Research Article

U-Net: Convolutional Network for Segmentation with DIC-C2DH-HeLa Dataset

Doi: 10.30508/KDIP.2022.338290.1032

Uranus Kazemi (Corresponding Author)*[1] | Mohammad Hossein Shakoor[2]

Abstract

Image segmentation is a basic issue in machine vision. One of the important tasks of machine vision and image processing is to recognize the pattern and one of the most important algorithms is U-Net segmentation. The U-Net algorithm has been identified as a popular algorithm in recent years due to its accurate response, high accuracy, high processing speed and learning, no need for large data sets for learning and no need for complex and expensive hardware. Image components and their fragmentation have become part of medical image processing. In this paper, we explain the U-Net algorithm and its convolutional network, as well as the most appropriate setting for the parameters and super parameters of this algorithm to optimize and achieve maximum accuracy in solving image-processing problems with this algorithm. In other words, a proposed method for segmenting medical images is performed on the DIC-C2DH-HeLa data set, which based on an architecture, the so-called "fully convolutional network", we have modified and expanded this architecture in such a way that with educational images Work very little and provide more segments that are detailed. The results showed that the proposed method has a higher accuracy than the other proposed method.

Keywords: U-Net Algorithm, deep learning, medical image processing, segmentation.

1- PhD Student in Software, Department of Computer Engineering, Arak University, Arak, Iran
2- Department of Computer Engineering, Faculty of Engineering, Arak University, Arak, Iran
*uranuskazemi@yahoo.com

-1 Introduction

In the past few years, deep convolution networks have surpassed many visual recognition tasks (Girshick, Donahue, Darrell, & Malik, 2014., Krizhevsky, A., Sutskever, & Hinton, 2017). While convolutional networks have been around for a long time (LeCun, Boser, Denker, Henderson, Howard, Hubbard, & Jackel, 1989), their success was limited by the size of the existing training suites and the size of the intended networks. For example, one of the improvements was the supervised training of a large network with eight layers and millions of parameters in the ImageNet dataset with 1 million training images. Since then, even larger and deeper networks have been trained (Simonyan, & Zisserman, 2014).

Typical use of convolutional networks is more for classification, where the output of an image is a single class tag. However, in many visual tasks, especially in biomedical image processing, the output must include localization; for example, a class tag is to be assigned to each pixel. In addition, thousands of educational images are often out of reach in biomedical work.

Identifying objects and positioning them in the image is an important issue in industry, medicine, military and security. Since the advent of artificial intelligence in the twentieth century, the identification of objects within the image has been an important issue and many artificial intelligence algorithms have been designed based on mathematical models in this regard.

After the advent of machine learning and the spread of image processing tools in the first decade of the present century, many supervised and unsupervised algorithms have been designed in this area that are generally better accurate than older mathematical models (Ciresan, Giusti, Gambardella, & Schmidhuber, 2012).

After the advent of deep learning in 2102, attempts to identify images based on deep learning models accelerated and achieved remarkable results. Identification accuracy, which with the best machine learning algorithms eventually reached 70%, increased to more than 90% with the new models and was used in new applications such as self-driving cars. Today, machines are getting closer and closer to human cognition, and in a sense, they have become smart.

One of the most widely used cases is the identification of objects in medical knowledge. Identifying objects, classifying them, and positioning them in the image, called fragmentation, can greatly reduce the costs and human error of this sensitive area. Fragmentation is one of the most important issues in image processing, image recognition and separation into its components, which determines the final success or failure of image analysis methods. However, there is no general method for successful fragmentation of all images in areas such as computer vision and Image processing has been done and still has a good research field due to its wide application. The accuracy of this study in areas such as remote sensing medicine and image retrieval is very important that helps to preserve and protect human life. Provides extensive fragmentation of images and application of methods in various fields.

Image segmentation includes four types, which are:

Image category: In this method, the objects in the image are only described and their position is not identified. For example, in Fig 1-a, there are bottles, cubes and glasses.

Identify objects: In this method, in addition to describing the existing objects, their location in the image is determined by an enclosed rectangle. Fig 1-b.

Semantic segmentation: In this method, in addition to identifying the objects in the image, all pixels of objects that are in the same category are painted or covered with a specific colour. As shown in Fig 1-c, this method is not able to distinguish between two objects of a single-layer. Sample segmentation: In this method, in addition to semantically identifying the objects in the image, each object is identified separately, and the algorithm distinguishes between all the objects in the image, even objects that belong to the same class and type (Rosebrock, 2017). This example is shown in Fig 1-d.

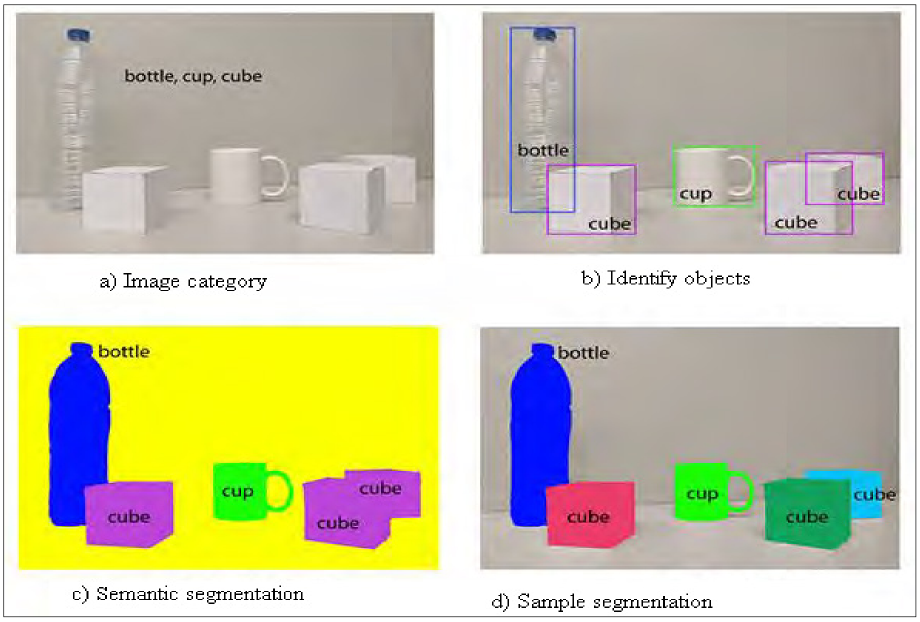

Fig. 1. Types of image segmentation (Rosebrock, 2017)

In this paper, we build a so-called "fully convolutional network" (Long, Shelhamer, & Darrell, 2015) based on a better architecture. We are modifying and extending this architecture to work with very few instructional images and to provide segmentation that is more detailed.
First, in this article, we talk about this algorithm, and then the proposed method and the degree of accuracy in it are examined.

2. U-Net Algorithm

The U-Net algorithm was developed in 2015 at the University of Freiburg, Germany by Olaf René Berger, Philipp Fischer and Thomas Brooks to solve this problem and increase processing speed and accuracy, using all-convolution networks (Ronneberger, Fischer, & Brox, 2015). This network has no dense or fully connected layers. This reduces the need for training data and leads to very accurate segmentation of image components. The main idea behind this algorithm is to create a sequential contraction path where pooling layers are replaced by sampling layers. After summarizing and collecting the desired features, the work is assigned to a sequential expansion path, which begins to reconstruct the original image and disseminate background information into it. As shown in Fig 2, the U-Net network consists of two paths or sub-networks of contraction and expansion, which are perfectly symmetrical and similar to each other, which have the same number of floors. Whatever is removed in the path of contraction is restored in the path of expansion.
There are various deep learning segmentation methods like Semantic Segmentation and Instance Segmentation, each of which has leading models. In this phase of the study, we decided for the U-net, which has attracted many attentions in the last few years and uses fully convolutional networks to perform the task of Semantic segmentation.

Fig 2, the architecture of the U-Net looks like a 'U', which acknowledges its name.

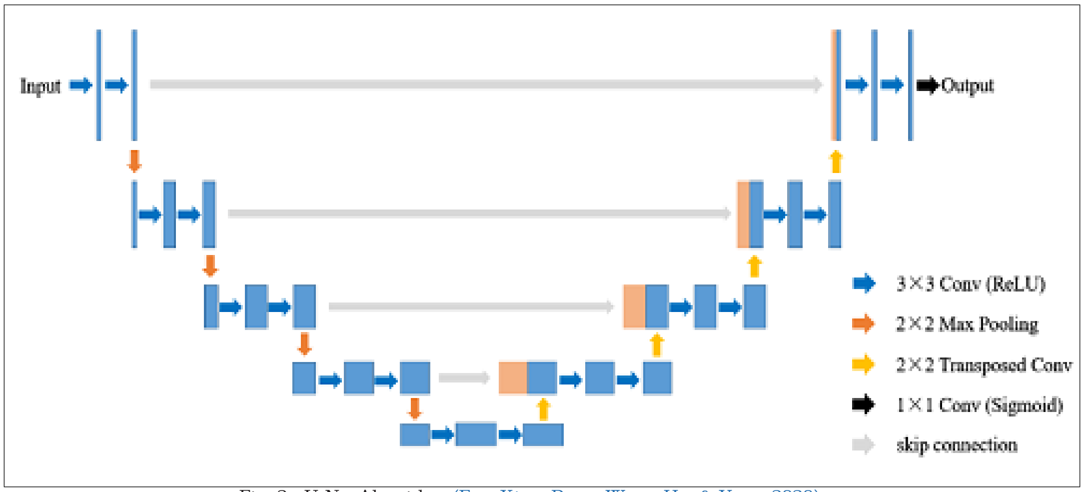

Fig. 2. U-Net Algorithm (Fan, Xing, Dong, Wang, Hu, & Yang, 2020)

2.1. Network Architecture

As shown in Fig 2, the U-Net network consists of two contraction and expansion paths, similar to the general form of convolutional networks, and consists of a convolutional layer that is responsible for the repeated application of 3 * 3 convolution filtering on images. After each layer of convolution, there is a layer of *Relu* or modified linear activator and then a layer of 2 x 2 max-pooling with step 2. Together, these three layers form a reducing sample layer. In each reducing class, feature channels are doubled. Conversely, in the expansion path, each layer receives an incremental diagram with a 2x2-convolution layer that is responsible for reducing feature channels and receiving the data to be added to the image from its peer in the contraction path. Then there is a 3 * 3 convolutional layer with a *Relu* layer.

In the last layer, a 1 * 1 convolutional layer maps each of the 64 feature vectors to a desired class. In total, the network has 23 layers of convection (Hariharan, Arbeláez, Girshick, & Malik, 2015., Kim, & Rhee, 2018., Seyedhosseini, Sajjadi, & Tasdizen, 2013).

2.2. Network Training

The network input contains the desired images and their fragmented layout in separate files. The network optimization algorithm is a random reduction gradient, and the amount of motion or momentum applied to the network should be a maximum of about 0.99, which allows the network to make the most of the images it has already seen in the updates. Grid energy function by a soft-max function Calculate at the pixel level applied to the output map in combination with the cross-entropy cost function.

One of the most important points in a deep cannulation network with many layers is the appropriate initialization to the network. Otherwise, parts of the network are over-activated and other parts are not involved in the calculations. Theoretically, the appropriate initial value for each neuron in the network is the value by which each property in the network has approximately a single variance. To calculate the initial value of each neuron, we use Formula 1, where N is the number of neuron inputs. For a 3 * 3 convolutional grid with 64 inputs that are the properties of the previous layer, the value of N will be according to Formula 2.

$$\text{Initial Weight}=\sqrt{(2/N)} \quad (1)$$

$$N=3\times3\times64=576 \quad (2)$$

2.3. Auxiliary data

When a small amount of training data is available, data augmentation can be a tool to teach the network the desired amount of flexibility or immutability.

Making changes such as rotation, colour change, shifting, adding noise, blurring, etc. on the image and especially changing the random form can be very instructive for the network.

To compensate for the lack of educational data, the U-Net algorithm also uses various data augmentation methods. In addition to providing raw educational data for the network, this has another benefit: body tissue reshaping is the most common type of change in medicine and is a very important problem.

The effective form is simulated and the resulting network can be applied to a variety of medical issues (Kim, & Rhee, 2018., Maddison, Huang, Sutskever, & Silver, 2014).

3. Proposed method

U-Net consists of two parts, which are the shrink section which is used to capture the background in the image and the “What” (meaning) and reduction “Where” (space). Expansion section that allows accurate localization. Fig 3 shows the structure of the proposed method then the proposed method is explained:

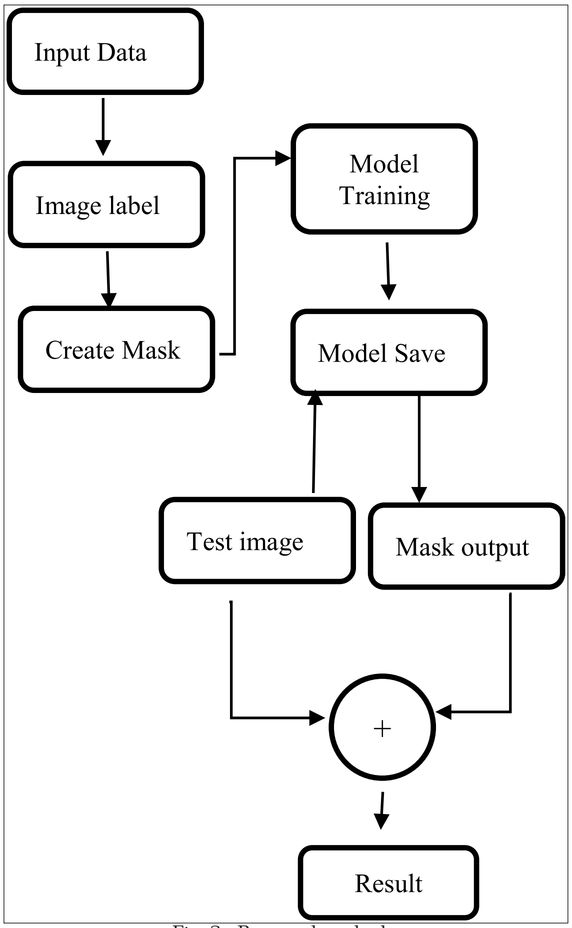

Fig. 3. Proposed method

The proposed method consists of three main steps:

A) Image input: In this section, the image is entered first, then the data is labelled, then masks are applied to the images. In other words, at this stage, the data is divided into two parts, test and train, and labelled.

B) Model training: In this part, the model is taught using the U-net algorithm (Malik, Robertson, Braun, & Greig, 2021) shown in Fig 2 that the network is based on the fully convolutional network and its architecture was modified and extended to work with fewer training images and to yield more precise segmentations, then the model is stored.

C) In this step, the image that is considered as a test is checked with the saved training model and the final model is proposed and the output is displayed.

The main idea in the proposed method is to complete a typical contract network with successive layers in which sampling operators replace the integration operators.

Hence, these layers increase the output quality. In order to localize, the high-resolution features of the shrinkage path are combined with the sample output. A sequential convolution layer can then learn to collect more accurate output based on this information.

As well, the major changes in our architecture is that we have a large number of feature channels in the sampling section that allow the network to propagate background information to higher resolution layers. As a result, the expansion path is more or less symmetrical with the contraction path, creating a U-shaped architecture.

To learn the network, input images and their related segmentation maps are used by implementing a random descending gradient (Jia, Shelhamer, Donahue, Karayev, Long, Girshick, & Darrell, 2014).

Due to the complexity, the output image with a fixed margin width is smaller than the input. To minimize overhead and maximize GPU memory usage, we prefer large input tiles to large batches, and therefore reduce the batches to a single image. Accordingly, we use a high momentum (0.99) so that a large number of previously seen instances determine the update at the current optimization stage.

In Algorithm 1, the pseudocode related to the proposed method is observed:

Algorithm 1: pseudocode of proposed method

```
1: Input: load input data
// 2D HeLa dataset
2: Pre-process the data
// image label and create mask
3: Split the data: train- test split
4: Define mode ()
//initialize the model with specified parameters
// pre-trained weights
    Model. Add (layers (Conv layers, Pooling layers, activation function)
    Model. Fit (X_train, Y_train)
    Model. Save ()
5: Training _data
//Initialise the hyper parameters (optimizer, Batch size, epochs, loss function)
6: Testing data
    Model. Evaluate ()
```

//evaluation of model

3.1. Dataset

The database used in this paper, 2D HeLa is a dataset of fluorescence microscopy images of HeLa cells stained with various organelle-specific fluorescent dyes. The images include 10 organelles, which are DNA (Nuclei), ER (Endoplasmic reticulum), Giantin, (cis/medial Golgi), GPP130 (cis Golgi), Lamp2 (Lysosomes), Mitochondria, Nucleolin (Nucleoli), Actin, TfR (Endosomes), Tubulin. The purpose of the dataset is to train a computer program to automatically identify sub-cellular organelles.

The training data is a set of 30 images (512x512 pixels) of the Transitional Section Abdominal Neural Cord Serial Transmission Electron Microscope (VNC) larvae. Each image is presented with a true segmentation map of cells (white) and membranes (black). The test suite is also publicly available (Fazeli, Roy, Follain, Laine, von Chamier, Hänninen, & Jacquemet, 2020., Hartmann, Müller, Soto-Rey, & Kramer, 2021., Pena, Fernandez, Tarr, Ren, Meyerowitz, & Cunha, 2020., Simonyan, & Zisserman, 2014)

3.2. Loss function and Evaluation Metrics

There are many loss functions to use for semantic segmentation problems but the most useful is Binary Cross Entropy. Cross entropy is better suited for a classification problem, and it will provide better results with each class.

Binary cross-entropy is intended to use with binary classification where the target value is 0 or 1. It will calculate a difference between the actual and predicted probability distributions for predicting class 1. The score is minimized and a perfect value is 0.

$$logloss = -\frac{1}{N}\Sigma_{i}^{N}\Sigma_{j}^{M} y_{ij} log(p_{ij}) \qquad (3)$$

Where N is number of rows and M is number of classes [16][20].

3.3. Implementation

The Proposed method is implemented with Keras functional API [17], which makes it extremely easy to experiment with different interesting architectures. Keras is a minimalist, highly modular neural networks library, written in Python and capable of running on top of either Tensor Flow. It was developed with a focus on enabling fast experimentation.

Output from the network is a 512*512, which represents mask that should be learned. Sigmoid activation function makes sure that mask pixels are in [0, 1] range.

3.4. Test the Proposed method

After training and fine-tuning the U-Net model, we got a loss of 0.3327, accuracy of 0.85. As shown in n Table I, the proposed method have been compared with the IMCB-SG (2014), KTH-SE (2014) and U-net (2015) (2021) methods in terms of accuracy (Badr, 2021., Chang, & Liao, 2019., Drozdzal, Chartrand, Vorontsov, Shakeri, Di Jorio, Tang, & Kadoury, 2018., Ronneberger, Fischer, & Brox, 2015., Malik, Robertson, Braun, & Greig, 2021).

Table I. Test the Model (epoch=100).

Method	Dataset	accuracy
IMCB-SG(2014)[18]	DIC-Hela	0.2935
KTH-SE(2014)[18]	DIC-Hela	0.4607
U-net(2015)[18][13]	DIC-Hela	0.7756
Proposed method	DIC-Hela	0.85

As can be seen in table I, the proposed method with selected datasets has reached a higher accuracy than other methods.

In table II, the proposed method is examined with different epochs, as it can be seen that in 100 epochs, a higher accuracy value is reached.

Table II. Proposed method with Different epochs.

Epoch	Dataset	accuracy
2	DIC-Hela	0.7733
5	DIC-Hela	0.801
10	DIC-Hela	0.81
20	DIC-Hela	0.814
100	DIC-Hela	0.855

In Fig 4, from part *a* to *i* there are different images of the data and the proposed method with 100 epochs. The images on the left are the main images and the images on the right are the predicted image using the proposed method.

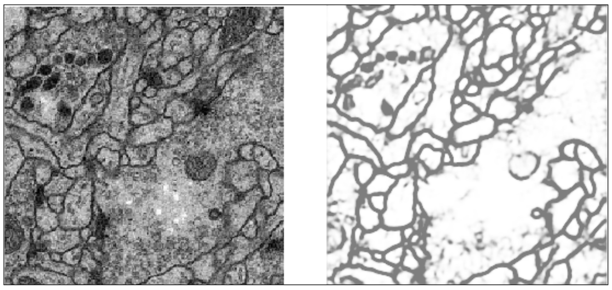

Fig 4. a. Original image and prediction image

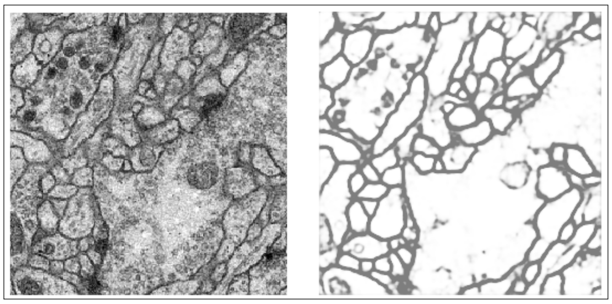

Fig 4.b. Original image and prediction image

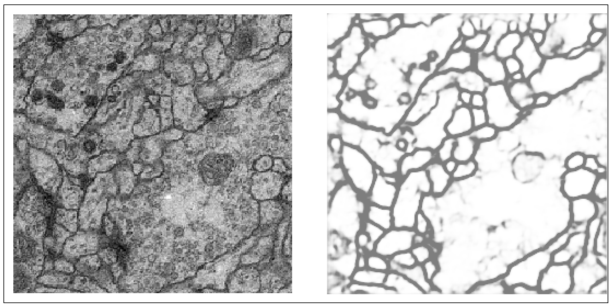

Fig 4.c. Original image and prediction image

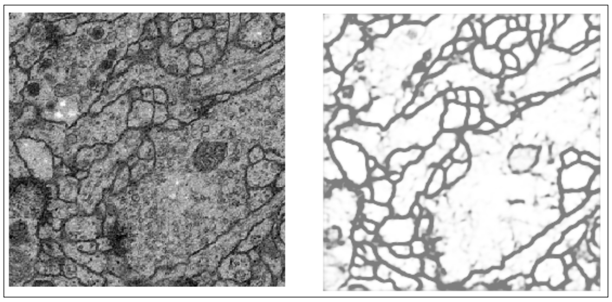

Fig 4. d. Original image and prediction image

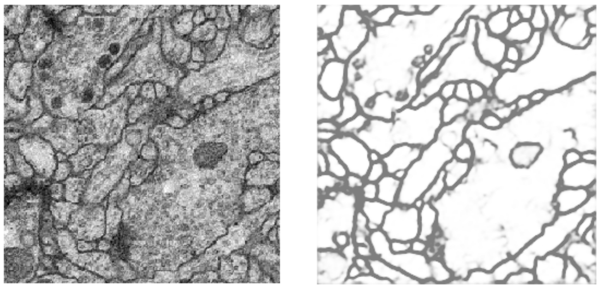

Fig 4. e. Original image and prediction image

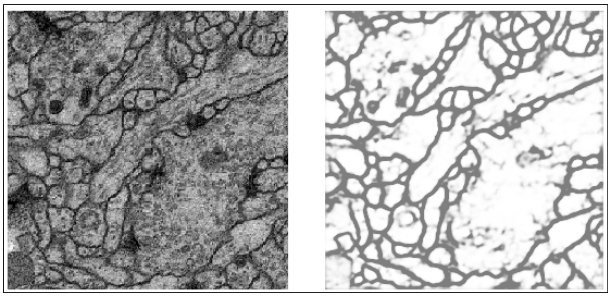

Fig 4. f. Original image and prediction image

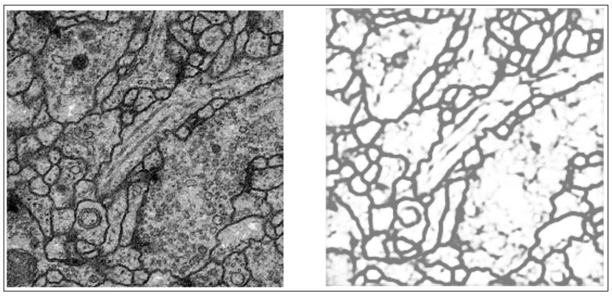

Fig 4. g. Original image and prediction image

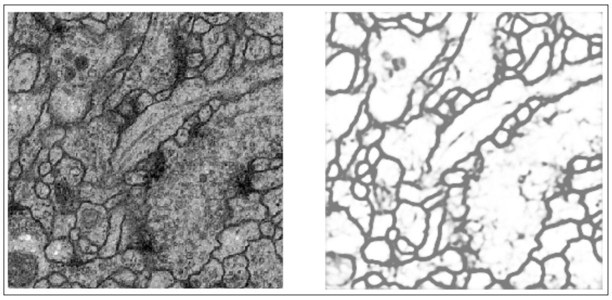

Fig 4. h. Original image and prediction image

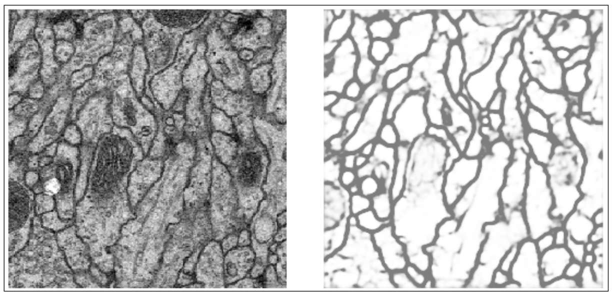

Fig 4. i. Original image and prediction image

In Fig 5, the proposed method is seen with two epochs. The proposed method with 2 epochs has reached an accuracy value of 0.7733.

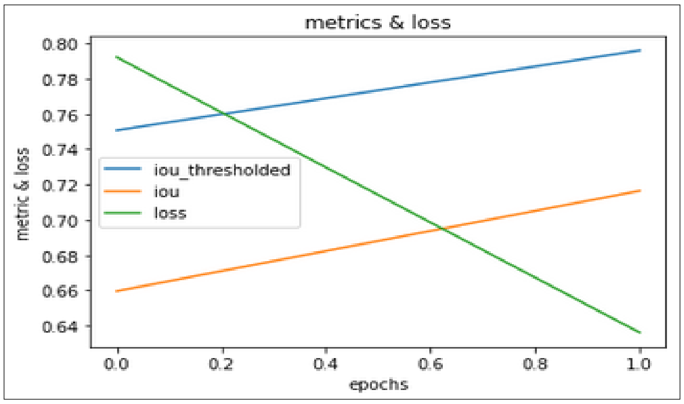

Fig 5.The proposed method with twoepochs

4. Conclusion

The deep learning model and the machine-learning model, strongly depend on the quality and quantity of educational data. In a sense, the more data we enter into the model, the better our performance will be.

U-Net inspires many new architectures. However, there is much to discover. There are many types of this architecture in the industry and therefore it is necessary to understand the first case to better understand them.

Therefore, in the proposed method, we were able to allow the network to propagate background information to higher resolution layers with a large number of feature channels in the sampling section. As a result, the expansion path is more or less symmetrical with the contraction path, creating a U-shaped architecture, and thus achieve higher accuracy than other methods.

U-Net modeling requires more extensive parameters in terms of tile size determination, over-reduction reduction methods, and more computational times. In the future, methods for pre-fitting U-Net models could be proposed.

References

1-Badr, E. (2021). Images in Space and Time: Real Big Data in Healthcare. ACM Comput*ing Surveys (CSUR)*, *54*(6), 1-38.

2-Chang, S. W., & Liao, S. W. (2019, October). KUnet: microscopy image segmentation with deep unet based convolutional networks. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)* (pp. 3561-3566). IEEE.

3-Ciresan, D., Giusti, A., Gambardella, L., & Schmidhuber, J. (2012). Deep neural networks segment neuronal membranes in electron microscopy images. *Advances in neural information processing systems*, *25*.

4-Drozdzal, M., Chartrand, G., Vorontsov, E., Shakeri, M., Di Jorio, L., Tang, A., ... & Kadoury, S. (2018). Learning normalized inputs for iterative estimation in medical image segmentation. *Medical image analysis*, *44*, 1-13.

5-Fan, J., Xing, L., Dong, P., Wang, J., Hu, W., & Yang, Y. (2020). Data-driven dose calculation algorithm

based on deep U-Net. *Physics in Medicine & Biology*, *65*(24), 245035.
6-Fazeli, E., Roy, N. H., Follain, G., Laine, R. F., von Chamier, L., Hänninen, P. E., ... & Jacquemet, G. (2020). Automated cell tracking using StarDist and TrackMate. *F1000Research*, *9*.
7-Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580-587).
8-Hariharan, B., Arbeláez, P., Girshick, R., & Malik, J. (2015). Hypercolumns for object segmentation and fine-grained localization. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 447-456).
9-Hartmann, D., Müller, D., Soto-Rey, I., & Kramer, F. (2021). Assessing the Role of Random Forests in Medical Image Segmentation. *arXiv preprint arXiv:2103.16492*.
10-Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., ... & Darrell, T. (2014, November). Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia* (pp. 675-678).
11-Kim, J. W., & Rhee, P. K. (2018). Image recognition based on adaptive deep learning. *The Journal of The Institute of Internet, Broadcasting and Communication*, *18*(1), 113-117.
12-Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, *60*(6), 84-90.
13-LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation*, *1*(4), 541-551.
14-Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3431-3440).
15-Maddison, C. J., Huang, A., Sutskever, I., & Silver, D. (2014). Move evaluation in Go using deep convolutional neural networks. *arXiv preprint arXiv:1412.6564*.
16-Malik, K., Robertson, C., Braun, D., & Greig, C. (2021). U-Net convolutional neural network models for detecting and quantifying placer mining disturbances at watershed scales. *International Journal of Applied Earth Observation and Geoinformation*, *104*, 102510.
17-Pena, F. A. G., Fernandez, P. D. M., Tarr, P. T., Ren, T. I., Meyerowitz, E. M., & Cunha, A. (2020, April). J regularization improves imbalanced multiclass segmentation. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)* (pp. 1-5). IEEE.
18-Ronneberger, O., Fischer, P., & Brox, T. (2015, October). U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention* (pp. 234-241). Springer, Cham.
19-Rosebrock, A. (2017). *Deep learning for computer vision with python: Starter bundle*. PyImageSearch.
20-Seyedhosseini, M., Sajjadi, M., & Tasdizen, T. (2013). Image segmentation with cascaded hierarchical models and logistic disjunctive normal networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2168-2175).
21-Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.