Semantic Segmentation of Endoscopic Images

Akshay Viswakumar
Student Number # 32971665
Electrical & Computer Engineering
The University of British Columbia
Vancouver, British Columbia V6T 1Z4

***Abstract*—Deep learning has proven itself as a powerful mechanism to train models that can perform semantic segmentation. UNets are a type of Fully Convolutional Network that, by way of its unique architecture, are capable of performing segmentation after being trained on very few samples. In this project, a pipeline is built for performing semantic segmentation of endoscopic images. Three UNet based neural networks are designed, constructed and trained to perform semantic segmentation of endoscopic images. The performance of these networks are then compared against the baseline UNet. Finally, I discuss the efficacy of UNets for the semantic segmentation task.**

***Index Terms*—Semantic Segmentation, Deep Learning, Fully Convolutional Network, UNet, Endoscopic Images**

I. Introduction

Endoscopy is a widely used procedure for the early detection of diseases in hollow organs such as the Oesophagus, Stomach, Colon and Bladder. Computer assisted techniques for accurate and consistent segmentation of lesions and other diseased regions can be useful for monitoring conditions as well as timely surgical planning. The objective is to design a neural network based solution that can semantically segment diseased regions in endoscopic images.

A. Background

Convolutional Neural Networks (CNNs) take inspiration from the mammalian visual system. In particular, the way how the brain processes visual information hierarchically often starting with simple structures such as oriented edges [1]. In the late 70s, Fukushima applied this idea to create the Neocognitron [2], a predecessor to CNNs, which was built to detect handwritten characters. The Neocognitron introduced the notion of a convolutional layer which was a spatially invariant filter that could be used to detect features irrespective of location in an image. The coefficients of the convolutional filter still had to be trained via some supervised learning algorithm. Fast forward to 1998, when LeCun *et al* [3] applied error backpropagation to train convolutional filter coefficients paving the way to CNNs as we now know them.

CNNs have largely been used for image classification. Networks such as ResNet-34 [4] and VGG-16 [5] are some popular CNN based deep neural networks that have achieved significant classification accuracy after having been trained on the vast ImageNet database. CNNs have also found prominence in image segmentation. This is not wholly surprising since segmentation can essentially be thought of as a dense classification problem where every pixel in an image is labelled. Pipeline based solutions (like R-CNN [6]) and Fully Convolutional Networks (FCNs) [7] are two popular approaches that extend CNNs to the segmentation task.

FCNs, unlike conventional CNNs do not contain fully connected layers of neurons. Additional convolutional layers are placed instead of the fully connected layers. As a result, the outputs of an FCN are spatial and allow for dense (pixel-wise) labelling. The use of ”skip” connections, to combine pooled outputs from initial layers with outputs from the penultimate layer, enable the combination of coarse, semantic and local appearance information resulting in a segmented output that is more spatially precise. UNet [8] is an FCN based architecture that takes these notions a step further. UNet and UNet-like architectures will be the focus of this project.

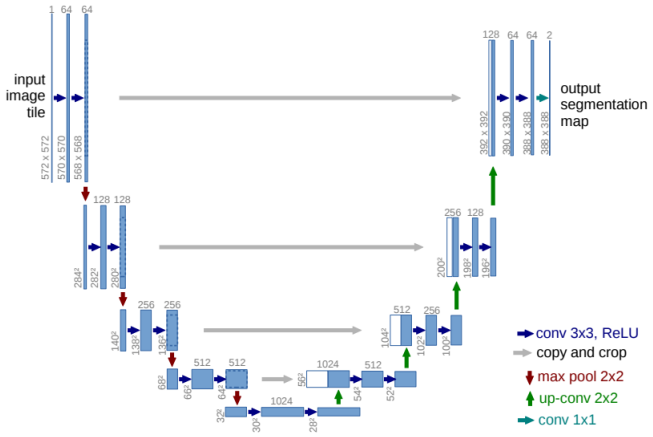


Fig. 1: Original UNet Architecture from [8]

Ronneberger *et al* developed the UNet architecture [8] to address two major issues prevalent in contemporary deep convolutional networks for segmentation: (1) Pipeline based approaches like R-CNN were quite slow and often ended up performing a lot of redundant processing due to overlaps. (2) There was always a trade-off between localization accuracy and overall context.

UNets are called so because of the U-shape that they form. In addition to the normal contracting (downsampling) network, a mirror-image expansive (upsampling) network is attached in succession. In the expansive network, transposed-convolution acts as an inverse of sorts to the convolution + pooling layers in the contracting network. The most apparent result of this is an output with higher resolution. For increased localization, high resolution feature maps from the contracting stages are combined with their equivalent up-sampled counterparts in the the expansive stages. The combination here refers to concatenation along the channel dimension. Refer to Figure 1 for an illustration of the original UNet architecture.

Another notable feature of UNet was that it could be used to accurately segment images after being trained on a very small number of samples (as little as 30 images in [8]). This aspect makes UNets a popular tool in biomedical segmentation problems where there is a dearth of training data.

B. Summary of Contribution

1) Built a complete pipeline for performing semantic segmentation on endoscopic images.
2) Constructed, trained and compared performance of UNet and 3 UNet-like architectures.
3) Discussion about the efficacy of UNet-like architectures for semantic segmentation tasks.

II. Materials

A. Dataset

The National Institute of Health Research (NIHR) has made public, a comprehensive dataset [9] of 386 RGB image frames collected from endoscopic video recordings. According to the NIHR, the dataset was compiled from endoscopic videos collected from at least five different medical research institutions around the world and are from a diverse group of patients. Experts have analyzed the image frames to identify and label diseased regions. These diseased regions may belong to one of five classes.

1) BE (Barret's Oesophagus)
2) HGD (High Grade Dysplasia)
3) Cancer
4) Polyp
5) Suspicious

The dataset also contains accurate masks for each diseased region that have been identified by experts. The masks were grayscale images with with intensity 255 used to represent diseased region and intensity 0 to represent everything else. Note that some frames contain more than one diseased region in which case there were multiple masks for that frame with each mask corresponding to one of the disease classes.

Two kinds of pre-processing steps were carried out before using the dataset. Note that these steps are separate from data augmentation which was used during training.

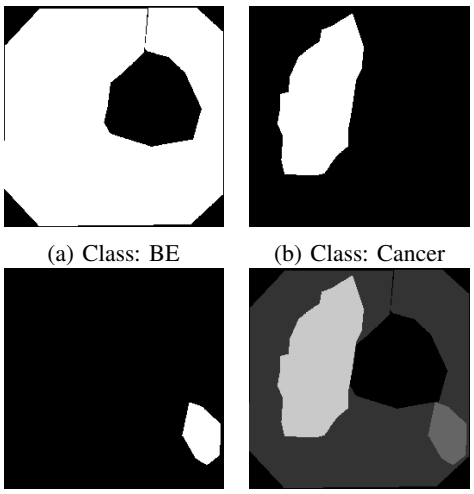
1) **Mask Concatenation:** For images containing more than one diseased regions, all separate masks were combined into one composite mask. The resulting mask was a grayscale image where each diseased region was assigned a specific intensity depending on the class of disease. The intensity mapping has been illustrated in TABLE II. Note that this step was to ensure that there would only be a single mask for any given image. Figure 2 illustrates an example of how a composite mask was created.
2) **Image Resize:** All images and corresponding masks were resized to a common size. Image acquisition did not seem to have been standardized and all images (and corresponding masks) were of different sizes.

TABLE I: Description of NIHR Dataset

	Description
Modality	Image Frames from Endoscopic Video
Color Space	RGB [for Images] ; Grayscale [for Masks]
Resolution	Varied
Image Count	386 Images ; 577 Masks

TABLE II: Intensity Mapping for Composite Mask

Class	Assigned Intensity
BE	51
HGD	151
Cancer	201
Polyp	251
Suspicious	101
Background	0



(a) Class: BE (b) Class: Cancer

(c) Class: Suspicious (d) Composite

Fig. 2: Mask Concatenation Illustration

B. Experimental Setup

All experiments were carried out in the MATLAB environment. While other frameworks such as Keras was considered initially, I resorted to using MATLAB since the environment was the least complicated to set up and could guarantee that all developed functions and scripts could be easily ported to

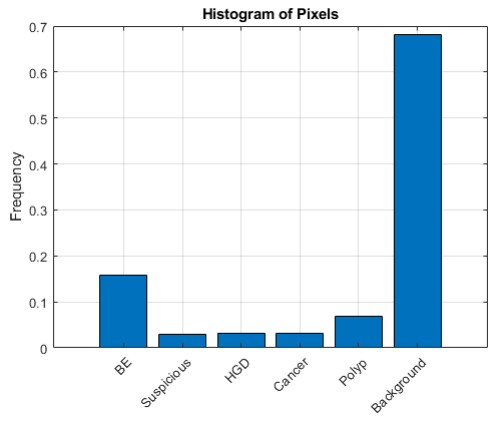


Fig. 3: Distribution of Class Labels in Training Dataset

different systems without the need for several independent third party libraries and modules.

Moreover, the MATLAB Deep Learning Toolbox has significantly improved over the last few versions and is currently capable of building complex deep learning networks from scratch with as much finesse and sophistication as one would expect from frameworks like Keras. All experiments were carried out on Matlab Version R2019b. All code developed as part of this project should work as is and without modification on all higher versions of MATLAB.

III. Methods

This section details some generalized considerations applicable to all network designs that were experimented with.

A. Loss Function

Creators of the original UNet [8] architecture used Cross Entropy (CE) in their design. They do not elaborate on their rationale for choosing this function, but it may not be all that surprising. The CE function is a staple in machine learning algorithms because of its appealing differentiable properties that help simplify gradient descent. CE loss can also be augmented with the use of weights to tackle class imbalance. For an N class problem with predicted probability $\hat{y}$ and true probability y the cross-entropy loss is given by expressions below.

$$C.E\ Loss = -\sum_{c=1}^{N} W_i y_i \log \hat{y}_i$$

where W_i is the weight associated with the i^{th} class

For all experiments that were part of this project, CE was used as the loss function of choice.

B. Correcting Class Imbalance

An examination of the training data revealed that there was a significant class imbalance. Figure 3 shows a histogram of all pixels in the training set binned on the basis of class membership. Pixels in the Background class dominate with a frequency of around 70%. This over-representation may hinder the network's ability to learn well. To counter this, each class was assigned an appropriate weight. A lower weight is meant to penalize an over-represented class. Class weights were calculated as the product of the median frequency out of all classes times the inverse frequency of a given class.

$$W_{(Class\ A)} = \frac{MedianFrquency_{(AllClasses)}}{Frequency_{(Class\ A)}}$$

This weight is then used as multiplier for the CE loss function in the pixel classification layer (final layer) of the neural network. The class weights computed for each class is shown in TABLE III.

TABLE III: Weights to Correct Class Imbalance

	Classes					
	BE	Suspicious	HGD	Cancer	Polyp	Background
Weight	0.6703	2.0406	1.3325	1.0597	0.9467	0.2789

As can be seen, the over-represented class (Background) is assigned the lowest weight and is penalized while the under-represented class (Suspicious) has the highest weight.

C. Optimizer

Some major challenges one faces when applying gradient descent to train a neural network are slow convergence, oscillations and local minima. A solution to these challenges involve modifications to how the learning rate is varied over the course of training. One of the earliest methods vary the learning rate involved the use of "momentum" where a fraction of the update from the previous iteration was added to the current update. Analogous to its physical counterpart, higher momentum could thus help escape from a point of local minima. Adaptive Moment Estimation (ADAM) [10] is a more modern approach to computing an adaptive learning rate. This is the optimizer of choice in many of the successful deep neural nets. ADAM is the optimizer of choice for this project.

D. Normalization

Normalization was performed on input images. The form of normalization was zero-mean normalization where the average intensity of an input image would be subtracted from all pixels of that image.

E. Regularization

Regularization techniques are used to combat situations where the neural network ends up memorizing the training data. Another term for this phenomenon is over-fitting. Regularization techniques modify the machine learning algorithm to prevent over-fitting. Early stopping is a form of regularization. That is, the value of the loss function over the validation set is periodically monitored and training is stopped if the validation loss increases over consecutive epochs. This may be a little too drastic when training deep networks where losses may oscillate over a few iterations before lowering further. A relaxed form of early stopping was applied to the networks that were trained as part of this project. That is, training

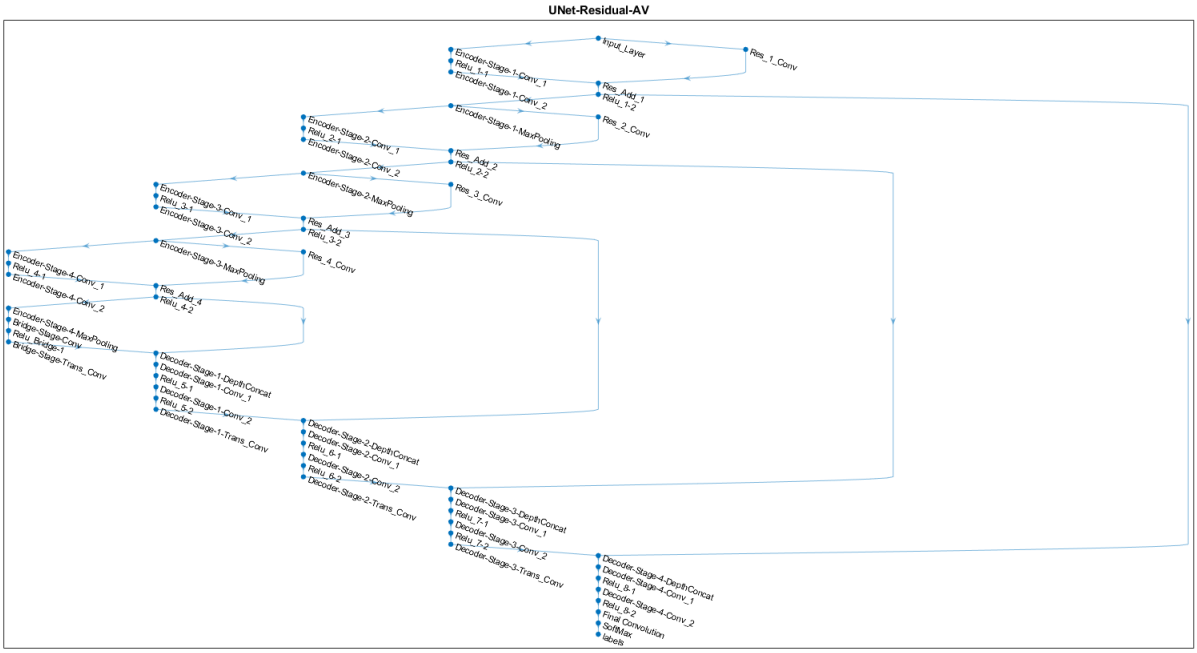


Fig. 4: Graph Diagram of UNet-Residual-AV

would stop only if the validation loss stopped decreasing for seven consecutive epochs. The only form of regularization used in this project is early stopping. This seemed to have been sufficient as over-fitting was not otherwise observed.

F. Data Augmentation

Data augmentation is a great way to derive more value from a small dataset. In the case of image data, change of scale, shifts, shears and rotations can help generate "new" data that can be used for training. For this project, the training set of images were augmented using horizontal/vertical shifts and rotations. Other means of augmentations such as shearing did not help improve performance and was therefore omitted.

G. Training

All networks were trained using mini-batch gradient descent. Each mini-batch consisted of 10 augmented inputs. Inputs were shuffled every Epoch. A higher mini-batch size could not be selected owing to limitations in GPU memory. The validation loss was computed once every 30 iterations (roughly once every Epoch). Early stopping was configured with a patience of 7. Checkpoints were configured to back up weights periodically through the course of training.

IV. Architectures

For this project, I've designed, constructed and trained three UNet-like architectures for the semantic segmentation problem. The design specifics of these three architectures are covered in this section. Note that Encoder and Decoder may be used in place of the terms Contractive and Expansive paths respectively. Both sets of terms refer to the same thing.

A. UNet-AV

UNet-AV is a complete reconstruction of UNet with two major differences. (1) UNet-AV does not have any dropout layers. (2) UNet-AV uses Xavier initialization [11] for initializing weights of the convolutional layers instead of He initialization [12]. UNet-AV takes in input images of dimension 256x256x3. It has an encoder depth of four meaning that there are four stages of contraction followed by four stage of expansion. The network resembles the illustration in Figure 1. All activation functions in intermediate layers are Rectified Linear Units (ReLU) and the activation function in the final layer is the SoftMax Function. Refer to the *DemoArchitecture_UNet_AV.m* script in the accompanying Demo package for a demonstration of how UNet-AV is defined and assembled.

B. UNet-Residual-AV

UNet-Residual-AV is a modification of UNet-AV. The modification involves the introduction of residual blocks to each encoder stage of the network. Like in ResNET [4], the residual block involves the use of an identity shortcut connection. That is, an input to an encoding stage is shorted and combined (via addition) with the output of the last convolution in that stage. The combined output is then passed through the final activation function for that stage. Note that a 1x1 convolution (with as many filters as are introduced in that layer) is performed in the residual path to ensure that the dimensions (along depth) are equal at the time of addition. The presence of the convolution does give this design some similarity to UNet++ [13] with the exception that we do not pass the intermediate output to the final layer. The decoder stage and all other parameters remains identical to UNet-AV. Refer to Figure 4 for an illustration of the UNet-Residual-AV network. Refer

True Class \ Predicted Class	BE	Suspicious	HGD	Cancer	Polyp	Background
BE	0.18	0.12	0.15	0.08	0.13	0.34
Suspicious	0.13	0.14	0.14	0.10	0.18	0.32
HGD	0.09	0.12	0.25	0.10	0.08	0.35
Cancer	0.07	0.15	0.08	0.09	0.31	0.30
Polyp	0.06	0.16	0.06	0.06	0.39	0.26
Background	0.08	0.10	0.11	0.06	0.19	0.46

(a) UNet

True Class \ Predicted Class	BE	Suspicious	HGD	Cancer	Polyp	Background
BE	0.39	0.02	0.08	0.01	0.09	0.40
Suspicious	0.21	0.04	0.09	0.02	0.23	0.41
HGD	0.12	0.04	0.34	0.02	0.05	0.44
Cancer	0.14	0.09	0.07	0.03	0.33	0.33
Polyp	0.05	0.03	0.02	0.01	0.51	0.37
Background	0.08	0.01	0.06	0.01	0.08	0.77

(b) UNet-AV

True Class \ Predicted Class	BE	Suspicious	HGD	Cancer	Polyp	Background
BE	0.38	0.03	0.07	0.04	0.07	0.41
Suspicious	0.21	0.06	0.09	0.07	0.20	0.37
HGD	0.13	0.05	0.27	0.06	0.03	0.46
Cancer	0.13	0.09	0.08	0.09	0.29	0.33
Polyp	0.09	0.03	0.02	0.02	0.43	0.42
Background	0.07	0.01	0.05	0.02	0.06	0.78

(c) UNet-Residual-AV

True Class \ Predicted Class	BE	Suspicious	HGD	Cancer	Polyp	Background
BE	0.41	0.12	0.12	0.06	0.03	0.27
Suspicious	0.25	0.20	0.13	0.08	0.10	0.25
HGD	0.08	0.10	0.45	0.11	0.04	0.21
Cancer	0.05	0.16	0.06	0.20	0.30	0.23
Polyp	0.03	0.07	0.01	0.03	0.57	0.29
Background	0.07	0.02	0.04	0.03	0.04	0.81

(d) UNet-VGG16-AV

Fig. 5: Confusion Matrices for Each Network based on Accuracy of Pixel Label Prediction

to the *DemoArchitecture_UNet_Residual_AV.m* script in the accompanying Demo package for a demonstration of how UNet-Residual-AV is defined and assembled.

C. UNet-VGG16-AV

Transfer learning is a popular paradigm in deep learning where networks trained to solve one problem are used to solve another similar problem. Today there exist multiple networks that have been trained to classify images from the vast ImageNet database. Such networks will, without doubt, have the ability to extract features better than an untrained network. Such capabilities may also extend when applied to a totally different dataset such as the EndoCV dataset used in this project. For the third design, I used the first 25 layers of a pre-trained VGG16 network (pre-trained on ImageNet) to replace the Encoder of UNet-AV. The Decoder and other parameters are similar to UNet-AV. This new UNet-VGG16-AV network was then trained on the EndoCV2020 dataset. Refer to the *DemoArchitecture_UNet_VGG16_AV.m* script in the accompanying Demo package for a demonstration of how UNet-Residual-AV is defined and assembled.

V. Results

All three networks were trained on the EndoCV2020 dataset. The original UNet was also trained on the same dataset to provide a baseline for comparison. Note that training was carried out until such a point where the validation loss would not lower any further. This typically occurred after 300 Epochs. The trained networks were then evaluated using the Test dataset. Quantitative and Qualitative Results have been collected and represented in Figures 5, 6 and 7.

A. Quantitative Comparison

Figure 5 displays the confusion matrices for each of the four networks based on Pixel Label prediction accuracy measured based on performance on the Test dataset. Accuracy in this context refers to the ratio of correctly classified pixels in each class to the total number of pixels belonging to that class according to the ground truth.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

$$TP/TN = True\ Positive/True\ Negative$$
$$FP/FN = False\ Positive/False\ Negative$$

Judging from the diagonal of the confusion matrices, the baseline UNet has the worst performance while UNet-VGG16-AV performs the best. UNet-AV and UNet-Residual-AV have comparable performance that lies in between the other two networks. All networks, in general, have trouble accurately classifying the under-represented classes, specifically Suspicious and Cancer. Three out of the four networks perform poorly for these two classes (less than 10%). Only UNet-VGG16-AV demonstrates classification accuracy greater than 20% for these two classes. In terms of Accuracy, it is evident that UNet-VGG16-AV performs the best.

Figure 6 shows some more statistics that have been computed based on the performance of each of these networks on the Test dataset. Figure 6 (a) reports some global statistics over all classes and Figure 6 (b) reports per-class statistics. A colorscale has been applied on each row to for easy comparison. UNet-VGG16-AV is clearly the best performing network and the baseline UNet is the worst performing one. But this is merely relative performance.

A closer inspection of the results of UNet-VGG16-AV will indicate that its performance is not exactly stellar in the context of the semantic segmentation problem. While this network has high accuracies for each class, the Intersection over Union (IoU) scores are pretty low (less than 10%) for under-represented classes such as Suspicious and Cancer. The IoU gives a measure of the amount of overlap between the predicted and ground truth segmented regions. These scores need to be higher for better semantic segmentation performance.

		Network Architecture			
		UNet	UNet-AV	UNet-Residual-AV	UNet-VGG16-AV
Statistics	Global Accuracy	0.399	0.659	0.662	0.705
	Mean Accuracy	0.251	0.346	0.334	0.440
	Mean IoU	0.124	0.221	0.220	0.285
	Weighted IoU	0.327	0.544	0.550	0.609
	Mean BF Score	0.147	0.228	0.221	0.257

Note: Color Scales Have Been Used for Comparison. In a given Row, Highest = Green & Lowest = Red.

(a) Global Statistics

Class	Statistic	Network Architecture			
		UNet	UNet-AV	UNet-Residual-AV	UNet-VGG16-AV
BE	Accuracy	0.175	0.390	0.381	0.413
	IoU	0.109	0.242	0.237	0.268
	Mean BF Score	0.159	0.205	0.205	0.233
Suspicious	Accuracy	0.137	0.041	0.062	0.195
	IoU	0.025	0.023	0.035	0.070
	Mean BF Score	0.053	0.062	0.073	0.103
HGD	Accuracy	0.248	0.335	0.268	0.451
	IoU	0.060	0.124	0.109	0.196
	Mean BF Score	0.073	0.105	0.083	0.139
Cancer	Accuracy	0.091	0.032	0.088	0.204
	IoU	0.019	0.021	0.037	0.067
	Mean BF Score	0.081	0.085	0.103	0.155
Polyp	Accuracy	0.394	0.512	0.426	0.570
	IoU	0.118	0.242	0.222	0.370
	Mean BF Score	0.095	0.117	0.088	0.184
Background	Accuracy	0.459	0.766	0.782	0.806
	IoU	0.412	0.671	0.682	0.736
	Mean BF Score	0.203	0.348	0.344	0.354

Note: Color Scales Have Been Used for Comparison. In a given Row, Highest = Green & Lowest = Red.

(b) Statistics per Class

Fig. 6: Statistics

B. Qualitative Comparison

A few images from the Test dataset were randomly chosen. Their ground truth segmentation outputs as well as predicted segmentation outputs from each of the four networks are tabulated and displayed in Figure 7. This helps compare the performance qualitatively. It also helps one better appreciate the statistics from the previous section. A couple of observations are listed.

1) Observations are largely in-line with the statistics from the previous section. The baseline UNet exhibits poor segmentation performance out of all four networks. Out of the remaining three, UNet-VGG16-AV does relatively better.
2) The predicted results are poor for under-represented classes. For instance, consider Row 4 and Row 6 where segmented regions are from the Cancer and Suspicious classes.
3) The predicted results are poor whenever there are more than two segmented regions in an image (Row 2 and Row 3). This could also be attributed to deficiencies in the dataset where the number of images having more than 2 classes of diseases is very less.
4) The Polyps class has lesser representation than the BE class, yet UNet-VGG16-AV is able to accurately segment it (Row 1 and Row 5).

VI. Discussion

The UNet architecture is very promising and lauded for its ability to learn from much sparser input. This need not necessarily translate to all segmentation problems. As has been seen in the course of this project, the baseline UNet does not perform well when trained on the endoscopic images. That said, by tweaking the design to remove dropout layers and changing the weight initialization method (UNet-AV), the network performance greatly improved. While this may be a positive development, it also makes apparent the highly empirical nature of deep learning.

Residual units don't seem to affect the performance of UNets all that much (as seen between UNet-AV and UNet-Residual-AV). This may be because of the fact that the UNet design already contains skip connections that help merge inferences between higher and lower layers although the nature of the combination is different (addition for residual block as opposed to concatenation for UNet skip connections). Perhaps having both structures is of little benefit.

Transfer learning definitely seems to improve overall performance. Having a VGG16 network, which was trained on ImageNet, as the Encoder in a UNet definitely helps boost performance as was seen with UNet-VGG16-AV. It may be worth plugging in different trained networks into the Encoder portion of UNet to see if higher performance can be attained.

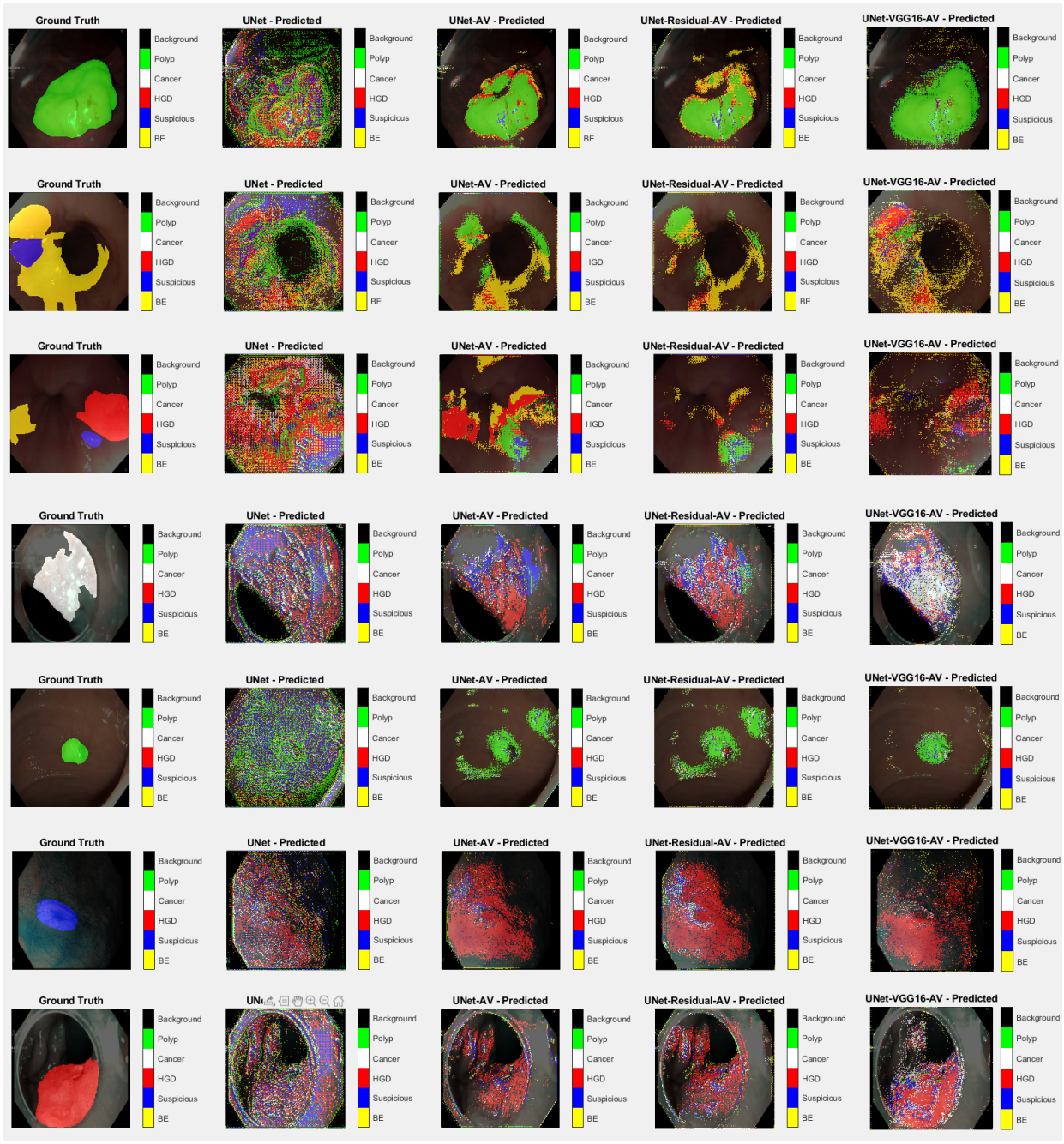


Fig. 7: Sample Segmentation Results

Aside from the perspective of design, the data is also a crucial factor. It would definitely help to have a dataset which wasn't inherently unbalanced. In addition to class level imbalance, there is also a dearth of samples containing multiple and concurrent classes of diseases. Intuitively this is a bigger threat since a lack of observable samples will definitely impede learning. Data augmentation cannot help matters in these scenarios either.

VII. Conclusions

The motive of this project was to design a solution based on UNet to perform semantic segmentation of endoscopic images in order to accurately identify diseased regions. Although a perfect solution was not achieved, the following contributions were made.

1) Built a complete pipeline for performing semantic segmentation of endoscopic images.
2) Constructed, trained and compared (qualitatively and quantitatively) performance of UNet and three unique

UNet-like architectures.

3) Discussed the efficacy of UNet-like architectures for semantic segmentation tasks.

For future work, it may be worthwhile to take advantage of transfer learning and plug in more varied designs into the encoder portion of the UNet. Additionally, there are underlying deficiencies with available data which need to be addressed in order to develop the best model.

REFERENCES

[1] D. H. HUBEL and T. N. WIESEL, "Receptive fields of single neurones in the cat's striate cortex," *The Journal of physiology*, vol. 148, no. 3, pp. 574–591, Oct 1959, 14403679[pmid]. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/14403679

[2] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, no. 4, pp. 193–202, 1980. [Online]. Available: https://doi.org/10.1007/BF00344251

[3] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.

[6] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

[7] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3431–3440.

[8] 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*. Cham: Springer International Publishing, 2015, pp. 234–241.

[9] "Endoscopy disease detection and segmentation (edd2020)," 2020. [Online]. Available: http://dx.doi.org/10.21227/f8xg-wb80

[10] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014.

[11] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, Y. W. Teh and M. Titterington, Eds., vol. 9. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256. [Online]. Available: http://proceedings.mlr.press/v9/glorot10a.html

[12] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ser. ICCV '15. USA: IEEE Computer Society, 2015, p. 1026–1034. [Online]. Available: https://doi.org/10.1109/ICCV.2015.123

[13] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Cham: Springer International Publishing, 2018, pp. 3–11.