

Learning Multimodal Volumetric Features for Large-Scale Neuron Tracing

Qihua Chen, Xuejin Chen*, Chenxuan Wang, Yixiong Liu, Zhiwei Xiong, Feng Wu

National Engineering Laboratory for Brain-inspired Intelligence Technology and Application, University of Science and Technology of China, Hefei 230027, China.

Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230088, China.

cqh@mail.ustc.edu.cn, xjchen99@ustc.edu.cn, {wangcx01, yxliu99}@mail.ustc.edu.cn,
{zwxiong, fengwu}@ustc.edu.cn

Abstract

The current neuron reconstruction pipeline for electron microscopy (EM) data usually includes automatic image segmentation followed by extensive human expert proofreading. In this work, we aim to reduce human workload by predicting connectivity between over-segmented neuron pieces, taking both microscopy image and 3D morphology features into account, similar to human proofreading workflow. To this end, we first construct a dataset, named FlyTracing, that contains millions of pairwise connections of segments expanding the whole fly brain, which is three orders of magnitude larger than existing datasets for neuron segment connection. To learn sophisticated biological imaging features from the connectivity annotations, we propose a novel connectivity-aware contrastive learning method to generate dense volumetric EM image embedding. The learned embeddings can be easily incorporated with any point or voxel-based morphological representations for automatic neuron tracing. Extensive comparisons of different combination schemes of image and morphological representation in identifying split errors across the whole fly brain demonstrate the superiority of the proposed approach, especially for the locations that contain severe imaging artifacts, such as section missing and misalignment. The dataset and code are available at <https://github.com/Levishery/Flywire-Neuron-Tracing>.

1 Introduction

The development of various imaging techniques, such as electron microscopy (EM), optical microscopy (OM), and Magnetic resonance imaging (MRI) scans, bring massive volumetric imaging data in biological and medical research fields. Segmenting 3D objects with complex structures and wide expansion in these high-resolution image volumes is non-trivial. Among various 3D segmentation tasks, connectomics, which aims to reconstruct and interpret neural circuits at the synaptic resolution, presents extreme challenges due to the huge volume of EM images and the significant morphological complexity of neurons. Notably, with the advent of advanced serial section electron microscopy imaging techniques, several EM data of mm^3 -scale volumes have been published for the connectomics community, with the data sizes ranging from terabyte to petabyte (??). Many

automated volumetric image segmentation methods, either two-stage (?????) or end-to-end (?), have been developed for neuron reconstruction. While only exploiting the pixel-level local context in limited regions at the nanometer resolution, these methods have difficulty segmenting complete neurons while a single neuron typically spans a cable length of over one thousand micrometers. When applied to large-scale serial section EM volume, complete neuron reconstruction becomes even more challenging due to various imaging artifacts, such as section missing and misalignment. Since a few merge errors could lead to several incorrectly merged neuronal processes and correcting the split errors is more straightforward, existing methods typically follow the consensus of over-segmentation where a neuronal process is segmented into many small fragments (??). For example, a single neuron from a fly brain (?) is usually segmented into hundreds of fragments by Flood-filling networks (FFN) (?). To reconstruct more complete neurons, intensive human proofreading efforts are required in current neuron tracing pipelines (??). Starting from the over-segmentation result, proofreaders trace complete neurons by inspecting the continuity in 3D morphology and image features between fragments and merging fragments when necessary, as Fig. 1 shows.

A few methods have been developed to automatically detect and correct split and merge errors from the automatic segmentation (??). ? reduce split errors by extracting biologically-inspired graphs that indicate potential connectivity of over-segments using hand-designed geometric constraints. They train a 3D convolutional neural network (CNN) to predict the probability of merging based on segment morphologies. However, their method is trained and tested on small densely annotated blocks and not well generalized to various brain regions. Moreover, morphology is insufficient for reliable connectivity prediction as human proofreaders have to frequently check the 3D morphology as well as EM images of two neighboring segments during long-range neuron tracing.

In this paper, we propose a novel approach for predicting connectivity between over-segmented neuron pieces by taking both microscopy images and 3D morphology features into account, following human proofreading principles. Long-range 3D morphology could provide rough global information, while sophisticated voxel-wise image features

*Corresponding author

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

capture fine-grained evidence of continuity within adjacent local regions for reliable predictions. However, learning effective representations for the two modalities and fusing them are non-trivial. We propose a novel connectivity-aware contrastive learning method to learn dense volumetric EM image embedding from large-scale segment connectivity annotations. Then we fuse the EM image features with the 3D volume or surface representations to conduct connectivity prediction. The contributions of this work include:

- We introduce a dataset that contains millions of connecting segment pairs across the whole fly brain region. The dataset is three orders of magnitude larger than existing datasets for neuron segment connection. Our dataset is comprehensive and could facilitate the development of automatic neuron-tracing techniques.
- We propose a novel connectivity-aware contrastive learning method to learn dense volumetric EM image embedding from sparse segment connectivity. The image embedding can be fused with any point- or voxel-based 3D representations for reliable split error correction.
- Different combination schemes of image and morphological representation are extensively evaluated for split error identification across the whole fly brain, demonstrating the superiority of our proposed approach.

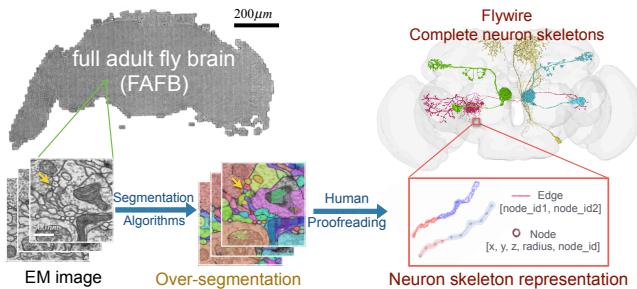


Figure 1: The pipeline of large-scale neuron reconstruction consists of EM image over-segmentation and human proofreading for complete neuron reconstruction. Each neuron is represented by a tree-structure skeleton.

2 Related Work

Voxel-based EM Neuron Segmentation. Given a terabyte- or petabyte-scale EM image volume, many learning-based methods have been proposed for automatic segmentation. A popular series of methods train voxel-based convolutional neural networks to learn affinity maps (???) and then apply non-parametric watershed transformation and agglomeration (??). Flood-filling networks (FFN) (?) combine these two steps into one by gradually expanding segments from seed voxels. Compared to affinity-based methods, FFN outperforms by a substantial margin, but at the expense of two orders of magnitude higher computational complexity. In this work, we choose to correct the split errors from FFN segmentation results, instead of modifying the dense voxel-wise segmentation.

Error Correction Methods. Although the aforementioned approaches determine most cell boundaries correctly, the remaining connection errors still require extensive human proofreading and correction. A few methods attempt to detect and correct errors automatically. ? firstly study error detection and correction for 3D neuron reconstruction. They train a 3D CNN to predict the map of split and merge errors and formulate error correction as an object mask pruning task. Other works focus on merge error correction using predefined neuron structural rules (??). ? apply biological prior to detect potential split errors, and then employ 3D CNNs to predict connection probability between the candidate pairs based on the segment morphology. We further explore the split error correction problem by introducing a comprehensive dataset and extensively studying different image and morphology representations for connectivity prediction.

Deep Metric Learning. The goal of metric learning is to learn a distance metric or embedding vectors such that similar samples are pulled closer and dissimilar samples are pushed away. Deep metric learning with pairwise (?) or triplet loss (?) performs favorably in image retrieval (?) and geo-localization (?). In the semantic and instance segmentation area, many algorithms employ metric learning to explore structural relations between pixels in the embedding space (???). The extracted pixelwise vectors of dense image embedding is significant helpful for downstream tasks. In this work, we also employ dense embeddings as a delicate image feature representation. The embedding vector of each voxel captures its neighboring appearance features, subsequently fused with point or voxel-based morphological representations to predict segment connectivity. Different from existing dense metric learning approaches, without requiring pixel-wise annotations, our method learns dense embeddings from sparse segment connectivity directly through a novel connectivity-aware contrastive loss.

3 Task Formulation and Dataset

Fig. 1 briefly illustrates the general pipeline of large-scale neuron reconstruction from EM image volume and various data used in our work. The huge EM image volume is first over-segmented into a set of 3D segments using an off-the-shelf 3D segmentation approach. Subsequently, human proofreading is performed to obtain complete neurons that usually span several brain regions. The reconstructed neurons are commonly represented by tree structures composed of nodes and edges, either directly traced using annotation software such as CATMAID (?) or skeletonized from proofread segment surface using skeletonization algorithms such as TEASAR (?). For each neuron, we register the over-segmented fragments with the neuron skeleton and obtain the connectivity relations between segments as the ground truth for training and testing.

In typical connectomics analysis workflows, human tracers start tracing from an interested neuron branch segment based on the over-segmentation results. They identify the truncation point by examining its 3D surface mesh. Subsequently, they magnify the questionable area and deter-

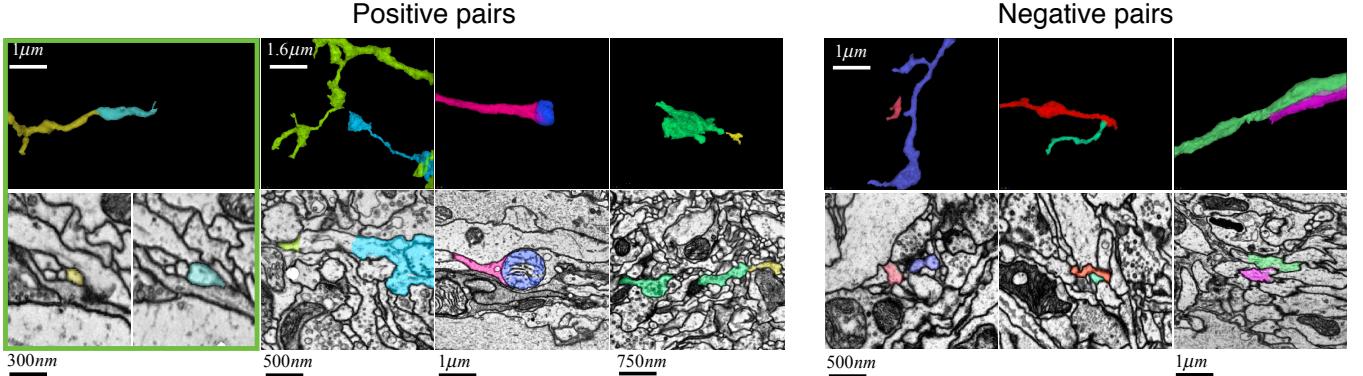


Figure 2: Example of sample pairs from FlyTracing. The first row shows their 3D meshes (segment instances denoted by color), with the representative EM image slices shown below. To determine whether two segments belong to the same neuron, human tracers frequently cross-reference their 3D meshes and adjacent EM image slices.

Table 1: Overview of the datasets used in (?) and our dataset **FlyTracing**. N/A denotes that the dataset is not public.

Dataset	Size (μm^3)	Method	# Seg. Pairs
PNI	5.0×10^3	Affinity	N/A
Kasthuri	5.5×10^2	Affinity	1.8×10^3
SNEMI3D	5.5×10^2	Affinity	$\sim 10^3$
FlyTracing	3.2×10^6	FFN	1.6×10^6

mine which segment adjacent to the truncation point maintains neuronal continuity with the initial segment of interest, achieved by cross-referencing their 3D meshes and adjacent EM image slices. For example, to trace the yellow branch in Fig. 2 (in the green box), a human tracer may zoom in to the terminal region, focusing on the lower-left image section. Then, the tracer transits to the adjacent section and sees that the area previously occupied by the yellow segment is now taken over by the blue segment, suggesting a potential connection between them. Typically, tracers proceed to verify their morphological continuity before merging.

To relieve the human proofreading workload for huge EM volumes, we propose to predict the connection probability of two segments, S_a and S_b , considering their 3D morphology and the EM images from adjacent sections, mimicking the human tracing behavior. We learn the prediction function f from a set of connected segment pairs ($f(S_a, S_b) = 1$) that should be merged during proofreading and unconnected pairs ($f(S_a, S_b) = 0$).

Dataset Construction

Dataset Overview. Available datasets for neural segment connection tasks mainly originated from densely annotated blocks, limited in scale and diversity. In contrast, we extract segment connectivity from the crowd-sourced proofreading results throughout an entire fly brain. As Table 1 shows, our dataset **FlyTracing** surpasses existing datasets by three orders of magnitude, regarding the volume size and number of connected segment pairs. The source EM im-

ages for FlyTracing are from a complete adult Drosophila brain, imaged at $4 \times 4\text{nm}$ resolution and sectioned with the thickness of 40nm , known as the “full adult fly brain” (FAFB) dataset (?). The image sections are first preprocessed through local re-alignment and irregular section substitution, and segmented through a multi-scale FFN segmentation pipeline (?), referred to as FAFB-FFN1. The proofread neuron skeletons are supplied by FlyWire (?). Despite the availability of affinity-based automatic segmentation from FlyWire, we choose FAFB-FFN1 due to its adherence to over-segmentation consensus, i.e., fewer merging errors than affinity-based segmentation results.

Segment-Neuron Registration. To generate the ground truth of connectivities of EM segment pairs, we register the FFN segmentation results with the proofread neuron skeletons. We design an automatic EM segment-neuron registration method that can be applied to any large-scale connectomics datasets with proofread neuron skeletons and over-segmentation results. With permission from Flywire, we obtain the surface meshes of the proofread neurons and skeletonize them using the skeletor tool (?). Since we focus on tracing neurites with tree-like structures, we cut off the cell body fibers from the neuron segments.

To register the massive over-segmented fragments with a human proofread neuron, given its neuron skeleton T_n , we associate each skeleton node with its nearest segment. Since the extracted skeletons and EM image segmentation results inevitably contain errors and noises, a few nodes are occasionally assigned to segments that do not belong to the right neuron. To mitigate the assignment errors, we calculate the chamfer distance from the segment skeleton T_S to the neuron skeleton T_{neu} , subsequently discarding the segments and their corresponding nodes whose $CD(T_S, T_{neu}) > 2\bar{r}$. Here, \bar{r} denotes the estimated average radius of the local branch: $\bar{r} = \frac{1}{|\Omega_S|} \sum_{i \in \Omega_S} r_i$, where Ω_S is the set of all the neuron skeleton nodes associated to segment S . Once every node is assigned with a corresponding segment, we traverse along the neuron skeleton and identify edges between nodes

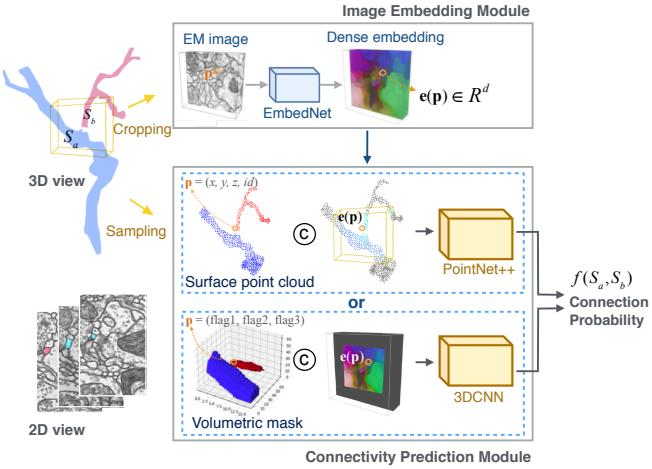


Figure 3: Our connectivity prediction framework fuses local volumetric image features extracted by the EmbedNet with 3D morphology, optionally represented by point cloud or volumetric masks.

that are assigned to different segments as bridging edges:

$$E_{\text{bridge}}(T_{\text{neu}}) = \{\text{edge}(\mathbf{v}_i, \mathbf{v}_j) | S_a \neq S_b\}, \quad (1)$$

where $\mathbf{v}_i, \mathbf{v}_j$ represent two adjacent nodes in T_{neu} , and S_a, S_b are their corresponding segments, indicating S_a and S_b should be merged as the same neuron via $\text{edge}(\mathbf{v}_i, \mathbf{v}_j)$.

Consequently, we collect the entire bridging edge set from 23,769 proofread neurons. The average count of bridging edges per neuron is 810. For each bridging edge $\text{edge}(\mathbf{v}_i, \mathbf{v}_j)$, assuming $\mathbf{c}_{i,j}$ is the midpoint of \mathbf{v}_i and \mathbf{v}_j , we add random shift to the coordinate of $\mathbf{c}_{i,j}$ and obtain $\hat{\mathbf{c}}_{i,j}$ as the estimated truncated point. The segment pair (S_a, S_b) is annotated as a positive connecting pair ($f(S_a, S_b) = 1$). Any segment S_c that is located in the cube of size $H \times W \times D$ centered at $\hat{\mathbf{c}}_{i,j}$ and $S_c \neq S_b$ is labeled with S_a as a negative pair ($f(S_a, S_c) = 0$).

Training and Test Block Partition. The positive segment pairs across the entire fly brain are partitioned into blocks based on their location in the brain, each spanning a volume size of $26 \times 26 \times 1\mu\text{m}^3$. We select 4,000 blocks, each of which contains a minimum of 350 positive segment pairs for training and testing. 1,000 blocks are selected randomly as the training and validation set for the image embedding network and pairwise connectivity prediction models. The rest 3,000 blocks are used for testing of connectivity prediction.

4 Our Method

We predict the connection probability of a pair of segments by integrating image embedding with their 3D morphological features. We design a novel connectivity-aware contrastive learning method to learn dense image embedding from sparse segment connectivity annotations.

3D Morphological Representation of Segments. The geometric property of segments provides rough clues for connection. For instance, certain patterns like L-shaped or

arrow-shaped junctions are uncommon neuronal structures. There are several 3D representation forms for expressing segment morphology. For example, EdgeNetwork (?) uses volumetric masks for segments and applies 3D convolutional neural network (3DCNN) to encode the segment morphology. Each voxel carries three labels indicating whether it belongs to segments S_a, S_b , and $S_a \cup S_b$ respectively. Besides the 3D segment masks, we also explore the surface point cloud as a more efficient morphology representation. Unlike 3DCNNs that apply dense operation for both surface voxel and inner voxels of the target segments, point-based models only process sparse surface points. Centering at the truncation point $\hat{\mathbf{c}}_{i,j}$ of the segment to trace, we crop a 3D volume and extract the segment section contours on each image slice and collect all the contour points to compose a point clouds. Each point $\mathbf{p} = (x, y, z, id)$, where (x, y, z) refer to the physical coordinate and $id \in \{0, 1\}$ signifies surface points from whether S_a or S_b . The point clouds are further downsampled by farthest point sampling (FPS) to a fixed number of points and normalized in a unit cube. We employ PointNet++ (?) as the morphological feature encoder and connectivity prediction model for the point cloud inputs.

Fusing Image Features with 3D Representations. The segment shape might be distorted due to image distortions (e.g. misalignment between adjacent EM sections) and segmentation errors. Consequently, human tracers rely on the original EM images as the decisive evidence. Sometimes the evidence could lie on just a few voxels. Therefore, we design an image embedding network called EmbedNet to generate delicate and discriminative features for every voxel. From a local EM image cube, our EmbedNet extracts a k -dimensional image feature vector for each voxel. To predict the segment connectivity, the image features are integrated with the point geometric features or voxel masks optionally, as illustrated in Fig. 3. For the volume representation, the segment masks and image embedding vectors are concatenated directly voxel by voxel. Regarding the surface point cloud, we use the average embedding vectors in a $7 \times 7 \times 3$ neighborhood of each point \mathbf{p} as its local image embedding $e(\mathbf{p})$. The k -D average embedding feature is then concatenated to (x, y, z, id) of each surface point.

Learning Volumetric Image Embedding. To learn the image feature representations, we follow the typical supervised contrastive learning framework (???) with a contrastive loss that encourages close embeddings between connected segments and distinct embeddings of segments from different neurons. Fig. 4 shows the pipeline of our EM image embedding learning. Existing dense embedding learning methods for segmentation usually apply a loss that mainly consists of two parts: intra-cluster variance that measures the deviation of the embeddings of all segment voxels from the cluster center and inter-cluster distance that forces distinct clusters. In comparison, we focus more on segment-level connectivity rather than voxel-wise segmentation. Consequently, we design a novel segment connectivity-oriented contrastive loss to enforce the similarity of the mean embeddings between segments belonging to the same neuron.

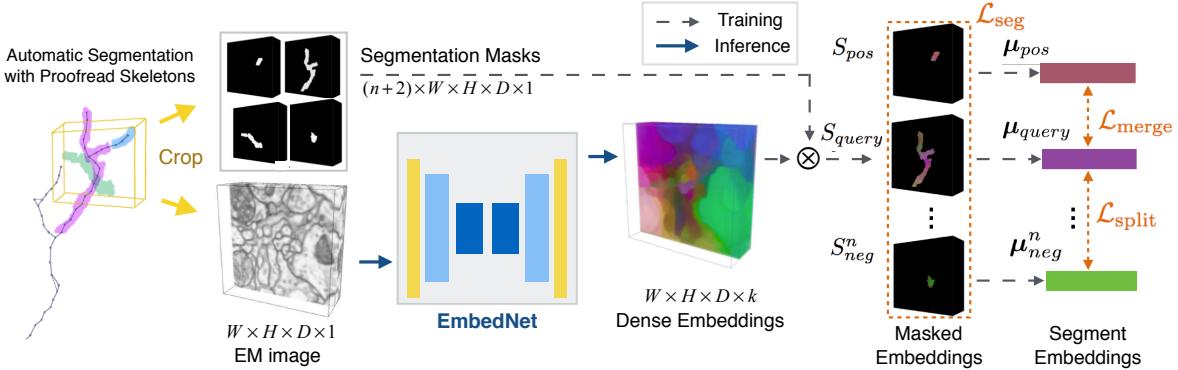


Figure 4: Our volumetric image embedding network. For a pair of adjacent segments, a small EM volume centered at the truncation point is cropped and fed into EmbedNet to extract per-voxel k -dimensional embeddings. The EmbedNet is trained via a contrastive loss based on the pairwise segment connectivity of the query segment and the other neighboring segments.

Our connectivity-oriented contrastive loss consists of a merge loss for positive pairs and a split loss for negative pairs. As shown in Fig. 4, assuming we have a positive connection pair that contains a query segment S_{query} and a positive segment S_{pos} from the same neuron, we crop an EM cube with the over-segmentation masks. Then n segments $\{S_{neg}^1, \dots, S_{neg}^n\}$ are randomly sampled from adjacent areas to form negative connection pairs with S_{query} and S_{pos} respectively. As the EmbedNet extracting per-voxel image embedding feature e , for each segment that contains M voxels within the cropped cube, we compute the mean embedding feature as the segment embedding: $\mu = \frac{1}{M} \sum_{i=1}^M e_i$. Then the merge loss and split loss are computed by:

$$\begin{aligned} \mathcal{L}_{\text{merge}} &= \|\mu_{query} - \mu_{pos}\|^2, \\ \mathcal{L}_{\text{split}} &= \frac{1}{n} \sum_{i=1}^n \max(2\delta_d - \|\mu_{query} - \mu_{neg}^i\|, 0)^2 \\ &\quad + \frac{1}{n} \sum_{i=1}^n \max(2\delta_d - \|\mu_{pos} - \mu_{neg}^i\|, 0)^2. \end{aligned} \quad (2)$$

This design enables our EmbedNet to make use of the large-scale pairwise connection dataset and learn robust features against diverse imaging quality. However, the learning of dense embedding is difficult to converge with only sparse segment-level connectivity supervision. Therefore, we follow existing segmentation-based embedding methods to further encourage the embedding to form compact clusters within each segment. Without dense segmentation annotations for each complete neuron, which are expensive to manually label for a large-scale dataset with diverse image characteristics, we use the automatic over-segmentation results of EM images as pseudo-segmentation masks and adopt the segmentation clustering loss defined in (?):

$$\mathcal{L}_{\text{seg}} = \mathcal{L}_{\text{int}} + \mathcal{L}_{\text{ext}} + \gamma \mathcal{L}_{\text{reg}}, \quad (3)$$

where $\gamma = 0.001$.

The overall loss is composed of the segment connectivity contrastive loss and the voxel segmentation clustering loss:

$$\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{\text{merge}} + \lambda_2 \mathcal{L}_{\text{split}} + \lambda_3 \mathcal{L}_{\text{seg}}. \quad (4)$$

During training, we use $\lambda_1 = 0.1$, $\lambda_2 = 1$ and $\delta_d = 1.5$. Since \mathcal{L}_{seg} is contradictory with $\mathcal{L}_{\text{merge}}$, we initialize λ_3 to 1 and linearly decrease it to 0.2.

5 Experiments and Results

Model Configurations. Our EmbedNet follows the architecture of residual symmetric U-Net (?), which tackles anisotropic serial-sectioning EM images with mixed 2D and 3D convolutional layers. We add a Squeeze-and-Excitation block (?) to each scale to enhance the embedding expression capability. We set the input volume size as $129 \times 129 \times 17$ under the voxel resolution of $16 \times 16 \times 40 \text{ nm}^3$, and the embedding dimension $k = 16$. The EdgeNetwork (?) employs a 3DCNN composed of three convolutional layers with filter sizes 16, 32, and 64, respectively, taking a cube with a side length of 1200 nm and resized to $52 \times 52 \times 18$ as the input. The point clouds are sampled within a cube in size of 2560 nm , and down-sampled to 2048 points by FPS.

Training Details. We train the embedding network with AdamW optimizer for $500k$ iterations with a batch size of 8, and apply data augmentation including random rotation, rescale, flip, and grayscale intensity augmentation. The initial learning rate is set to 0.002 with warmup and a step decay scheduler. The number of negative sample pairs $n = 20$. To enhance the learning from EM images with section missing and misalignment, we further fine-tune our EmbedNet on a subset of training blocks where the embedding network has a higher loss at the first round of training.

The EdgeNetworks (with and without image embedding features) are trained with AdamW optimizer for $45k$ steps with a batch size of 128. The PointNet++ models are trained with AdamW optimizer for $200k$ steps with a batch size of 92. Because in real neuron tracing scenarios, there are much more adjacent segment pairs that do not belong to the same neuron, we train our connectivity prediction network with more negative segment pairs by setting the ratio between positive samples and negative samples to 3 : 7. All the 422k positive segment pairs in the 1000 training data blocks are included during training and validation.

Table 2: Performance comparison on our FlyTracing test blocks.

Dataset	Avg. on 3K test blocks			Misalignment			Missing sections			Mixed degradation		
	Rec.	Prec.	F1	Rec.	Prec.	F1	Rec.	Prec.	F1	Rec.	Prec.	F1
Method												
Connect-Embed o.	0.890	0.926	0.908	0.738	0.886	0.805	0.890	0.937	0.914	0.810	0.907	0.856
	0.859	0.969	0.911	0.626	0.944	0.752	0.690	0.958	0.802	0.699	0.960	0.809
	0.881	0.970	0.923	0.658	0.946	0.776	0.748	0.952	0.838	0.713	0.960	0.818
	0.870	0.969	0.916	0.668	0.937	0.782	0.766	0.945	0.846	0.749	0.955	0.840
EdgeNetwork + Intensity + Seg-Embed + Connect-Embed	0.883	0.971	0.925	0.673	0.937	0.783	0.748	0.960	0.841	0.743	0.962	0.839
	0.847	0.939	0.891	0.628	0.893	0.737	0.693	0.895	0.781	0.672	0.903	0.771
	0.865	0.949	0.905	0.646	0.911	0.756	0.767	0.892	0.825	0.748	0.914	0.823
	0.914	0.954	0.934	0.739	0.930	0.824	0.812	0.924	0.864	0.816	0.914	0.862
	0.932	0.972	0.952	0.834	0.941	0.884	0.909	0.963	0.935	0.883	0.949	0.915

Results of Segment Connectivity Prediction

Metrics. To evaluate the segment connectivity prediction performance, we use the recall and precision metrics. On the 3000 testing blocks, all the 1,178k positive segment pairs are used for evaluation. For each positive pair, we pick one segment as the query segment of the pair and randomly sample one of its neighboring segments from another neuron near the truncation point to form a negative pair. Two segments S_a, S_b are determined as connected as the same neuron if $f(S_a, S_b) > 0.5$.

Comparison of Multiple Methods. We evaluate various combinations of image and morphology representations and report the results in Table 2. For the image features, we also evaluate ‘Intensity’ and ‘Seg-Embed’. ‘Intensity’ corresponds to the raw voxel intensity of the EM image, while ‘Seg-Embed’ refers to the image embedding approach for neuron segmentation proposed in (?). In our experiment, the ‘Seg-Embed’ embedding is trained using dense neuron segmentation annotation from the CREMI dataset, sourced from three sub-volumes of FAFB. Notably, the embedding in (?) serves as a surrogate of the affinity map for neuron segmentation, rather than for correcting segmentation split errors. Image features are not exploited in the vanilla ‘EdgeNetwork’ or ‘PointNet++’ model. In addition to these combinations, we also present the prediction performance using single modalities. ‘Connect-Embed o.’ denotes that we determine $f(S_a, S_b) = 1$ if the distance between the mean embedding of the two segments $\|\mu_{query} - \mu_{candidate}\| < \delta_d$.

As Table 2 shows, the combination of point cloud representation and our connectivity-aware embedding “+Connect-Embed” performs the best. Compared with 3D voxel-based model EdgeNetwork, PointNet++ reaps greater advantages from incorporating image features. Since 3DCNNs suffer cubical computation increases when enlarging the input volume size, the EdgeNetworks have to consider the trade-off between input resolution and field of view, preventing them from getting the advantages of local fine-grained image features and global morphological features simultaneously. On the contrary, our proposed fusion of point cloud with image embedding effectively preserves local sophisticated image characteristics, while encompassing mor-

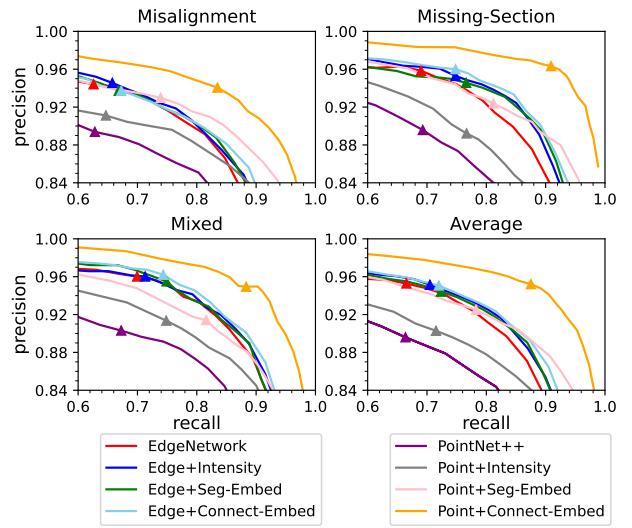


Figure 5: Precision-recall curves of different models on challenging blocks with image degradation. The triangle markers denote the performance using threshold $f(S_a, S_b) > 0.5$.

phology information across a broad field of view within a fixed number of input points. In addition, among the three manners of image feature encoding, including intensity only, segmentation-based embedding, and our connectivity-aware embedding, our embedding is the most effective for segment connectivity prediction.

Particularly for the EM volume blocks that suffer from severe image degradations, such as misalignment or section missing, the proposed PointNet++ (+Connect-Embed) yields remarkable enhancements to competitors. Among the 3,000 test blocks, we select 10 blocks containing 300 ~ 800nm misalignment, 10 blocks containing 2 ~ 3 missing sections, and 5 blocks suffering from both degradations (typically one section missing followed by misalignment of 200 ~ 300nm). Fig. 5 depicts the precision-recall curves of the models on the low-quality EM blocks. Notably, the curve of our proposed feature combination consistently outperforms the others by a substantial margin. This significant improvement is mainly brought by the proposed embeddings that can compensate for the information loss of seg-

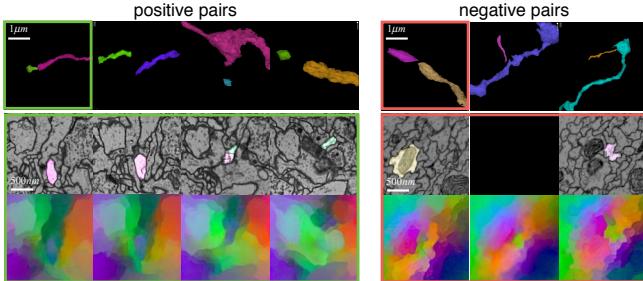


Figure 6: Visualization of several segment pairs whose connection relations are correctly predicted by model PointNet++ with ‘Connect-Embed’. In the two lower rows, we show the representative EM image sections and corresponding embeddings of the samples in the color boxes.

ment morphology at the regions with image degradation and shape distortion. The raw voxel intensity retains only sparse low-level information from the image when combined with the point cloud, therefore providing limited improvement. On the other hand, without the supervision of pair-wise contrast of segments throughout the brain, Seg-Embed (?) fails to generate robust embeddings against imaging artifacts. In contrast, our embedding approach provides both structural and fine-grained image features.

Embedding Visualization. In Fig. 6, we show two sets of segment pairs whose connection relationships are correctly classified by our PointNet++ with ‘Connect-Embed’ model. We visualize the embeddings using PCA to project the high-dimensional embedding vector onto the RGB color space. Without the image embedding, the PointNet++ model fails to predict the connectivity of these segment pairs because of the ambiguous morphology. The positive pairs are often subject to image degradation, resulting in morphological distortions or too small segments without sufficient evidence for connection. On the other hand, the morphology of misclassified negative pairs is hard to distinguish from positive connectivity pairs. In contrast, our embedding provides distinctive clues for neuron tracing, effectively complementing the distorted morphological features. Despite the variations in the location and appearance of the neurons across sections, our embedding remains discriminative and consistent.

Ablation Study of EmbedNet. We measure the discriminative ability of EmbedNet under different loss configurations. The discriminative ability is defined as the rank of $\|\mu_{query} - \mu_{pos}\|$ among all $n + 1$ candidates $\{\|\mu_{query} - \mu_{pos}\|, \|\mu_{query} - \mu_{neg}^1\|, \dots, \|\mu_{query} - \mu_{neg}^n\|\}$. We report the mean rank on 3 test blocks. The adaptive weighting strategy performs the best since L_{seg} gives dense supervision of voxel-wise segmentation in earlier steps, while decreasing the weight enforces the model to focus on learning discriminative features from pairwise connection annotations, i.e., L_{split} and L_{merge} , in later steps.

Tracing Result on FAFB Mushroom Body. We also evaluate our proposed model PointNet++ with the ‘Connect-Embed’ within a real tracing scenario on the FAFB dataset.

Table 3: Discriminative ability (the lower the better) of the embedding network with different weights of L_{seg} (λ_3), where ∞ indicates that the model is trained with L_{seg} only, and ‘Adaptive’ denotes the decreasing strategy of λ_3 .

λ_3	Block A	Block B	Block C	Average
∞	4.49	4.18	3.52	4.07
1	2.46	2.14	1.85	2.15
0.1	1.73	1.60	1.36	1.57
0	2.34	2.07	2.04	2.15
Adaptive	1.59	1.45	1.19	1.41

Following the evaluation setting described in (?), we evaluate the segmentation with a set of 166 ground truth neuronal skeletons produced by human tracers (?) using modified expected run length (ERL) metric from (?). We crop a volume of size $64 \times 44 \times 56 \mu\text{m}^3$ from the mushroom body region which contains around 10% of the ground truth skeleton nodes and exclude the subvolumes in the training set. The candidate pairs and truncated points are estimated from the endpoints of segment skeletons as (?). Using FAFB-FFN1 as the initial EM image segmentation, with a merging threshold of 0.98, our automatic proofreading with pairwise segment connectivity prediction increases ERL by $28.8\mu\text{m}$ (a relative increase of 8.1%).

6 Conclusion

We explore automatic split error correction in the neuron reconstruction pipeline by introducing a comprehensive dataset and extensively studying different image and morphology representations for connectivity prediction. Additionally, we propose a novel connectivity-aware contrastive learning method to learn discriminative EM image embeddings. Integrating the image embeddings with point clouds shows superiority over other methods in identifying connectivity between over-segmented neuron fragments in a huge volume of electronic microscopy images of a fly brain. For the future work, we intend to study beyond pairwise connection prediction. Reasoning the segment connectivity by considering multiple candidates together and the tracing history could yield more accurate decision and facilitate fully automatic complete neuron tracing.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grant 62076230 and the Fundamental Research Funds for the Central Universities under Grant WK3490000008. We thank the Princeton FlyWire team and members of the Murthy and Seung labs, as well as members of the Allen Institute for Brain Science, for the development and maintenance of FlyWire (supported by BRAIN Initiative grants MH117815 and NS126935 to Murthy and Seung). We also acknowledge members of the Princeton FlyWire team and the FlyWire consortium for neuron proofreading and annotation.

Hic optio obcaecati delectus vitae deserunt ad quas paratur doloribus, iure fugit aut aspernatur earum dolorem

possimus? Harum facilis omnis modi quos, delectus atque ea. Facere culpa earum suscipit necessitatibus impedit repudiandae eveniet eius voluptates ex, id itaque incidente deserunt dolorum at alias soluta voluptas, ratione animi explicabo dolorem consequatur repellendus numquam.