A k-Dense-UNet for Biomedical Image Segmentation

Zhiwen Qiang, Shikui Tu(✉), and Lei Xu(✉)

Department of Computer Science and Engineering,
Centre for Cognitive Machines and Computational Health (CMaCH), SEIEE School,
Shanghai Jiao Tong University, Shanghai, China
{q7853619,tushikui,leixu}@sjtu.edu.cn

Abstract. Medical image segmentation is the premise of many medical image applications including disease diagnosis, anatomy, and radiation therapy. This paper presents a k-Dense-UNet for segmentation of Electron Microscopy (EM) images. Firstly, based on the characteristics of the long skip connection of U-Net and the mechanism of short skip connection of DenseNet, we propose a Dense-UNet by embedding the dense blocks into U-Net, leading to deeper layers for better feature extraction. We experimentally show that Dense-UNet outperforms the popular U-Net. Secondly, we combine Dense-UNet with one of the newest U-Net variants called kU-Net into a network called k-Dense-UNet, which consists of multiple Dense-UNet submodules. Skip connections are added between the adjacent submodules, to pass information efficiently, helping the model to identify fine features. Experimental results on the ISBI 2012 EM dataset show that k-Dense-UNet achieves better performance than U-Net and some of its variants.

Keywords: Image segmentation · Electron Microscopy · Skip connection

1 Introduction

High-resolution Electron Microscopy (EM) image segmentation has great value on many medical image applications, and it has shown important value in anatomy, radiation therapy, and biomedical research. Manual labeling the element in the EM images is normally done by a human neuroanatomist. However, since the medical image data is very complicated, it is time-consuming to manually label such data. Thus, artificial intelligence technology has gradually become a popular direction of medical image segmentation.

In recent years, deep learning approaches based on Convolutional Neural Networks [1–7] have been used on the EM image segmentation task. One of the most well-known attempts is U-Net [2]. It consists of a contraction path and a symmetric expansion path. To enable precise localization, high-resolution features from the contracting path are combined with the upsampled output. Such

© Springer Nature Switzerland AG 2019
Z. Cui et al. (Eds.): IScIDE 2019, LNCS 11935, pp. 552–562, 2019.
https://doi.org/10.1007/978-3-030-36189-1_46

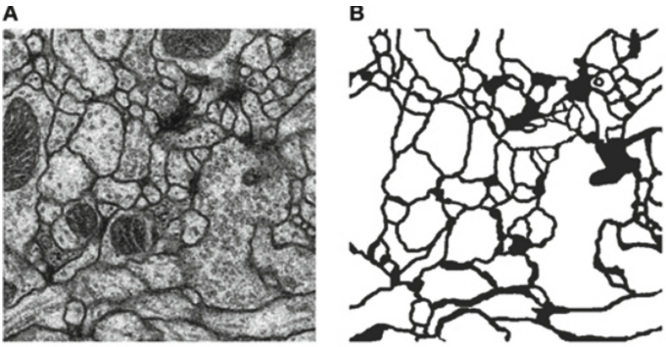

Fig. 1. Examples in the ISBI 2012 EM dataset: an EM image (A) and its ground truth segmentation (B).

skip connections enabled U-Net to work well on biomedical image segmentation tasks.

However, deep learning networks normally suffer the vanishing gradient problem, which limits their depth. He *et al.* [8] proposed ResNet, with skip connection between layers, the network's depth can be improved without damaging its performance. Huang *et al.* [9] proposed DenseNet, by using dense connections. It achieved better performance than ResNet in certain datasets.

Many following works were built on U-Net, and they tend to increase the depth of U-Net by certain metrics. FusionNet [3] applied residual blocks in U-Net to enable the model to have a larger depth to achieve better performance. This model also combined long and short skip connection together. kU-Net [7] consists of multiple submodule U-Net to sequentially extract information at different scales, from the coarsest scale to the finest scale. Each submodule will propagate information to the subsequent submodule to help feature extraction.

Actually, key features of recent models such as U-net, ResNet, and DenseNet have been found in Least Mean Square Error Reconstruction (Lmser) self-organizing network, which was first proposed in 1991 [17,18]. Lmser is a further development of autoencoder (AE) with favorable features, including Duality in Paired Neurons (DPN) and Duality in Connection Weight (DCW), which come from folding AE along the central coding layer. DPN can be regarded as adding shortcut connections between the paired neurons. The feedback links from decoder to encoder can be regarded as the skip connections between two U-Nets in kU-Net. More advances about Lmser are referred to a recent review in [19].

In this paper, Dense-UNet is proposed on segmentation of EM images. Not only we experimentally show that Dense-UNet outperforms U-Net, but also we proceed to a version called k-Dense-UNet that integrates Dense-UNet and kU-Net. Experimental results on the ISBI 2012 EM dataset show that k-Dense-UNet achieves better performance than U-Net and some of its variants.

Our contributions are as follows:

- We present k-Dense-UNet for EM image segmentation. It integrates Dense-UNet and kU-Net. There are corresponding long skip connections between adjacent modules, which can pass coarser-scale information to the next submodule, helping the model to identify finer features.
- Experimental results show that the proposed method can achieve better performance than U-Net and some of its variants in the ISBI 2012 EM challenge. Ablation study demonstrates that the skip connection between the adjacent submodules can enhance and refine the segmentation outputs.

2 Related Work

2.1 Deep Learning Methods for EM Image Segmentation

One of the earliest work was done by Ciresan *et al.* [1]. He implemented a succession of convolutional and max-pooling layers to predict the segmentation. Their work won the ISBI 2012 challenge. Long *et al.* [10] proposed the FCN structure to replace fully connected layers with convolutional layer which can preserve the spatial information. Since then, many variants of FCN have been proposed for EM image segmentation task. Shen *et al.* [11] created a multi-stage and multi-recursive-input FCN. The model can predict outputs at a different level in each stage, and combining all the predictions with the original images to generate the next stage's input. Ronneberger *et al.* [2] proposed U-Net architecture, which consists of four downsampling steps and four corresponding upsampling steps. Long skip connection layers exist between the downsampled feature map and the commensurate upsampled feature map, which can preserve low-level information. This model won the ISBI 2015 challenge. However, it still suffers the vanishing gradient problem, which limits the depth of U-Net. He *et al.* [8] proposed the residual blocks and demonstrated that short skip connections between layers can reduce the influence of vanishing gradients. Quan *et al.* [3] presented FusionNet, which embedded U-Net with residual blocks to combine short and long skip connections.

2.2 DenseNet Architecture

DenseNet [9] was proposed in 2017, by using dense connections, it reaches better results compares to ResNet [8] and pre-activated ResNet [12] on multiple datasets (CIFAR-10, CIFAR-100 [13], SVHN Small-Scale Dataset [14]). In DenseNet, each layer obtains additional inputs from all preceding layers and passes on its own feature-maps to all subsequent layers. This so-called dense block structure enables the network to be thinner and compact, which lead to higher computational efficiency and memory efficiency. We refer the readers to [9] for the detailed architecture of DenseNet.

2.3 kU-Net Structure

kU-Net [7] is the combination of U-Net submodules, it was observed that human experts tend to first zoom out the image to determine the target object and then zoom in to obtain the accurate boundaries of the targets. The kU-Net structure contains two mechanisms which can simulate such human behaviors.

- kU-Net contains a sequence of submodule U-Nets to enable the information extraction carried at different scales sequentially.
- The information extracted by the submodule U-Net in a coarser scale will be propagated to the subsequent submodule U-Net to enable the feature extraction at a finer scale.

We refer the readers to [7] for the detailed structure of kU-Net.

3 Methods

3.1 Overview of the Proposed Network

k-Dense-UNet is the combination of Dense-UNet and kU-Net, an example of its architecture is shown in Fig. 2. It takes advantage of Dense-UNet's feature extraction and combines the idea of kU-Net to gradually extract the features to a finer scale. Similar to U-Net, the upsampling part of the submodule Dense-UNet is skip-connected to the subsequent Dense-UNet's max pooling part, which is equivalent to transferring the coarser information to the next sub-module to achieve more precise image segmentation result.

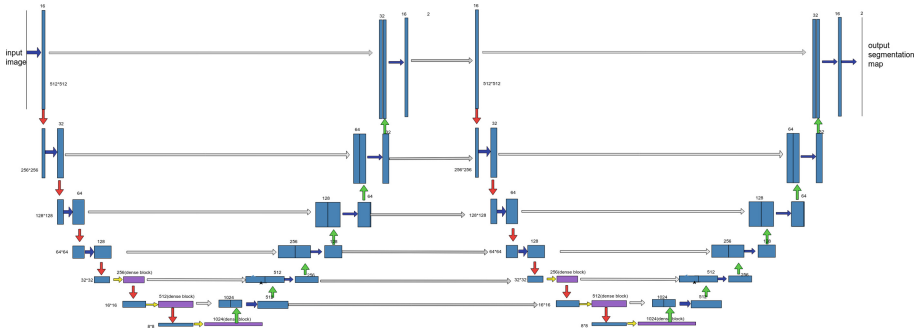

Fig. 2. Architecture of k-Dense-UNet(k = 2)

In practice, all the convolutional layers adopt 3×3 kernels with stride size as 1. For all the upsampling layers, 3×3 kernels are applied with stride size as 2. Activation functions are set as ReLUs. Batch normalization [16] is implemented to reduce over-fitting and increase the model's learning rate.

3.2 Dense-UNet Architecture

DenseNet has more diverse features and tend to have richer modes since each layer receives all of the previous layers as input: the so-called dense block structure. At the same time, because features of all complexity levels are used. DenseNet performs well when training data is insufficient.

Based on the above advantages of DenseNet, we embedded the dense block into U-Net to obtain more sufficient feature extraction and get a more precise segmentation map. This resulted in the Dense-UNet shown in Fig. 3. The gray arrow indicates the long skip connection between the max pooling layer and the corresponding upsampling layer, the red arrow indicates the 2×2 max pooling operation, the green arrow indicates the upsampling operation. It is worth noting that after the max pooling of the network, three dense blocks are embedded instead of the original two-convolution with batch normalization and ReLu activation function. These dense blocks are represented by purple rectangular blocks in the figure, and the corresponding operations are indicated by yellow arrows.

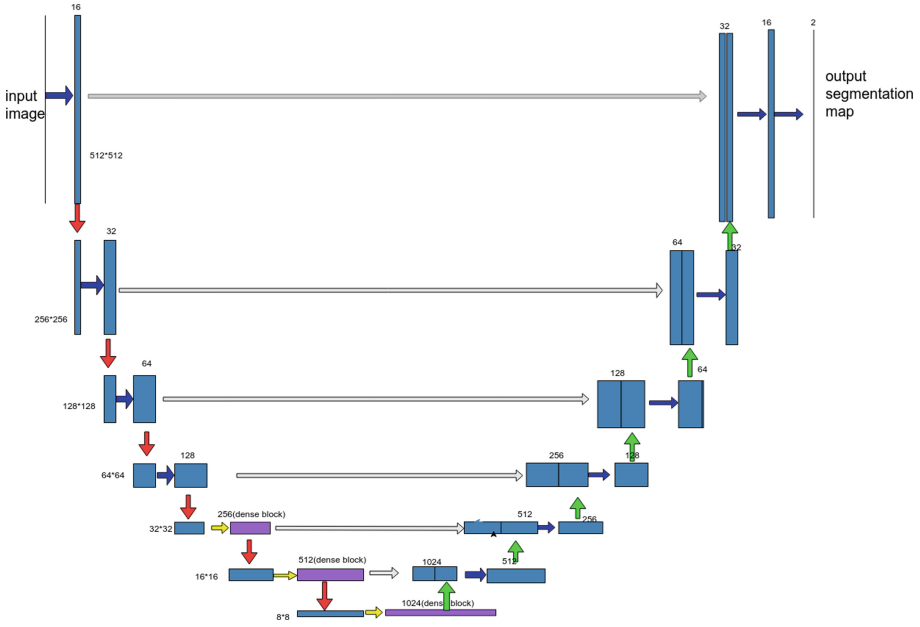

Fig. 3. Dense-UNet architecture (Color figure online)

Let's take the second dense block as an example to describe its operation shown in Fig. 4: Set the input x to obtain the output y through the dense operation. The dense operation is defined as: input through 128 1×1 convolution kernels, 128 3×3 Convolution kernel, 512 1×1 convolution kernels. The network is subject to batch regularization and ReLu activation functions to reduce

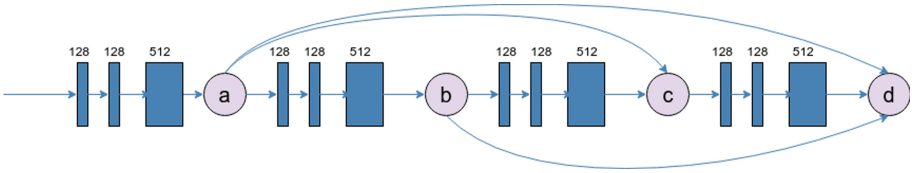

Fig. 4. Dense Block structure

gradient disappearance and over-fitting after each convolution operation. For the second dense block ($16 \times 16 \times 512$), the input is subjected to the dense operation to obtain the output a. a after the dense operation to get the output b, b after the dense operation and add a to get c, c after the dense operation and add a and b to get d, which is the final output.

The dense operation for the three dense blocks are shown in Table 1.

Table 1. Dense operation

Type	Dense operation	Times
Dense Block 1	$(1 \times 1, 64)(3 \times 3, 64)(1 \times 1, 64)$	3
Dense Block 2	$(1 \times 1, 128)(3 \times 3, 128)(1 \times 1, 512)$	4
Dense Block 3	$(1 \times 1, 256)(3 \times 3, 256)(1 \times 1, 1024)$	6

Similar to DenseNet, the k^{th} layer receives the feature-maps of all preceding layers, $x_0, \ldots, x_{k-1}$ as input:

$$x_k = H_k([x_0, x_1, \ldots, x_{k-1}])$$

where $[x_0, x_1, \ldots, x_{k-1}]$ refers to the concatenation of the feature-maps produced in layers $0, \ldots, k-1$.

It is worth noting that in order to reduce the complexity and size of dense blocks, a 1×1 convolution is added, and then a 3×3 convolution input is performed, which can greatly reduce the amount of calculation without damaging the accuracy of the model. This is also the design of the bottleneck layer of ResNet.

3.3 k-Dense-UNet Formation

In the k-Dense-UNet model, the internal operation of each sub-module is similar to Dense-UNet, and the dark blue arrow indicates the two-convolution with batch normalization and ReLu activation function, the red arrow represents the max pooling of 2×2 scale, and the green arrow represents the upsampling operation, the operation yellow arrow stands is consistent with that described in Dense-UNet, the grey arrow represents skip connections between the adjacent

submodules and within the submodules. The purple rectangular block represents the dense block, and its definition and implementation are the same as the Dense-UNet. The submodules consist of six downsampling steps followed by six upsampling steps. There are six skip connections between the adjacent submodules, corresponding to the long skip connection inside the submodules.

4 Experiments and Results

4.1 Dataset and Evaluation Metrics

We use the ISBI 2012 EM Segmentation Dataset to test the effectiveness of our model. Figure 1 shows an example of the dataset. The training part of the dataset contains 30 pairs of EM images and ground truth labels. The testing part contains 30 EM images without the ground truths.

The Evaluation metrics are the Foreground-restricted Rand Scoring after border thinning: V^{Rand}. The details of this metric can be found in [15].

4.2 Experiments on Loss Function

In order to compensate for the different frequency of pixels in a certain class form the training set, We use weighted loss function in all the experiments to force the network to learn the small borders between cells.

In this experiment, we compare three loss functions: weighted-bce, weighted-dice, and weighted dice & weighted-bce loss function. The model is U-Net. The training dataset is 30 pairs of EM images. As this dataset is small, we applied several data augmentation techniques to enlarge the dataset, which includes rotation, horizontal and vertical flip. We training the model using Adam optimizer with a learning rate of 2×10^{-4}. The training metric is IOU score. We also applied the EarlyStopping, ReduceLROnPlateau methods to improve the performance of the model.

The predicted segmentation images obtained form these loss functions are shown in Fig. 5. Among them, A, B, C are the result of the weighted dice loss function, weighted dice& weighted-bce loss function, weighted bce loss function respectively. D, E, F are the same enlarged part of A, B, C respectively.

We observed that the boundary of the predicted image obtained by weighted dice& weighted-bce loss function is more complete than the rest two. The result of the above models is shown in Table 2, which are sorted by V^{Rand}. Since

Table 2. Results of three types of loss functions sorted by V^{Rand}

Model	Loss function	V^{Rand}
U-Net	Weighted-dice	0.886397133
U-Net	Weighted-bce	0.941848881
U-Net	Weighted-bce& weighted-dice	**0.950320002**

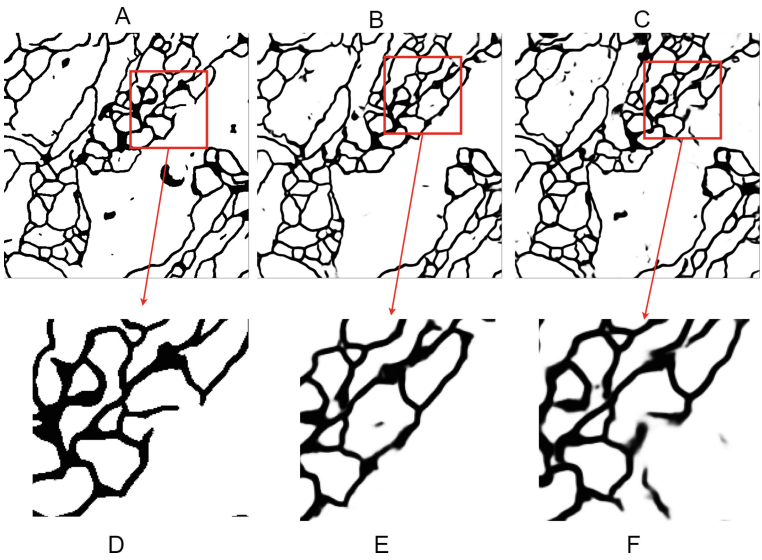

Fig. 5. Comparison of different loss function's result

weighted dice& weighted-bce loss function also has the highest V^{Rand} score, we use this loss function in the rest of the experiments.

4.3 Experiments on Differenet Backbones

This experiment first tested the V^{rand} of U-Net as the benchmark. Then, similar to Dense-UNet, we embedded ResNet50, ResNet101, ResNeXt, and SEResNet into U-Net as the backbone to test the V^{rand} of these networks.

The optimization function of the model is Adam. The training metric is IOU score. The loss function is weighted bce & weighted dice function. All models are trained for 20 epochs. Besides the data augmentation methods used in the previous experiment, we also adopted the elastic transformation method.

Table 3. Results of U-Net with different backbones sorted by V^{Rand}

Model	Backbone	V^{Rand}
U-Net	ResNet50	0.931212484
U-Net	SeresNet50	0.9429939
U-Net	ResNext50	0.948937035
U-Net	None	0.950320002
U-Net	ResNet101	**0.957619972**

The results of U-Net with different backbones are shown in Table 3. It can be seen that embedded ResNet101 as the backbone can achieve the best result among the five.

4.4 Ablation Study

We conduct the ablation study to evaluate the effectiveness of k-Dense-UNet, and the results are shown in Table 4. For this experiment, the optimization function is Adam. The training metric is IOU score. The loss function is weighted bce & weighted dice function. All models are trained for 20 epochs. This time we use the real-time data augmentation method. We define a phase which means 30 images has been processed, and one epoch contains 300 phases. It is worth noting that due to this method, pictures processed at each stage phase contains subtle differences, that is, each training image is unique.

The models involved in this experiment include U-Net, kU-Net (k = 3), kU-Net (k = 2), U-Net with ResNet101 as the backbone, Dense-UNet and k-Dense-UNet (k = 2). We can see a mild increase in performance with the embedding of the dense blocks. The performance if further significantly improved when embedding the dense block structure into kU-Net. We can summarize this experiment as follows:

- kU-Net structure can propagate coarser scales to subsequent modules to assist in finer feature extraction.
- Dense-UNet takes advantage of DenseNet's feature extraction capabilities which can achieve better results than U-Net and U-Net (ResNet101 as backend)
- The parameter k in kU-Net increases the input window size of the network exponentially. The smaller k value is sufficient to process many biomedical images (k = 2): the model of kU-Net (k = 3) is not as good as k = 2. The result might be that the model is too complicated, which lead to the network to a certain degree of over-fitting.

Table 4. Results of U-Net with different backbones sorted by V^{Rand}

Model	Backbone	V^{Rand}
U-Net	None	0.956101213
kU-Net (k = 3)	None	0.959437719
kU-Net (k = 2)	None	0.963030825
U-Net	ResNet101	0.963493141
Dense-UNet	DenseNet	**0.964117979**
k-Dense-UNet (k = 2)	DenseNet	**0.972352852**

5 Conclusion

In this paper, by embedding dense blocks into U-Net, we present Dense-UNet for biomedical image segmentation. It can obtain more sufficient feature extraction and get a more precise segmentation map compared to U-Net. Moreover, by integrating kU-Net and Dense-UNet, we proposed k-Dense-UNet, which takes advantage of Dense-UNet's feature extraction capabilities and combines the idea of kU-Net to gradually extract the features to a finer scale. By harnessing the short skip connection in the dense block, the long skip connection in the Dense-UNet submodules and the skip connection between the adjacent submodules, we can achieve more precise image segmentation maps. Experimental results on the ISBI 2012 EM dataset show that the proposed method can achieve better results compared to U-Net and some of its variants.

Acknowledgement. The first author would like to thank Yuze Guo for helpful discussions. This work was supported by the Zhi-Yuan Chair Professorship Start-up Grant (WF220103010), and Startup Fund (WF220403029) for Youngman Research, from Shanghai Jiao Tong University.

References

1. Ciresan, D., Giusti, A., Gambardella, L.M., Schmidhuber, J.: Deep neural networks segment neuronal membranes in electron microscopy images. In: NIPS (2012)
2. 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
3. Quan, T.M., Hildebrand, D.G.C., Jeong, W.-K.: FusioNnet: a deep fully residual convolutional neural network for image segmentation in connectomics, CoRR, vol. abs/1612.05360 (2016)
4. Chen, H., Qi, X., Cheng, J.-Z., Heng, P.-A.: Deep contextual networks for neuronal structure segmentation. In: AAAI (2016)
5. Chen, K., Zhu, D., Lu, J., Luo, Y.: An adversarial and densely dilated network for connectomes segmentation. Symmetry **10**, 467 (2018)
6. Drozdzal, M., et al.: Learning normalized inputs for iterative estimation in medical image segmentation. Med. Image Anal. **44**, 1–13 (2018)
7. Chen, J., Yang, L., Zhang, Y., Alber, M., Chen, D.Z.: Combining fully convolutional and recurrent neural networks for 3D biomedical image segmentation. In: NIPS (2016)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
9. Huang, G., Liu, Z., Van Der Maaten, L., et al.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4700–4708 (2017)
10. Shelhamer, E., Long, J., Darrell, T.: Fully convolutional networks for semantic segmentation. TPAMI (2017)
11. Shen, W., Wang, B., Jiang, Y., Wang, Y., Yuille, A.L.: Multi-stage multi-recursive-input fully convolutional networks for neuronal boundary detection. In: ICCV (2017)

12. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 630–645. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_38
13. Krizhevsky, A., Hinton, G.: Learning multiple layers of features from tiny images. Technical report, University of Toronto (2009)
14. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning. In: NIPS Workshop on Deep Learning and Unsupervised Feature Learning (2011)
15. Arganda-Carreras, I., et al.: Crowdsourcing the creation of image segmentation algorithms for connectomics. Front. Neuroanat. **9**, 142 (2015)
16. Ioffe, S., Szegedy, C.: Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
17. Xu, L.: Least MSE reconstruction for self-organization: (i)&(ii). In: Proceedings of 1991 International Joint Conference on Neural Networks, pp. 2363–2373 (1991)
18. Xu, L.: Least mean square error reconstruction principle for self-organizing neural-nets. Neural Netw. **6**(5), 627–648 (1993)
19. Xu, L.: An overview and perspectives on bidirectional intelligence: Lmser duality, double IA harmony, and causal computation. IEEE/CAA Journal of Automatica Sinica **6**(4), 865–893 (2019)