

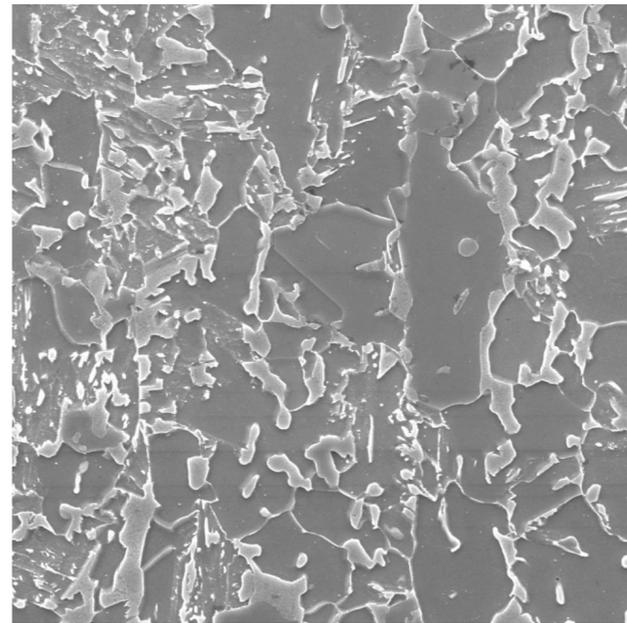
Steel Image Segmentation

2023 May 16

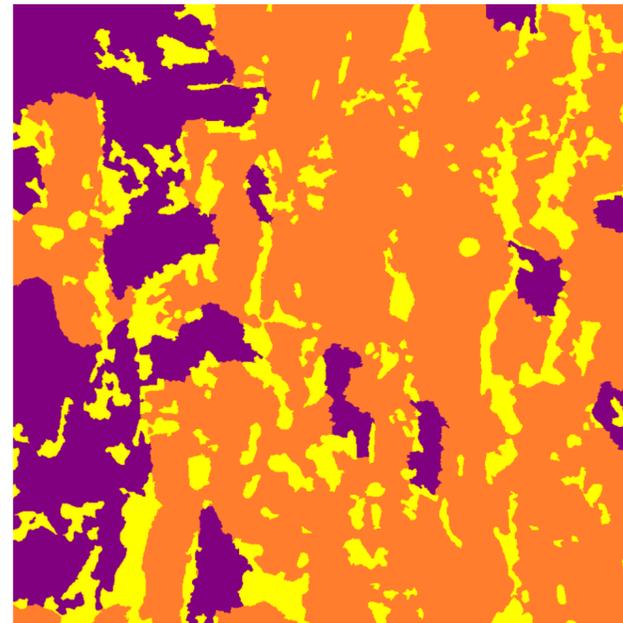
Aim

To identify and label the components of magnified steel images by training on one pair of known set of input and predicting results on different type and magnification of images.

1. Predicting Magnified Images of same steel type
2. Predicting images of different steel types

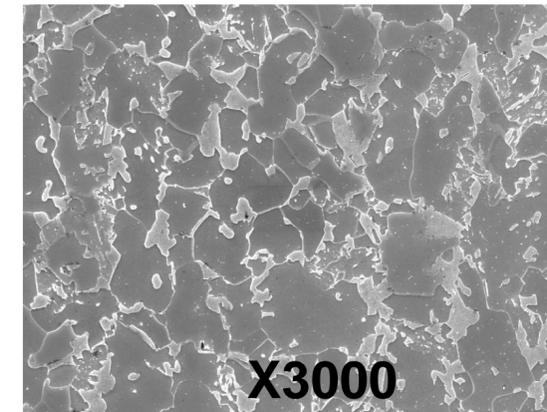


Steel Image



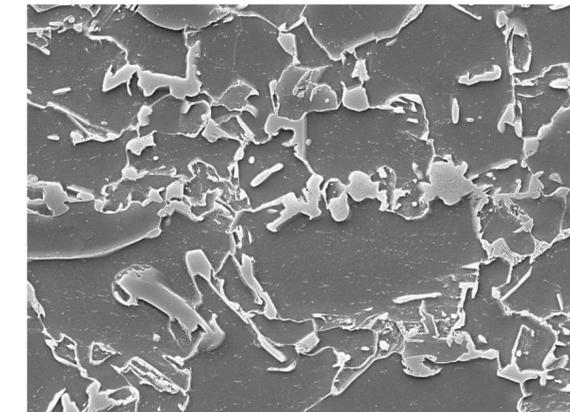
Labeled Image

 Martensite  Ferrite  Bainite



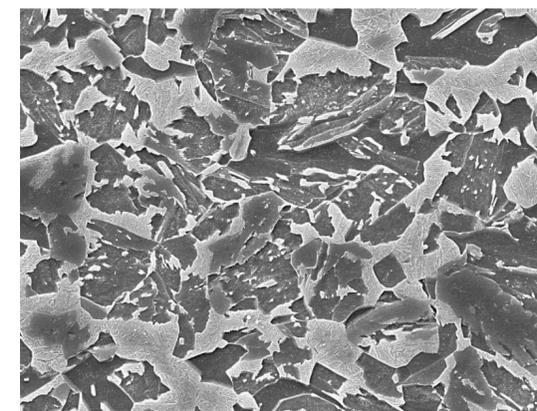
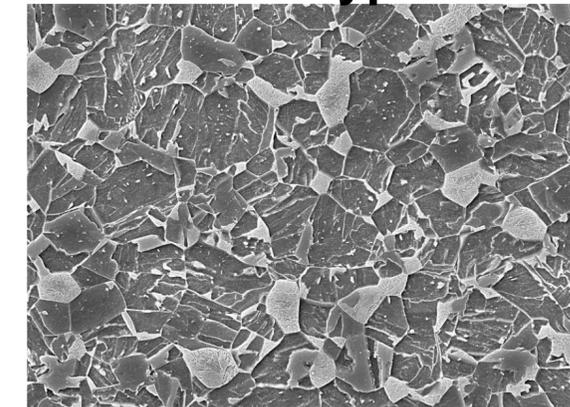
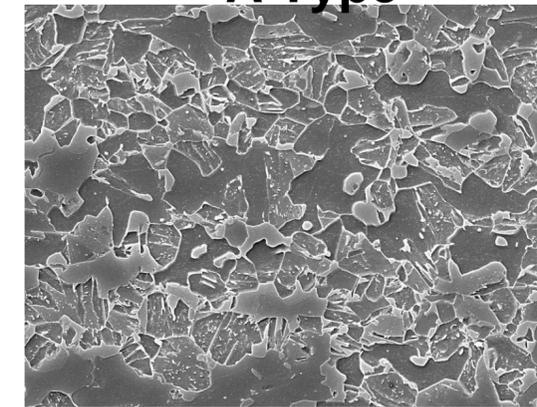
X3000

A Type



X5000

D3 Type

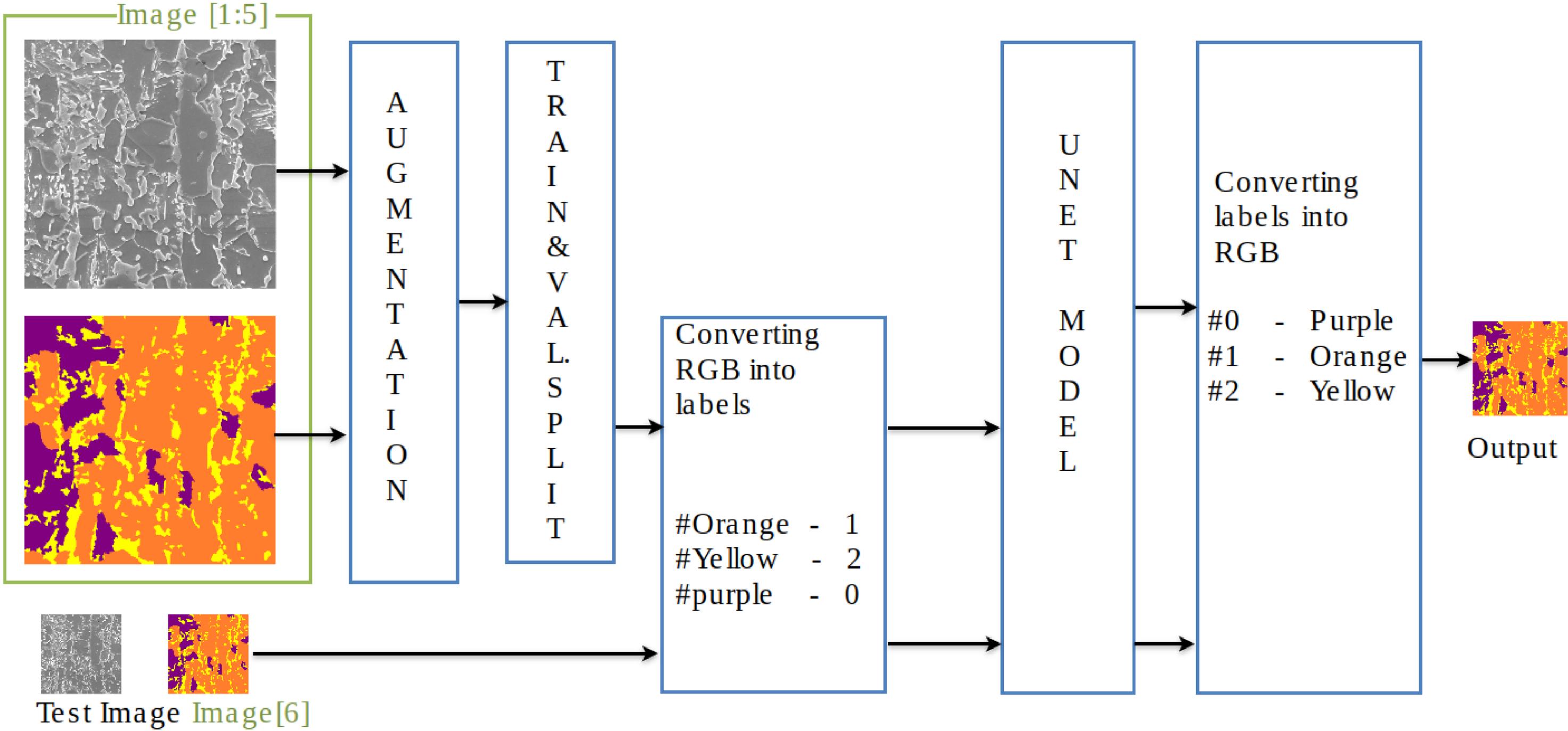


H2 Type

Dataset

- Consists a total of 6 pair of images (original-label).
- 5 image pairs are selected for training and 1 image pair is kept for testing.
- Inferencing is performed on different set of images having different properties than the trained image pairs.
 - X3000 magnified images
 - X5000 magnified images
 - A type steel image
 - D3 type steel image
 - H2 type steel image

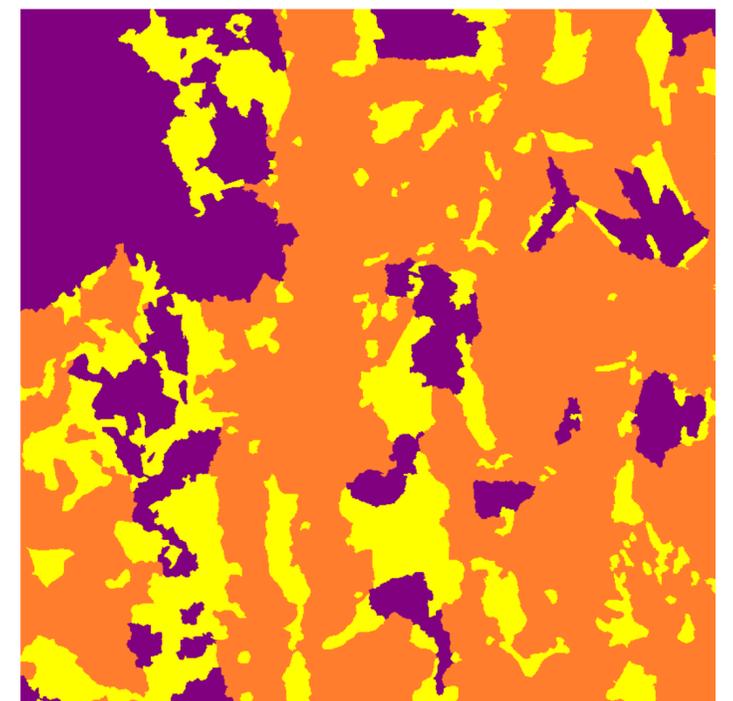
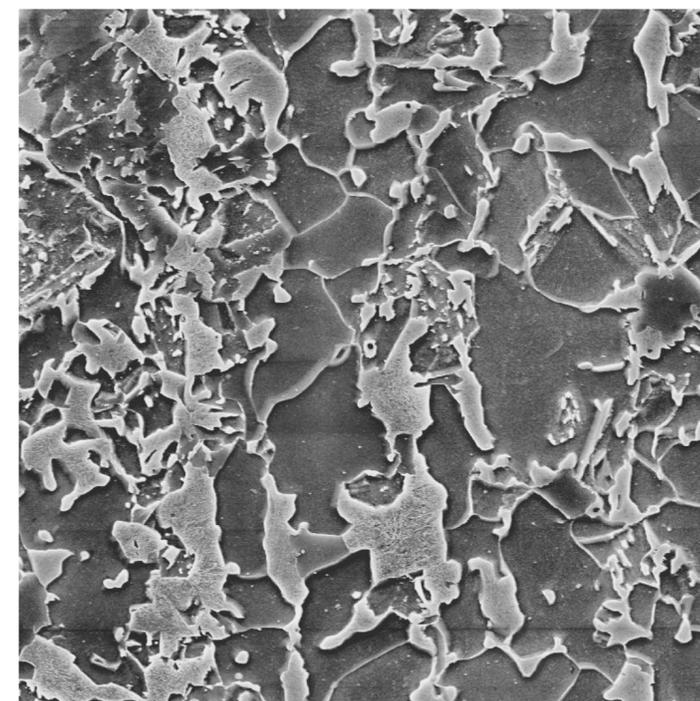
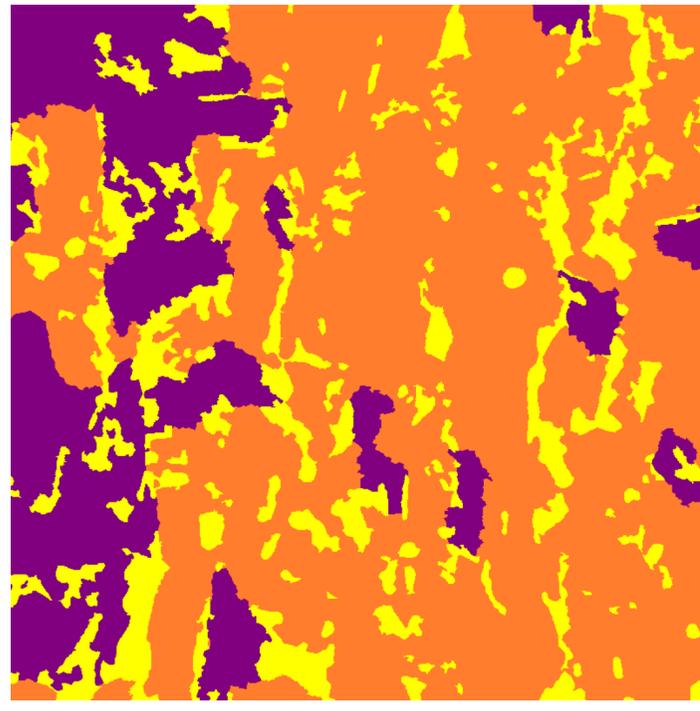
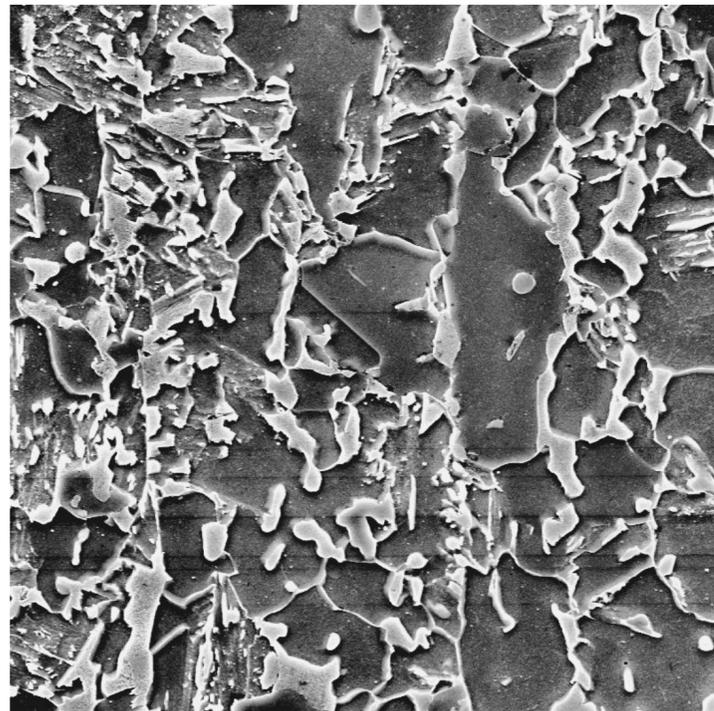
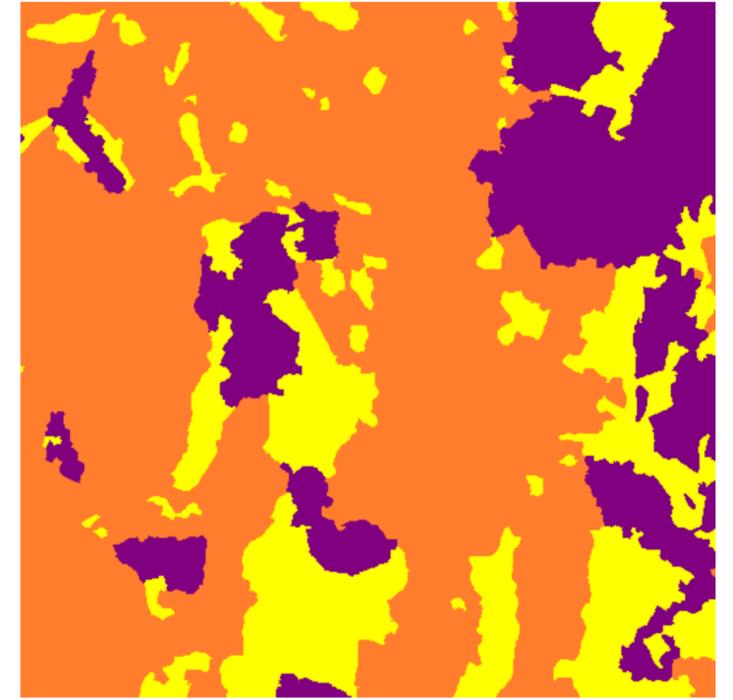
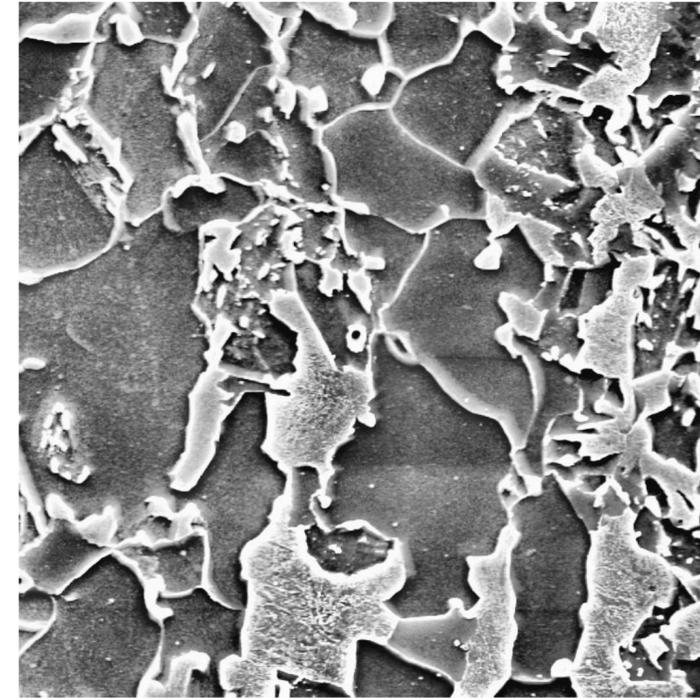
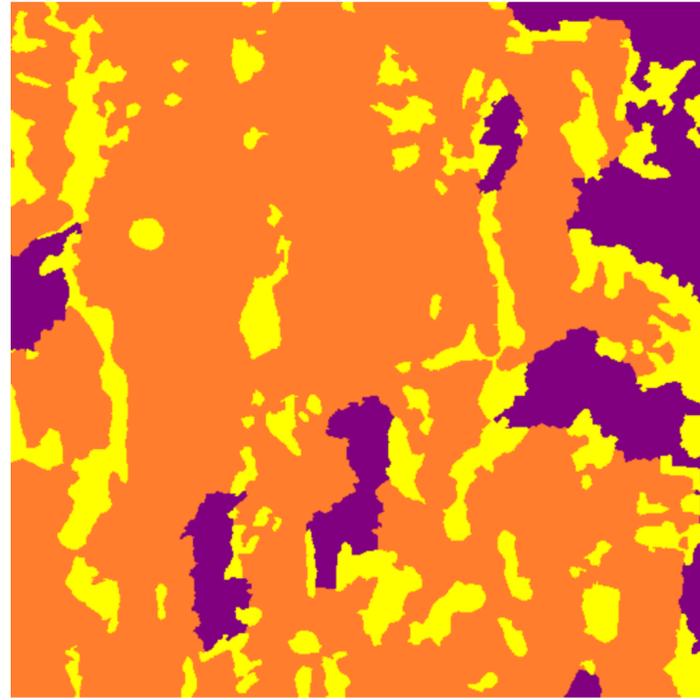
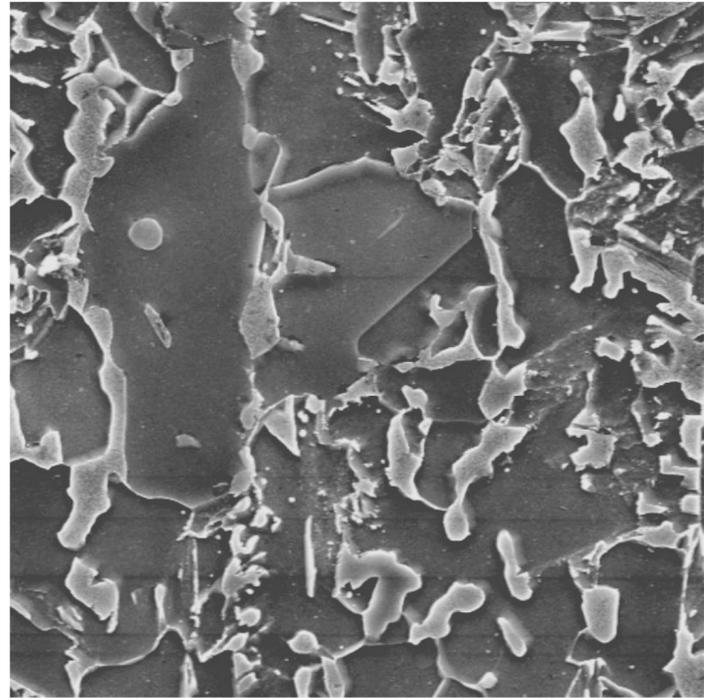
In Brief



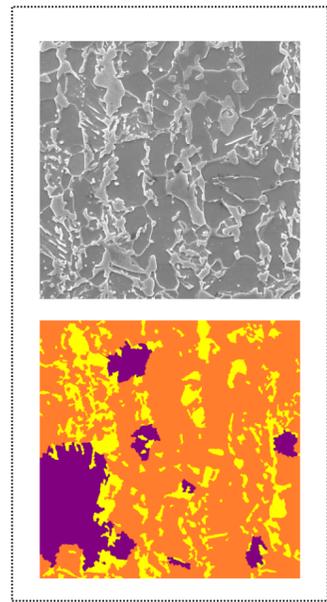
Augmentation

- The following Augmentations were performed on selected 5 image pairs
 - **Magnification (1-2.5x)**
 - To train the model on scalable image patterns
 - **Sliding Window (5 Pixels)**
 - To generate more number of images which slight change in image characteristics
 - **Random Intensity (0-10)**
 - As the intensity of Yellow/Orange/Purple label varies, generating more random intensity images will help train model better
 - **Random Gamma (1-10)**
 - To generate images and train the model on variable brightness and contrast as different steel types have different brightness and contrast properties.
 - **Horizontal and Vertical Flip**
 - **Rotation (10 degrees)**
 - Flip and Rotations are applied to train the model for detecting microstructures in images

Sample Augmented Images



Model Outlook

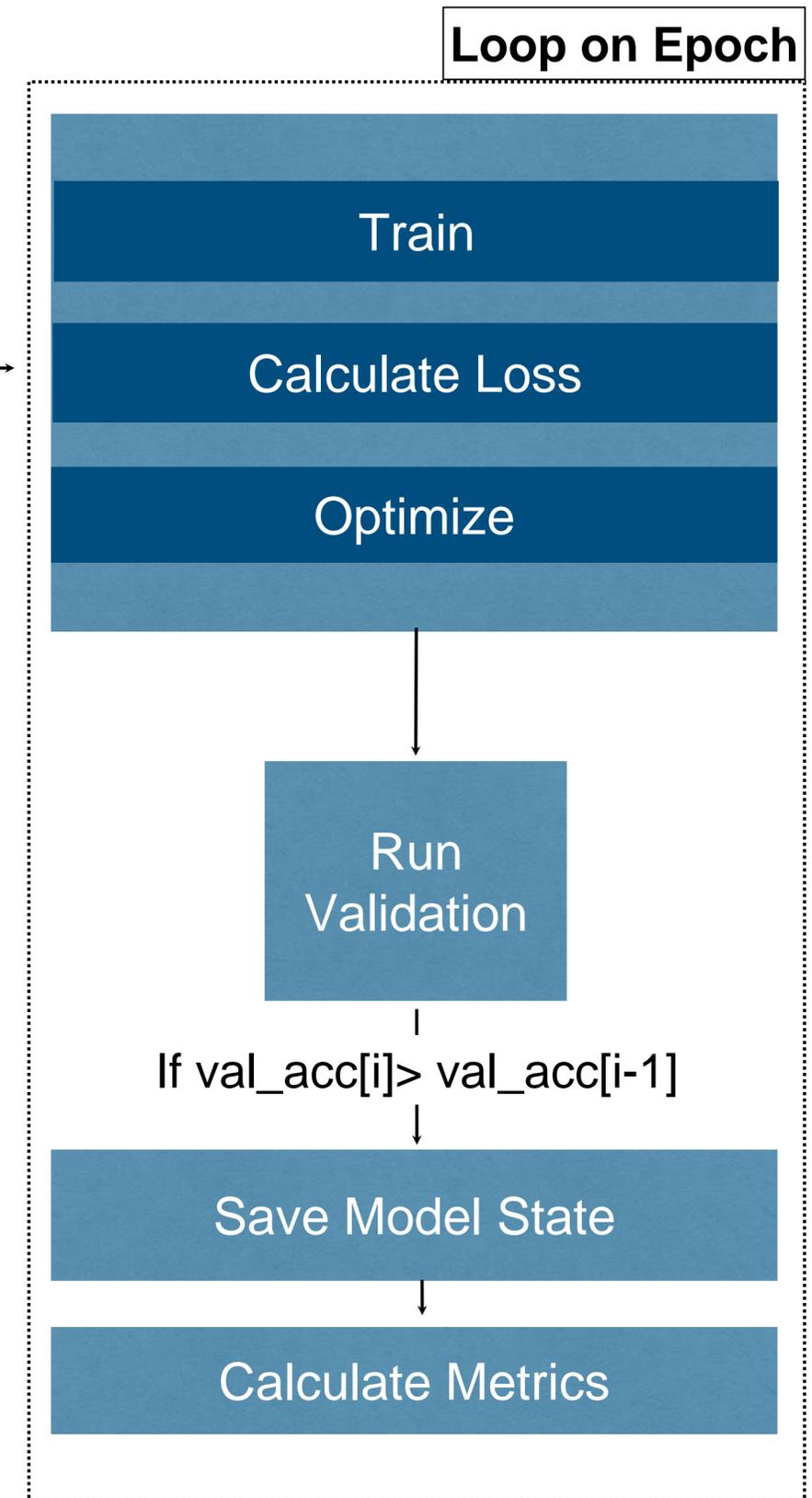


Input Images

Convert to corresponding labels of {0, 1, 2}

reshape

Set Model
Set Optimizer
Set Loss



Setup

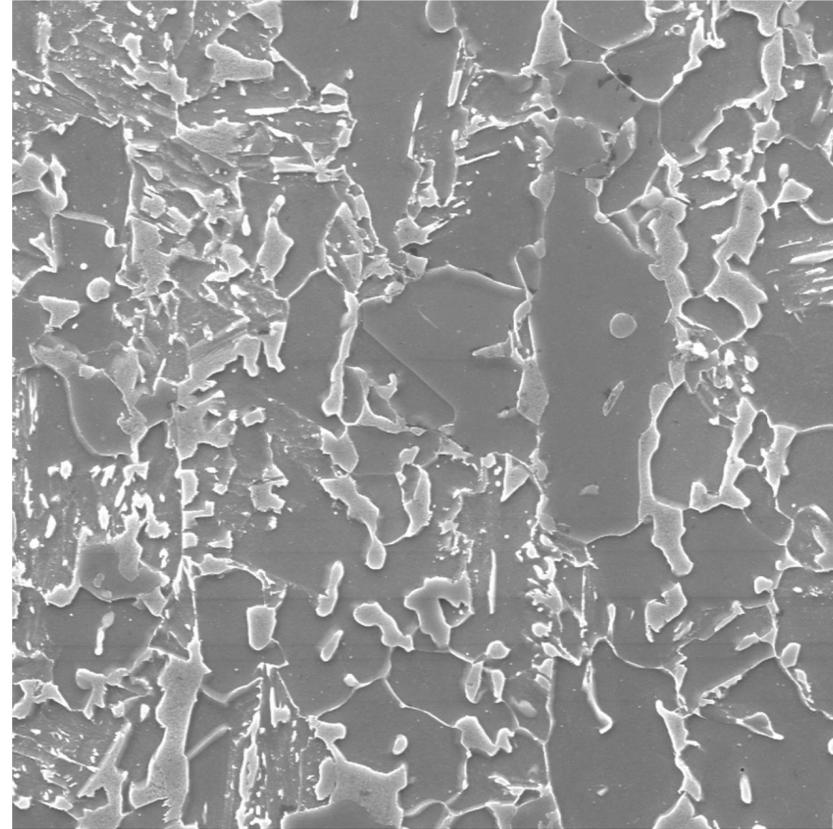
- kernel size - 3
- Batch Size - 16
- Learning Rate - 0.00001
- Channel - 16
- Depth - 4

Model	UNET
Optimizer	Adam
Loss	Focal (0.25)

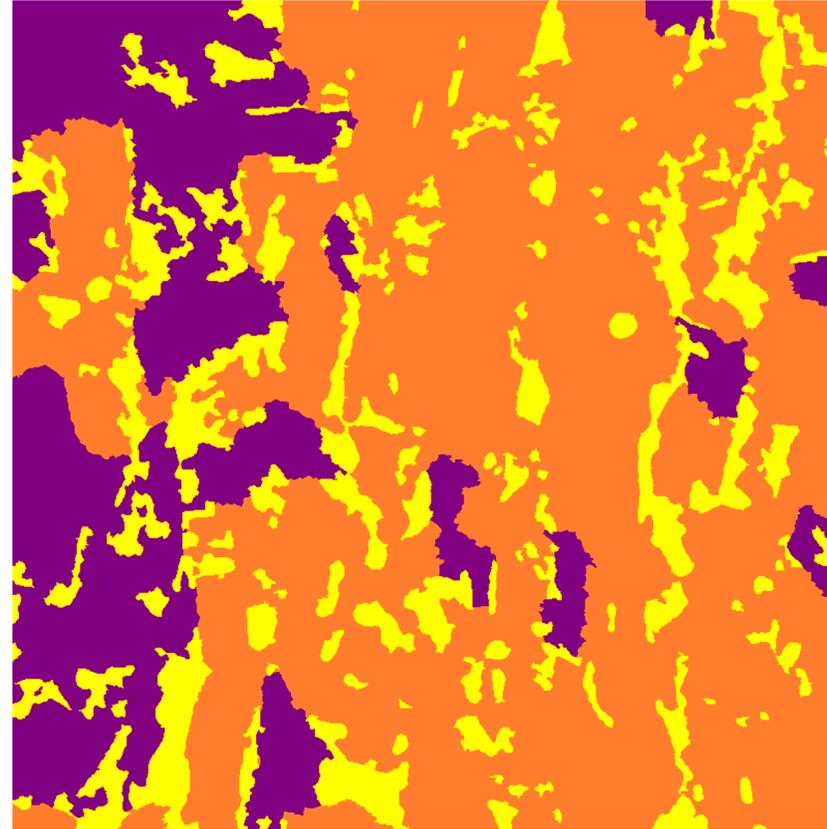
Metrics

Loss, Confusion Matrix, mIoU, Accuracy

Output Results

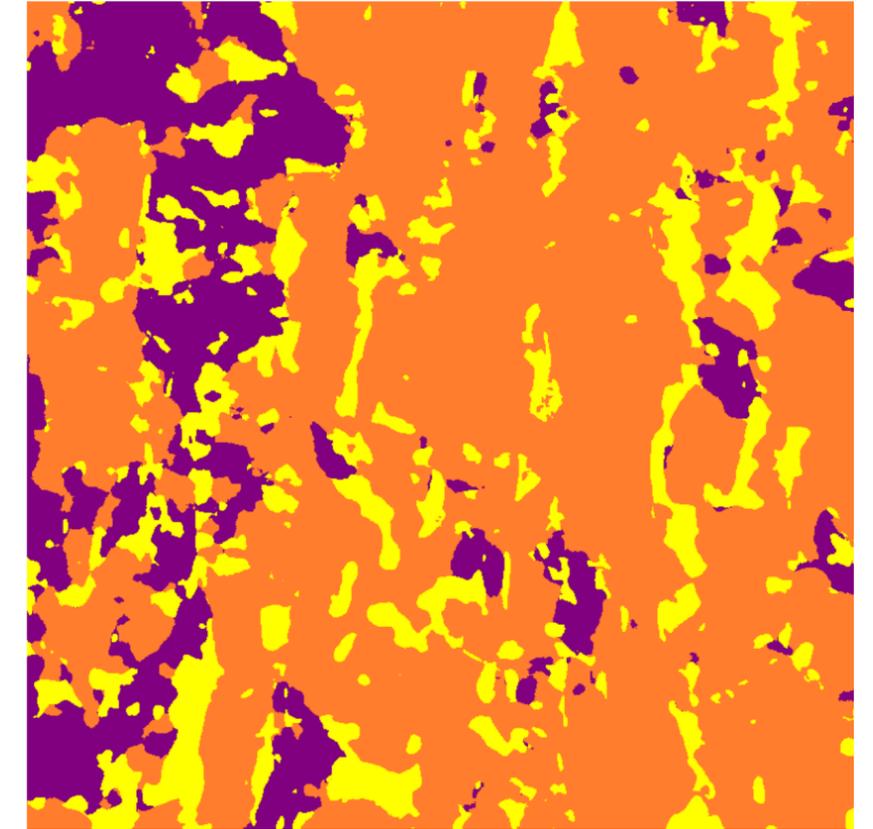


Original



Label

Test Image - 6.bmp



Prediction

Training Accuracy- 91.68
Validation mIOU - 0.4684
Validation Acc - 95.31
Test Accuracy - 87.98
Dice Coefficient - 0.8609

Activation	ReLU
BlurPooling	None

Inference Results

Activation ReLU
BlurPooling None

	Avg (10)	Avg (8)	(1)	(1)	(1)
	x3000	x5000	test1 A type	test2 D3 type	test3 H2 type
Accuracy(Dice)	83.97 (0.8976)	90.01 (0.8976)	77.41 (0.7716)	64.75 (0.645)	73.89 (0.7364)

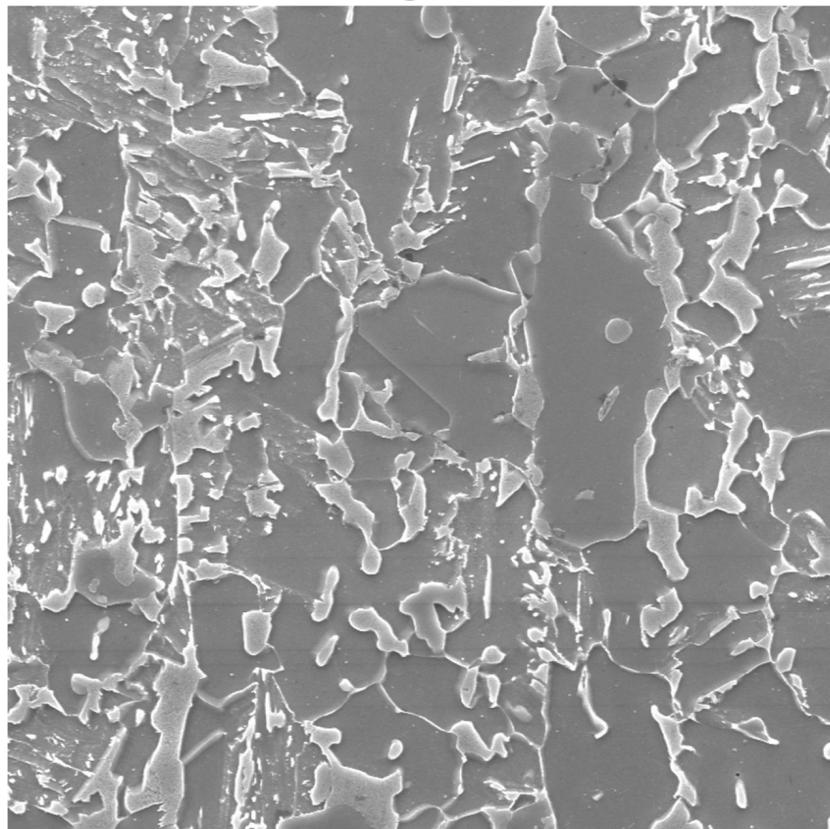
Accuracy Percentage (Dice Coeff)

Epochs - 200
Train Image Size - 800x800
Test Image Size - 1280x1024

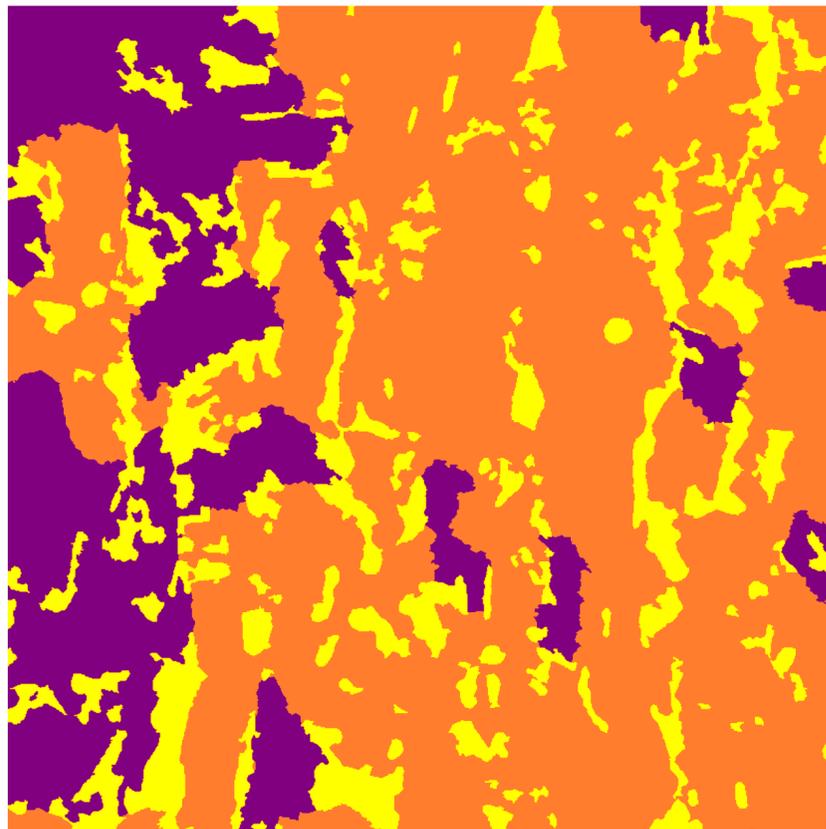
Inference Images

Test Image image 6.bmp

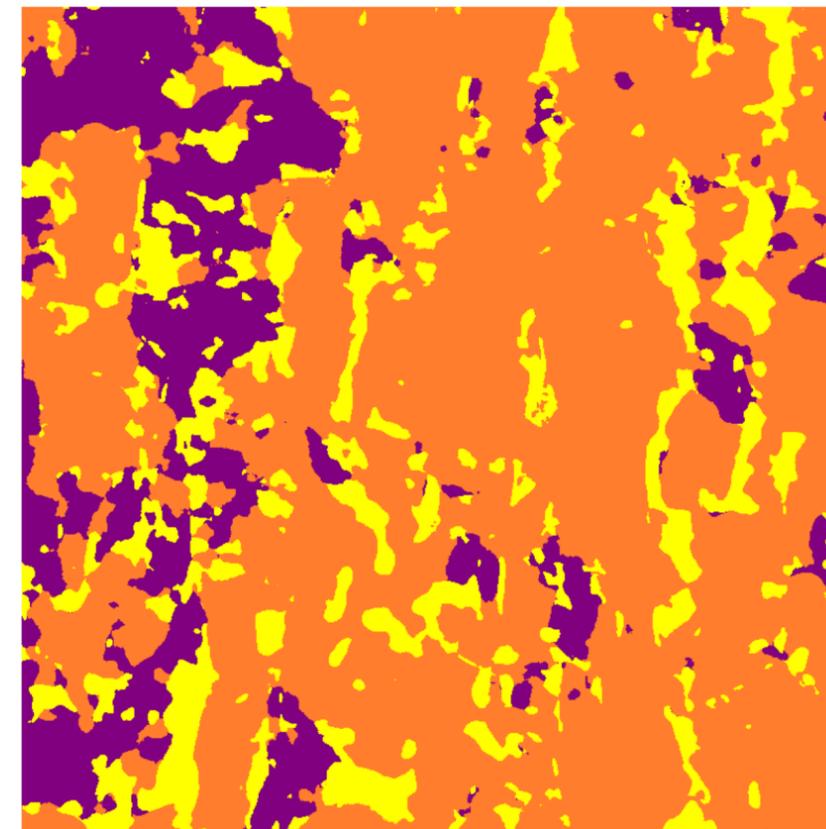
Original



Label



Prediction



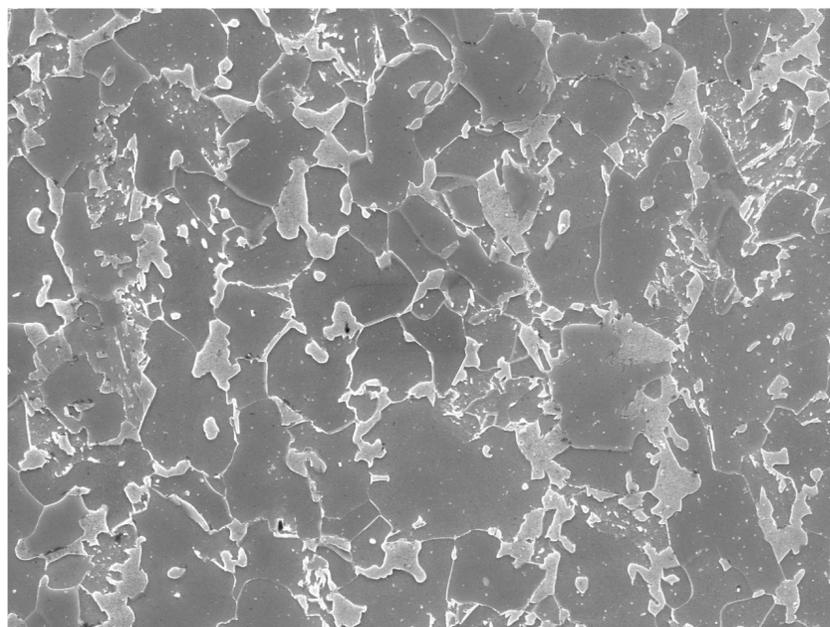
Accuracy - 87.98

Inference Images

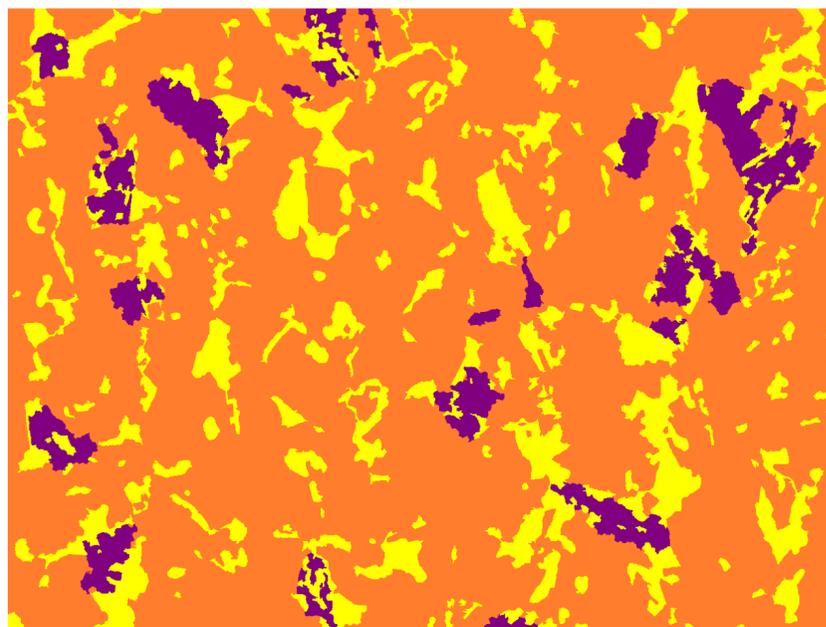
x3000

image 4.bmp

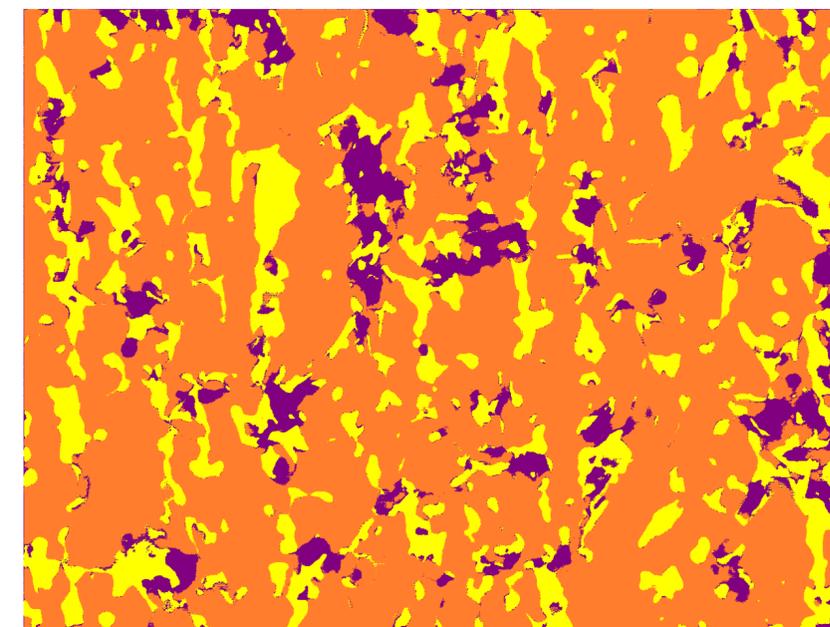
Original



Label



Prediction



Accuracy – 84.8



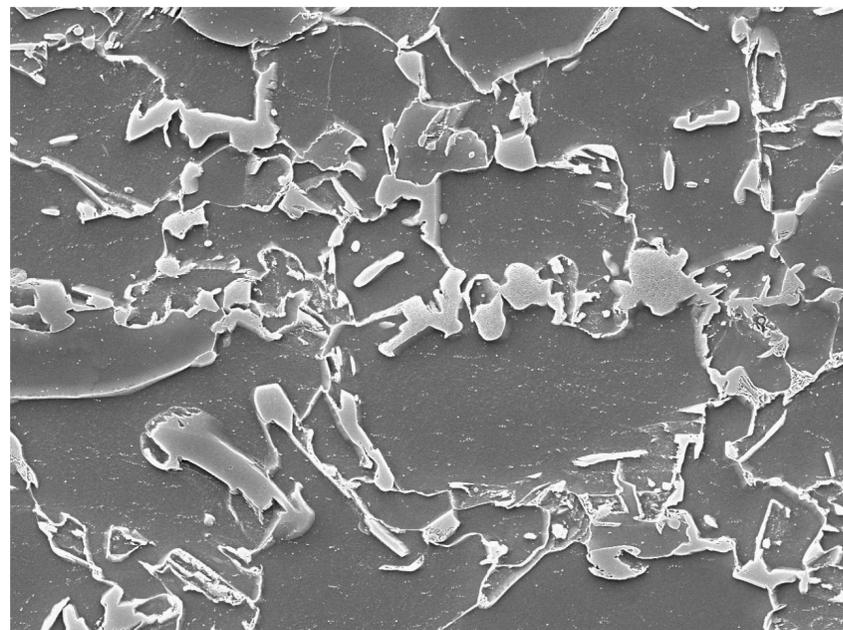
Accuracy/Class

Inference Images

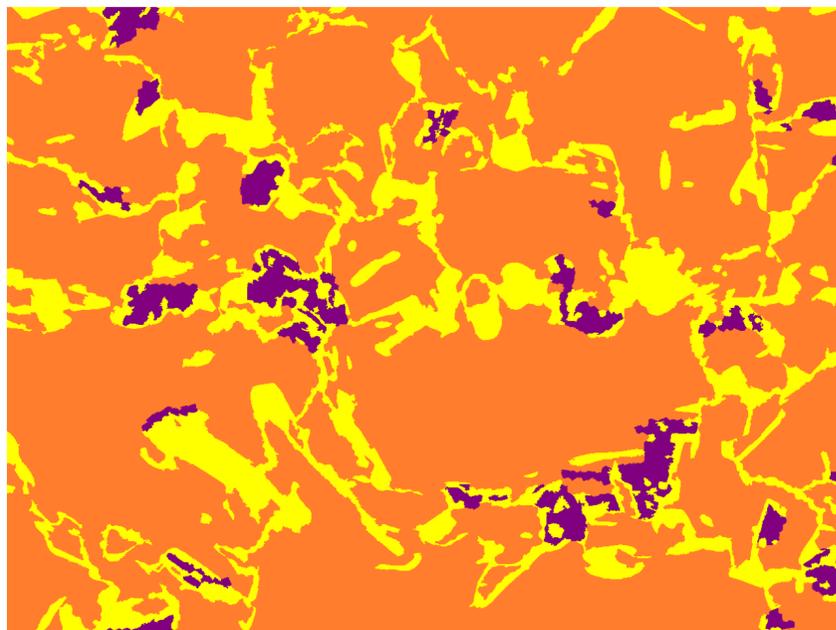
x5000

image 2.bmp

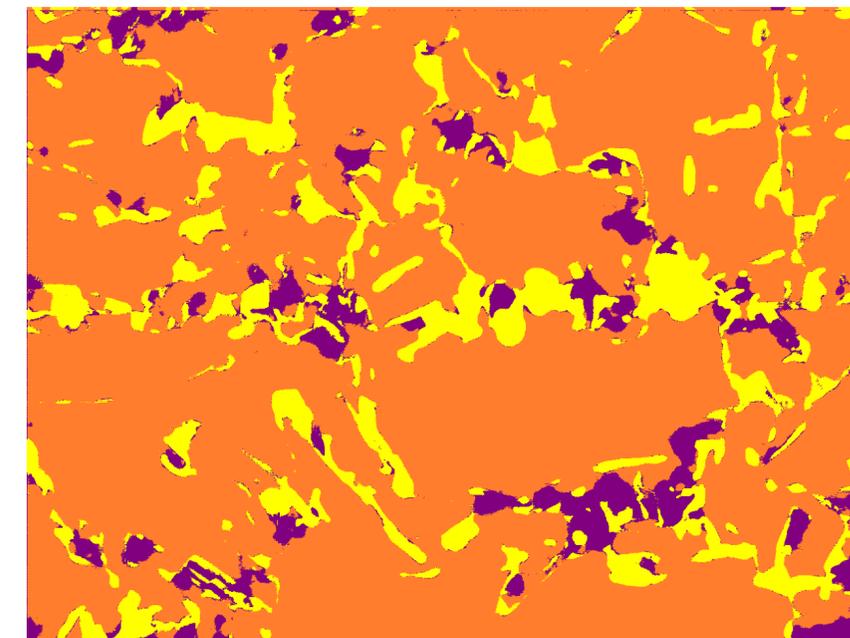
Original



Label



Prediction



Accuracy - 88.2

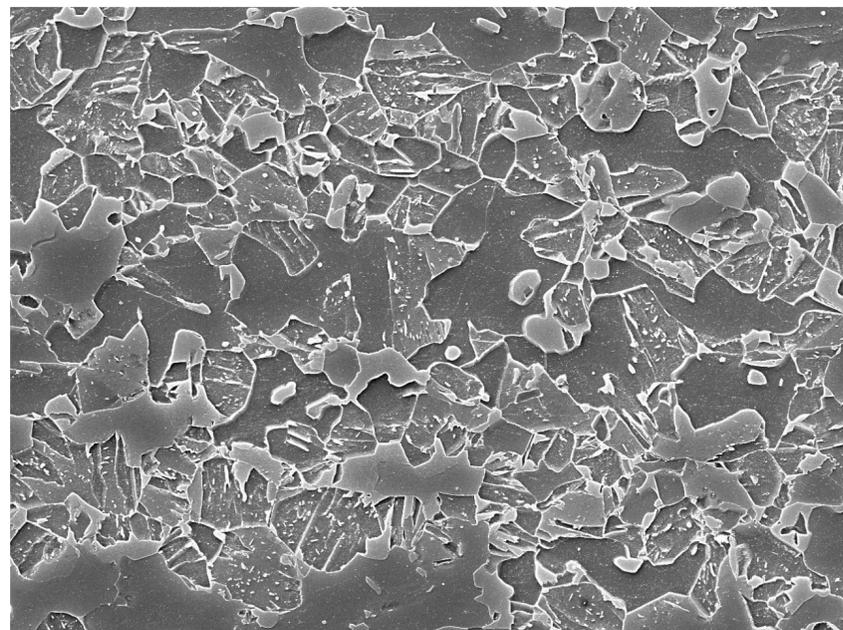


Accuracy/Class

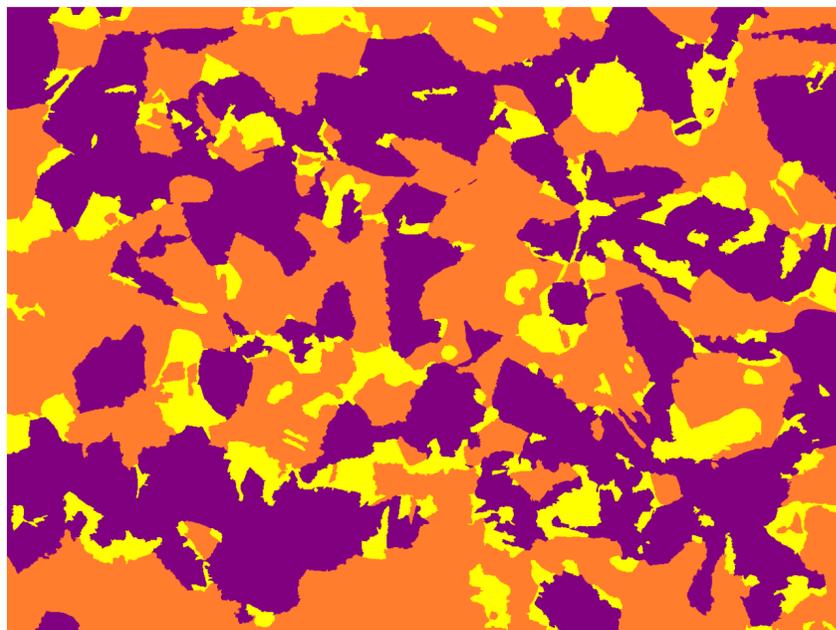
Inference Images

test1 A type

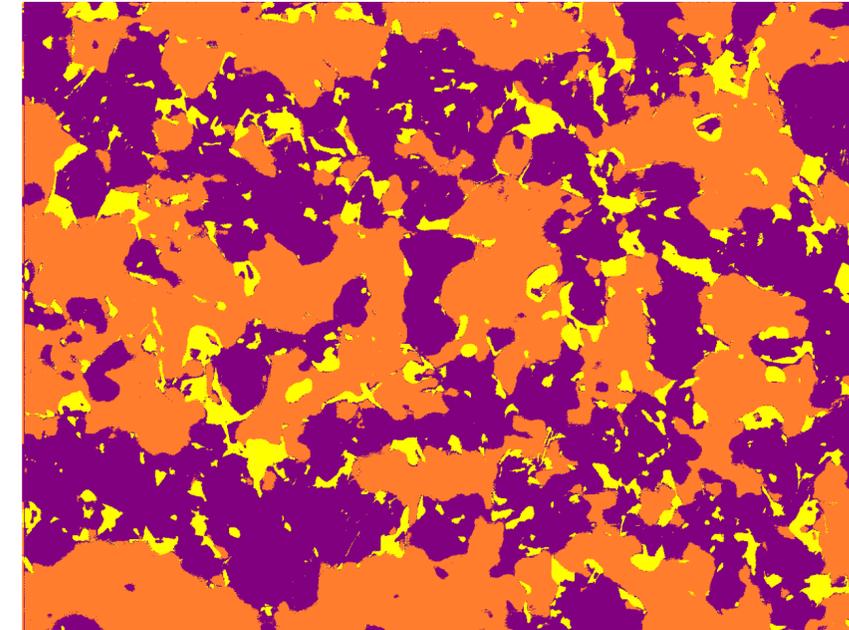
Original



Label



Prediction



Accuracy – 77.4

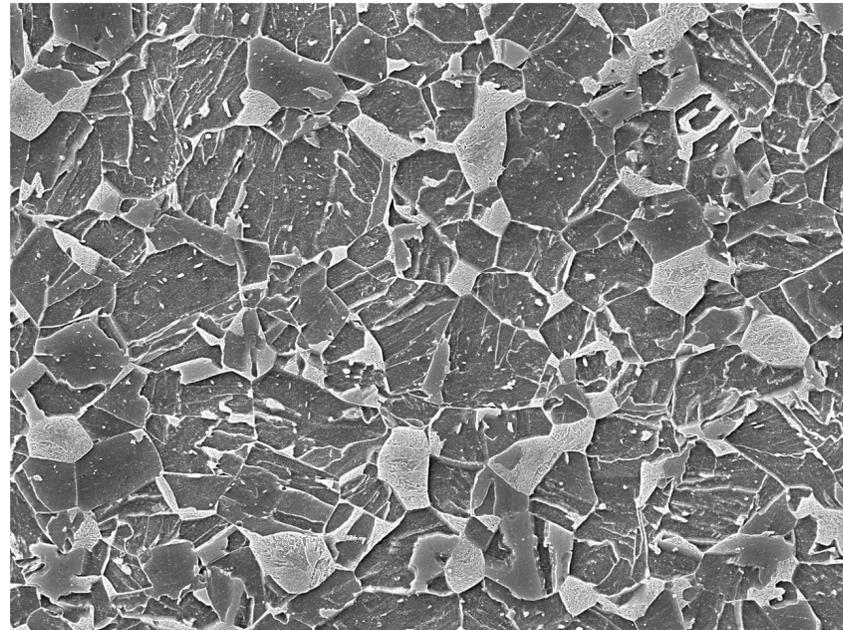


Accuracy/Class

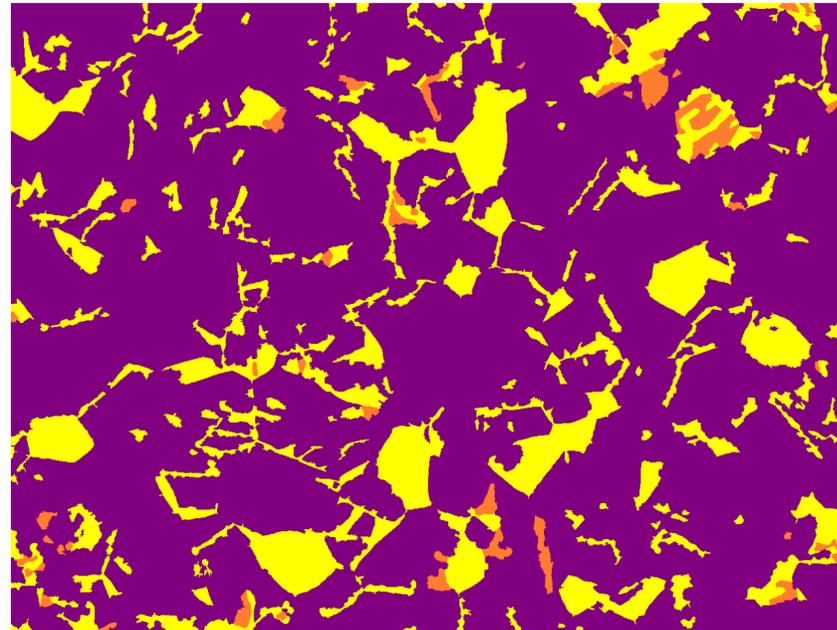
Inference Images

test2 D3 type

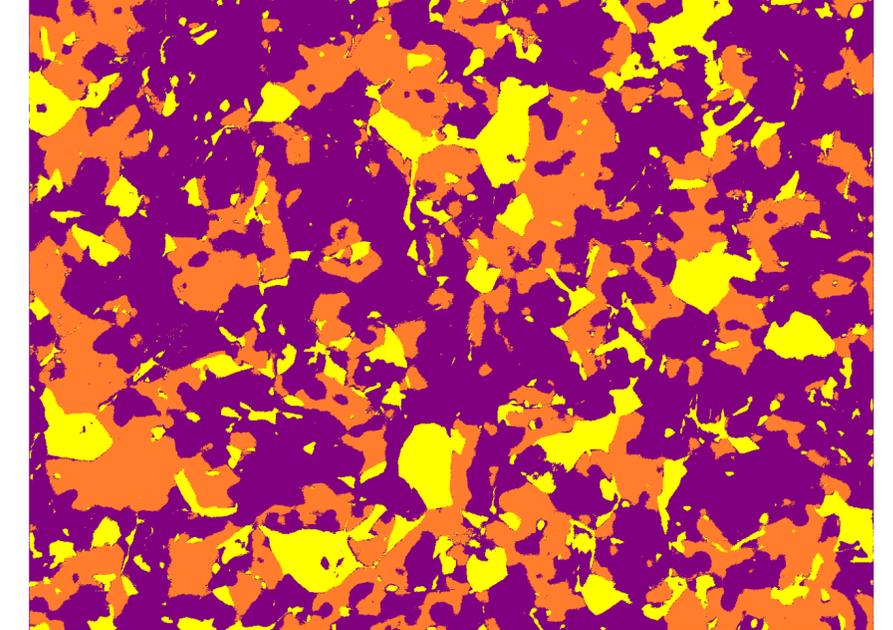
Original



Label



Prediction



Accuracy – 64.7

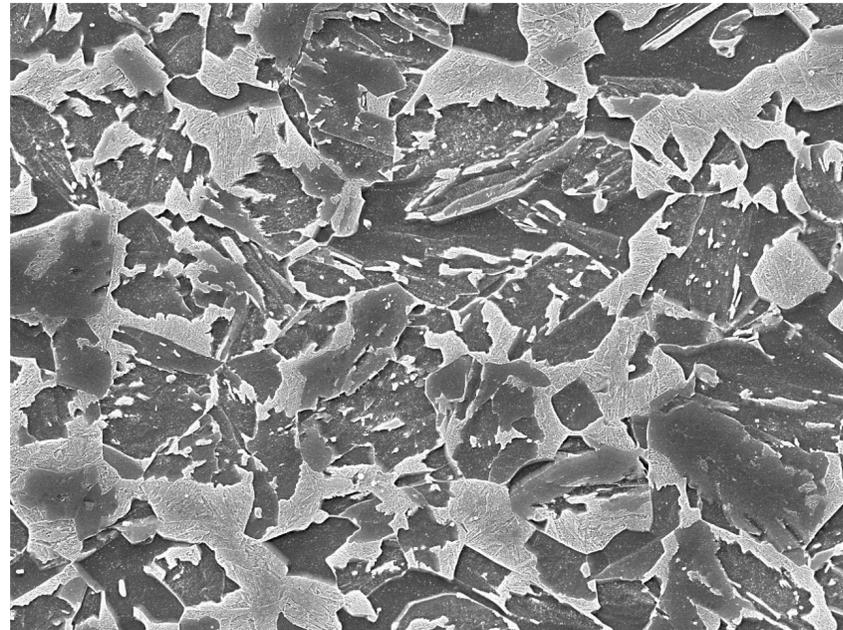


Accuracy/Class

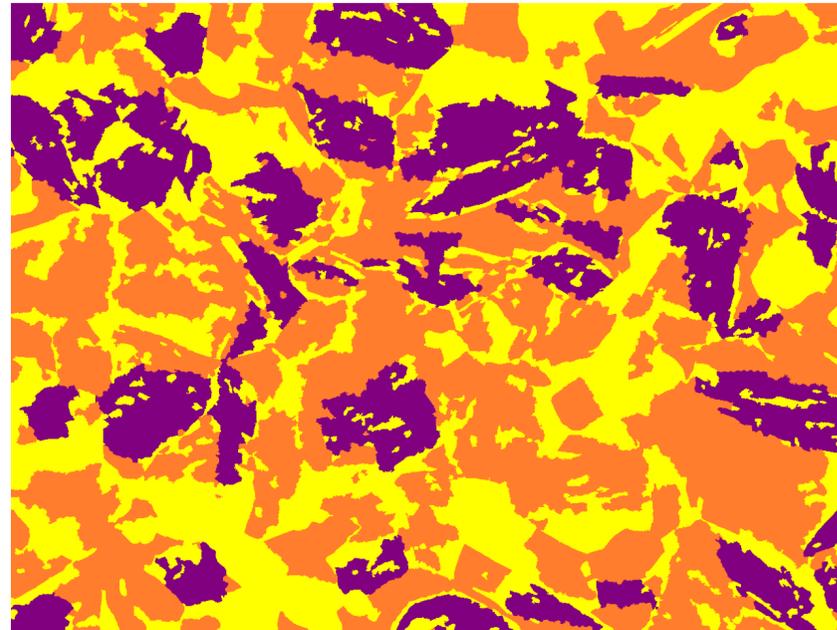
Inference Images

test3 H2 type

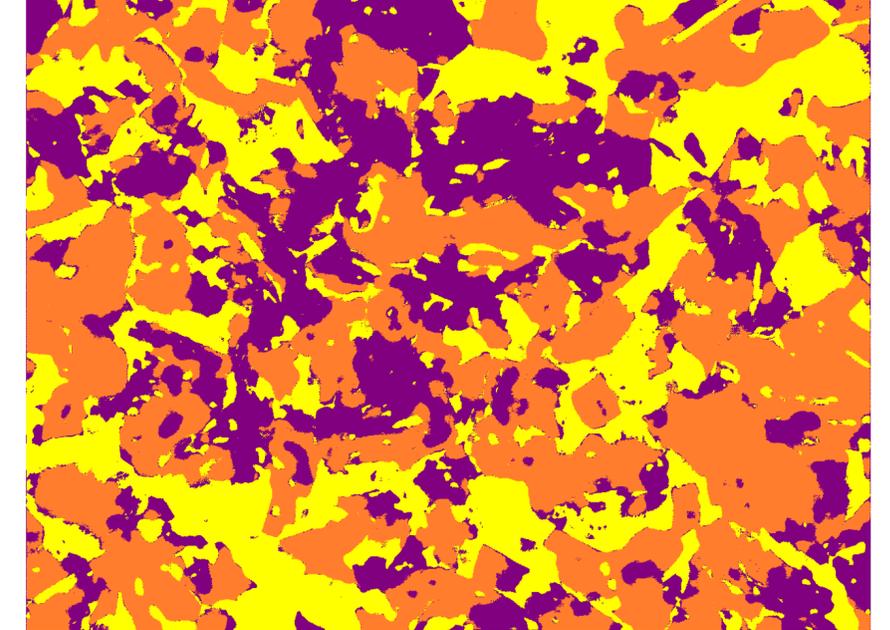
Original



Label



Prediction

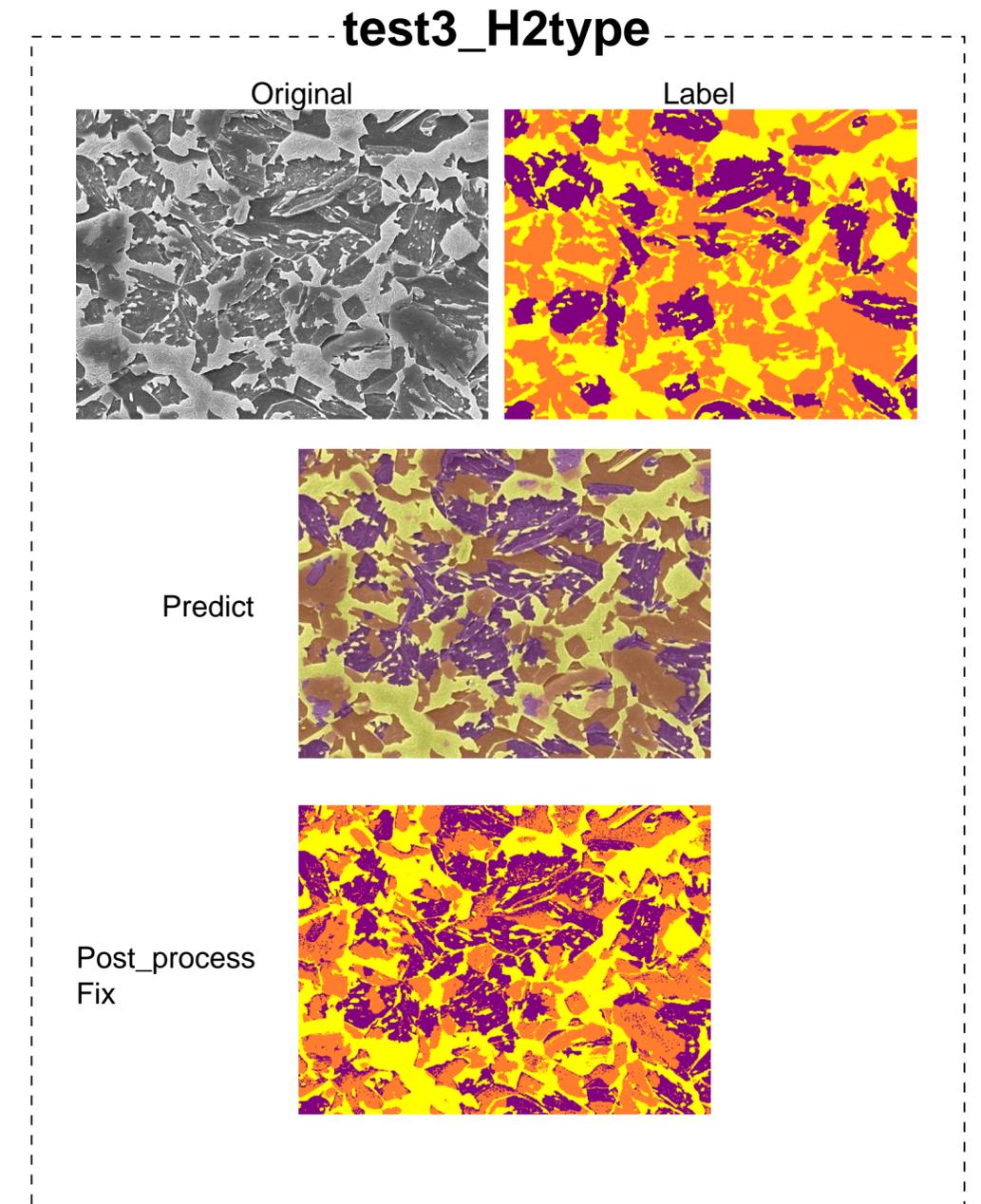
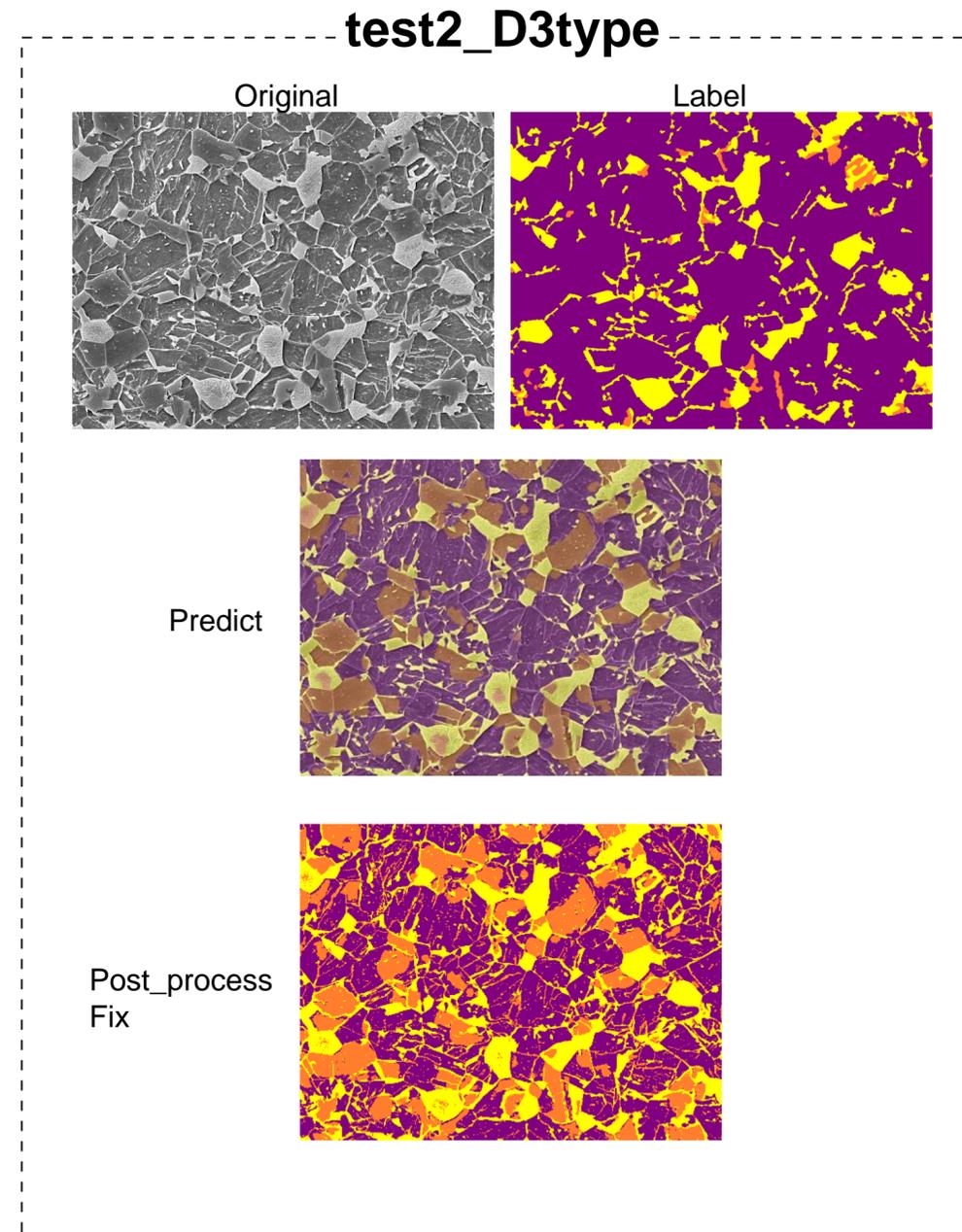
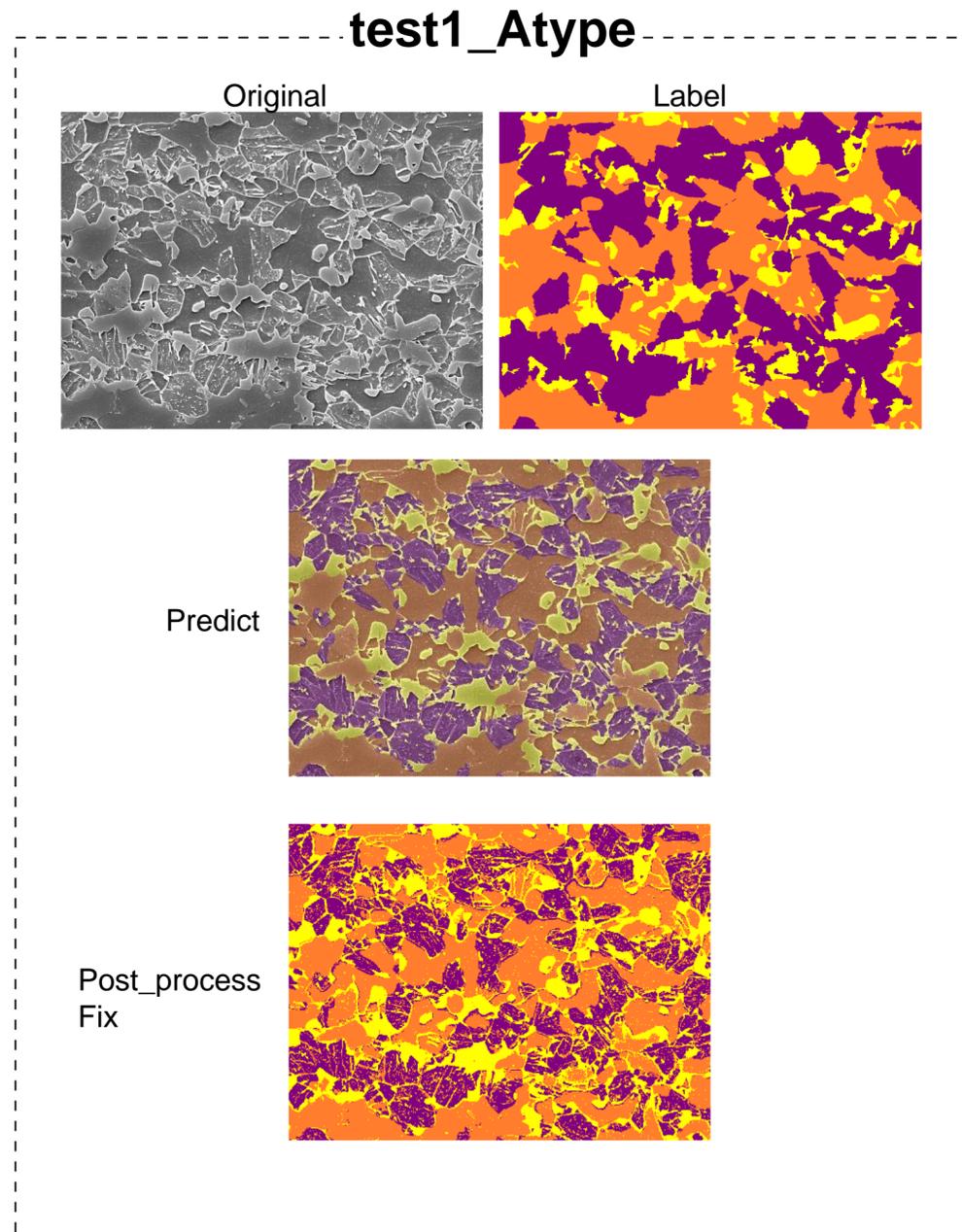


Accuracy – 73.8



Accuracy/Class

Baseline Outputs



Accuracy - 68.6

Dice - 0.6842

Accuracy - 62.92

Dice - 0.6267

Accuracy - 73.57

Dice - 0.7332

83.48 76.56 55.62

Acc/Color

25.07 44.85 30.08

Ratio

86.72 26.86 58.61

Acc/Color

16

25.07 44.85 30.08

Ratio

87.04 63.25 72.96

Acc/Color

25.07 44.85 30.08

Ratio

1. Activation Functions

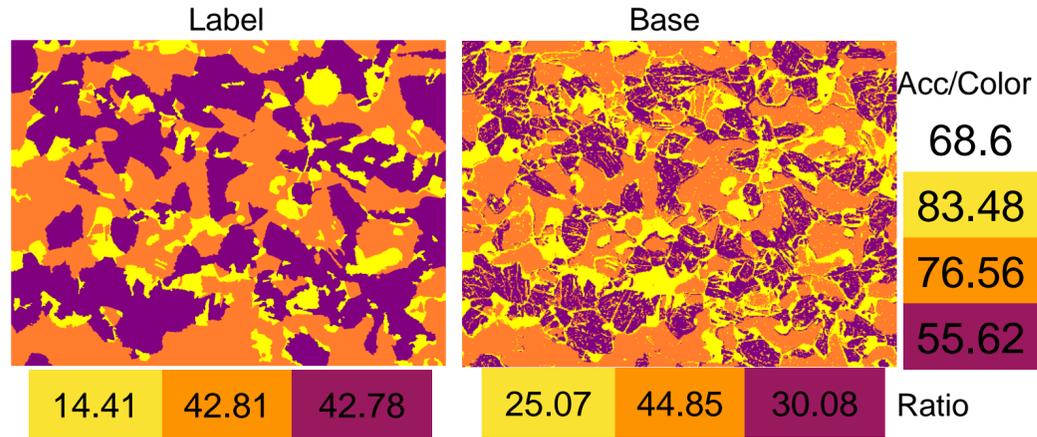
Experimented with various activation functions to find the best suited activation function.

	Avg (10)	Avg (8)	(1)	(1)	(1)	None
	x3000	x5000	test1 A type	test2 D3 type	test3 H2 type	
Siren	86.12 (0.8487) 63.72 96.44 31.28	89.77 (0.8852) 66.23 97.78 34.78	74.87 (0.7383) 47.56 90.92 58.72	40.02 (0.3998) 60.69 67.79 33.98	69.8 (0.6955) 72.51 90.25 24.6	
ReLU	83.97 (0.8372) 60.91 91.41 55.1	90.01 (0.8976) 74.33 94.19 62.19	77.41 (0.7716) 49.97 81.79 82.27	64.75 (0.6450) 73.98 32.14 63.51	73.89 (0.7364) 81.51 73.96 62.44	
LeakyReLU	84.77 (0.8453) 66.19 94.85 49.93	88.53 (0.8831) 62.8 97.16 38.47	75.19 (0.7493) 43.32 91.42 69.69	59.79 (0.5955) 74.22 41.21 59.79	69.69 (0.6944) 70.4 86.41 37.05	
Swish	84.21 (0.8396) 63.99 95.29 44.86	88.67 (0.8842) 63.36 97.36 36.21	72.41 (0.7217) 46.51 92.39 61.15	56.74 (0.5649) 77.34 47.08 52.52	70.00 (0.6976) 66.89 90.23 36.41	

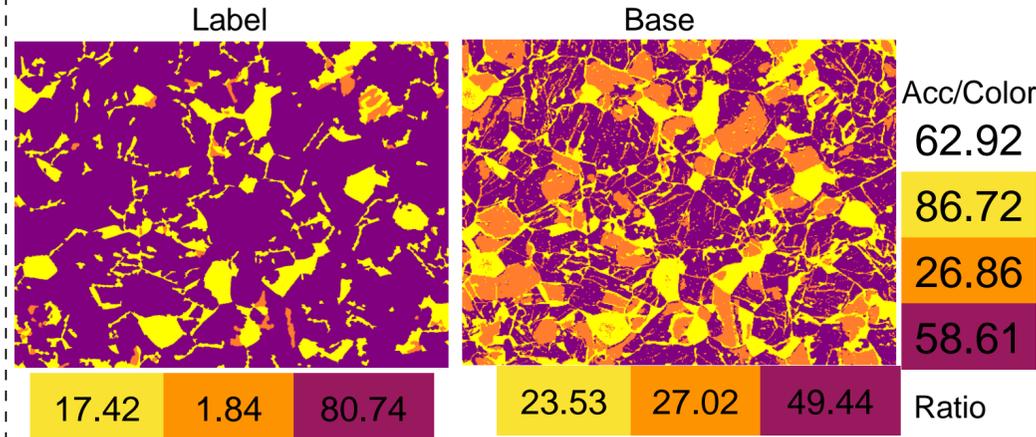
Accuracy Percentage (DICE)

Comparisons of Inference Images

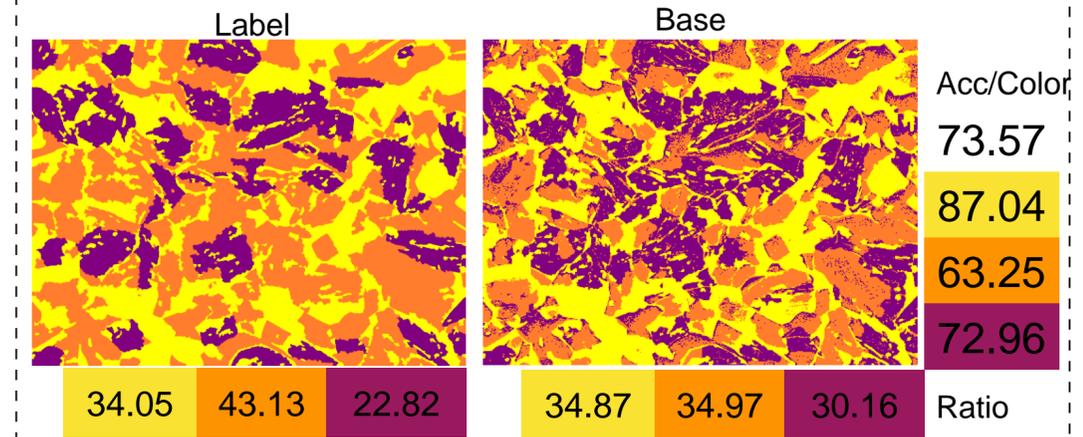
test1_Atype



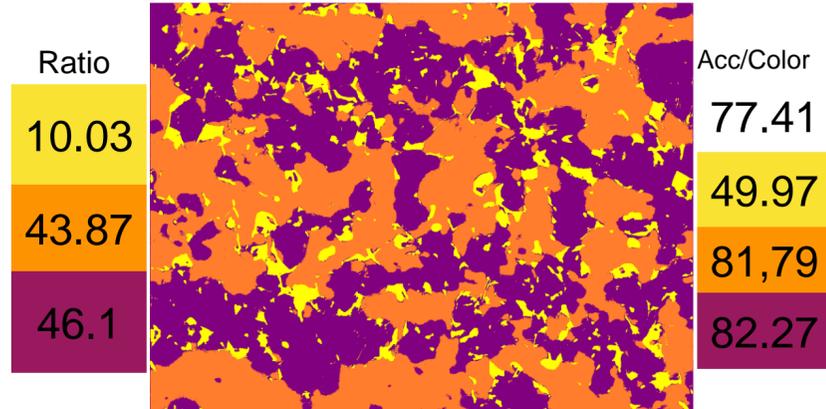
test2_D3type



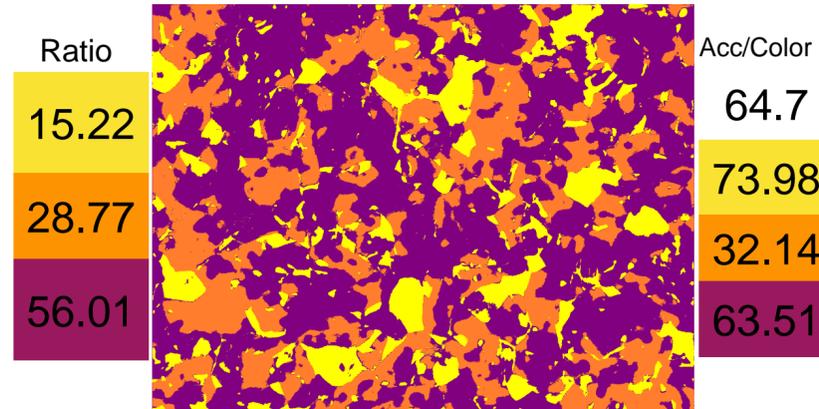
test3_H2type



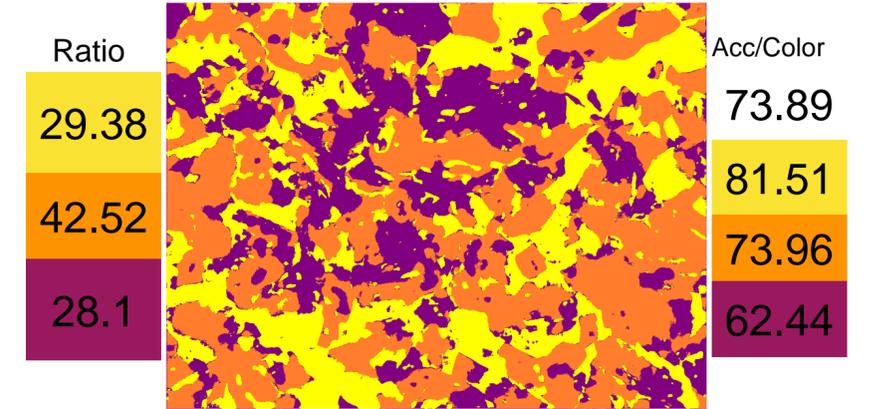
ReLU+Focal



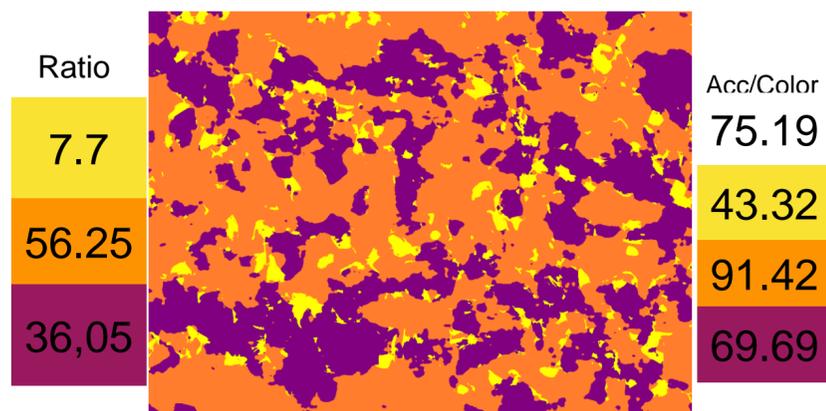
ReLU+Focal



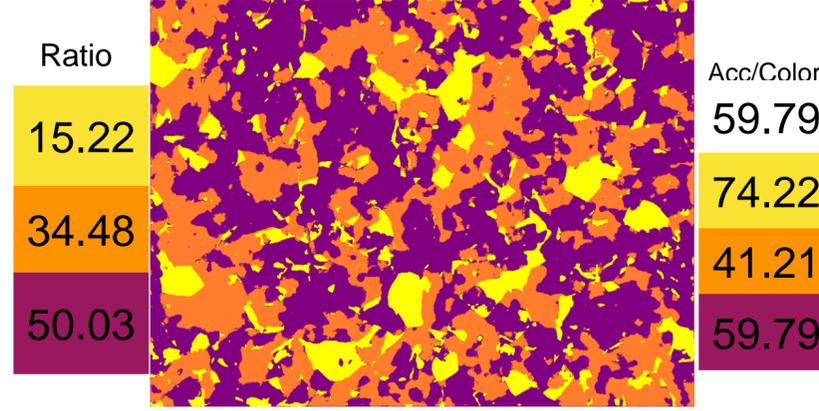
ReLU+Focal



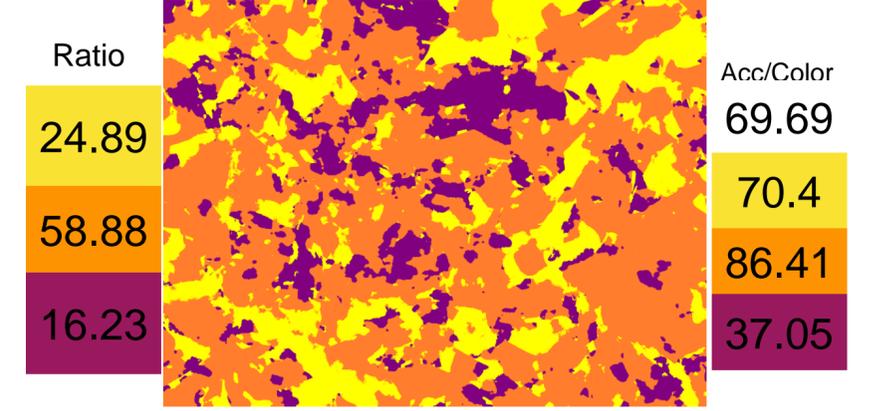
LeakyReLU+Focal



LeakyReLU+Focal

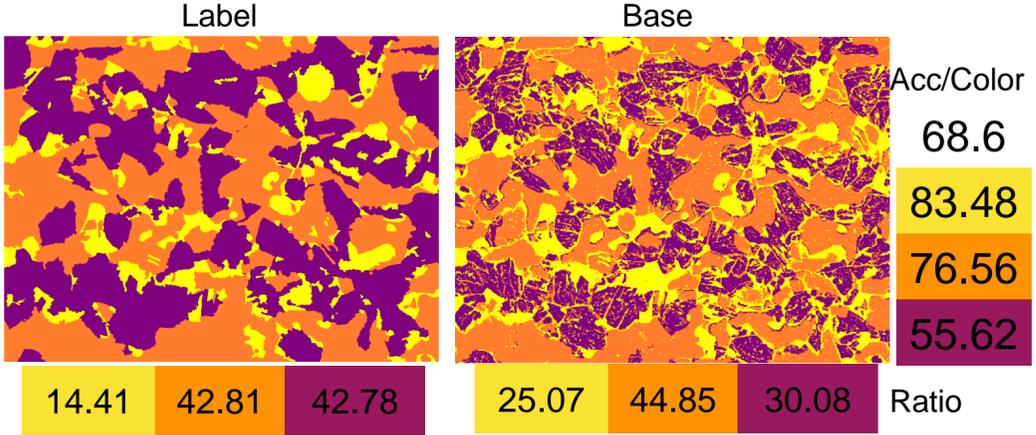


LeakyReLU+Focal

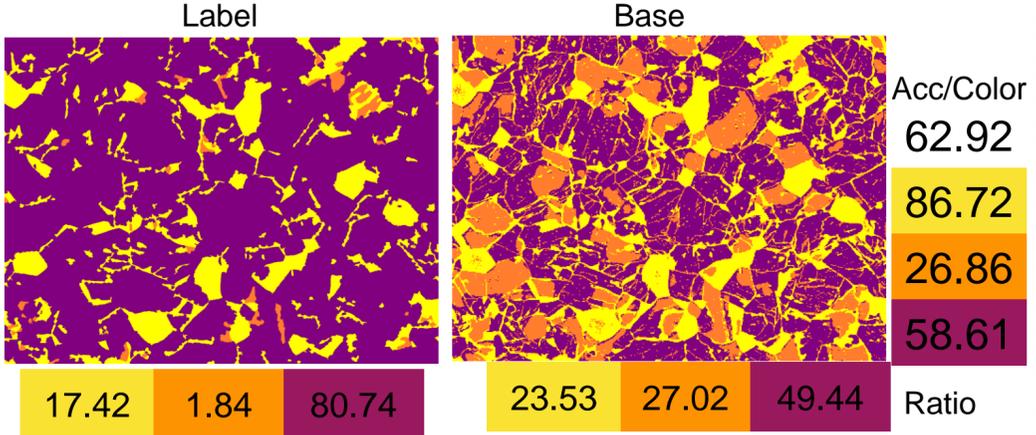


Comparisons of Inference Images

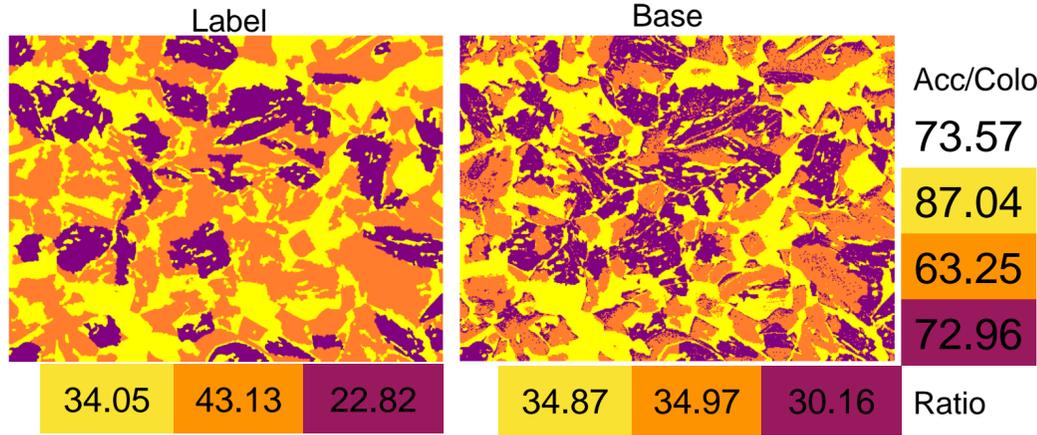
test1_Atype



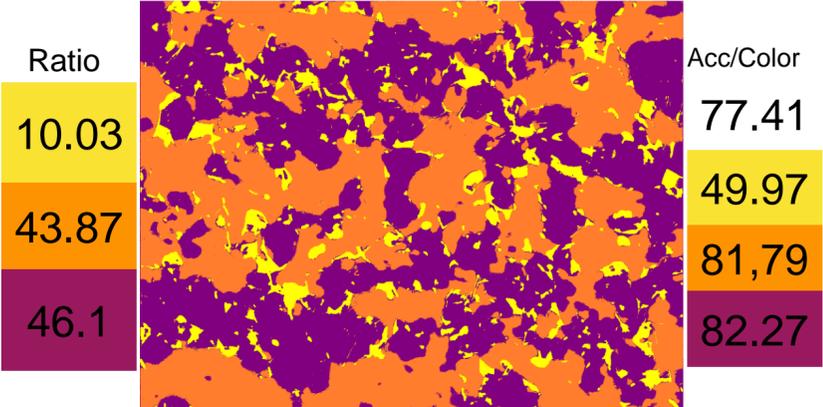
test2_D3type



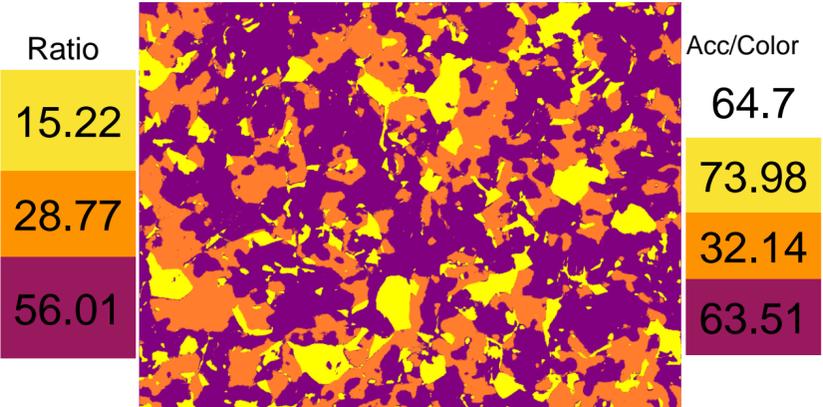
test3_H2type



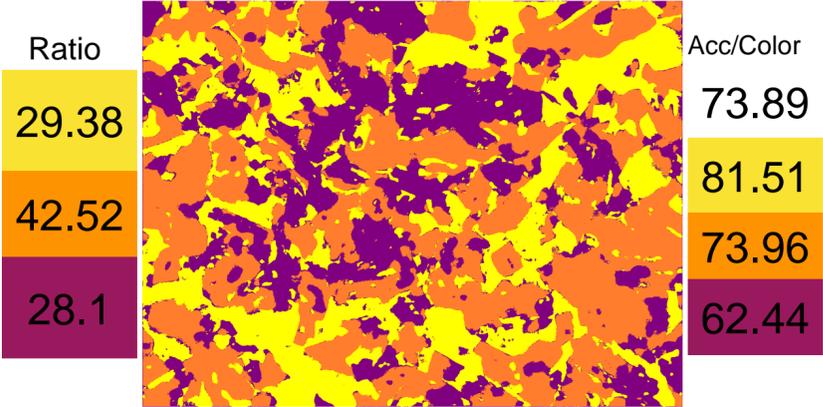
ReLU+Focal



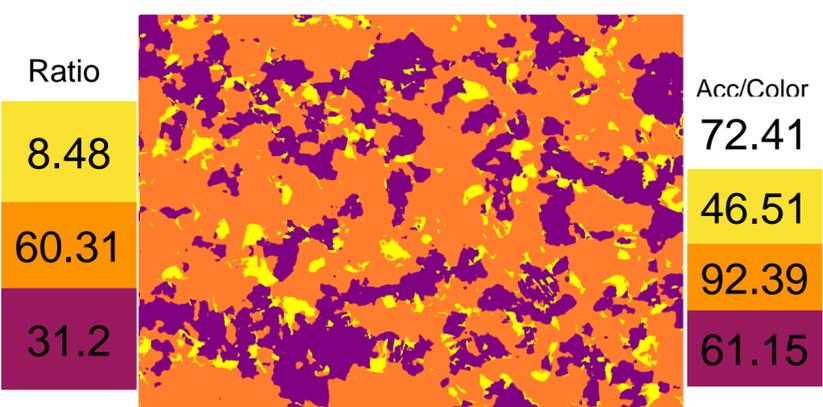
ReLU+Focal



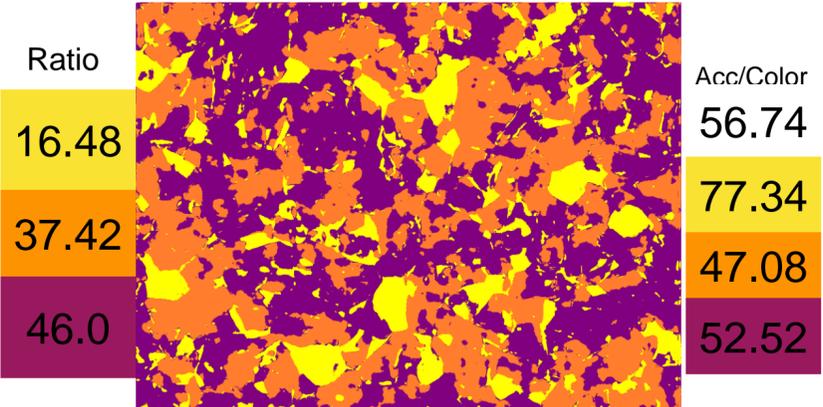
ReLU+Focal



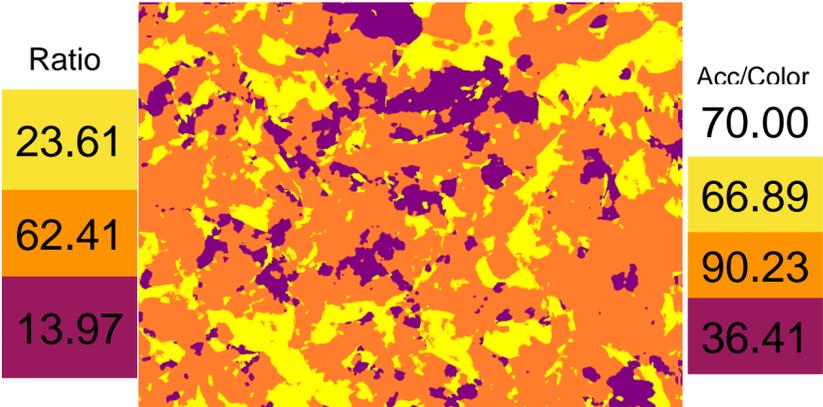
Swish+Focal



Swish+Focal



Swish+Focal



2. Effects of Blur Pooling

Experimented with 4 types of Blur Pooling in the model as it showed to predict the micro structures better in previous related works.

- Down Blur Pooling
- Up Blur Pooling
- Up and Down Blur Pooling

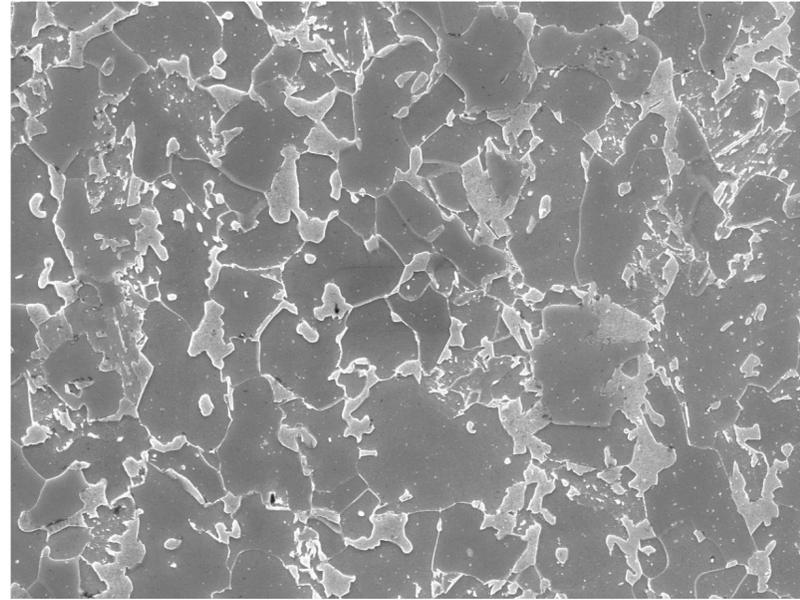
Blur Pooling Inference Results

Inference Images

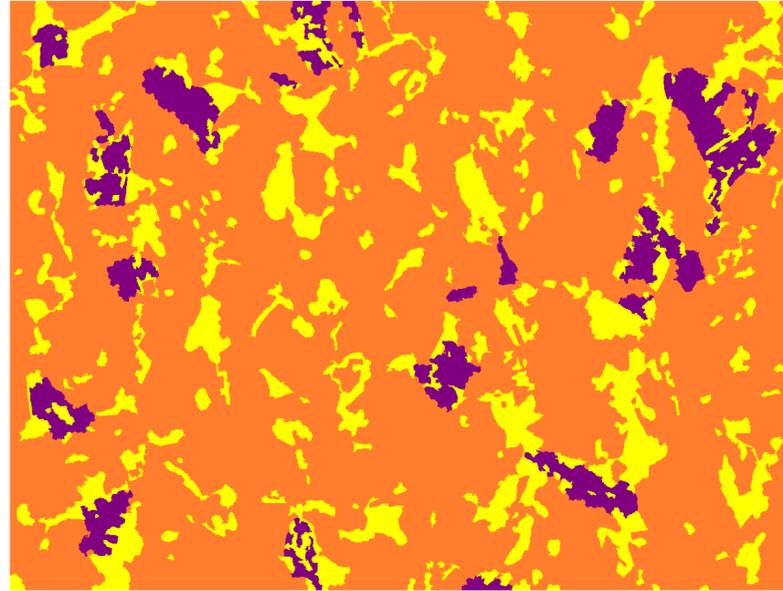
x3000

image 4.bmp

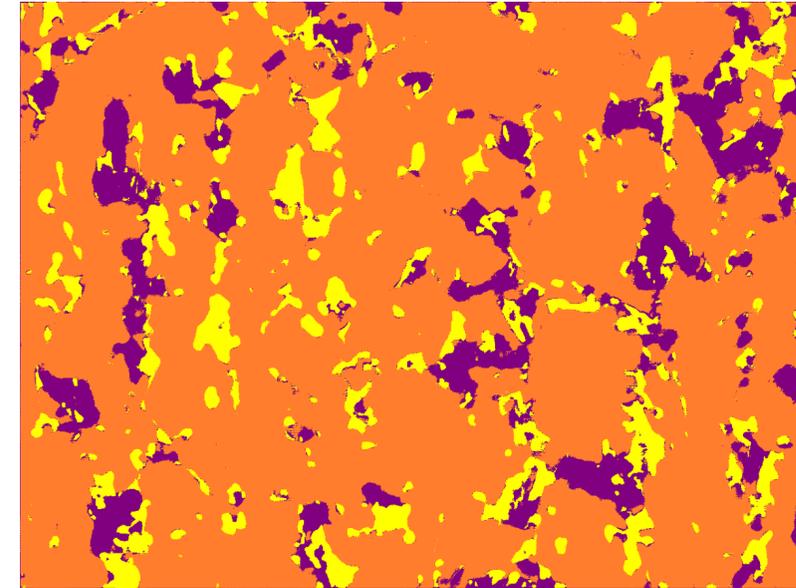
Original



Label



Prediction



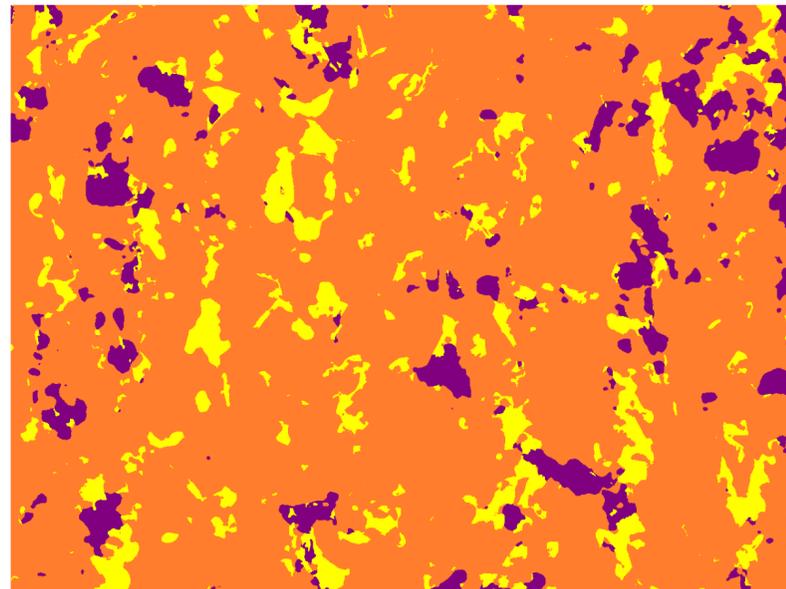
ReLU None

59.71
91.6
65.57

Accuracy - 84.2

Prediction

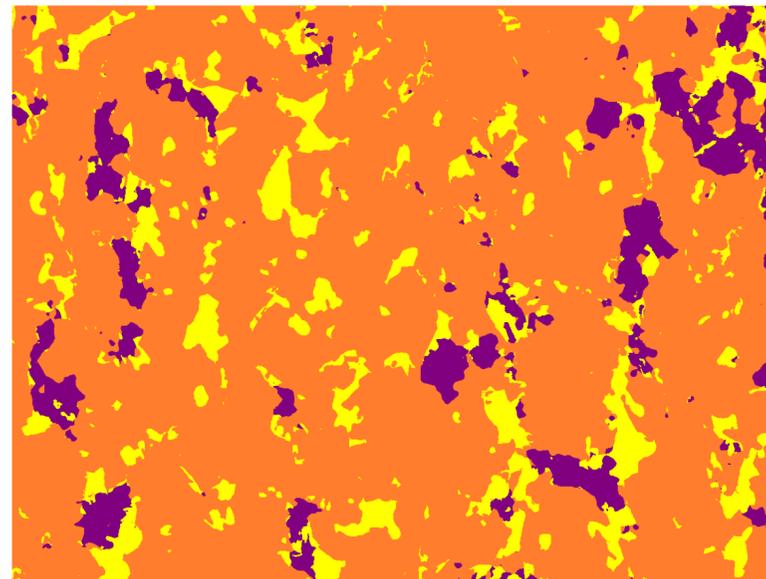
Down



58.82
95.21
52.19

Accuracy - 85.9

Up

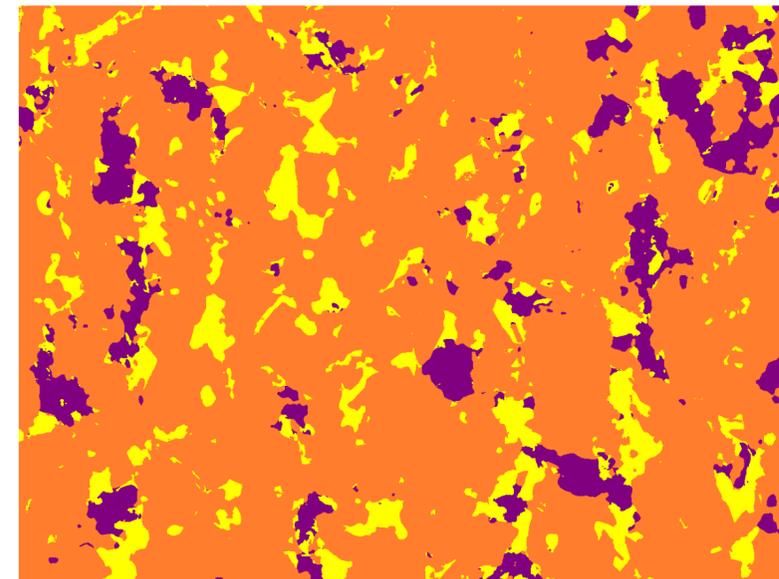


67.25
94.53
60.63

Accuracy - 87.5

21

DownUp



67.2
94.4
60.97

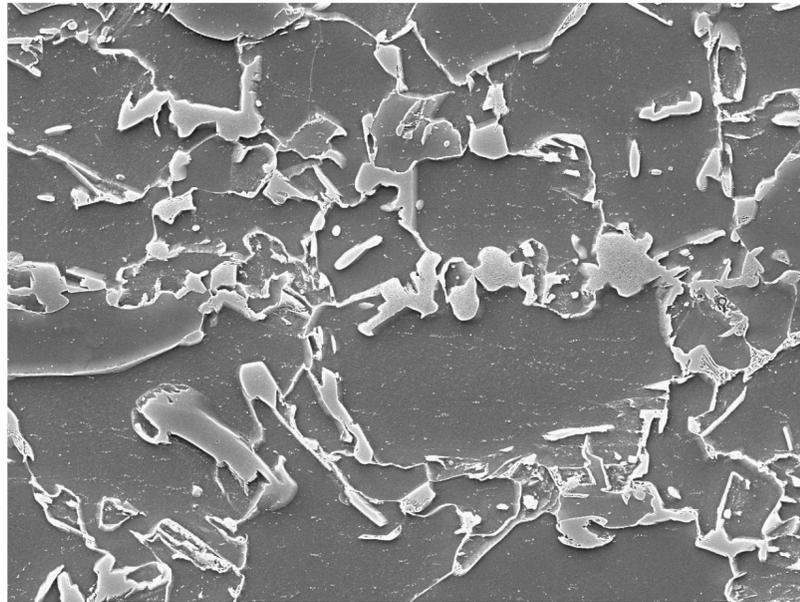
Accuracy - 87.4

Inference Images

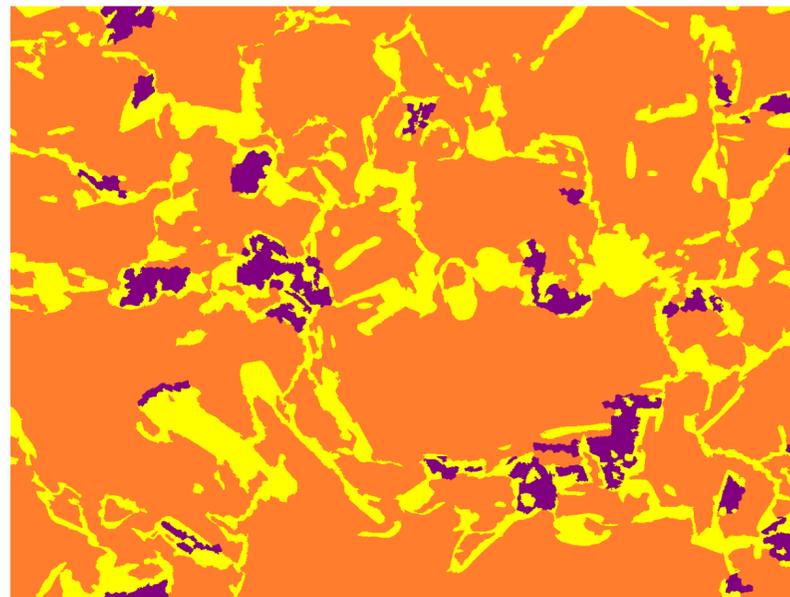
x5000

image 2.bmp

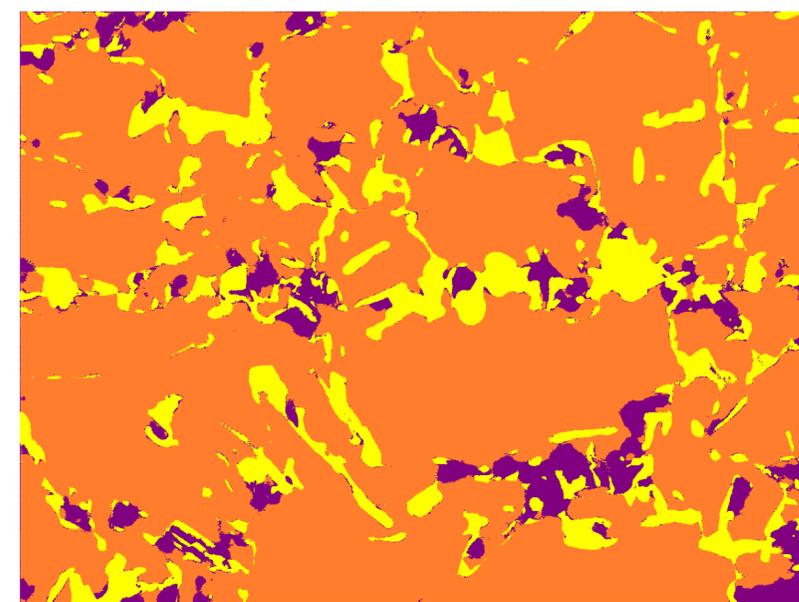
Original



Label



Prediction



ReLU

None

71.57

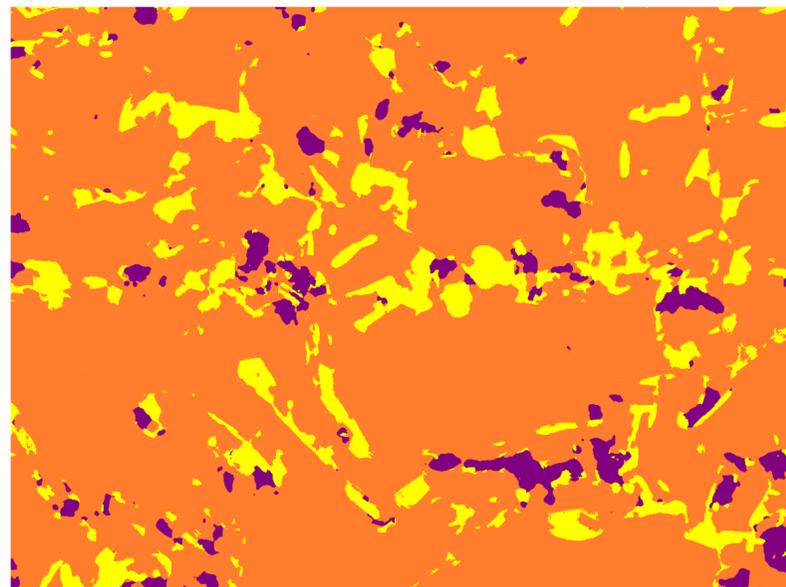
94.36

57.32

Accuracy - 88.1

Prediction

Down



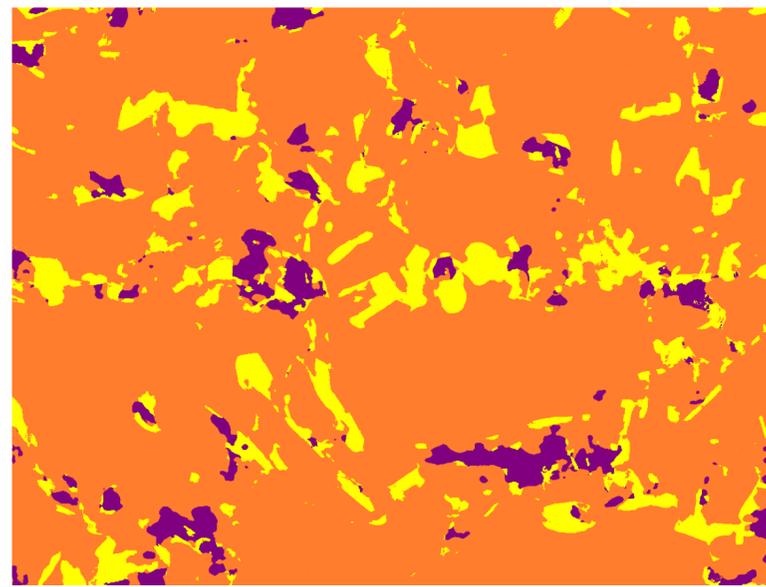
59.73

97.57

38.94

Accuracy - 87.5

Up



59.04

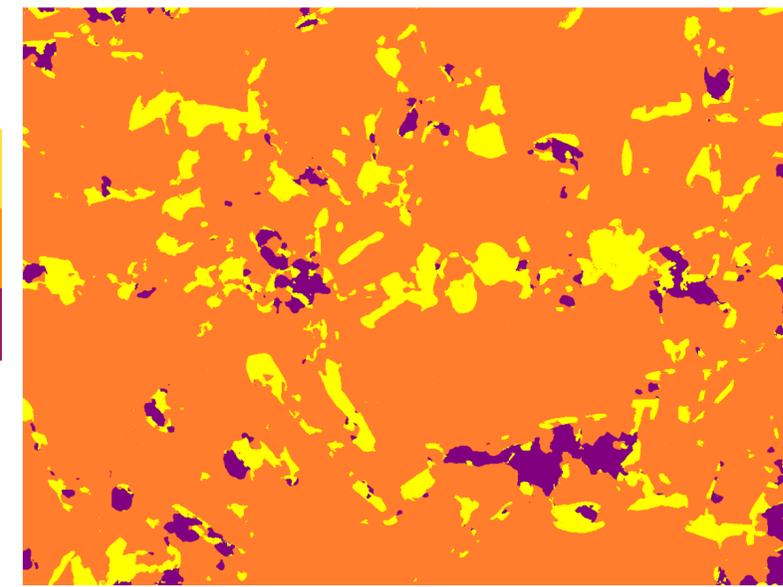
97.27

40.64

Accuracy - 87.2

22

DownUp



57.2

97.61

31.89

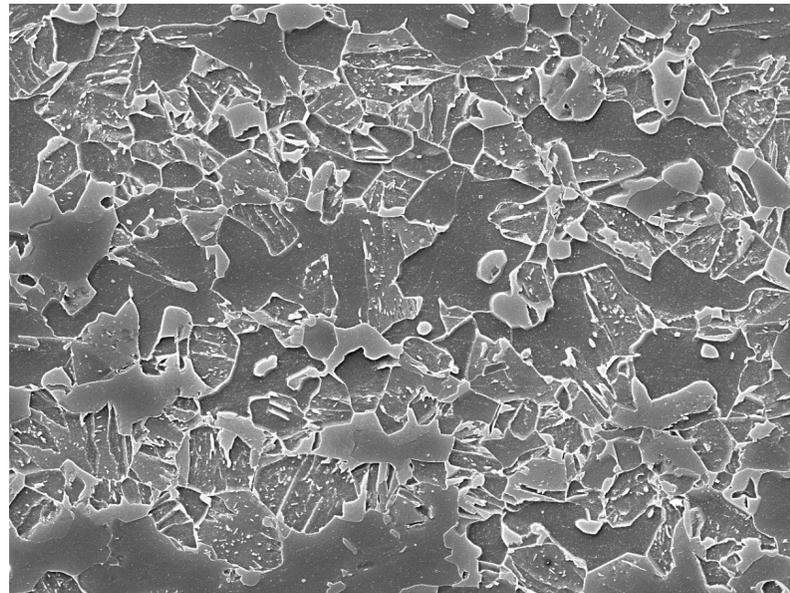
Accuracy - 86.7

Inference Images

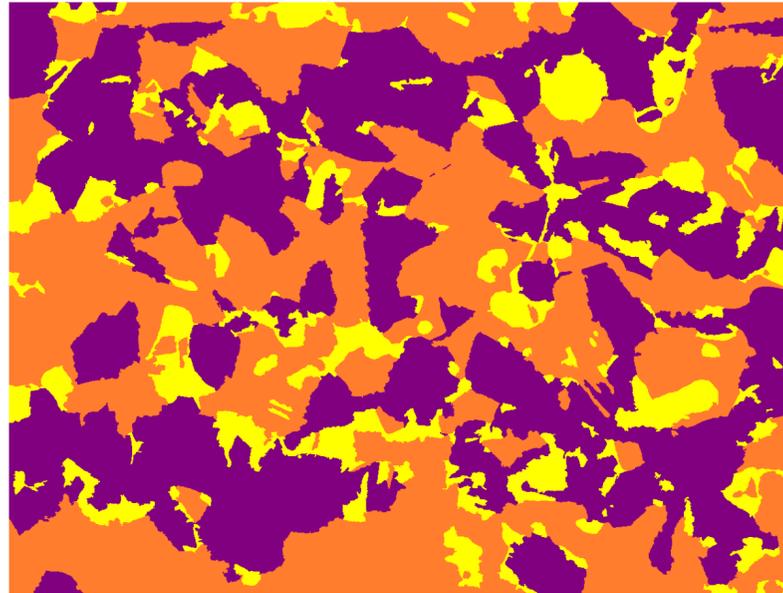
test1 A type

image 0.bmp

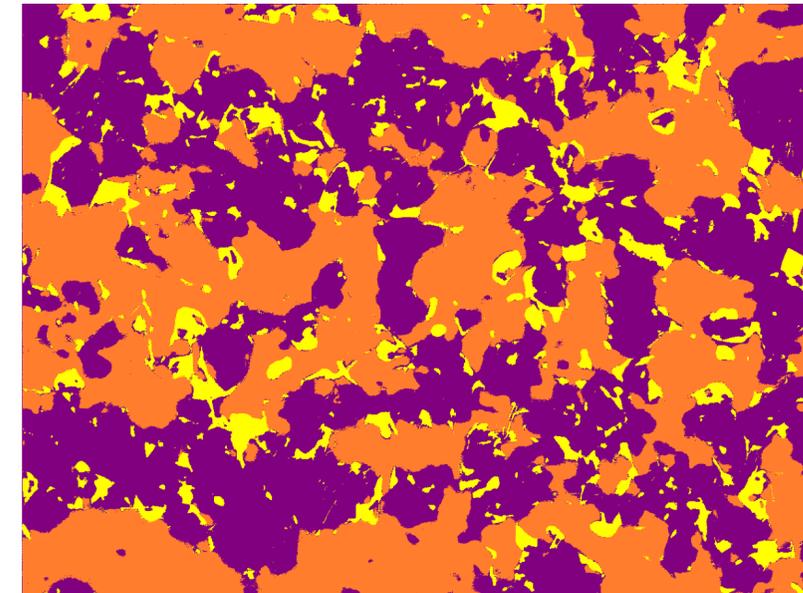
Original



Label



Prediction



ReLU

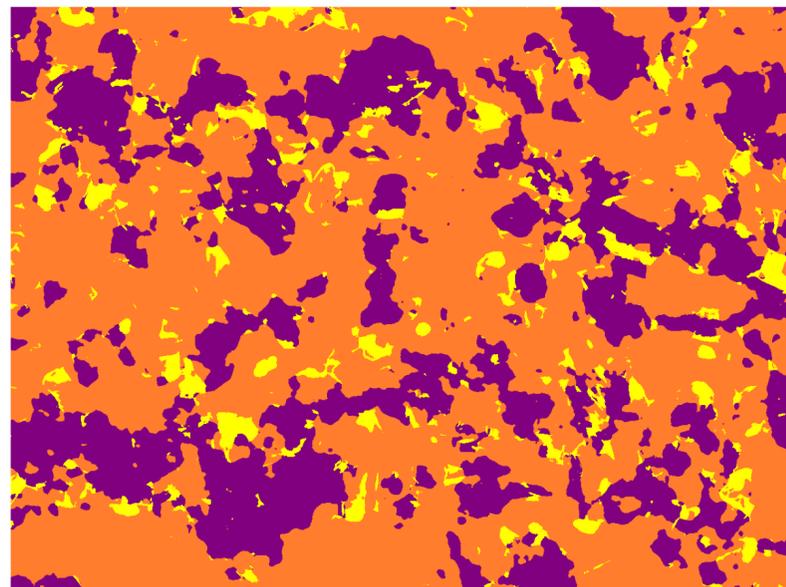
None

49.97
81.79
82.27

Accuracy - 77.4

Prediction

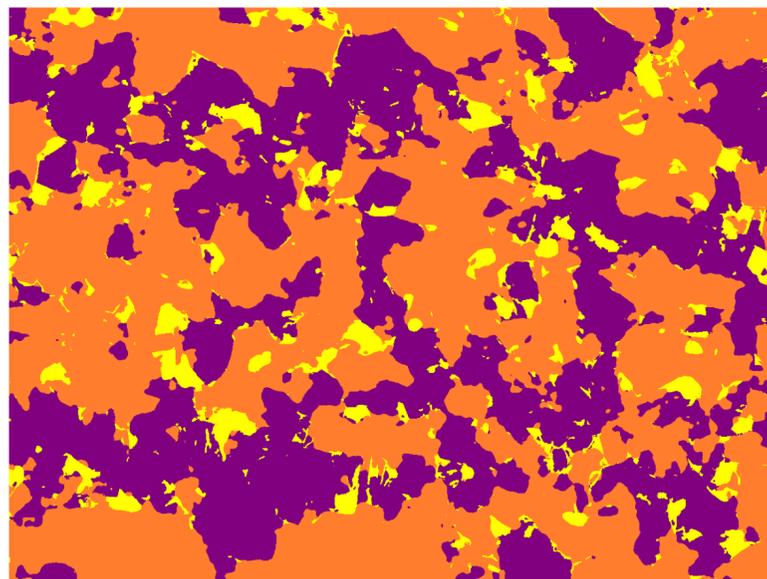
Down



39.52
93.37
58.82

Accuracy - 70.8

Up

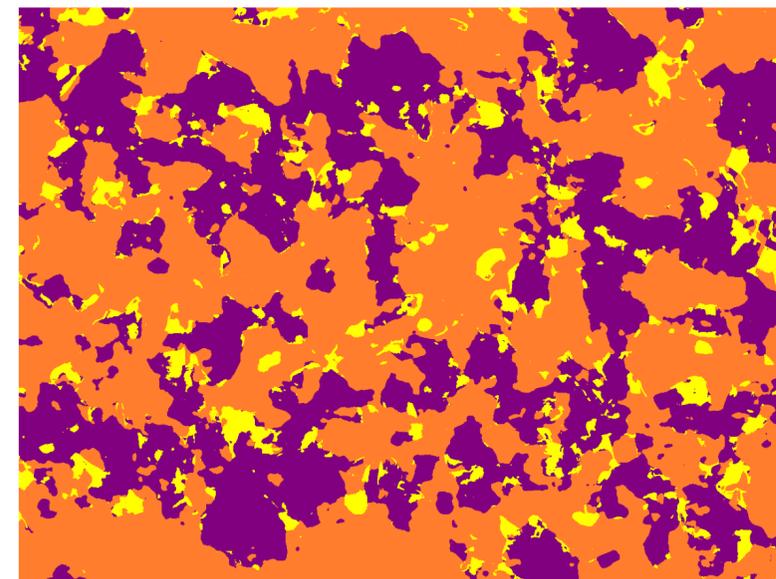


47.46
89.86
73.47

Accuracy - 76.7

23

DownUp



44.11
90.72
68.56

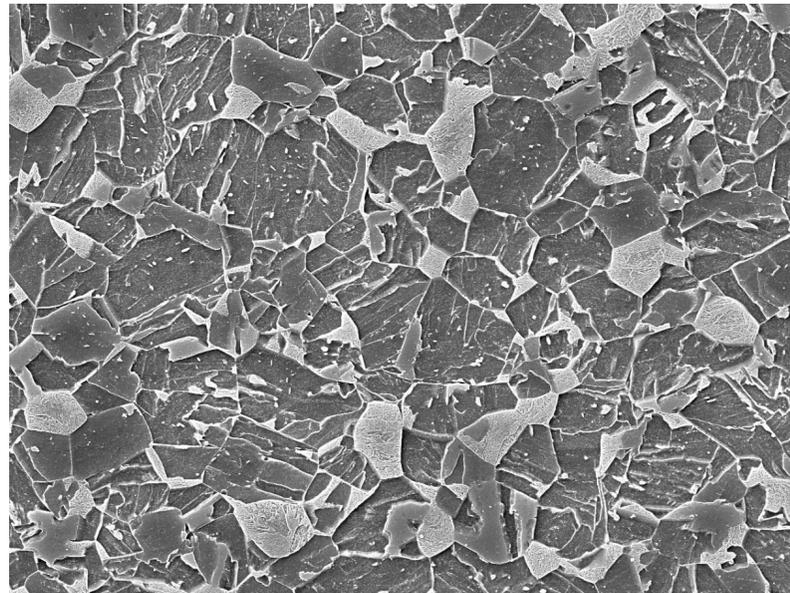
Accuracy - 74.5

Inference Images

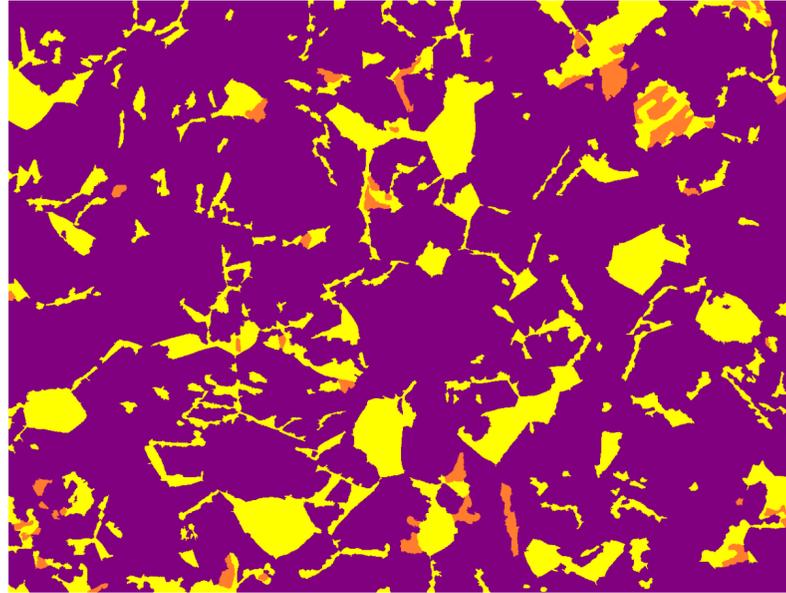
test2 D3 type

image 0.bmp

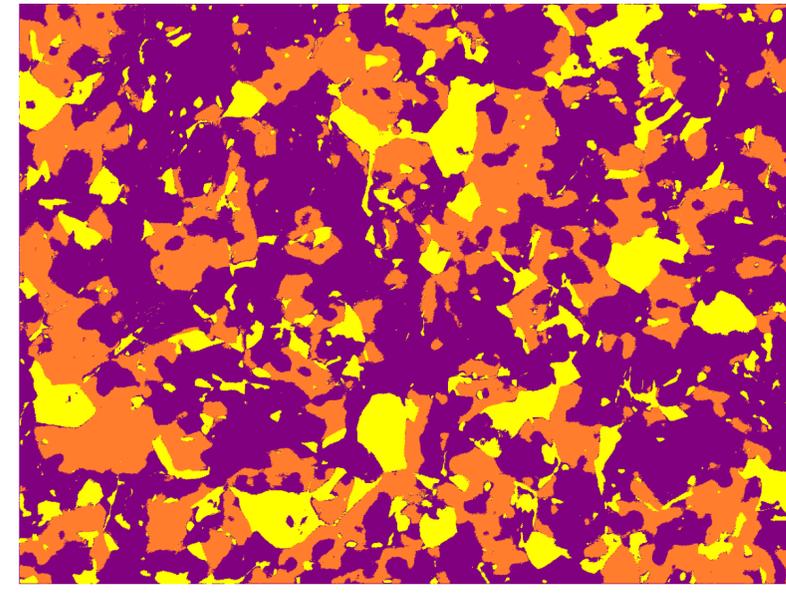
Original



Label



Prediction



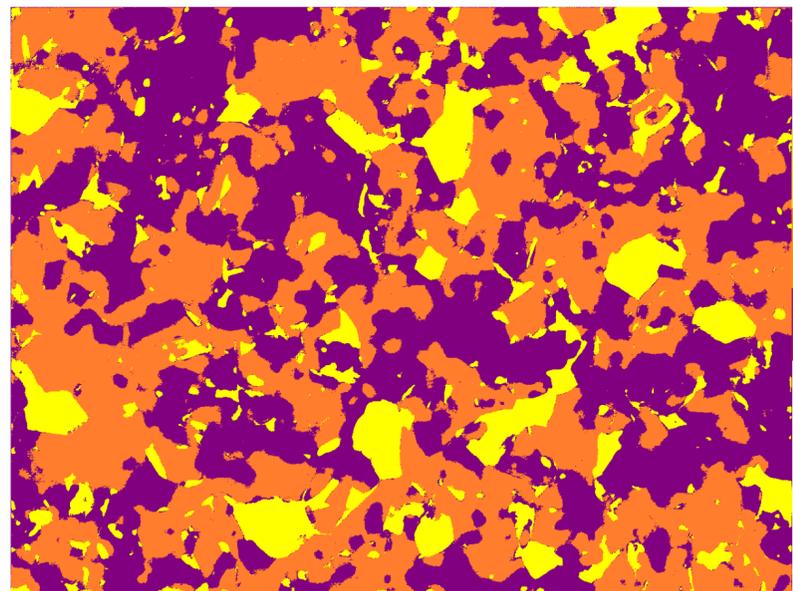
ReLU None

73.98
32.14
63.51

Accuracy - 64.75

Prediction

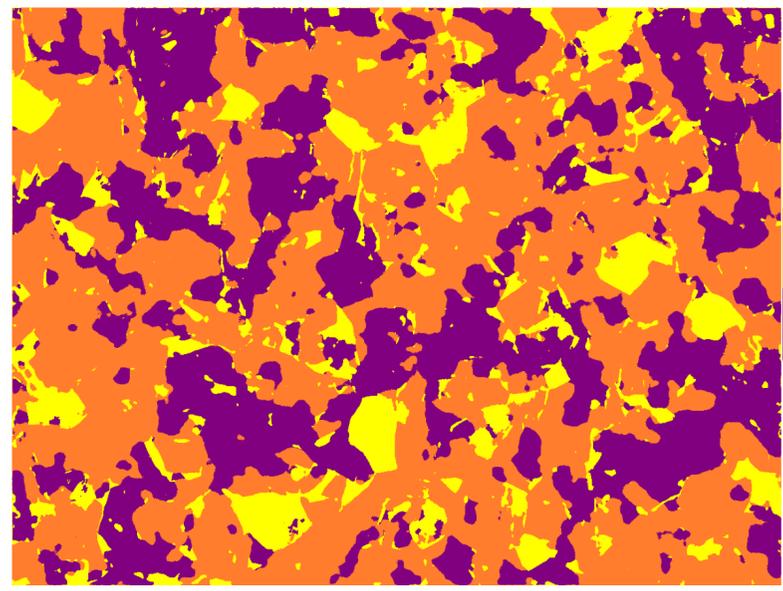
Down



68.16
56.58
49.1

Accuracy - 52.5

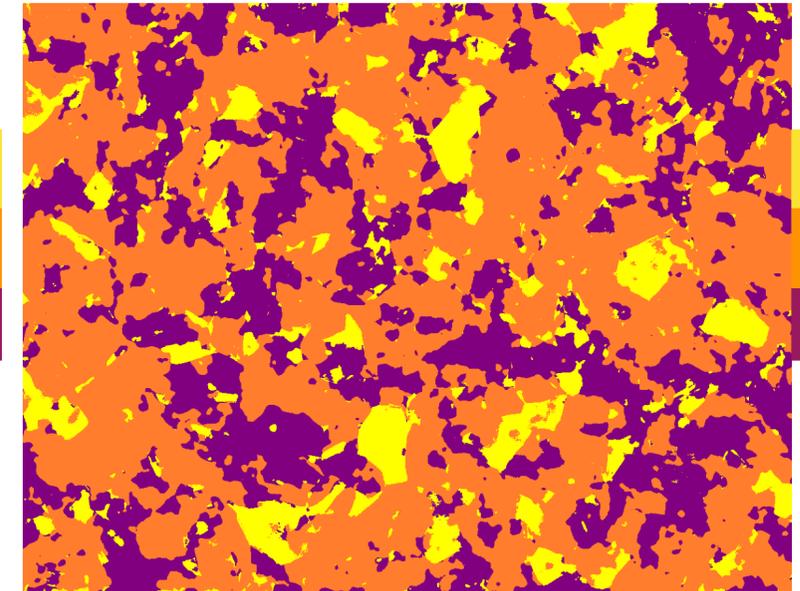
Up



64.73
67
40.3

Accuracy - 45.04 24

DownUp



62.47
75.92
38.02

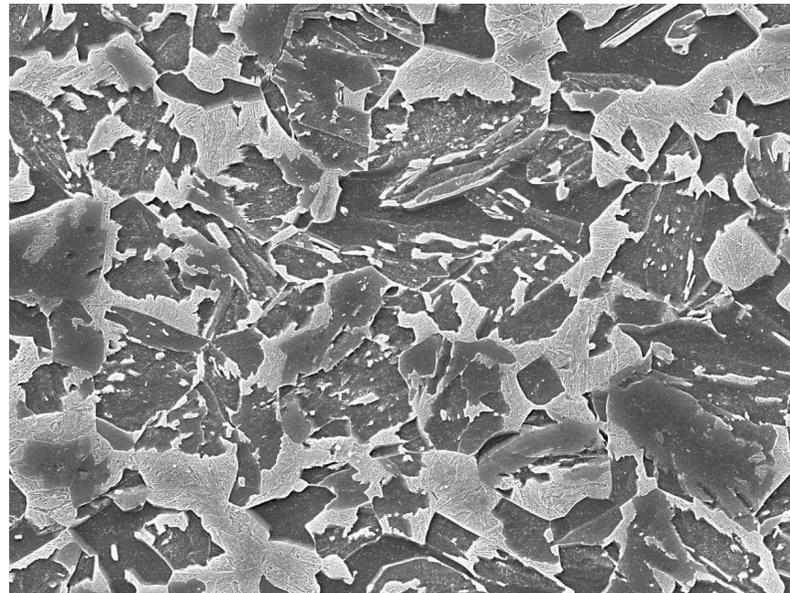
Accuracy - 40.24

Inference Images

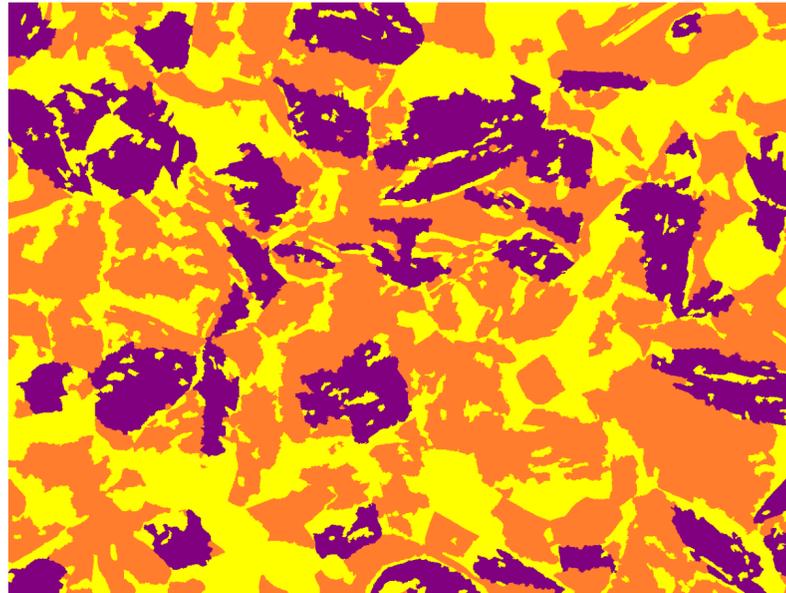
test3 H type

image 0.bmp

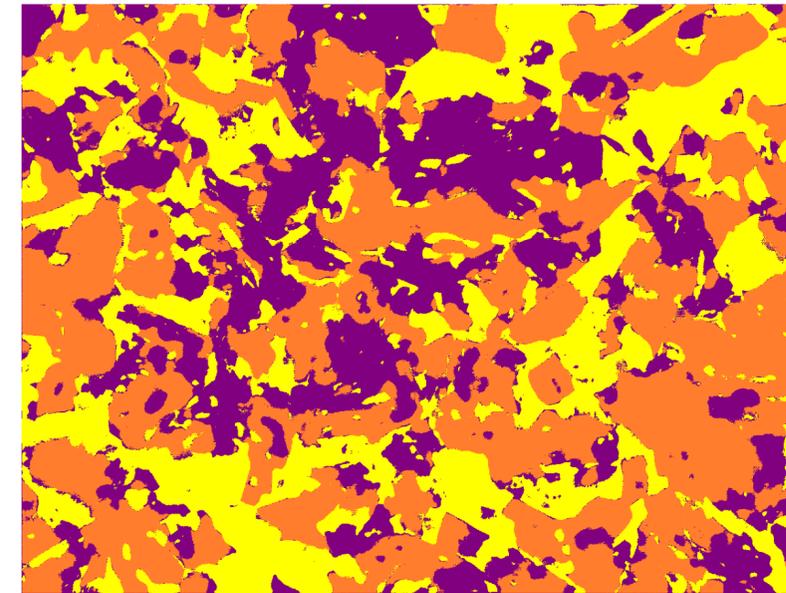
Original



Label



Prediction



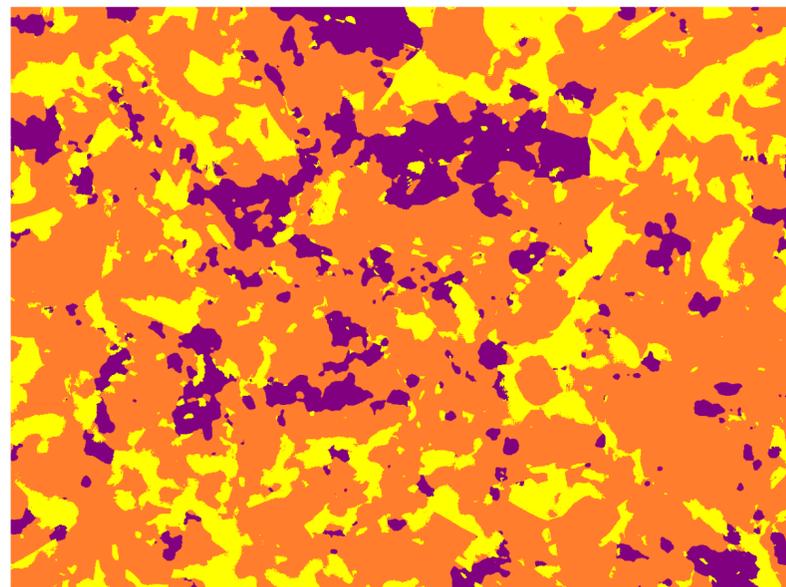
ReLU None

81.51
73.96
62.44

Accuracy - 73.89

Prediction

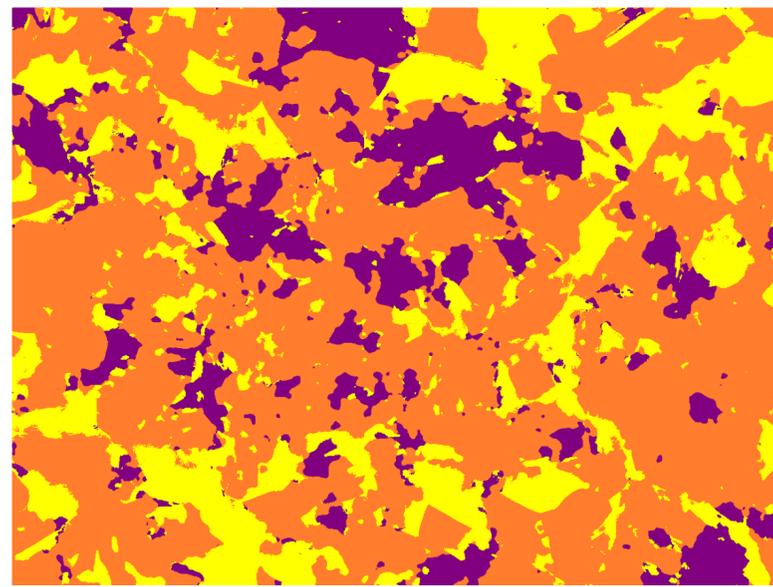
Down



61.95
90.87
33.89

Accuracy - 68.02

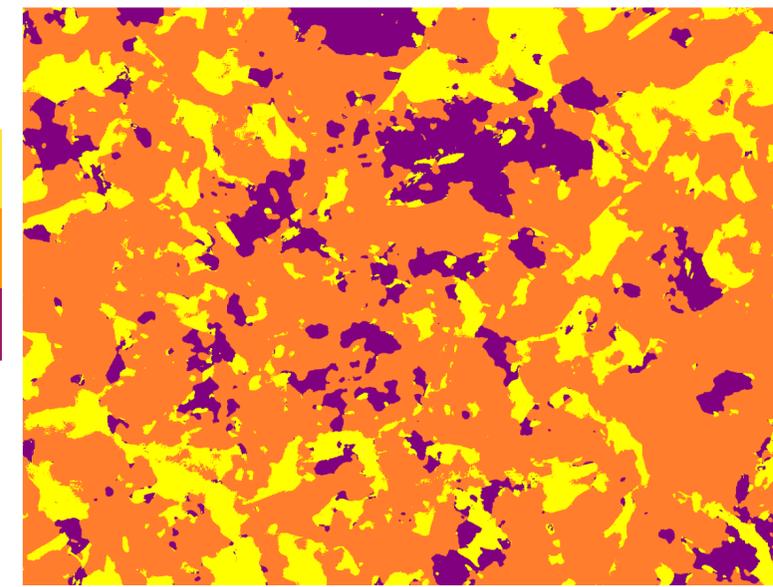
Up



65.77
89.5
35.02

Accuracy - 69.0 25

DownUp



60.87
90.34
31.09

Accuracy - 66.78

Remarks

	Avg (10)	Avg (8)	(1)	(1)	(1)
Blur Pooling	x3000	x5000	test1 A type	test2 D3 type	test3 H2 type
None	83.97 (0.8372) 60.91 91.41 55.1	90.01 (0.8976) 74.33 94.19 62.19	77.41 (0.7716) 49.97 81.79 82.27	64.75 (0.6445) 73.98 32.14 63.51	73.89 (0.7364) 81.51 73.96 62.44
Down	83.53 (0.8329) 60.94 95.16 42.59	87.69 (0.8743) 61.7 97.14 38.38	70.83 (0.7061) 39.52 93.37 58.82	52.5 (0.5227) 68.16 56.58 49.1	68.02 (6777) 61.95 90.87 33.89
Up	84.41 0.8416 65.54 94.72 47.2	87.93 (0.8768) 62.26 96.91 43.96	76.73 (0.7648) 47.76 89.86 73.47	45.04 (0.4479) 64.73 67 40.3	69.0 (0.6875) 65.77 89.5 35.05
Down + Up	84.3 (0.8405) 64.93 94.63 44.15	87.51 (0.8726) 61.05 97.46 33.51	74.5 (0.7425) 44.11 90.72 68.56	40.24 (0.3999) 62.47 75.92 24.87	66.78 (0.6653) 60.87 90.34 31.09

Accuracy (DICE)

- There was little-to-no effects of blur pooling in the experimentation. There was a little increase in performance in x3000 images but rest of inference images did not perform well.

3. Comparison between Losses

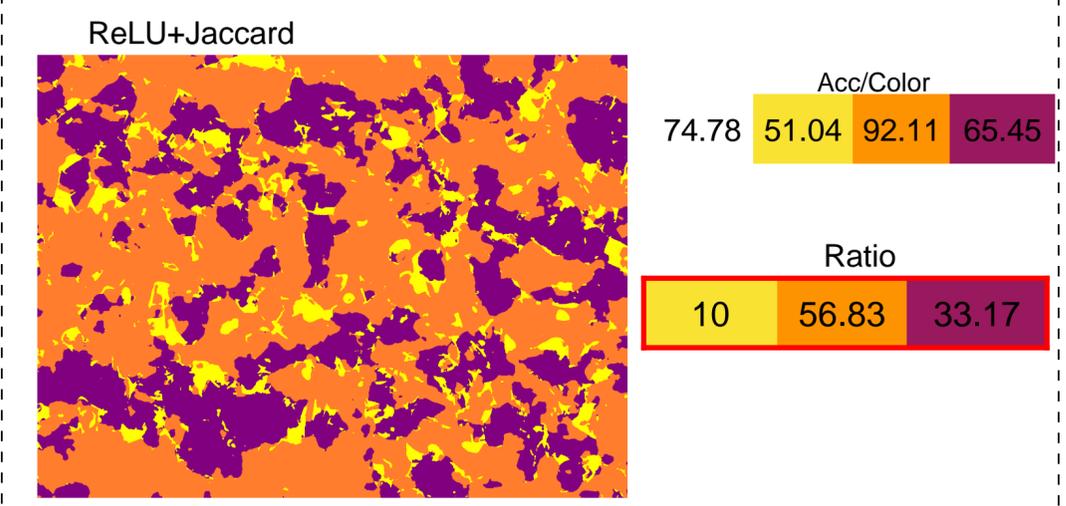
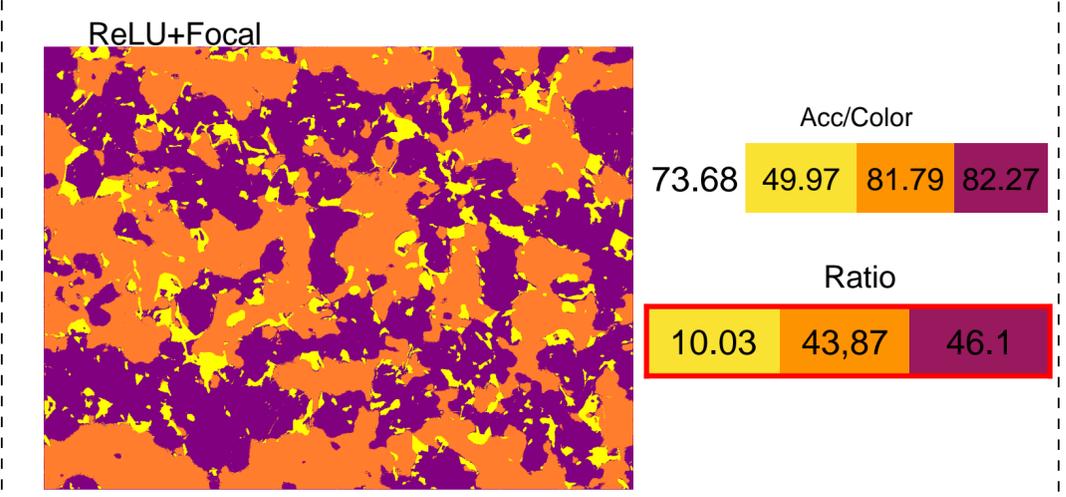
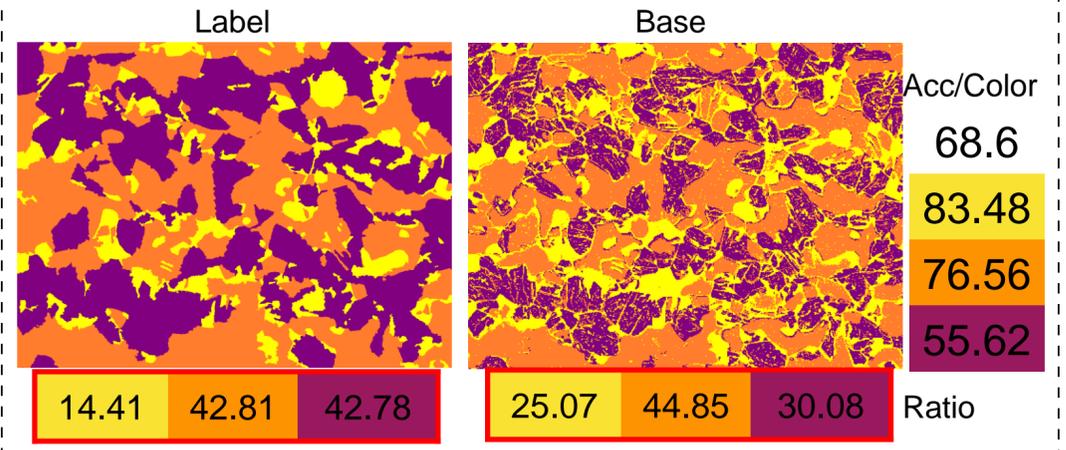
Experimented with different loss values to find the appropriate loss function for the model.

- Focal Loss - is used to accurately identify and localize objects within an image by focusing more on hard to detect objects in the image.
- Jaccard Loss - is used to produce segmentation masks that closely align with the ground truth masks, promoting precise and accurate delineation of object boundaries.

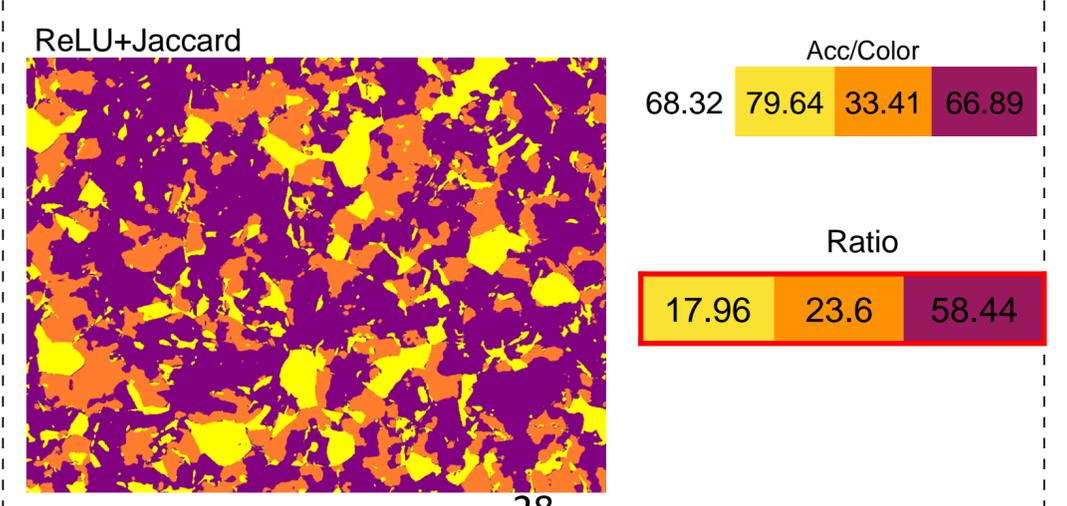
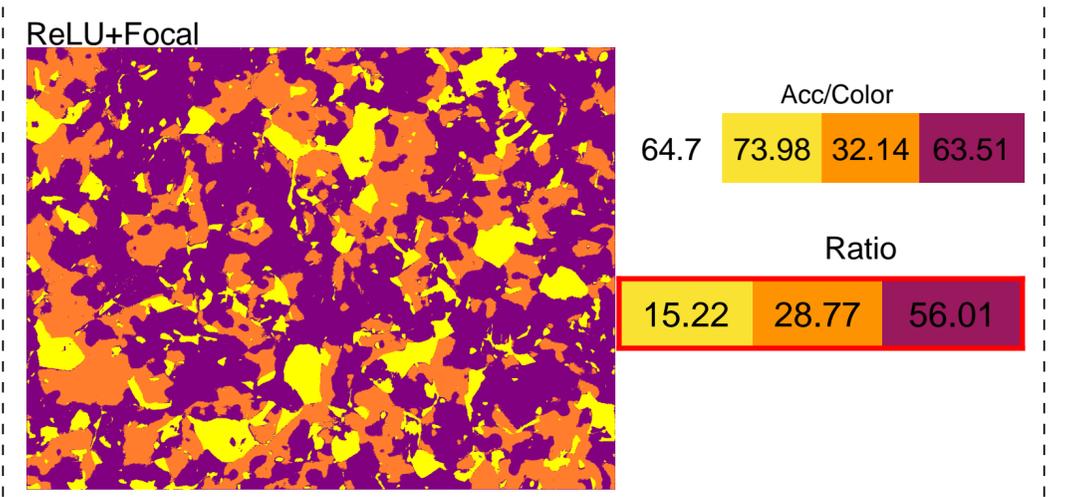
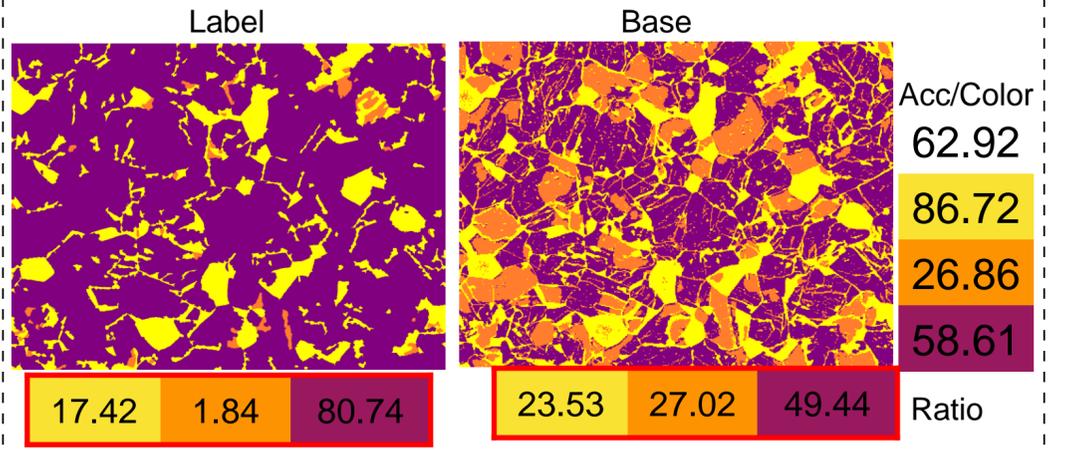
Inference Images and Comparison

3 class- Ratio

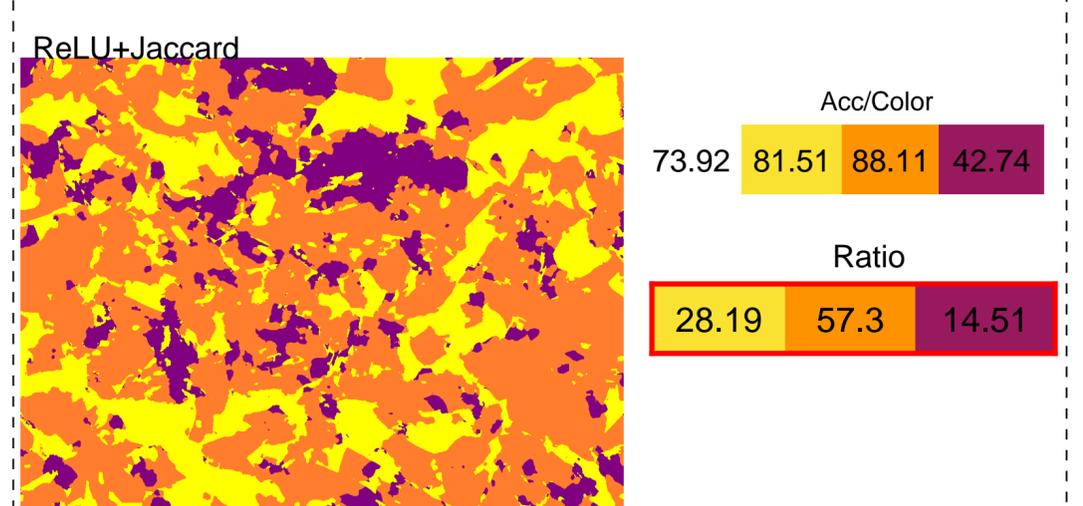
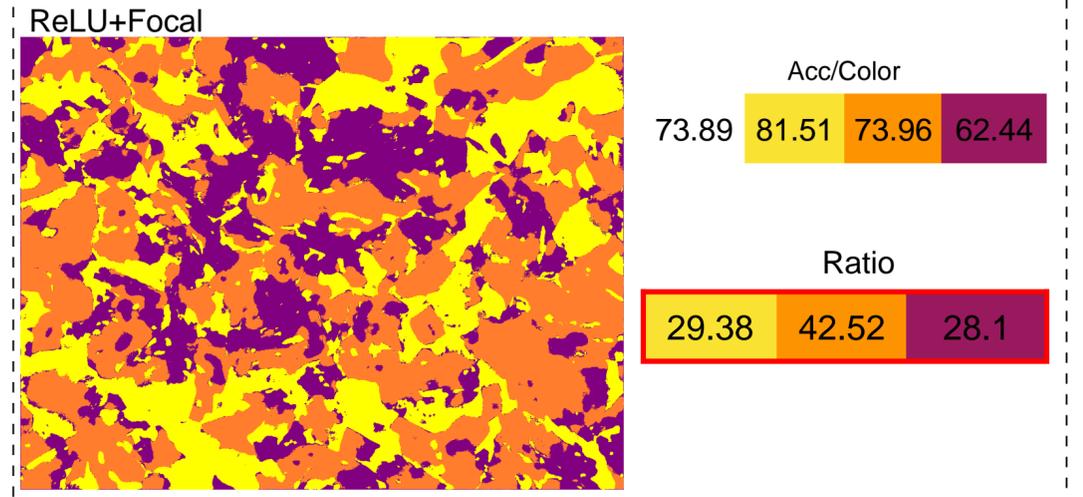
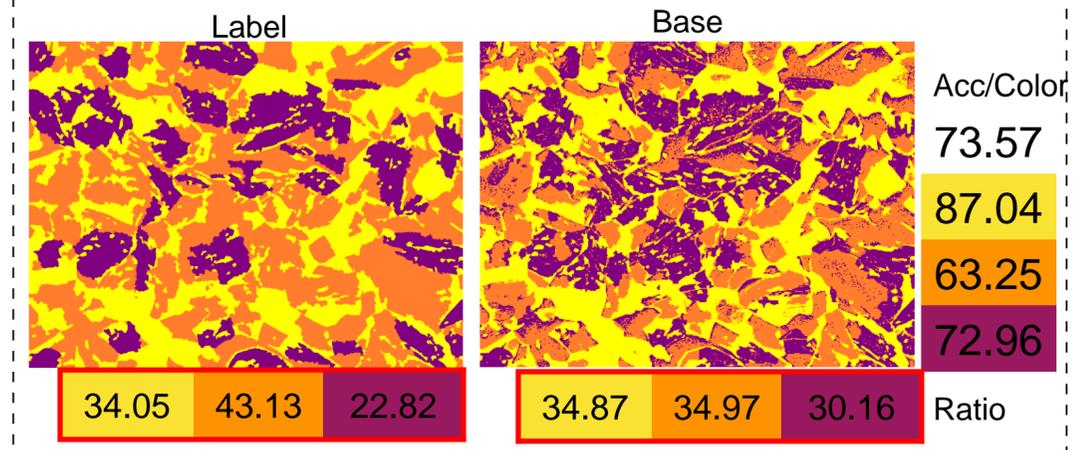
test1_Atype



test2_D3type



test3_H2type



Remarks

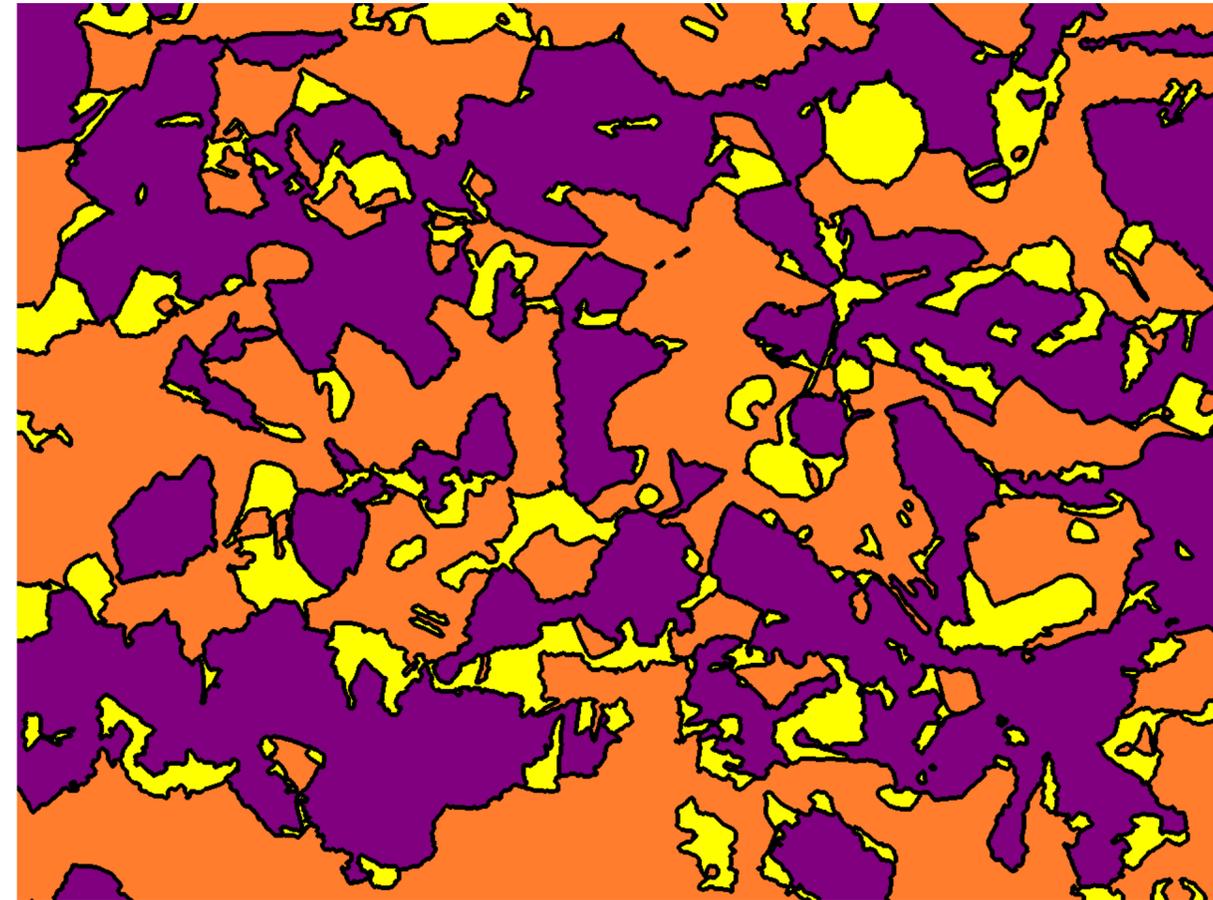
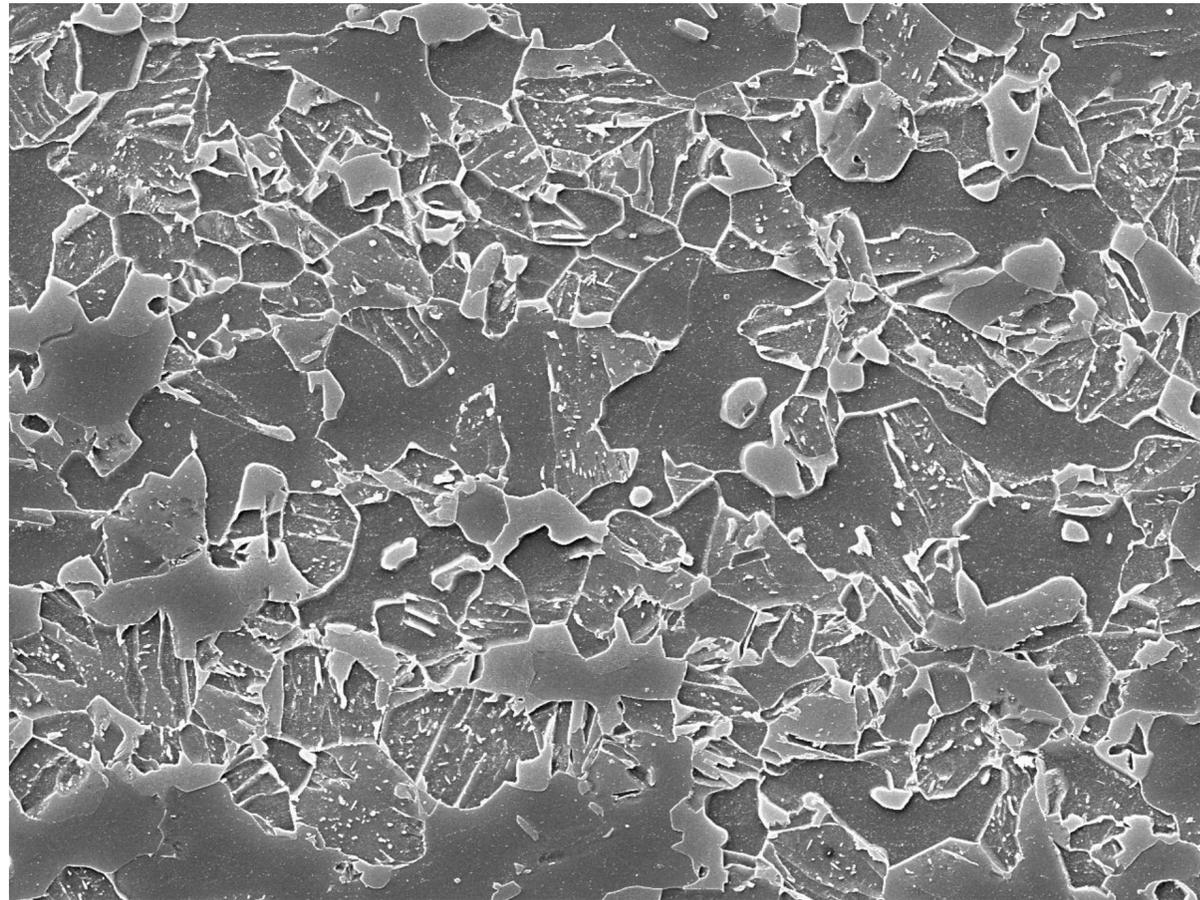
None

	Avg (10)	Avg (8)	(1)	(1)	(1)
	x3000	x5000	test1 A type	test2 D3 type	test3 H2 type
ReLU + Focal After 200 epochs	83.97 60.91 91.41 55.1	89.84 74.33 94.19 62.19	77.4 49.97 81.79 82.27	64.75 73.98 32.14 63.51	73.89 81.51 73.96 62.44
ReLU + Jaccard After 200 epochs	84.11 68.11 93.77 45.33	89.29 70.87 96.57 41.73	74.78 51.04 92.11 65.45	68.32 79.64 24.41 66.89	73.92 78.52 88.11 37.62

- Although Focal loss did better in x5000 images, Jaccard loss performed better than Focal Loss in other inference images.

4. Classification with Boundary Class

Previous related studies suggested that using boundary classes increased performance of image segmentation. Therefore we conducted experiment to test out the same.

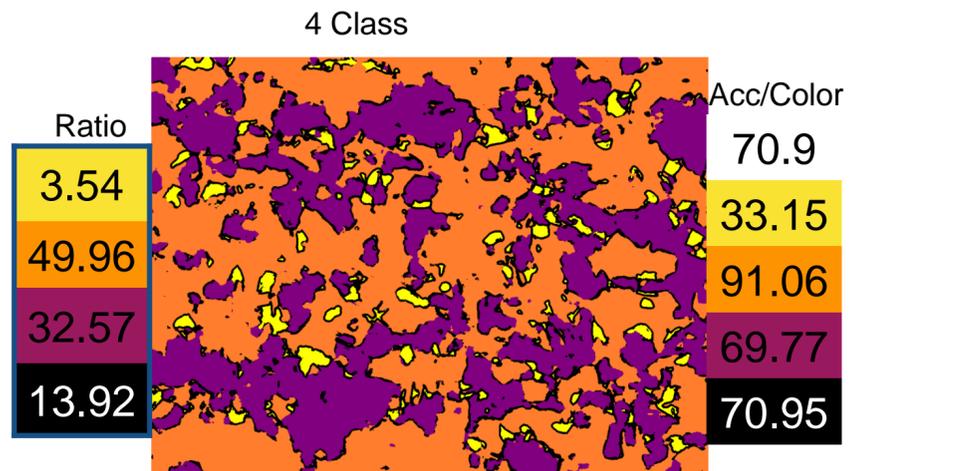
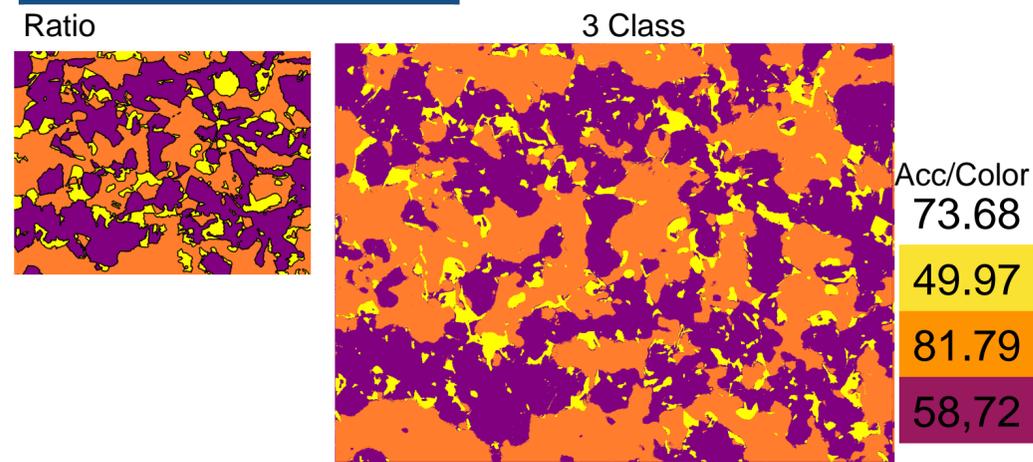
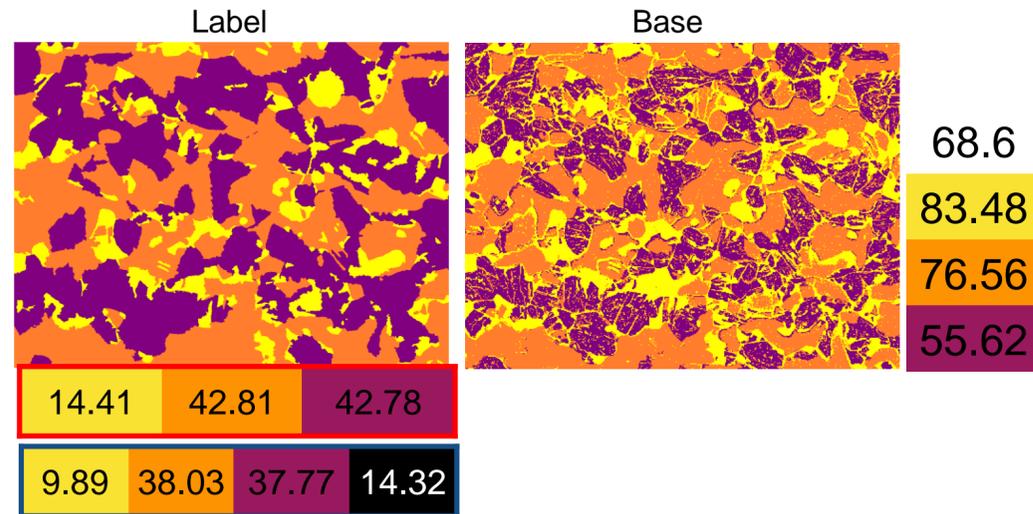


Inference Images and Comparison

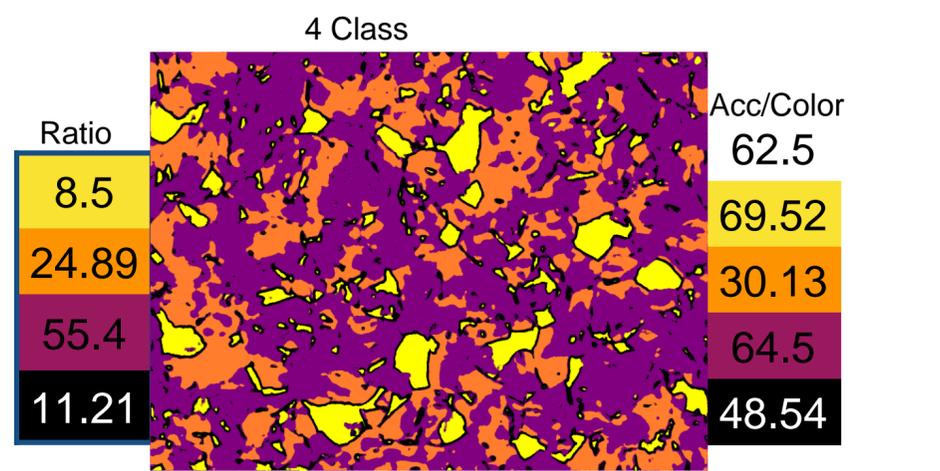
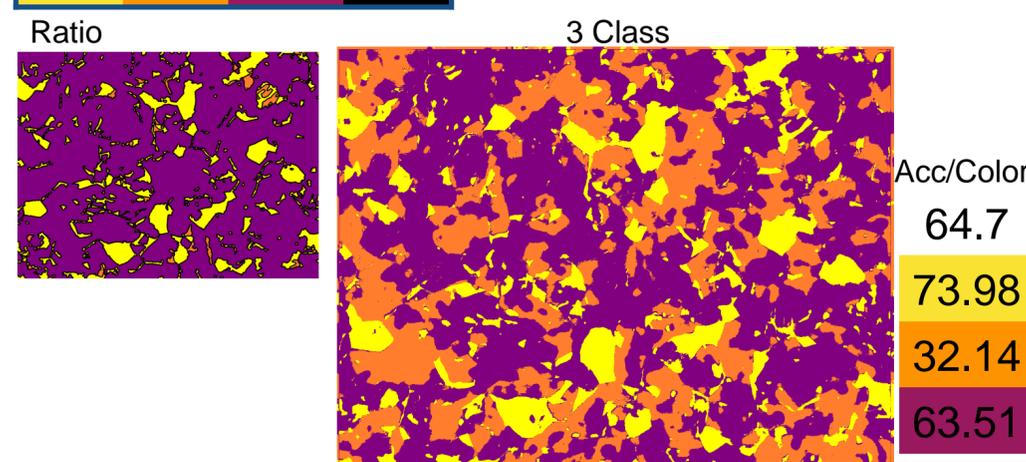
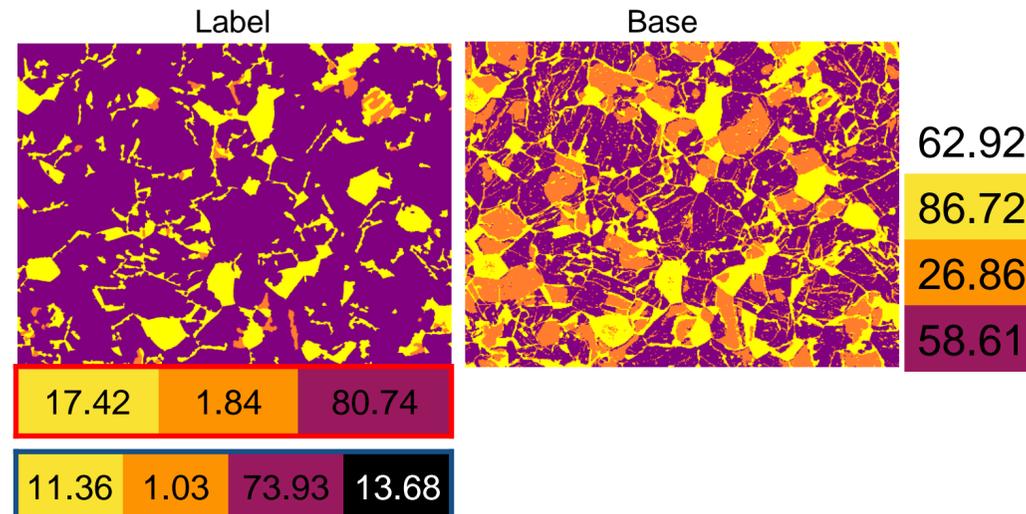
3 class- Ratio

4 class- Ratio

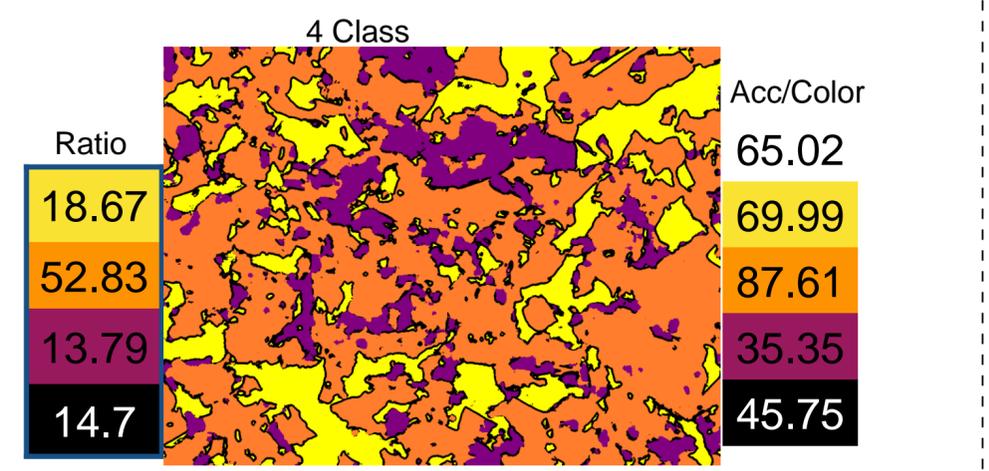
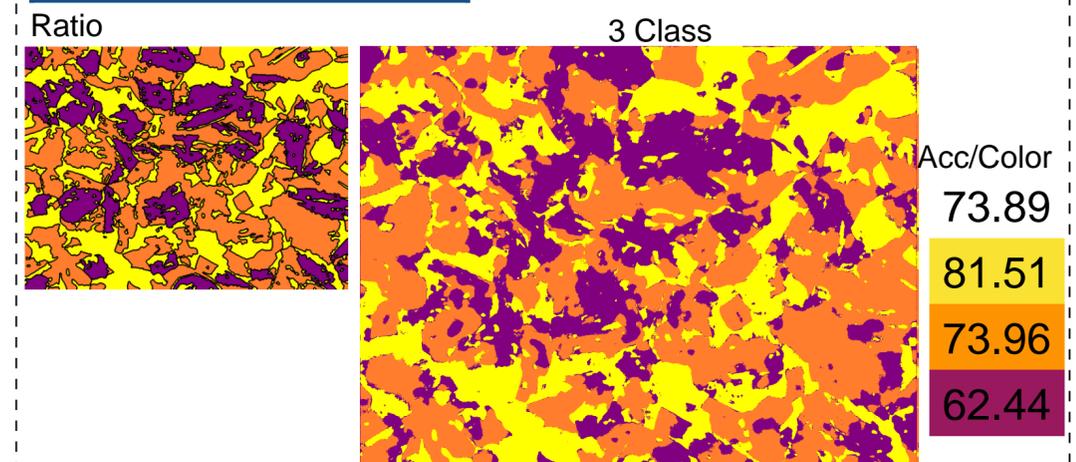
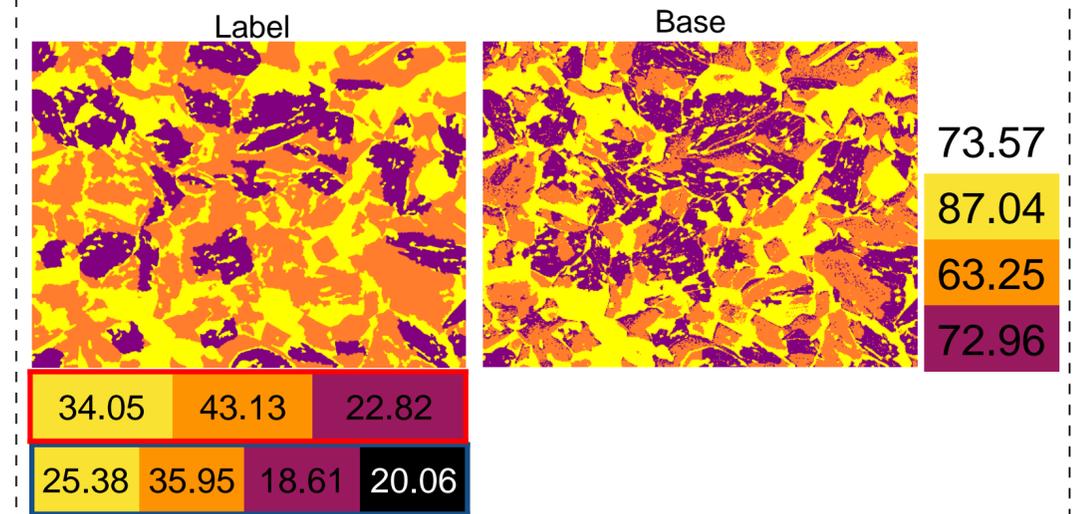
test1_Atype



test2_D3type



test3_H2type



Remarks

None

	Avg (10)	Avg (8)	(1)	(1)	(1)
	x3000	x5000	test1 A type	test2 D3 type	test3 H2 type
3 Class - ReLU After 200 epochs	83.97 (0.8397) 60.91 91.41 55.1	89.84 (0.8976) 74.33 94.19 62.19	77.4 (0.7716) 49.97 81.79 82.27	64.75 (0.645) 73.98 32.14 63.51	73.89 (0.7364) 81.51 73.96 62.44
4 Class - ReLU After 200 epochs	80.04 (0.7979) 57.72 95.32 40.51 57.48	84.94 (0.8469) 57 97.46 33.94 53.37	70.95 (0.707) 33.15 91.06 69.77 46.74	62.53 (0.6228) 69.52 30.13 64.5 48.54	65.02 (0.6477) 69.99 87.61 35.35 45.75

Accuracy Percentage (DICE)

- Inclusion of boundary class did not result in any performance improvements. The output results are lower than 3 Class classification in all inference images.

4 Class Focal vs Jaccard Loss

ReLU
None

	Avg (10)	Avg (8)	(1)	(1)	(1)
	x3000	x5000	test1 A type	test2 D3 type	test3 H2 type
Focal	80.04 (0.7979)	84.94 (0.8469)	70.95 (0.707)	62.53 (0.6228)	65.02 (0.6477)
	57.72 95.32 40.51 57.48	57 97.46 33.94 53.37	33.15 91.06 69.77 46.74	69.52 30.13 64.5 48.54	69.99 87.61 35.35 45.75
Jaccard	80.75 (0.805)	83.94 (0.8369)	72.28 (0.7198)	64.04 (0.6379)	65.86 (0.6561)
	60.37 95.58 38.04 58.62	60.07 98.08 32.44 44.41	40.01 89.37 66.70 46.84	56.81 52.42 40.4 46.84	40.1 89.37 66.69 67.82

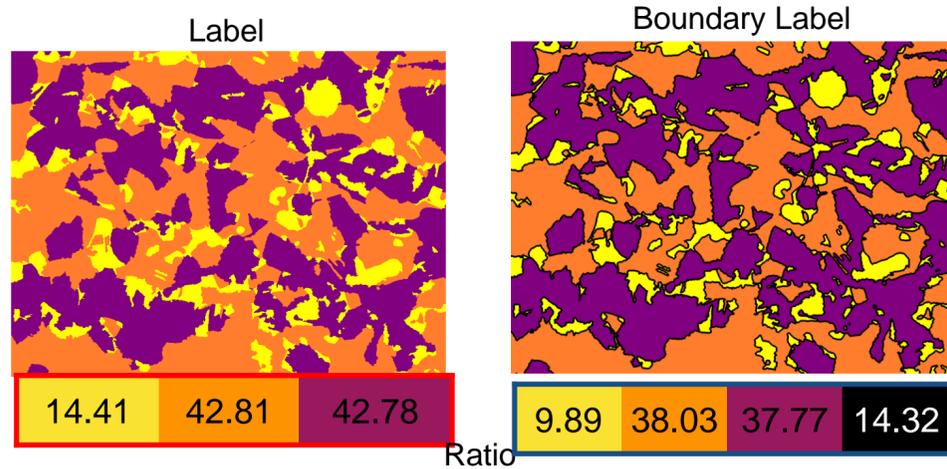
Accuracy Percentage (DICE)

Inference Images and Comparison

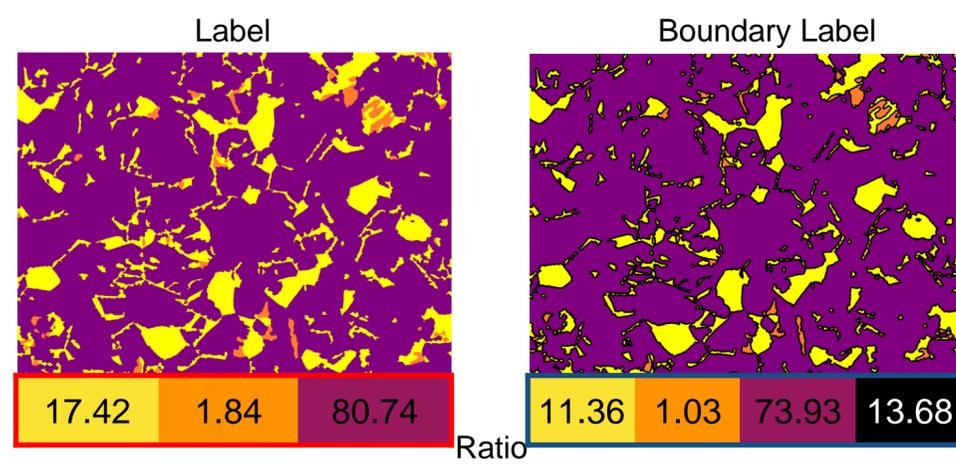
3 class- Ratio

4 class- Ratio

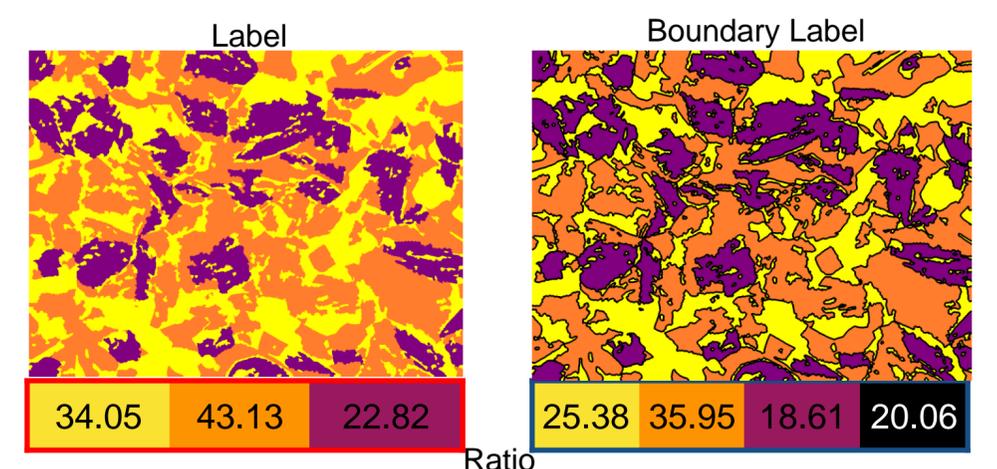
test1_Atype



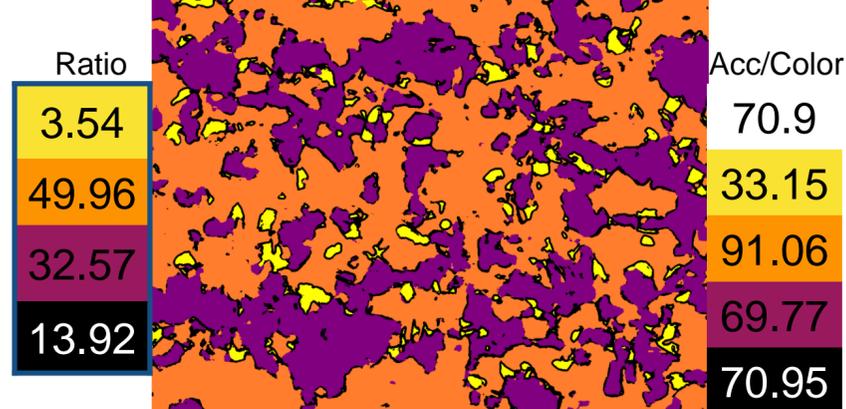
test2_D3type



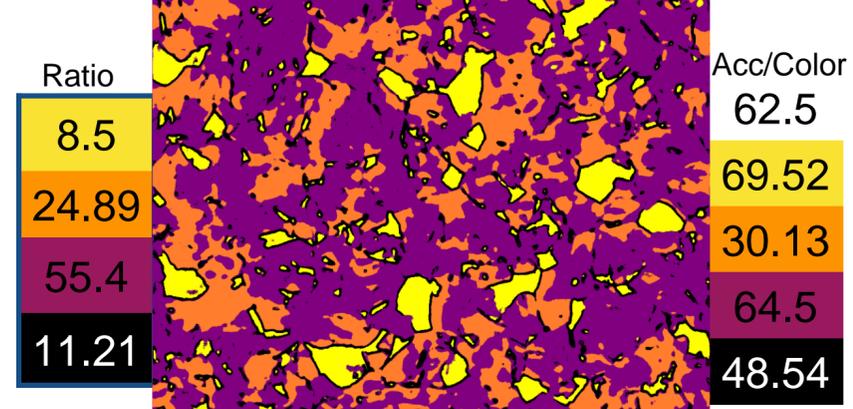
test3_H2type



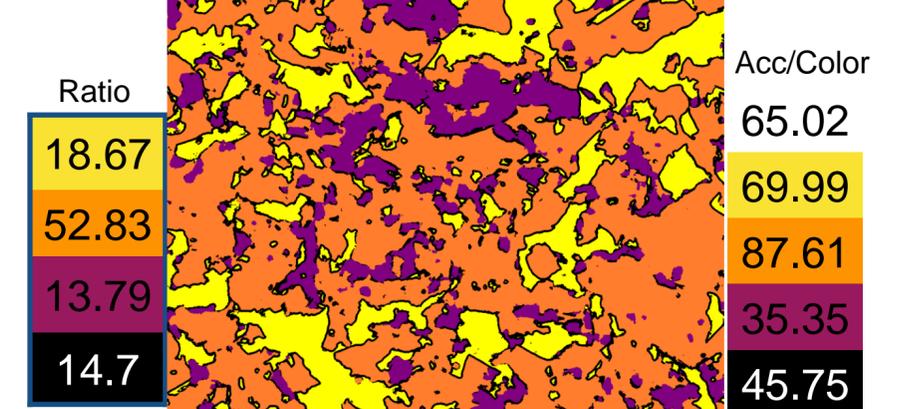
ReLU + Focal [4 Class]



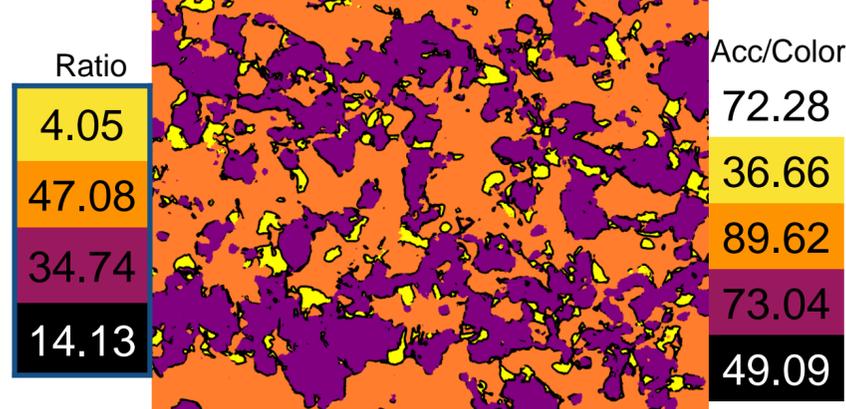
ReLU + Focal [4 Class]



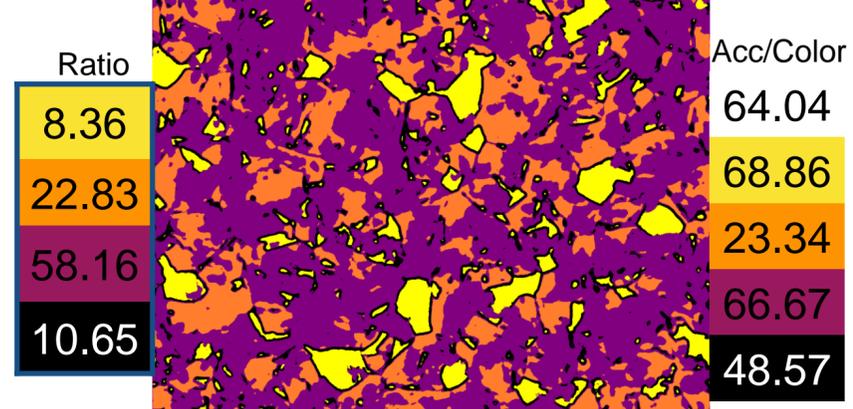
ReLU + Focal [4 Class]



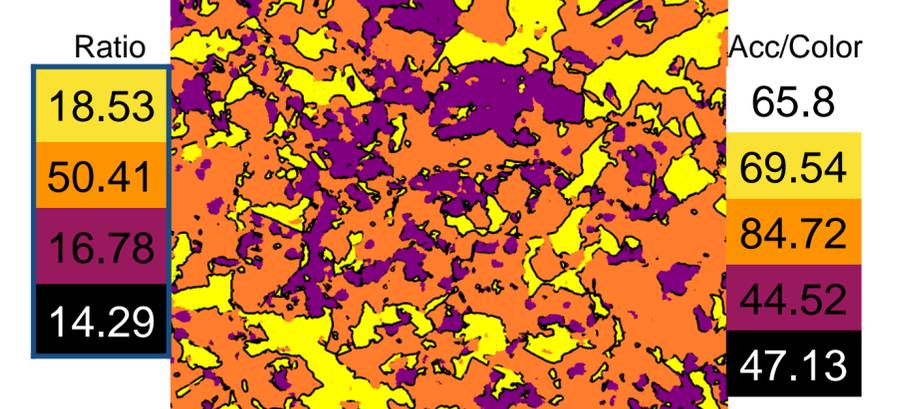
ReLU + Jaccard [4 Class]



ReLU + Jaccard [4 Class]



ReLU + Jaccard [4 Class]



Conclusions

Removing Noise and Blur and controlling the augmentation parameters during augmentation increased model performance.

Application of blur effects doesn't impact the performance of the model.

Using Jaccard loss instead on Focal Loss increased performance.

The performance of Model with boundary class was a little under par compared to the 3 Class Model.

Using Unet model with ReLU activation and Jaccard loss without boundary class seems to provide better result.

	x3000	x5000	test1 A type	test2 D3 type	test3 H2 type
Baseline	-	-	68.6 (0.6842)	62.92 (0.6267)	73.57 (0.7332)
3Class- ReLu+Jaccard	84.11(0.8386)	89.29 (0.8904)	74.78 (0.7453)	68.32 (0.6807)	73.92 (0.7367)