Save 50% with the IET Summer Sale

Use code SUMMER23 to save on over 650+ selected engineering and technology books.

*Discount only available on selected print books between 21 August to 15 September

The Institution of Engineering and Technology

Browse the sale titles

IET Image Processing

Special Section: AI-Powered 3D Vision

IET Journals
The Institution of Engineering and Technology

DenseUNet: densely connected UNet for electron microscopy image segmentation

ISSN 1751-9659
Received on 16th November 2019
Revised 16th January 2020
Accepted on 26th March 2020
E-First on 3rd September 2020
doi: 10.1049/iet-ipr.2019.1527
www.ietdl.org

Yue Cao[1,2], Shigang Liu[1,2], Yali Peng[1,2] ✉, Jun Li[3]

[1]*Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi'an 710062, People's Republic of China*
[2]*School of Computer Science, Shaanxi Normal University, Xi'an 710119, People's Republic of China*
[3]*School of Computer Science and Technology, Nanjing Normal University, Nanjing 210046, People's Republic of China*
✉ *E-mail: pengyl@snnu.edu.cn*

Abstract: Electron microscopy (EM) image segmentation plays an important role in computer-aided diagnosis of specific pathogens or disease. However, EM image segmentation is a laborious task and needs to impose experts knowledge, which can take up valuable time from research. Convolutional neural network (CNN)-based methods have been proposed for EM image segmentation and achieved considerable progress. Among those CNN-based methods, UNet is regarded as the state-of-the-art method. However, the UNet usually has millions of parameters to increase training difficulty and is limited by the issue of vanishing gradients. To address those problems, the authors present a novel highly parameter efficient method called DenseUNet, which is inspired by the approach that takes particular advantage of recent advances in both UNet and DenseNet. In addition, they successfully apply the weighted loss, which enables us to boost the performance of segmentation. They conduct several comparative experiments on the ISBI 2012 EM dataset. The experimental results show that their method can achieve state-of-the-art results on EM image segmentation without any further post-processing module or pre-training. Moreover, due to smart design of the model, their approach has much less parameters than currently published encoder–decoder architecture variants for this dataset.

1 Introduction

Biomedical images analysis [1–4] is a useful tool for researchers to quickly and accurately process a great mount of image data. There are a wide variety of applications for biomedical images, some of which include being able to distinguish between different neurons to identify specific pathogens or disease. These applications are especially useful in the healthcare sector and pharmaceutical industry. Electron microscopy (EM) images with extremely high resolution are an important tool of research in biomedical images analysis. As an example, EM image and its corresponding neuron membrane segmentation result is illustrated in Fig. 1. The neuron membrane segmentation labels are binary images, where white regions are the pixels of segmented objects (which correspond to neurons) and black regions are membranes. The neuron membrane segmentation result has contributed to distinguish individual neurons, which is an indispensable step to reconstruct neural circuits [6]. However, neurons with anisotropic shapes and membranes with various thickness increase the difficulties of neuron membrane segmentation greatly. Therefore, the requirement for the accuracy and effectiveness of algorithm of automatic segmentation has become increasingly higher [7–10].

Traditional automatic segmentation algorithms [11–14] with hand design are being replaced with approaches that use machine learning algorithm from manual ground truth. Following the significant improvement of deep neural networks [15–18] in many computer vision tasks, such as object detection [15, 19–21], action recognition [22–25], and image restoration [26–29]. With the help of convolutional neural network (CNN) strong ability to learn and extract features, CNN-based methods have shown great potential in image segmentation. CNN is typically realised by a contracting path built from convolutional and pooling layers, which can capture coarse semantic features. CNNs heavily lose the input resolution and spatial information through successive pooling layers.

To gain the output with the same resolution, encoder–decoder architecture [30, 31] adds an expanding path, which is a different contracting path where pooling operators are replaced by deconvolutional operators. Fully convolutional networks (FCNs) [31] introduce skip connections between the contracting path and the expanding path. Thus, upsampling feature-maps are summed with feature-maps skipped from the contracting path, spatial information from the contracting path is combined with the upsampling output. An encoder network (the contracting path) takes the input and output a feature-map. These feature-maps hold the information, the features that represents the input. The decoder network (the expanding path) takes the feature vector from the encoder network and gives the best closest match to the actual input or intended output. The expanding path recovers the input image spatial resolution at the output of the model and enables precise localisation. UNet [30] uses channel-wise concatenation, which is used to combine the feature maps from the contracting path and expanding path to replace element-wise summation adopted in FCN [31]. By reducing the resolution of the feature maps to increase the receptive field, the encoder–decoder network has advantages of high memory efficiency and low computational consumption. Moreover, the UNet [30] demonstrates superior semantic segmentation performance over plain CNN models because of its multi-scale architecture.

The classic UNet has not achieved satisfactory results on this task, because it mixes information of multi-scale features from the original EM images and enlarges the size of receptive field.

Inspired by UNet, several UNet architecture variants have been developed to improve learning performance. Residual network [32] has shown great ability at easing the training of very deep networks and making the weights of network sparser. Inspired by this idea, fully convolutional residual networks (FC-ResNet) [33] and FusionNet [34] use a residual block to modify FCN [31]. However, such encoder–decoder architecture variants usually have millions of parameters and their network depth is limited by the issue of vanishing gradients [35] when the signal back-propagates across many layers.

Recently, the significant network depth has been shown to be helpful for image classification [32]. DenseNet [36] directly connects each layer to every other layer in a feed-forward fashion, then the network will be more accurate and easier to train.

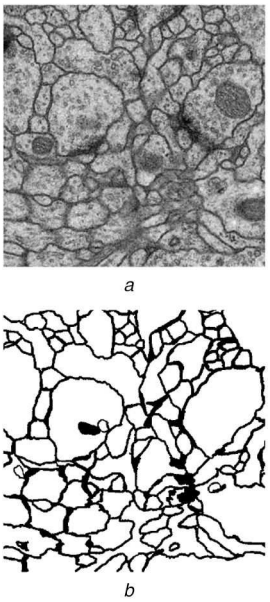


Fig. 1 *An example EM image and its neuron membrane segmentation result from ISBI 2012 EM dataset [5]*
(a) EM image, ***(b)*** Corresponding segmentation result

DenseNets are built from dense block (DB), where each DB is an iterative concatenation of previous feature maps. If one DB has L layers, this DB will have $(L(L+1))/2$ direct connections.

In this study, we propose densely connected UNet (DenseUNet) to use DB to modify UNet. The small modification has some interesting implications: (i) DenseUNet reduces the number of parameters and are more efficient in the parameters usage; (ii) DenseUNet alleviates the vanishing-gradient problem [35]; (iii) DenseUNet encourages feature reuse and strengthen feature propagation, the feature maps of all preceding layers are used as inputs and its own feature maps are used as inputs in all subsequent layers, which makes it easy to reuse the information from previously computed feature maps; (iv) dense connections have a regularising effect so DenseUNet reduces overfitting on this EM image segmentation task with small training set sizes; (v) by applying dense connections, DenseUNet decreases the amount of parameters compared to the other UNet variant architecture without reduction of network depth. In addition, we propose a weighted loss to make our architecture a very good fit for image segmentation. The details of DenseUNet architecture are shown in Fig. 2.

The main contributions of this study can be summarised as follows:

- We extend the DenseNet to UNet for EM image segmentation to get segmentation result accurately and overcome the vanishing gradient problem.
- We propose a weighted loss to improve the objective and subjective quality of segmentation result images effectively.
- DenseUNet achieves promising results on the publicly available ISBI 2012 EM dataset [5] without any further post-processing module or pre-training.

2 Related work

In this section, some algorithms about EM image segmentation will be briefly introduced, which includes traditional and deep learning-based methods..

Early automatic EM image segmentation methods include watershed [37] and hierarchical clustering [38–41], which rely on post-processing to drive the segmentation, but those methods cannot obtain satisfactory segmentation results. By building a conditional random field, Optree-idsia [37] selects a collection of nodes from the watershed merging tree to enforce structure consistency to segmentation outputs. Nevertheless, accurate EM image segmentation is still a challenging graph cut algorithm [42–44] for EM image segmentation methods are direct and short time-consuming in the testing stage. However, those algorithms are complex and depend much on experts' knowledge.

Deep learning techniques take advantage of a lot of training data directly to learn membrane boundaries. Automated EM image segmentation based on deep learning can effectively improve the quantitative analysis of disease. Ciresan *et al.* [45] proposed a deep neural network as a solution to the EM image segmentation problem and won first place in ISBI 2012 EM challenge. Long short-term memory (LSTM) networks are recurrent neural networks, which are initially designed for sequence processing. They achieved state-of-the-art results on challenging tasks such as handwriting recognition, large vocabulary speech recognition, and machine translation. PyraMiD-LSTM [46] use LSTM to retrieve contextual information. Nevertheless, the segmentation performance of those methods is always limited by the depth of networks. In recent years, the generative adversarial network (GAN) [47] has been more and more popular and introduced in various vision tasks. The adversarial loss [47] has successfully been applied to produce realistic images. The adversarial and densely dilated network (ADDN) [48] combines the GAN objective with the dice loss to alleviate the blurry effects. However, ADDN [48], which uses two networks (generative network and discriminator network), consumes so much computing power. To extract feature better, CUMedVision [49] fuses multi-level contextual information to segment the membrane to handle the large variation of size. Inspired by CUMedVision [49], Shen *et al.* [50] presented a multi-stage multi-recursive input FCNs to train network divide into multiple sequential stages. Furthermore, those deep neural networks are limited by memory and have the shortcoming of high computation cost.

Those methods based on encoder–decoder architecture are presented to reduce the computing complexity and memory consumption [30, 31]. FCN [31] was proposed to solve the end-to-end semantic segmentation problem. By increasing the connectivity within FCN [31], FC-ResNet [33] and FusionNet [34] achieve well segmentation result. PolyMtl [51] binds FCN and FC-ResNet [33] together, FCN is as pre-processor for normalising images and FC-ResNet [33] generates segmentation prediction. However, those encoder–decoder architecture variants are limited by spending a lot of time adjusting parameters and the issue of vanishing gradients [35].

In the study, we propose a novel highly parameter efficient network using dense connectivity and weighted loss to improve segmentation accuracy. In contrast to aforementioned encoder–decoder architecture variants, we design a UNet-based network whose convolutional layers are replaced by DBs to overcome the vanishing gradient problem [35].

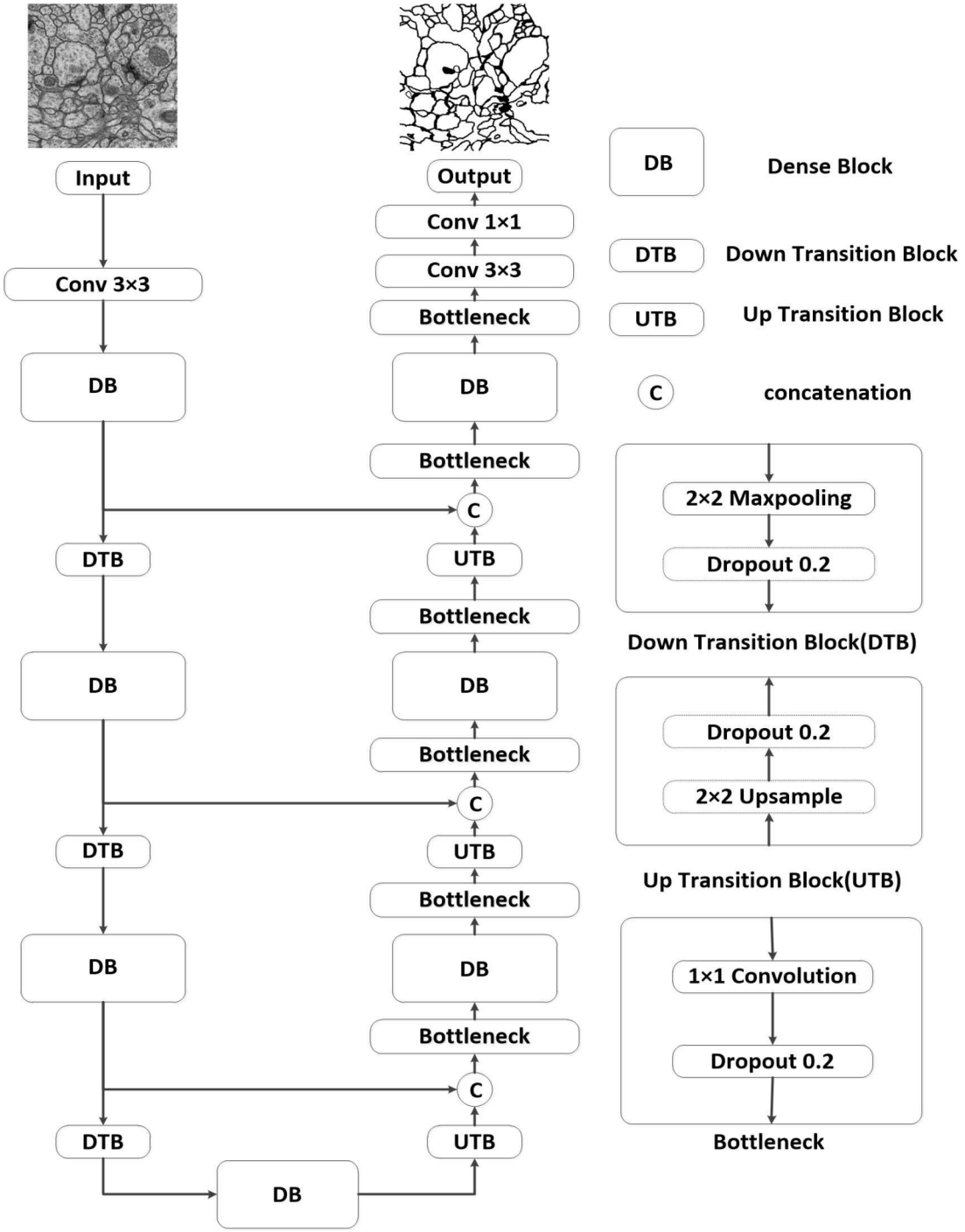


Fig. 2 *The architecture of the proposed network. Our architecture is built from DB. Our architecture is composed of a contracting path with three down transitions block (DTB) and an expanding path with three up transitions block (UTB). A circle represents concatenation and arrows represent connectivity patterns in the network*

3 Proposed method

In the section, we introduce the proposed method in detail. Fig. 3 shows the basic block DB in our architecture.

Fig. 2 shows the architecture of the proposed DenseUNet, which is made up of a contracting path and an expanding path. The core blocks of the proposed network consists of four types: DTB, UTB, bottleneck, and DB. Firstly, we start by explaining the design of DB. Then, the detailed architecture of each component is elaborated. We, respectively, add the dropout layer [52] in DTB, UTB and bottleneck. Overfitting is a possibility as in deep convolutional neural network. The dropout layer [52] is used to prevent the overfitting problem and improve the segmentation accuracy. In addition to the DB, the bottleneck also can reduce the parameter size in DenseUNet without segmentation performance decline. Finally, we introduce the weighted loss.

3.1 Dense block (DB)

We apply the DB to the network to solve the vanishing gradient problem [53]. Each feature map is connected by dense connectivity to drive our model to reuse the information of previous feature maps.

We denote x_1 as the input of DB. One DB comprises L layers, each of which implements a non-linear transformation $F_l(\cdot)$, where l indexes the layer. $F_l(\cdot)$ is implemented by 3×3 convolution layers followed by a rectified linear unit in our network. x_l denotes input and y_l denotes the output for the lth layer, respectively, which can be expressed by

$$y_l = F_l(x_l) \tag{1}$$

where x_1 is the input of the first layer ($l = 1$). The input x_l of the lth layer (if $l \geq 2$) in DB can be described by

$$x_l = [y_{l-1}, \ldots, y_1, x_1] \tag{2}$$

where $[, \ldots,]$ represents the concatenation operation. DB encourages the reuse of features and makes all layers in the architecture receive a direct supervision signal. If each function $F_l(\cdot)$ produces g feature maps, it follows that the lth layer has $n + g \times (l-1)$ input feature maps. As mentioned in Fig. 3, we set $g = n/4$ and $L = 4$ in our DB, so the output of DB has $2n$ feature-maps. By applying DB, we can improve the information flow so that our proposed network can solve the vanishing-gradient problem [53] and get high segmentation accurate.

3.2 Architecture details

In the section, we introduce the proposed DenseUNet in detail. In Table 1, we can see the detailed DenseUNet used in our experiments.

As the EM image is the input image, the proposed network first executes a 3×3 convolution operation to generate feature maps for the DB. As mentioned in Table 1, the initial convolution layer generates 32 feature maps.

Next, in the contracting path, the output feature maps of the DB are down-sampled by a downsampling factor of two in the DTB. The DTB consists of two layers: 2×2 max pooling layer with stride 2 and a dropout layer with a dropout rate of 0.2. Thus, the output feature maps of DTB are half in size than the input feature maps with the same number of channels of the input feature maps. Our proposed network follows the above procedure three times along the contracting path, which causes our network to $2^3 = 8$ times the number of feature maps.

After the operations of one DB, our proposed network follows the expanding path, which is the inverse process of the contracting path. In the expanding path, the UTB up-samples the output feature-maps by an upsampling factor of two. The UTB consists of two layers: a 2×2 upsampling layer and a dropout layer with a dropout rate of 0.2. Unlike UNet [30], which uses a 2×2 deconvolution layer, the upsampling layer uses the nearest neighbour interpolation. When upsampling the feature maps, UTB doubles the size of the feature maps, with two times as many as number of channels of the input feature maps. After the operation of one UTB, we use one concatenation operation, which connects the input of DTB to the output of the UTB in different levels. Building a great number of paths would lead to a computationally intractable number of feature maps. To reduce computation and the feature-maps explosion in the expanding path of the network, we introduce bottleneck both before and after the DB to reduce the dimensionality of the feature maps. Bottleneck is composed of a 1×1 convolution followed by a dropout operation with a dropout rate of 0.2. Our network follows the above procedure three times along with the expanding path, which is similar to the contracting path.

After the expanding path, the proposed network computes the penult 3×3 convolution and the final 1×1 convolution followed by sigmoid to generate the final segmentation output.

3.3 Weighted loss

We experimented with both binary cross-entropy and weighted loss functions. $q_i \in [0, 1]$ is the ith output of the last network layer passed through a sigmoid non-linearity and $y_i \in \{0, 1\}$ is the corresponding label. The binary cross-entropy is then defined as follows:

$$L(q, y) = \sum_{i=1} y_i \log(q_i) + (1 - y_i)\log(1 - q_i) \tag{3}$$

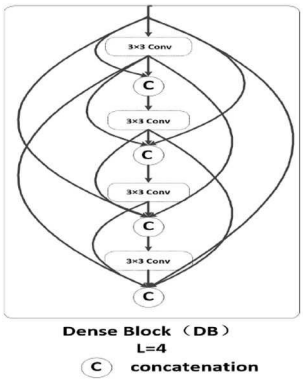


Fig. 3 *Architecture of DB ($L = 4$)*

Table 1 Detailed DenseUNet used in the experiments

Block type	Detail	Output resolution	Output channel number
input		512×512	1
Conv 3×3		512×512	32
DB	$L = 4$, $g = 8$	512×512	64
DTB		256×256	64
DB	$L = 4$, $g = 16$	256×256	128
DTB		128×128	128
DB	$L = 4$, $g = 32$	128×128	256
DTB		64×64	256
DB	$L = 4$, $g = 64$	64×64	512
UTB		128×128	256
concatenation		128×128	512
bottleneck		128×128	256
DB	$L = 4$, $g = 64$	128×128	512
bottleneck		128×128	256
UTB		256×156	126
concatenation		256×256	256
bottleneck		256×256	128
DB	$L = 4$, $g = 32$	256×256	256
nottleneck		256×256	128
UTB		512×512	64
concatenation		512×512	128
bottleneck		512×512	64
DB	$L = 4$, $g = 16$	512×512	128
bottleneck		512×512	64
Conv 3×3		512×512	2
Conv 1×1	sigmoid	512×512	1

The weighted loss is (see (4)) , where parameter m controls the weight. Compared with the binary cross-entropy loss functions, the weight of segmented objects do not change, but the weighted loss gives the remaining objects, which correspond mostly with membranes having a higher weight. In subsequent experiments, we set $m = 5$ in weighted loss.

4 Experiments

4.1 ISBI 2012 EM dataset

We evaluate the performance of the proposed DenseUNet to EM image segmentation in ISBI 2012 Challenge dataset [5]. To accelerate the research in automatic segmentation of neural structures, IEEE International Symposium on Biomedical Imaging (ISBI) launched a challenge for segmenting neuronal structures in EM image stacks. ISBI 2012 EM dataset provides EM images with

$$L(q, y) = \sum_{i=1} [(1 - y_i)(m - 1) + 1][y_i \log(q_i) + (1 - y_i)\log(1 - q_i)] \tag{4}$$

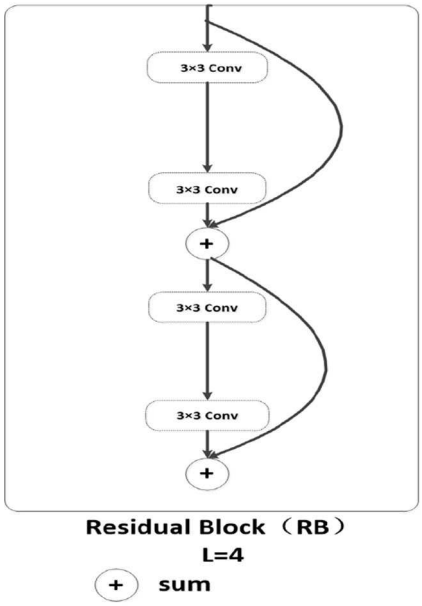


Fig. 4 *Architecture of residual block (L = 4)*

sufficient details for identification of individual neurons. The training data consists of 30 images (512 × 512 pixels) assembled from the EM of the Drosophila first instar larva ventral nerve cord with corresponding binary labels. The test set is another set of 30 images, but corresponding labels are not provided.

4.2 Experimental setup

The DenseUNet is implemented using Keras open-source deep learning library. Training and deployment of the network are conducted on a PC equipped with an Intel i7 CPU with a 32 GB main memory and an NVIDIA GTX Geforce 1080Ti GPU. We train DenseUNet with RMSprop, with an initial learning rate of 10^{-3} and an exponential decay of 0.995 after each epoch. The network is initialised using Xavier method. For all experiments, we fine tune our models with full size images and a learning rate of 10^{-4}.

4.3 Data augmentation

It is a cumbersome and time-consuming task to get enough training data. Thus, for EM images and the corresponding ground-truth neuron membrane segmentation images, we need to use data augmentation to obtain massive training data. We rotate the EM images by 90°, 180°, and 270°, respectively, and we take the reflection operation of those four sets to augment training data. Besides the basic rotation and reflection operation, we also augmented the data using elastic distortions, shearing, mirroring, and perspective skewing. We used full resolution (512 × 512) images as input without applying random cropping for data augmentation.

4.4 Evaluation metric

We adopt the standard metric maximal foreground-restricted rand score after thinning (V_{rand}) described in [5] to evaluate segmentation quality. We denote S as the segmentation result and T as the annotated images. Here p_{ij} denotes the joint probability that a pixel belongs to segment i in S and segment j in T

$$V_{\text{split}}^{\text{Rand}} = \frac{\sum_{ij} p_{ij}^2}{\sum_j (\sum_i p_{ij})^2} \tag{5}$$

$$V_{\text{merge}}^{\text{Rand}} = \frac{\sum_{ij} p_{ij}^2}{\sum_i (\sum_j p_{ij})^2} \tag{6}$$

Table 2 Ablation study on ISBI 2012 EM dataset

Method	V_{rand}
DenseUNet	0.983
DenseFCN	0.975
ResidualUNet	0.974

Foreground-restricted Rand F-score is introduced by the official ranking system. By calculating $V_{\text{split}}^{\text{Rand}}$ and $V_{\text{merge}}^{\text{Rand}}$ as (5) and (6), respectively, V_{rand} can be obtained

$$V_{\text{rand}} = \frac{2V_{\text{split}}^{\text{Rand}} V_{\text{merge}}^{\text{Rand}}}{V_{\text{split}}^{\text{Rand}} + V_{\text{merge}}^{\text{Rand}}} \tag{7}$$

4.5 Ablation study

To explore the effectiveness of different modules, we perform the following ablation studies.

4.5.1 Comparison with different basic blocks and connections: Compared with the proposed DB, the residual block is shown in Fig. 4, which is used in our ablation studies.

The residual block in ResNets [32], which adds the identity mapping of the input to the output of a layer, consists of a non-linear unit and a shortcut identity mapping. The input x_{l+1} of the $(l+1)$th layer (if $l \geq 2$) in the residual block can be expressed with the following formula:

$$x_{l+1} = x_l + y_l \tag{8}$$

To validate the effectiveness of DenseUNet, we train three different networks, including: DenseUNet, DenseFCN, and ResidualUNet. The quantitative segmentation results in ISBI 2012 EM dataset are summarised in Table 2.

Hindering during the training procedure completely, the vanishing gradient problem is a critical issue restricting segmentation performance. Compared with ResidualUNet using skip connection to solve this problem, DenseUNet connects each layer to all the other layers and reuse previous feature-maps in the proposed DB. By solving the vanishing gradient problem more effectively, DenseUNet has better segmentation results than ResidualUNet.

Compared with DenseFCN adopting element-wise summation operation, DenseUNet uses the concatenation operation to reuse the features matching resolution levels from the contracting path to the expanding path. Minimising information loss caused by downsampling operation, DenseUNet has better segmentation performance than DenseFCN.

4.5.2 Comparison with different loss: To verify the effectiveness of the weighted loss, we compare results by DenseUNet with weighted loss and DenseUNet with cross-entropy.

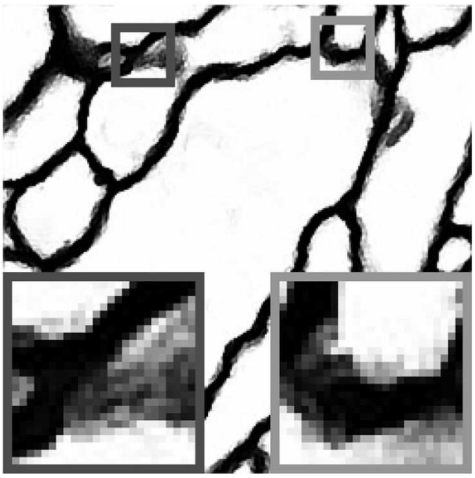
To better illustrate the effectiveness of the weighted loss, the visual effect of segmentation results is shown in Fig. 5. We can see that the improvement by the weighted loss is big. From Fig. 5, the weighted loss encourage thick membrane boundary and suppress the blurry artefacts of neural boundary. With the help of weighted loss, the amount of oversegmentation is reduced.

To quantitative exhibit the effectiveness of the weighted loss, the segmentation results in ISBI 2012 EM dataset are summarised in Table 3. In summary, the weighted loss can improve the accuracy of segmentation and gain good visual effect.

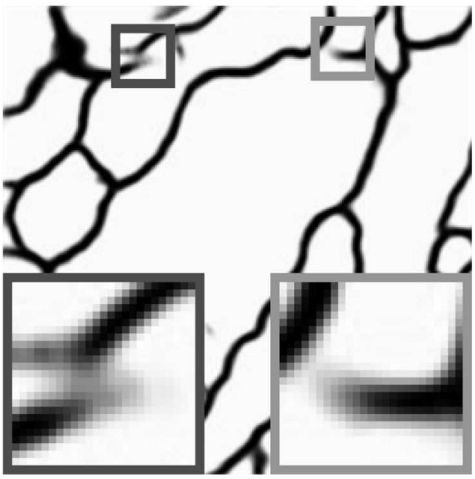
4.6 Comparison with state-of-the-art approaches

In Table 4, we compare our DenseUNet with existing published entries on ISBI 2012 EM dataset.

The performance of different submitted methods is reported on the leader board for everyone to see. Compared with the results of previous methods on ISBI 2012 EM dataset, segmentation results provided by DenseUNet is competitive. Existing plain CNN-based



DenseUNet with weighted loss



DenseUNet with cross-entropy

Fig. 5 *DenseUNet with weighted loss and DenseUNet with cross-entropy*

Table 3 Comparison with different loss on ISBI 2012 EM dataset

Method	V_{rand}
DenseUNet with weighted loss	0.983
DenseUNet with cross-entropy	0.978

methods for EM image segmentation [45, 55] do not make full use of multi-scale features from the original EM images and are limited in their depth and the size of receptive field, thereby achieving relatively low segmentation accuracy.

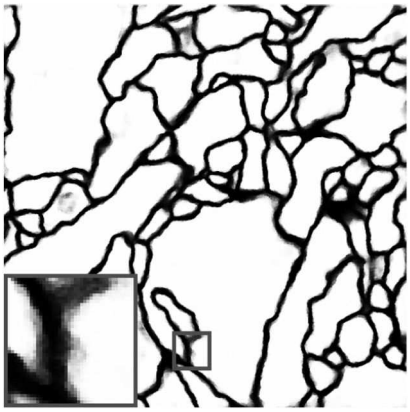
By exploiting multi-scale features fully and increasing the receptive field, those methods adopting the encoder–decoder architecture improve the segmentation accuracy. We compare our DenseUNet with other methods based on encoder–decoder architecture in Section 4.7.

In Figs. 6 and 7, we compare the visual effect with PyraMiD-LSTM [46] and FC-ResNet [33]. Although EM images contain some noise and small image alignment errors, our segmentation results get clear and smooth in the neuron regions and preserve good boundaries. PyraMiD-LSTM [46] is subjected to artefacts in the neuron regions. FC-ResNet [33] gains discontinuous and rough boundary in segmentation results, which might be attributed to overfitting or inappropriate loss function.

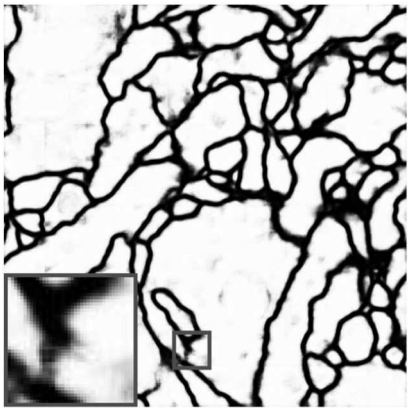
In summary, the proposed DenseUNet applies the encoder–decoder structure of the modified UNet, which enables our network to efficiently use limited memory to store information of feature maps and extract multi-scale features fully. Thereby, DenseUNet exhibits better results than other methods.

Table 4 Comparison with state-of-the-art approaches

Method	V_{rand}
DenseUNet	0.9834
ADDN [48]	0.9832
PolyMtl [51]	0.9806
M2FCN [50]	0.9780
FusionNet [34]	0.9780
Masters [54]	0.9771
CUMedVision [49]	0.9768
UNet [30]	0.9728
motif [55]	0.9725
Liu's method [56]	0.9714
Optree-idsia [37]	0.9699
Ciresan's method [45]	0.9690
FC-ResNet [33]	0.9690
PyraMiD-LSTM [46]	0.9677



DenseUNet



PyraMiD-LSTM

Fig. 6 *Comparison with PyraMiD-LSTM [46]*

4.7 Comparison of other methods based on encoder–decoder architecture

The details can be seen in Table 5, whereas other modified encoder–decoder architecture methods enlarge the receptive field of their networks while reducing their computational complexity and enable their networks to efficiently use limited memory [30, 31], limitation that the training can be stuck in a local minimum remained unsolved. While retaining the benefits of encoder–decoder architecture, DenseUNet overcomes the limitation of the conventional methods and exceeds the performance of state-of-the-art EM image segmentation solutions.

Moreover, other approaches rely much on post-processing and pre-processing to boost their performance; some methods averaged several trained models to promote the ranking. Those details can also be seen in the third, fourth, and fifth column in Table 5.

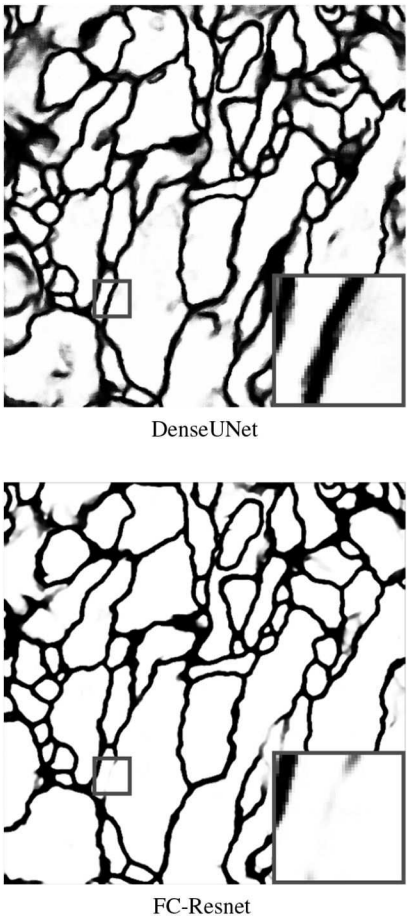


Fig. 7 *Comparison with FC-ResNet [33]*

Table 5 Comparison with other methods based on encoder–decoder architecture published entries on ISBI 2012 EM dataset

Method	V_{rand}	Post-processing	Pre-processing	Average over	Parameters [*M*]
DenseUNet	0.983	no	no	1	4
PolyMtl [51]	0.981	no	no	10	13
FusionNet [34]	0.978	yes	yes	8	31
CUMedVision [49]	0.977	yes	no	6	8
UNet [30]	0.973	no	yes	7	33
FC-ResNet [33]	0.969	no	yes	1	11

Compared these modified encoder–decoder architectures by taking the advantage of dense connection, DenseUNet does not use any further post-processing module nor pre-training module.

From the sixth column in Table 5, those encoder–decoder architecture variants have a larger number of parameters, which costs a lot of memory and time for parameter adjustment. The final actual amount of parameters we use is much smaller than other modified encoder–decoder architecture methods. This is mainly because the network uses a lot of bottleneck with a 1×1 convolution instead of the previous 3×3 convolution other methods use. In addition, those encoder–decoder architecture variants can suffer from the vanishing gradient problem [53]. We apply DB to our network to successfully resolve those problems and our network use the minimum amount parameters.

5 Conclusion

In this study, a novel encoder–decoder architecture-based EM) image segmentation method with weighted loss is presented. Experiments on the ISBI 2012 EM dataset and training dataset demonstrate that the presented methods could enhance segmentation accuracy of the segmentation results. Compared with the modified encoder–decoder architecture on the ISBI 2012 EM dataset, our method could provide competitive segmentation results without any further post-processing module nor pre-training. It should be mentioned that the presented method uses the minimum number of parameters. Therefore, our DenseUNet can avoid redundant parameters and is a lightweight network structure. With the help of weighted loss, blurry segmentation images with artefacts will be avoided. In the future, we will extent the presented method into 3D EM image segmentation.

6 Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61873155, 61702313, 61672333, and 61703096), the National Natural Science Foundation of Shaanxi Province (no. 2018JM6050), the Natural Science Foundation of Jiangsu Province (no. BK20170691), the Transfer and Promotion Plan of Scientific and Technological Achievements of Shaanxi Province (no. 2019CGXNG-019), the Innovation Chain of Key Industries of Shaanxi Province (no. 2019ZDLSF07-01), the Key Science and Technology Program of Shaanxi Province, China (no. 2016GY-081).

7 References

[1] Peng, Y., Liu, S., Qiang, Y., *et al.*: 'A local mean and variance active contour model for biomedical image segmentation', *J. Comput. Sci.*, 2019, **33**, pp. 11–19

[2] Liu, S., Peng, Y., Ben, X., *et al.*: 'A novel label learning algorithm for face recognition', *Signal Process.*, 2016, **124**, pp. 141–146

[3] Li, K., Yang, J., Liu, L., *et al.*: 'SPA: sparse photorealistic animation using a single RGB-D camera', *IEEE Trans. Circuits Syst. Video Technol.*, 2017, **27**, (4), pp. 771–783

[4] Peng, Y., Li, L., Liu, S., *et al.*: 'Space-frequency domain based joint dictionary learning and collaborative representation for face recognition', *Signal Process.*, 2018, **147**, pp. 101–109

[5] Ignacio, A., Turaga, S.C., Berger, D.R., *et al.*: 'Crowdsourcing the creation of image segmentation algorithms for connectomics', *Front. Neuroanatomy*, 2015, **9**, pp. 142–150

[6] Jurrus, E., Paiva, A.R.C., Watanabe, S., *et al.*: 'Detection of neuron membranes in electron microscopy images using a serial neural network architecture', *Med. Image Anal.*, 2010, **14**, (6), pp. 770–783

[7] Du, B., Wei, Q., Liu, R.: 'An improved quantum-behaved particle swarm optimization for endmember extraction', *IEEE Trans. Geosci. Remote Sens.*, 2019, **57**, (8), pp. 6003–6017

[8] Tu, B., Zhang, X., Kang, X., *et al.*: 'Spatial density peak clustering for hyperspectral image classification with noisy labels', *IEEE Trans. Geosci. Remote Sens.*, 2019, **57**, (7), pp. 5085–5097

[9] Liu, S., Peng, Y.: 'A local region-based chan-vese model for image segmentation', *Pattern Recognit.*, 2012, **45**, (7), pp. 2769–2779

[10] Peng, Y., Liu, S., Wang, X., *et al.*: 'Joint local constraint and fisher discrimination based dictionary learning for image classification', *Nuerocomputing*, 2019, 2020, **398**, pp. 505–519

[11] Yang, Y., Li, B., Li, P., *et al.*: 'A two-stage clustering based 3D visual saliency model for dynamic scenarios', *IEEE Trans. Multimed.*, 2019, **21**, (4), pp. 809–820

[12] Li, K., Yang, J., Lai, Y., *et al.*: 'Robust non-rigid registration with reweighted position and transformation sparsity', *IEEE Trans. Vis. Comput. Graphics*, 2019, **25**, (6), pp. 2255–2269

[13] Peng, Y., Zhang, L., Liu, S., *et al.*: 'Kernel negative ϵ dragging linear regression for pattern classification', *Complexity*, 2017, **2017**, pp. 2691474:1–2691474:14

[14] Tu, B., Huang, S., Fang, L., *et al.*: 'Hyperspectral image classification via weighted joint nearest neighbor and sparse representation', *IEEE J. Sel. Top. Appl. Earth Obser. Remote Sens.*, 2018, **11**, (11), pp. 4063–4075

[15] Du, B., Ru, L., Wu, C., *et al.*: 'Unsupervised deep slow feature analysis for change detection in multi-temporal remote sensing images', *IEEE Trans. Geosci. Remote Sens.*, 2019, **57**, (12), pp. 9976–9992

[16] Gong, C., Liu, T., Tao, D., *et al.*: 'Deformed graph Laplacian for semisupervised learning', *IEEE Trans. Neural Netw. Learn. Syst.*, 2015, **26**, (10), pp. 2261–2274

[17] Yang, Y., Liu, Q., He, X., *et al.*: 'Cross-view multi-lateral filter for compressed multi-view depth video', *IEEE Trans. Image Process.*, 2019, **28**, (1), pp. 302–315

[18] Du, B., Tang, X., Wang, Z., *et al.*: 'Robust graph-based semisupervised learning for noisy labeled data via maximum correntropy criterion', *IEEE Trans. Cybern.*, 2019, **49**, (4), pp. 1440–1453

[19] Liu, H., Xu, B., Lu, D., *et al.*: 'A path planning approach for crowd evacuation in buildings based on improved artificial bee colony algorithm', *Appl. Soft Comput.*, 2018, **68**, pp. 360–376

[20] Liu, J., Wang, C., Su, H., *et al.*: 'Multistage GAN for Fabric Defect Detection', ', *IEEE Trans. Image Process.*, 2020, **29**, 3388–3400

[21] Yang, J., Guo, J., Yue, H., *et al.*: 'CdNet: CNN-based cloud detection for remote sensing imagery', *IEEE Trans. Geosci. Remote Sens.*, 2019, **57**, (8), pp. 6195–6211

[22] Liu, H., Liu, B., Zhang, H., *et al.*: 'Crowd evacuation simulation approach based on navigation knowledge and two-layer control mechanism', *Inf. Sci.*, 2018, **436-437**, pp. 247–267

[23] Yang, J., Zhu, Y., Li, K., *et al.*: 'Tensor completion from structurally-missing entries by low-tt-rankness and fiber-wise sparsity', *IEEE J. Sel. Top. Signal Process.*, 2018, **12**, (6), pp. 1420–1434

[24] Yu, J., Zhu, C., Zhang, J., *et al.*: 'Spatial pyramid-enhanced netvlad with and weighted triplet loss for place recognition', *IEEE Trans. Neural Netw. Learn. Syst.*, 2020, **31**, (2), pp. 661–674

[25] Yang, W., Zhou, L., Li, T., *et al.*: 'A face detection method based on cascade convolutional neural network', *Multimedia Tools Appl.*, 2019, **78**, (17), pp. 24373–24390

[26] Zhang, K., Zuo, W., Zhang, L.: 'Deep plug-and-play super-resolution for arbitrary blur kernels'. IEEE Conf. on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019, pp. 1671–1681

[27] Peng, Y., Zhang, L., Liu, S., *et al.*: 'Dilated residual networks with symmetric skip connection for image denoising', *Neurocomputing*, 2019, **345**, pp. 67–76

[28] Liu, P., Zhang, H., Zhang, K., *et al.*: 'Multi-level wavelet-CNN for image restoration'. IEEE Conf. on Computer Vision and Pattern Recognition Workshops, Salt Lake City, UT, USA, 18–22 June 2018, pp. 773–782

[29] Zhang, K., Zuo, W., Zhang, L.: 'FFDNet: toward a fast and flexible solution for CNN-based image denoising', *IEEE Trans. Image Process.*, 2018, **27**, (9), pp. 4608–4622

[30] Ronneberger, O., Fischer, P., Brox, T.: 'U-Net: convolutional networks for biomedical image segmentation'. Part III Proc. 18th Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI 2015), Munich, Germany, 5–9 October 2015, pp. 234–241

[31] Long, J., Shelhamer, E., Darrell, T.: 'Fully convolutional networks for semantic segmentation', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014, **39**, (4), pp. 640–651

[32] He, K., Zhang, X., Ren, S., *et al.*: 'Deep residual learning for image recognition'. 2016 IEEE Conf. on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016, pp. 770–778

[33] Drozdzal, M., Vorontsov, E., Chartrand, G., *et al.*: 'The importance of skip connections in biomedical image segmentation'. Proc. Deep Learning and Data Labeling for Medical Applications – First Int. Workshop (LABELS 2016), and Second Int. Workshop (DLMIA 2016), Held in Conjunction with MICCAI 2016, Athens, Greece, 21 October 2016, pp. 179–187

[34] Quan, T.M., Hildebrand, D.G.C., Jeong, W.: 'FusionNet: a deep fully residual convolutional neural network for image segmentation in connectomics'. CoRR, 2016, abs/1612.05360

[35] Liu, P., Zhang, H., Lian, W., *et al.*: 'Multi-level wavelet convolutional neural networks', *IEEE Access*, 2019, **7**, pp. 74973–74985

[36] Huang, G., Liu, Z., van der Maaten, L., *et al.*: 'Densely connected convolutional networks'. 2017 IEEE Conf. on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017, pp. 2261–2269

[37] Uzunbaş, M.G., Chen, C., Metaxsas, D.: 'Optree: a learning-based adaptive watershed algorithm for neuron segmentation'. Int. Conf. on Medical Image Computing and Computer-Assisted Intervention, Boston, MA, USA, September 2014, pp. 97–105

[38] Peng, Y., Li, L., Liu, S., *et al.*: 'Weighted constraint based dictionary learning for image classification', *Pattern Recognit. Lett.*, 2020, **130**, pp. 99–106

[39] Zhang, C., Fu, H., Hu, Q., *et al.*: 'Generalized Latent Multi-View Subspace Clustering', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020, **42**, (1), pp. 86–99

[40] Liu, S., Li, L., Jin, M., *et al.*: 'An optimized coefficient vector and representation based classification methods for face recognition', *IEEE Access*, 2019, **8**, pp. 8668–8674

[41] Tu, B., Zhou, C., He, D., *et al.*: 'Hyperspectral classification with noisy label detection via superpixel-to-pixel weighting distance', *IEEE Trans. Geosci. Remote Sens.*, 2020, **58**, (6), pp. 4116–4131

[42] Yang, J., Gan, Z., Li, K., *et al.*: 'Graph-based segmentation for RGB-D data using 3-d geometry enhanced superpixels', *IEEE Trans. Cybern.*, 2015, **45**, (5), pp. 913–926

[43] Fu, S., Liu, W., Tao, D., *et al.*: 'HESGCN: Hessian graph convolutional networks for semi-supervised classification', *Inf. Sci.*, 2020, **514**, pp. 484–498

[44] Liu, S., Peng, Y., Sun, Z., *et al.*: 'Self-calibration of projective camera based on trajectory basis', *J. Comput. Sci.*, 2019, **31**, pp. 45–53

[45] Ciresan, D.C., Giusti, A., Gambardella, L.M., *et al.*: 'Deep neural networks segment neuronal membranes in electron microscopy images'. Advances in Neural Information Processing Systems 25: 26th Annual Conf. on Neural Information Processing Systems 2012.Proc. meeting held on 3–6 December 2012, Lake Tahoe, Nevada, United States, 2012, pp. 2852–2860

[46] Stollenga, M.F., Byeon, W., Liwicki, M., *et al.*: 'Parallel multi-dimensional lstm, with application to fast biomedical volumetric image segmentation'. Advances in Neural Information Processing Systems 28: Annual Conf. on Neural Information Processing Systems 2015, Montreal, Quebec, Canada, 7–12 December 2015, pp. 2998–3006

[47] Zuo, W., Wu, X., Lin, L., *et al.*: 'Learning support correlation filters for visual tracking', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019, **41**, (5), pp. 1158–1172

[48] Chen, K., Zhu, D., Lu, J., *et al.*: 'An adversarial and densely dilated network for connectomes segmentation', *Symmetry*, 2018, **10**, (10), p. 467

[49] Chen, H., Qi, X., Cheng, J., *et al.*: 'Deep contextual networks for neuronal structure segmentation'. Proc. Thirtieth AAAI Conf. on Artificial Intelligence, Phoenix, Arizona, USA, 12–17 February 2016, pp. 1167–1173

[50] Shen, W., Wang, B., Jiang, Y., *et al.*: 'Multi-stage multi-recursive-input fully convolutional networks for neuronal boundary detection'. Proc. IEEE Int. Conf. on Computer Vision, Venice, Italy, October 2017, pp. 2391–2400

[51] Drozdzal, M., Chartrand, G., Vorontsov, E., *et al.*: 'Learning normalized inputs for iterative estimation in medical image segmentation', *Med. Image Anal.*, 2017, **44**, pp. 1–13

[52] Phaisangittisagul, E.: 'An analysis of the regularization between l2 and dropout in single hidden layer neural network'. 7th Int. Conf. on Intelligent Systems, Modelling and Simulation, Bangkok, Thailand, 25–27 January 2016, pp. 174–179

[53] Hochreiter, S.: 'The vanishing gradient problem during learning recurrent neural nets and problem solutions', *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, 1998, **6**, (2), pp. 107–116

[54] Wiehman, S., de Villiers, H.: 'Semantic segmentation of bioimages using convolutional neural networks'. Int. Joint Conf. on Neural Networks, Vancouver, BC, Canada, 24–29 July 2016, pp. 624–631

[55] Wu, X.: 'An iterative convolutional neural network algorithm improves electron microscopy image segmentation'. CoRR, 2015, abs/1506.05849

[56] Liu, T., Jones, C., Seyedhosseini, M., *et al.*: 'A modular hierarchical approach to 3d electron microscopy image segmentation', *J. Neurosci. Methods*, 2014, **226**, pp. 88–102