Segmentation of Nucleus in Histopathological Images Using Deep Learning Architectures

Ogun Ayaz, Hamdullah Usta, Gokhan Bilgin
Department of Computer Engineering
Yildiz Technical University
Davutpasa Campus 34220 Istanbul, Turkey
{ogunayaz6138, hmdllh.st.17}@gmail.com, gbilgin@yildiz.edu.tr

***Abstract*—The aim of this study is to develop a image segmentation system for Histopathological images by using Deep Learning Methods. In today's world cancer is a world wide problem for a lot of people from all around the world. Over several years in scientific studies showed us that scientist tried to find a way to cure cancer or reduce the damage or increase the chance of survival from cancer. As studies shows that early diagnosis of the cancer plays a huge part on the treatment process and highly increases the chance of survival of the cancer patients. Most of the cancer diagnosis system uses a medical image analyze. However most of the cancer types requires is a large amount of test and examination on medical cell images. It is a hard and time consuming process. As a result of this, there is a need has arisen for image segmentation. With the development on computer science, computers started to play a big role for making this process much more fast and accurate. Recent development on Machine Learning and Deep Learning methods can provide efficient image segmentation for large amount of data in a agreeable time line. There was a lot of studies which focused on traditional methods for image segmentation but in this study we aim on the recent development on Deep Learning methods which is used for image segmentation. First of all, an introduction about image segmentation for Histopathological images by Deep Learning Methods is given. After that, literature review about recent studies which is covered our main topic is given. Then, our feasibility studies and system analysis is addressed. Then, implementation and performance studies is given. Finally, an assessment of this study on image segmentation by Deep Learning methods into clinical images are given. The main topic as Deep Learning methods we targeted U-Net, Mask R-CNN, UNet++ and YOLO. We implemented those models to the medical histopathologial image data. We provided the process, result and analyzed outcome and the summary for all the models that we used in this paper.**

***Keywords*—Histopathological images, deep learning, nuclei segmentation, instance segmentation, semantic segmentation.**

I. Introduction

Deep learning have been used widely to a lot of different problems in the past few years. Especially for object detection and image segmentation which is mostly used in fields of medical, industry and military [1]. With the acceleration of developments in computer science, Deep learning techniques started to lead in the way to reach solution in the real life problems. Deep learning techniques was used for computer vision, natural language processing and analysis [2], [**?**]. In most of these fields deep learning techniques has passed their computations. In this paper we focused on the medical field application and the way they implemented deep learning models into histopathological image segmentation process [3].

In today's world, humankind faces variety of problems each day. Numerous studies try to find a way to help people on this and save humankind as much as possible. As many studies shows that early diagnosis plays a key role to increase the possibility of recovery and provide a better result. With the development on Neural Network and deep learning models, they started to use for segmenting and classifying medical images [4], [5]. Especially cancer diagnosis deeply needs improvement because there are a lot of images to analyze and find out if it there is a high possibility of cancer. That's why a lot of studies try to address this problem by studying on deep learning technologies [6], [7]. Convolutional Neural Network has been used for couple times on deep learning technologies. We have seen multiple new techniques such as SegNet, U-Net, UNet++, Mask R-CNN, DeepLabVx and so on. Most of these new techniques has been used for medical image segmentation. Also some of them is provided specially for medical image segmentation. We focused on U-Net,Mask R-CNN and UNet++ in this study because we believe that these deep learning algorithms is more suitable for us to work and achieve our goal for image segmentation.

In this paper, we provide an overview of recent development on deep learning techniques which has been used to build a image segmentation model for Histopathological images. We provide a wide detailed information and comparison over Mask R-CNN, U-Net and UNet++ deep learning models with result and deep analysis over our implementations.

II. Materials and Methods

A. Experimental Data set

In this study, we used a data set contains large amount of nuclei images which was collected under various conditions. It is very important to identify cells before machine learning. Because there are more than 30 trillion cells in the human body, and the nuclei of these cells contain DNA that defines us. Recognizing nuclei means recognizing cells. And so, as researchers on the job, we learn about biological processes

978-1-6654-3663-2/21/ $31.00

and enable us to find cures for diseases. We got the data from 2018 Data Science Bowl which we gave the cite to the website. In this study, Over 500 images were used from the 400MB data set for each model. The data set can be found in Kaggle platform to perform .[8]

B. Implementation of Deep Learning Algorithms

Our methodology is described as a summarized flowchart in 1. First of all we collect medical data. After that, these images goes though image process in order to be ready for deep learning. Then, data is used in image segmentation system by deep learning algorithms. Last, we analyze performance of the algorithms.

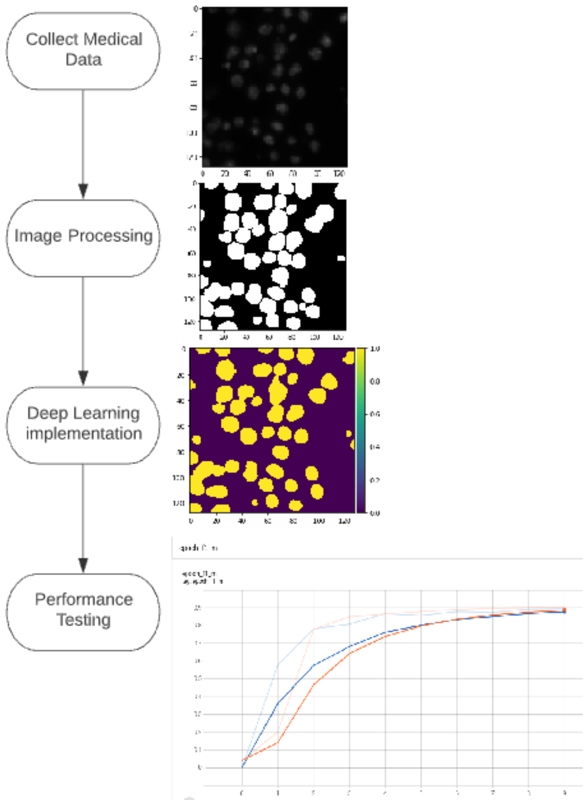

Fig. 1: The flow chart of our experimental studies

U-Net is a convolutional neural network which was first developed at 2015 by Ronneberger, Olaf, Philipp Fischer, and Thomas Brox in the University of Freiburg. The papar called "U-Net: Convolutional Networks for Biomedical Image Segmentation". As you can understand from the title. This convolutional neural network was developed especially for medical image segmentation [9]. U-Net is considered more successful than other deep learning method when it is come to the medical images. It is more effective and powerful even with limited medical images. The features that makes U-Net different from other method is dimension reduction and dimension increase process which is called pooling layer. U-Net networks has two main part which are called a contracting and an expansive path. This gives an U shape on the architecture. This is why this deep learning algorithm called U-net [10]. The contracting path is a common convolutional network that consists of convolutions rectified linear unit (ReLU) and a max pooling operation. The expansive path combines the feature and information through convolution process and combine it with high-resolution from the contracting path. In the Fig.2a, input data are provided whereas the intermediate result after image processing can be seen in Fig.2b and final output is shown in Fig.2c.

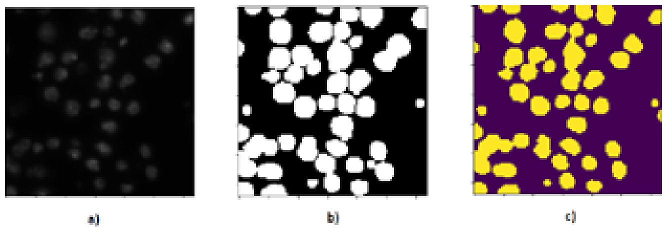

Fig. 2: The output of our experimental U-net study.

Mask R-CNN is introduced in 2017 by Kaiming He, Georgia Gkioxari, Piotr Dollar and Ross Girshick [11]. Mask R-CNN is considered as a extension of Faster-RCNN. Faster-RCNN is a popular object detection method. To sum up, Mask R-CNN is extension of Faster-RCNN which implement output model for predicting a mask for each object detected [12]. Mask R-CNN is built on top of Faster R-CNN. In addition to Faster R-CNN process. Mask R-CNN returns the object mask. Faster R-CNN uses ConvNet to extract maps. Mask R-CNN uses ResNet 101 architecture to extract. This extraction acts as an input for the fallowing layer. After Backbone Model process, Mask R-CNN takes the maps. It applies region proposal network to the maps. In this part, it predicts if an object is exists in the region or not. Then Mask R-CNN proceed with the feature which region proposal network predicted there is an object exists. After Region proposal network part, Region of Interest applies a pooling layer and convert shapes to the same which Mask R-CNN gathers from Region Proposal Network. The regions go through a fully connected network in order to predict class label and bounding. Faster R-CNN uses same path as Mask R-CNN until now. From now on, Mask R-CNN also generates mask which is segmented. Firstly, Mask RCNN finds the region of interest. After that, Mask R-CNN compute Intersection over Union from all predicted regions with ground truth [13]. Mask R-CNN selects regions only which has a Intersection over Union greater or equal to 0.5. Once Mask R-CNN gathers the Region of Interest according to Intersection over Union calculation. Mask R-CNN adds a mask branch to the architecture. After this process, it returns the segmentation mask per region which has an object according to Region Proposal Network. Masks scaled up for interface. In the end, model segments all the object where Mask R-CNN predict the masks for objects [14]. In the Fig.3 the input and output of our studies on Mask R-CNN are shown in order.

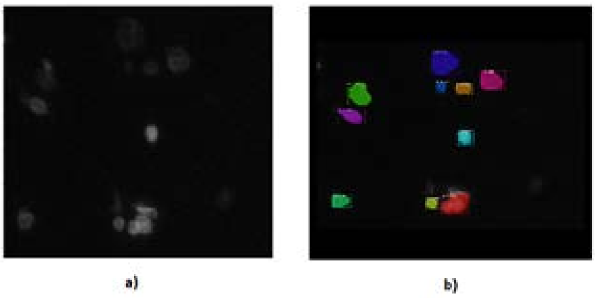

Fig. 3: The output of our experimental Mask R-CNN study.

U-Net++, which called as Nested U-Net, is much more powerful deep learning model which is used especially for medical semantic image segmentation. UNet++ is basically improved and more controlled encoder - decoder U-Net CNN architecture. There are three different feature between U-Net and UNet++. First, there are convolution layers on skip pathways which connects encoder and decoder. Secondly UNet++ has skip pathways has connection between them which helps to improve flow of gradient. Lastly, UNet++ has a deep control over architecture that allow model trimming [15]. UNet++ which is actually nested Unet. As you can understand from the name of it. UNet++ architecture simply improved UNet architecture. The architecture provides encoder and decoder sub networks. UNet++ improved skip pathways which provide communication sub networks for deep supervision. Skip pathways provide solid convolution layers. The blocks combine the output of previous layer with corresponding out of the lower layer with the same dense. To sum up, skip pathways fill the gap between encoder and decoder paths. As you can see in the previous graph that we provided as green part. Those represent the skip pathways. Deep Supervision offers a two different option for model [15]. First one is accurate mode which the outputs of branches are averaged. Second one is fast mode which model select one of the segmentation branches as final segmentation map. The selection is the key for speed gain and model pruning. In the Fig.4 we provided the input and output of the experimental result of our studies in order.

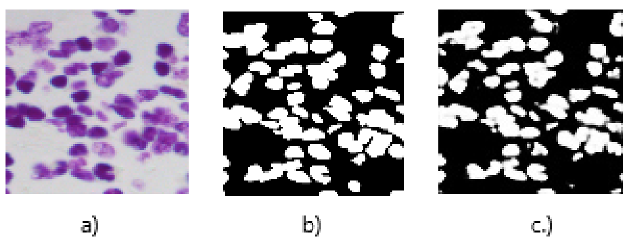

Fig. 4: The output of our experimental U-Net++ method.

YOLO which is also stands for You Only Look Once. This deep learning model is introduced in 2016. This is introduced as a new ways to the real time object detection. They framed this process as a regression problem. Other deep learning models evaluate the images more than once but YOLO developed a way to to this one process [16]. This is what gave them a speed process. When this modal introduced it provided a better speed comparing to other deep learning models. YOLOv5 is what we ran as an implementation so we provided a information architecture. It is a single stage object detection model. It has three important parts. Model Backbone is used to get critical feature from input images that we gave the model. YOLOv5 uses CSP which is Cross Stage Partial Networks. Model Neck is used to create feature pyramids which improve object scaling. This technology is very necessary for finding the exact object with different dimension. It is used for unseen part of data mostly. YOLOv5 uses PAnet as model neck. Model Head basically the final part where model predict. It collapses boxes on feature and get the final result with metrics [17]. In the Fig.5, the input and output of YOLOv5 experimental studies are given in order.

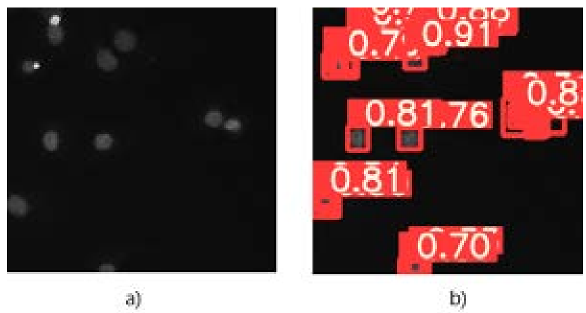

Fig. 5: The output of our experimental YOLOv5 study.

III. Experimental Results

In this section, we provided some performance metrics that we calculated during our implementation which gives us a way to evaluate our model and deep learning methods. In the next table, we provided performance metrics from our implementations.

TABLE I: Performance Metrics

Metrics \Methods	U-Net	Mask R-CNN	UNet++	YOLO
Loss	0.0946	0.0676	0.206	0.166
F1 metric	0.9083	0.6424	0.9185	0.954
Precision(mAP)	0.9167	0.7797	0.9224	0.954
Recall(mAR)	0.9014	0.6244	0.9148	0.954

TABLE II: Computational Environment

Environment\Methods	U-Net	Others
Environment	Local Machine	Google Colab Free
Ram	8 gb	16 gb
GPU	NVIDIA 920 M	NVIDIA Tesla K80
GPU Capacity	4 gb	12 gb
Limit	No Limit	8-12 hours

We faced more obstacles when we worked on Mask R-CNN implementation. In our experience, that's because Mask R-CNN has abilities to offer more than U-Net. One of it is the real time object detection which is the main reason for a lot of people to choose Mask R-CNN over other deep learning algorithms. Also it offers instance segmentation. One of obstacle we faced is the setting up the development environment. Both of the deep learning algorithms require a

TABLE III: EXPERIMENTAL RESULTS

Metrics\Models	U-Net/U-Net++	Mask R-CNN	YOLO
Data	Same	Less	Less
Epoch	10	10	250
Batch Size	16	1	1
Time to train	45 minutes	8 hours	40 minutes
Implementation	Slightly Easier	Slightly Harder	Easier
Data Processing	Slightly Easier	Harder	Harder
Visualization	Slightly Worse	Better	Better
Model Score	Best	Worse	Better

specific configurations on environments which is a little bit hard for us to work with since we both worked remote and have a different environments set up already build on. We faced a lot of different errors which took a lot of time to find solution. The other big problem is training out model. In deep learning model, we need to train our model in order to have a better and stable result. U-Net does not require much data and power to train a model so it was easy for us to train our model. On the other hand, Mask R-CNN requires a detailed data and a lot of machine power to train a stable model. We could not be able to complete training our Mask R-CNN model in our local machines that why we decided to go with Google Colab for training. However, we had to use Free version so we only had 8 hours to train our model which was slightly enough to have a stable result. Google Colab is capable of offering us a great environment but it also hard for us to work with because in every session we had to upload our data to drive or Google Colab so it takes so much time. Plus we had to download model after each epoch otherwise session may time out or our internet connection may lose so we can lose our trained model. To sum up, if our goal is to segment images and do not need to separate them from each other. You may want to work with U-Net since it offers semantic segmentation and requires less time and data to train. On the other, if your goal is to segment images and also find out their specific classification, you may choose Mask R-CNN which offers real time object detection and instance segmentation but you will need a larger data and time to train your model and powerful machine. On the other hand with YOLO and UNet++ we were able to work smoothly. We get a better result and the whole process went without any major problem. We also used Colab to perform the training which was the main reason why we didn't got any major obstacle in the first place.

IV. CONCLUSION

To sum up, in this study we worked on a system where we were able to segment medical data into a better form where any further analyze will be able to apply much more easy and Cancer detection or object detection methods will be able to perform much more fluently. To be able to reach this goals we studied on two of most popular and successful deep learning methods which used for instance and semantic segmentation. Our U-net and U-Net++ implementation was successfully performed semantic segmentation on histopathological images where we were able to archive high F1 score. On the other hand, YOLO also gave a great result on F1 score which is probably the high number of Epoch that we gave but still it is a great result. Also our Mask R-CNN implementation was able to perform instance segmentation on histopathological images which means that our implementation was able to separate objects of our medical data from each other. Technically we used python and tensorflow library which is very powerful and stable way to perform deep learning algorithms.

REFERENCES

[1] A. Albayrak, A. U. Akhan, N. Calik, A. Capar, G. Bilgin, B. U. Toreyin, B. Muezzinoglu, I. Turkmen, and L. Durak-Ata, "A whole-slide image grading benchmark and tissue classification for cervical cancer precursor lesions with inter-observer variability," *Medical & Biological Engineering & Computing*, vol. 59, pp. 1545—-561, 2021.
[2] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, "Image segmentation using deep learning: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.
[3] M. H. Hesamian, W. Jia, X. He, and P. Kennedy, "Deep learning techniques for medical image segmentation: achievements and challenges," *Journal of Digital Imaging*, vol. 32, no. 4, pp. 582–596, 2019.
[4] L. Liu, J. Cheng, Q. Quan, F.-X. Wu, Y.-P. Wang, and J. Wang, "A survey on u-shaped networks in medical image segmentations," *Neurocomputing*, vol. 409, pp. 244–258, 2020.
[5] I. O. Sigirci, A. Albayrak, and G. Bilgin, "Detection of mitotic cells using completed local binary pattern in histopathological images," in *IEEE 23nd Signal Processing and Communications Applications Conference, SIU'15*, 2015, pp. 1078–1081.
[6] M. Van Eijnatten, R. van Dijk, J. Dobbe, G. Streekstra, J. Koivisto, and J. Wolff, "Ct image segmentation methods for bone used in medical additive manufacturing," *Medical engineering & physics*, vol. 51, pp. 6–16, 2018.
[7] N. Hatipoglu and G. Bilgin, "Segmentation of histopathological images with convolutional neural networks using fourier features," in *IEEE 23nd Signal Processing and Communications Applications Conference, SIU'15*, 2015, pp. 455–458.
[8] J. C. Caicedo, A. Goodman, K. W. Karhohs, B. A. Cimini, J. Ackerman, M. Haghighi, C. Heng, T. Becker, M. Doan, C. McQuin *et al.*, "Nucleus segmentation across imaging experiments: the 2018 data science bowl," *Nature Methods*, vol. 16, no. 12, pp. 1247–1253, 2019.
[9] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention, MICCAI'15*. Springer, 2015, pp. 234–241.
[10] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.
[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision, ICCV'17*, 2017, pp. 2961–2969.
[12] P. Sharma. (2019) Computer vision tutorial: Implementing mask r-cnn for image segmentation (with python code), available at www.analyticsvidhya.com/blog/2019/07/computer-vision-implementing-mask-r-cnn-image-segmentation.
[13] Renu Khandelwal. (2019) Computer vision: Instance segmentation with mask r-cnn, available at www.towardsdatascience.com/computer-vision-instance-segmentation-with-mask-r-cnn-7983502fcad1.
[14] W. Abdulla. (2020) Mask r-cnn for object detection and instance segmentation on keras and tensorflow, available at https://github.com/matterport/Mask_RCNN.
[15] Z. Zhou, M. M. R. 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*. Springer, 2018, pp. 3–11.
[16] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European Conference on Computer Vision, ECCV'14*. Springer, 2014, pp. 740–755.
[17] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR'16*, 2016, pp. 779–788.