Medical Image Segmentation Using U-Net

***Abstract*—The U-Net is a convolutional neural network that has a special type of architecture that makes this neural network effective for solving medical image segmentation problems. It has been proven that when we are dealing with deep neural networks, the more data we train, the better result we obtain. However, one of the challenges in some medical image processing problems is the volume of the dataset, but as the number of people dealing with a special illness like cancers or other rare diseases is much lower than healthy people, that is why a dataset with a huge amount of data is not accessible for scientists in all situations. To address this issue, scientists proposed U-Net, a simple but highly effective neural network with an encoder part or contracting path for capturing context, and decoder part or expansive path for accurate localization respectively without using any fully connected layers. The great advantage of the U-Net is, we deal with fewer training data but obtain precise results. My experiment on two datasets shows that we can train a small dataset but still have a precise yield using this state-of-the-art fully convolutional network.**

Index Terms-Feature maps, image segmentation, U-Net, encoder, decoder.

I. Introduction

These days semantic segmentation has become one of the widespread issues in image processing and computer vision, as it covers many applications from autonomous cars to understand the environment of self-driving cars, and medical image diagnostics to analyze CT scan images. To clarify this point, the process of semantic segmentation is assigning one label to each pixel of the image. The aim is labeling each pixel to the corresponded class that helps researchers to do prediction about the regions of the image which are more interesting for them[1].

The task of semantic segmentation is understanding and analyzing regions of interest at the pixel level. To clarify this point, this process enables researchers to have access to the structure of the medical CT scans and process images at the pixel level. As a result, it plays a huge role to detect tumors in the brains or lungs, or other organs, and understand CT scans of people dealing with modern coronavirus (COVID-19), cell segmentation, and other medical applications[2], [3].

In recent years, deep neural networks including convolutional neural networks or CNNs have played a significant role in image processing tasks, and have obtained precise results in comparison with traditional methods. The success of convolutional neural networks depends on the size of the training datasets and the considered model. Since then, convolutional neural networks have been modified to train larger datasets and obtain more precise results [4], [5],[6]. Ciresan et al.[7].used a deep neural network to do a segmentation task on biological neuron membranes, as a pixel classifier which predicted a label of each pixel that is divided into membrane or non-membrane, from raw pixel values in a square window centered on it [4], [7]. Although Ciresan et al.[7]. won the ISBI 2012 EM segmentation challenge, his proposed strategy had two downsides. Firstly, the network was considerably slow, as it had to be run for each patch, which was highly time-consuming, as patches had some overlapping together, that is why his proposed network had the redundancy problem that makes the running process slow [4].Secondly, there was a trade-off between context usage and localization accuracy. To clarify this point, the more patches were larger, the more max-pooling layers they required, and as a result, it reduced the localization accuracy, and small patches allowed the network to understand the little context [4]. More recent strategies had been done on the convolutional neural networks for medical image segmentation, to outperform more precise results than before. The U-Net deep neural network is one of the convolutional neural networks that have been shown promising results for medical image segmentation purposes [4].

The U-Net was designed for semantic segmentation and the original U-Net paper was proposed for image segmentation task to solve the challenge of ISBI cell tracking was held in 2015. This network is called U-Net, as the architecture of this loos like ”U”. The architecture of the U-Net consists of one encoder or contracting path in the left part of the network to capture context and convert features maps to vector after extracting them by learning, and one decoder or expansive path on the right side of the network to do precise localization and reconstruct images from features vectors [4]. In the between the part of the network, we have concatenation part or skip connections that concatenate features maps from encoder to the decoder part of each layer that enables the network to get localized information and makes semantic segmentation possible [4]. What makes U-Net outstanding from other convolutional neural networks is, it can be trained on very few images and shows a promising result precisely. Furthermore, the U-net does not have Cirecan’s et al[7] model problem, as it is fast, and it can complete the segmentation of a 512x512 image in less than a second on a modern GPU [4].

One of the huge benefits of the U-Net is, this network can be trained on very few images of the dataset and shows a promising yield. The contracting path consists of successive layers including convolutions and pooling that are replaced by transposed convolutions, that is why the resulting image will be more clear. To clarify this point, high-resolution features from the encoder part are concatenated by skip connections to the upconvolution output for localization, then after one

training based on the result of the network and backpropagation process, the value of the weights will be modified to obtain a better result. Furthermore, The U-Net architecture does not have any fully connected layers or (FC), and it uses only convolutions and transposed convolutions with different filters [4].

One important part of the U-Net architecture is the upsampling with different channels, which allows the network to do propagation and as a result, the better yield will be shown in the next training. That is why the decoder part is less symmetric than the encoder part. As the purpose is to show the promising result of the U-net for datasets with fewer images, data augmentation is necessary to obtain a more precise result, and it also is the cheapest way to train more data. The beauty of the original U-net paper is it does not have many mathematical equations and everything is about the architecture of the network and it describes only one equation [4]. To summarize the main part of this report includes:

- I introduced the architecture of the U-Net including encoder, decoder, and skip connections.
- I showed that the U-Net has promising results with fewer data.

The remaining parts of this report are as follows. In section II, I described image segmentation methods that had been done before. In section III, I described the architecture of the U-net in more detail. In section IV and V, I showed my observations and results. The final parts are conclusions and references.

II. RELATED WORK

The aim of the semantic segmentation in processing medical images is labeling each pixel of the interesting objects in medical images, and it can be useful in medical treatments as it plays an important role in detecting diseases, and it also gives researchers full access to the structure of the image. Before emerging deep neural networks, scientists had to apply traditional methods of semantic segmentation to understand and analyze medical images, these methods including:

- **Clustering Algorithms:**As the aim of segmentation is separating the image into coherent interested regions, the clustering algorithm is applied for the segmentation task by extracting global characteristics of the considered image to partition the region of interests from the background [8]. One of the reputable clustering methods is K-Means Clustering that partitions data points into groups while K represents the number of groups. Although clustering algorithms in particular K-Means Clustering are efficient in medical image segmentation, it has some limitations. The drawback of this algorithm is, the prior idea is needed to determine K before applying the algorithm and it must be defined precisely,as the inappropriate value of K will cause unsatisfactory results [9].
- **Thresholding:**The simplest method for biomedical image segmentation is using the thresholding method. This strategy is based on classifying pixels to segment regions of interest. The task of this method is partitioning pixels into some levels to segment images using a defined Thresholding. Although this method can successfully be used for cancer detection, ultrasound images, brain MRIs [10], it also has some drawbacks. The downside of this method is, it is useful for high-contrast images and it does not have the precise result for low-contrast images [9].
- **Edge-Based Segmentation:**This method is one of the most commonly used techniques and is based on edge detection that partitions different regions of interest. The task is highlighting discontinues in grey level and color, as edges show boundaries between objects. Moreover, in this method it is critical to building the border of detected edges like an edge chain as weak or fake edges will be removed using thresholding. Although medical image segmentation based on edge detection is useful, it has some drawbacks. The first downside of this technique is, the desired result is highly affected by noise presence. Secondly, fake and weak edges have this probability to be in the detected edges and have a minus effect on the output result. Lastly, it requires to be used with the region-based technique for medical image segmentation tasks [11].
- **Model-Based Segmentation:**This method is based on the structure of the organ with a repetitive form of geometry and can be used as a probabilistic model for shape and geometry variation. Although this method is useful for analyzing biomedical images, it has some limitations. The first drawback is it requires manual iteration to building the initial model and choosing the right parameters. Secondly, standard deformable models can show weak convergence to concave boundaries of organs [11].

There are other traditional segmentation methods for processing biomedical images including region-based segmentation, region merging, split and merge, fuzzy-based methods, foreground extraction, active contour, and many other traditional methods which are not as accurate and useful as the U-net in particular for datasets with fewer images as they have mentioned limitations.

Recent advances in machine learning and deep learning have provided strategies for finding patterns in medical images. The great advantage of deep neural networks is obtaining feature representations for images directly because there is no need for extracting features manually like techniques that I mentioned in the traditional medical image segmentation methods. Moreover, deep neural networks in particular convolutional neural networks have shown good performance in different clinical applications in particular medical image segmentation [12]. The process of convolutional neural networks or CNNs for semantic segmentation is assigning one label to each pixel of the image that enables researchers to do predictions about the regions of interest. There are some kinds of convolutional neural networks (CNNs) that researchers have applied to medical datasets to obtain favorable results, these networks

including:

- **AlexNet:**AlexNet was proposed by Alex Krizhevsky and in collaboration with Ilya Sutskever and Geoffrey Hinton in 2012. This convolutional neural network consists of eight layers including five convolution layers with max-pooling and three fully connected layers. Images with the size of 227x227 are feed to the network as input and after passing five layers of convolutions and max-pooling they are sent to fully connected layers. One of the applications of AlexNet in medical image segmentation is the early detection of Alzheimer's disease or AD. AD detection is based on partitioning the hippocampus, cortical thickness, and brain volume in the brain CT scans [13]. Although AlexNet proposed to solve the image classification problem, Lei Cai et al [13] and his research team carried out related research in the AD diagnosis and proposed a deep neural network based on the AlexNet. Moreover, another variant of the AlexNet like 3D AlexNet is effective in automated segmentation of prostate tumors with satisfying results [14]. This convolutional neural network (CNN) can be fine-tuned to be used in polyp detection and it plays an important role in the understanding of colonoscopy videos [15]. Although AlexNet can be applied to image analysis segmentation tasks, it is not as efficient as U-Net as it can be modified to be a basis for some image segmentation tasks and it is not widely used like the U-Net.
- **VGG:**VGG was innovated after AlexNet with two versions including VGG-19 which is larger and slower with less accuracy, and a smaller and faster version called VGG-16. This model was created for image recognition tasks and can support up to 19 layers. VGG uses consecutive 3x3 filters that make it more effective than the AlexNet because usage of these several small kernels increases the depth of the network and as a result, more complex patterns can be extracted while the cost of the computation remains small [13]. Although VGG was innovated for image recognition tasks, it can be fine-tuned to be used in image segmentation purposes in ultrasound images [16]. It also can be used with UNet as it is lightweight which is why it can be concatenated to the UNet to make the deeper network for medical image segmentation tasks like automatic polyps detection [17]. It also can be used with SegNet for lung parenchyma segmentation tasks with satisfying results [18]. Moreover, it also can be combined with dilated convolution for lung segmentation of CT scan images with more accuracy and fewer computations cost [19]. Although VGG was not exclusively proposed for medical image segmentation tasks, it can be a backbone of other convolutional neural networks to obtain promising results of course it is not widely used as the UNet for medical diagnosis.
- **Inception:**Inception has revolutionized classification tasks using convolutional neural networks. This model has better performance than the previous convolutional neural networks and it also has better speed and accuracy. It also has some variants and the number of layers depends on the version of the model, for example, Inception-V3 has 48 layers. This model is not inclusively designed for medical image segmentation but some variants of this network can be integrated with the UNet to perform medical image segmentation tasks [20]. To clarify this point, this model can be modified to be combined with other convolutional neural networks to obtain better performance for brain tumor and heart segmentation tasks [21].

While traditional methods of image segmentation and previous convolutional neural networks have shown satisfying results in biomedical image segmentation, they are not exclusively designed and trained for the task of the medical applications and there have been some ideas for obtaining better results. The UNet was a breakthrough in biomedical image analysis as it can be used for a wide range of meical image segmentation tasks including cell, lung, polyps and tumor detection, and other medical applications. What makes UNet stand out from previous approaches is the architecture of this model that leads to better performance and more promising results even for datasets with fewer images.

III. U-Net Architecture

In this section, I describe the U-Net architecture including encoder, decoder, and skip connections parts based on the original U-Net paper which was presented by Olaf Ronneberger and his colleagues from the Computer Science Department and BIOSS Centre for Biological Signalling Studies of the University of Freiburg in Germany.

A. Motivation and Problem Statement

In general deep neural networks aim to classify an image, to clarify this point, an input image is fed to a convolutional neural network and a label is returned as the output, while in medical applications we need to understand the exact location of the disease and localize the area of the abnormality, hopefully, the U-Net is designed specifically to address this issue. To clarify this point, the U-Net does a classification task on each pixel [4], [22].

The U-Net was proposed to solve the image segmentation problems in the medical area using datasets with limited images. Before the invention of the U-Net, deep neural networks had to be trained on large datasets to obtain precise results, while the U-Net solved this issue using a simple but effective data augmentation technique. The original U-Net paper is based on the dataset of the ISBI challenge to partition neural structure in electron microscopic stacks using a tiff file image containing 30 images. Not only this network can be useful for small and large datasets, but also the segmentation process is done fast [4], [22].

The other challenge in medical image segmentation is deformations of real images including very low-contrast structures,

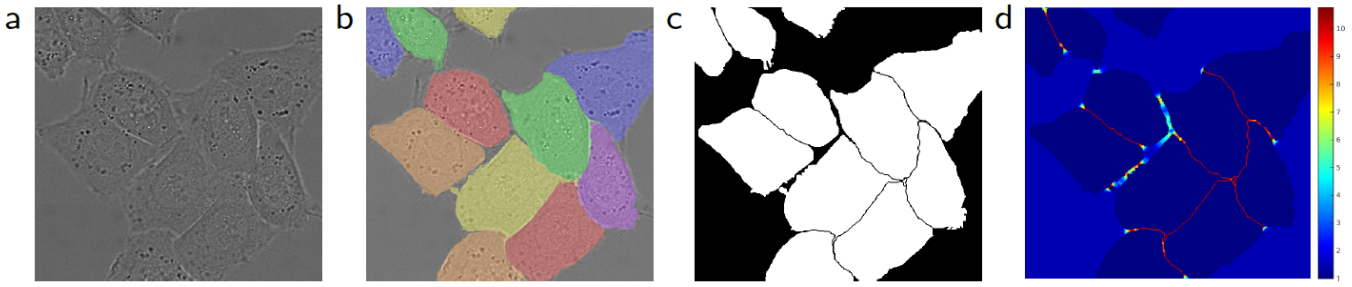

Fig. 1. HeLa cells recorded by DIC (differential interference contrast) microscopy. a) raw image b) Ground truth segmentation. Different colors show different examples of HeLa cells. c) Created segmentation mask (black and white) d) A map with a lost weight in pixels to allow the network to learn edge pixels.

fuzzy membranes, and other cell compartments that can be simulated and lead to negative impacts on the output result. The U-Net had solved this problem with a useful strategy called elastic deformation to prevent mentioned problems [4], [22].

One of the most common issues in medical image analysis is determining the frontiers when parts of the same class of the image stuck together. The solution to this issue is choosing values with large weights in loss function to separate background cells from touching cells (see Figure 1 which is taken from the original U-Net paper), [4], [22].

The U-Net does not have any fully connected layers or (FC), the architecture of this elegant network contains only convolution layers with max-pooling operations in the encoder part, and up-convolution along with convolution layers in the decoder part and concatenation path. This network has 23 layers and inventors of this deep neural network won the prize of the ISBI 2015 challenge [4], [22]

B. Contracting Path

The encoder part or contracting path is the left part of the U-Net which extracts features using two 3x3 convolutions followed by an activation function called rectified Linear Unit or ReLU. Each layer is followed by a 2x2 max-pooling operation with stride 2 and the number of feature channels is doubled until the decoder part. The number of feature channels at the first layer is 64, then it increases to 128, then it is doubled to 256, then it becomes 512 finally, it reaches 1024. The convolutions extract features and max-pooling reduces the size of the feature map to reduce the number of parameters. To clarify this point, the calculations in this part aim to decrease the complexity. Overall a pooling layer consists of a pixel that represents some pixels (see Figure 2 which shows max-pooling operation).

What is obvious is that this part of the network is similar to other convolutional neural networks (CNNs) (see Figure 3 which is taken from the original U-Net paper), [4], [22]. The equation related to 3x3 convolution with Rectified Linear Unit (ReLU) is like below:

$$b_{x,y,l} = ReLU(\sum_{\substack{i\in\{-1,0,1\}\\ j\in\{-1,0,1\}\\ k\in\{1,\dots,K\}}} w_{i,j,k,l}.a_{x+i,y+j,k} + c_l)$$

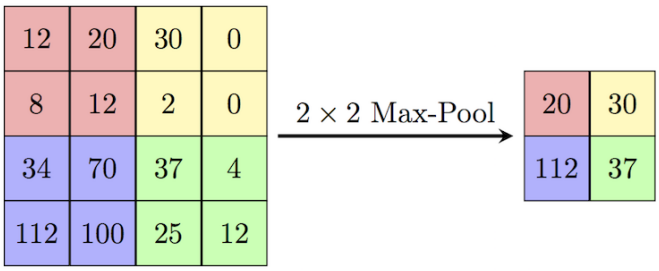

Fig. 2. Max-Pooling Operation

In convolution equation a is input feature map, b is output feature map, and w relates to weights. Moreover, if a is the input feature map and b is the output feature map the equation relates to max-pooling is like below:

$$b_{x,y,k} = \max_{\substack{i\in\{0,1\}\\ j\in\{0,1\}}} (a_{2x+i,2y+j,k})$$

C. Expanxive Path

The decoder part or expansive path is the right part of the network and this part makes the U-Net different from other convolutional neural networks, as other convolutional neural networks(CNNs) have fully connected layers(FC), while the U-Net does not have any fully connected layers and it only has 2x2 up-convolutions instead of them. This part aims to reconstruct the image from feature vectors and enable accurate localization. Each step of the decoder part has the transpose convolution which divides the number of feature channels. Besides 2x2 up-convolutions with the rectified linear unit (Relu) in each step, there are 3x3 convolutions with the Relu activation function. To clarify this point, the sampled output is combined with high-resolution features that came from the encoder part by connection path (see Figure 3 which is taken from the original U-Net paper), then convolutions in the decoder part have this task to generate better output result based on this information.The equation of the 2x2 up-convolution is like below, in this equation a is the input feature

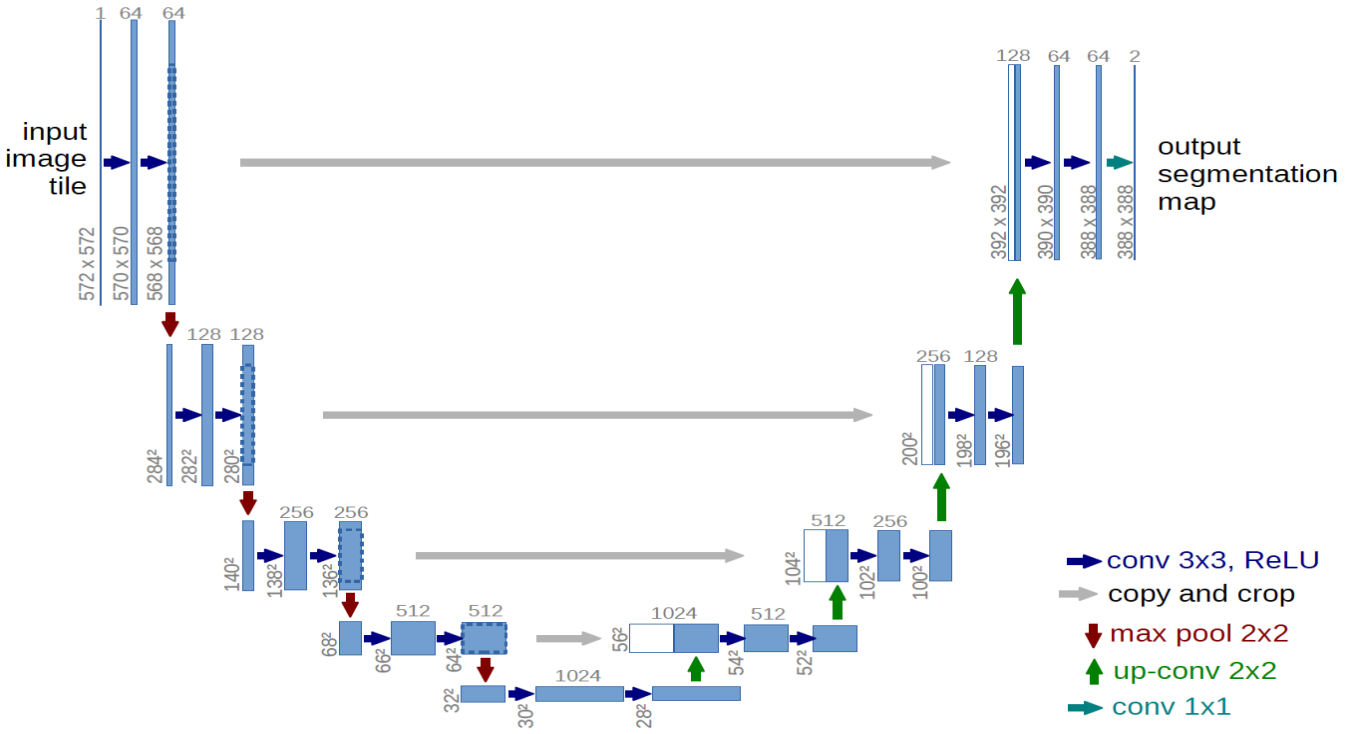

Fig. 3. U-net architecture (example for 32x32 pixels in the lowest resolution). Blue boxes show multi-channel feature maps. The number of channels is written on the top of each box. The x-y-size is denoted at the lower-left edge of each box. White boxes correspond to copied feature maps. The arrows represent the different operations.

map, b is the output feature map, w relates to the weights, and c is biased [4], [22].

$$b_{2x+i,2y+j,l} = ReLU(\sum_{\substack{i\in\{0,1\}\\ j\in\{0,1\}\\ k\in\{1,\dots,K\}}} w_{i,j,k,l}.a_{x,y,k} + c_l)$$

D. Connection Path

Skip connections are a critical part of the U-Net to concatenate extracted features from the encoder part to the decoder part. To clarify this point, high-resolution features from the result of the convolutions are combined with the output of the up-sampling operation. The connection or concatenation part is useful for reconstructing the image from the feature vectors by transmitting important features and information from the contracting path to the expansive path [4], [22].

E. Training

The input images and corresponded labels are trained as an end-to-end learning strategy. As convolutions do not have paddings, the output image will be smaller than the input image but in the implementation process using Google Colab, this issue can be solved by determining the output size of the image. For training the network using my datasets I used SGD and Adam optimizers based on my model. The momentum of the SGD optimizer is the highest (0.99) and the learning rate of this optimizer is 0.01 [4].

Based on the original U-Net paper the equation related to the softmax is:

$$P_k(x) = exp(a_k(x))/(\sum_{k'=1}^{K})exp(a_{k'}(x)))$$

*W*here k is feature channels and $a_k(x)$ is the activation of the feature channels and K is the number of classes and $p_k(x)$ is the approximated maximum function.

Cross-entropy is a measure that calculates the difference between two probability distributions and can be used as a loss function in deep neural networks. To clarify this point, Cross-entropy will compute a score that shows the average difference between the true and predicted probability distributions for all classes in the deep neural network. Based on my dataset and my image segmentation task I used the binary-crossentropy loss function for compiling my U-Net model [4]. Based on the original U-Net paper the equation related to the crossentropy is like below:

$$E = \sum_{i\in\Omega} w(x)log(p_{l(x)}(x))$$

In the above equation, l is the true label where l: $\Omega \rightarrow \{1,...,K\}$, and w is a weight map and highlights more important pixels in the training process where w: $\Omega \rightarrow \mathcal{R}$.

Ronneberger et al and his team pre-computed the weight map for each label segmentation to compensate frequency difference of pixels of the certain class in the training data

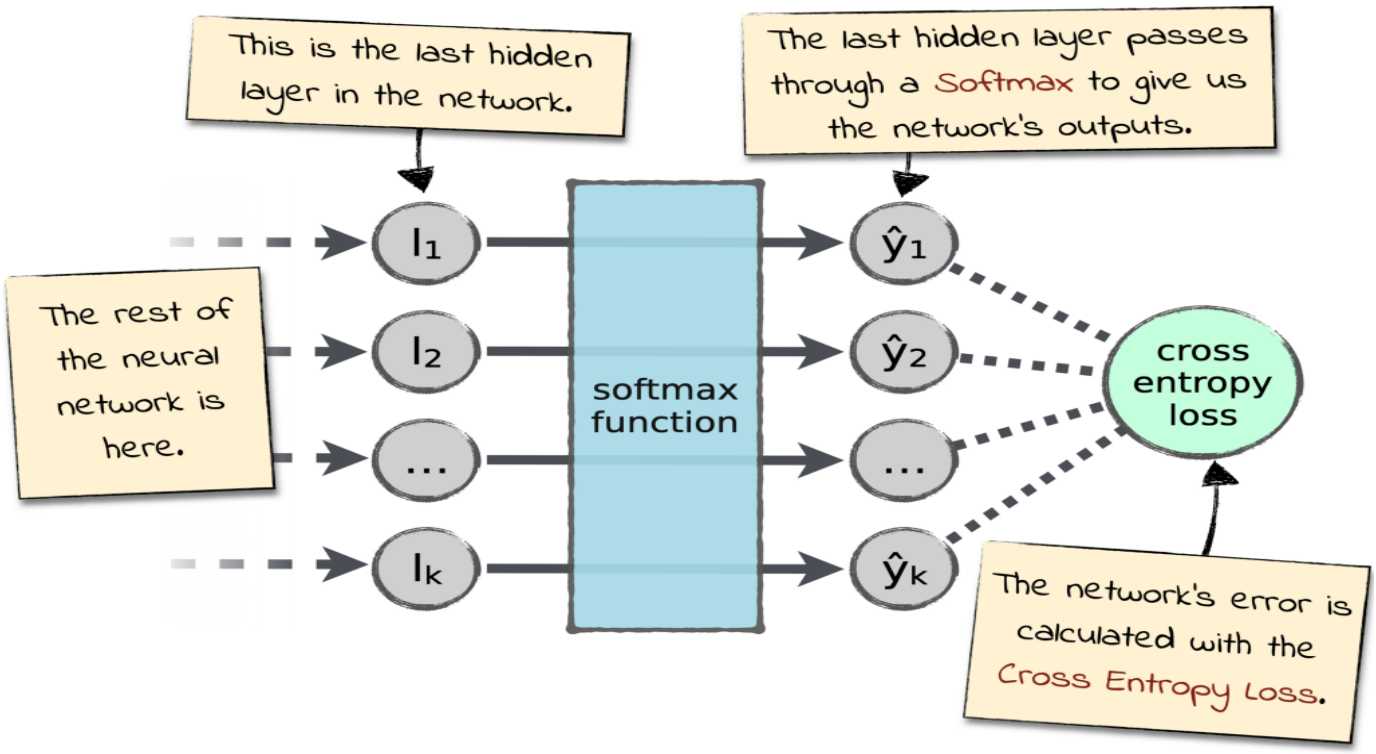

Fig. 4. The operation of the cross-entropy loss function.

and as a consequence, the U-Net was forced to learn a small separation of frontiers(see Figure 1 which is taken from the original U-Net paper) [4].

The partitioning of frontiers can be computed using morphological operations and the weight map equation can be computed like below [4]:

$$w(x) = w_c(x) + w_0.exp(-(d_1(x) + d_2(x))^2/2\sigma^2)$$

In the above equation w_c is the weight map and has this task to stabilize frequencies of the classes and is like below:

$w_c : \Omega \rightarrow \mathcal{R}$

In the mentioned equation d_1 and d_2 are distances of the borders to the nearest cell and the second nearest cell respectively and are like below [4]:

d_1 , $d_2 : \Omega \rightarrow \mathcal{R}$

In convolutional neural networks, it is critical to choose appropriate values for the weights as it has a straightforward impact on the performance of the network and the output result(see Figure 4 from LaptrinhX website, it is the best picture for showing the operation of the cross-entropy loss function, it describes the task of this function and easy to understand as it does not contain any equations and depicts this function most simply). It is noticeable that the weights of the original U-Net had initialized based on the Gaussian distribution [4].

F. Data Augmentation

When a dataset consists of few images, data augmentation is the best strategy to generate more data at a low cost and obtain better results. To clarify this point, data augmentation is the cheapest way to train more data, in particular in some cases that we have limited images. For the ISBI challenge, dataset shift and rotation are needed and I also used horizontal and vertical flip besides other parameters [4].

IV. Experiments

In this part, I did some experiments using two different datasets on two different styles of coding with different parameters to generate different results. The codes of experiments are in my Github.

A. Datasets

I analyzed the performance of my U-Net model using two different datasets including Lung Segmentation and ISBI-2015 challenge for the binary class segmentation task. The detailed descriptions of the mentioned datasets are like below:

- **Lung Segmentation:**This dataset is a collection of CT images with segmented lungs. It consists of 1021 images with the size of 256x256 and as the number of images for this dataset is quite enough I did not use the data augmentation technique for this dataset. I used CT images with corresponded masks for my experiment.
- **ISBI-2015 Challenge:**This dataset is the dataset of the original U-Net paper and consists of electron microscopy images. The original dataset is a tiff file with 30 images but I changed the tiff file to jpg for my experiment. This dataset has limited images from a serial section transmission electron microscopy or (ssTEM) of Drosophila

first instar larva ventral nerve cord or (VNC) [4]. As the number of images in this dataset was limited I used the data augmentation technique for obtaining better r results.

B. Baseline

I used the U-Net which is a convolutional neural network or (CNN) and was proposed in 2015 for medical image segmentation. The architecture of this elegant network consists of a contracting path or encoder part for learning features or feature extraction or derive context to generate feature vectors. It also has an expanding path or decoder part for reconstructing the image from feature vectors and accurate localization. Skip connection paths are another critical part of this network for transferring features from the encoder part to the decoder part.

The size of the input images in the original U-Net paper is 572x572 but I resized the size of the input images to reduce computations time as I did not have GPU and I had to execute my codes on Google Colab. The convolution blocks in the encoder part of the U-Net have 64,128,256,512 and 1024 filters but I changed the size of them in some experiments to decrease the number of computations and make a comparison with different results.

I used different activation functions and different optimizers and metrics to generate distinctive results. I also changed the number of filters of the convolutions in some experiments. The total parameters in my experiments vary from 1,940,817 to 31,054,145 based on the architecture of my model.

C. Implementation Details

All of my experiments are executed on Google Colab. As my device was macOS I did not have any available GPUs that is why I had to use Google Colab. It allows users to run python codes and it has some advantages including free access to GPU which accelerates the running process, requires zero configuration and it is easy to share as it allows users to share the public link of their codes or share it using Gmail addresses in private.

I implemented my experiments using the Keras learning tool which is a powerful deep learning API running on top of the Tensorflow and it allows fast experimentation. It is a highly productive interface for solving machine learning and deep learning problems. Convolutional neural networks (CNNs) can be implemented easily using Keras as it consists of convolution, max-pooling, transposed convolution, and many other necessary layers which data scientists need for developing their convolutional neural networks. It also supports data augmentation technique using ImageDataGenerator class in Keras deep learning library.

Notably, one of the drawbacks of the Google Colab is it has usage limitations and it does not have unlimited access which makes the running process a bit exhausting.

D. Evaluation Metrics

To evaluate the performance of my U-Net model I used different metrics including:

- **Jaccard:**Jaccard index or the Intersection over Union (IoU) is critical for calculating the percentage of the overlap between predicted and ground truth masks. To clarify this point, it measures the mutual pixels between ground truth and predicted masks divided by all pixels of both masks and it is like below:

$$Jaccard = IoU = \frac{|GroundTruth \cap Predicted|}{|GroundTruth \cup Predicted|}$$

- **Dice-Coefficient:**This is a useful loss function for image segmentation tasks that measures overlap between two samples. The range of this metric is between 0 and 1 and 1 shows the best and complete overlapping of two samples and it is like below:

$$Dice - Coefficient = \frac{2|GroundTruth \cap Predicted|}{|GroundTruth| + |Predicted|}$$

V. Results and Visualizing Predictions

I conducted 10 experiments in two datasets and the result of my experiments with visualized predictions is described below.

A. ISBI-2015 Challenge Dataset

If we consider training data and training label in figure 5 the predicted data in my experiments are like below(see Figure 5 which is my training data and training label and predicted result from left to right respectively) and (see Table 1 which shows details of the first experiment).

TABLE I
ISBI-2015 Challenge Dataset-Experiment 1

loss	iou	iou_thresh	val_loss	val_iou	val_iou_thresh
0.4493	0.6471	0.7786	0.5336	0.6448	0.7762

In experiment 2 I removed the last Batchnormalization as I wanted to make a comparison between outputs. If we consider the training data and training label of experiment 1 which is shown in figure 5 the predicted result will be like below(see Figure 6 for the predicted result and Table 2 which shows the final result after training all epochs).

TABLE II
ISBI-2015 Challenge Dataset-Experiment 2

loss	iou	iou_thresh	val_loss	val_iou	val_iou_thresh
0.4486	0.6465	0.7775	0.5325	0.6607	0.7762

In experiment 3 I changed the number of filters in the convolutions from 64 to 16 to reduce the number of computations. If we consider the training data and training label of experiment 1 which is shown in figure 5 the predicted result will be like below(see Figure 7 for the predicted result and Table 3 which shows the final result after training all epochs).

In experiment 4 I changed the number of filters in the convolutions from 64 to 16 and I resized the image from 512

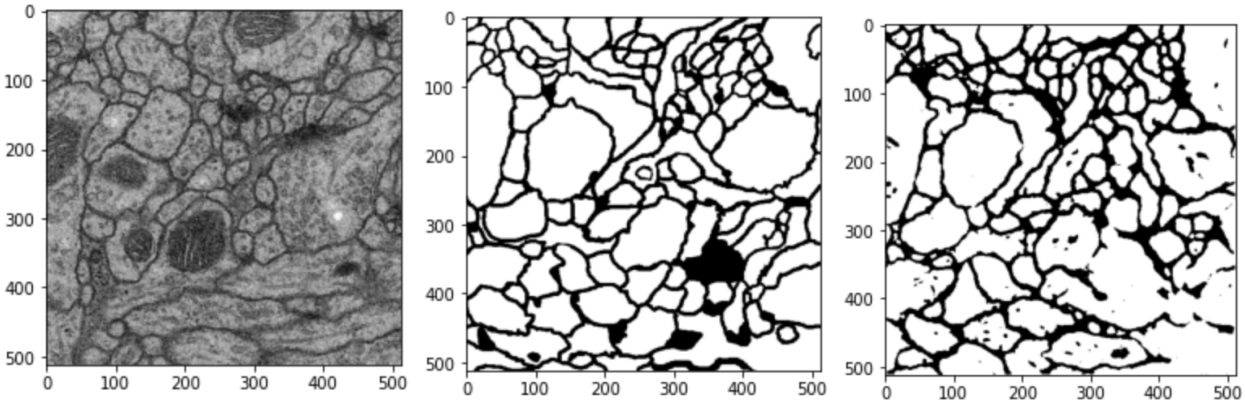

Fig. 5. Training Data is the left image, the training label is the center image, and the predicted result is the right image.

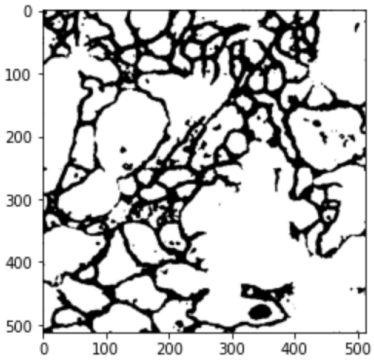

Fig. 6. Predicted result of experiment 2

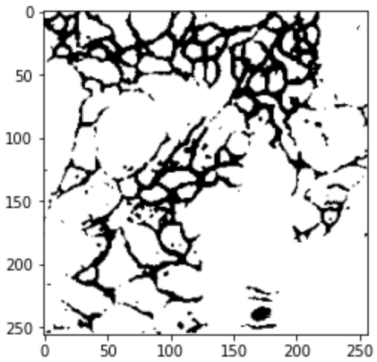

Fig. 7. Predicted result of experiment 3

TABLE III
ISBI-2015 CHALLENGE DATASET-EXPERIMENT 3

loss	iou	iou_thresh	val_loss	val_iou	val_iou_thresh
0.4597	0.6325	0.7782	0.5401	0.6725	0.7765

to 256 and I changed my metrics to dice-coefficient. If we consider the training data and training label of experiment 1 which is shown in figure 5 the predicted result will be like below(see Figure 8 for the predicted result and Table 4 which shows the final result after training all epochs).

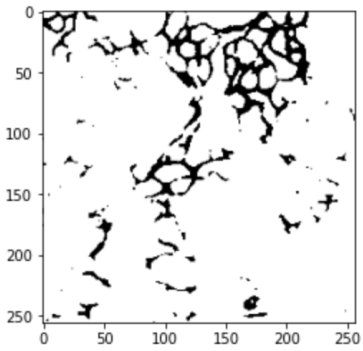

Fig. 8. Predicted result of experiment 4

TABLE IV
ISBI-2015 CHALLENGE DATASET-EXPERIMENT 4

loss	jaccard	dice_coef	val_loss	val_jaccard	val_dice_coef
0.4570	0.6385	0.7790	0.5461	0.6799	0.8094

In experiment 5 I used the accuracy metric. Although for image segmentation accuracy of the model is not an appropriate metric, I was curious to see the result. The result of the experiment with the code of this experiment is in my Github.

B. Lung Segmentation Dataset

If we consider test data and test label in figure 9 the predicted data in my experiments are like below(see Figure 9 which is my test data and test label and predicted result from left to right respectively) and (see Table 5 which shows details of the first experiment of Lung Segmentation dataset).

The first experiment is done on 256x256 images based on IoU and IoU_Thresholded metrics with the first convolution filter size of 16 and ReLU activation function and Adam optimizer.

In experiments 7 and 8 I used 256x256 images with IoU metrics and ReLU activation functions and the only difference is, in experiment 7 I used a filter size of 64 in the convolutions

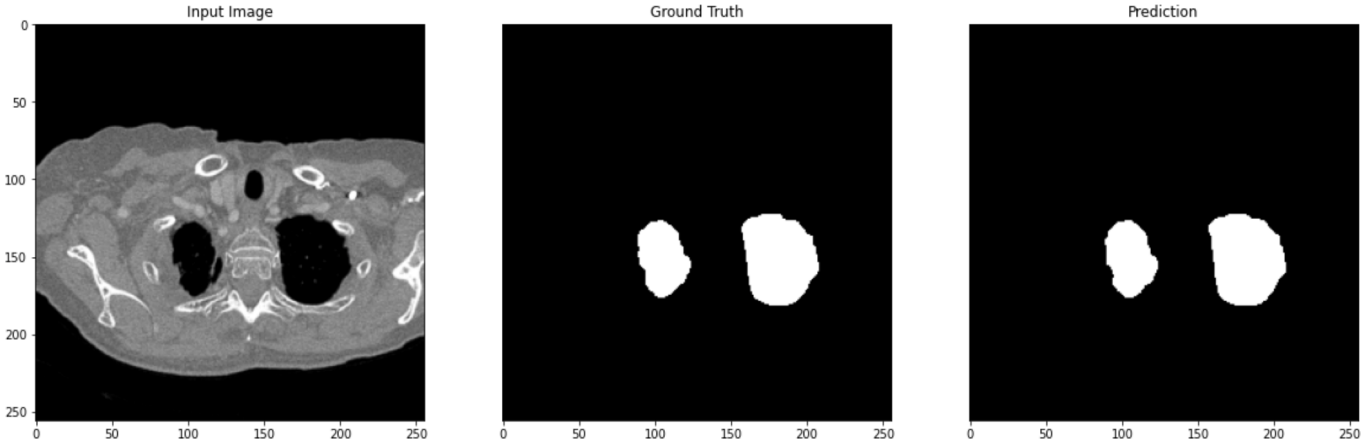

Fig. 9. Test Data is the left image, the Test label is the center image, and the predicted result is the right image

TABLE V
LUNG SEGMENTATION DATASET-EXPERIMENT 6

loss	iou	iou_thresh	val_loss	val_iou	val_iou_thresh
0.0093	0.9673	0.9791	0.0083	0.9680	0.9817

and Adam optimizer while in experiment 8 I used the filter size of 16 and SGD optimizer. I should mention that the predicted result of these experiments is similar to experiment 6 that is why I do not show the predicted result of these experiments here but it is available in my Github(See Table 6 and Table 7 for the result of experiment 7 and 8).

TABLE VI
LUNG SEGMENTATION DATASET-EXPERIMENT 7

loss	iou	iou_thresh	val_loss	val_iou	val_iou_thresh
0.0097	0.9680	0.9798	0.0093	0.9665	0.9802

TABLE VII
LUNG SEGMENTATION DATASET-EXPERIMENT 8

loss	iou	iou_thresh	val_loss	val_iou	val_iou_thresh
0.0109	0.9626	0.9758	0.0098	0.9660	0.9783

In experiment 9 I used 256x256 images with accuracy metric and ReLU activation functions and SGD optimizer. The predicted image is a bit different from other experiments(see Figure 10 which shows the predicted result of experiment 9).

In experiment 10 I used 256x256 images with Jaccard and dice_coefficient metrics and ReLU activation function and Adam optimizer. The predicted image is the same as experiments 6, 7, and 8 that is why I do not show it here but the Code is available in my Github(see Table 8 which shows the result of Experiment 10).

VI. CONCLUSION

My results show that the U-Net is useful for medical image segmentation and it has promising and precise results. In some cases like the ISBI-2015 challenge dataset, we need a data augmentation technique to obtain precise results but in some cases like the Lung Segmentation dataset, a data augmentation strategy is not necessary.

Fig. 10. Predicted result of the Experiment 9

TABLE VIII
LUNG SEGMENTATION DATASET-EXPERIMENT 10

loss	jaccard	dice_coef	val_loss	val_jaccard	val_dice_coef
0.0067	0.9760	0.9879	0.0062	0.9767	0.9882

The U-Net has high performance for both large and few datasets and it does segmentation tasks at high speed. Based on my observations and experiments it is effective enough to be trained on the Google Colab and I did not need high-speed and powerful GPUs. In my point of view, U-Net is reliable for image segmentation tasks in particular medical image segmentation applications.

I should mention that this convolutional neural network has some variants which were designed later and the performance of this network had improved using other convolutions, choosing different filters or optimizers and activation functions. I

should mention that my aim was not obtaining the best result I just aimed to implement the original U-Net paper and apply some changes to observe changes.

VII. Experiment Codes

All source codes of experiments are available in my Github.

References

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

[2] Saleh F., Aliakbarian M.S., Salzmann M., Petersson L., Gould S., Alvarez J.M., "Built-in Foreground/Background Prior for Weakly-Supervised Semantic Segmentation," *in: Leibe B., Matas J., Sebe N., Welling M. (eds) Computer Vision* ,ECCV 2016.

[3] Baris Kayalibay and Grady Jensen and Patrick van der Smagt, "CNN-based Segmentation of Medical Imaging Data,"*in: CoRR*, abs/1701.03056, 2017.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in Proc. *International Conference on Medical image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.

[5] LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D, "Backpropagation applied to handwritten zip code recognition".in Proc.*Neural Computation*, 1(4), 541-551 (1989).

[6] Simonyan, K., Zisserman, A, "Very deep convolutional networks for large-scale image recognition".*arXiv:1409.1556*, 2014.

[7] Ciresan, D.C., Gambardella, L.M., Giusti, A., Schmidhuber, J, "Deep neural net- works segment neuronal membranes in electron microscopy images".*NIPS*, pp. 2852-2860 (2012).

[8] Yanhui Guo and Amira S. Ashour, "Neutrosophic sets in dermoscopic medical image segmentation",(2019).

[9] Dr. A. Ben Hamza, "Image Processing Course, Lecture: Image Segmentation",(2021).

[10] Ismail Yaqub Maolood, Yahya Eneid Abdulridha Al-Salhi, Songfeng Lu, "Thresholding for Medical Image Segmentation for Cancer using Fuzzy Entropy with Level Set Algorithm".*Open Med (Wars)*, 13: 374–383, 2018.

[11] Neeraj Sharma and Lalit M. Aggarwal, "Automated medical image segmentation techniques".*J Med Phys*, 2010.

[12] Intisar Rizwan I Haque, Jeremiah Neubert, "Deep learning approaches to biomedical image segmentation".*Informatics in Medicine Unlocked*,Volume 18, 100297, 2020.

[13] Lei Cai, Jingyang Gao, Di Zhao, "A review of the application of deep learning in medical image classification and segmentation".*Ann Transl Med*, 2020.

[14] Jun Chen, Zhechao Wan, Jiacheng Zhang, Wenhua Li, Yanbing Chen, Yuebing Li, Yue Duan, "Medical image segmentation and reconstruction of prostate tumor based on 3D AlexNet".*Computer Methods and Programs in Biomedicine*,Volume 200, March 2021.

[15] Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, R. Todd Hurst, Christopher B. Kendall, Michael B. Gotway, and Jianming Liang, "Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning?".*arXiv:1706.00712v1*, 2017.

[16] Mohammad Hesam Hesamian, Wenjing Jia, Xiangjian He, Paul Kennedy, "Deep Learning Techniques for Medical Image Segmentation: Achievements and Challenges".*Journal of Digital Imaging volume*, 32, pages582–596(2019).

[17] Debesh Jha, Michael A. Riegler, Dag Johansen,pal Halvorse, Havard D.Johanse, "DoubleU-Net: A Deep Convolutional Neural Network for Medical Image Segmentation".*arXiv:2006.04868v2*, 2020.

[18] Abdulhadi Omar, "Lung CT Parenchyma Segmentation using VGG-16 based SegNet Model".*International Journal of Computer Applications*, Volume 178 – No. 44, August 2019.

[19] Lei Geng, Siqi Zhang, Jun Tong, Zhitao Xiao, "Lung segmentation method with dilated convolution based on VGG-16 network".*Computer Assisted Surgery*, Volume 24, 2019.

[20] Zhang Ziang, Wu Chengdong, Sonya Colemanb, Dermot Kerrb, "DENSE-INception U-net for medical image segmentation".*Computer Methods and Programs in Biomedicine*, Volume 192, August 2020.

[21] Surayya Ado Bala1, Shri Kant, "Dense Dilated Inception Network for Medical Image Segmentation".*(IJACSA) International Journal of Advanced Computer Science and Applications*, Vol. 11, No. 11, 2020.

[22] Ayyüce Kızrak, "Deep Learning for Image Segmentation: U-Net Architecture".*Medium Website*, Sep 6, 2019.