Supplementary Materials on

Two Stream Active Query Suggestion for Active Learning in Connectomics

Zudi Lin[1], Donglai Wei[1], Won-Dong Jang[1], Siyan Zhou[1], Xupeng Chen[2⋆], Xueying Wang[1], Richard Schalek[1], Daniel Berger[1], Brian Matejek[1], Lee Kamentsky[3⋆], Adi Peleg[4⋆], Daniel Haehn[5⋆], Thouis Jones[6⋆], Toufiq Parag[7⋆], Jeff Lichtman[1], and Hanspeter Pfister[1]

[1] Harvard University [2] New York University [3] MIT [4] Google [5] University of Massachusetts Boston [6] Broad Institute [7] Comcast Research
Email: linzudi@g.harvard.edu

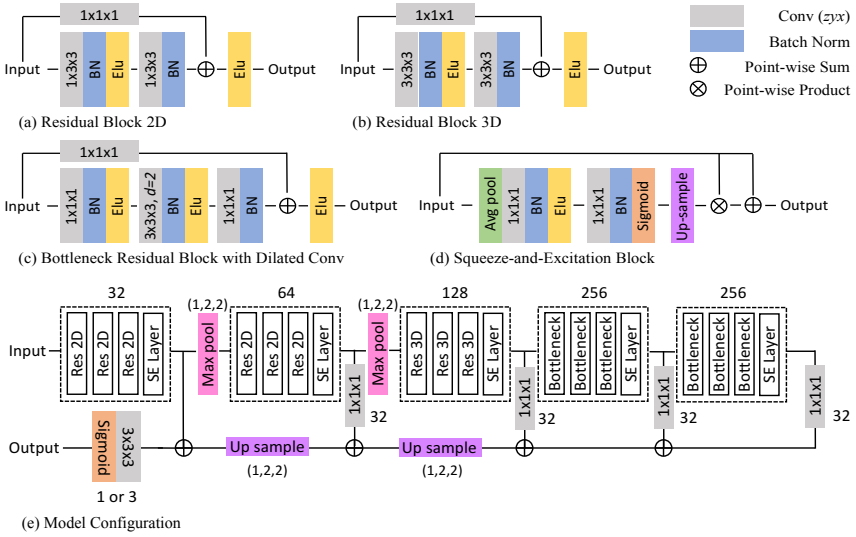

Fig. S-1. The architecture details of our base model. Due to the anisotropy nature of the electron microscopy (EM) image stacks, the resolution of the voxels along the z-axis is about $4\times$ lower than x, y resolution. Therefore we use 2D residual blocks at the first two stages in the network while putting stacked 3D residual blocks for the following stages after pooling modules make the voxels roughly isotropic. To ease the computational burden of 3D convolutions, we fix the maximum number of filters at a convolutional layer to be 256 (*e.g.*, the 5th block).

We thank the readers for viewing the supplementary document. We provide further details of our base model (Sec. S-1), annotation interface (Sec. S-2), EM-

⋆ Works were done at Harvard University

Table S-1. Architectures of the mask and image variational autoencoders (VAEs). Please note that latent dimension of images are larger because reconstructing original EM images is harder than reconstructing quantized segmentation masks.

Layers	Output Size	Mask VAE	Image VAE
Conv $\times 2$	112×112	3×3 kernel $\times 32$	
Pool	56×56	2×2 max pool	
Conv $\times 2$	56×56	3×3 kernel $\times 64$	
Pool	28×28	2×2 max pool	
Conv $\times 2$	28×28	3×3 kernel $\times 128$	
Pool	14×14	2×2 max pool	
Conv $\times 2$	14×14	3×3 kernel $\times 256$	
FC-Mean	1×1	20 fully-conn	1000 fully-conn
FC-Std	1×1	20 fully-conn	1000 fully-conn
FC-Dec	1×1	50176 fully-conn	
Deconv	28×28	2×2 kernel $\times 256$	
Conv $\times 2$	28×28	3×3 kernel $\times 128$	
Deconv	56×56	2×2 kernel $\times 128$	
Conv $\times 2$	56×56	3×3 kernel $\times 64$	
Deconv	112×112	2×2 kernel $\times 64$	
Conv	112×112	3×3 kernel $\times 32$	
Conv	112×112	3×3 kernel $\times 1$	

R50 dataset (Sec. S-3) and CIFAR-10 classification experiments (Sec. S-4) in this document.

S-1 Details for the Base Model

U-Net Model. For the detection module, we implement the backbone asymmetric 3D U-net model [5] with a 3D feature pyramid network (FPN) [4] (Figure S-1). At the end of each stage (several consecutive residual blocks [1] with the same number of filters), a squeeze-and-excitation block [2] is added to reweight the channel-wise features. Our squeeze-and-excitation block does not use global average pooling (GAP) identical to the original model for image classification, but use standard average pooling to reduce the x, y dimension only by $4\times$ to preserve more spatial information. The output feature maps for each stage are mapped to 32 channels with a $1 \times 1 \times 1$ convolution layer and combined with point-wise summation (also with up-sampling for low-resolution feature maps). For synapse polarity mask detection in the JWR dataset, the final convolution layer has a channel number of 3[1], while for synaptic clefts detection in CREMI challenge and mitochondria segmentation, the output channel number is 1.

Variational Autoencoder (VAE) Model. For the clustering module in query suggestion, we use the same variational autoencoder model (VAE) [3] for both mask and image input. As shown in Table S-1, we use 'Conv', 'Pool', 'FC', and

[1] To post-process synaptic polarity masks: `https://github.com/zudi-lin/pytorch_connectomics/blob/master/connectomics/utils/processing/process_syn.py`

'Deconv' to denote convolution, pooling, fully-connected, and deconvolution (or transposed convolution) layers, respectively. The output of the 'FC-Mean' layer is used as the latent vector for clustering. Note that the output of the 'FC-Dec' layer is reshaped into a $14 \times 14 \times 256$ tensor.

Learning-Loss Module. The main insight of the learning-loss module [6] is to suggest the samples with the highest estimated prediction errors of the main tasks (*e.g.*, classification, and detection). For implementation, the design choice is to combine several feature maps from the CNN backbone with global average pooling (GAP) and fully connected layers. Following the design choices of the original learning-loss module [6], we combine the outputs from the four stages, as shown in Figure S-1 (before dimension reduction denoted by $1 \times 1 \times 1$ `conv`). The number of channels of the four feature maps is $\{256, 128, 64, 32\}$, and the latent dimension of the learning-loss module is 32. Finally, a fully-connected layer maps the concatenated feature vector into a single scalar as the estimated loss of the main task. As suggested by Yoo *et al.* [6], the energy function is the relative difference between pairs of samples in a batch to discard the change of the absolute loss scale during training.

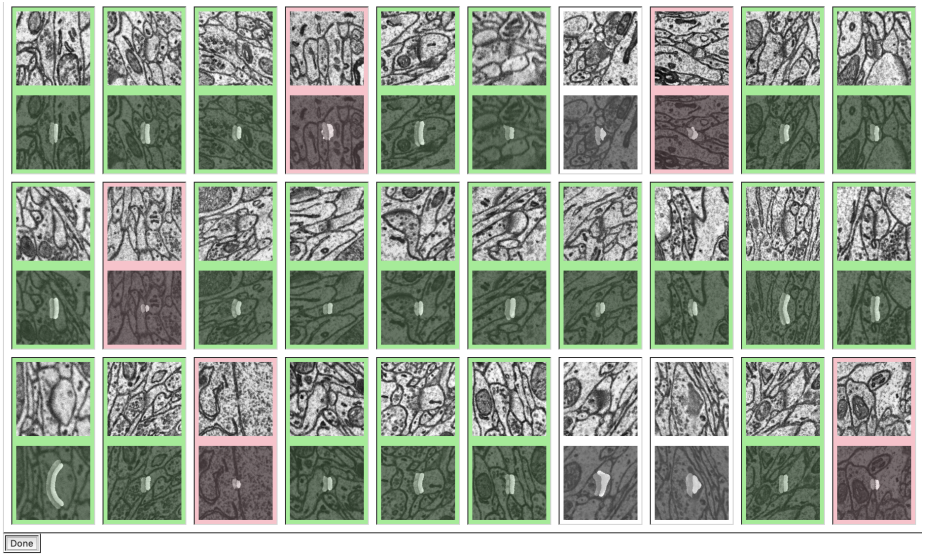

Fig. S-2. Synapse annotation interface: randomly ordered view.

S-2 Details for the Annotation Interface

As shown in Figure S-2 and S-3, we select the representative 2D slice of the 3D object (synapse in this example) and align them along the vertical axis in the

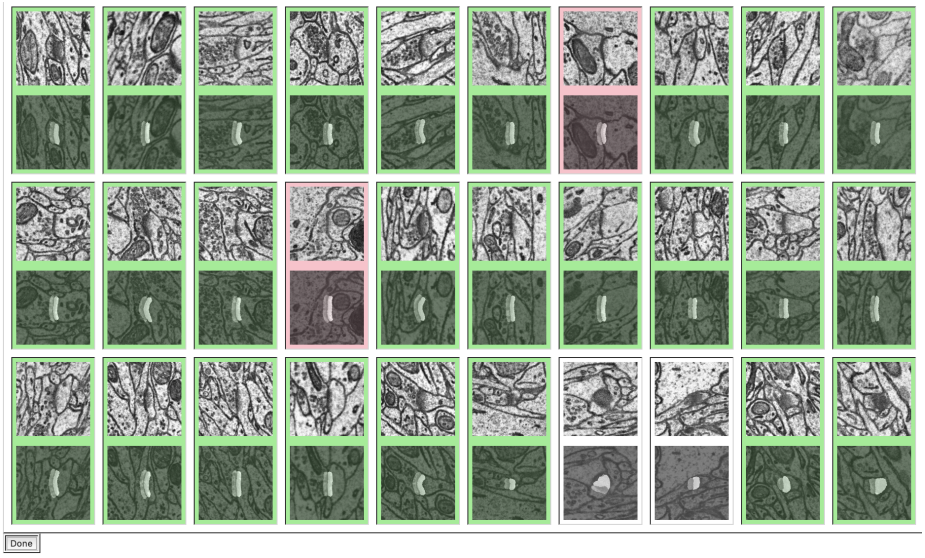

Fig. S-3. Synapse annotation interface: clustered view (**ours**).

interface for human proofreading. Both the re-aligned gray-scale image and the composite image with predicted masks are shown to annotators. The synapses are either randomly ordered or sorted by the clustering result in the mask VAE space. For each object, we first assume it to be 'correct' and mark as green. If an annotator clicks it once, the color will be switched to red and recorded as a 'wrong' prediction. If an annotator clicks it twice, it will be marked as white and annotated as an 'unsure' one. We display synapses and mitochondria in a 10×10 grid view. After finishing each page, an annotator just needs to click the 'done' button at the end of the page. For the comparison of the human annotation, we let half of the annotators annotate synapses displayed by random ordering. The other half of the annotators annotate synapses sorted in accordance with the clustering result. The unsure synapses and mitochondria are further proofread by showing multiple slides to make more confident decisions.

S-3 Details for the EM-R50 Dataset

Below is a more detailed description of the EM-R50 (**E**lectron **M**icroscopy images of **R**at with a volume size of $(\mathbf{50}\ \mu m)^3$) dataset with dense synapses and mitochondrion annotation.

Image Acquisition. We started with harvesting the whole brain tissue by transcardial perfusion of a functionally-imaged adult rat. The primary visual cortex of the rat was subsequently sectioned by a vibratome and further dissected into a 1 mm^3 tissue block. The block was then prepared using a reduced-osmium staining protocol followed by resin embedding of the entire piece of the cortex.

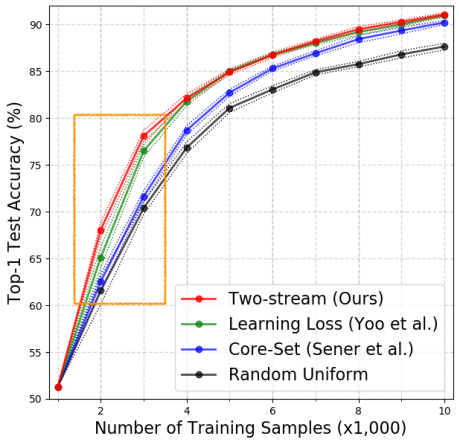

Fig. S-4. Active learning experiments on CIFAR-10 image classification. The accuracy improvement of our proposed method is most significant with a relatively small number of samples (2k and 3k out of total 50k images, highlighted with the orange rectangle), which is consistent with our observations on the EM-R50 connectomics datasets. The performance tends to saturate after ten iterations. Mean and standard deviation are shown from 5 runs.

A custom ATUMtome was used to cut and collect serial sections in 30 nm thickness on a continuous length of carbon-coated Kapton tape. Segments of tape were adhered to silicon wafers to create an ultrathin-section library. Wafers were then optically imaged for individual section positioning and focus mapping. Automated EM image acquisition was finally carried out in the determined region in each serial section at a spatial resolution of 4nm×4nm using a multi-beam scanning electron microscope throughout the wafers. We then post-process the raw images and create the EM-R50 dataset with a spatial resolution of 8×8×30 nm^3 for each voxel.

Annotation Consistency. In order to ensure the accuracy of the dataset, each detected instance is proofread by two annotators independently. We notice their agreement is over 98% for both synapses and mitochondria, which shows the accuracy of the constructed dataset.

Dataset Statistics. We annotated around 104K synapses and 72K mitochondria in the 50 cubic micron volume. The synapse density is 0.832 instance/μm^3, and mitochondria occupy 7.6% of the volume in terms of the number of pixels.

S-4 Details for the CIFAR-10 Experiment

In addition to the five-round CIFAR-10 active learning experiments shown in the main paper where the annotation budget ($\sim$ 5%) is similar to the annotation budget for the JWR50 connectomics dataset, we also conducted ten-round active

learning experiments (Figure S-4). Besides achieving better performance under the same limited budget, our method is still higher or on par with previous state-of-the-art approaches when the performance saturates.

References

1. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016) 2
2. Hu, J., Shen, L., Sun, G.: Squeeze-and-excitation networks. In: CVPR (2018) 2
3. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. ICLR (2013) 2
4. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017) 2
5. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI (2015) 2
6. Yoo, D., Kweon, I.S.: Learning loss for active learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 93–102 (2019) 3