility</i>	4
3.1.2 <i>Hardware feasibility</i>	4
3.2 Communication feasibility	5
3.3 Legal feasibility	5
3.4 Economic feasibility	5
3.5 Time feasibility	6
4 System Analysis	8
4.1 Dataset	8
4.1.1 Annotation and Convert XML files to Mask Files	10
4.1.2 Patching WSI Images / Masks	10
4.2 Training Dataset	10
5 System Design	12
5.1 U-NET	12
5.2 Model Architecture	13
5.3 Model Training	15

5.4 Post-Processing the Predicted Mask for ROI Images	15
6 Implementation	18
6.1 U-net Results	18
6.1.1 Training Results for Unet (0,100)	18
6.1.2 Training Results for Unet (3,98)	18
6.1.3 Training Results for Unet (20,90)	19
6.1.4 Training Results for Unet (50,100)	19
6.1.5 Test Results for U-net Models	20
6.1.6 Discussion of U-net Results	20
6.2 U-net with Pretrained backbones Models Results	21
6.2.1 Training Results for VGG-Unet (20,90)	21
6.2.2 Training Results for res-Unet (0,100)	21
6.2.3 Test Results for VGG-Unet and res-Unet Models	22
6.2.4 Discussion of Vgg-Unet and Res-Unet Models Results	22
6.3 Test Results on ROI from Test Dataset	22
6.3.1 Image 12-1 IOU results	23
6.3.2 Image 20-2 IOU results	29
6.3.3 Image 20-3 IOU results	31
6.4 Test Results on ROI from External Dataset	34
6.4.1 Image 121 IOU results	34
6.4.2 Image 356 IOU results	36
6.4.3 Image 358 IOU results	38
References	41

LIST OF SYMBOLS

H&E stain	Hematoxylin and Eosin stain
CAD	Computer Aided Diagnostic System
WSI	Whole Slide Imaging
CNN	Convolutional Neural Network

LIST OF FIGURES

Figure 2.1	Hematoxylin and Eosin (H&E) stain	2
Figure 2.2	Human Skin Layers	3
Figure 3.1	Gantt chart	7
Figure 4.1	epidermis layer	8
Figure 4.2	The General Workflow of the Project	9
Figure 4.3	Masked and ROI Images	10
Figure 4.4	Patches Percentage (11022 patches) based on Epidermis Pixels Percentage in every patch after deleting the patches without Epidermis	11
Figure 5.1	U-Net Architecture	12
Figure 5.2	Model Architecture	14
Figure 5.3	Patches Percentage (11022 patches) based on Epidermis Pixels Percentage in every patch after deleting the patches without Epidermis	15
Figure 5.4	Workflow to eliminate false positive regions found deep inside the biopsy	16
Figure 5.5	Workflow of get-border function	17
Figure 6.1	results for Unet model with range (0, 100)	18
Figure 6.2	results for Unet model with range (3, 98)	19
Figure 6.3	results for Unet model with range (20, 90)	19
Figure 6.4	results for Unet model with range (50, 100)	20
Figure 6.5	results for VGG-Unet model with range (20, 90)	21
Figure 6.6	results for the res-Unet model with range (0, 100)	21
Figure 6.7	IOU result for U-net (0, 100) 	23
Figure 6.8	IOU result for U-net (3, 98) 	24
Figure 6.9	IOU result for U-net (20, 90) 	25
Figure 6.10	IOU result for U-net (50, 100) 	26
Figure 6.11	IOU result for VGG-Unet (20, 90) 	27
Figure 6.12	IOU result for res-Unet (0, 100) 	28
Figure 6.13	IOU result for U-net (0, 100) 	29
Figure 6.14	IOU result for U-net (3, 98) 	29

Figure 6.15 IOU result for U-net (20, 90)		29
Figure 6.16 IOU result for U-net (50, 100)		30
Figure 6.17 IOU result for VGG-Unet (20, 90)		30
Figure 6.18 IOU result for res-Unet (0, 100)		30
Figure 6.19 IOU result for U-net (0, 100)		31
Figure 6.20 IOU result for U-net (3, 98)		31
Figure 6.21 IOU result for U-net (20, 90)		32
Figure 6.22 IOU result for U-net (50, 100)		32
Figure 6.23 IOU result for VGG-Unet (20, 90)		33
Figure 6.24 IOU result for res-Unet (0, 100)		33
Figure 6.25 IOU result for U-net (0, 100)		34
Figure 6.26 IOU result for U-net (3, 98)		35
Figure 6.27 IOU result for U-net (20, 90)		35
Figure 6.28 IOU result for U-net (50, 100)		35
Figure 6.29 IOU result for VGG-Unet (20, 90)		36
Figure 6.30 IOU result for res-Unet (0, 100)		36
Figure 6.31 IOU result for U-net (0, 100)		36
Figure 6.32 IOU result for U-net (3, 98)		37
Figure 6.33 IOU result for U-net (20, 90)		37
Figure 6.34 IOU result for U-net (50, 100)		37
Figure 6.35 IOU result for VGG-Unet (20, 90)		38
Figure 6.36 IOU result for res-Unet (0, 100)		38
Figure 6.37 IOU result for U-net (0, 100)		38
Figure 6.38 IOU result for U-net (3, 98)		39
Figure 6.39 IOU result for U-net (20, 90)		39
Figure 6.40 IOU result for U-net (50, 100)		39
Figure 6.41 IOU result for VGG-Unet (20, 90)		40
Figure 6.42 IOU result for res-Unet (0, 100)		40

LIST OF TABLES

Table 3.1	Software options	4
Table 3.2	Hardware requirements for a healthy system operation	5
Table 3.3	Expenditures table	5
Table 6.1	test results of the U-net models for every slide	20
Table 6.2	test results of the U-net models for every slide	22
Table 6.3	Mean IOU Results Before and After Processing for Test Dataset's ROIs	22
Table 6.4	Mean IOU Results Before and After Processing for External Dataset's ROIs	34

ABSTRACT

TISSUE LAYERS SECTION SEGMENTATION

Muhammed Johar

Giyat MOSA

Department of Computer Engineering
Senior Project

Advisor: Prof. Songül VARLI

Skin diseases such as Cancers and Tumors could be dangerous and aggressive. Applying early diagnostics can prevent catastrophic situations for the patient. Manual diagnostics is done by applying Histopathological examination of Hematoxylin and Eosin stain (H&E stain) on biopsies taken from the patient to take high-definition microscopic slides of the skin cells and with the capability of obtaining those slides in digital form as images to be examined by specialists.

Computer Aided Diagnostic (CAD) system can be used to automate the examination part of the diagnostic process. This means speeding up the process with consistent results obtained for the specialists to decide and start with the most effective treatment options as soon as possible.

Automated Epidermis Segmentation Algorithm, one of the initial steps in a Computer Aided Diagnostic (CAD) system.

In our project, we develop a model of Semantic segmentation to Epidermis tissue cells from the Non-Epidermis cells and white background using Convolutional Neural Network (CNN) with U-net architecture

Keywords: Skin Cancers, Skin Tumors, Histology, Hematoxylin and Eosin stain (H&E stain), Computer Aided Diagnostic (CAD), Whole Slide Imaging (WSI), Semantic Segmentation, Neural Network, CNN, U-Net

1

Introduction

Computer-aided diagnosis (CAD) is a technology used in the medical field to take advantage of the calculation power of computers as a tool in interpreting medical images to help in diagnosing diseases. CAD can be used for various medical applications, such as detecting cancer, heart disease, and other medical conditions.

Automated epidermis segmentation is one of the many applications included in CAD systems that have shown good results in improving the diagnosis and treatment of skin diseases.

In medical imaging, automated epidermis segmentation is a great tool for diagnosing skin diseases such as melanoma and psoriasis. It can also be used in cosmetic analysis to review the results taken from skincare products.

The operation of segmentation is typically used to specify the region between the epidermis and the underlying dermis layer. This can be applied using many techniques such as normal techniques like edge detection and thresholding. Machine learning algorithms are also used to classify the pixels in images as epidermis pixels or not.

Our project tends to create a model using U-Net architecture for the auto segment of the epidermis regions in WSI images The Cancer Genome Atlas (TCGA) Repository. In the preprocessing step, we will carefully select 50 whole slide images(WSIs) and we have to do annotations, masks, and patches for each slide before implementing the U-Net model.

2

Preview

In this chapter, we will have a quick preview of the medical concepts of the project.

2.1 Hematoxylin and Eosin (H&E) stain

Hematoxylin and Eosin (H&E) stain is a commonly used staining method to visualize the structure of tissues and cells. Hematoxylin stains the acidic structures of cells, such as the nuclei and ribosomes in blue or purple. Eosin stains the basic structures, such as the cytoplasm and extracellular matrix in pink or red. Figure 2.1.

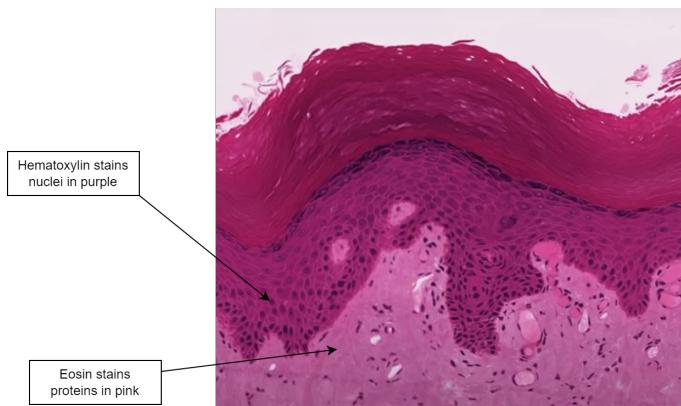


Figure 2.1 Hematoxylin and Eosin (H&E) stain

2.2 Histology (The Study of Tissues)

Histology is the study of the structure of tissues using microscopic tools. It contains a check of cells and tissues by using specified methods to realize the various layers of the tissue and their structure.[1]. Type of Tissues: The four primary types of tissues in histology are:

1. **Epithelial tissue.**
2. Connective tissue.

3. Muscular tissue.

4. Nervous tissue.

2.2.1 The skin

The skin, the largest organ in the human body, is composed of three primary tissue layers: the epidermis, dermis, and hypodermis(subcutaneous tissue).

1. Epidermis: This is the outermost layer of the skin. It provides a barrier against the outside environment.
2. Dermis: The second layer of skin, contains the blood vessels and the terminal nerve. The dermis layer involves many sublayers such as the papillary and the reticular layer.
3. Hypodermis (Subcutaneous Tissue): This is the deepest layer of the skin and is composed of adipose tissue and connective tissue such as blood vessels and nerves.

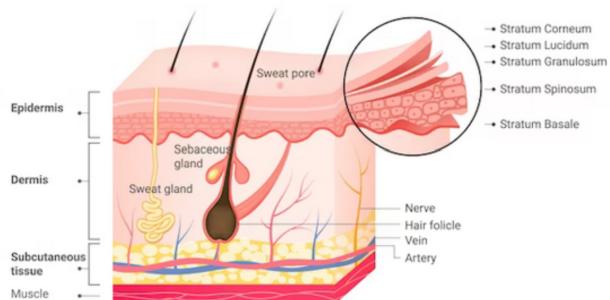


Figure 2.2 Human Skin Layers

3

Feasibility

This section outlines the feasibility study conducted during the project's development process.

3.1 Technical feasibility

3.1.1 Software feasibility

Table 3.1 provides an explanation of the operating systems and development environments.

Table 3.1 Software options

	Option 1	Option 2
OS	MacOS	Windows 11
Language	C++	Python
IDE	Visual Studio Code	Google Colab/Kaggle Notebook

Python will be used for the project due to its user-friendly nature and the availability of high-performance libraries capable of executing intricate mathematical operations. To avoid confusion among our team, Windows 11 was chosen as the operating system. the model can be trained on a local machine or in a Google Colab/Kaggle environment.

3.1.2 Hardware feasibility

Training a CNN is a computationally demanding task, therefore it is advisable to use a machine equipped with a dedicated GPU for optimal performance. Nevertheless, after obtaining a trained model, the model is expected to function smoothly on any average computer. 3.2.

Table 3.2 Hardware requirements for a healthy system operation

	Option 1	Option 2
Processor	Core i7-7700HQ	Core i7-7820HQ
RAM	16 GB	16 GB
Storage	1 TB GB	512 GB
GPU	Nvidia GeForce GTX 1050	AMD Radeon Pro 560

3.2 Communication feasibility

Git along with GitLab is used for project collaboration, version control, and source control. Also, online virtual meetings are conducted using Zoom software and Google Meet.

3.3 Legal feasibility

The project adheres to all relevant laws and regulations, respects existing patents, and does not infringe on any protected rights.

3.4 Economic feasibility

Table 3.3 displays the expenses incurred on behalf of the project.

Table 3.3 Expenditures table

	amount	Price per unit	Total price	Brand
Computer 1	1	10000 TL	10000 TL	ASUS
Computer 2	1	15000 TL	15000 TL	MacBook Pro 2017
Software Developer Expense	2	500 TL (daily)	86,000 TL	-
Total	-	-	111000 TL	-

Note: Since some development software and platforms are either open-source or provided through student licenses, there has been no expenditure on behalf of the software.

3.5 Time feasibility

The design and implementation process of the project is as specified in the Gantt chart in Figure 3.1.

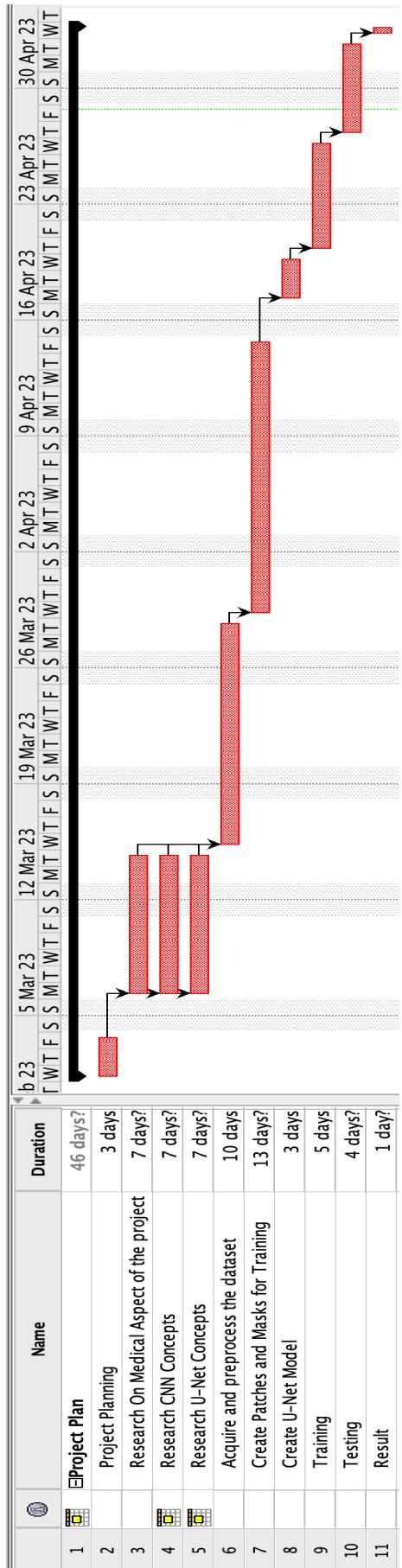


Figure 3.1 Gantt chart

4

System Analysis

In this chapter, the following tasks of identifying the problems, the goals, and data collecting will be discussed.

As shown in Figure 4.1 a biopsy of skin contains 3 layers of tissues which are subcutaneous tissue, dermis, and epidermis. Our goal is to design such a system that creates a limiting line around the epidermis layer and specifies the classes of the epidermis.

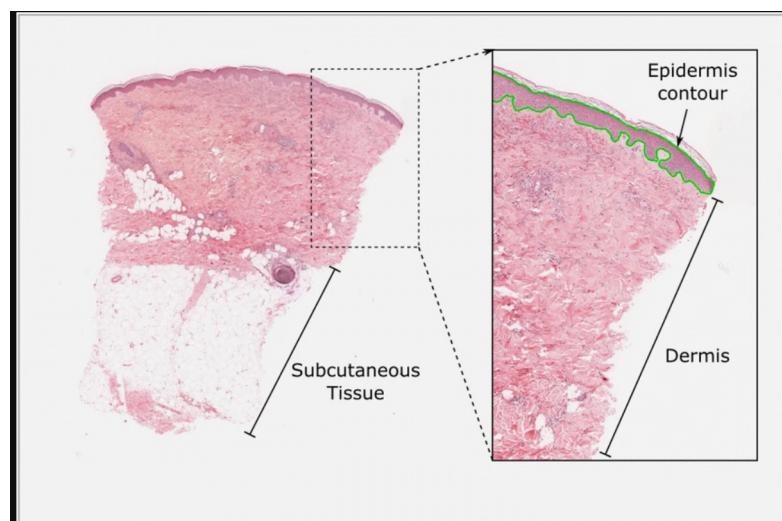


Figure 4.1 epidermis layer

4.1 Dataset

The dataset for training and testing is taken from the GDC National Cancer Institute dataset repository[2] which contains about 900 unlabeled WSI we carefully select from them 50 WSI images with a total size of 42.6 GB. The next step is to annotate those 50 WSI Images to obtain labeled images ready to be used in supervised learning models.

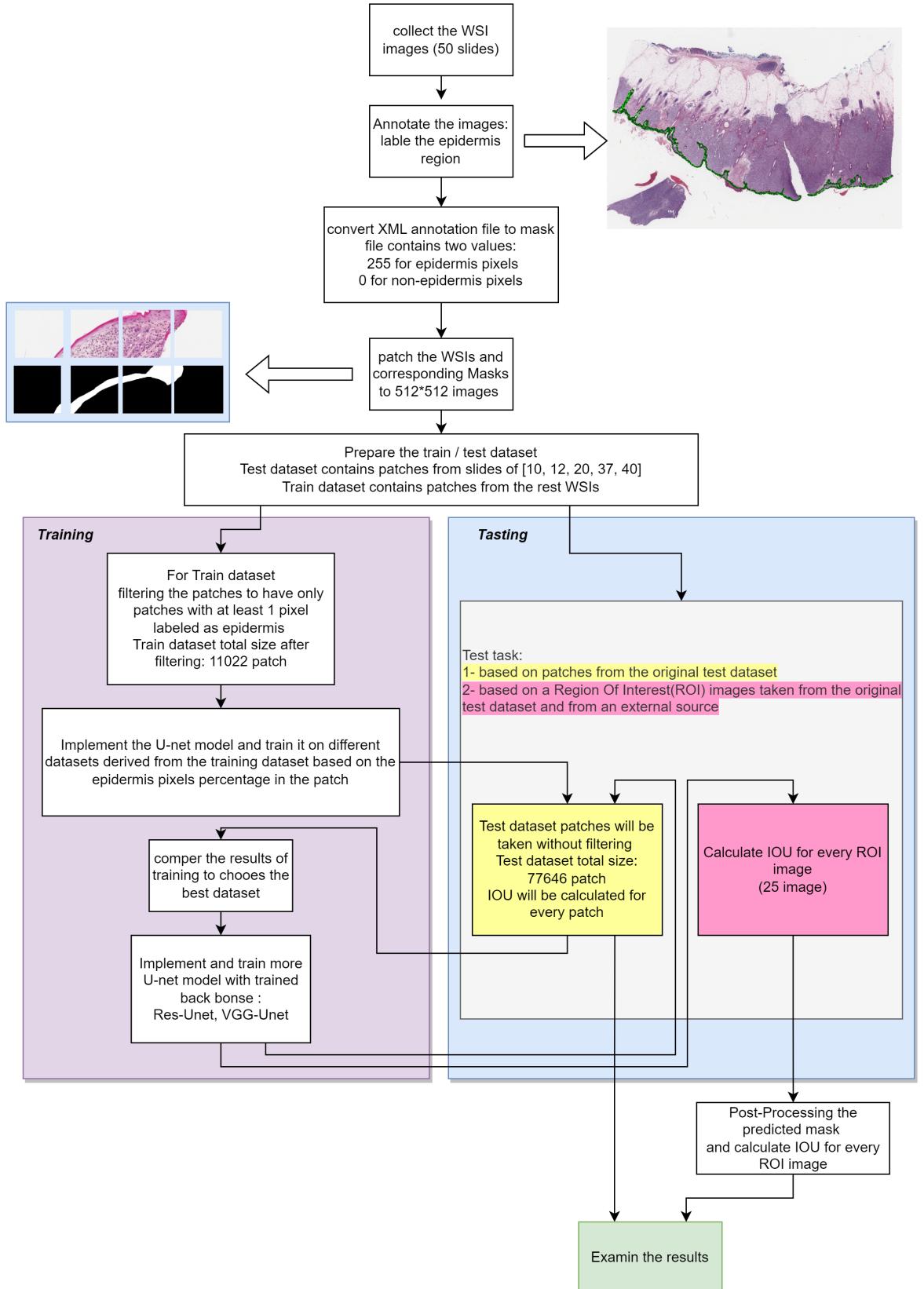


Figure 4.2 The General Workflow of the Project

4.1.1 Annotation and Convert XML files to Mask Files

The annotation step is done by using the Sedeen Viewer Application which gives us the annotation for every WSI image as an XML file, by using the points information of the annotation we construct a mask image that contains two values 255 corresponds to epidermis pixels and 0 for non-epidermis pixels. the result is shown in Figure 6.42.

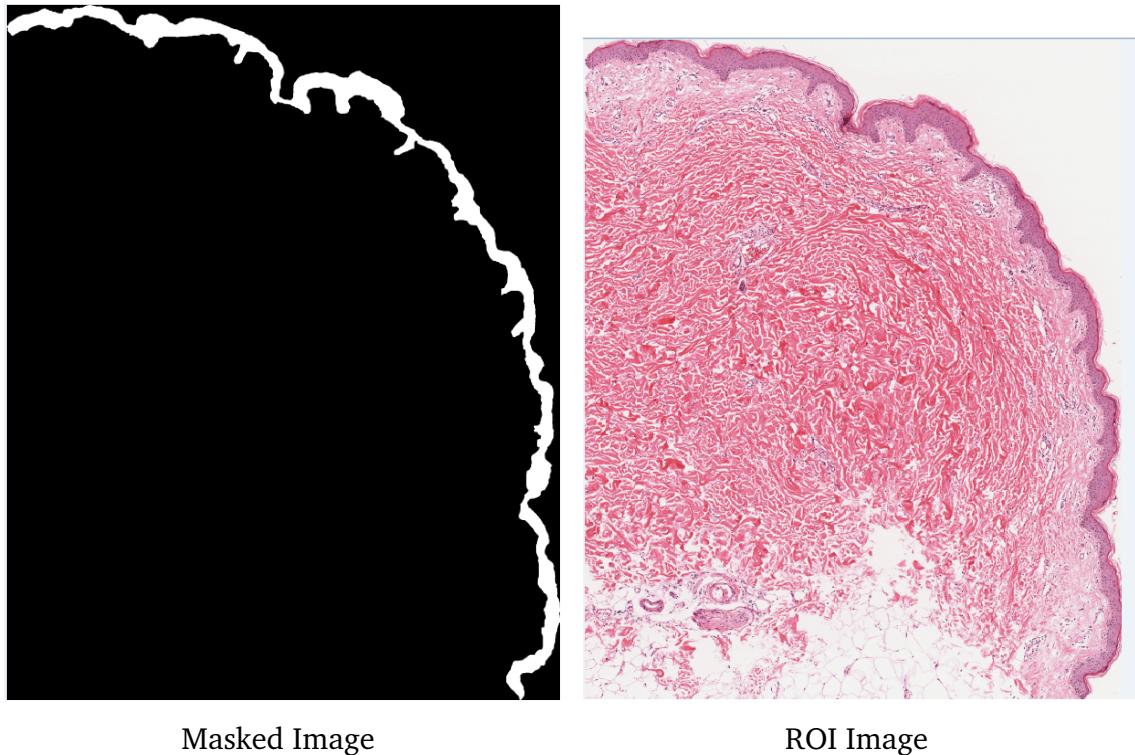


Figure 4.3 Masked and ROI Images

4.1.2 Patching WSI Images / Masks

After obtaining the corresponding mask for every WSI image, Those Images and their masks must be patched to 512*512 image patches to be able to feed them to our model

4.2 Training Dataset

After preserving 5 WSI images for testing, the rest WSI Images give us a total of 580033 patches. After analyzing the training patches, about 85% of the patches contained only background pixels (the whole mask contains value 0 pixels), which will give us an unbalanced training dataset, for this reason, we applied filtering to obtain only images that contain at least one white (255) pixel. The training dataset size after applying the filter is 11022 patches.

Figure 5.3 shows the percentage distribution of the patches based on the percentage

of epidermis pixels for each patch.

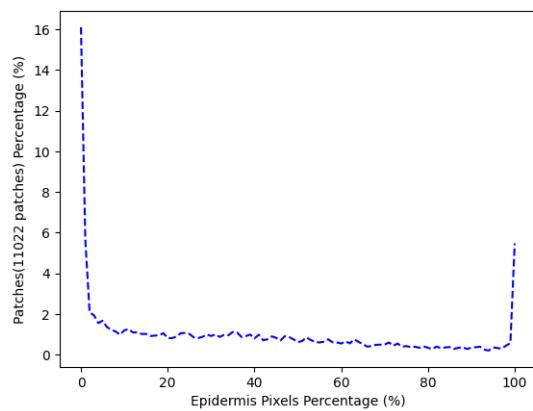


Figure 4.4 Patches Percentage (11022 patches) based on Epidermis Pixels Percentage in every patch after deleting the patches without Epidermis

5 System Design

5.1 U-NET

The architecture shows that such a network can be trained end-to-end from very few images and beat the main best method (a sliding-window convolutional network) on the challenge of the segmentation of neuronal structures in microscopic images. Moreover, the network is fast. Segmentation of a 512x512 image takes less than a second on a recent GPU[3] Figure5.1

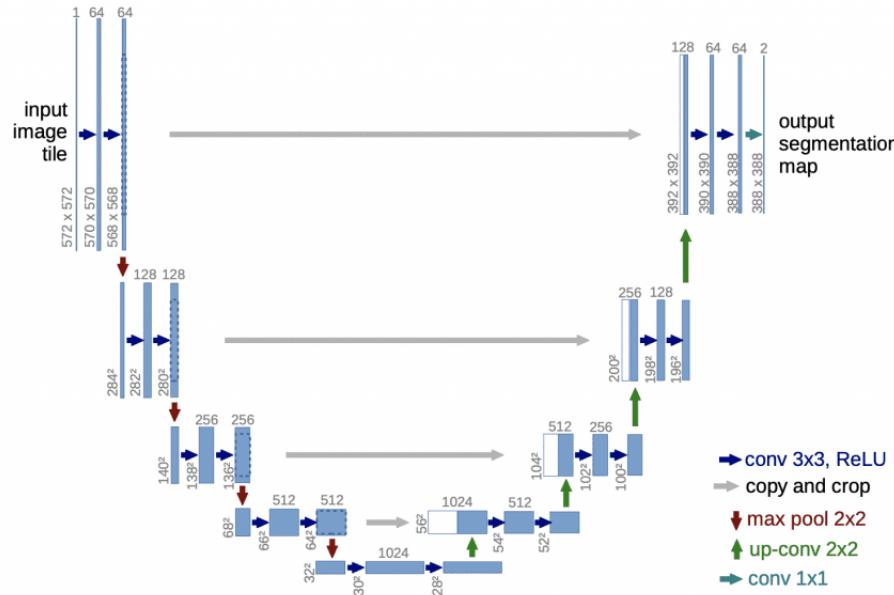


Figure 5.1 U-Net Architecture

5.2 Model Architecture

In this section we explain the main steps that we applied to create our model, the model was built with 6 steps with the help of Unet architecture:

1. Input: Images with shape (512, 512, 3), these images represent the patches that we take from WSI images.
2. Downsampling: The model starts to sort the layers taken after by the max pooling operation. In this layer, we make a calculation of images and increase the number of features.
3. Bottleneck: This layer appears after more steps of Downsampling, this layer includes the convolutional layers to make more unique features.
4. Upsampling: In this step, the model provides upsampling to remake the calculation of the included images. It uses convolutional transpose to unsample the feature maps and link them with the comparing feature maps from the downsampling.
5. Classification: After performing the upsampling procedure, the model adds more convolutional layers in order to refine the features. To create the output segmentation mask, a convolutional layer a sigmoid activation function is used. The output object including each pixel is indicated that makes up the final mask.
6. Model Creation: Finally, the model was created with the help of Keras and Tensorflow APIs, specifying the mentioned steps above.

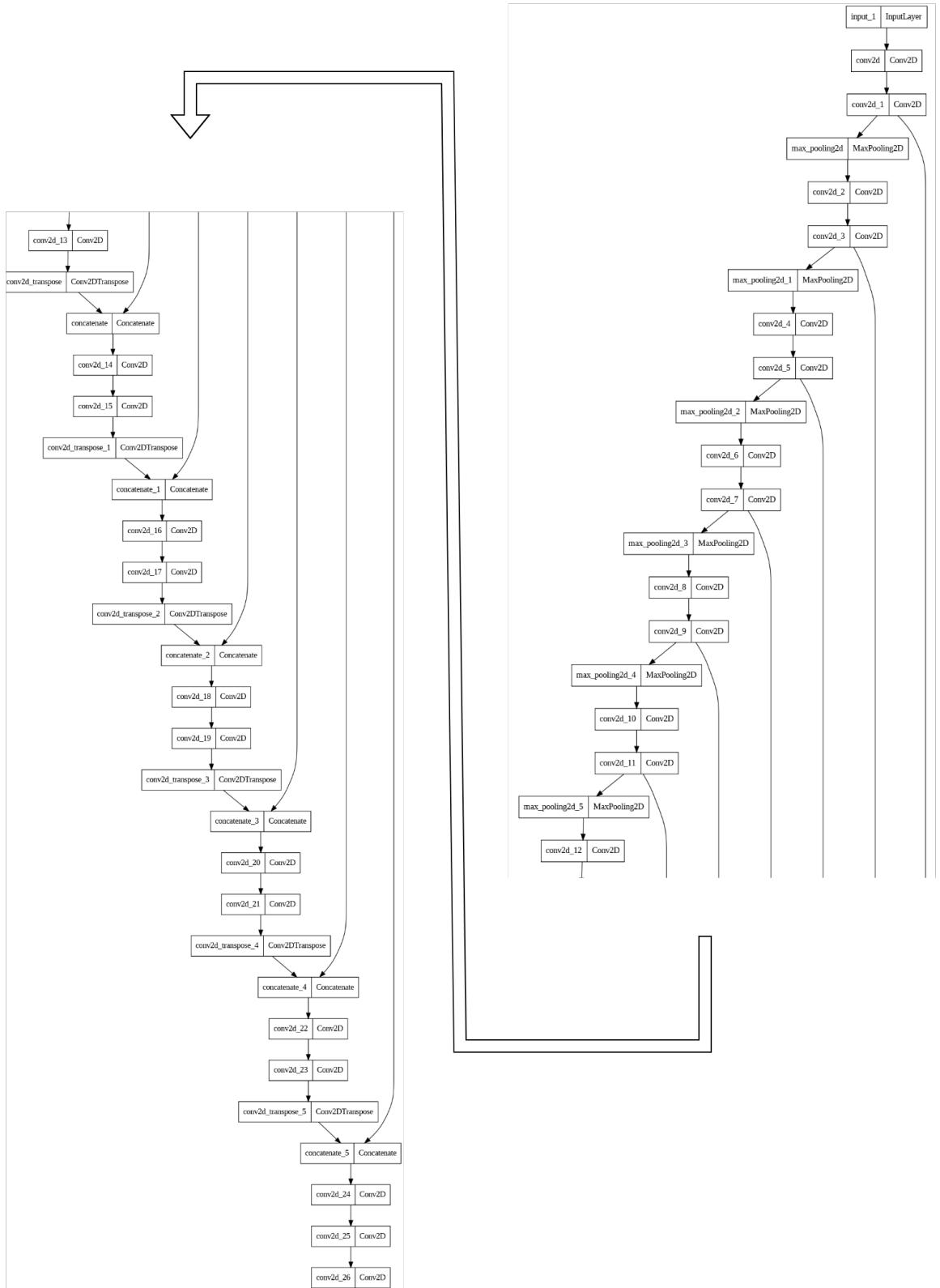


Figure 5.2 Model Architecture

5.3 Model Training

After implementing the basic U-net architecture we decide to train the model with several ranges derived from the filtered dataset based on the percentage of the epidermis in every patch, and we trained the model in four different ranges as the following:

1. Dataset with range (0, 100) of epidermis percentage in every patch.
2. Dataset with range (3, 980) of epidermis percentage in every patch.
3. Dataset with range (20, 90) of epidermis percentage in every patch.
4. Dataset with range (50, 100) of epidermis percentage in every patch

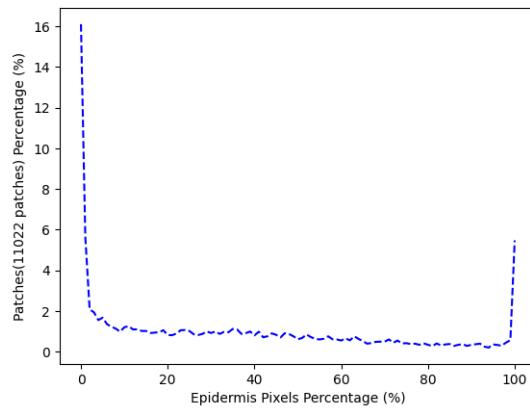


Figure 5.3 Patches Percentage (11022 patches) based on Epidermis Pixels Percentage in every patch after deleting the patches without Epidermis

Based on the testing result we will choose the range size to train another two U-NET models with different pre-trained backbones.

5.4 Post-Processing the Predicted Mask for ROI Images

After obtaining the mask for ROI images we decided to apply the morphological operation on the predicted mask to eliminate the effect of false positive value in the predicted mask. The figure 5.4 and figure 5.5 show the post-processing flow.

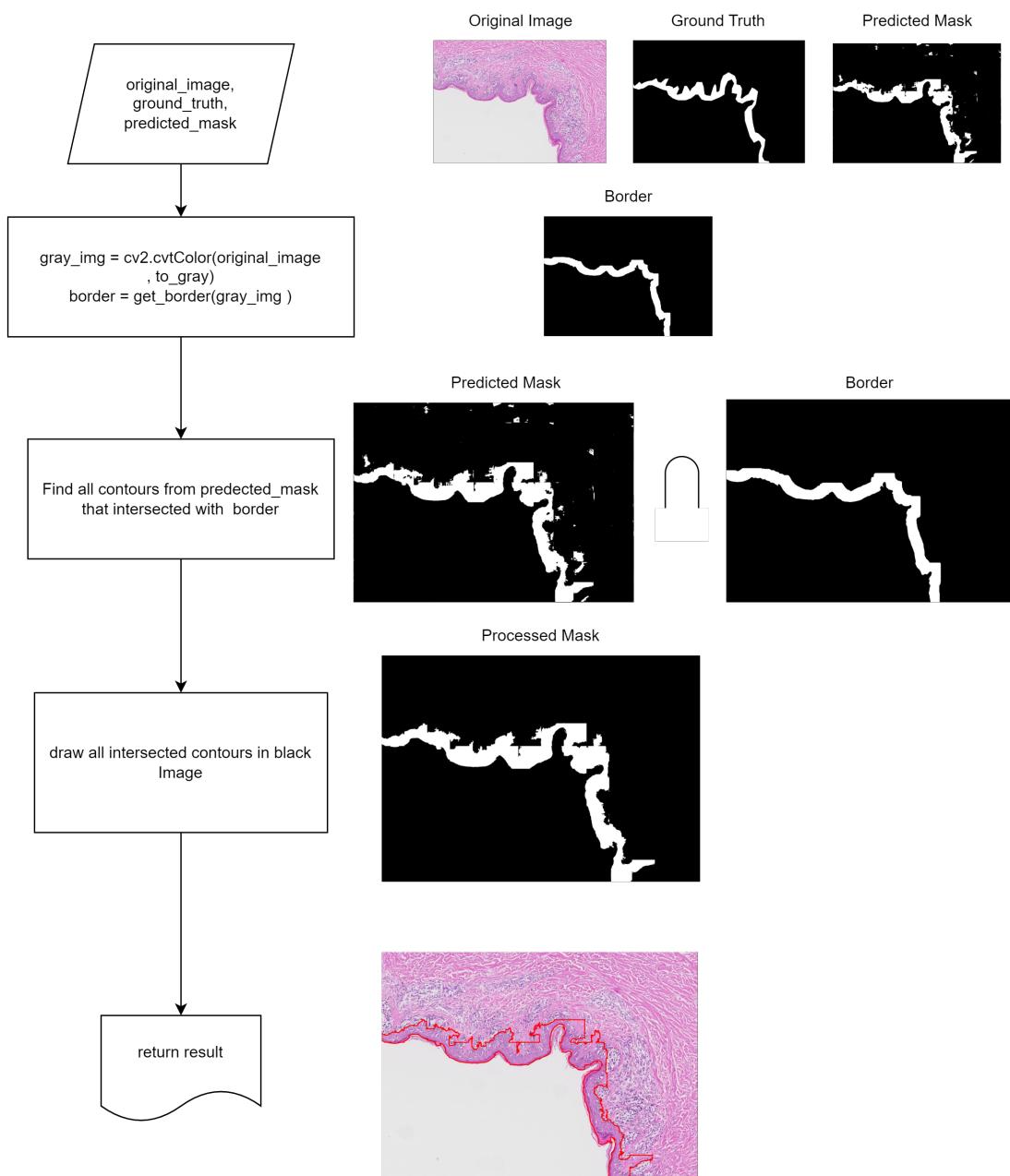


Figure 5.4 Workflow to eliminate false positive regions found deep inside the biopsy

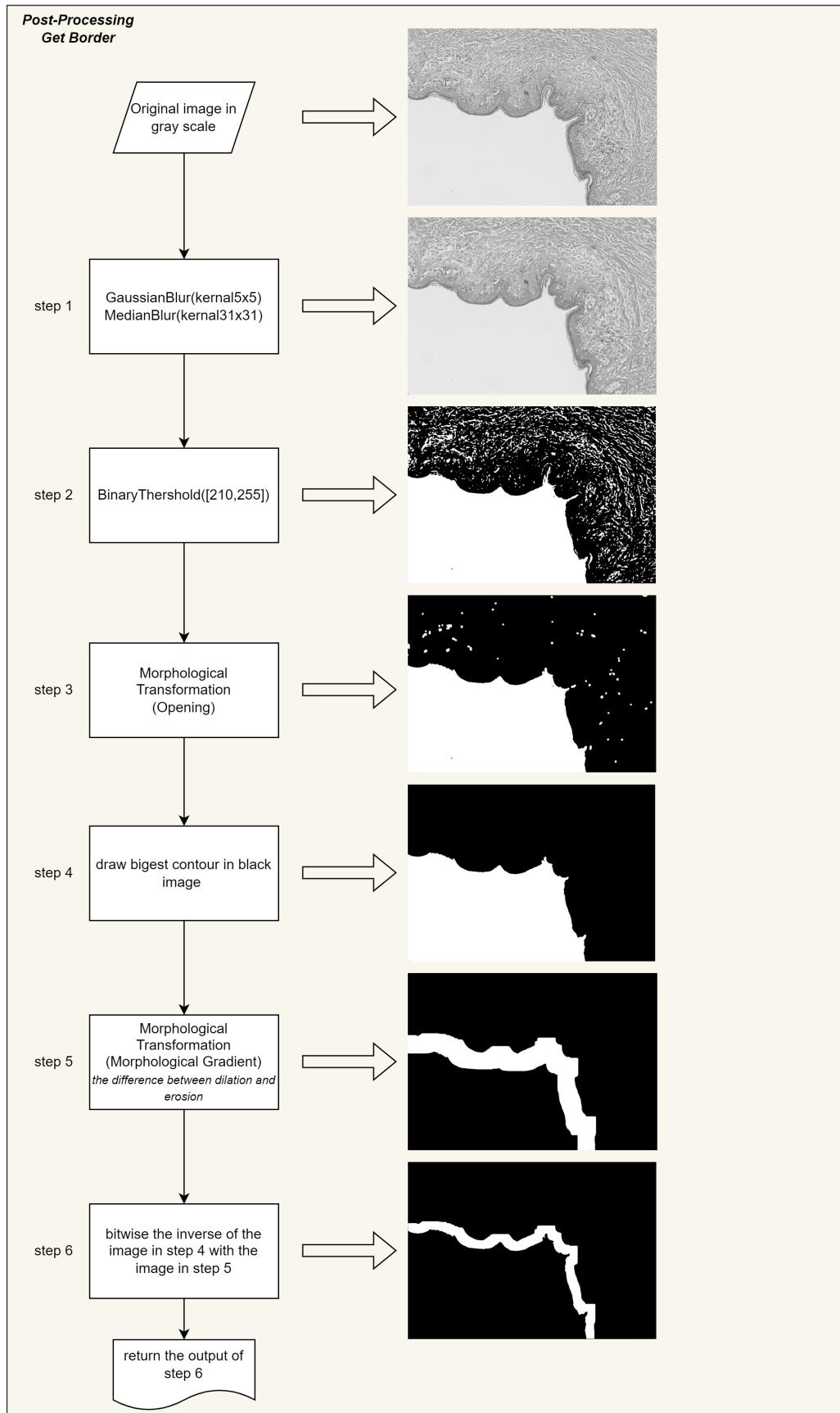


Figure 5.5 Workflow of get-border function

6

Implementation

In this chapter, we show the training results and discuss the results

6.1 U-net Results

the results of training the U-net model with 4 different datasets are as the following:

6.1.1 Training Results for Unet (0,100)

Figure 6.1 shows the train history for U-net with range (0,100), trained 60 epochs, optimizer rmsprop, and loss function binary_crosentropy

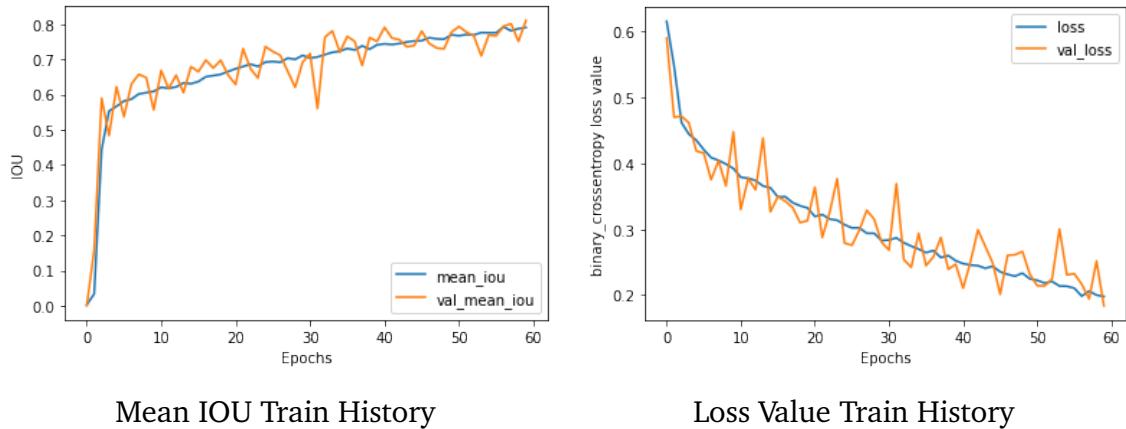


Figure 6.1 results for Unet model with range (0, 100)

6.1.2 Training Results for Unet (3,98)

Figure 6.2 shows the train history for U-net with range (3,98), trained 60 epochs, optimizer rmsprop, and loss function binary_crosentropy

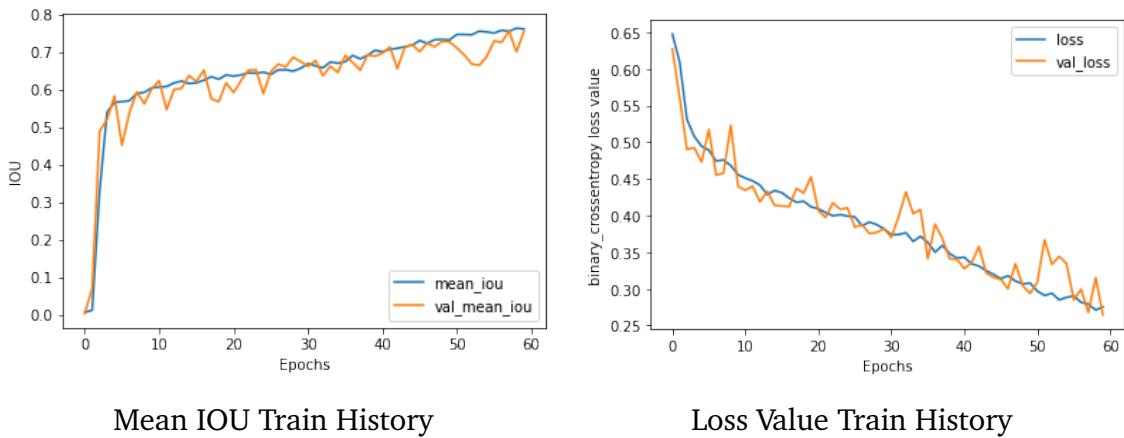


Figure 6.2 results for Unet model with range (3, 98)

6.1.3 Training Results for Unet (20,90)

Figure 6.3 shows the train history for U-net with range (20,90), trained 60 epochs, optimizer rmsprop, and loss function binary_crossentropy

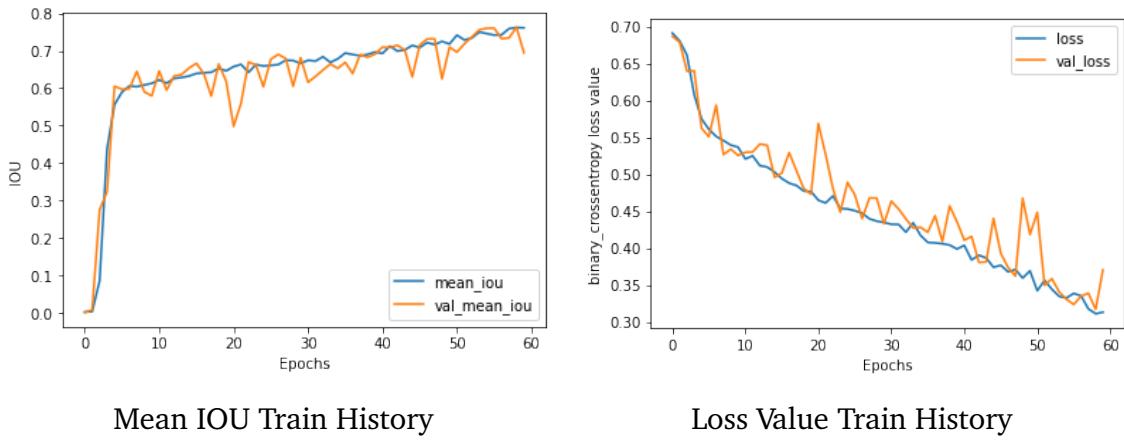


Figure 6.3 results for Unet model with range (20, 90)

6.1.4 Training Results for Unet (50,100)

Figure 6.4 shows the train history for U-net with range (50,100), trained 60 epochs, optimizer rmsprop, and loss function binary_crossentropy

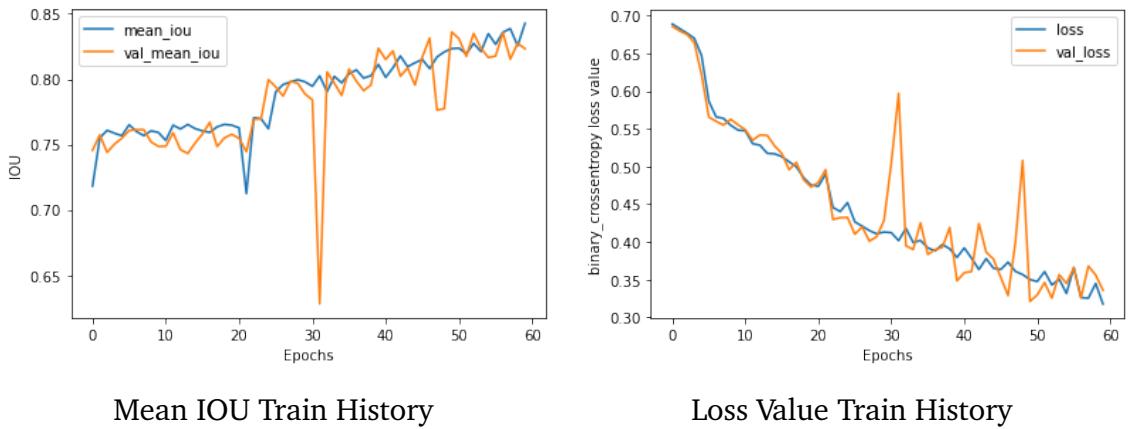


Figure 6.4 results for Unet model with range (50, 100)

6.1.5 Test Results for U-net Models

After training U-net models we evaluated those models with 5 WSI images that never showed to the models in the training stage, the patch number for the test dataset is 77646 patches without any filtering to simulate real-world situations when there is no ground truth to use for filtering.

Table 6.1 test results of the U-net models for every slide

slide-no	U-Net (0, 100)	U-Net (3, 98)	U-Net (20, 90)	U-Net (50, 100)
10	0.204488	0.120543	0.145777	0.027739
12	0.115882	0.103909	0.058755	0.006375
20	0.095401	0.077114	0.121916	0.013348
37	0.018658	0.014764	0.017252	0.007369
40	0.012592	0.014382	0.016892	0.006262
All	0.089404	0.066142	0.072118	0.012218

6.1.6 Discussion of U-net Results

Table 6.1 shows very low IOU results in comparison with training results. Keeping in mind that the epidermis pixels percentage in every slide doesn't exceed 1%-2% from the whole slide we can see that the false positive value is very high.

Since the models trained on filtered data shows us that the models predict non-epidermis tissues as the epidermis, and this result is reasonable because the u_net model doesn't have any idea about the location of the input patch in the WSI image so any tissues that have the same texture as the epidermis will be falsely predicted as epidermis tissues.

The suggested solution to this problem is to produce a new class for non-epidermis

tissues 3 classes for epidermis tissues, and non-epidermis tissues, and white background.

6.2 U-net with Pretrained backbones Models Results

The best results were given by the U-net model trained in range (0, 100) and (20, 90), so we decided to train two other models:

6.2.1 Training Results for VGG-Unet (20,90)

Figure 6.5 shows the train history for VGG-Unet with range (20,90), trained 60 epochs, optimizer adam, and loss function loss_dice

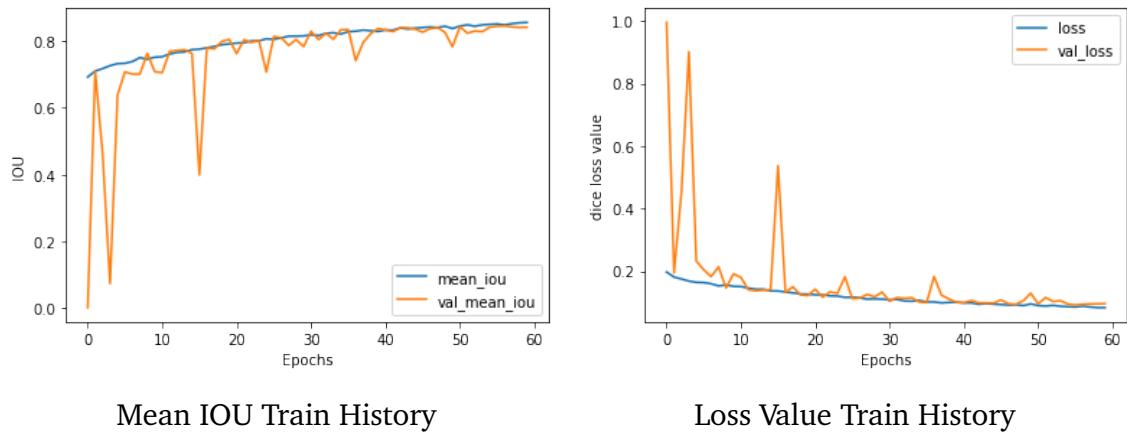


Figure 6.5 results for VGG-Unet model with range (20, 90)

6.2.2 Training Results for res-Unet (0,100)

Figure 6.6 shows the train history for res-Unet with range (0,100), trained 60 epochs, optimizer rmsprop, and loss function binary_crossentropy

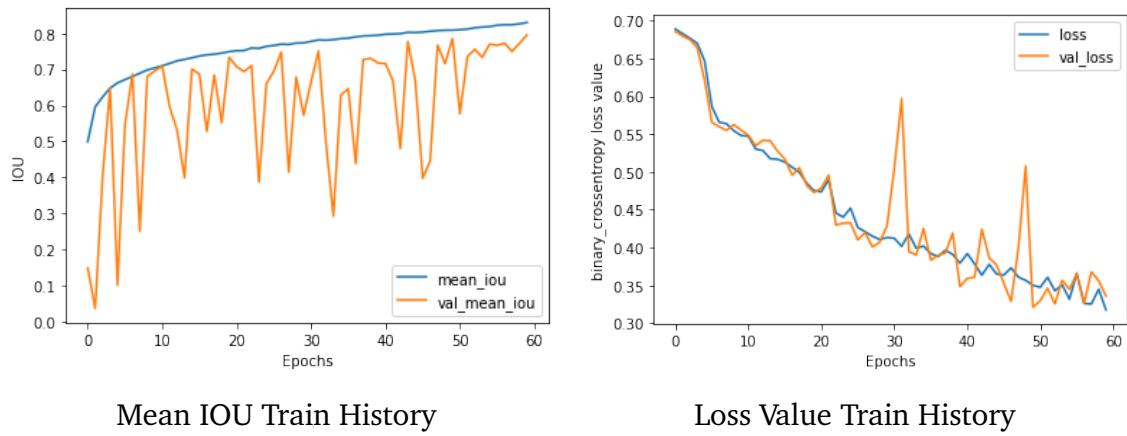


Figure 6.6 results for the res-Unet model with range (0, 100)

6.2.3 Test Results for VGG-Unet and res-Unet Models

The result of evaluating VGG-unet and res-unet models with 5 WSI images that never showed to the models in the training stage, the patch number for the test dataset is 77646 patches without any filtering to simulate real-world situations when there is no ground truth to use for filtering.

Table 6.2 test results of the U-net models for every slide

slide-no	VGG-UNet (20, 90)	res-UNet (0, 100)
10	0.063121	0.132663
12	0.080115	0.101572
20	0.064229	0.060507
37	0.012805	0.023029
40	0.012401	0.021822
All	0.046535	0.067918

6.2.4 Discussion of Vgg-Unet and Res-Unet Models Results

Table 6.2 shows very low IOU results compared to training results. As expected in addition to the previous reason in U-net, the VGG-Unet model with about 25 million parameters and res-unet with about 8 million parameters compared to U-net with about 500 thousand parameters, the res-unet, and VGG-unet required more training. But because of the limitation of training in the Google Colab environment, the training session duration is 12 hours, we stopped training after 60 epochs.

6.3 Test Results on ROI from Test Dataset

Table 6.3 shows the IOU results on ROIs taken from the test dataset

Table 6.3 Mean IOU Results Before and After Processing for Test Dataset's ROIs

Model Name (Range)	IOU Before Post-Processing	IOU After Post-Processing
Unet (0,100)	0.75	0.78
Unet (3,98)	0.69	0.75
Unet (20,90)	0.57	0.59
Unet (50,100)	0.28	0.28
VGG_Unet(20,90)	0.70	0.73
Res_Unet(0,100)	0.77	0.77

To show the results per ROI we choose 3 ROI images taken from the test dataset, the first two digits of the name of the images correspond to the name of the slide third digit corresponds to the region number from the slide.

6.3.1 Image 12-1 IOU results

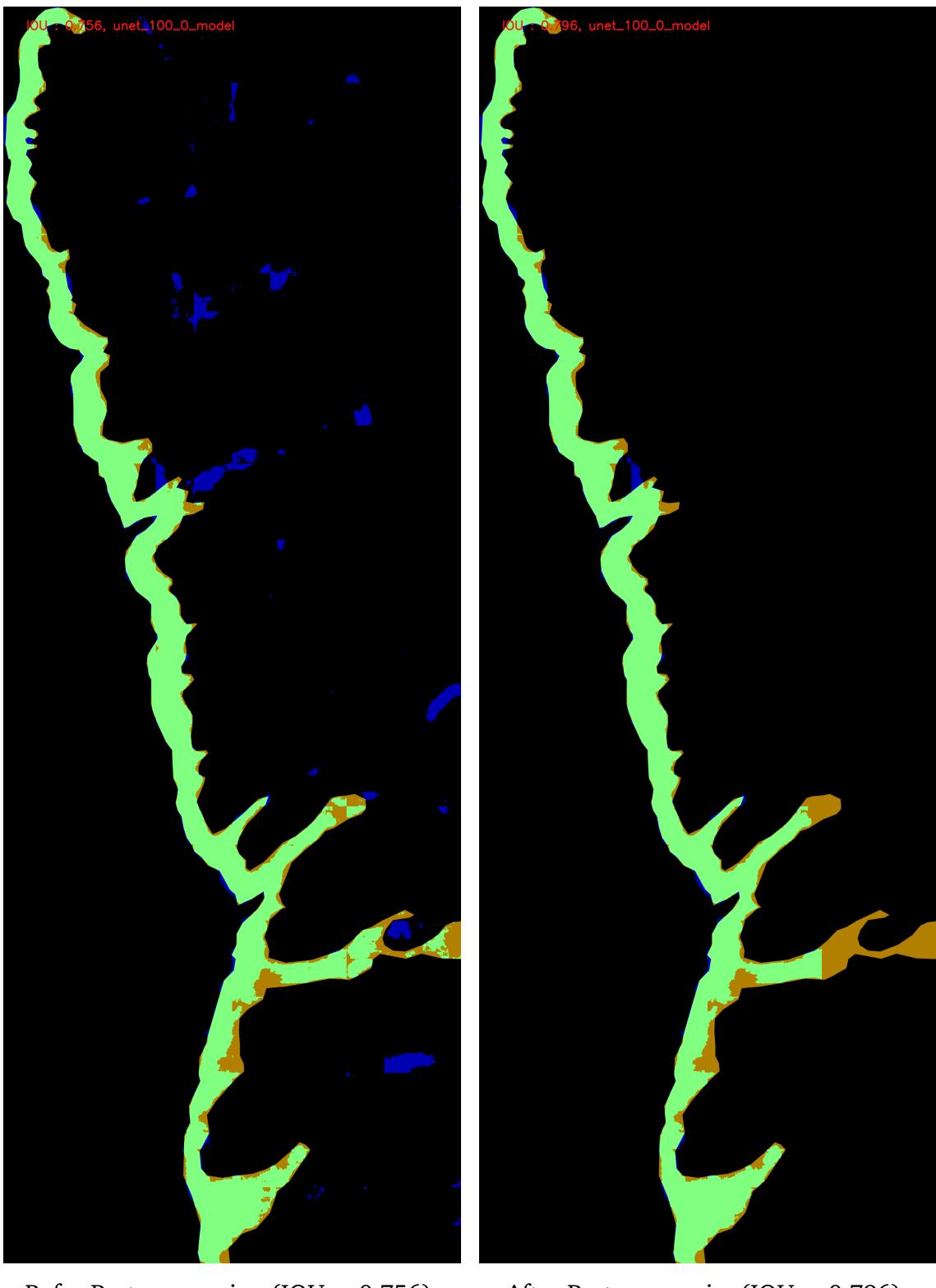
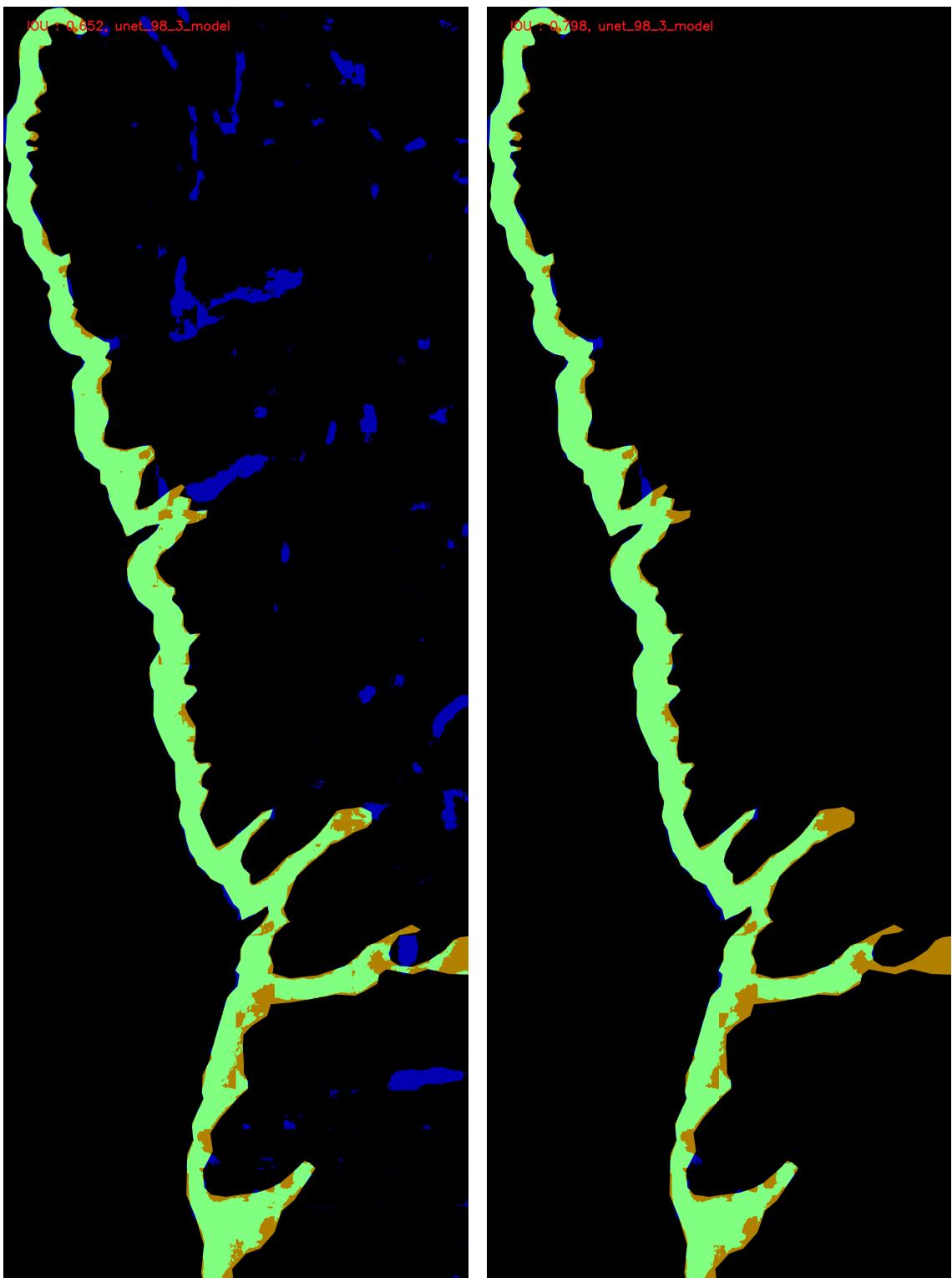


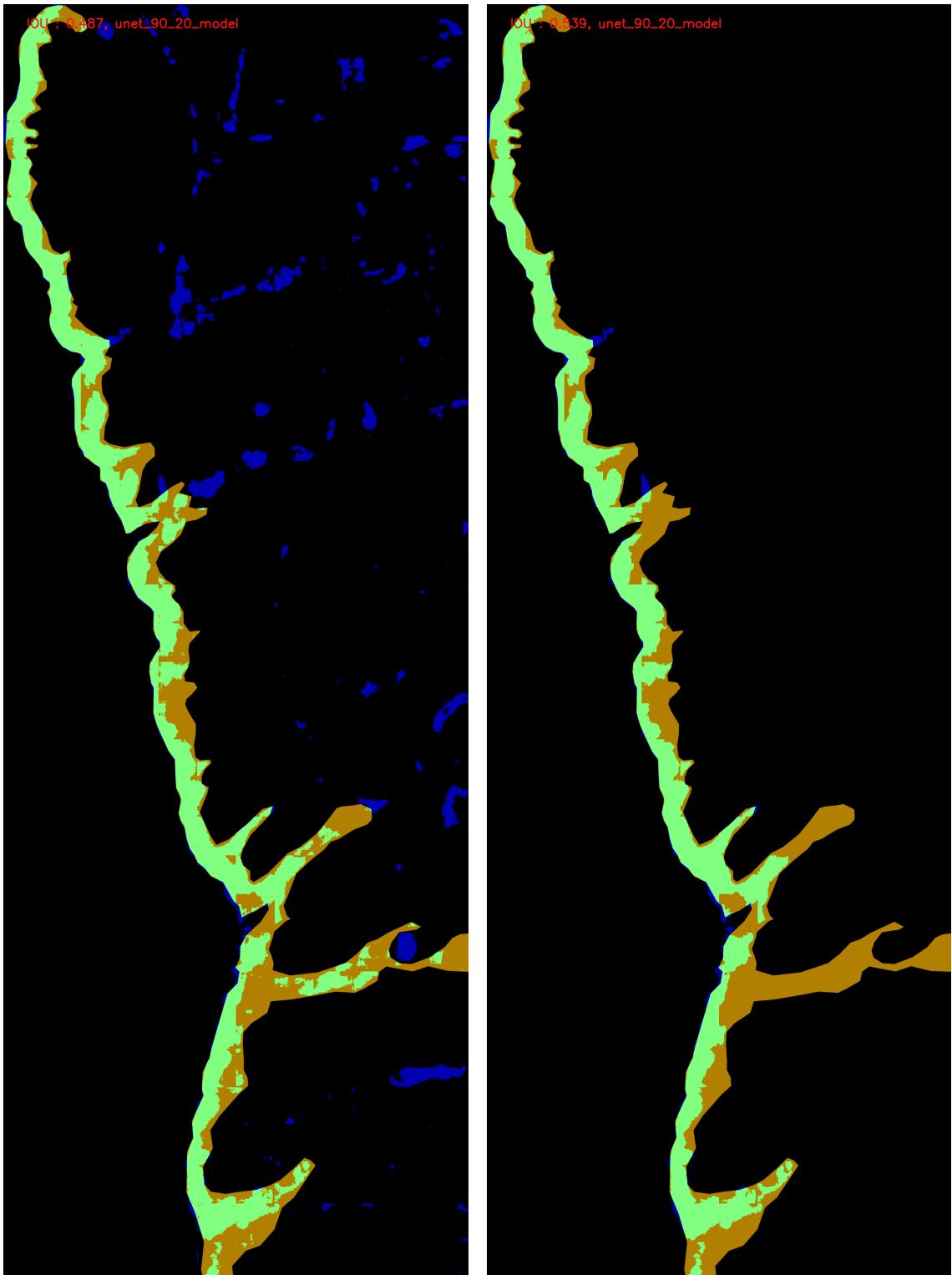
Figure 6.7 IOU result for U-net (0, 100) FP FN TP



Before Post_processing (IOU = 0.652)

After Post_processing (IOU = 0.798)

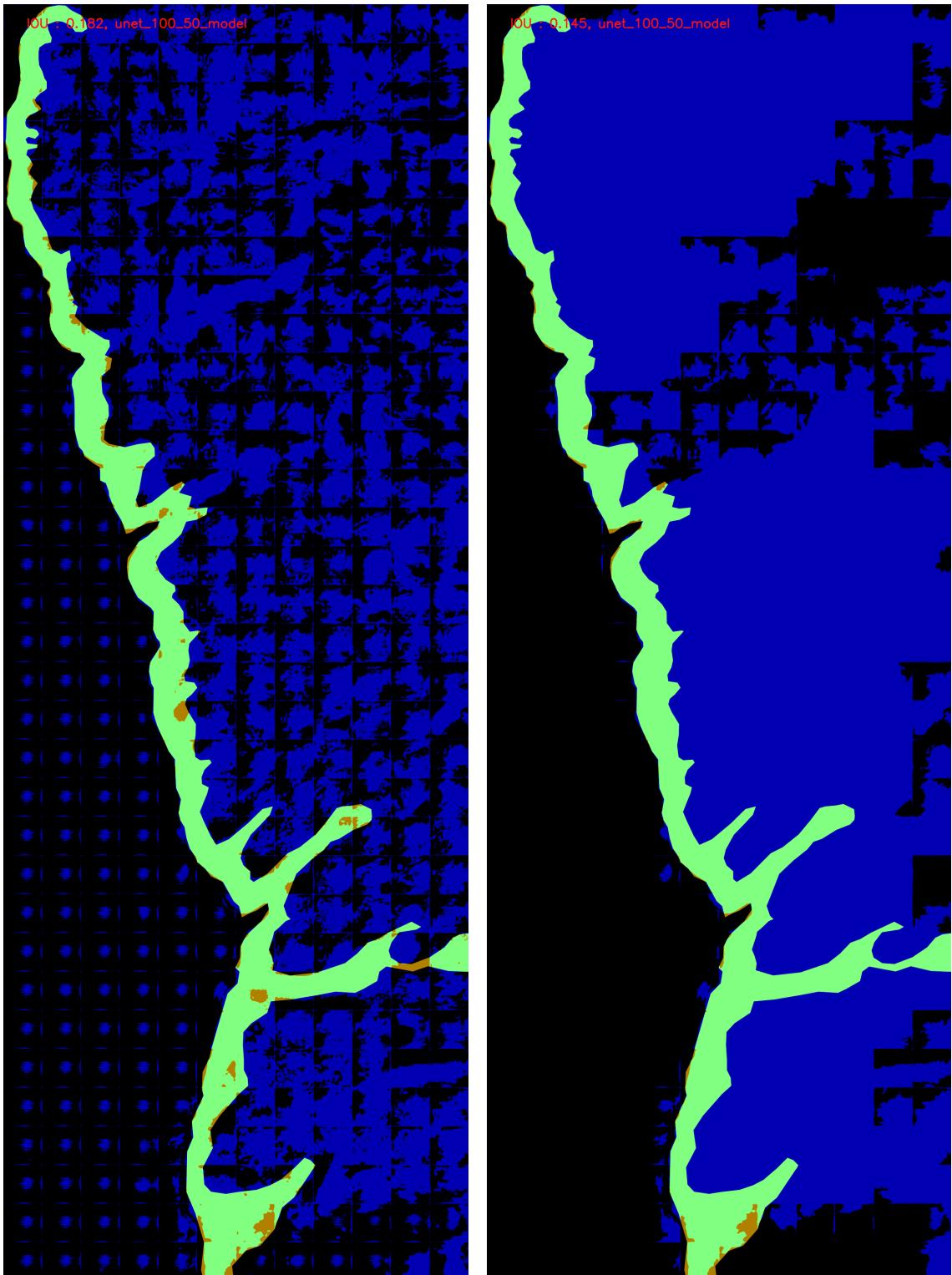
Figure 6.8 IOU result for U-net (3, 98) FP FN TP



Before Post_processing (IOU = 0.487)

After Post_processing (IOU = 0.539)

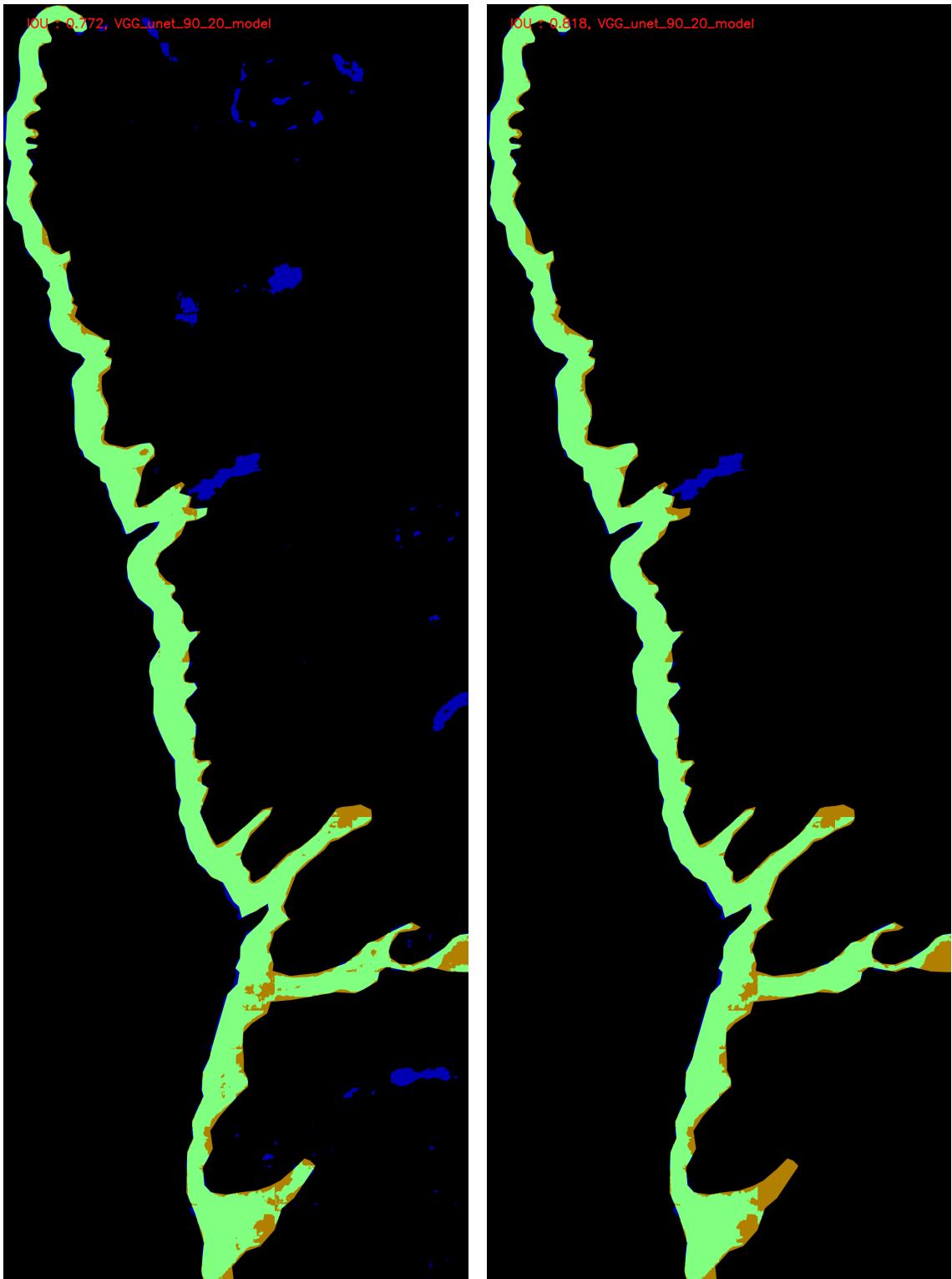
Figure 6.9 IOU result for U-net (20, 90) FP FN TP



Before Post_processing (IOU = 0.182)

After Post_processing (IOU = 0.145)

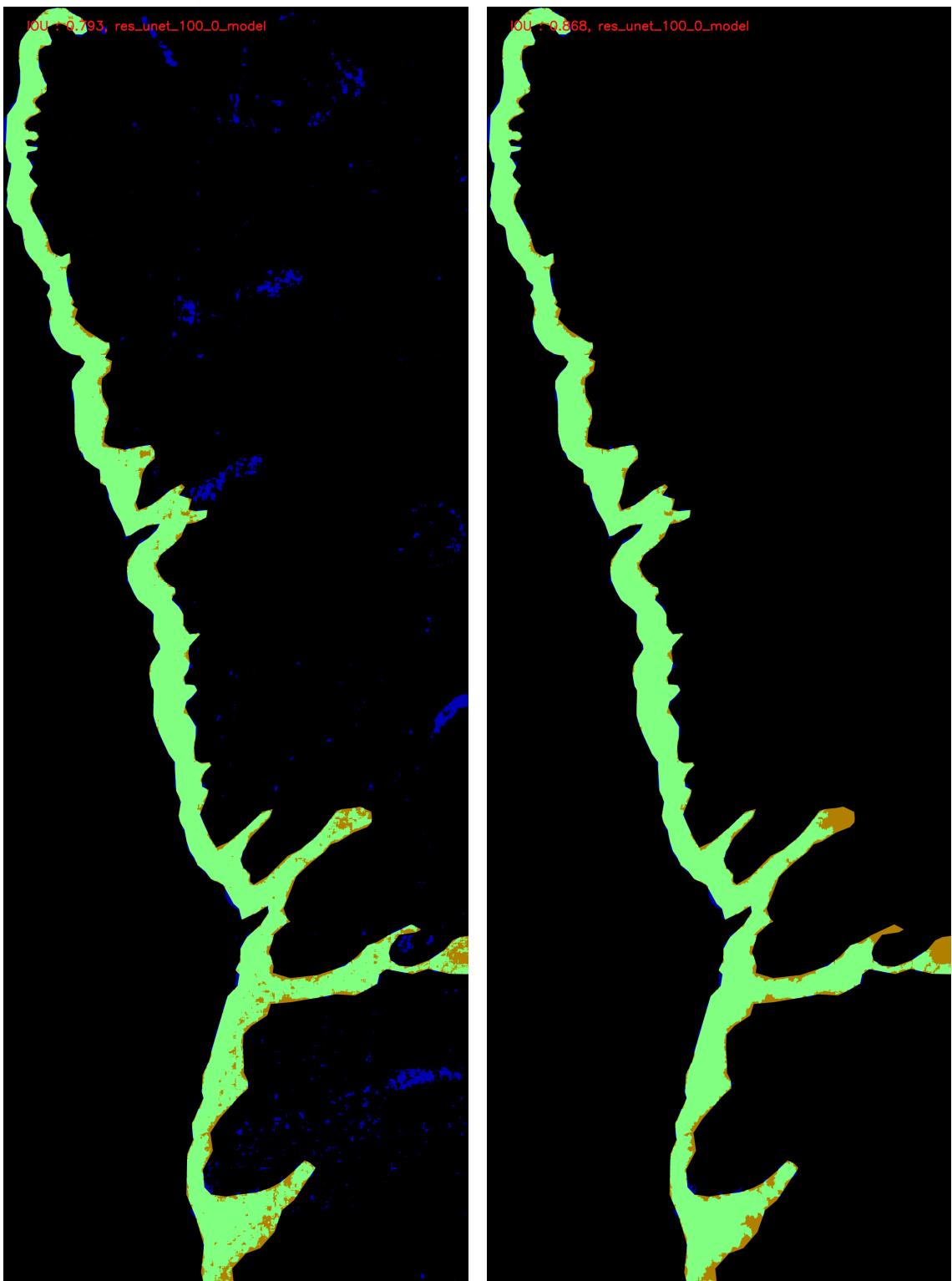
Figure 6.10 IOU result for U-net (50, 100) FP FN TP



Before Post_processing (IOU = 0.772)

After Post_processing (IOU = 0.818)

Figure 6.11 IOU result for VGG-Unet (20, 90) FP FN TP



Befor Post_processing (IOU = 0.793)

After Post_processing (IOU = 0.868)

Figure 6.12 IOU result for res-Unet (0, 100) FP FN TP

6.3.2 Image 20-2 IOU results

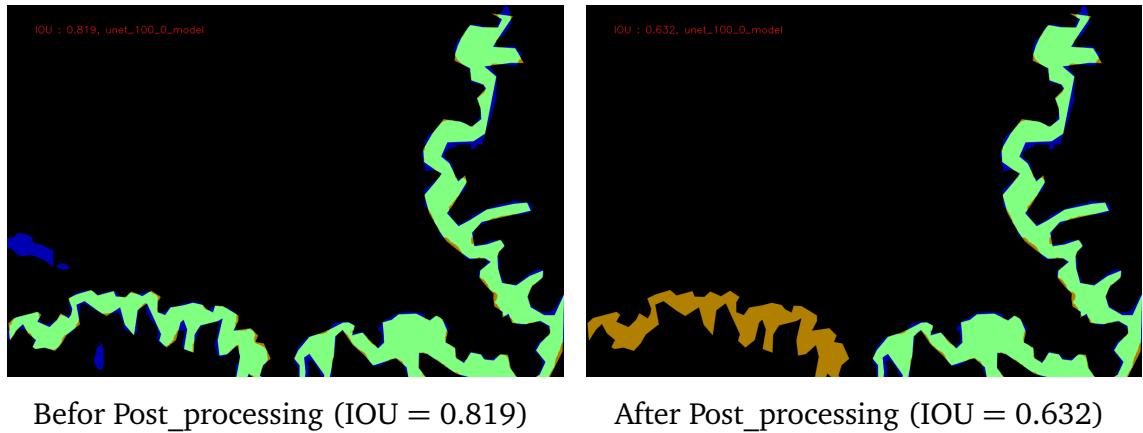


Figure 6.13 IOU result for U-net (0, 100) FP FN TP

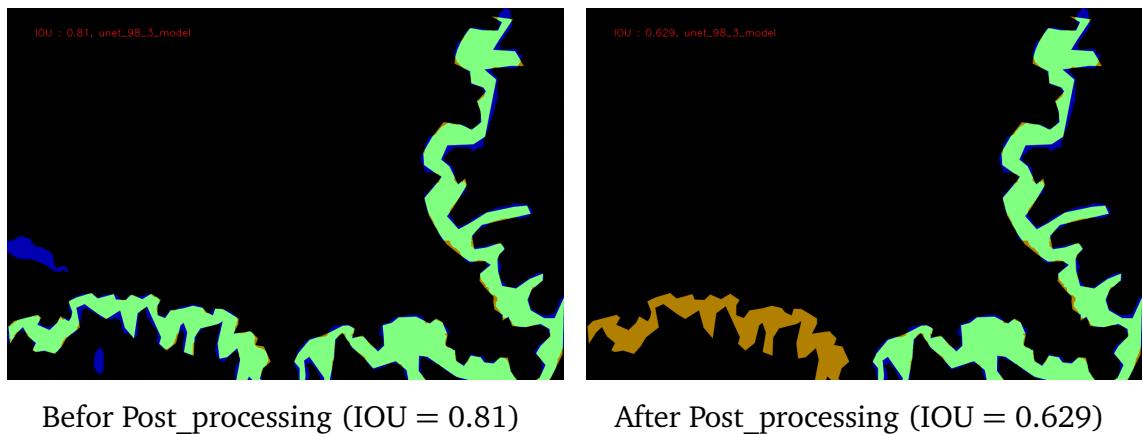


Figure 6.14 IOU result for U-net (3, 98) FP FN TP

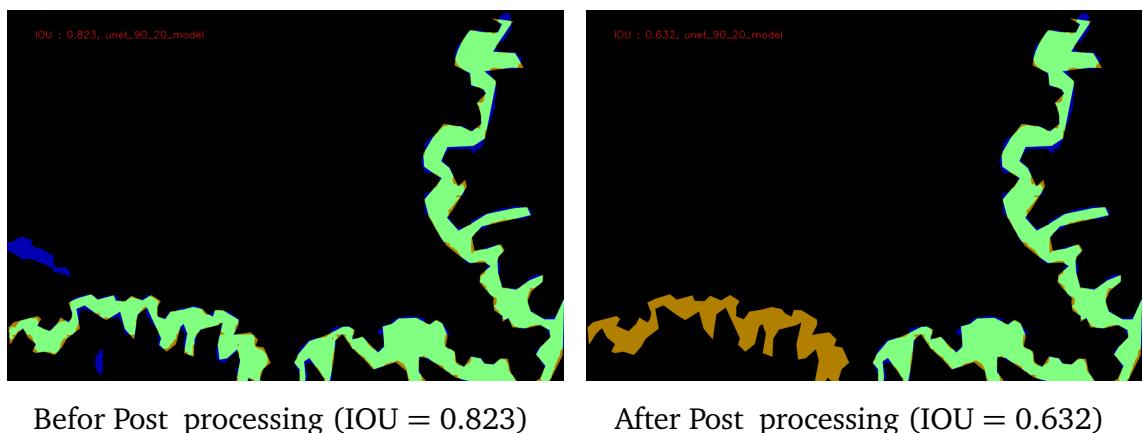


Figure 6.15 IOU result for U-net (20, 90) FP FN TP

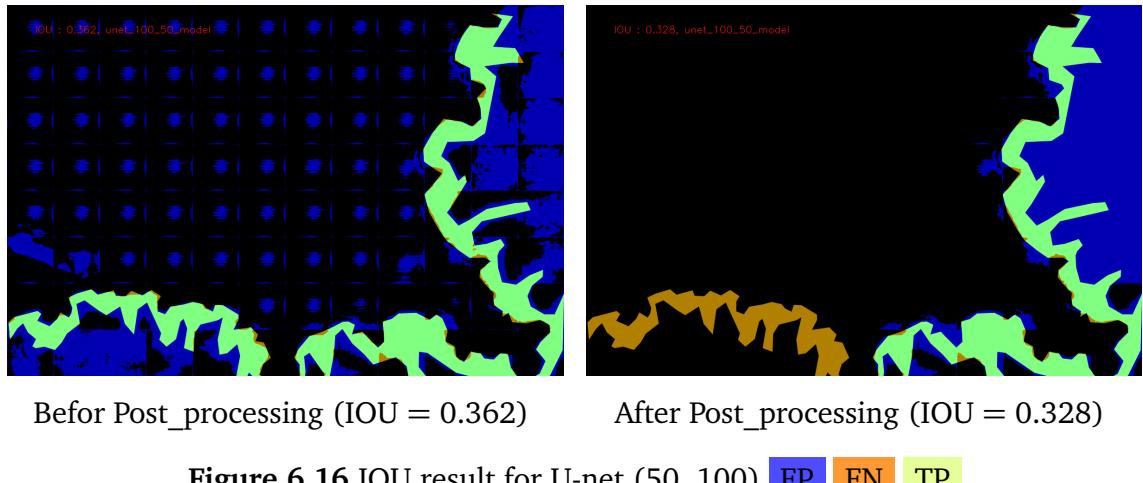


Figure 6.16 IOU result for U-net (50, 100)

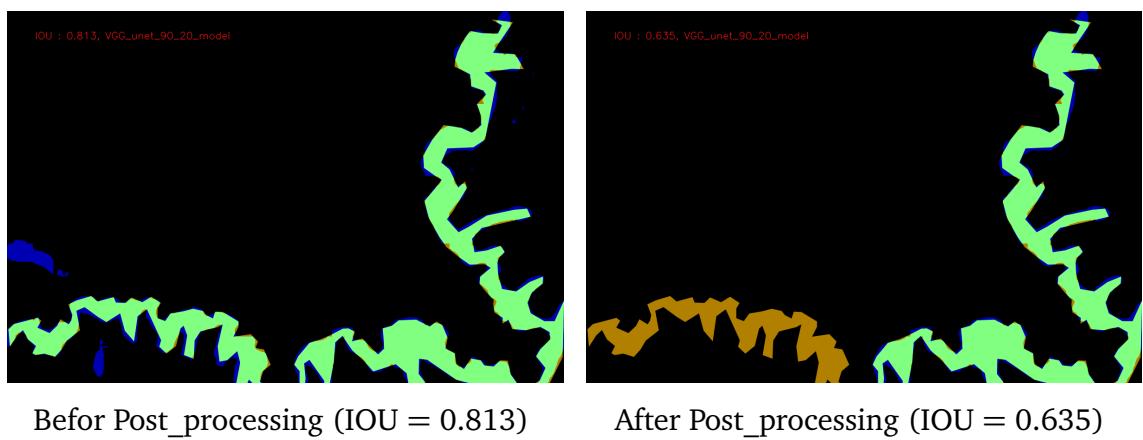


Figure 6.17 IOU result for VGG-Unet (20, 90)

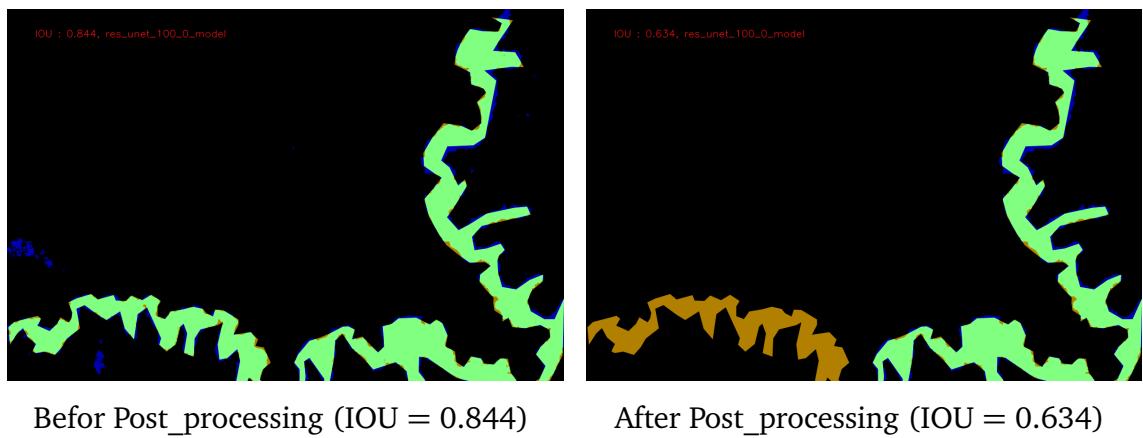
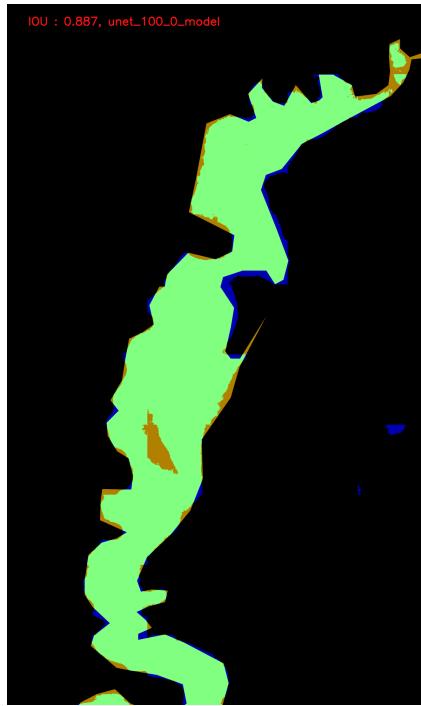
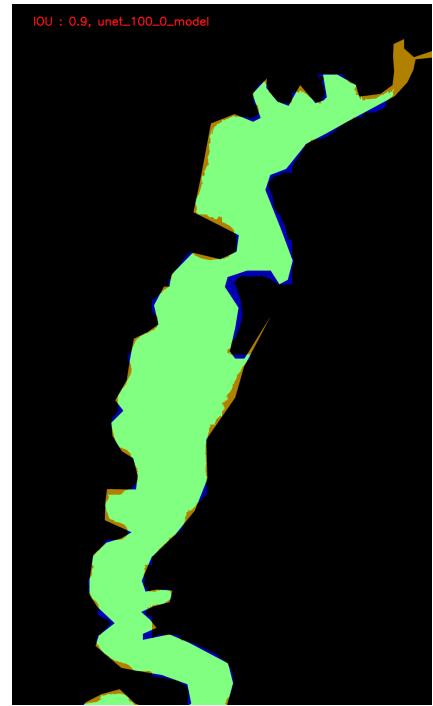


Figure 6.18 IOU result for res-Unet (0, 100)

6.3.3 Image 20-3 IOU results

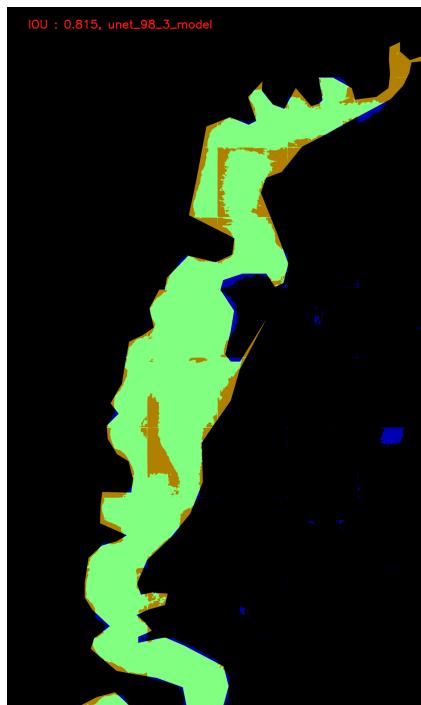


Befor Post_processing (IOU = 0.887)

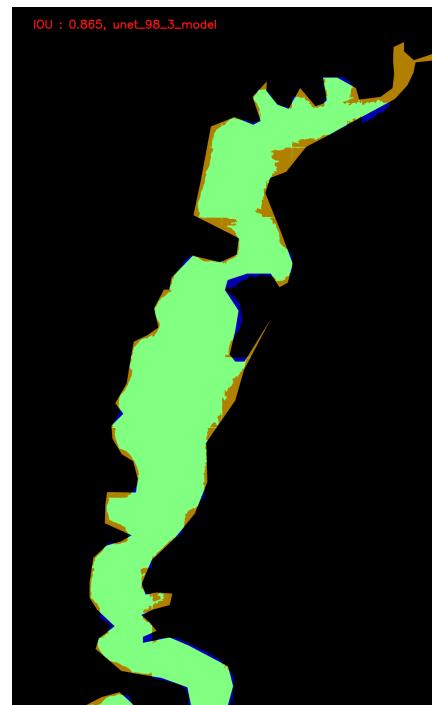


After Post_processing (IOU = 0.9)

Figure 6.19 IOU result for U-net (0, 100) **FP** **FN** **TP**

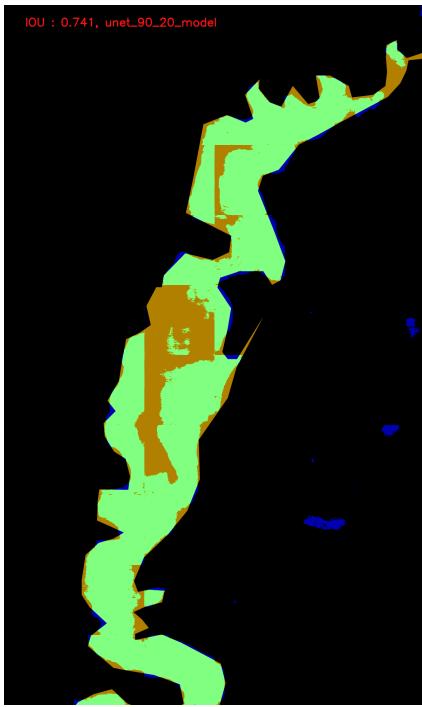


Befor Post_processing (IOU = 0.815)

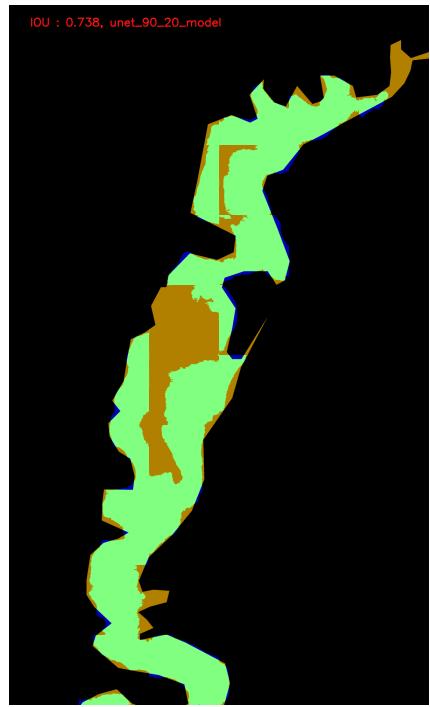


After Post_processing (IOU = 0.865)

Figure 6.20 IOU result for U-net (3, 98) **FP** **FN** **TP**

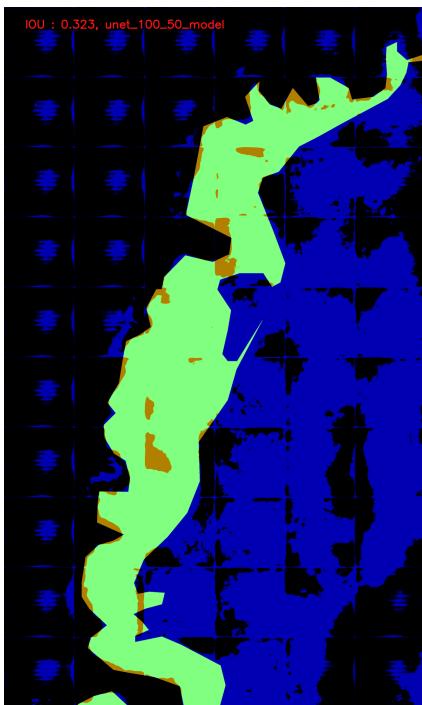


Befor Post_processing (IOU = 0.741)

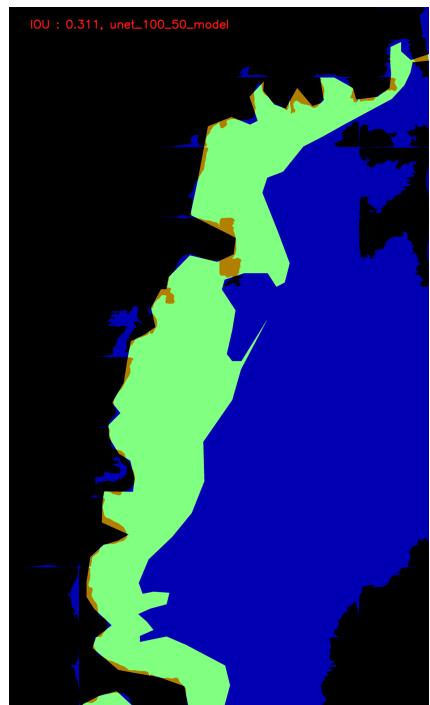


After Post_processing (IOU = 0.738)

Figure 6.21 IOU result for U-net (20, 90) FP FN TP

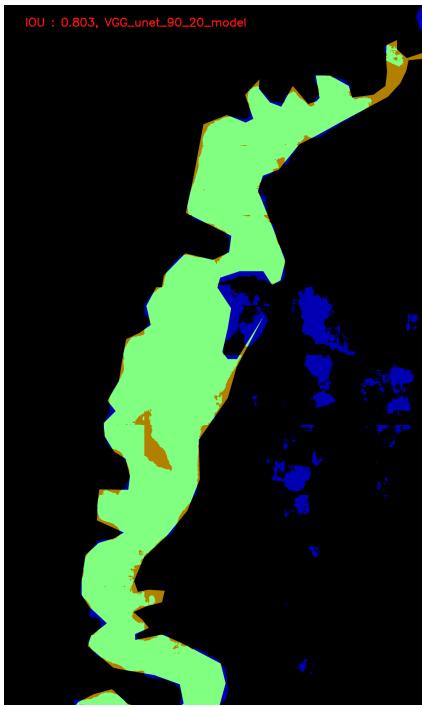


Befor Post_processing (IOU = 0.323)

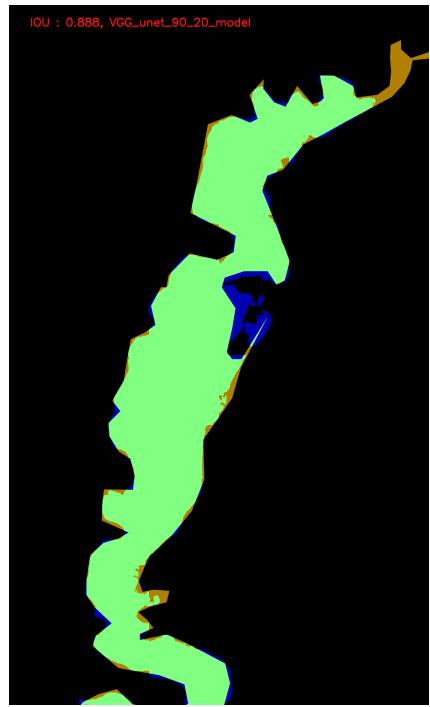


After Post_processing (IOU = 0.311)

Figure 6.22 IOU result for U-net (50, 100) FP FN TP

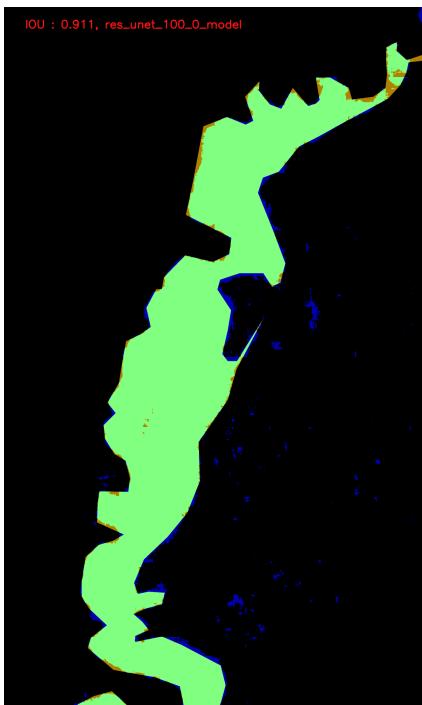


Befor Post_processing (IOU = 0.803)

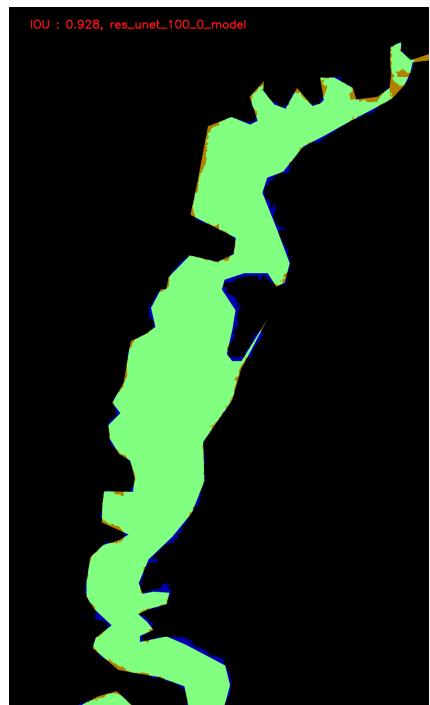


After Post_processing (IOU = 0.888)

Figure 6.23 IOU result for VGG-Unet (20, 90) FP FN TP



Befor Post_processing (IOU = 0.911)



After Post_processing (IOU = 0.928)

Figure 6.24 IOU result for res-Unet (0, 100) FP FN TP

6.4 Test Results on ROI from External Dataset

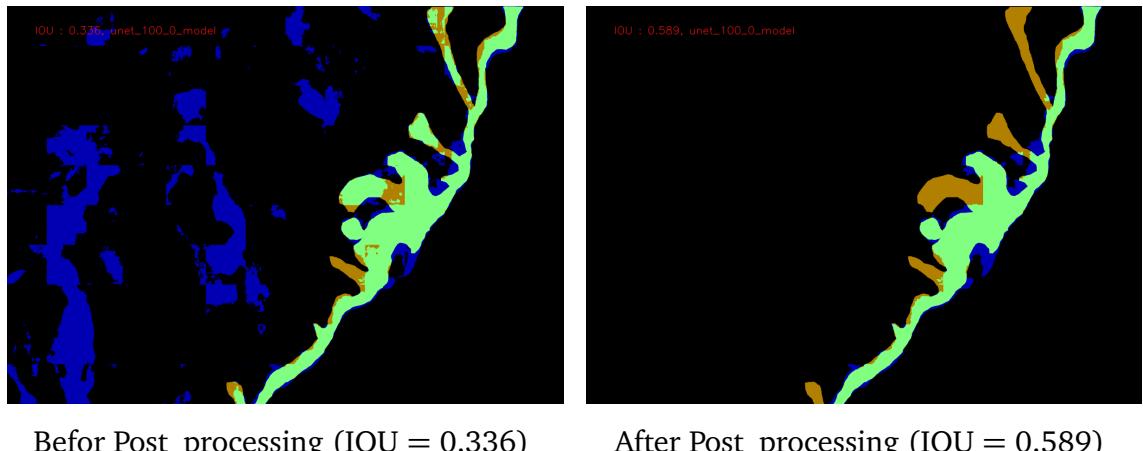
Table 6.4 shows the IOU results on ROIs taken from the test dataset

Table 6.4 Mean IOU Results Before and After Processing for External Dataset's ROIs

Model Name (Range)	IOU Before Post-Processing	IOU After Post-Processing
Unet (0,100)	0.37	0.63
Unet (3,98)	0.27	0.59
Unet (20,90)	0.37	0.48
Unet (50,100)	0.10	0.09
VGG_Unet(20,90)	0.32	0.57
Res_Unet(0,100)	0.06	0.03

The following results show IOU before and after post-processing for 3 ROI images from external datasets.

6.4.1 Image 121 IOU results



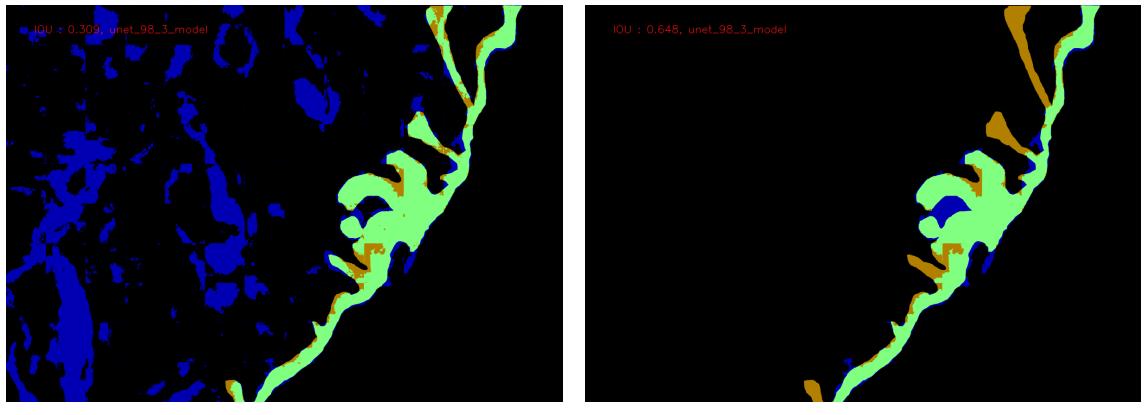


Figure 6.26 IOU result for U-net (3, 98) FP FN TP

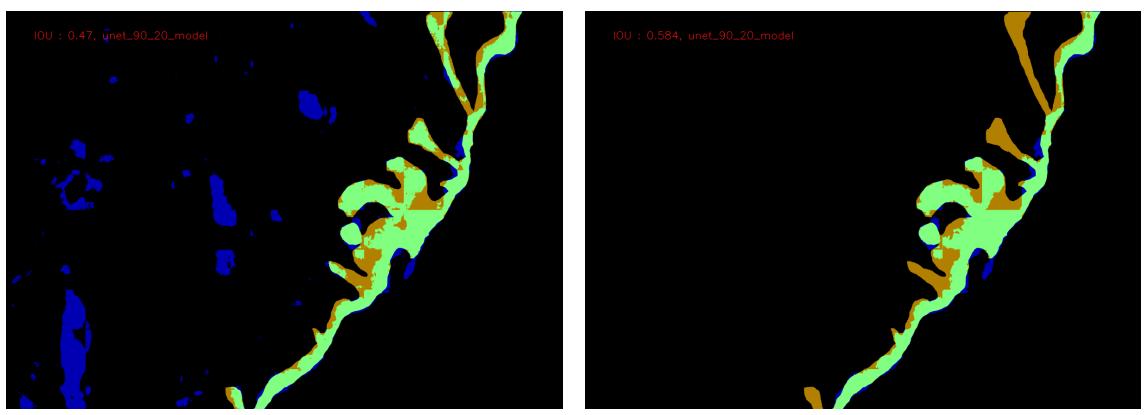


Figure 6.27 IOU result for U-net (20, 90) FP FN TP

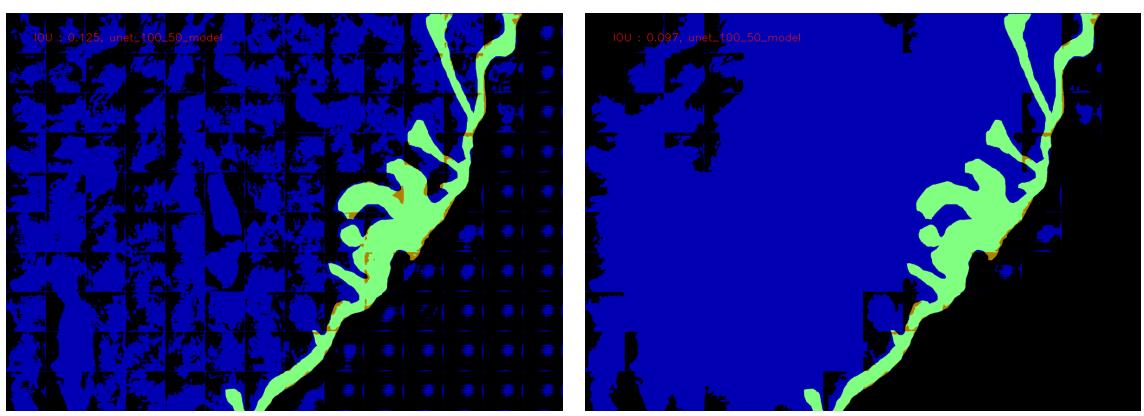


Figure 6.28 IOU result for U-net (50, 100) FP FN TP

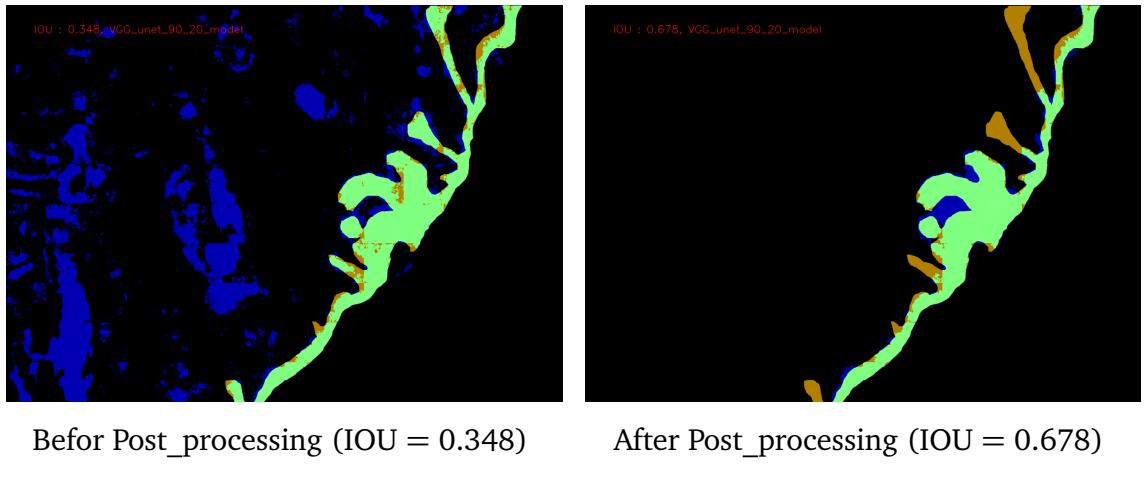


Figure 6.29 IOU result for VGG-Unet (20, 90) 

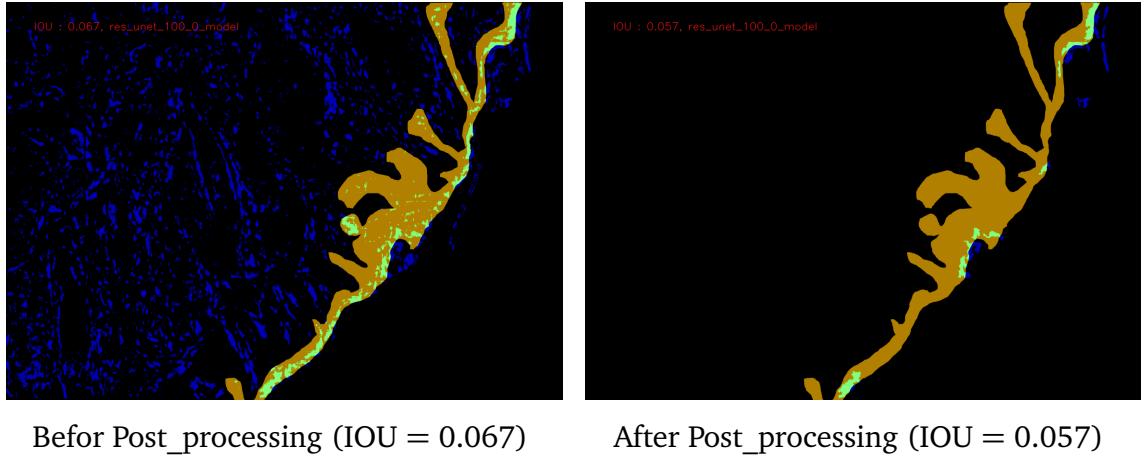


Figure 6.30 IOU result for res-Unet (0, 100) 

6.4.2 Image 356 IOU results

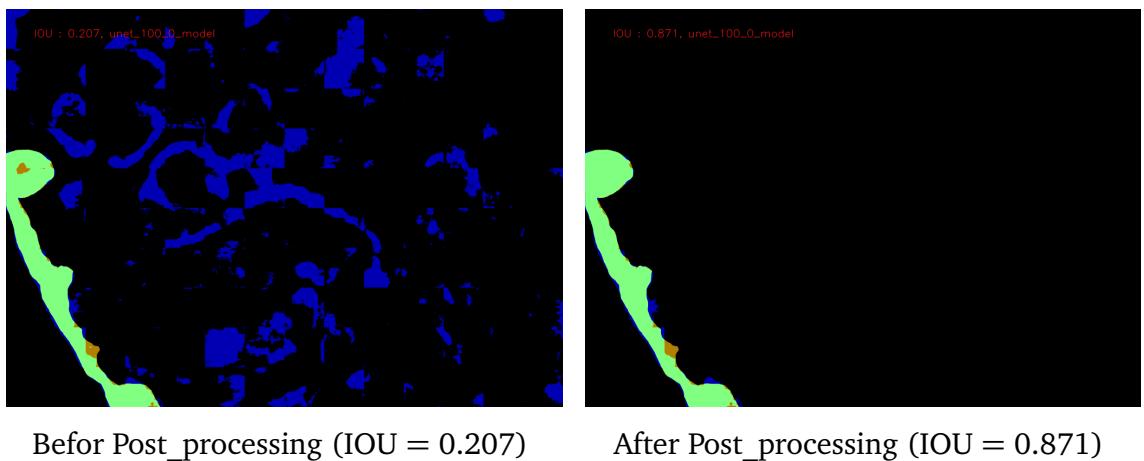


Figure 6.31 IOU result for U-net (0, 100) 

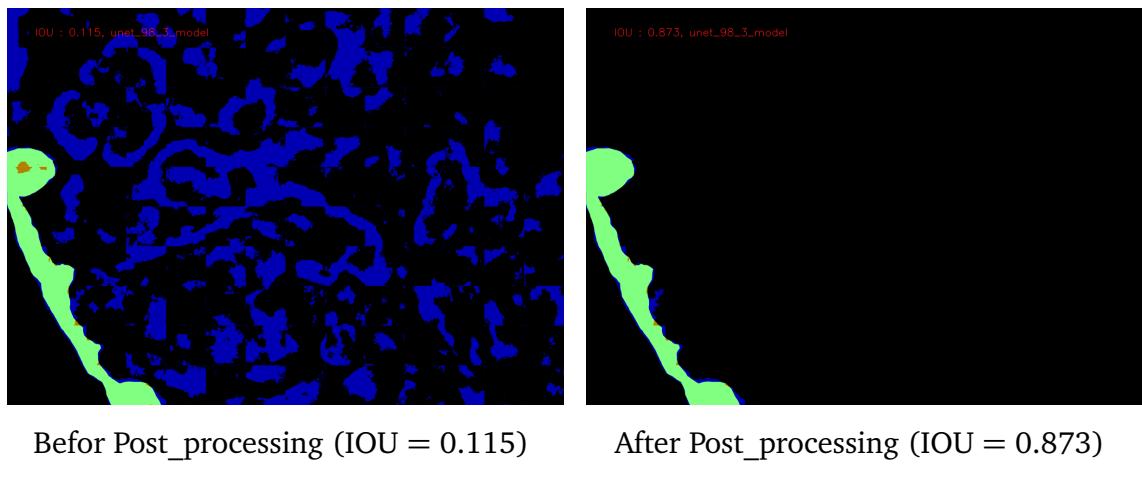


Figure 6.32 IOU result for U-net (3, 98)

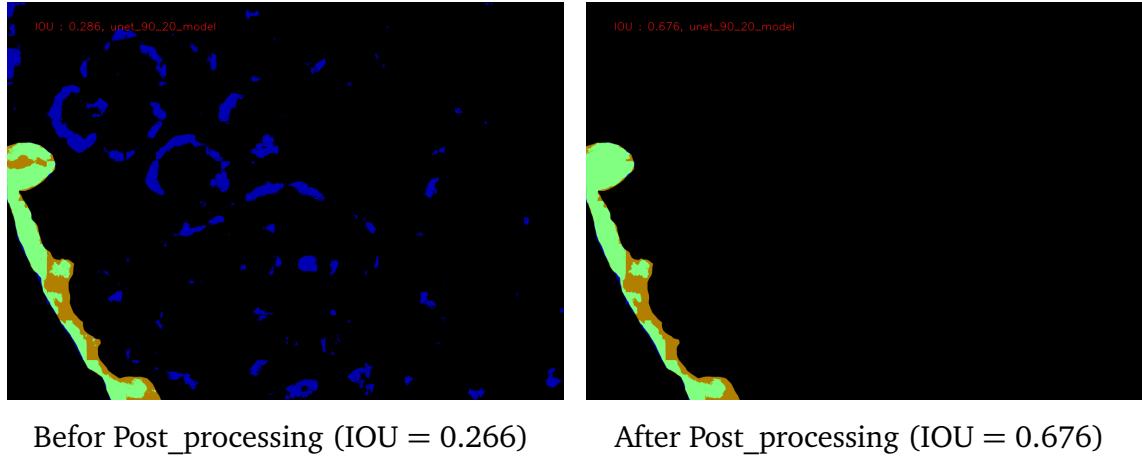


Figure 6.33 IOU result for U-net (20, 90)

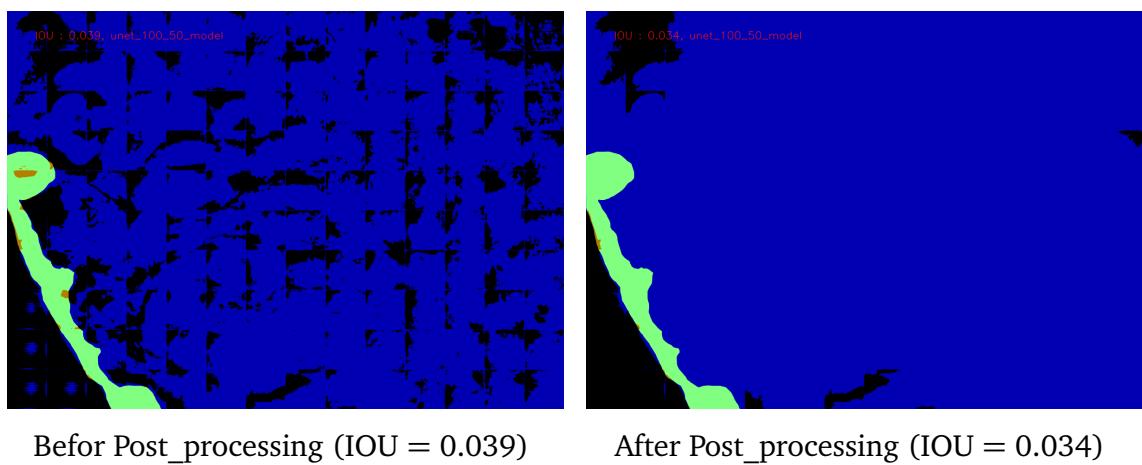


Figure 6.34 IOU result for U-net (50, 100)

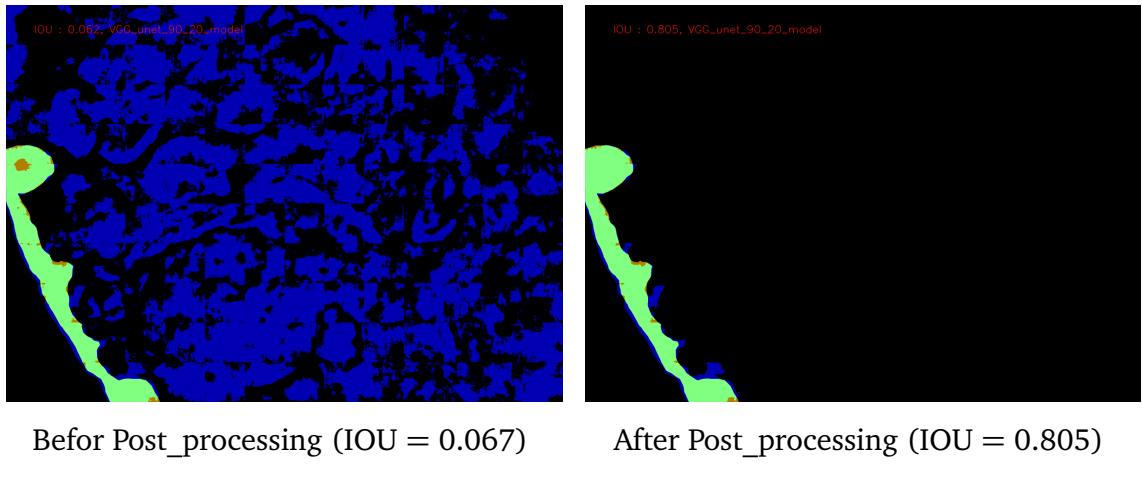


Figure 6.35 IOU result for VGG-Unet (20, 90) 

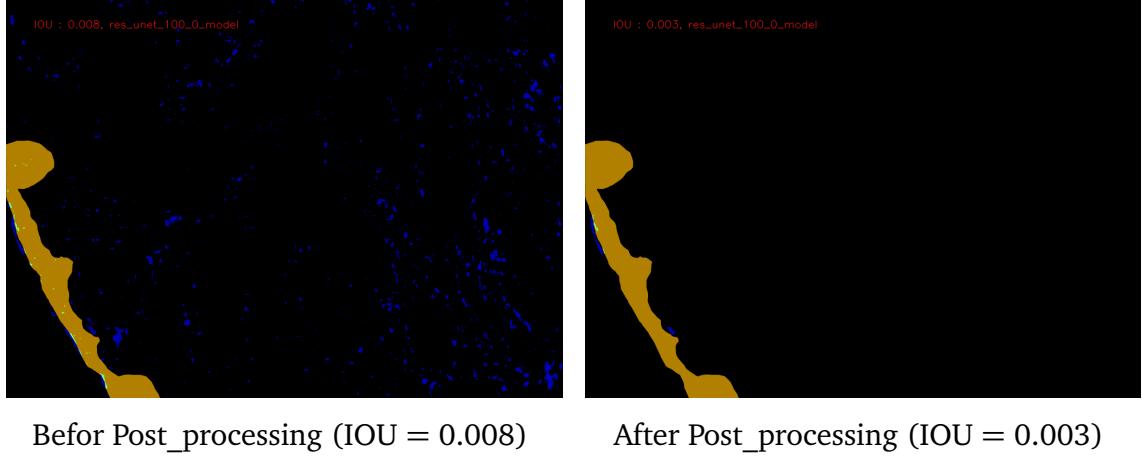


Figure 6.36 IOU result for res-Unet (0, 100) 

6.4.3 Image 358 IOU results

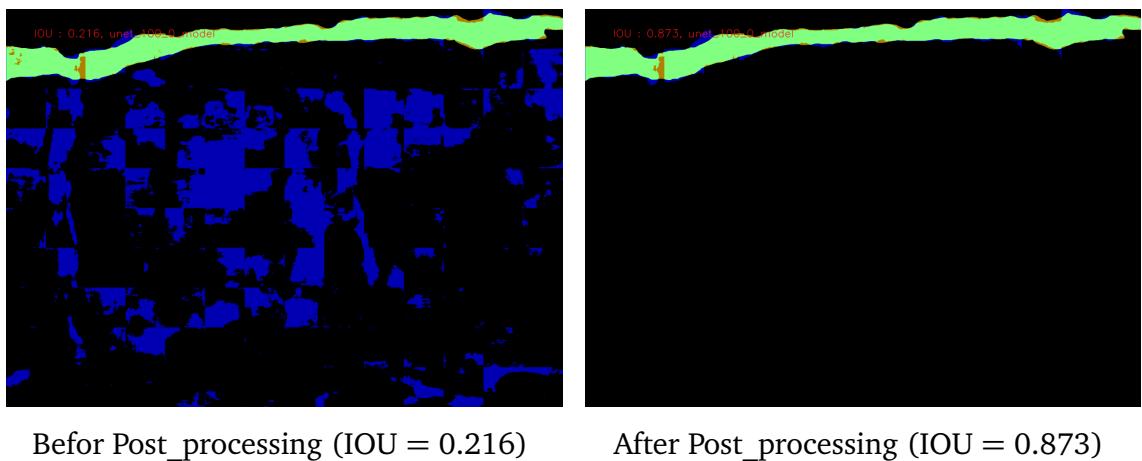
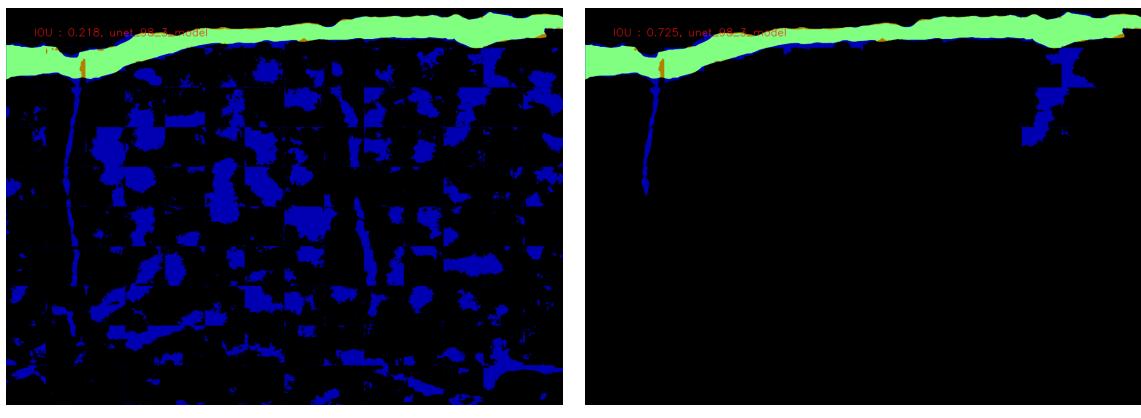
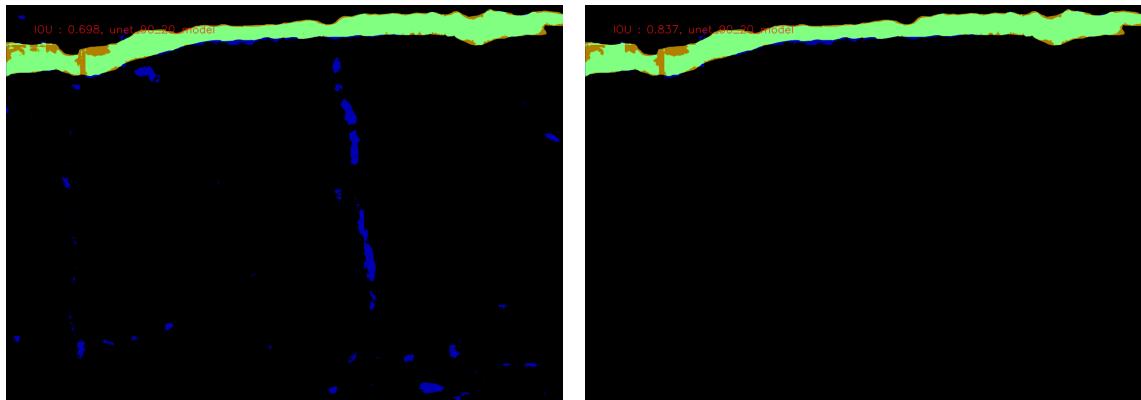


Figure 6.37 IOU result for U-net (0, 100) 



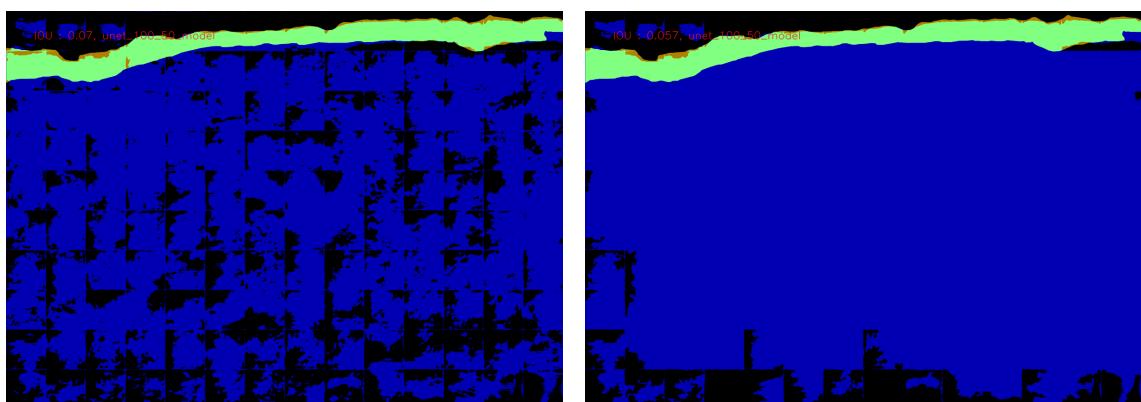
Befor Post_processing (IOU = 0.218) After Post_processing (IOU = 0.725)

Figure 6.38 IOU result for U-net (3, 98) FP FN TP



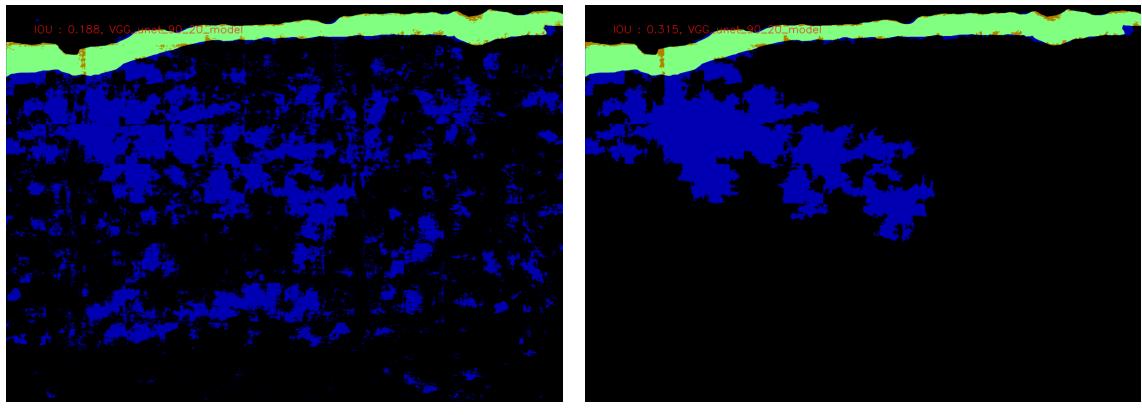
Befor Post_processing (IOU = 0.698) After Post_processing (IOU = 0.837)

Figure 6.39 IOU result for U-net (20, 90) FP FN TP



Befor Post_processing (IOU = 0.07) After Post_processing (IOU = 0.057)

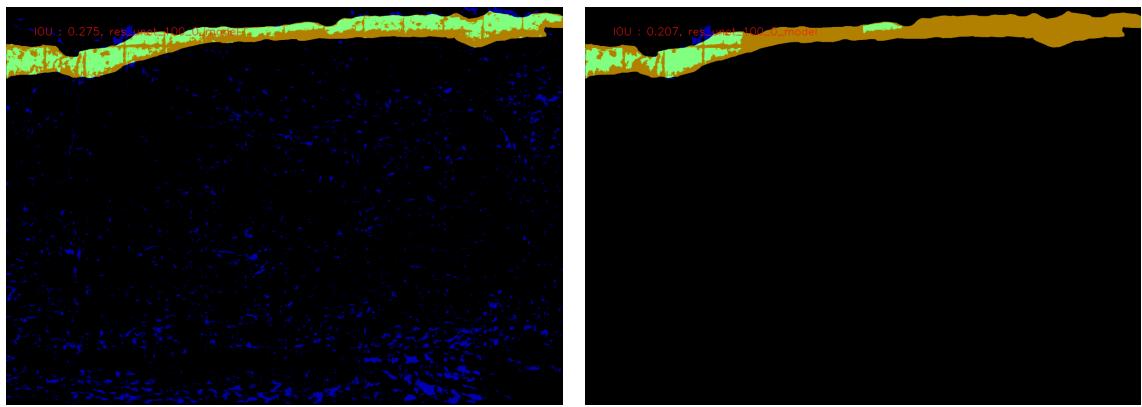
Figure 6.40 IOU result for U-net (50, 100) FP FN TP



Befor Post_processing (IOU = 0.188)

After Post_processing (IOU = 0.315)

Figure 6.41 IOU result for VGG-Unet (20, 90)



Befor Post_processing (IOU = 0.275)

After Post_processing (IOU = 0.207)

Figure 6.42 IOU result for res-Unet (0, 100)

References

- [1] P. Wojciech and H. R. Michael, *Histology: A Text and Atlas: With Correlated Cell and Molecular Biology*, English, International edition. Wolters Kluwer Health, ISBN: 978-1975115364.
- [2] *Gdc data portal*, Online, Accessed on June 22, 2023. [Online]. Available: https://portal.gdc.cancer.gov/repository?filters=%7B%22op%22%3A%22and%22%2C%22content%22%3A%5B%7B%22content%22%3A%7B%22field%22%3A%22files.cases.primary_site%22%2C%22value%22%3A%5B%22skin%22%5D%7D%2C%22op%22%3A%22in%22%7D%2C%7B%22op%22%3A%22in%22%2C%22content%22%3A%7B%22field%22%3A%22files.data_type%22%2C%22value%22%3A%5B%22Slide%20Image%22%5D%7D%7D%5D%7D
- [3] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*, Springer, 2015, pp. 234–241.