



MIRD-Net for Medical Image Segmentation

Yongfeng Huang¹, Xueyang Li¹(✉), Cairong Yan¹, Lihao Liu¹, and Hao Dai²

¹ School of Computer Science and Technology, Donghua University,
Shanghai 201620, China

{yfhuang,cryan}@dhu.edu.cn, weike10015@gmail.com, 2181757@mail.dhu.edu.cn

² Foot and Ankle Department, Shanghai Guang Hua Hospital of Integrated
Traditional Chinese and Western Medicine, Shanghai 200052, China
13564799291@163.com

Abstract. Medical image segmentation is a fundamental and challenging problem for analyzing medical images due to the approximate pixel values of adjacent tissues in boundary and the non-linear feature between pixels. Although fully convolutional neural networks such as U-Net has demonstrated impressive performance on medical image segmentation, distinguishing subtle features between different categories after pooling layers is still a difficult task, which affects the segmentation accuracy. In this paper, we propose a Mini-Inception-Residual-Dense (MIRD) network named MIRD-Net to deal with this problem. The key point of our proposed MIRD-Net is MIRD Block. It takes advantage of Inception, Residual Block (RB) and Dense Block (DB), aiming to make the network obtain more features to help improve the segmentation accuracy. There is no pooling layer in MIRD-Net. Such a design avoids loss of information during forward propagation. Experimental results show that our framework significantly outperforms U-Net in six different image segmentation tasks and its parameters are only about 1/50 of U-Net.

Keywords: Medical image segmentation · Inception · RB · DB

1 Introduction

Medical image segmentation is the key to determining whether medical images can provide a reliable basis for clinical diagnosis and treatment. However, the borders between tissues in medical images may be blurred by the imaging acquisition, which increases the difficulty on segmentation. The classical CNN (non-fully convolutional networks) such as [18] and Residual connections network (Res-Net) [6] can only classify separate examples and not a whole segmented pixel, because the fully connected layers are used at the end of the network, which can only mark the category of the whole image and not per pixel. Nevertheless, in many medical imaging tasks, especially in medical segmentation, a class label is desired to be assigned to each pixel. The breakthrough by Cirosan

et al. [3] was due to sliding-window setup which can predict the class label of each pixel by providing a local context (neighbor region) around that pixel as input. They won the EM segmentation challenge at ISBI 2012.

However, the approach proposed by Ciresan et al. [3] has some limitations: (1) it needs a long time to process the training image because the network must be run for each neighbor region, and there is a slight redundancy due to overlapping between the neighboring regions. (2) it is hard to keep balance in context and localization accuracy. Smaller neighbor regions make the network see context weakly, while larger neighbor regions need more pooling layers that reduce the localization accuracy. Fully convolutional network (FCN) uses the convolutional layer to replace the fully connected layer, getting the probability of each pixel rather than the scalar of the whole image, which improves the accuracy of segmentation [17]. Moreover, the advantage of FCN is indeed the possibility to have a whole image and its segmentation as training inputs, rather than feeding all possible separate sub-images centered on each labelled pixel like the strategy in Ciresan et al. [3].

Inspired by FCN, U-Net, a symmetrical and fully convolutional network, was proposed [16] and widely used because of its elegant architecture. The network has a contracting path and an expanding path that is more or less symmetric to the contracting path, yielding an architecture like letter U. In the expanding path, pooling operators are replaced by upsampling operators to increase the resolution of the output, making the high resolution features from the contracting path combined with the upsampling output through the skip connections, which allows the network to learn more precise feature based on this information. However, the U-Net architecture has one drawback that is difficult to improve performance by shallowing or deepening its depth. Technically, the network with deeper depth is supposed to learn more features and results in better segmentation, while gradients may vanish during the training period, making the network hard to train [8, 20].

In recent years, some variants of U-Net have been proposed [9, 13, 25]. And these network contain the approximate backbone consisting of downsampling layers, upsampling layers, and skip connection (see Fig. 1). The differences among them are the use of different modules and the connected way between layers. Residual Block (RB) [6] and Dense Block (DB) [7] are widely integrated into U-Net due to their scalability. And they can also make it is easy to train the network with deep depth, enabling the network to learn more represented information. Furthermore, the concatenation in Dense-Net makes the final classifier use features from all previous layers (different from classical CNN approaches), resulting in better performance of classification. The challenge is to create a network that excels in accuracy without gradient vanishing and with fewer parameters.

Motivated by previous work and existing problem in U-net, we propose a new symmetrical network named MIRD-Net. It integrates Residual Block (RB) [6] and Dense Block (DB) [7] into the inception architecture [20], aiming to excel in accuracy with fewer parameters. The exploration of our network consists of four steps. First, we choose ten layers (including the pooling layers, the

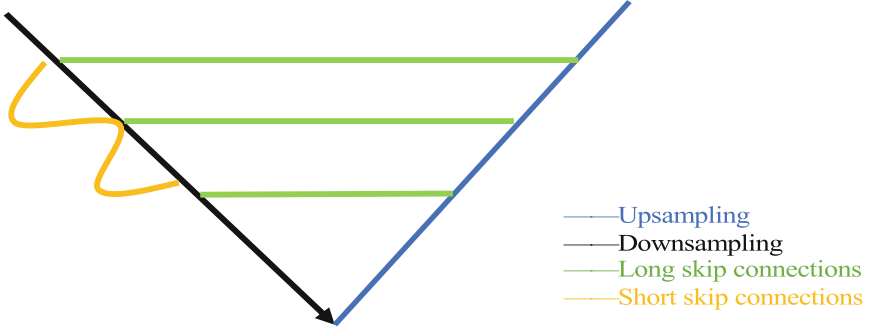


Fig. 1. Approximate backbone in the variants of U-Net.

convolutional layers, the upsampling layers) as a backbone of the network. Secondly, we try to add RBs as functional modules to the up-down sampling path, and the positions of blocks are discussed through experiments. Thirdly, the backbone is equipped with two DBs when the best positions of RBs are determined. We combine two DBs with RBs, getting two Mini-Inception-Residual-Dense Blocks (MIRD) to replace two RBs which are located in downsampling path. Finally, the pooling layers are replaced with 3×3 convolutional layers. The main contributions are as follows: (1) a shallower backbone to decrease the number of parameters. The combination of inception architecture, RB and DB makes the network learn more represented features; (2) simple and flexible implementation of our proposed network architecture; (3) great performance for challenging medical image segmentation tasks.

2 Related Work

2.1 FCN in Medical Images

CNNs have reached the-state-of-the-art in medical segmentation after FCN was proposed, consisting of symmetrical backbone with downsampling path and upsampling path, which allows combining the feature extracted by downsampling with the feature recovered by upsampling through skip connections. Korez et al. [10] proposed a 3D version of FCN to process the MRI image of the human spine. Zhou et al. [24] combined 2D FCN with 3D Majority voting algorithm, achieving great performance in Three-Dimensional segmentation task of human torso CT. Olaf Ronneberger et al. [16] extended FCN to a symmetrical U-Net and won the first prize on the ISBI cell tracking challenge 2015.

2.2 Improvements Based on U-Net

Comparing U-Net with FCN, one important modification in U-Net is skip connections, making the network to fuse the information of the up-down sampling path, which can generate high resolution and more accurate mask. In addition,

the U-shaped architecture can be straightened into Line-shaped network approximately, which is similar to the Dense-Net where skip connections are used [7]. Inspired by Dense-Net, Z. Zhou et al. [25] altered U-Net by transforming skip connections into dense skip connections, which makes each node connected with all previous nodes like Dense-Net. Drozdal et al. [4] demonstrated the importance of skip connections in U-Net and combined cross entropy and dice coefficient as a loss function. Cicek et al. [2] proposed a 3D version of U-Net to implement 3D image segmentation by inputting continuous 2D slices. Fausto et al. [14] converted the 3D version of U-Net to V-net and used dice coefficient instead of binary cross entropy as a loss function to segment the prostate MRI image. Brosh et al. [1] added skip layers to the first downsampling layer and the last upsampling layer in U-Net individually, which can discover the lesion of brain MRI precisely. X. Li et al. [12] proposed H-DenseUnet with mixed dense connections, reducing the memory consumption of GPU during the training step and excelling in Liver MICCAI 2017. Steven Guan et al. [5] designed FD-Unet to remove artifacts of 2D PAT images reconstructed from sparse data and compared FD-Unet with the standard U-Net in terms of reconstructed image quality.

2.3 Functional Operations

In addition to the improvements in architecture, advances are being made in some functional operations. Pooling layers as a basic module are widely used in CNNs, which can enlarge the Receptive Field (RF) to make network get more effective information during the training period. However, Pooling operations also lose some spatial information due to reducing the size of images. Theoretically, we cannot remove pooling layers and enlarge the size of convolutional kernels directly, because the larger kernel would result in increasing computational consumption. The larger kernel can be replaced by multiple smaller kernel, keeping the parameter low, which can be seen as imposing a regularization on the larger kernel [18]. Assuming that now we have the 3×3 kernel and the 7×7 kernel, and separately implementing the 3×3 kernel three times, the 7×7 kernel once on the same image. According to (1), we can get the same size of output if other conditions (S and P) are consistent. Moreover, F is assumed to be the channels both of input and output, then a single 7×7 convolution would require $7 \times 7 \times F \times F = 49F^2$ parameters, the triple 3×3 convolutions are parametrized by $3(3^2F^2) = 27F^2$.

$$N = (W - F + 2P)/S + 1 \quad (1)$$

where W is the size of an input image, N is the size of an output image through convolutional operations, F is the size of kernels, P is padding size and S is sliding step. Yu. F et al. [22] used the dilated convolution to replace the pooling operation, which has two advantages. First, it can enlarge the RF without losing information like the pooling operation. Secondly, it can be applied in well situations where the image requires global information. Conditional Random Field (CRF) has been used in the field of image segmentation since 2011 [11]. Later, the CRF was added as a functional module to the back end of the neural network to optimize the segmentation result [23].

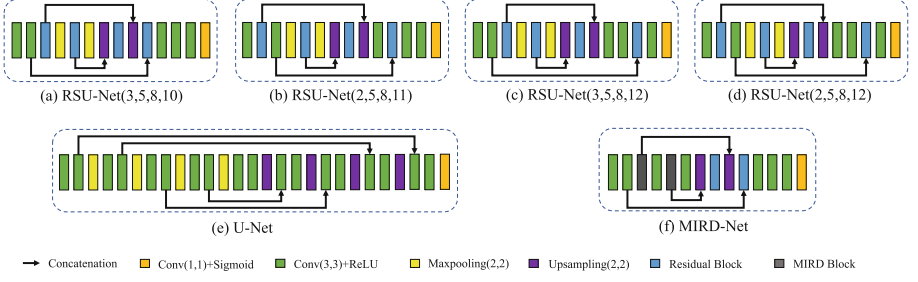


Fig. 2. Brief description of U-Net, Residual-Shallow U-Net (RSU-Net), MIRD-Net. Residual Block, Dense Block and MIRD Block are shown in Fig. 3, Fig. 4 and Fig. 5 respectively. And (3,5,8,10), (2,5,8,11), (3,5,8,12), (2,5,8,12) represent the positions of Residual Block in the Residual-Shallow U-Net.

3 MIRD-Net

3.1 Overview

The MIRD-Net proposed by us is briefly shown in Fig. 2(f). It consists of skip connections, downsampling path and upsampling path, but has four different points from U-net: (1) a shallower backbone is used in MIRD-Net, aiming to keep parameter low; (2) the MIRD-Net has no pooling layers, such a design avoids loss of information during forward propagation; (3) MIRD-Net is also designed with MIRD Blocks (Mini-Inception-Residual-Dense Block), which makes the network learn more represented features; (4) two Residual Blocks (RB) are embedded in the upsampling path.

3.2 MIRD Block

Residual Block. Experiments have shown that the extraction of features is affected by the depth of the network [19,20]. Increasing the layers of a network can make it learn more features, but it can also be accompanied by over-fitting, gradients vanishing and other issues, which leads to the extracted features not being fully used. K. He et al. [6] proposed a residual network, which can reuse the feature from the previous layer (see in Fig. 3) and ease the training of deeper networks.

$$x_l = F_l(x_{l-1}) + x_{l-1} \quad (2)$$

where x_l represents the output of the current layer, x_{l-1} is the output of the previous layer, and $H_l(\cdot)$ is the non-linear calculation including Conv, ReLU [21], BN [8] in the Residual Block.

Inspired by that, we first reduce the number of layers of U-Net [16] to keep the parameters low, then depositing four Residual Blocks (RB) on up-down sampling path (two RBs on upsampling path and another two RBs on downsampling path) to optimize the performance of the network. Theoretically, the number of Residual Blocks can be chosen alternatively but guided by the target of low

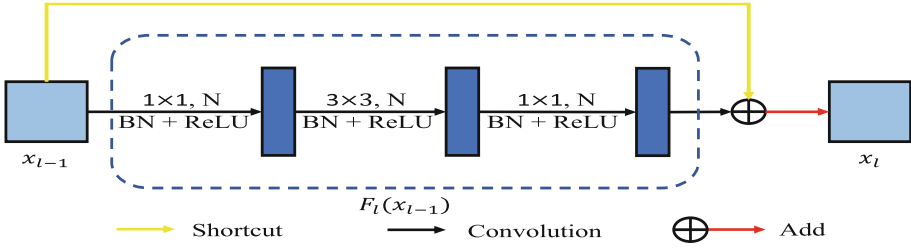


Fig. 3. The Residual Block used in the experiment, 3×3 and 1×1 are the size of filters with N channels, and $F_l(x_{l-1})$ includes Conv, ReLU [21] and BN (Batch Normalization [8])

parameters and good performance, four Residual Blocks are a more reasonable choice. In the case of four Residual Blocks, we have a further discussion on the position where the Residual Blocks are located (see Fig. 2(a–d)). And after the position determined, we optimize the Residual Block to get a more elegant block in the same position.

Dense Block. Within the Dense Block [7], each layer is connected to all previous layers through concatenation as used in U-Net [16], which has several advantages: (1) it strengthens feature propagation; (2) it alleviates the gradient vanishing during the training period; (3) it makes the feature reused. Figure 4 shows the layout of a Dense Block. Formally, the x_l layer are connected with all previous layers ($x_{l-1}, x_{l-2}, \dots, x_0$):

$$x_l = f[* (x_{l-1}, x_{l-2}, \dots, x_0)] \quad (3)$$

where x_l represents the output of the current layer, $x_{l-1}, x_{l-2}, \dots, x_0$ are the output of all previous layers connected to x_l , $*(\cdot)$ is the concatenation operation, $f[\cdot]$ is the non-linear calculation including Conv, ReLU [21], BN [8] in the Dense Block.

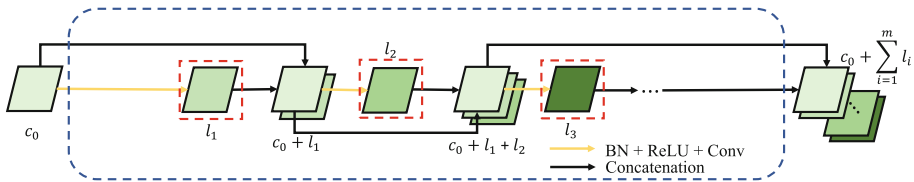


Fig. 4. The architecture of a Dense Block with m convolution layers, c_0 is the channel of input image and l_i (growth rate) is the channel of the convolved image. ReLU [21] and BN [8] are attached to each convolution layer in a Dense Block.

The Dense Block is effective for our proposed network, which mainly leads to three major advantages: (1) the parameter space can be managed simply through the l_i (growth rate); (2) generally, it is hard to make sure that gradients flow smoothly in back propagation. But the dense connections in Dense Block can alleviate the gradient vanishing; (3) the datasets used in our experiments are small. Therefore, it is important to reuse the features, which can make the network get more information. The dense connections comprehensively utilize features from previous layers (instead of only the last layer), thus making it easier to get a smooth decision function with better performance.

Motivated by Residual Block and Dense Block, we integrate them into the inception architecture [20], which is named Mini-Inception-Residual-Dense Block (see in Fig. 5). And depositing two MIRD Blocks on downsampling path where two Residual Blocks are located to replace them, while removing pooling layers. The reason that drives us to remove the pooling layers is because pooling operations could discard some pixel-level information. Let us assume the x_l is the output of MIRD Block, and the x_{l-1} is the input of MIRD Block, the relation between x_l and x_{l-1} is defined in (4):

$$x_l = F(H(G(x_{l-1}))) + x_{l-1} \quad (4)$$

where $G(\cdot)$ is the function of Inception Block, $H(\cdot)$ is the calculation in Dense Block, $F(\cdot)$ is the calculation in Residual Block.

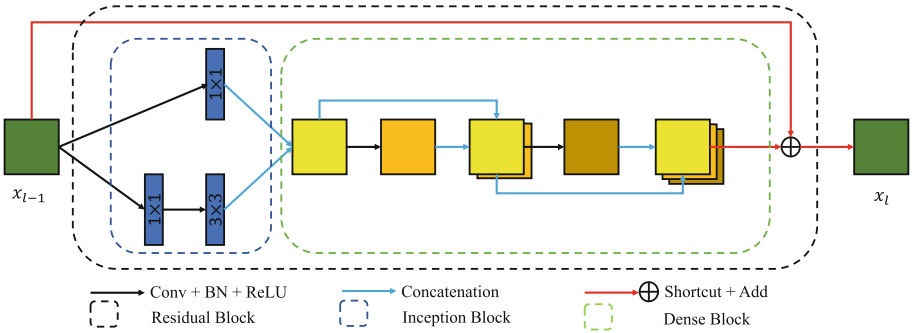


Fig. 5. Mini-Inception-Residual-Dense Block.

3.3 Evaluation Metrics

We use the number of parameters of each network and a well-known Dice coefficient for evaluation. The size of each dataset used in our experiment is small like the cells dataset used in U-net (only 30 images), which is inappropriate to divide them into three parts including training set, validation set and test set. Therefore, we split each dataset into five subsets (F1–F5) equally and run a 5-fold

cross-validation used in [15]. The MDice (Mean Dice coefficient) and StdDice (Std of Dice coefficient) are defined in (5) (6):

$$MDice = \frac{1}{r \times m} \sum_{i=1}^r \sum_{j=1}^m \frac{2|A_{ij} \cap B_{ij}|}{|A_{ij}| + |B_{ij}|} \quad (5)$$

$$StdDice = \sqrt{\frac{\sum_{i=1}^r (\frac{1}{m} \sum_{j=1}^m \frac{2|A_{ij} \cap B_{ij}|}{|A_{ij}| + |B_{ij}|} - MDice)^2}{r - 1}} \quad (6)$$

where A_{ij} is the predicted image, B_{ij} is the ground truth corresponding to A_{ij} , and m is the number of images in one subset, r is the fold used in cross-validation. The medical segmentation tasks in our experiments are binary classification problem, so the ground truth B_{ij} is the 0–1 matrix.

4 Experiments

4.1 Experimental Platform and Datasets

The experiment was conducted on a computer with Intel(R) Core (TM) i7-7700 CPU @ 3.60 GHz, Nvidia GeForce GTX 1080 Ti, 16 GB RAM, and Samsung SSD 850 EVO 500 GB. The operating system is Windows 10(1801). All experiments were run under the Keras framework.

Electron microscope image of cells dataset used in U-Net contains 30 images [16]. The size of each image is 512×512 pixels. To compare with U-Net, we choose 30 images in other five datasets (Retinal extraction vessel, Nuclei, Lung, Cervical Cytology, Skin Lesion) respectively, which makes the size of datasets consistent. The detailed information about datasets is presented in Table 1.

Table 1. Detailed information of datasets.

Data name	Source	Image size	Modality
Retinal vessel	grand-challenge.org	$512 \times 512 \times 3$	Non-mydratic camera
Cells	ISBI 2012	512×512	Microscopy
Nuclei	Data Science bowl 2018	360×360	Microscopy
Lung	Kaggle	512×512	CT
Cervical cytology	grand-challenge.org	512×512	Microscopy
Skin lesion	ISIC 2017	$512 \times 512 \times 3$	Dermoscopy

4.2 Implementation Details

To start with, we explore the impact of depth on U-Net [16] by reducing and increasing the layers of U-Net. Secondly, based on shallower U-Net, we reduce more layers to get smaller backbone and add four Residual Blocks (RB) into

up-down sampling path (two RBs on upsampling path and another two RBs on downsampling path) and have a discussion on the position of Residual Block. Thirdly, based on the best position where the RBs are located; Inception, Dense Block and Residual Block are incorporated into Mini-Inception-Residual-Dense Block to replace the RBs in downsampling path, while the pooling layers also are removed.

For hyperparameters, each convolution in the block is followed by BN [8] and ReLU [21], using Adam optimizer with the following parameters: $\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1e-8$. The sigmoid function is used in the last layer because our target was a binary classification problem. Due to the small size of Computer's graphics memory, a batch size of 3 was used while setting the epochs to 30. The training image and its corresponding labels are simultaneously rotated counterclockwise by 90° , 180° , and 270° to enlarge the dataset, the kernel size is 3×3 and the stride is 1 in convolutional layers except the specific layer in the block.

For the block we used in the experiment, 1×1 convolutional layer is attached to the output of MIRD Block. And $f(\cdot)$ in the Dense Block (Eq. (2)) actually includes BN-ReLU-Conv(1×1)-BN-ReLU-Conv(3×3). The cross-entropy is used as the loss function for all the networks.

4.3 Results and Comparison

We apply deeper U-Net (DU-Net), U-Net, shallower U-Net (SU-Net), Residual-Shallow U-Net with different positions of Residual Blocks (see Fig. 2(a-d)) and MIRD-Net on six segmentation tasks (see Table 1). The Dice coefficient and the parameters of networks discussed are reported in Table 2 and Table 3. The segmented results on some example images are shown in Fig. 6.

Table 2. Average Dice coefficient and its standard deviation for 5-fold cross validation.

Models	Skin		Lung		Nuclei		Cervical		Vessel		Cells	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
DU-Net	0.455	0.063	0.632	0.069	0.851	0.085	0.844	0.025	0.651	0.105	0.847	0.032
U-Net	0.514	0.044	0.748	0.087	0.897	0.102	0.886	0.077	0.698	0.087	0.869	0.115
SU-Net	0.469	0.085	0.667	0.041	0.902	0.068	0.882	0.058	0.721	0.098	0.866	0.076
RSU(25812)	0.603	0.095	0.709	0.077	0.896	0.064	0.855	0.133	0.715	0.105	0.892	0.034
RSU(25811)	0.617	0.117	0.731	0.124	0.909	0.125	0.864	0.058	0.714	0.094	0.887	0.072
RSU(35812)	0.624	0.047	0.772	0.076	0.911	0.118	0.897	0.029	0.730	0.098	0.895	0.046
RSU(35810)	0.656	0.126	0.794	0.089	0.928	0.097	0.902	0.047	0.742	0.086	0.902	0.021
MIRD-Net	0.742	0.076	0.810	0.067	0.954	0.075	0.925	0.017	0.765	0.047	0.919	0.029

Table 3. The parameters of each network ($\times 10^6$).

DU-Net	U-Net	SU-Net	RSU(25812)	RSU(25811)	RSU(35812)	RSU(35810)	MIDR-Net
40.61	31.03	25.42	0.47	0.51	0.47	0.51	0.59

Table 2 shows average Dice coefficient and its standard deviation for 5-fold cross validation. When compared to U-Net, DU-Net decreases the accuracy, but SU-Net has better performance on Nuclei and Vessel. It shows that DU-Net is likely to overfit. The Residual Blocks in different positions of up-downsampling path can affect the performance of the network. RSU(35810) (Fig. 2(a)) performs best in all four RSU-Nets we discussed and outperforms U-Net in six datasets. Moreover, it can be seen that there is obvious improvement by MIRD-Net, which achieves elegant results. The parameters of MIRD-Net are only about 1/50 of U-net (Table 3), which saves the storage memory.

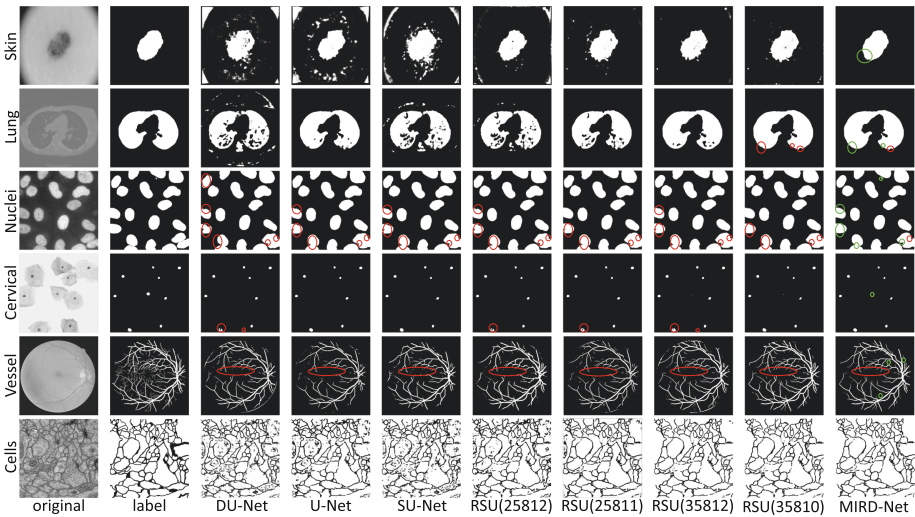


Fig. 6. Some results processed by DU-Net, U-Net, SU-Net, RSU-Net(2,5,8,12), RSU-Net(2,5,8,11), RSU-Net(3,5,8,12), RSU-Net(3,5,8,10) and MIRD-Net. (Color figure online)

For the slight differences which are hard to see directly, we use red and green circles to highlight each of them (Fig. 6). The region in red circles represents incomplete correct mask which is compared to the label, the green circles in the results of MIRD-Net show the better performance than that of other networks in the same region. Despite a few incomplete correct masks still exist in the final results, MIRD-Net outperforms the other networks discussed by us in segmenting tiny structure and the edge of target.

The reasons why MIRD-Net has a better segmentation result than that of U-Net are as following: (1) there are no pooling layers in MIRD-Net, such a design helps alleviate loss of information during forward propagation; (2) the different kernels (1×1 and 3×3) used in MIRD-Block can make the network obtain large-structure information and tiny-structure information simultaneously; (3)

MIRD-Net not only use the standard skip connections used in U-Net but also reuse the feature from previous layer in MIRD-Block, which results in more represented features learned by the network; (4) the connections used in MIRD Block can alleviate the gradient vanishing during the training period.

5 Conclusions

In this paper, we propose a new symmetric deep neural network for medical image segmentation. The new network takes advantage of Inception, Res-Net and Dense-Net, outperforming U-Net in six different image segmentation tasks. Its parameters are only about 1/50 of U-Net. Furthermore, the MIRD Block of our proposed architecture can also be simply added to other backbones as a functional module. The shortcoming is the way to select the position of MIRD Block, and we have not proven that the position of MIRD Block is the best choice in theory. In the future, the research would focus on the relevance between performance and the position of MIRD Block in different backbones, finding a better strategy to determine the position of MIRD Block and simplifying this structure.

References

1. Brosch, T., Tang, L.Y.W., Yoo, Y., Li, D.K.B., Traboulsee, A., Tam, R.: Deep 3D convolutional encoder networks with shortcuts for multiscale feature integration applied to multiple sclerosis lesion segmentation. *IEEE Trans. Med. Imaging* **35**(5), 1229–1239 (2016)
2. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D U-Net: learning dense volumetric segmentation from sparse annotation. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) *MICCAI 2016*. LNCS, vol. 9901, pp. 424–432. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46723-8_49
3. Ciresan, D., Gambardella, L., Giusti, A., Schmidhuber, J.: Deep neural networks segment neuronal membranes in electron microscopy images. In: *Advances in Neural Information Processing Systems (NIPS)*, vol. 25, pp. 2843–2851, December 2012
4. Drozdal, M., Vorontsov, E., Chartrand, G., Kadoury, S., Pal, C.: The importance of skip connections in biomedical image segmentation. In: Carneiro, G., et al. (eds.) *LABELS/DLMIA 2016*. LNCS, vol. 10008, pp. 179–187. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46976-8_19
5. Guan, S., Khan, A., Sikdar, S., Chitnis, P.: Fully dense UNet for 2D sparse photoacoustic tomography artifact removal. *IEEE J. Biomed. Health Inform.* (2019)
6. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778 (2016)
7. Huang, G., Liu, Z., van der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2261–2269, July 2017

8. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. In: Proceedings of the International Conference on International Conference on Machine Learning (ICML), pp. 448–456, July 2015
9. Jegou, S., Drozdal, M., Vazquez, D., Romero, A., Bengio, Y.: The one hundred layers Tiramisu: fully convolutional DenseNets for semantic segmentation. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 3–11, July 2017
10. Korez, R., Likar, B., Pernuš, F., Vrtovec, T.: Model-based segmentation of vertebral bodies from MR images with 3D CNNs. In: Ourselin, S., Joskowicz, L., Sabuncu, M.R., Unal, G., Wells, W. (eds.) MICCAI 2016. LNCS, vol. 9901, pp. 433–441. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46723-8_50
11. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected CRFs with Gaussian edge potentials. In: Advances in Neural Information Processing Systems (NIPS), vol. 24, pp. 109–117, December 2011
12. Li, X., Chen, H., Qi, X., Dou, Q., Fu, C., Heng, P.: H-DenseUNet: hybrid densely connected UNet for liver and tumor segmentation from CT volumes. *IEEE Trans. Med. Imaging* **37**(12), 2663–2674 (2018)
13. Mehta, S., Mercan, E., Bartlett, J., Weaver, D., Elmore, J.G., Shapiro, L.: Y-Net: joint segmentation and classification for diagnosis of breast biopsy images. In: Frangi, A.F., Schnabel, J.A., Davatzikos, C., Alberola-López, C., Fichtinger, G. (eds.) MICCAI 2018. LNCS, vol. 11071, pp. 893–901. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00934-2_99
14. Milletari, F., Navab, N., Ahmadi, S.: V-Net: fully convolutional neural networks for volumetric medical image segmentation. In: International Conference on 3D Vision, pp. 565–571, October 2016
15. Paul, A., Mukherjee, D.P.: Mitosis detection for invasive breast cancer grading in histopathological images. *IEEE Trans. Image Process.* **24**(11), 4041–4054 (2015)
16. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
17. Shelhamer, E., Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **39**(4), 640–651 (2017)
18. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations (ICLR), May 2015
19. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2818–2826, June 2016
20. Szegedy, C., et al.: Going deeper with convolutions. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9 (2015)
21. Glorot, X., Bordes, A., Bengio, Y.: Deep sparse rectifier neural networks. In: International Conference on Artificial Intelligence and Statistics (AISTATS), pp. 315–323, April 2011
22. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: International Conference on Learning Representations (ICLR), May 2016
23. Zheng, S., et al.: Conditional random fields as recurrent neural networks. In: 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1529–1537, December 2015

24. Zhou, X., Ito, T., Takayama, R., Wang, S., Hara, T., Fujita, H.: Three-dimensional CT image segmentation by combining 2D fully convolutional network with 3D majority voting. In: Carneiro, G., et al. (eds.) LABELS/DLMIA 2016. LNCS, vol. 10008, pp. 111–120. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46976-8_12
25. Zhou, Z., Rahman Siddiquee, M.M., Tajbakhsh, N., Liang, J.: UNet++: a nested U-Net architecture for medical image segmentation. In: Stoyanov, D., et al. (eds.) DLMIA/ML CDS 2018. LNCS, vol. 11045, pp. 3–11. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00889-5_1