Benchmark Analysis for Modified UNet Models with Medical Images - A Case Study with Eosinophilic Esophagitis (EoE) Images and Future Research Opportunities

Kevin Lin
School of Data Science
University of Virginia
Charlottesville, VA
pex7ps@virginia.edu

Jason Wang
School of Data Science
University of Virginia
Charlottesville, VA
jyw5hw@virginia.edu

Ian Liu
Dept. of Computer Science
University of Virginia
Charlottesville, VA
yl4dt@virginia.edu

Judy Fox
School of Data Science
University of Virginia
Charlottesville, VA
ckw9mp@virginia.edu

Donald Brown
School of Data Science
University of Virginia
Charlottesville, VA
deb@virginia.edu

***Abstract*—This paper examines the effectiveness of modified U-Net models for processing biopsy images of Eosinophilic Esophagitis (EoE), a chronic disease characterized by the presence of eosinophils in the esophagus. The original U-Net model has been proven to be effective in generating outperformed prediction accuracy while processing semantic segmentation and biomedical image segmentation tasks, however, its requirement on excessive computational costs is not neglectable. To address this issue, cutting-edge convolutional mechanisms like Dilated Convolutional Layers and Separable Convolutional Layers are proposed and implemented, and their performance is assessed. In addition, to assess the efficiency of these modifications, a series of experiments are conducted on various GPUs to obtain a benchmark comparison, model generalizability and scalability are tested. We also suggest possible avenues for further research to improve the efficiency of the modified U-Net models. These may include exploring the use of transfer learning, implementing other advanced convolutional layers, and exploring different architectures for the modified U-Net models.**

***Index Terms*—Image Segmentation, Medical Imaging, Computer Vision, Deep Learning**

I. Introduction

Eosinophilic esophagitis (EoE) is a chronic disease characterized by the prevalence of a type of white blood cell (eosinophil) in the esophagus. EoE affects approximately 5 to 10 per 10,000 people and can be seen in 2-7% of patients that undergo endoscopy for any reason [1]. EoE is believed to be triggered by dietary components in patients and is increasing in prevalence [2]. Clinical symptoms include swallowing difficulties, food impaction, and chest pain [3]. For diagnosis, patients presenting symptoms related to EoE undergo an endoscopy where the collected biopsy tissue samples are then evaluated for concentration of eosinophils. The accepted criterion for pathologists to diagnose EoE involves identifying at least one High-Power Field (HPF; 400× magnification adjustment) within a patient's tissue biopsy slide that contains 15 or more eosinophils [4]. Our dataset is obtained from the Gastroenterology Data Science Lab at the UVA Hospital. A sample image is given in Fig. 1. Each image is 512x512x3 large and there are 514 images/masks in the dataset spanning 30 UVA Medical Center patients. We keep the three channels [r,g,b] for the image but will import the masks as grayscale. All data is obtained from subjects under conditions of academic use only and no personal health information (PHI) is present in the data. On the subject of eosinophil detection, we can use the cross-domain adaptation method proposed in [5] by subsetting the dataset and introducing it incrementally to the model (e.g. Slides from patients that had 0-4 eosinophils, 5-9 eosinophils, 10-14 eosinophils, 15+ eosinophils). With this division we would also get not only a binary answer on whether or not the patient has EoE but also if the patient could be a potential borderline case (10-14 eosinophils). Additionally, there would be further information about the distribution of the number of eosinophils per slide and give the model a chance to learn what the data presents at the boundary conditions. The such results can provide a measure of uncertainty through respective confusion matrices for each subset of the data in addition to one for the full dataset.

The main approach for EoE is that of a segmentation problem and not a classification problem. This work with the cross-domain adaptation method would provide insight into whether there is or is not EoEs in a given image versus a segmentation algorithm that would be able to label pixels as eosinophils or other cells or just empty space. We could approach a segmentation model with a similar methodology by trying to subset the data and feeding it periodically to the model. For measures of uncertainty, we can use a modified Monte Carlo Dropout approach using this cross-domain adaptation. We will treat the data as discrete since the image is made up of a three-dimensional tuple [r, g, b]. Since we have a very large dataset, we will use variational approximation, specifically minimizing KL divergence, in order to approximate the distribution of the images. Since maximizing the evidence lower bound (ELBO) is mainly impractical [6], we will be using a Monte-Carlo approximation. We can use Monte Carlo (MC) Dropout as a Bayesian approximation to identify which image segments

correspond to different cells. In Bayesian neural networks, each weight is represented by a probability distribution, in which we will assume Gaussian instead of a number. The learning aspect corresponds to Bayesian inference which we will use MC Sampling. We will measure uncertainty for every pixel using cross-entropy over two classes of "background" and "foreground". In addition, a Monte Carlo Dropout U-Net is equivalent to the deep Gaussian process used in Bayesian Neural Networks [7]. Essentially, we can minimize the KL divergence using approximation through Monte Carlo integration to get an unbiased estimate. For the Monte Carlo Dropout U-Net, we apply dropout before every weight where dropout is defined as switching off neurons at each training step.

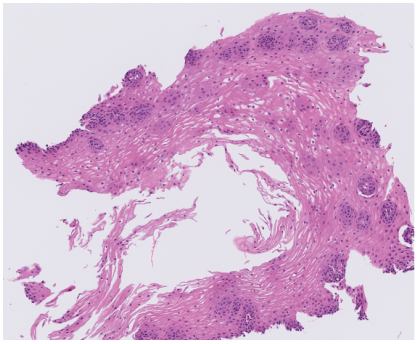

Fig. 1. An Example Image Data from Gastroenterology Data Science Lab

II. Methodology

A. Prior Work

The U-Net is convolutional network architecture for fast and precise segmentation of images [8]. It has been shown to outperform what was previously considered the best method (a sliding-window convolutional network) on the ISBI challenge for segmentation of neuronal structures in electron microscopic stacks. For the U-Net model, we used 23 convolutional layers with batch normalization. In both the encoder and decoder steps, we used a ReLU activation function, and for the final layer we used a sigmoid activation function. Loss was computed using binary cross entropy and ADAM was used as the optimizer. To prevent overfitting, we implemented early stopping and data augmentation.

The underlying architecture of the U-Net model is shown in Fig. 2. At first, the encoder is used to obtain and normalize the transformation of the input volume, using a Leaky ReLU activation function at each layer. At the bottleneck of this architecture, the volume will be in the size of 2×2×2 which represents the reduction of dimensionality prior to using a sigmoid activation function for segmentation. The decoder then up-samples this transformed 2×2×2 volume to reconstruct the image with this segmentation.

Once again, our dataset is 514 images from the Gastroenterology Data Science Lab and sized 5112x512x3. To make improvements on the project conducted by the pioneers who first utilized U-Net Models on EoE imaegs [9], we augment the data through flipping and rotation for each image which

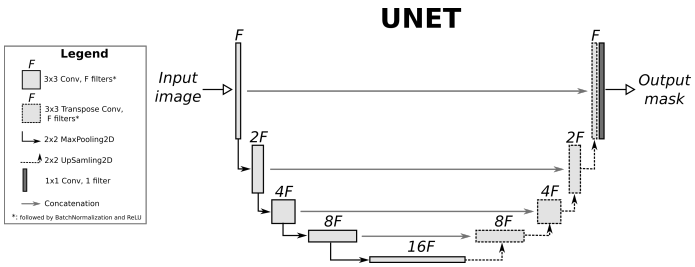

Fig. 2. U-Net Architecture

means that for each input image, we have created a flipped and rotated image as well, effectively tripling our input dataset. Data augmentation here makes the model generalize better due to the larger amount of training data. For comparison, our U-Net results are shown in Table I.

Most significantly, we can see that our approach has at least one order of magnitude less in size which means our model is less complex. Given that the same U-Net approach, we can assume, holding all other factors such as GPU availability constant, our approach shall run faster and more efficiently. Even at their most complicated model, our model performance easily outperforms Adorno et. al.'s. As their model outperformed other approaches such as Residual U-Net, R2U-Net, and Attention U-Net, our approach is expected to outperform these approaches.

However, it is important to note that we devoted a significant amount of time to tune parameters. Additionally, we used binary cross entropy as our loss during training which is not the ideal metric when dealing with medical segmentation data. These results further the statement that suboptimal approaches with good parameter estimates are better than optimal approaches with bad parameter estimates. Further work here would be to try to show that these results generalize to other medical imaging datasets and to test to see if using dice loss as our training evaluation metric would improve performance.

Regarding the metric that is used to measure performance, we choose to use the DICE Coefficient, which is commonly used in image segmentation tasks to evaluate the similarity between the predicted segmentation mask and the ground truth mask. It is a popular metric because it is intuitive, easy to interpret, and provides a robust evaluation of the model's performance. Reasons are below:

- It is more sensitive to small differences: The Dice coefficient measures the overlap between two sets, which is especially important when evaluating the performance of image segmentation models. It is more sensitive to small differences in segmentation masks than other metrics such as pixel accuracy or Jaccard index.
- It provides a balanced evaluation: The Dice coefficient takes into account both false positives and false negatives, providing a balanced evaluation of the segmentation performance. This is important because in some applications, false positives may be more tolerable than false negatives, and vice versa.

- It is robust to class imbalance: In image segmentation tasks, there may be a class imbalance where one class has significantly fewer pixels than the other classes. The Dice coefficient is robust to class imbalance and provides a fair evaluation of the model's performance.

The DICE Coefficient mathematically expressed below,

$$DICE = \frac{2 * |X \cap Y|}{|X| + |Y|} \tag{1}$$

Where X is the predicted set of pixels and Y is the ground truth.

B. Computer Vision Approaches

When preforming image segmentation tasks in medical imaging, one of the biggest challenges is not only to achieve high accuracy but also obtaining faster inference. This is often achieved by lightening the model by the number of layers or parameters. The previously discussed U-Net architecture is comprised of many 2D convolutional layers, where a 2D convolution is applied over an input signal that is composed of many input planes. Current research has identified many other convolution mechanisms that can improve performance as well as decrease training time in U-Nets [10]–[12].

- Dilated Convolutional Layers: Dilated convolution is a technique that adapts the traditional convolution kernel by inserting spaces between consecutive elements. The dilation factor (l) determines how much space is put between the consecutive elements. The purpose is to expand the receptive field of the U-Net without having to increase the computational cost. In fact, with dilated convolutions the computational cost should stay the exact same as the number of parameters stays the same as well. A novel U-Net variant containing dilated convolutions where instead of two standard 2D convolutions was proposed and implemented and the results showed the modification outperformed the vanilla U-Net while using fewer parameters and therefore increased training efficiency [13].
- Separable Convolutional layers: Separable convolution layers are similar to dilated in a way that they try and reduce the number of parameters and computational cost. Separable layers instead try and divide a single convolution into two or more convolutions that produce the same output. This reduces the number of individual multiplications and should improve training efficiency. There are two types of separable convolutions: spatially separable convolutions and depth-wise separable convolutions. Spatially separable convolutions operates on the image width and height while depth-wise separable convolutions operates on the depth dimension which is common in images as they contain an RGB channel. This approach has demonstrated the advantages of using separable convolution in the U-Net architecture [14]. By using depth-wise convolutions in the entire network, they were able to make a lightweight CNN that was a smaller model, had fewer parameters, and therefore had faster inference time. This proposed SD-UNet was able to achieve comparable or even sometimes better results than vanilla U-Nets and current state of the art.

III. Models, Experiments and Benchmark Analysis

A. Dilatyed Convolution Layers

To investigate the effect of dilated convolution layers on the performance and efficiency of U-Nets, we took the previous architecture discussed above and modified each convolution 2D layer into a dilated convolution layer. This just impacts the encoder part of the U-Net as the decoder still needs to have transposed convolution layers. Next we trained U-Nets at various different levels of dilation factors. Previous research suggested that we keep the dilation factor between 2-16, which led us to select four factors [2,4,8,16]. We were also facilitated by the concept of pyramid dilated convolution where the dilation factor is changed inside an encoder block [15].

As previously mentioned, the dilated layers were kept constant throughout, but we also wanted to explore how variable dilation factors affected performance. Given we have two convolution layers in each block, we set the first dilated convolution to a dilation factor of 3, and the second dilated convolution to a dilation factor of 6. All training was conducted twice on a single A100 GPU with 80GB of RAM in PyTorch 1.10. The main purpose of dilated layers is to improve efficiency, so we also noted the time it took to train the entire U-Net with each level of the dilation factor, results are shown in Table I.

B. Separable Convolution Layers

Similarly to the dilated convolution layers, we wanted to investigate the effects of separable convolution layers on the performance and training efficiency of U-Nets. Again we replaced each convolution layer in the U-Net encoder with a spatially separable convolution as well as a depth-wise convolutional layer. The results are presented in Table I below. Training was again done on an A100 GPU with 80GB of RAM to compare training times to the previous dilated results. Time to train is presented in Table I again.

C. Benchmark Analysis

The benchmark analysis consists two subtasks: rerun the models on A100 and V100 GPUs with 80GB of RAM. Reasons are below:

- **Performance optimization:** Different GPUs may have different architectures, memory cap activities, and processing speeds. By training the same models on multiple GPUs, we wanted to know if we can determine which GPU configuration provides the best performance for the modified U-Net models. We believe that this approach could help us further optimize the training process and improve the overall performance of the models.
- **Robustness testing:** Training a model on different GPUs can help us to ensure that the modes are robust and can handle a range of hardware configurations. We believe

this is particularly important because our models are intended to be deployed on different types of hardware, as it can help to identify potential performance issues or hardware-specific bugs.

- **Reproducibility:** By training the same models on multiple GPUs,we ensure that the results are fully reproducible and not dependent on a specific hardware configuration. This helps to increase the confidence in the results and ensure that the model is reliable and robust.
- **Resource allocation:** Depending on the resources available, it may be necessary to distribute the training process across multiple GPUs to complete the training in a reasonable amount of time if the models get to be deployed in the real medical setting. In this case, comparing the results of the model trained on different GPUs can help to determine the optimal distribution of resources.

IV. Discussion

A. Dilated Layers

As mentioned and shown in Tanle I, all levels of dilation as well as the pyramid dilated convolution architecture performed worse than the original U-Net. The dilation factor that preformed best was at a level of 4, followed closely by the pyramid implementation. This could be due to a variety of reasons, including the range of the receptive field. As the dilation factor increased from 4, the performance dramatically worsened, dropping all the way to a median dice score of 0.126. This is due to the fact that the receptive field is too large in comparison to the original 512x512 image, and therefore all local pixel relationships are lost. The pyramid implementation worked fairly well as it was in the dilation factor range of 3 and 6, which explains why its performance was very similar to a dilation factor of 4. All dilation factors increased the training time of the vanilla U-Net by a couple seconds, with higher dilation factors and pyramid implementation increasing the number of seconds. This is explained by the fact that dilated layers increase the receptive field without increasing computational requirements. Next steps would include changing the architecture of the UNet to adapt to the increase in receptive field and decrease the training time.

B. Separable Layers

Both versions of separable convolutions led to similar but worse performance when compared to the vanilla U-Net. Spatially separated convolutions led to a median Dice score of 0.533 while depth-wise led to a median Dice score of 0.558. This is in line with the premise of separable convolutions where a single convolution is divided into two or more convolutions that lead to similar results. In this case performance was degraded but not significantly. The area which performed much worse than the original U-Net was in training time, where spatially separated convolutions and depth-wise separated convolutions were trained in 383.41 and 376.16 seconds respectively. This is over 2x the amount of time required to train the original U-Net. This could be due to many factors including the TensorFlow implementation of these layers. However more research needs to be done to understand why the compute time increased so dramatically.

C. Future Work

Overall all adaptations of U-Net hindered the performance and training efficiency when compared to the original U-Net. All of these adaptations included changes to the convolutional layers, however the architecture was kept the same. Future work could include changing the architecture of the U-Net in regards to the type of convolutional layers used. For example, with the larger receptive field of dilated layers, not as many encoder blocks would be required as it would account for more blocks. This would also significantly improve training efficiency as well as improve performance as well. Similar conclusions can be said for separable convolutional layers, where different implementations can improve training efficiency as well as performance.

To further advance the progress of this project and address the issues of overfitting for more complex networks, we intend to refine the existing network structure. Primarily, in cases where suboptimal boundary segmentation results are obtained, we plan to integrate an upsampling branch structure. Furthermore, we aim to augment the network architecture by incorporating separate branches for learning edge information, such as GSCNN. Additionally, we will explore other U-Net-based network models, such as U-Net++ [16], R2U-Net, Attention U-Net, and Residual U-Net. Furthermore, we assume the subpar performance can also be addressed by utilizing other types of the loss function. Focal Loss, MSE and DICE loss as well as their combination will be implemented and tested out.

As shown in Table I, the difference between the final medium dice in benchmark training on different GPUs is more than 0.1 which may be too large. We believe that this happened due to unstable gradient decline unstable(e.g. The batch size in the benchmark was set to 4-5), and the effect of the BN layer is reduced. Further, we will try to reduce the length and width of the input by 2-8 times and increase the batch size by random cropping.

Acknowledgment

The authors would like to thank Dr. Sana Syed and Adam Greene of the UVA Gastroenterology (GI) Data Science Laboratory for obtaining the EoE dataset through biopsies of UVA Medical Center patients. Additionally, the staff of the GI Data Science Laboratory provided direct medical feedback on our results.

References

[1] E. S. Dellon, "Epidemiology of eosinophilic esophagitis," *Gastroenterology Clinics*, vol. 43, no. 2, pp. 201–218, 2014.

[2] S. Carr, E. S. Chan, and W. Watson, "Eosinophilic esophagitis," *Allergy, Asthma & Clinical Immunology*, vol. 14, no. 2, pp. 1–11, 2018.

[3] T. M. Runge, S. Eluri, C. C. Cotton, C. M. Burk, J. T. Woosley, N. J. Shaheen, and E. S. Dellon, "Causes and outcomes of esophageal perforation in eosinophilic esophagitis," *Journal of clinical gastroenterology*, vol. 51, no. 9, p. 805, 2017.

TABLE I
BENCHMARK RESULTS ON A-100 AND V-100 GPUS IN SECONDS

Benchmark	GPU	Original U-Net	DF-2	DF-4	DF-8	DF-16	Pyramid	Separable-Layer	Depth-Wise
Training Time	A100_KJ	145.05	145.64	149.53	150.33	151.07	151.13	383.41	376.16
	A100_IL	138.98	141.28	140.59	141.11	142.4	141.482	322.19	N/A
	V100_IL	169.23	180.74	175.17	174.934	181.436	175.31	375.75	N/A
DICE Score	A100_KJ	0.689	0.492	0.529	0.372	0.126	0.525	0.533	0.558
	A100_IL	0.673	0.56	0.434	0.423	0.211	0.535	0.534	N/A
	V100_IL	0.582	0.492	0.565	0.412	0.186	0.399	0.598	N/A
Data_Loading	A100_KJ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	A100_IL	2.472	2.503	2.57	2.538	2.685	2.426	2.43	N/A
	V100_IL	3.345	3.364	3.36	3.462	3.432	3.251	3.473	N/A
Model_Loading	A100_KJ	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
	A100_IL	43.28	54.506	55.289	57.103	58.318	54.315	43.457	N/A
	V100_IL	74.1	90.01	90.535	92.607	96.427	89.283	73.962	N/A

[4] G. T. Furuta, C. A. Liacouras, M. H. Collins, S. K. Gupta, C. Justinich, P. E. Putnam, P. Bonis, E. Hassall, A. Straumann, M. E. Rothenberg *et al.*, "Eosinophilic esophagitis in children and adults: a systematic review and consensus recommendations for diagnosis and treatment: sponsored by the american gastroenterological association (aga) institute and north american society of pediatric gastroenterology, hepatology, and nutrition," *Gastroenterology*, vol. 133, no. 4, pp. 1342–1363, 2007.

[5] T. Hassan, B. Hassan, M. U. Akram, S. Hashmi, A. H. Taguri, and N. Werghi, "Incremental cross-domain adaptation for robust retinopathy screening via bayesian deep learning," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–14, 2021.

[6] J.-T. Chien and C.-J. Tsai, "Amortized mixture prior for variational sequence generation," in *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2020, pp. 1–6.

[7] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[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: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.

[9] W. Adorno III, A. Catalano, L. Ehsan, H. V. von Eckstaedt, B. Barnes, E. McGowan, S. Syed, and D. E. Brown, "Advancing eosinophilic esophagitis diagnosis and phenotype assessment with deep learning computer vision," in *Biomedical engineering systems and technologies, international joint conference, BIOSTEC... revised selected papers. BIOSTEC (Conference)*, vol. 2021. NIH Public Access, 2021, p. 44.

[10] X.-X. Yin, L. Sun, Y. Fu, R. Lu, and Y. Zhang, "U-net-based medical image segmentation," *Journal of Healthcare Engineering*, vol. 2022, 2022.

[11] G. Du, X. Cao, J. Liang, X. Chen, and Y. Zhan, "Medical image segmentation based on u-net: A review," *Journal of Imaging Science and Technology*, 2020.

[12] C. Huang, H. Han, Q. Yao, S. Zhu, and S. K. Zhou, "3d u 2-net: A 3d universal u-net for multi-domain medical image segmentation," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II*. Springer, 2019, pp. 291–299.

[13] S. Wang, S.-Y. Hu, E. Cheah, X. Wang, J. Wang, L. Chen, M. Baikpour, A. Ozturk, Q. Li, S.-H. Chou *et al.*, "U-net using stacked dilated convolutions for medical image segmentation," *arXiv preprint arXiv:2004.03466*, 2020.

[14] P. K. Gadosey, Y. Li, E. A. Agyekum, T. Zhang, Z. Liu, P. T. Yamak, and F. Essaf, "Sd-unet: Stripping down u-net for segmentation of biomedical images on platforms with low computational budgets," *Diagnostics*, vol. 10, no. 2, p. 110, 2020.

[15] W. Zhang, X. Lu, Y. Gu, Y. Liu, X. Meng, and J. Li, "A robust iris segmentation scheme based on improved u-net," *IEEE Access*, vol. 7, pp. 85 082–85 089, 2019.

[16] 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: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018, Proceedings 4*. Springer, 2018, pp. 3–11.