Medical Image Analysis (2020)

Contents lists available at ScienceDirect

Medical Image Analysis

journal homepage: www.elsevier.com/locate/media

Progressive Learning for Large-scale Neuronal Population Reconstruction

Given-name1 Surname1[a,*], Given-name2 Surname2[a,1], Given-name3 Surname3[b], Given-name4 Surname4[b]

[a]*Affiliation 1, Address, City and Postal Code, Country*
[b]*Affiliation 2, Address, City and Postal Code, Country*

ARTICLE INFO

Article history:

Keywords: Deep learning, Image segmentation, Neuronal population reconstruction, Large-scale images, Optical microscopy

ABSTRACT

Reconstruction of neuronal populations from large-scale optical microscopy (OM) brain images is essential to investigate neuronal circuits and brain mechanisms. To mitigate the impact of low quality of images on neuron reconstruction, some studies have been conducted to extract neuron voxels using deep neural networks (DNNs). However, training such a DNN usually relies on a large number of images with voxel-wise annotations, which is very costly to obtain in terms of both finance and labor. In this paper, we propose an unsupervised progressive learning framework for robust neuron reconstruction free of manual annotations. Our framework consists of a neuron tracing module and a segmentation network, which mutually complement and progressively promote each other. Based on the progressively learned model, reconstructing neurons from local noisy image blocks is feasible with allowed memory. To reconstruct neuronal populations from a terabyte-sized large-scale image, we introduce a new automatic reconstruction algorithm by adaptively tracing neurons block by block and assembling individual neurons continuously and smoothly.

We build a dataset called "VISoR-40" that consists of 40 OM image blocks from mouse cortical regions. Extensive experimental results on our VISoR-40 dataset and the public BigNeuron dataset demonstrate the effectiveness and superiority of our method for neuronal population reconstruction and single neuron reconstruction, respectively. Furthermore, we successfully apply our method to the reconstruction of dense neuronal populations in a large-scale mouse brain slice.

1. Introduction

Reconstruction of neuronal morphology is of great significance for brain studies. Obtaining a blue print of the brain architecture, including the morphologies and interconnectivities of neuronal populations, allows for measuring and visualizing neuronal structure, understanding neuronal identity, and determining potential connectivity. Therefore, complete reconstruction of neuronal populations from large-scale brain images is essential to investigate the mechanism of the nervous system, analyze brain changes, and facilitate our understanding of how the brain works or fails to work properly in the diseases such as dementia and Alzheimer's disease Petrella et al. (2003); Giorgio and De Stefano (2013). One of the key techniques in this endeavor is optical microscopy (OM), which allows detailed visualization of neurons in the brain and makes the reconstruction of every neuron possible Senft (2011), as shown in Fig. 1. However, despite numerous efforts devoted, this task is still one of the main challenges in computational neuroscience.

The challenges in large-scale neuronal population reconstruction are mainly caused by complex morphology of neu-

⋆This is an example for title footnote coding.

*Corresponding author: Tel.: +0-000-000-0000; fax: +0-000-000-0000; *e-mail:* author3@author.com (Given-name3 Surname3)

[1]This is author footnote for second author.

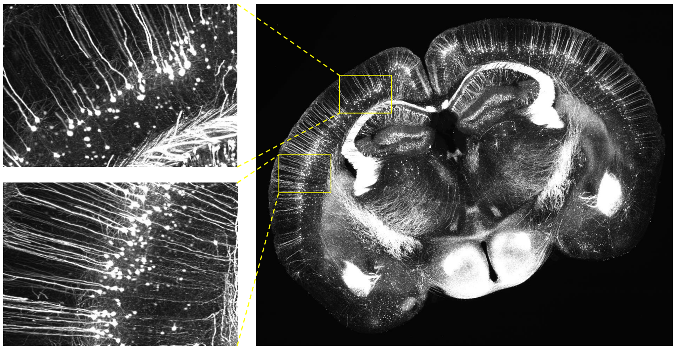

Fig. 1. A 3D mouse brain slice captured by the VISoR imaging system Wang et al. (2019). The dense neuronal populations, large variety of intensities, image noises and large volume pose significant challenges for automatic reconstruction of neuronal populations from the large-scale image.

rons, low quality and large volume of OM brain images. Due to the complicated process of imaging acquisition and the uneven distribution of fluorescence markers in neurons, the intensities of voxels that are occupied by neurons vary dramatically in highly noisy and inhomogeneous environments. Moreover, to detail the structure of neurons in the brain, high resolution of OM images are required. Such an image typically contains trillions of voxels and the image size is formidably large in a terabyte scale. It is even impractical to directly load the entire large image into computer memory before reconstructing neurons when considering the memory cost and tracing time. Manual reconstruction of neuronal populations from OM images is an extremely laborious and time-consuming task, which requires a sophisticated knowledge base of neuron morphology, and is difficult, if not impossible, to be performed at a large scale. Therefore, an effective and automated large-scale neuronal population reconstruction algorithm for these challenging situations is highly desired in practice.

To achieve neuron reconstruction from large-scale OM brain images, some attempts have been made in recent years, such as Neuron Crawler Zhou et al. (2015), UltraTracer Peng et al. (2017) and MEIT Wang et al. (2018). Given a large-scale image, a common solution is to trace the neuron block by block and each local image block is cropped from the raw image. Despite their great improvement in this task, several limitations still remain to be resolved.

One main bottleneck of these methods is that the base tracer used for neuron tracing from image blocks are all conventional tracing schemes, which typically employ traditional image processing algorithms. Unfortunately, these algorithms using hand-crafted features and rules have difficulty in reconstructing neurons from low-contrast and noisy image blocks. To improve the reconstruction quality from noisy image blocks, some deep learning based approaches are proposed recently. However, they require a great amount of manually annotated data for network training. To take advantage of both conventional methods and deep learning techniques, we propose to leverage these two paradigms in a Progressive Learning manner for Neuronal Population Reconstruction (PLNPR).

We observe the approximate locations of neurons can be effectively inferred by existing conventional methods. Although it is hard for them to reconstruct all neurons with complete structures, they still offer important clues to obtain complicated patterns of neurons. Therefore, we can utilize conventional methods to produce pseudo labels for network training. More specifically, our iterative framework trains a deep segmentation network progressively without using manual annotations. The network is expected to learn more comprehensive features of neurons from noisy labels. With a more powerful network for segmenting neurons, the neuron reconstruction could be improved. Then we progressively refine the network with better neuron reconstruction results as pseudo labels, and reconstruct more complete neurons with better neuron segmentation.

Another main bottleneck of existing large-scale neuron reconstruction methods is that they mainly focus on large-scale single neuron reconstruction and the images they processed usually contain only one single neuron. However, a large-scale OM brain image often contains dense neuronal populations in practice. To reconstruct dense neuronal populations from large-scale or even ultra-large-scale noisy images, we introduce UltraNPR, a block-by-block tracing algorithm which utilizes the proposed deep-learning based PLNPR method as the base tracer and designs a new tracing strategy and a fusion algorithm for the large-scale neuronal population reconstruction application.

A preliminary version of this work appears in Zhao et al. (2019), where only the robust neuron reconstruction from local image blocks is investigated. In this paper, we tackle with the more challenging large-scale neuronal population reconstruction. The contributions of this work are summarized as follows:

1. We propose a general unsupervised progressive learning framework along with a neuron segmentation network for neuron reconstruction.
2. We propose a new large-scale neuronal population reconstruction algorithm which is capable of reconstructing dense neuronal populations from terabyte-sized OM brain images.
3. We build a new dataset called "VISoR-40" which consists of 40 OM image blocks from mouse cortical regions to facilitate neuroscience research.
4. Extensive experiments on our VISoR-40 dataset and the public BigNeuron dataset demonstrate the effectiveness and superiority of our progressive learning algorithm for neuronal population reconstruction and single neuron reconstruction, respectively.

The remainder of this paper is organized as follows. In Section 2, we make a survey of related work. In Section 3, we first introduce the PLNPR algorithm, then elaborate the UltraNPR algorithm. We report the experiment results and discussions in Section 4. Finally, conclusions are drawn in Section 5.

2. Related Work

2.1. Techniques for Robust Neuron Reconstruction

Early conventional techniques for neuron reconstruction typically employ traditional image processing algorithms, such

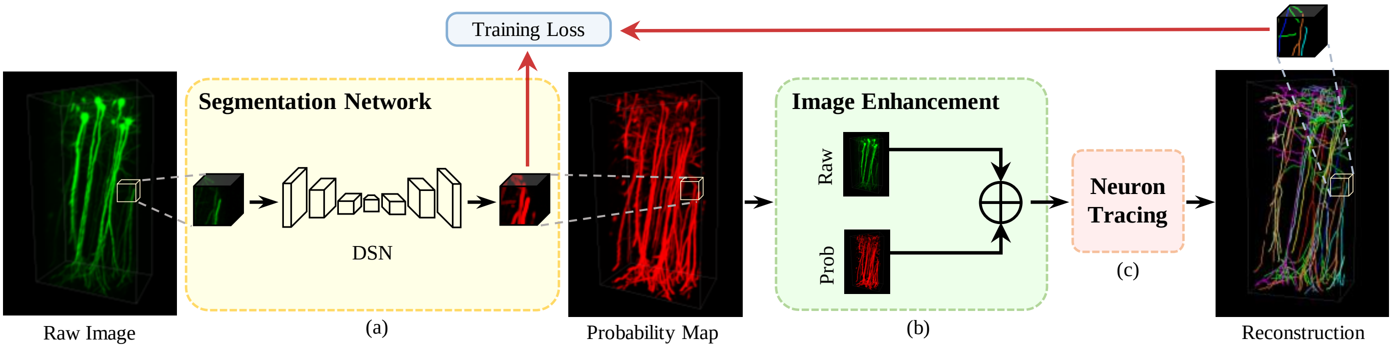

Fig. 2. Diagram of our progressive learning algorithm for neuron reconstruction in an image block. (a) The segmentation network extracts neuron signals from the raw image. (b) The output probability map is employed to enhance the raw image in order to preserve both global structures and local signal details, which facilitates the neuron tracing module (c) for more complete neuronal population reconstruction. To train the segmentation network, we use the reconstructed neurons as pseudo labels (red arrows), and iteratively refine the network learning and neuron reconstruction with a set of images. The black arrows show the reconstruction pass from an image during testing.

as snakes Wang et al. (2011); Cai et al. (2006), principal curves Bas and Erdogmus (2011), graph theory Peng et al. (2010); Yang et al. (2013); De et al. (2016), model-fitting Zhao et al. (2011); Santamaría-Pang et al. (2015), watershed Navlakha et al. (2013); Sümbül et al. (2016), energy minimization Quan et al. (2013); Liu et al. (2016), mean-shift clustering Frasconi et al. (2014), ray-shooting Wu et al. (2014); Liu et al. (2019), fast-marching Peng et al. (2011); Xiao and Peng (2013); Liu et al. (2018) and so on. Unfortunately, all these conventional algorithms rely on hand-crafted features and carefully tuned parameters, and usually tend to fail when the image quality is poor.

To improve the reconstruction performance from low-quality image blocks, some machine learning based methods were introduced to extract neuron voxels for neuron reconstruction. This kind of methods employs various classifiers with hand-crafted features, such as support vector machine (SVM) Chen et al. (2015), minimum spanning tree Basu et al. (2016), Bayesian probabilistic model Radojević and Meijering (2017), Bootstrap aggregating Wang et al. (2017), gradient boosting decision trees (GBDT) Gu et al. (2017), Markov chain Monte Carlo (MCMC) Skibbe et al. (2015); Skibbe et al. (2019) and so on. However, the main limitation of these methods is that hand-crafted features usually suffer from limited representation capability for accurate recognition, considering more challenging and complex image blocks.

Recently, we have evidenced an increasing development of deep learning based methods Li et al. (2017); Zhou et al. (2018); Xu et al. (2016), which bring the power of DNNs to improve the reconstruction performance. Instead of manually designing sophisticated features, these methods learn feature representations in a data-driven way and extract more distinctive features. When more complicated classifiers are employed to segment neuron voxels from image blocks, these methods can achieve more robust reconstruction results. Though great improvement of neuron reconstruction could be achieved, these deep learning based methods rely on strong supervision for network training, i.e., manual annotations for neuron voxels. Unfortunately, due to the complicated morphology of neurons and the low quality of OM images, such annotations are very costly to obtain in terms of both time and labor. In comparison, we propose a novel iterative framework to progressively improve the 3D DNN-based neuron reconstruction performance without using manual annotations.

2.2. *Large-scale Neuron Reconstruction*

Most existing neuron reconstruction methods focus on robust and accuracy neuron reconstruction from local noisy image blocks. Despite substantial advancements in these methods, they often need to load all image voxels into memory before the reconstruction and the sheer volume of a large-scale image is far beyond the processing capability for them, especially on the memory cost and tracing time. In recent years, some attempts have been made to reconstruct neurons from large-scale OM images, such as Neuron Crawler Zhou et al. (2015), UltraTracer Peng et al. (2017) and MEIT Wang et al. (2018). To tackle the challenges caused by the large volume of images, a common solution is to reconstruct neuronal morphology block by block. Each block is cropped from the raw image and is much smaller in size than the raw image. Therefore, existing tracing methods, such as APP2 Xiao and Peng (2013), MOST Wu et al. (2014) and FMST Yang et al. (2019), can be directly used as the base tracer in their methods to trace neurites in each block. Then the reconstructed neurites in all blocks are assembled to obtain the final reconstruction.

However, all of these methods focus on single neuron reconstruction and the images they process usually contain only one single neuron. Since these methods are mainly designed for large-scale single neuron reconstruction, they are not useful for dense neuronal populations, in which neurons frequently contact or close to each other in the image. When the closely spaced neurites that belong to different neurons are not be distinguished, neurons can not be separately reconstructed. Therefore, a complete reconstruction of dense neuronal populations from large-scale images still remains challenging for existing methods. In this work, we introduce a new algorithm to reconstruct neuronal populations from large-scale or even ultra-large-scale images.

3. Proposed Method

In this section, we describe the details of our methods, including PLNPR and UltraNPR. Given a noisy and large-scale OM brain image, PLNPR aims to obtain a robust and accurate reconstruction of neurons from local noisy image blocks, while UltraNPR aims to reconstruct complete neuronal populations from a large-scale image more efficiently.

3.1. PLNPR for Robust Neuronal Population Reconstruction

To reconstruct neuronal populations from noisy OM images, our PLNPR algorithm consists of three key components: a segmentation network, an image enhancement module and a neuron tracing module, as shown in Fig. 2. The segmentation network is designed to extract neuron voxels from noisy and complex backgrounds. Then, instead of using a binary mask of the segmented voxels, we employ a probability map predicted by the network and integrate it with the raw image intensities in order to simultaneously preserve the global neuron structure and local neurite details. After that, the neuron tracing module is applied to reconstruct neurons from the enhanced image block. We progressively train the segmentation network with the reconstructed neurons inferred from conventional tracing methods as labels, and improve the reconstruction results from better neuron segmentation. Based on the iterative learning process, the powerful DNNs and tracing methods mutually complement and promote each other to gradually improve the neuron reconstruction performance.

3.1.1. Progressive Learning without Annotations

In order to train the segmentation network to learn discriminative features for extracting neuron voxels, we use the reconstructed neurons to provide pseudo labels. In each iteration of segmentation and reconstruction, we apply the NGPST Quan et al. (2015) as the neuron tracing module to reconstruct neurons from image blocks. This module can be replaced by any tracing method that does not require manual annotations for training. It takes an image block $\mathbf{B}$ in size of $S \times H \times W$ as input and reconstructs a neuronal population with separated neurons. From the reconstruction results, we produce a binary mask $\mathbf{M}$ indicating foreground by $\mathbf{M}(x) = 1$ and background by $\mathbf{M}(x) = 0$ for a voxel x.

Given N image blocks, we train our segmentation network using the neuron masks $\{\mathbf{M}_i, i = 1, \cdots, N\}$ generated from the neuron tracing module. Details of the network architecture and training strategy are described in Sec. 3.1.2. The output of the neuron segmentation network is a 3D probability map $\mathbf{P}$, which is computed by a voxel-wise softmax activation function. $\mathbf{P}(x) \in [0, 1]$ indicating the probability of a voxel x to be a neuron part. Then, by fusing the predicted probability map with the raw intensities, the raw block is further enhanced in order to preserve both local signals and global structures simultaneously. Details are introduced in Sec. 3.1.3. When the enhanced block is used as input to the neuron tracing module, more complete neuronal populations can be reconstructed.

3.1.2. DNN for 3D Neuron Segmentation

Considering the size, morphology and intensity of neurons vary significantly, extracting neuron voxels from image blocks is not a trivial problem. In recent years, many 3D DNNs, such as 3D U-Net Çiçek et al. (2016), 3D DSN Dou et al. (2017) and DenseVoxNet Yu et al. (2017) have demonstrated an outstanding capability in various biological and biomedical image segmentation tasks. Therefore, we take advantage of 3D segmentation networks to extract more representative features to meet the challenges of neuron segmentation. In this work, the 3D DSN is extended as our neuron segmentation network to balance the performance and computation burden.

The 3D DSN network consists of convolutional layers, pooling layers and deconvolutional layers, all in 3D fashion. The convolutional layers and pooling layers act as feature extractor, while the deconvolutional layers followed by softmax layer aim to up-sample the feature maps to the same size as the input. To further boost the information flow within the network, two more branches are employed to connect the shallower layers to the output layer. These connections strengthen the gradient propagation, stabilize the learning process, and further taps the potential of the limited training data to learn more discriminative features.

Although 3D DSN has achieved excellent performance for 3D organ segmentation Dou et al. (2017), it is still prone to overfitting in our case due to the limited training data. One common solution to this problem is to combine the predictions of several DNN models at the test time. However, this approach will greatly reduce the efficiency of the system. Instead of training several models simultaneously, we employ a more efficient alternative solution for network training, which is known as Dropout Srivastava et al. (2014). This technique can significantly reduce node interactions and help the network to learn more robust features that better generalize to new data. In our network, the dropout with a rate of 0.5 is applied to every convolutional layer.

Another challenge of training 3D DNN is the memory limitation because the 3D feature images are huge with respect to the input size. Therefore, for each input image block, we crop a group of cubes in size of $160 \times 160 \times 160$ with 30% overlaps, and set batch size to 1 during training. Correspondingly, at the test time, we stitch these overlapped probability cubes together using average blending to get a probability map in the same size with the input block.

In addition, the volume of neuron (foreground) voxels is usually much smaller than that of background in an OM image. To cope with this imbalance problem, a data balancing technique is introduced for network training. Specifically, when computing the training loss, we only consider the neuron voxels and a certain portion of background voxels, which is randomly selected as non-neuron samples. The number of non-neuron voxels used for training is set as 10 times that of neuron voxels.

Last but not least, to have the same physical resolution with the lateral dimension in OM image blocks, voxels along the axial dimension are interpolated after the imaging process. However, it makes the image quality along different dimensions inhomogeneous. In order to alleviate the impact of this problem on network training, a random transposition process is employed for each cube as data augmentation.

3.1.3. Image Enhancement

After finishing the network training, we use the trained model to predict the probability cube for each input cube. Then, by stitching all probability cubes together, we can obtain the probability map $\mathbf{P}$ for the entire raw image block $\mathbf{B}$. Each element in $\mathbf{P}$ indicates the probability that the corresponding voxel in $\mathbf{B}$ belongs to the neuron. To utilize the probability map, one natural way is to reconstruct neurons directly from it. However, since the pseudo labels is not as accurate as manual annotations, the performance of the segmentation network is limited. As a result, the predicted probability map may not cover all neurons and some local details may be lost, especially for the first few iterations. In our method, an enhanced representation is generated by fusing the probability map and original raw image block, in order to keep detailed structures and suppress noise signals effectively. Specifically, a new probability map $\widetilde{\mathbf{P}}$ is first constructed by linearly mapping the value range of $\mathbf{P} \in [0, 1]$ to the value range $[\mathbf{B}_{min}, \mathbf{B}_{max}]$ of $\mathbf{B}$.

$$\widetilde{\mathbf{P}}(x) = (\mathbf{B}_{max} - \mathbf{B}_{min})\mathbf{P}(x). \tag{1}$$

Then, base on the raw image block $\mathbf{B}$ and the probability map $\widetilde{\mathbf{P}}$, an enhanced image block $\mathbf{E}$ is computed as

$$\mathbf{E}(x) = \alpha\widetilde{\mathbf{P}}(x) + (1 - \alpha)\mathbf{B}(x), \tag{2}$$

where $\alpha \in [0, 1]$ is a weight to control the contributions of voxel x in the original intensity and the probability map. By feeding the enhanced blocks to the neuron tracing module, neuronal populations can be reconstructed more completely.

With more reliable reconstruction results for supervision, the segmentation network could be further trained to learn more discriminative and representative features for producing probability maps, which in turn benefits the tracing module to reconstruct neurons in the next iteration. As shown in Fig. 3(a), due to the noises and low contrast in the raw image block, the intensity of neuron voxels is inhomogeneous, which makes some neurons subtle. At first, comparing with the ground truth (GT) shown in Fig. 3(k), the neurons reconstructed from the raw image block are incomplete and many neurites are missing, as shown in Fig. 3(g). Then, by utilizing the pseudo labels derived from imperfect reconstruction, the segmentation network can be trained to learn features for global trajectories. Fig. 3(b) shows the predicted probability map, which demonstrates the enhanced trajectories. With more iterations of neuron reconstruction and network training, more distinctive and long-range trajectory features can be progressively captured by the network, as shown in Fig. 3(c)(d)(e). By combining the original image intensities with the predicted probability map, both local signal details and global trajectories are well preserved in the enhanced block, as Fig. 3(f) shows. Iteration by iteration, the completeness and accuracy of neuron reconstruction are increasingly improved, as shown in Fig. 3(h)(i)(j).

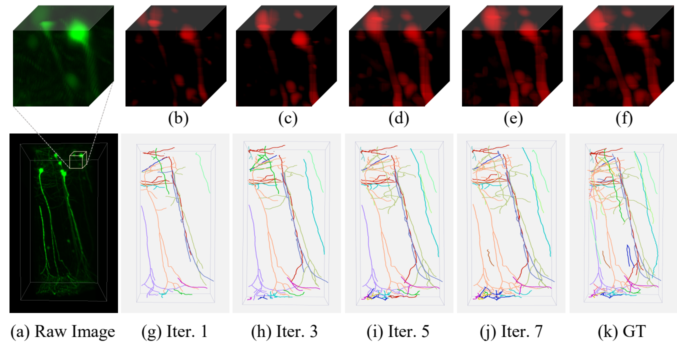

Fig. 3. Our progressive learning technique gradually improves the segmentation network to extract neuron signals from (a) raw image which has noises and low contrast. (b)(c)(d)(e) The probability maps generated by the segmentation network at different iterations. (f) Combing the probability map and the raw intensity, the enhanced block preserves both global trajectory and local details. (g)(h)(i)(j) More and more complete and accurate reconstruction of the neuronal populations can be obtained with more iterations. (k) The manually labeled neurons are shown for comparison. Separated neurons are shown in different colors.

3.2. Large-scale Neuronal Population Reconstruction

To reconstruct neuronal populations from a large-scale 3D image, our UltraNPR consist of four key components: a soma detection module, a local reconstruction module, a block search module, and a neurites fusion module, as shown is 4. At first, the large-scale image is divided into overlapped 3D blocks of the same size. The soma detection module is used to detect somas from the large-scale image. The local reconstruction module is based on our PLNPR algorithm to reconstruction neurons from low-quality image blocks. Then, to reconstruct the large-scale image block by block, we repeatedly use the block search module to determine what the next block needs to be reconstructed. After that, the neurites fusion module is applied to obtain a complete and continuous reconstruction from adjacent blocks.

3.2.1. Initial Soma Detection

To detect somas from the large-scale image efficiently, we apply soma detection algorithm Quan et al. (2013) on each block separately. However, due to the overlap between blocks, somas in the overlapped area would be detected repeatedly. To tackle this over-detection error, we map all detected somas from local blocks to the coordinate system of the original image and merge the overlapped somas by averaging their position and radius. The soma detection results from the large-scale image are visualized in Fig. 4 of the supplementary file. These somas are also used to define the initial blocks for neuron reconstruction in the next step.

3.2.2. Block Search Policy and Local Reconstruction

The neuronal population reconstruction $\mathbf{R}$ in a large-scale image can be modeled as the joint performance of the reconstruction in each block $\mathbf{B}_i$. Given an image $\mathbf{I}$, performing a local tracing method T to reconstruct neuronal populations block by block can be formulated as follows

$$\mathbf{R} = T(\mathbf{I}) = T(\mathbf{B}_1, \mathbf{B}_2, \cdots, \mathbf{B}_N). \tag{3}$$

As somas are where signals from the dendrites are joined and pass on, the blocks containing somas are the most probable locations to start the tracing. Therefore, the initial blocks that

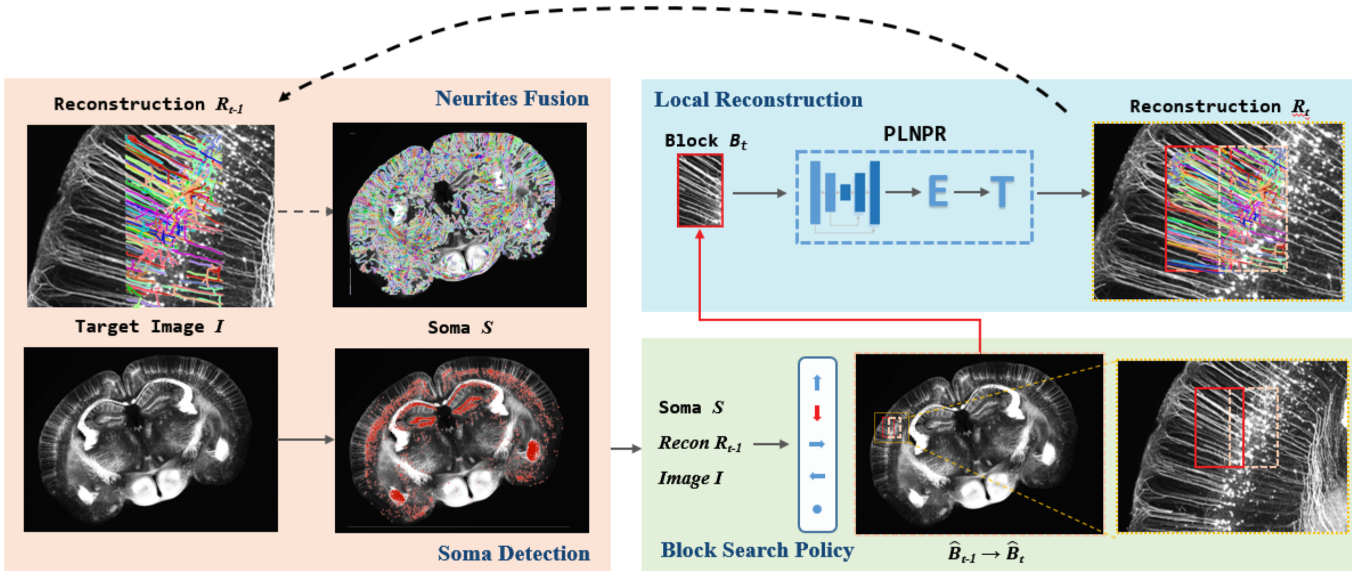

Fig. 4. Diagram of our UltraNPR algorithm for neuronal population reconstruction in a large-scale brain slice.

contain somas are firstly reconstructed by our PLNPR method. The reconstruction produced by PLNPR is represented by a series of neuronal compartments. The neuronal compartments which are close to the block boundary typically indicate the continuity of the structure of neurons. We call these compartments as terminal tips and use them as the potential continuous signals for searching the next blocks to grow the neuronal structure from each initial block. Then, neurons in the next blocks are reconstructed by PLNPR based on the terminal tips provided by adjacent blocks that have already been reconstructed. By iteratively searching the next blocks, UltraNPR can reconstruct neuronal populations more and more completely.

In addition, due to the dense distribution of neurons in the brain image, fragmented neurites belonging to different neurons could be in the same block, which means some blocks may be repeatedly reconstructed starting from different initial blocks. To more efficiently explore the structure of neuronal populations, for each searching direction of an initial block, the iterative searching process will be terminated when the next block contains somas or no new terminal tips could be detected.

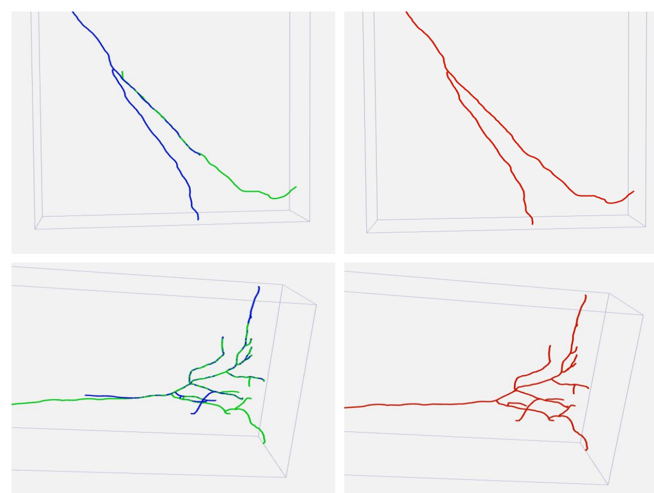

Fig. 5. Two examples of correcting the over-tracing and topological discrepancy errors by our fusion algorithm. These reconstructions are shown in skeleton mode for better visualization.

3.2.3. *Neurites Fusion from Adjacent Blocks*

Since neurons would be split into fragmented neurites when dividing the raw image into blocks, neurites from adjacent blocks need to be maximally connected, and the connection should be as continuous and smooth as possible. As fragment neurites in the overlapping area may belong to different neurons, we first match them by comparing the overlapping region between every two neurites from adjacent blocks. Then, the matched neurites are assembled to get a continuous and smooth reconstruction.

However, simply assembling them would cause over-tracing error and topological discrepancy, as shown in Fig. 5. The over-tracing error is caused by the overlap between blocks when the respective neurites from these blocks are assembled. The topological discrepancy is mainly caused by the lack of context information for tracing methods, which makes the reconstruction near the block boundary often inaccurate and unreliable. But this region is inside the adjacent block due to the overlap between them, which means substantial context information of this region can be provided to the tracing methods and leads to a better reconstruction. Therefore, when assembling two matched neurites from adjacent blocks, we only consider neuronal compartments which are not near the boundary of the corresponding block.

Based on the above observations, a fusion algorithm is designed to assemble the matched neurites. Concretely, we first calculate the length of two neurites and select the longer one as the reference neurite. Here, we use $\mathcal{N}_{\mathcal{A}}$ to denote the refer-

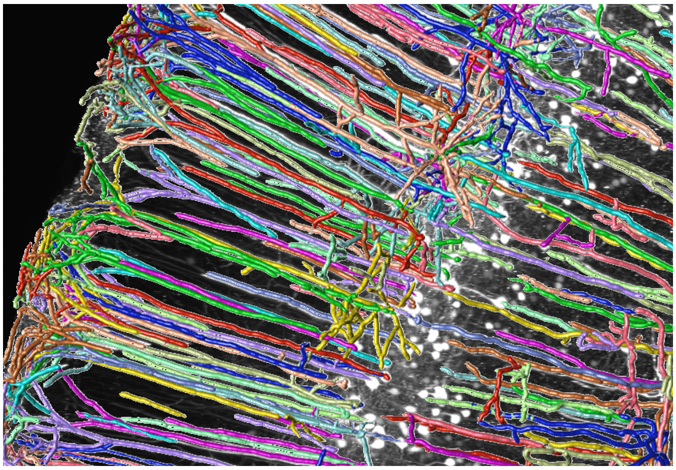

Fig. 6. One example of reconstructing neuronal populations from four adjacent blocks using our method. We can observe that, the fragmented neurites from adjacent blocks are assembled continuously and smoothly.

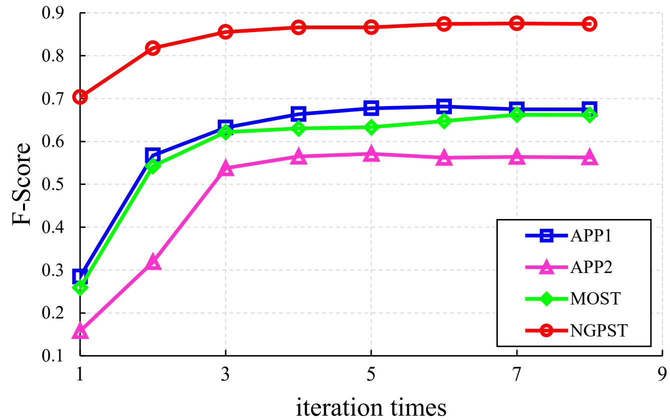

Fig. 7. F-Score of neuron reconstruction using four neuron tracing methods APP1 Peng et al. (2011) and its variant APP2 Xiao and Peng (2013), MOSTWu et al. (2014) and NGPST Quan et al. (2015) on the VISoR-40 test dataset at eight iterations. For each of the four neuron tracing methods, our approach progressively improves their reconstruction results.

ence neurite, and $\mathcal{N}_{\mathcal{B}}$ to denote another neurite. Then, for each branch $\mathcal{H}_{\mathcal{B}}$ in neurite $\mathcal{N}_{\mathcal{B}}$, we search in neurite $\mathcal{N}_{\mathcal{A}}$ to see if there is a branch $\mathcal{H}_{\mathcal{A}}$ that overlaps with it. If yes, three steps will be performed to assemble the two branches together. Firstly, we remove the neuronal compartments in branch $\mathcal{H}_{\mathcal{A}}$ and branch $\mathcal{H}_{\mathcal{B}}$ that near the boundary of the corresponding block, and all branches that are connected to the removed compartments will also be removed. Here, this distance is set to be 70 voxels. Secondly, for each neuronal compartment in branch $\mathcal{H}_{\mathcal{B}}$, if there are compartments of branch $\mathcal{H}_{\mathcal{A}}$ around it, it will be removed; otherwise, it is considered to be valid. The searching radius is empirically set to be 10 voxels. Thirdly, the first compartment in the modified branch $\mathcal{H}_{\mathcal{B}}$ is connected to $\mathcal{H}_{\mathcal{A}}$ by searching the nearest compartment in branch $\mathcal{H}_{\mathcal{A}}$. When all branches $\mathcal{H}_{\mathcal{B}}$ overlapping with the neurite $\mathcal{N}_{\mathcal{A}}$ are processed, we can obtain the assembled neurite $\mathcal{N}_{\mathcal{AB}}$. If there are remaining branches $\mathcal{H}_{\mathcal{B}}$ which are not overlapped with neurite $\mathcal{N}_{\mathcal{A}}$, the $\mathcal{N}_{\mathcal{AB}}$ will be updated by assembling the remaining branches $\mathcal{H}_{\mathcal{B}}$.

Fig. 5 shows two examples of correcting the over-tracing and topological discrepancy errors by our algorithm. As shown in Fig. 6, the neuronal population from four adjacent blocks are successfully reconstructed from the noisy images by our Ultra-NPR method and the fragment neurites in adjacent blocks are assembled continuously and smoothly.

4. Experiments and Results

In this section, we conduct an extensive evaluation of our PLNPR method for neuronal population reconstruction on the VISoR-40 dataset, and single neuron reconstruction on the BigNeuron dataset Peng et al. (2015). Then we apply our UltraNPR algorithm for neuronal population reconstruction from a large-scale OM brain image.

4.1. Evaluation of PLNPR on VISoR-40 Dataset

4.1.1. VISoR-40 Dataset

Though many neuron tracing techniques have been proposed, no OM image dataset of large-scale dense neuronal populations has been released up to now. We construct a publicly-available neuron image dataset, which we call "VISoR-40" [2], in order to validate our method and, more importantly, to encourage the development of more generalized algorithms for neuronal population reconstruction. The VISoR-40 dataset consists of 40 OM image blocks ranging in size from $419 \times 1197 \times 224$ to $869 \times 1853 \times 575$. These blocks were captured by the VISoR imaging system Wang et al. (2019) at a physical resolution of $0.5 \times 0.5 \times 0.5\ \mu m^3$ per voxel, which is feasible for the identification of every neuron at the mesoscopic scale. In addition, to preserve enough signal details, all blocks have 16-bit dynamic range of intensity. We randomly select 32 blocks for progressively training the segmentation network. Then the remaining blocks with manual annotations are used as the testing data. Each testing block is first labeled manually and independently by two experienced technicians. Then, by cross-checking each other's result, their agreed annotation is approved by an expert to generate the final ground truth.

4.1.2. Experimental Settings and Evaluation Metrics

Pytorch is adopted to implement the DSN model. At each iteration of the progressive learning, the network is trained from scratch with weights initialized from Gaussian distribution with zero-mean and variance of 0.01. The optimization is realized with the stochastic gradient descent algorithm with the Adam update rule (batch size of 1, weight decay of 0.0005, momentum of 0.9). The base learning rate is set to 0.001 and descended with "poly" learning rate policy (power of 0.9 and the maximum iteration number of 24000). The cube size is selected as $160 \times 160 \times 160$ considering the limitation of GPU memory.

To quantitatively evaluate our method, four commonly used metrics defined in Quan et al. (2015), including Precision, Recall, F-Score, and Jaccard, are computed to measure the fidelity between a reconstruction and the ground truth. Their definitions

[2] VISoR-40 dataset has been available at `https://braindata.bitahub.com`.

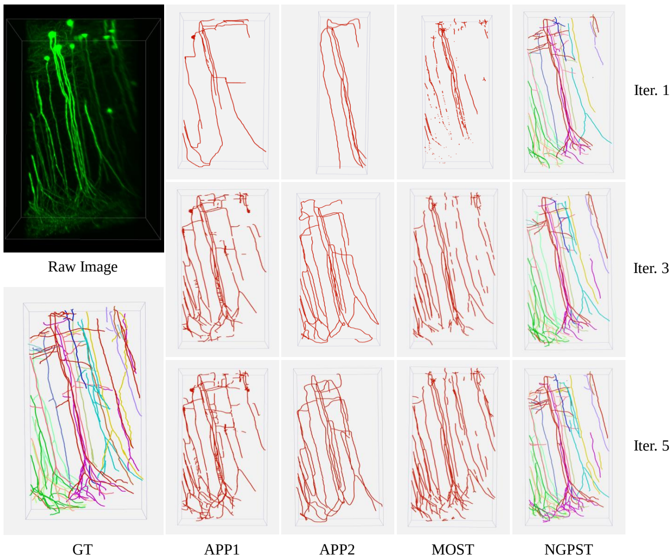

Fig. 8. Reconstruction results of the neuronal population in a test image from the VISoR-40 dataset at different iterations (top to bottom) using four neuron tracing methods APP1 Peng et al. (2011), APP2 Xiao and Peng (2013), MOSTWu et al. (2014) and NGPST Quan et al. (2015).

Table 1. F-Score of neuron reconstruction on a test image from the VISoR-40 dataset at different iterations using four neuron tracing methods APP1 Peng et al. (2011), APP2 Xiao and Peng (2013), MOSTWu et al. (2014) and NGPST Quan et al. (2015).

Method	iter-1	iter-3	iter-5
APP1 Peng et al. (2011)	0.315	0.599	0.599
APP2 Xiao and Peng (2013)	0.251	0.511	0.538
MOST Wu et al. (2014)	0.322	0.588	0.721
NGPST Quan et al. (2015)	0.758	0.825	0.888

are defined as follows:

$$Precission(R, G) = \frac{|R \cap G|}{|R|} = \frac{|TP|}{|R|}, \tag{4}$$

$$Recall(R, G) = \frac{|R \cap G|}{|G|} = \frac{|TP|}{|G|}, \tag{5}$$

$$F-Score(R, G) = \frac{2|R \cap G|}{|R| + |G|} = \frac{2|TP|}{|R| + |G|}, \tag{6}$$

$$Jaccard(R, G) = \frac{|R \cap G|}{|R \cup G|} = \frac{|TP|}{|R \cup G|}, \tag{7}$$

where R denotes the set of points on the reconstructed neurons, G denotes the set of neuron points in the ground truth and TP denotes the set of true positive points, $|\cdot|$ denotes the number of points in a set. The four metrics are first computed on each individual neuronal tree according to the manually labeled skeleton, and then averaged in a neuronal population weighted by the total length of the neuronal processes of each neuron, the same as Quan et al. (2015).

4.1.3. Progressive Learning

The key idea of PLNPR is to progressively improve the performance of neuron reconstruction by making the neuron segmentation network and the conventional tracing method com-

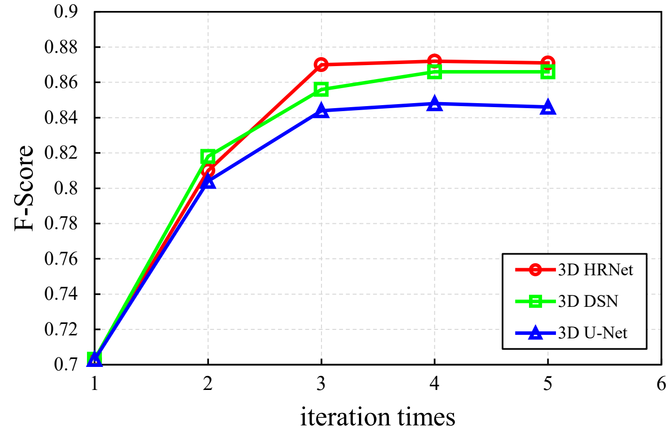

Fig. 9. F-Score of neuron reconstruction on the VISoR-40 test dataset at five iterations. Combining any one of the three neuron segmentation networks, our approach progressively improves the reconstruction performance.

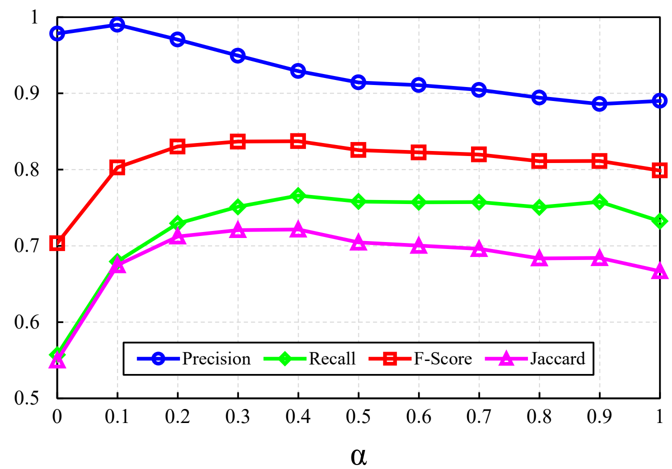

Fig. 10. Neuron reconstruction performance with different α in Eq. (2) for image enhancement on the VISoR-40 test dataset. From left to right, the value of α increases from 0 to 1 by a step of 0.1.

plementary and synergistic without using any manual annotations. In order to demonstrate the performance improvement, four extensively used tracing methods are tested as the neuron tracing module in our framework. They are APP1 Peng et al. (2011), APP2 Xiao and Peng (2013), MOSTWu et al. (2014) and NGPST Quan et al. (2015) respectively. Their implementations are available in the software Vaa3D Peng et al. (2014). Eight iterations are tested on our VISoR-40 dataset, and the improvement of neuronal population reconstruction can be seen in the Fig. 7. Here, we only show the F-Score which is widely used to reflect the overall performance of neuron reconstruction. Moreover, the neuron reconstruction performance on a testing block at different iterations are shown in the Fig. 8 and Table. 1. More qualitative and quantitative results are reported in the supplementary materials. It can be observed that, our progressive learning strategy effectively facilitates conventional tracing methods to reconstruct more complete neuronal populations. In addition, after about five iterations of the progressive learning, the reconstruction is relative complete and the room of performance improvement is becoming smaller with further

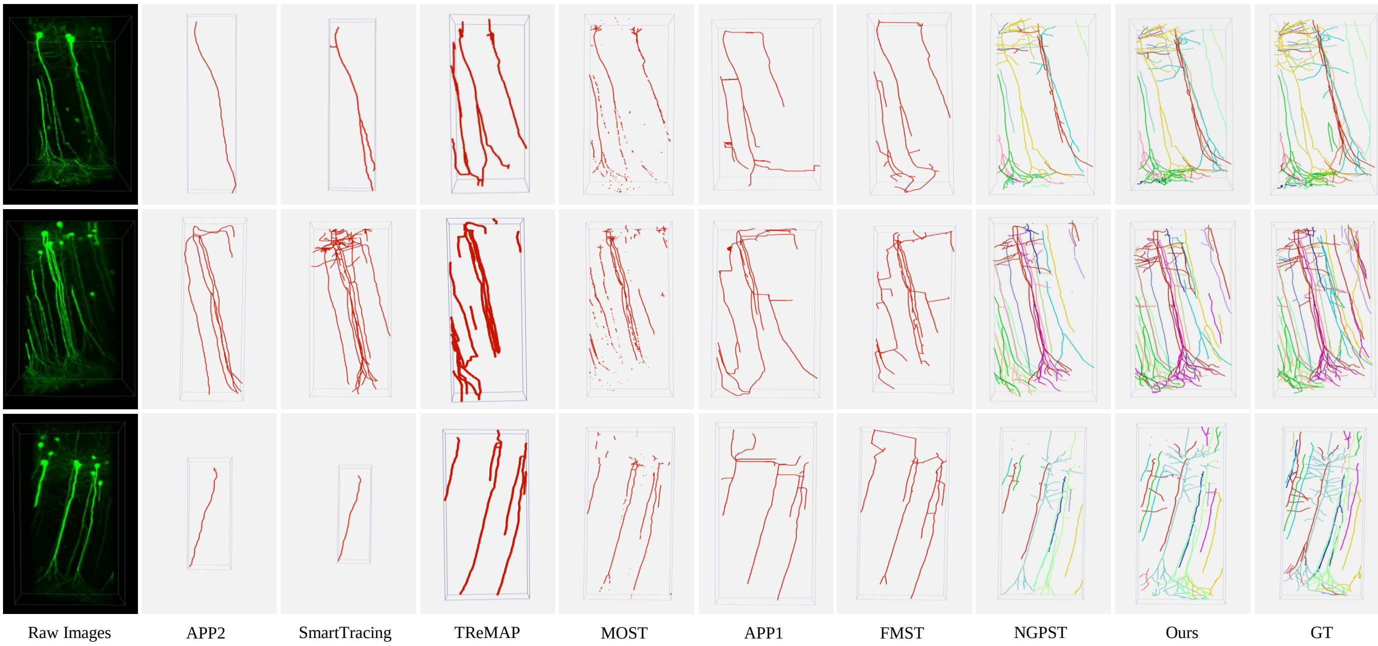

Fig. 11. Comparison of neuronal population reconstruction results using neuron reconstruction methods FMST Yang et al. (2019), APP1 Peng et al. (2011), APP2 Xiao and Peng (2013), SmartTracing Chen et al. (2015), MOST Wu et al. (2014), NGPST Quan et al. (2015) and our PLNPR on three test images from the VISoR-40 dataset. We can observe that, our method reconstructs more complete and accurate neuronal populations compared to other methods.

iterations.

4.1.4. *Neuron Segmentation Network*

To further verify the effectiveness and robustness of our progressive learning strategy, we test three commonly used deep segmentation networks for extracting neuron voxels in our framework. They are 3D DSN Dou et al. (2017), 3D U-Net Çiçek et al. (2016) and a 3D version of HRNet Sun et al. (2019), respectively. Five iterations are tested on our VISoR-40 dataset, and the F-Score of reconstruction results is shown in the Fig. 9. It can be seen that our PLNPR algorithm can effectively improve the neuron reconstruction performance by combining any one of the three neuron segmentation networks. The network and tracing method can complement and promote each other, leading to more complete reconstruction.

4.1.5. *Enhancement Parameter*

In order to explore the influence of parameter α in Eq. (2) for image enhancement, we adopt different values for α, and the results are shown in Fig. 10. $\alpha = 0$ means that the raw image block is directly used as input for the tracing module. $\alpha = 1$ means that only the probability maps are used as input for neuron tracing. It indicates that the performance is improved by combing the probability map with the raw intensity, mainly because that the probability map reflects the long-range trajectory structures while the original intensity preserves signal details to some extent. In this paper, we empirically select $\alpha = 0.1$ to reduce the influence of false positive predictions in probability maps due to the limited performance of the DNN model trained by pseudo labels and increase the robustness of the whole framework.

4.1.6. *Comparison on VISoR-40 Dataset*

To prove the effectiveness of our method on neuronal population reconstruction, we compare it with seven widely used neuron tracing methods, which include APP2 Xiao and Peng (2013), SmartTracing Chen et al. (2015), TReMAP Zhou et al. (2016), MOST Wu et al. (2014), APP1 Peng et al. (2011), FMST Yang et al. (2019) and NGPST Quan et al. (2015). The parameters of tracing methods are manually adjusted for each image to get optimal performance in the experiments. Table 2 compares the quantitative results of different methods with regard to Precision, Recall, F-Score and Jaccard. "Ours" means that we utilize NGPST with DSN enhancement after progressive learning on the VISoR-40 dataset. From Table 2, we can observe that our method makes a significant improvement compared with other methods. The neuronal populations reconstructed from three testing image blocks are visualized in Fig. 11. It can be observed that our method outperforms others in both sparse and dense neurons. Conventional methods Xiao and Peng (2013); Peng et al. (2011); Wu et al. (2014); Zhou et al. (2016) and machine learning based methods Chen et al. (2015); Yang et al. (2019) tend to extract the main trunk of neurons, while missing a large portion of subtle neurites. Therefore, these methods have very high precision but significantly lower recall. Although NGPST Quan et al. (2015) has a better performance of neuronal identity, it still remains difficult to extract subtle neuron voxels by using hand-crafted features. In comparison, our method benefits from the progressively trained DSN, and reconstructs more complete neurons from challenging blocks, even there exhibit noises, low contrast and blending of fluorescence in the blocks.

Table 2. Performance comparison with different methods for neuronal population reconstruction on the VISoR-40 dataset.

Method	Precision	Recall	F-Score	Jaccard
APP2 Xiao and Peng (2013)	**0.980**	0.091	0.157	0.091
SmartTracing Chen et al. (2015)	0.961	0.133	0.205	0.128
TReMAP Zhou et al. (2016)	0.917	0.147	0.253	0.145
MOST Wu et al. (2014)	0.969	0.151	0.258	0.151
APP1 Peng et al. (2011)	0.935	0.169	0.284	0.167
FMST Yang et al. (2019)	0.884	0.179	0.296	0.176
NGPST Quan et al. (2015)	0.978	0.557	0.703	0.549
Ours	0.971	**0.801**	**0.875**	**0.781**

4.2. Evaluation of PLNPR on BigNeuron Dataset

4.2.1. BigNeuron Dataset

To validate our PLNPR method on the single neuron reconstruction, we employ the BigNeuron Peng et al. (2015) dataset which is a well-known community-derived neuron dataset. This dataset totally consists of about $20,000$ 3D OM images, acquired from a variety of species and optical imaging systems. Some images have the corresponding manual annotations for evaluation. Unlike our VISoR-40 dataset which is built for the evaluation of neuronal population reconstruction, each block in the BigNeuron dataset contains a single neuron or fragmented neurites which is appropriate for single neuron reconstruction. In this work, 68 images we used from the BigNeuron dataset are the same as Li2017 Li et al. (2017). These images are from a variety of species, and each image has the corresponding expert manual reconstruction. 3/4 of these images are sampled for network training and the remaining images are used for evaluation to compute the neuron tracing performance.

4.2.2. Experimental Settings and Evaluation Metrics

We evaluate the proposed progressive learning method on the single neuron reconstruction application. The DSN model is progressively trained on the VISoR-40 dataset first and then fine-tuned on the BigNeuron dataset using pseudo labels generated by NGPST Quan et al. (2015) instead of the provided manual annotations. The learning rate was initialized as 1×10^{-4} and decayed using the "poly" learning rate policy. The maximum iteration number is set to 24000. We cropped patches of size $160 \times 160 \times 8$ as input to the network, considering consumption of the GPU memory, and also unequal image sizes along three directions.

Since the implementation of most learning-based tracing methods, such as Li et al. (2017) are not publicly available, to compare with Li et al. (2017), the testing data and three evaluation metrics for evaluation are the same as those used in Li et al. (2017). The three measurements defined in Peng et al. (2010) include entire structure average (ESA), different structure average (DSA) and percentage of different structures (PDS). For all of these three scores, larger values indicate higher discrepancy between the tracing results and the manual reconstruction.

4.2.3. Comparison on BigNeuron Dataset

On the BigNeuron dataset, we compare with seven widely used tracing methods to validate the effectiveness of our proposed method. They are MOST Wu et al. (2014), FMST Yang et al. (2019), APP2 Xiao and Peng (2013), TReMAP Zhou et al. (2016), NGPST Quan et al. (2015), SmartTracing Chen et al. (2015) and Li2017 Li et al. (2017), respectively. Fig. 12 visualizes the neurons reconstructed from two testing blocks. Our method reconstructs more accurate neuronal morphology compared with other methods. In Table 3, we list the weighted average of the DSA, PDS and ESA on all testing data using different methods. The weight of each testing block is proportional to the neuron length identified in the corresponding manual annotation. From Table 3, we can observe that our method outperforms others in both PDS and ESA metrics and also achieves comparable performance in DSA metric with the best one. In particular, unlike Li et al. (2017) requiring on a strongly supervised network, our method obtains even better performance without any manual annotations. With ever-increasing number of unlabeled neuron datasets are collected, our method can easily utilize them to further improve the performance of neuron reconstruction.

4.3. Evaluation of UltraNPR on a Mouse Brain Slice

We use UltraNPR algorithm to reconstruct a neuronal population from a large-scale OM brain image. The image used in this work is shown in Fig. 1, a mouse brain slice with image resolution of $25397 \times 18516 \times 869$ (761GB in size), in which a large number of neurons are densely distributed. Considering the unique image size along three directions, the block size is set to be $1120 \times 2048 \times 869$. In addition, we introduce 300 voxels overlap between adjacent blocks to avoid false continuation and increase the robustness of the reconstruction. UltraNPR is performed on the full image and takes just over one day to reconstruct the image on a cluster computer with 64 GB of working memory and 20 NVIDIA 1080Ti GPUs. As shown in Fig. 13, a neuronal population is successfully reconstructed from the large-scale image, which consists of 5348 neurons.

Seven neurons selected from the reconstructed neuronal population are visualized in Fig. 14. These reconstructions provide detailed neuronal structures and enable further neuronal morphology analysis. In summery, our results suggest that UltraNPR is capable of reconstructing neuronal populations from noisy and large-scale OM brain images.

Neuron length, number of branches.

Since the manual annotation of neuronal populations from noisy OM images is difficult to obtain, let along the annotation

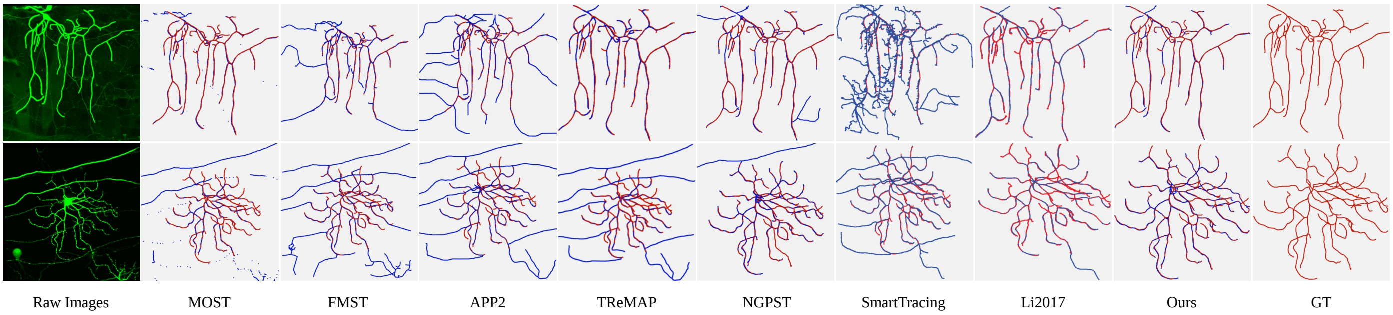

Fig. 12. Comparison of single neuron reconstruction results using MOST Wu et al. (2014), FMST Yang et al. (2019), APP2 Xiao and Peng (2013), TReMAP Zhou et al. (2016), NGPST Quan et al. (2015), SmartTracing Chen et al. (2015), Li2017 Li et al. (2017) and our PLNPR on two testing images from the BigNeuron dataset. Here, we overlay each reconstruction result in blue with the corresponding ground truth (GT) which are denoted by red. We can observe that, our method reconstructs more complete and accurate neurons compared to other methods.

Table 3. Performance comparison in terms of Precision, Recall, F-Score, Jaccard, entire structure average (ESA), different structure average (DSA) and percentage of different structures (PDS) with different methods for single neuron reconstruction on the BigNeuron test dataset.

Method	Precision	Recall	F-Score	Jaccard	ESA	DSA	PDS
MOST Wu et al. (2014)	0.619	0.361	0.456	0.295	31.730	38.211	0.633
FMST Yang et al. (2019)	0.575	0.629	0.601	0.429	17.878	23.459	0.558
APP2 Xiao and Peng (2013)	**0.799**	0.492	0.608	0.437	13.457	17.923	0.562
TReMAP Zhou et al. (2016)	0.771	0.415	0.539	0.369	11.269	17.941	0.539
NGPST Quan et al. (2015)	0.710	0.680	0.695	0.532	10.168	14.880	0.587
SmartTracing Chen et al. (2015)	0.701	0.648	0.674	0.508	8.532	11.609	0.543
Li2017 Li et al. (2017)	-	-	-	-	4.917	**7.972**	0.461
Ours	0.790	**0.707**	**0.746**	**0.595**	**4.784**	8.309	**0.451**

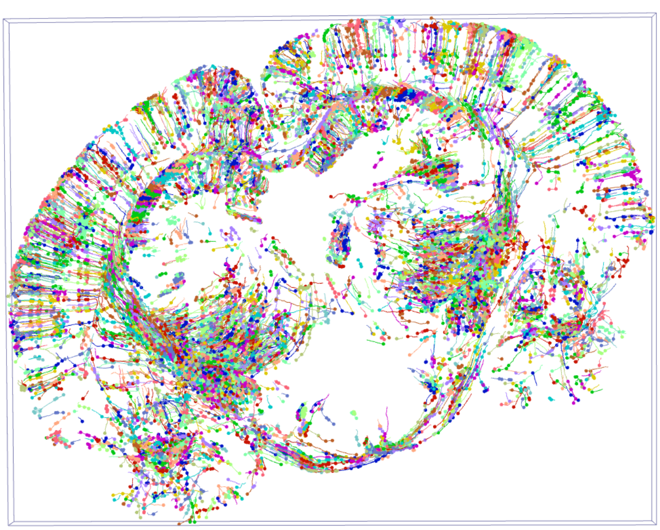

Fig. 13. The reconstruction result of neuronal populations in a large-scale 3D mouse brain slice using our UltraNPR method.

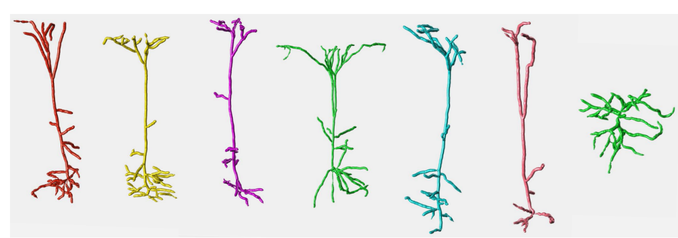

Fig. 14. Single neurons selected from the reconstructed neuronal populations in a mouse brain slice using our UltraNPR method.

of large-scale neuronal populations from a mouse brain slice. In this section, some qualitative results are visualized for verifying the effectiveness of our UltraNPR method.

5. Conclusion

In this work, we propose PLNPR, an unsupervised progressive learning framework for neuronal population reconstruction from noisy and low-quality OM image blocks. Without using any manual annotations, we take advantage of neuron tracing techniques and deep segmentation networks, and make them mutually complement and promote each other progressively. We extensively validate the proposed PLNPR for neuron reconstruction and the results demonstrate the effectiveness and superiority of our method. Furthermore, we introduce UltraNPR, a reconstruction algorithm that makes possible the neuronal population reconstruction from large-scale OM brain images. We also construct a new neuron dataset "VISoR-40" which consists of 40 OM image blocks for the evaluation of neuronal population reconstruction. This dataset will be published to facilitate further work on brain research, including but not limited to neuron counting, neuron reconstruction and neuron morphology analysis, and so on.

References

Bas, E., Erdogmus, D., 2011. Principal curves as skeletons of tubular objects. Neuroinformatics 9, 181–191.

Basu, S., Ooi, W.T., Racoceanu, D., 2016. Neurite tracing with object process. IEEE Trans. Med. Imag. 35, 1443–1451.

Cai, H., Xu, X., Lu, J., Lichtman, J.W., Yung, S.P., Wong, S.T.C., 2006. Repulsive force based snake model to segment and track neuronal axons in 3D microscopy image stacks. NeuroImage 32, 1608–1620.

Chen, H., Xiao, H., Liu, T., Peng, H., 2015. Smarttracing: self-learning-based neuron reconstruction. Brain Informatics 2, 135. doi:10.1007/s40708-015-0018-y.

Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O., 2016. 3D U-Net: learning dense volumetric segmentation from sparse annotation, in: Proc. MICCAI, pp. 424–432.

De, J., Cheng, L., Zhang, X., Lin, F., Li, H., Ong, K.H., et al., 2016. A graph-theoretical approach for tracing filamentary structures in neuronal and retinal images. IEEE Trans. Med. Imag. 35, 257–272. doi:10.1109/TMI.2015.2465962.

Dou, Q., Yu, L., Chen, H., Jin, Y., Yang, X., Qin, J., et al., 2017. 3D deeply supervised network for automated segmentation of volumetric medical images. Med. Imag. Anal. 41, 40–54.

Frasconi, P., Silvestri, L., Soda, P., Cortini, R., Pavone, F.S., Iannello, G., 2014. Large-scale automated identification of mouse brain cells in confocal light sheet microscopy images. Bioinformatics 30, 587–593. doi:10.1093/bioinformatics/btu469.

Giorgio, A., De Stefano, N., 2013. Clinical use of brain volumetry. J. Magn. Reson. Imaging 37, 1–14.

Gu, L., Zhang, X., Zhao, H., Li, H., Cheng, L., 2017. Segment 2D and 3D Filaments by Learning Structured and Contextual Features. IEEE Trans. Med. Imag. 36, 596–606.

Li, R., Zeng, T., Peng, H., Ji, S., 2017. Deep learning segmentation of optical microscopy images improves 3-D neuron reconstruction. IEEE Trans. Med. Imag. 36, 1533–1541. doi:10.1109/TMI.2017.2679713.

Liu, M., Chen, W., Wang, C., Peng, H., 2019. A Multiscale Ray-Shooting Model for Termination Detection of Tree-Like Structures in Biomedical Images. IEEE Trans. Med. Imag. 38, 1923–1934.

Liu, S., Zhang, D., Liu, S., Feng, D., Peng, H., Cai, W., 2016. Rivulet: 3D neuron morphology tracing with iterative back-tracking. Neuroinformatics 14, 387. doi:10.1007/s12021-016-9302-0.

Liu, S., Zhang, D., Song, Y., Peng, H., Cai, W., 2018. Automated 3-D Neuron Tracing With Precise Branch Erasing and Confidence Controlled Back Tracking. IEEE Trans. Med. Imag. 37, 2441–2452.

Navlakha, S., Ahammad, P., Myers, E.W., 2013. Unsupervised segmentation of noisy electron microscopy images using salient watersheds and region merging. BMC Bioinform. 14, 294. doi:10.1186/1471-2105-14-294.

Peng, H., Hawrylycz, M., Roskams, J., Hill, S., Spruston, N., Meijering, E., et al., 2015. BigNeuron: large-scale 3d neuron reconstruction from optical microscopy images. Neuron 87, 252–256.

Peng, H., Long, F., Myers, G., 2011. Automatic 3D neuron tracing using all-path pruning. Bioinformatics 27, 239–247. doi:10.1093/bioinformatics/btr237.

Peng, H., Ruan, Z., Atasoy, D., Sternson, S., 2010. Automatic reconstruction of 3d neuron structures using a graph-augmented deformable model. Bioinformatics 26, 38–46.

Peng, H., Tang, J., Xiao, H., Bria, A., Zhou, J., Butler, V., et al., 2014. Virtual finger boosts three-dimensional imaging and microsurgery as well as terabyte volume image visualization and analysis. Nature Communications 5, 4342.

Peng, H., Zhou, Z., Meijering, E., Zhao, T., Ascoli, G.A., Hawrylycz, M., 2017. Automatic tracing of ultra-volumes of neuronal images. Nature Methods 14, 332.

Petrella, J.R., Coleman, R.E., Doraiswamy, P.M., 2003. Neuroimaging and early diagnosis of alzheimer disease: A look to the future. Radiology 226, 315–336.

Quan, T., Zheng, T., Yang, Z., Ding, W., Li, S., Li, J., et al., 2013. NeuroGPS: automated localization of neurons for brain circuits using L1 minimization model. Scientific Reports 3, 1414.

Quan, T., Zhou, H., Li, J., Li, S., Li, A., Li, Y., et al., 2015. NeuroGPS-Tree: automatic reconstruction of large-scale neuronal populations with dense neurites. Nature Methods 13, 51–54.

Radojević, M., Meijering, E., 2017. Automated neuron tracing using probability hypothesis density filtering. Bioinformatics 33, 1073–1080. doi:10.1093/bioinformatics/btw751.

Santamaría-Pang, A., Hernandez-Herrera, P., Papadakis, M., Saggau, P., Kakadiaris, I.A., 2015. Automatic morphological reconstruction of neurons from multiphoton and confocal microscopy images using 3d tubular models. Neuroinformatics 13, 297–320.

Senft, S.L., 2011. A brief history of neuronal reconstruction. Neuroinformatics 9, 119–128.

Skibbe, H., Reisert, M., Maeda, S., Koyama, M., Oba, S., Ito, K., Ishii, S., 2015. Efficient monte carlo image analysis for the location of vascular entity. IEEE Trans. Med. Imag. 34, 628–643.

Skibbe, H., Reisert, M., Nakae, K., Watakabe, A., Hata, J., Mizukami, H., et al., 2019. Pat–probabilistic axon tracking for densely labeled neurons in large 3-d micrographs. IEEE Trans. Med. Imag. 38, 69–78.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R., 2014. Dropout: A simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. 15, 1929–1958.

Sümbül, U., Roossien, D., Cai, D., Chen, F., Barry, N., Cunningham, J.P., et al., 2016. Automated scalable segmentation of neurons from multispectral images, in: Proc. Adv. Neural Inf. Process. Syst., pp. 1912–1920.

Sun, K., Xiao, B., Liu, D., Wang, J., 2019. Deep high-resolution representation learning for human pose estimation, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit., pp. 5693–5703.

Wang, C.W., Lee, Y.C., Pradana, H., Zhou, Z., Peng, H., 2017. Ensemble neuron tracer for 3d neuron reconstruction. Neuroinformatics 15, 185–198.

Wang, H., Zhang, D., Song, Y., Liu, S., Gao, R., Peng, H., Cai, W., 2018. Memory and Time Efficient 3D Neuron Morphology Tracing in Large-Scale Images, in: Proc. DICTA, pp. 1–8.

Wang, H., Zhu, Q., Ding, L., Shen, Y., Yang, C.Y., Xu, F., et al., 2019. Scalable volumetric imaging for ultrahigh-speed brain mapping at synaptic resolution. National Science Review URL: https://doi.org/10.1093/nsr/nwz053, doi:10.1093/nsr/nwz053.

Wang, Y., Narayanaswamy, A., Tsai, C.L., Roysam, B., 2011. A broadly applicable 3-D neuron tracing method based on open-curve snake. Neuroinformatics 9, 193–217.

Wu, J., He, Y., Yang, Z., Guo, C., Luo, Q., Zhou, W., et al., 2014. 3D BrainCV: Simultaneous visualization and analysis of cells and capillaries in a whole mouse brain with one-micron voxel resolution. NeuroImage 87, 199–208.

Xiao, H., Peng, H., 2013. APP2: automatic tracing of 3D neuron morphology based on hierarchical pruning of a gray-weighted image distance-tree. Bioinformatics 29, 1448–1454. doi:10.1093/bioinformatics/btt170.

Xu, K., Su, H., Zhu, J., Guan, J., Zhang, B., 2016. Neuron segmentation based on CNN with semi-supervised regularization, in: Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops, pp. 1324–1332. doi:10.1109/CVPRW.2016.167.

Yang, J., Gonzalez-Bellido, P.T., Peng, H., 2013. A distance-field based automatic neuron tracing method. BMC Bioinform. 14, 93. doi:10.1186/1471-2105-14-93.

Yang, J., Hao, M., Liu, X., Wan, Z., Zhong, N., Peng, H., 2019. FMST: an automatic neuron tracing method based on fast marching and minimum spanning tree. Neuroinformatics 17, 185–196. doi:10.1007/s12021-018-9392-y.

Yu, L., Cheng, J.Z., Dou, Q., Yang, X., Chen, H., Qin, J., et al., 2017. Automatic 3D cardiovascular mr segmentation with densely-connected volumetric convnets, in: Proc. MICCAI, pp. 287–295.

Zhao, J., Chen, X., Xiong, Z., Liu, D., Zeng, J., Zhang, Y., et al., 2019. Progressive learning for neuronal population reconstruction from optical microscopy images, in: Proc. MICCAI, pp. 750–759.

Zhao, T., Xie, J., Amat, F., Clack, N., Ahammad, P., Peng, H., et al., 2011. Automated reconstruction of neuronal morphology based on local geometrical and global structural models. Neuroinformatics 9, 247–261.

Zhou, Z., Kuo, H.C., Peng, H., Long, F., 2018. Deepneuron: an open deep learning toolbox for neuron tracing. Brain Informatics 5, 3.

Zhou, Z., Liu, X., Long, B., Peng, H., 2016. TReMAP: Automatic 3D neuron reconstruction based on tracing, reverse mapping and assembling of 2d projections. Neuroinformatics 14, 41. doi:10.1007/s12021-015-9278-1.

Zhou, Z., Sorensen, S.A., Peng, H., 2015. Neuron Crawler: An automatic tracing algorithm for very large neuron images, in: Proc. IEEE Int. Symp. Biomed. Imag., pp. 870–874. doi:10.1109/ISBI.2015.7164009.