# Dense U-Net Deep Learning Technique for Semantic Iris Segmentation

**k. Vijayakeerthi**
*UG student*
*Dept. of Computer Science and Business Systems*
*Panimalar Engineering College*
Chennai-600123, India

**A. Asmitha yugasre**
*UG student*
*Dept. of Computer Science and Business Systems*
*Panimalar Engineering College*
Chennai-600123, India

**S. Swathi**
*UG student*
*Dept. of Computer Science and Business Systems*
*Panimalar Engineering College*
Chennai-600123, India

**S. SriVidhya**
*UG student*
*Dept. of Computer Science and Business Systems*
*Panimalar Engineering College*
Chennai-600123, India

**V. Neela Priyadharshini**
*UG student*
*Dept. of Computer Science and Business Systems*
*Panimalar Engineering College*
Chennai- 600123, India

**Abstract:** he process of iris segmentation is crucial to iris identification. To maintain the accuracy of an iris identification system based on iris segmentation, errors existing in the present stage are limited. It works badly because of a non-ideal environment such as ambient lack of user involvement and light noise. Under these non-ideal circumstances, current segmentation techniques based on local gestures are unable to acknowledge the iris bounds, and the mistakes caused at the segmentation step are passed to all future stages, resulting in reducing dependability and accuracy. This research presents a Dense U-Net-based iris segmentation technique to overcome the issues with existing algorithms in complicated scenes and cross- device applications. Iris is segmented by merging a dense network with a traditional U-net, which uses narrower and fewer parameters and information, which makes training deep networks easier. The obtained results were close to the floor truths, the accuracy was also excellent, and our proposed model performs better in classification.

Keywords: Iris segmentation, Unet, DenseNet, Dense Unet, Deep learning, Accuracy.

## 1. Introduction

The segmentation technique divides an image into areas. Remote sensing and tumor detection are two example of applications where segmentation is used. Segmentation is an image processing technique for separating objects and textures in images. Semantic segmentation and instance segmentation are two examples of segmentation. But semantic segmentation is the subject of our discussion.

### A. Semantic segmentation

An image is used as input to semantic segmentation [22], and the result is a category for each pixel. In this approach, the most crucial goal is to categorize each pixel into one of the numerous groups. To put it another way, all pixels belonging to the bicycle would be grouped and all pixels belonging to the rider would be grouped, as would those belonging to the background, and so on.

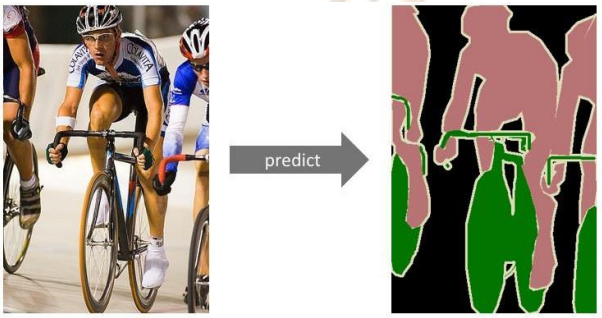


**Fig-1 Example illustrating semantic segmentation**

The semantic segmentation can be implemented using U-Net architecture. Basically, U-Net was developed by olaf Ronneberger for segmenting biomedical images. In addition to that dense network is included with traditional U-Net for segmenting IRIS images for better accuracy.

A Traditional U-Net [11][12] has 3 parts. The first part is the contraction path which is also called Downsampling. The second part is Bridge. And the third part is the Expansion part which is also called Upsampling.

The encoder, which is constructed with multiple convolutional layers, pooling layers, and dense network layers, where the image's context is collected. And decoder is the second part that uses transposed convolutions to discover the localization. Hence, it contains a dense network there are no limitations on image shape.

### B. Convolutional layer

Basically, the convolutional layer [13] takes 2 arguments

a) Dimensions of input image L1xW1xH1 in the case of the 3D image.
b) The second argument is k filters. The size of the k filters is X x X x channels.
The hyperparameters are filters, size of filters, stride, and padding (zero padding).

The output of the convolutional layer is also a 3D image because the input image is also a 3d image L2 x W2 x H2

Where L2 = filters

W2=(W1-size of the filter+2*padding)/Stride+2

H2=(H1-size of the filter+2*padding)/Stride+2

### C. Max pooling layer

The major aim of the max-pooling layer is to shrink the dimensions of feature maps. So that the network contains minimum number of parameters.

The inputs to the Max pooling layer are the shape of the image L1 x W1 x H1 hyperparametersters are filter size and Stride.

Output from the max-pooling layer will be an image of L2 x W2 x H2

Where L2= L1

W2=(W1-size of the filter)/Stride+1

H1=(H1-size of the filter)/stride+1

In general, the pixel with the highest value is chosen from each block of a particular size of the input feature map to create a pooled feature map. The most crucial function is to maintain just the most significant characteristics of each region and discard the rest. The breadth and height of an image in a convolutional network gradually diminish owing to max-pooling, downsampling, which aids in aiming on the details in the inner layers and on a larger context of the picture. As the number of channels grows, it becomes possible to extract numerous complicated features -from the image.

### D. Transposed Convolution

It's a technique for upsampling an image with settings that can be learned. Fractionally strided convolution or deconvolution is another name for it. Transposed convolution is the inverse of traditional convolution, resulting in a high-resolution picture from a low-resolution image. Backpropagation can be used to learn the parameters.

### E. Importance of iris segmentation

Deeply learned features are used in conjunction with a traditional pipeline in the current methodologies. Alternatively, they can employ a fine-tuned pre-trained model for iris recognition. For a variety of reasons, the iris has been an essential organ in the human body for identification of facial or biometric.

It Among all the internal organs, the cornea has the least amount of vulnerability and is shielded by the delicate and transparent corneal membrane. In order to control the diameter of the pupil, sphincter pupillae and dilator pupillae, which form the geometric shape of the iris, must work in combination. In addition to its delicate texture, the iris is hard to distinguish from other flowers. Iris segmentation is the most powerful iris recognition technology since incorrectly segmented iris areas can destroy biometric templates, which results in very poor recognition. As described in, there are several approaches for iris segmentation. Occlusion, picture quality, light, and other variables have an impact on iris recognition.

## 2. Literature review

Dugimpudi Abhishek reddy et al. [1] have proposed some methodologies for segmenting iris images. The obtained results were very near to the floor truths and quite impressive.

S. Prabhakar et al. [2] have discussed biometric recognition systems, like fingerprint and iris recognition. And they have discussed their entry into the field of security.

James R.Matey et al. [3] have discussed the working of iris recognition systems in less confined conditions and how images are acquired for iris recognition.

H. Proença et al. [4] have discussed mainly the examination of error rates and their impact on segmentation stage accuracy.

H. Hofbauer et al. [5] have provided a ground truth images database for segmenting iris images, they discussed methods for using the database and choosing an algorithm for segmentation of iris.

Olaf Ronneberger et al. [6] U-Net and its use in biological image segmentation were introduced for the first time by Olaf Ronneberger. In the realm of biomedical, the U-Net design had shown positive outcomes.

R. Jillela et al. [7] have discussed various issues with the segmentation process and presented some key iris segmentation strategies.

A.Pando et al. [8] have discussed the relevance of biometrics in daily life is examined in-depth, with a focus on how biometrics' incorporation into security has revolutionized the industry.

S.Minaee et al. [9] have discussed VGG-Net and extracted deep characteristics from it, then utilized in iris identification and evaluated it on two prominent iris datasets, the suggested technique produced good results with a high accuracy rate.

## 3. Proposed Methodology

As mentioned above, layers that are present in the U-net model, let us modify the U-net model and add the dense network to it. Then the traditional U-Net becomes Dense U-Net. Below is the architectural diagram of dense U-Net and sample images of CASIA4i dataset

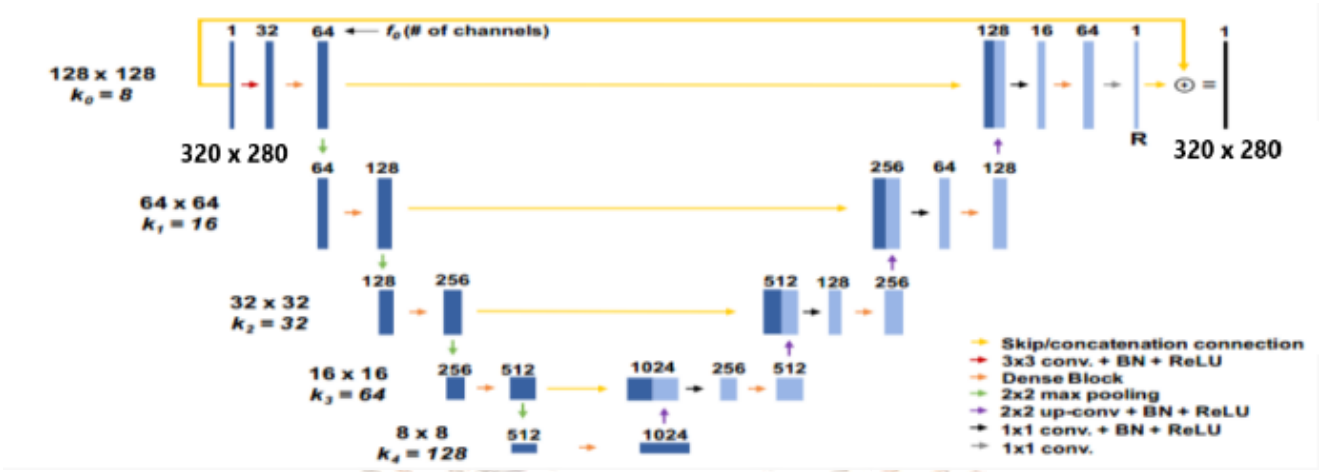


**Fig-2 Dense connected U-Net architectural diagram**

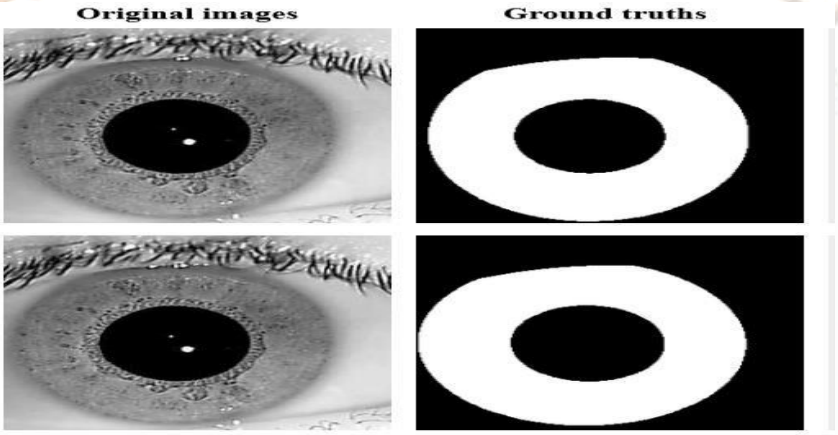


**Fig-3 Sample iris dataset**

### A. Dense Network

A dense network is one in which each node's number of links approaches the maximum no.of nodes. Each node is connected to nearly every other node. A completely connected network is one in which every node is connected to every other node in the same way. The neural network of the brain, epidemic spread, and telecommunication networks are examples of higher link density. Alpha go is the first computer program- based convolutional neural network, that has beat top players(Human) in the board game. A convolutional network model has also performed well in lung disease detection compared to medical specialists. It has shown the value and promise in areas like finance, security, and autopilot..,etc. In the brain, neurons are often organized as densely connected networks. This suggests that dense networks may be beneficial for processing information. Our understanding of how the brain works can be improved by studying dense networks. Below is the schematic diagram of the dense network.

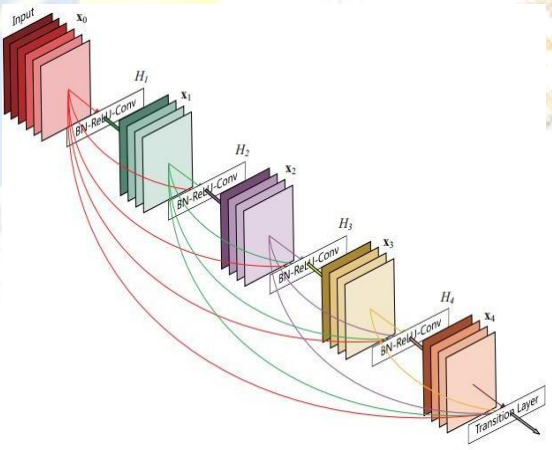


**Fig-4 Schematic diagram of dense network**

Researchers have made significant progress in the dense-connected convolutional neural network research over the last few years, extending the 1D layer-by-layer connection methods of classic neural networks to a 3D

cross-layer mode. In a typical neural network, each layer receives input signals from the top layer and then transfers extracted features to the next layer. In the dense U-Net, each layer of the network is connected immediately. This means that all layers of a network can have their inputs based on the outputs of their predecessors. To reduce redundant information, each layer of Dense U-Net learns a relatively small number of features that are passed on as input to all layers behind it. It In spite of the lack of dense connections, each layer of the network has to learn a lot if the fit phenomenon is not to occur. Dense U-Net also describes a bottleneck layer and transition layer, which are designed to optimize the reuse of features at the layer level and to compress local computation at the layer level. This allows further exploration and enhancement of the model.

**B. Dense U-Net**

The following Fig. 2 depicts a densely connected U-Net architecture. A dense U-Net [14] network has a u-shaped network structure with two components, encoding and decoding, that provides semantic segmentation. The image is deconstructed at various layers as a result of the convolution kernel pooling being used to continuously extract features in the coding section, as seen in Fig. 2. Finally, a decoding technique, expansion path, and convolution functions are used to perform semantic segmentation at the pixel level. The network may learn different global and native properties of pictures (images) using this method at different scales. Dense connections can facilitate more rapid training. The number of network characteristic graphs is indicated by the letter K. The convolution layer's coaching is updated frequently, just like every other layer's.

The previous convolutional layer in a dense block is connected to all or any subsequent convolutional layers through a channel cascade. As shown in Figure 4, the feature maps in the Lth layer's output are also present in the F + k(l+1) layer's input, where is the number of feature maps included in the input. Each 3x3 convolution is followed by a 1x1 convolution to minimise the number of input feature graphs to four in order to enhance computational speed. The connection between the inputs and outputs of the ℓ layers is the dense block's ultimate output.

The suggested CNN structure, which uses dense connections, decomposes the convolution kernel into input and output. A convolution kernel of zero size is pplied to the input picture, and then the result is merged in dense connections. In Through this approach, it is possible to combine both local and global information since a network's learned features are plentiful. The features of CNN are more easily learned through dense connections. Using dense connections, the proposed CNN structure allows the size of the convolution kernel to be decomposed between input and output. A convolution kernel of no size is applied to the input picture, and the output is then combined using dense connections. In this approach, the network's learned features are extra plentiful, and both native and global details could be considered. CNN has a greater no.of characteristics that could be grasped via dense connections.

**C. Defining Dense Connected U-net architecture**

The existing traditional U-net model is taken and fine-tuned so that it performs well with our CASIA4I dataset. Additionally, dense layer is added to the traditional U-net model. The input to this network is 320x320x1, and the output is the same size as the input but with the segmented iris from the original image. Convolution, Dense, max pooling, and transposed convolution layers are piled up in shape of a "U" in the network. There are more activities going on internally, such as batch normalization and activation, which are common in ConvNets.

**4. Dataset Description**

In this paper IITD iris dataset[23], UBIRIS v2 [24], and CASIA4i [25] were used. IITD dataset contains 2240 images extracted from 224 subjects, having 320x240 dimensions. UBIRIS v2 dataset contains nearly 11000 images. In this paper, 2250 images have been taken. All the images of the UBIRIS v2 dataset have 400x300 dimensions. we have reshaped images of UBIRIS v2 dataset from 400 x 300 to 320 x 240. CASIA4i dataset contains 5270 images and has shape of 320 x 280. For downloading UBIRIS v2 and IITD datasets we need to fill the application forms of respective websites. Users need to sign on the application form, scan it and send mail to the authors. Then the authors provide an unzipping key. Using that unzipping key we can unzip the file. For the UBIRIS4i iris dataset, we need to create an account on the website named idealtest.org. Then the dataset can be downloaded. For the sake of model, the shape of CASIA4i dataset has been reduced from 320 x 280 to 320 x 240. Hence, all the datasets has been reduced to 320 x 240.

**5. Implementation**

**A. Preprocessing**

There are two parts in all the three datasets. First part is, set of eye images and second part is ground truths. Ground truths are the segmented iris images. This. At first the images are paired with ground truths (segmented iris images) and undergoes training. In the validation part segmented images are compared with ground truths. According to that accuracy is calculated. Using the accuracy Error rate is calculated.

### B. Error Rate calculation

Error rate can be calculated by using below formula.

$$Error\ Rate = \frac{1}{N \times m \times n} \sum \sum_{x,y \epsilon (m,n)} G_(x, y_) \ \text{XOR}\ (x, y)$$

Where N,m,n are no.of height and width of the images tested. G is the Ground truth and M is the segmented images. x and y are the co-ordinates of M and G. Here XOR operation is used for evaluating dispute pixels between ground truths and segmented images.

### C. Experimental setup

The code execution is carried out in Kaggle.com. Kaggle is a platform for efficient writing and running Python programs. It is primarily utilized for deep learning applications because it has a lot of resources, including RAM, GPU, and CPU. And it provides TPU (Tensor processing Unit) for processing tensors. Below table-1 provides the detailed information about the software and hardware specifications of Kaggle.com.

| Specification | Description |
|---|---|
| GPU | NVIDIA Tesla P100-PCIE-16GB |
| CPU | Intel Xenon CPU @2.30GHz |
| RAM | ~13 GB |
| Disk space | ~60 GB |
| Programming language | Python 3.8 |

**Table-1 Software and hardware components used**

The CASIA4i dataset, which comprises of 5270 images of eyes and related floor truths, is trained using the Dense U-Net architecture described above. The experiment is carried out inorder to obtain highly segmented iris images with high accuracy

### C. Results and simulation

The experiment is performed on the Kaggle using the dense U-Net architecture on the IITD dataset, UBIRIS v2, CASIA4i, and more number of hyperparameters are used. Adam Optimizer is one of them; the loss is binary crossentropy, the batch size is 16, and 100 epochs are used for training. Despite a large number of epochs, callbacks are utilized to prevent the model from becoming overfitted on the three datasets. Callbacks allow us to have more control over the training process by stopping it when it reaches a given level of accuracy/loss and saving the model as a checkpoint after each successful epoch, allowing us to tweak the learning rates over time. Early stopping is also included in the project. This acts as a trigger. It helps in preventing in overfitting of the model. And below table- 2 contains error rates calculated for every model.

| Model | IITD | UBIRIS v2 | CASIA4i |
|---|---|---|---|
| $CNN_{28}$ | 1.37% | 1.11% | 1.19% |
| $CNN_{40}$ | 1.27% | 1.01% | 1.12% |
| $CNN_{56}$ | 1.14% | 0.95% | 0.99% |
| $CNN_{80}$ | 1.07% | 0.90% | 0.95% |
| FCN | 0.96% | 0.68% | 0.62% |
| U-Net | 0.88% | 0.88% | 0.75% |
| Dense U-Net | 0.77% | 0.55% | 0.69% |

**Table-2 represents the Error rates obtained by the models for three different iris datasets**

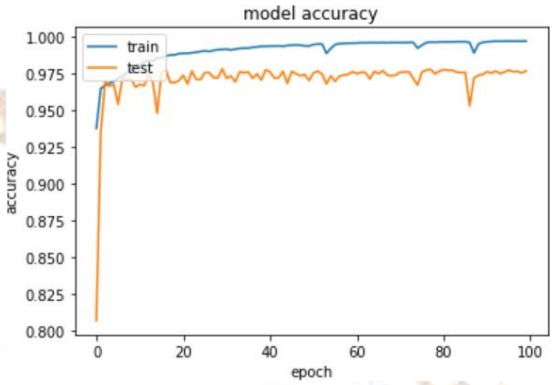


**Fig-5 Accuracy during training and testing**

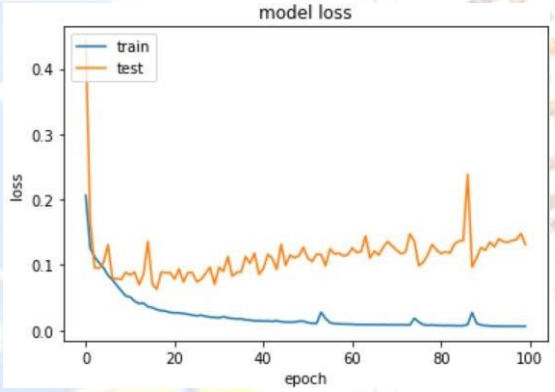


**Fig-6 loss during training and testing**

**Above Fig-5 and Fig-6 are model accuracy and model loss during training and validation process.**

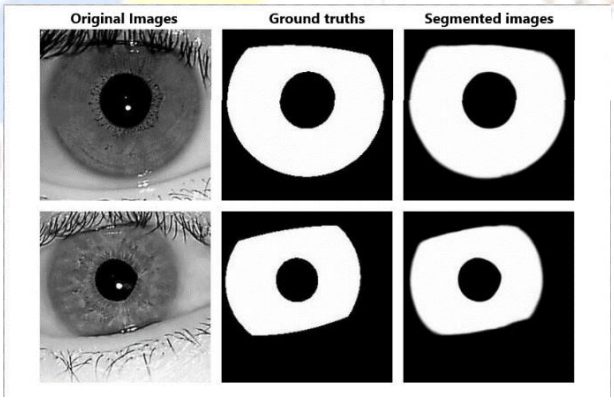


**Fig-7 comparison between original images, Ground truths and segmented images**

Result obtained was very impressive and the segmented images are very close to the ground truths. Overall error

rate was obtained by Dense Unet was very good and impressive. For IITD dataset the Error rate was 0.77%, For UBIRIS v2 dataset the Error rate was 0.55%, and for CASIA4i dataset the Error rate was 0.69%. As per the results Dense connected U-Net model has better accuracy and Error rate. The training accuracy was around 99.8% and validation accuracy was around 97.96%. The performance is very good as compared to previous models.

## 6. Conclusion

For iris segmentation, we described a Dense U-Net-based method. This CNN-based Dense U-Net segmentation approach produces very accurate Iris segmentation that is quite near to the ground truth images. Even though the outcomes were superior, this model only required a little quantity of training data. In terms of semantic segmentation, Dense U-Net excels. As a result, we could conclude that employing Deep Learning for segmenting iris delivers superior score in the realm of biometrics than traditional methods. If the dataset is big enough, it may yield even better findings. In terms of semantic segmentation. As a result, we may conclude that adopting a deep learning algorithm for segmenting iris, outperforms traditional methods in the field of biometrics.

## 7. References

[1] Dugimpudi Abhishek reddy “Semantic segmentation of iris using U-Net in deep learning” International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-9 Issue-1, May 2020

**[2]** S. Prabhakar, S. Pankanti, and A. K. Jain, “Biometric recognition: security and privacy concerns,” IEEE Secur. Priv. Mag., vol. 1, no. 2, pp. 33–42, 2003.

[3] J. R. Matey et al., “Iris on the Move: Acquisition of Images for Iris Recognition in Less Constrained Environments,” Proc. IEEE, vol. 94, 2006

[4] H. Proença and L. A. Alexandre, “Iris recognition: Analysis of the error rates regarding the accuracy of the segmentation stage,” Image Vis. Comput., vol. 28, no. 1, pp. 202–206, 2010

[5] H. Hofbauer, F. Alonso-Fernandez, P. Wild, J. Bigun, and A. Uhl, “A ground truth for Iris segmentation,” in Proceedings - International Conference on Pattern Recognition, 2014

[6] U-Net: Convolutional Networks for Biomedical Image Segmentation , Olaf Ronneberger, Philipp Fischer, and Thomas Brox, Computer Science Department and BIOSS Centre for Biological Signalling Studies, University of Freiburg, Germany

[7] R. Jillela and A. A. Ross, “Methods for Iris Segmentation,” in Handbook of Iris Recognition, M. J.Burge and K. W. Bowyer, Eds. London: Springer London, 2013, pp. 239–279.

[8] A.Pando, “Beyond Security: Biometrics Integration Into Everyday Life,” forbes. [Online].

[9] S.Minaee, A Abdolrashidiy, and Y.Wang, “An experimental study of deep convolutional features for iris recognition,” in 2016 IEEE Signal Processing in Medicine and Biology Symposium, SPMB 2016 - Proceedings, 2017.

[10] R. P.Wildes, ``Iris recognition: An emerging biometric technology," *Proc. IEEE*, vol. 85, no. 9, pp. 1348_1363, Sep. 1997.

[11] O. Ronneberger, P. Fischer, and T. Brox, ``U-Net: Convolutional networks for biomedical image segmentation," 2015, *arXiv:1505.04597*. [Online]. Available: https://arxiv.org/abs/1505.04597

[12] Z. Zeng, W. Xie, Y. Zhang, and Y. Lu, ``RIC- Unet: An improved neural network based on Unet for nuclei segmentation in histology images," *IEEE Access*, vol. 7, pp. 21420_21428, 2019.

[13] Y. Yang, P. Shen, and C. Chen, ``A robust iris segmentation using fully convolutional network with dilated convolutions," in *Proc. IEEE Int. Symp. Multimedia (ISM)*, Dec. 2018, pp. 9_16.

[14] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, ``Densely connected convolutional networks," 2016, *arXiv:1608.06993*. [Online]. Available: https://arxiv.org/abs/1608.06993

[15] J. Kim, J. K. Lee, and K. M. Lee, ``Accurate image super-resolution using very deep convolutional networks," 2015, *arXiv:1511.04587*. [Online]. Available: https://arxiv.org/abs/1511.04587

[16] Y. S. Han, J. Yoo, and J. C. Ye, ``Deep residual learning for com- pressed sensing ct reconstruction via persistent homology analysis," 2016, *arXiv:1611.06391*. [Online]. Available: https://arxiv.org/abs/1611.06391

[17] C. Rathgeb, A. Uhl, and P. Wild. *Iris Biometrics: From Segmentation to Template Security*. vol. 59. Berlin, Germany: Springer, 2012.

[18] S. Z. Li and A. K. Jain, *Encyclopedia of Biometrics*. Berlin, Germany: Springer, 2010.

[19] J. Daugman, ``How iris recognition works," *EEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 21_30, Jan. 2004.

[20] W. Xu and F. F. Quan, ``Research on application of iris recognition based on improved algorithm," *Comput. Technol. Develop.*, vol. 23, no. 3, pp. 34_37, Mar. 2013.

[21] F. Zhao, Y. Chen, F. Chen, X. He, X. Cao, Y. Hou, H. Yi, X. He, and J. Liang, ``Semi-supervised cerebrovascular segmentation by hierarchical convolutional neural network," *IEEE Access*, vol. 6, pp. 67841_67852, 2018.

[22] Tobji, Rachida & Di, Wu & Ayoub, Naeem. (2019). FMnet: Iris Segmentation and Recognition by Using Fully and Multi-Scale CNN for Biometric Security. Applied Sciences. 9. 2042. 10.3390/app9102042.

[23] Ajay Kumar and Arun Passi, "Comparison and combination of iris matchers for reliable personal authentication," *Pattern Recognition*, vol. 43, no. 3, pp. 1016-1026, Mar. 2010.

[24] Hugo Proença, Sílvio Filipe, Ricardo Santos, João Oliveira, Luís A. Alexandre; The UBIRIS.v2: A Database of Visible Wavelength Iris Images Captured On-The-Move and At-A-Distance, IEEE Transactions on Pattern Analysis and Machine Intelligence, August, 2010, volume 32, number 8, pag. 1529-1535, ISSN: 0162-8828, Digital Object Identifier http://doi.ieeecomputersociety.org/10.1109/TPAMI.2009.66

[25] http://biometrics.idealtest.org/findTotalDbByMode.do?mode=Iris