Cross-Classification Clustering: An Efficient Multi-Object Tracking Technique for 3-D Instance Segmentation in Connectomics

Yaron Meirovitch[1,2*], Lu Mi[1*], Hayk Saribekyan[1], Alexander Matveev[3], David Rolnick[1], Nir Shavit[1,4]
[1]MIT, [2]Harvard University, [3]Neural Magic Inc., [4]Tel-Aviv University

Abstract

Pixel-accurate tracking of objects is a key element in many computer vision applications, often solved by iterated individual object tracking or instance segmentation followed by object matching. Here we introduce cross-classification clustering (3C), a technique that simultaneously tracks complex, interrelated objects in an image stack. The key idea in cross-classification is to efficiently turn a clustering problem into a classification problem by running a logarithmic number of independent classifications per image, letting the cross-labeling of these classifications uniquely classify each pixel to the object labels. We apply the 3C mechanism to achieve state-of-the-art accuracy in connectomics – the nanoscale mapping of neural tissue from electron microscopy volumes. Our reconstruction system increases scalability by an order of magnitude over existing single-object tracking methods (such as flood-filling networks). This scalability is important for the deployment of connectomics pipelines, since currently the best performing techniques require computing infrastructures that are beyond the reach of most laboratories. Our algorithm may offer benefits in other domains that require pixel-accurate tracking of multiple objects, such as segmentation of videos and medical imagery.

1. Introduction

Object tracking is an important and extensively studied component in many computer vision applications [1, 13, 14, 16, 23, 54, 57, 59]. It occurs both in video segmentation and in 3-D object reconstruction based on 2-D images. Less attention has been given to efficient algorithms performing simultaneous tracking of multiple interrelated objects [14] in order to eliminate the redundancies of tracking multiple objects via repeated use of single-object tracking. This problem is relevant to applications in medical imaging [10, 11, 22, 29, 30, 38] as well as videos [12, 43, 55, 58].

*These authors equally contributed to this work
{yaronm,lumi}@mit.edu

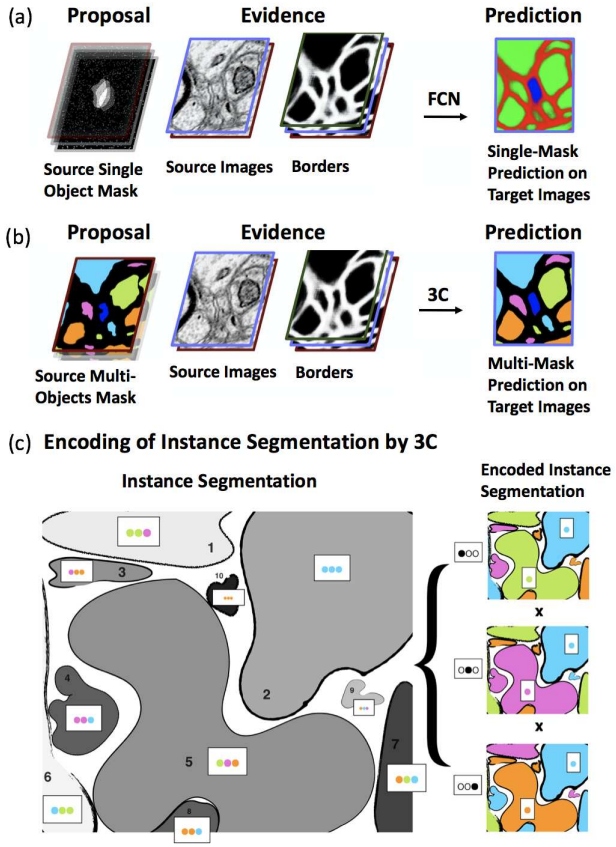


Figure 1. (a) Single-object tracking using flood-filling networks [21], (b) Multiple-object tracking using our cross-classification clustering (3C) algorithm, (c) The combinatorial encoding of an instance segmentation by 3C. One segmented image with 10 objects is encoded using three images, each with 4 object classes.

The field of connectomics, the mapping of neural tissue at the level of individual neurons and the synapses between them, offers one of the most challenging settings for testing algorithms to track multiple complex objects. Such synaptic level maps can be made only from high-resolution images taken by electron microscopes, where the sheer volume of data that needs to be processed (petabyte-size im-

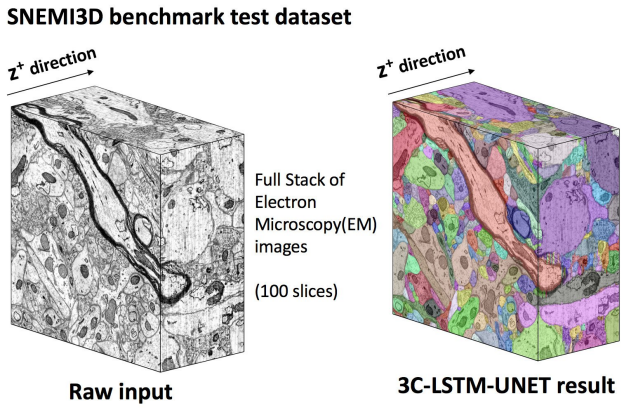


Figure 2. The raw input electron microscopy (EM) full image stack and our 3C-LSTM-UNET results in the SNEMI3D benchmark.

age stacks), the desired accuracy and speed (terabytes per hour [33]), and the complexity of the neurons' morphology, present a daunting computational task. By analogy to traditional object tracking, imagine that instead of tracking a single sheep through multiple video frames, one must track an entire flock of sheep that intermingle as they move, change shape, disappear and reappear, and obscure each other [32].

As a consequence of this complexity, several highly successful tracking approaches from other domains, such as the "detect and track" approach [14], are less immediately applicable to connectomics.

Certain salient aspects are unique to the connectomics domain: **a)** All objects are of the same type (biological cells); sub-categorizing them is difficult and has little relevance to the segmentation problem. **b)** Most of the image is foreground, with tens to hundreds of objects in a single megapixel image. **c)** Objects have intricate, finely branched shapes and no two are the same. **d)** Stitching and alignment of images can be imperfect, and the distance between images (z-resolution) is often greater than between pixels of the same image (xy-resolution), sometimes breaking the objects' continuity. **e)** Some 3-D objects are laid out parallel to the image stack, spanning few images in the z direction and going back and forth in that limited space with extremely large extensions in some image planes.

In this work, we introduce 3C, a technique that achieves volumetric instance segmentation by transferring segmentation knowledge from one image to another, simultaneously classifying the pixels of the target image(s) with the labels of the matching objects from the source image(s). This algorithm is optimized for the setting of connectomics, in which objects frequently branch and come together, but is suitable for a wide range of video-segmentation and medical imaging applications.

The main advantage of our solution is its ability, unlike prior single-object tracking methods for connectomics [21, 37], to simultaneously and jointly segment neighboring, intermingled objects, thereby avoiding redundant computation. In addition, instead of extending single masks, our detectors perform clustering by taking into account information on all visible objects.

The efficiency and accuracy of 3C are demonstrated on four connectomics datasets: the public SNEMI3D benchmark dataset, shown in Figure 2, the widely studied mouse somatosensory cortex dataset [24] (*S1*), a Lichtman Lab dataset of the V1 region of the rat brain (*ECS*), and a newly aligned mouse peripheral nervous system dataset (*PNS*), where possible, comparing to other competitive results in the field of connectomics.

1.1. Related Work

A variety of techniques from the past decade have addressed the task of neuron segmentation from electron microscopy volumes. An increasing effort has been dedicated to the problem of densely segmenting all pixels of a volume according to foreground object instances (nerve and support cells), known as *saturated reconstruction*. Note that unlike everyday images, a typical megapixel electron microscopy image may contain hundreds of object instances, with very little background ($<$10%). Below, we briefly survey the saturated reconstruction pipelines that seem to us most influential and related to the approach undertaken here.

Algorithms for saturated reconstruction of connectomics data have proved most accurate when they combine many different machine learning techniques [6, 31]. Many of these techniques use the hierarchical approach of Andres et al. [2] that employs the well-known hierarchical image segmentation framework [3, 15, 39, 47]. This is still the most common approach in connectomics segmentation pipelines: first detecting object borders in 2-D/3-D and then gradually agglomerating information to form the final objects [6, 9, 20, 27, 31, 34, 35, 52]. The elevation maps obtained from the border detectors are treated as estimators of the true border probabilities [9], which are used to define an over-segmentation of the image, foreground connected components on top of a background canvas. The assumption is that each of the connected components straddles at most a single true object. Therefore it may need to be agglomerated with other connected components (heuristically [17, 18, 26] or based on learned weights of handcrafted features [2, 27, 40, 41]), but it should not be broken down into smaller segments. Numerous 3-D reconstruction systems follow this bottom-up design [6, 7, 27, 31, 35, 40, 41, 44]. A heavily engineered implementation of hierarchical segmentation [31] still occupies the leading entry in the (still active) classical SNEMI3D connectomics contest of 2013 [5], evaluated in terms of the uniform instance segmentation correctness metrics (normalized Rand-Error [53] and Variation of Information [36]).

A promising new approach was recently taken with

the introduction of flood-filling networks (FFN; [21]) by Januszewski et al. and concurrently and independently of MaskExtend [37] by Meirovitch et al. As seen in Figure 1(a), these algorithms take a mask defining the object prediction on a source image(s), and then use a fully convolutional network (FCN) to classify which pixels in the target image(s) belong to the singly masked object of the source image(s). This process is repeated throughout the image stack in different directions, segmenting and tracking a single object each time, while gradually filling the 3-D shape of complex objects. This provides accuracy improvements on several benchmarks and potentially tracks objects for longer distances [21] compared to previous hierarchical segmentation algorithms (e.g., [6]). However, these single-object trackers are not readily deployable for large-scale applications, especially when objects are tightly packed and intermingled with each other, because then individual tracking becomes highly redundant, forcing the algorithm to revisit pixels of related image contexts many times[1]. Furthermore, the existing single-object detectors in connectomics [21, 37] and in other biomedical domains (e.g. [4, 8, 19, 48]) do not take advantage of the multi-object scene to better understand the spatial correlation between different 3-D objects. The approach taken here generalizes the single-object approach in connectomics to achieve simpler and more effective instance segmentation of the entire volume.

1.2. Contribution

We provides a scalable framework for 3-D instance segmentation and multi-object tracking applications, with the following contributions:

- We propose a simple FCN approach, tackling the less studied problem of mapping an instance segmentation between two related images. Our algorithm jointly predicts the shapes of several objects partially observed in the input.
- We propose a novel technique that turns a clustering problem into a classification problem by running a logarithmic number of independent classifications on the pixels of an image with N objects (for possibly large N, bounded only by the number of pixels).
- We show empirically that the simultaneous tracking ability of our algorithm is more efficient than independently tracking all objects.
- We conduct extensive experimentation with four connectomics datasets, under different evaluation criteria and a performance analysis, to show the efficacy and efficiency of our technique on the problem of neuronal reconstruction.

[1]Such approaches thus take time linear in the number of objects and in the number of pixels, with a large constant that depends on the object density.

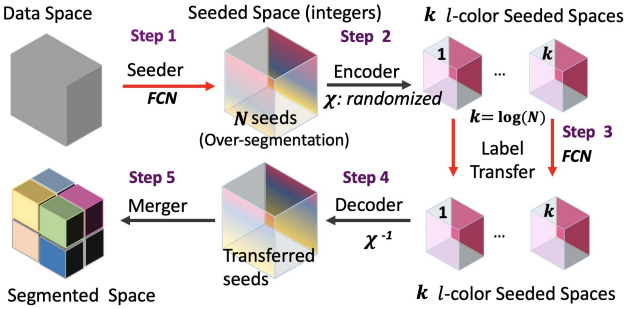


Figure 3. A high level view of our 3-D instance segmentation pipeline.

2. Methodology

We present *cross-classification clustering* (henceforth *3C*), a technique that extends single object classification approaches, simultaneously and efficiently classifying all objects in a given image based on a proposed instance segmentation of a context-related image. One can think of the context-related image and its segmentation as a collection of labeled masks to be simultaneously remapped together to the new target image, as in Figure 1(b). The immediate difficulty of such simultaneous settings is that this generalization is a clustering problem: unlike FFNs and MaskExtend (shown in Figure 1(a)), that produce a binary output ("YES" for extending the object and otherwise "NO"), in any volume, we really do not know how many classification labels we might need to capture all the objects, or more importantly how to represent those instances in ways usable for supervised learning. Overcoming this difficulty is a key contribution of 3C.

Cross-Classification Clustering: We begin by explaining the main idea behind 3C and differentiating it from single-object methods such a FFNs. We then provide a top-down sketch of our pipeline and describe how it can be adapted to other domains.

Our goal is to extend a single-object classification from one image to the next so as to simultaneously classify pixels for an *a priori* unknown set of object labels. More formally, suppose that we have images X_{prev} and X_{next}, where X_{prev} has been segmented and X_{next} must be segmented consistent with X_{prev}. Given two such images, we seek a classification function f that takes as input a voxel v of X_{next} and a segmentation s of X_{prev} (an integer matrix representing object labels) and outputs a decision label. The function f outputs a label if and only if v belongs to the object with that label in s. If s is allowed to be an over-segmentation (i.e., several labels representing the same object) then the output of f should be one of the compatible labels.

For simplicity, let us assume that the input segmentation s has entries from the label set $\{1, \cdots, N\}$. We de-

fine a new space of labels, the length-k strings over a predetermined alphabet A (here represented by colors), where $|A| = l$ and $n{=}|A|^k \geq N$ is an upper bound on the number of objects we expect in a classification. We use an arbitrary encoding function, χ, that maps labels in $\{1, \cdots, N\}$ to distinct random strings over A of length k. In the example in Figure 1(c), A is represented by $l{=}4$ colors, and $k{=}3$, so we have a total of $4^3{=}64$ possible strings of length 3 to which the $N{=}10$ objects can be mapped. Thus, for example, object 5 is mapped to the string (Green, Purple, Orange) and object 1 is (Green, Green, Purple). We can define the classification function f on string labels as the product of k traditional classifications, each with an input segmentation of labels in A, and an output of labels in A. Slightly abusing notation, let the direct product of images $\chi(s) = \chi_1(s) \times \cdots \times \chi_k(s)$ be the relabeling of the segmentation s where each image (or tensor) $\chi_i(s)$ is the projection of $\chi(s)$ in the i-th location (a single coloring of the segmentation) and $\times$ is the concatenation operation on labels in A. Then we can re-define f on $\chi(s)$ as

$$f(v, \chi(s)) = f'(v, \chi_1(s)) \times \cdots \times f'(v, \chi_k(s)), \qquad (1)$$

where each $f'(\chi_k(s))$ is a traditional classification function. The key idea is that f' is a classification of v based on an instance segmentation with l predetermined labels. In the example in Figure 1(c), even though in the map representing the most significant digit of the original objects 5 and 1, they are both Green, when we perform the classification and take the cross labeling of all three maps, the two objects are classified into distinct labels.

3-D reconstruction system: Our 3-D reconstruction system consists of the following steps (shown in Figure 3):

1) Seeding and labeling the volume with an initial imperfect 2-D/3-D instance segmentation that overlaps all objects except for their boundaries (over-segmentation).

2) Encoding the labeled seeds into a new space using the 3C rule χ.

3) Applying a fully convolutional network $\log(N)$ times to transfer the projected labeled seeds from the source image to the target images, and then take their cross labeling.

4) Decoding the set of network outputs to the original label space using the inverse 3C rule χ^{-1}.

5) Agglomerating the labels into 3-D consistent objects based on the overlap of the original seeding and the segments predicted from other sections.

To initially seed and label the volume (Step 1), we compute and label 2-D masks that over-segment all objects. For this we follow common practice in connectomics, computing object borders with FCNs and searching for local minima in 2-D on the border elevation maps. Subsequently (Step 2), we use χ to encode the seeds of each section, resulting in a k-tuple over the l-color alphabet for each seed ($k{=}5$ and $l{=}4$ in Figure 4).

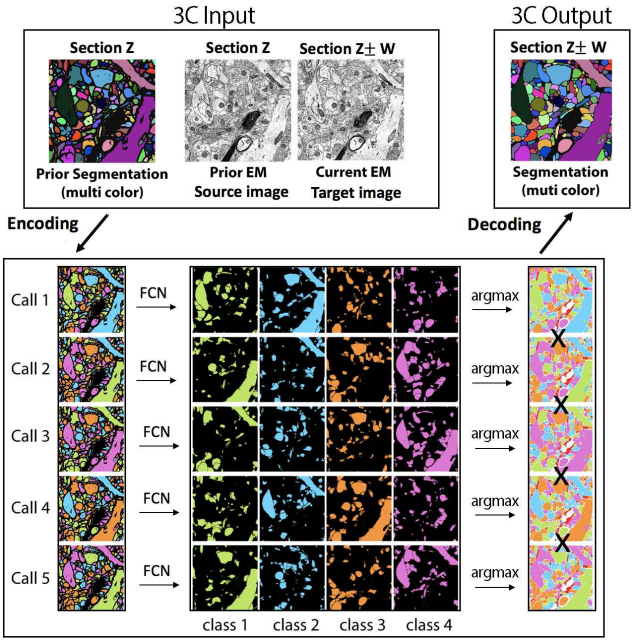


Figure 4. The instance segmentation transfer mechanism of 3C: Encoding the seeded image as k l-color images using the encoding rule χ ($k{=}\log(N)$; here $k{=}5$ and $l{=}4$). Applying a fully convolutional network k times to transfer each of the seed images to a respective target. Decoding the set of k predicted seed images using the decoding rule χ^{-1}.

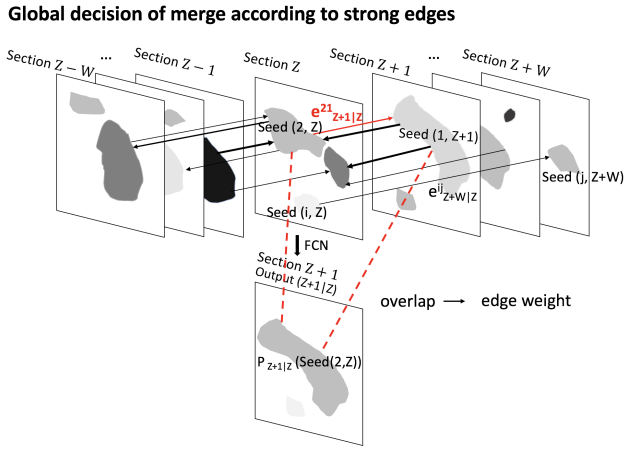


Figure 5. A schematic view of a global merge decision. The edge weight between seed i and seed j at sections Z and $Z{+}W$, respectively. $e^{ij}_{Z+W|Z}$ is calculated by the ratio of the overlapping areas of seed j and the 3C prediction of seed i from images Z to $Z{+}W$. Seeds that over-segment a common object tend to get merged due to a path of strong edges.

A fully convolutional neural network then predicts the correct label mapping between interrelated images of sections Z and $Z{\pm}W$, which determines which pixels in target image $Z{\pm}W$ belong to which seeds in source image Z (step

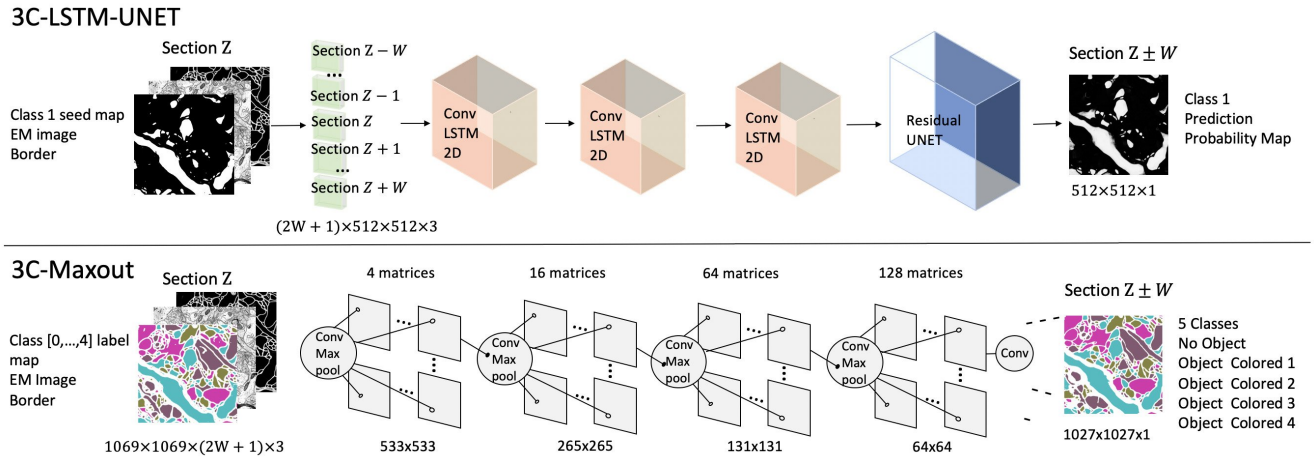


Figure 6. A schematic view of the 3C networks. The input layers have 3 channels of the raw image, seed mask and border probability, for $2W+1$ consecutive sections (images). The output is a feature map of seed predictions in section $Z \pm W$ (binary or labeled). Top: 3C-LSTM-UNET. Network architecture was implemented for the SNEMI3D dataset to optimize for accuracy. The inputs are processed with three consecutive Conv-LSTM modules, followed by a symmetric Residual U-Net structure. Bottom: 3C-Maxout. Network architecture was implemented for the Harvard Rodent Cortex and PNS datasets to optimize for speed.

3). All seeds here are represented by a fixed number l of colors, and prediction is done $\log(N)$ times based on Equation 1. For decoding, all $\log(N)$ predictions are aggregated for each pixel to determine the original label of the seed using χ^{-1} (Step 4). For training, we use saturated ground truth of the 3-D consistent objects. This approach allows us to formalize the reconstruction problem as a tracking problem between independent images, and to deal with the tendency of objects to disappear/appear in different portions of an enormous 3-D dataset.

We now describe how the 3C seed transfers are utilized to agglomerate the 2-D masks (as shown in Figure 5). For agglomeration (Step 5), the FCN for 3C is applied from all source images to their target images, which are at most W image sections apart from each other across the image stack (along the z dimension). We collect overlaps between all co-occurring segments, namely, those occurring by the original 2-D seeding, and those by the 3C seed transfer from source to target images. This leaves $2W+1$ instance segmentation cases for each image (including the initial seeding), which directly link seeds of different sections. Formally, the overlaps of different labels define a graph whose nodes are the 2-D seed mask labels and the directed weighted edges are their overlap ratio from the source to the target. Instead of optimizing this structure (as in the *Fusion* approach of [25]), we found that agglomerating all masks of sufficient overlap delivers adequate accuracy even for a small W. We do however make forced linking on lower probability edges to avoid "orphan" objects that are too small, which is biologically implausible. We provide further details in the Supplementary Materials.

We note that 3C does not attempt to correct potential merge errors in the initial seeding. These can be addressed post-hoc by learning morphological features [49, 60] or global constraints [34].

Adaptation to other domains: To leverage 3C for multi-object tracking in videos, a domain-specific seeder should precede cross classification (e.g. with deep coloring [28]). Natural images are likely to introduce spatially consistent object splits across frames and hence a dedicated agglomerating procedure should follow. The 3C technique can be readily applied to other medical imaging tasks, with seed transfers across different axes for isotropic settings.

3. Experiments

The SNEMI3D challenge is a widely used benchmark for connectomic segmentation algorithms dealing with anisotropic EM image stacks [5]. Although the competition ended in 2014, several leading labs recently submitted new results on this dataset, improving the state-of-the-art. Recently Plaza et al. suggested that benchmarking connectomics accuracy on small datasets as SNEMI3D is misleading as large-scale "catastrophic errors" are hard to assess [44, 45]. Moreover, the clustering metrics such as Variation of Information [36] and Rand Error [53] are inappropriate since they are not centered around the connectomics goal of unraveling neuron shape and inter-neuron connectivity. We therefore conduct experiments on three additional datasets and show the Rand-Error results only on the canonical SNEMI3D dataset. To assess the quality of 3C at large scale, we demonstrate results on the widely studied dataset by Kasthuri et al. [24] (*S1 Dataset*). To further assess our results in terms of the end-goal of connectomics, neuronal connectivity, we evaluate the synaptic connectivity of the

3C objects using the NRI metric [46] (*ECS Dataset*). In the final experiment we focus on the tracking ability of 3C (*PNS Dataset*).

3.1. SNEMI3D Dataset

In order to implement 3C on the SNEMI3D dataset, we first created an initial set of 2-D labeled seeds over the entire volume. These were generated based on the regional 2-D minima of the border probability map. This map was generated by a Residual U-Net, which is known for its excellent average pixel accuracy in border detection [31, 50]. Next, the 3C algorithm was used to transfer 2-D labeled seeds through the volume, as shown in Figure 4. Finally, the original 2-D labeled seeds and transferred labeled seeds were agglomerated if their overlap ratio exceeded 0.1. We found that $W{=}2$ delivers adequate accuracy. All orphans were greedily agglomerated to their best match. In order to achieve better accuracy, we tested 3C with various network architectures, and evaluated their accuracy. To date, convolutional LSTMs (ConvLSTM) have shown good performance for sequential image data [56]. In order to adapt these methods to the high pixel-accuracy required for connectomics, we combined both ConvLSTM and U-Net. The network is trained to learn an instance segmentation of one image based on the proposed instance segmentation of a nearby image with similar context. We found that the LSTM-UNET architecture has validation accuracy of 0.961, which outperforms other commonly used architectures. A schematic view of our architecture is given in Figure 6. Details are provided in the Supplementary Materials.

In order to illustrate the accuracy of 3C, we submitted our result to the public SNEMI3D challenge website alongside two common baseline models, the 3-D watershed transform (a region-growing technique) and Neuroproof agglomeration [41]. Our watershed code was adopted from [35]. Similar to other traditional agglomerating-techniques, Neuroproof trains a random forest on merge decisions of neighboring objects [40, 41, 42]. These baseline methods were fed with the same high-quality border maps used in our 3C reconstruction system. The comparisons of 2-D results with ground truth (section $Z{=}30$) are shown in Figure 7. Our result has fewer merge- and split-errors, and outperforms the two baselines by a large margin. Furthermore, 3C compares favorably to other state of art works recently published in Nature Methods [6, 21]. In the SNEMI3D challenge leaderboard the Rand-Error of 3C was 0.041, compared with the 0.06 achieved by a human annotator. Our accuracy (ranked 3rd) outperforms most of the traditional pipelines many by a large margin, and is slightly behind the slower neuron-by-neuron FFN segmentation for this volume. The leading entry is a UNET-based model learning short and long range affinities [31]. The results are summarized in Table 1.

Model	Rand	VI	VI_{split}	VI_{merge}	Complexity
Watershed	0.113	0.67	0.55	0.12	-
Neuroproof	0.104	0.55	0.42	0.13	-
Multicuts	0.068	0.41	0.34	0.07	-
3C	**0.041**	**0.31**	**0.19**	**0.12**	**O**$(V \log N)$
FFN	0.029	-	-	-	O(VN)
Human val.	0.060	-	-	-	-

Table 1. Comparison of Watershed, Neuroproof [41], Multicut [6], human values, 3C and FFN [21] on the SNEMI3D dataset for Rand-Error, Variation of Information *VI*, VI split, VI merge. Time Complexity: N is the number of objects and V is number of pixels. For empirical comparison see the performance section. We do not have access to the FFN and human outputs and hence their VI metric is missing.

3.2. Harvard Rodent Cortex Datasets (*ECS*, *S1*)

We describe two additional tests: (1) 3C on datasets with known synaptic connectivity (subsets of *ECS* and *S1*), and (2) a lightweight agglomeration-free reconstruction applied to a large-scale dataset (*S1*).

Connectivity-based test: Following [45], which recently advocated connectivity-based evaluation of connectomics, the accuracy of the pipeline was evaluated using the NRI metric [46]. In a nutshell, the NRI ranges between 0 and 1, measuring how well a given neuronal segmentation preserves the object connectivity between neural synapses (1 being optimal).

For the first test, we used a lightweight yet successful FCN model [35] (*Maxout*) (for border and 3C computations), reconstructing the test set of [24] (*S1*) (3C with FOV of 109 pixels). Maxout is currently the fastest border detector in connectomics, which was previously successfully used for single-object tracking [37]. Details of architecture and training are presented in the Supplementary Materials.

The NRI score of the 3C-Maxout segmentation was 0.54, compared to 0.41 of a traditional agglomeration pipeline [35]. For the second test, we were granted permission to reconstruct a recently collected rat cortex dataset of the Lichtman group at Harvard (*ECS*). This test allowed the comparison of 3C to the excellent agglomerative approach of [42] (4th on SNEMI3D), while using exactly their U-Net [50] border predictions as inputs to our 3C network. On the test set our NRI score was 0.86, compared to 0.73 for the agglomeration pipeline.

Large-scale reconstruction (*S1*): We also ran a fast version of 3C on the entire S1 dataset (90 gigavoxels: 1840 slices, 6 nm x 6 nm x 30 nm per voxel). In this experiment, we omitted the agglomeration step of the reconstruction algorithm to achieve better scalability and let 3C run on 3-D masks computed by local minima of the border probability maps. This implementation is highly scalable since it has no agglomeration step, while the 3C masks are updated on-

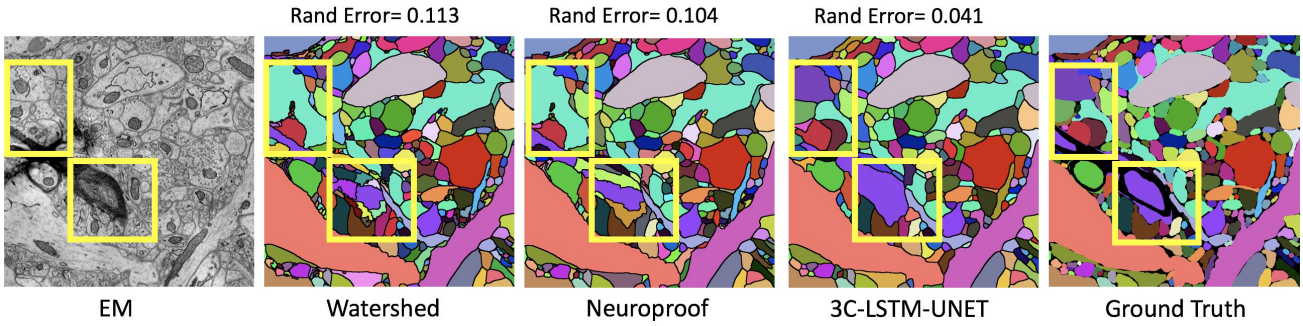


Figure 7. SNEMI3D: The 3C-LSTM-UNET Results compared with baseline techniques: Watershed, Neuroproof, and ground truth.

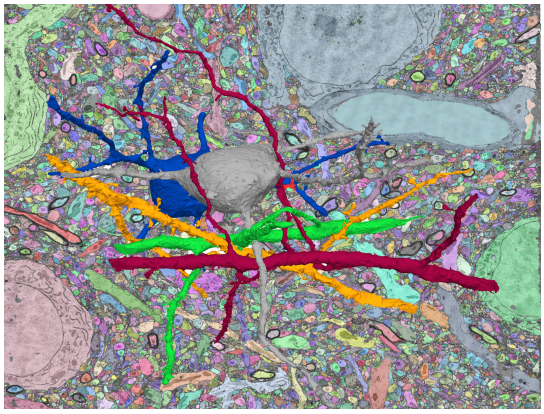


Figure 8. Results on Kasthuri et al. [24] *S1*. Fast lightweight 3C-Maxout operating on 3-D seeds, without agglomeration. Background: Segmented section. Foreground: five 3-D reconstructed objects.

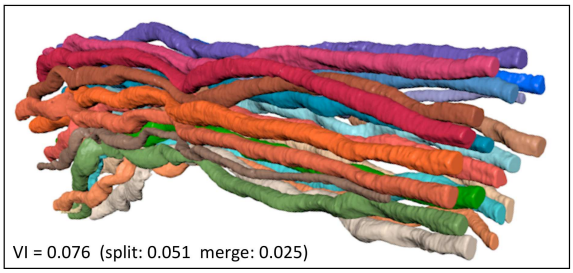


Figure 9. 3C-Maxout results of recursively tracking all objects (axons) directly from the PNS raw images (no post-processing).

the-fly in a streaming fashion every 100 slices. The Maxout implementation is attractive for large-scale systems because it is efficiently parallelized on multi-core systems with excellent cache behavior on CPUs [35]. Figure 8 shows five objects that span the whole volume of S1.

3.3. Peripheral Nervous System (*PNS*) Dataset

Next, we tested the ability of the 3C framework to track objects recursively based on raw images in a streaming mode, that is, independently of any agglomerating or post-processing steps.

We chose a previously unpublished motor nerve bundle from a (newborn) mouse contributed by the Lichtman lab at Harvard for this purpose because it is a closed system in which all objects are visible in the first and last image sections of the 915-images dataset. This dataset is important to neurobiologists since it contains the entire neural input (21 axons) of a complete muscle.

Again, we applied the 3C algorithm using the lightweight FCN Maxout architecture of [35]. 3C was able to track all objects without erroneous merges; results are shown in Figure 9. Out of the 21 axons, 20 were recursively reconstructed to their full extent (split errors in only one object). One extremely thin axon disappeared from the image and reappeared after 7 sections and was not reconstructed. The axon run-length for all reconstructed axons was above 70 microns (and $>$ 900 sections) until all of these exited the volume on the last slice in the image stack.

This benchmark demonstrates: **a)** 3C-Maxout performs well in a tracking task directly from raw images, even in the difficult connectomics regime, and **b)** our training procedures display satisfactory generalization abilities, learning from a relatively small number of examples.

4. Scalability Comparisons

In this section, we compare the relative scalability of 3C to FFN and MaskExtend, as far as possible without having access to the full FFN pipeline. 3C is a generalization of the FFN and MaskExtend techniques [21, 37], which augment the (pixel) support set of each object, one at a time. The 3C technique simultaneously augments all the objects from its input image(s) after a logarithmic number of iterations (see Figure 4). This allows us directly to compare the two types of approaches based on the number of iterations required, ignoring details of implementation.

FFN: We compare the number of FCNs calls in FFN and 3C assuming both algorithms reconstruct all objects flawlessly. We assume both algorithms use the same FCN model. Although 3C and FFN invoke FCNs a logarithmic versus a linear number of times, respectively, FFN runs on

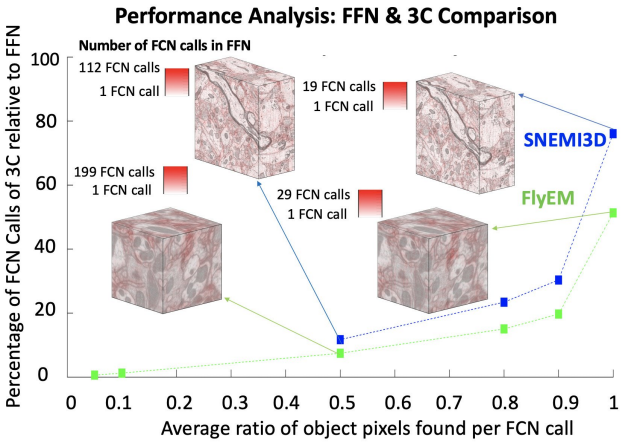
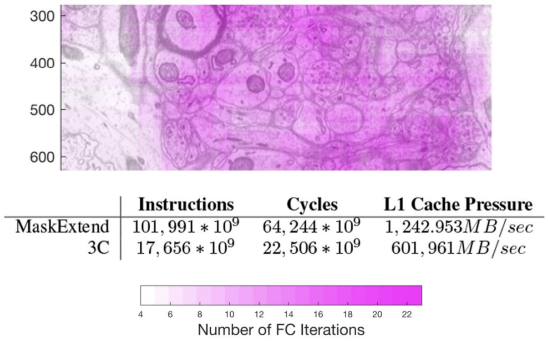


Figure 10. Compute cost per pixel using FFN-Style segmentation. We computed the number of times a pixel is participating in object detection (red) for two public datasets (SNEMI3D, FlyEM), and compared to the number of classification calls in 3C.



	Instructions	Cycles	L1 Cache Pressure
MaskExtend	$101,991 * 10^9$	$64,244 * 10^9$	$1,242.953 MB/sec$
3C	$17,656 * 10^9$	$22,506 * 10^9$	$601,961 MB/sec$

Figure 11. Compute cost per pixel with single-object tracking methods [37]. The number of calls per pixel is color-coded in purple. For the highly dense areas 23 calls of the object detector are required. The table depicts the performance counter statistics for the execution of [37] and the 3C-Maxout FCN on a stack of 100 images.

smaller inputs, centered around small regions of interest.

At each iteration, FFN will output an entire 3-D mask of the object around the center pixel. We assume that a fraction of those pixels will require revisiting (zero for best-case scenario). Figure 10 depicts the number of FCN calls and their ratio for FlyEM [51] and SNEMI3D [5] for FFN and 3C. In 3C, each pixel participates in an FCN call a number of times logarithmic in the number of objects visible in the field of view (the FCN calls for FFN are color-coded red in the data cubes of Figure 10). The y-axis depicts the ratio between the FCN calls by 3C to that for FFN, for several ratios of object pixels found per FCN call. A zero ratio means that no pixels are found for the object in a single FCN call, whereas 1 means that all object pixels are found and require no further revisiting. We can see from the plot that, assuming error-free reconstruction, 3C is more efficient than FFN when there is a fraction of object pixels that require revisiting after a single call of the FCN. The revisiting of some pixels is also reported by [21], as the 3-D output has greater uncertainty far from the initial pixels. For a revisit ratio of 0.5, 3C is more than 10x faster than FFN on FlyEM.

MaskExtend: Figure 11 repeats the above procedure with MaskExtend [37], comparing its FCN calls with 3C. MaskExtend is more wasteful than 3C, propagating some pixels into its FCN model 23 times. The instruction and cycle counts as well as the L1 Cache pressure are larger for MaskExtend (equal multi-core infrastructure and inference framework [35]).

5. Conclusion

In this paper, we have presented cross-classification clustering (3C), an algorithm that tracks multiple objects simultaneously, transferring a segmentation from one image to the next by composing simpler segmentations. We have demonstrated the power of 3C in the domain of connectomics, which presents an especially difficult task for image segmentation. Within the space of connectomics algorithms, 3C provides an end-to-end approach with fewer "moving parts," improving on the accuracy of many leading connectomics systems. Our solution is computationally cheap, can be achieved with lightweight FCNs, and is at least an order of magnitude faster than its relative, flood-filling networks. Although the main theme of this paper was tackling neuronal reconstruction, our approach also promises scalable, effective algorithms for broader applications in medical imaging and video tracking.

Acknowledgements

We would like to thank Jeff Lichtman and Kai Kang for allowing us to access the PNS dataset, Marco Badwal for alignment, and Daniel Berger and Casimir Wierzynski for insightful comments. This research was supported by the National Science Foundation (NSF) under grants IIS-1607189, IIS-1447786, CCF-1563880, IIS-1803547 and by a grant from the Intel corporation.

References

[1] Amit Adam, Ehud Rivlin, and Ilan Shimshoni. Robust fragments-based tracking using the integral histogram. In *Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 798–805. IEEE, 2006.

[2] Björn Andres, Ullrich Köthe, Moritz Helmstaedter, Winfried Denk, and Fred A Hamprecht. Segmentation of SBFSEM volume data of neural tissue by hierarchical classification. In *Joint Pattern Recognition Symposium*, pages 142–152. Springer, 2008.

[3] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(5):898–916, 2011.

[4] Assaf Arbelle, Jose Reyes, Jia-Yun Chen, Galit Lahav, and Tammy Riklin Raviv. A probabilistic approach to joint cell tracking and segmentation in high-throughput microscopy videos. *Medical image analysis*, 47:140–152, 2018.

[5] Ignacio Arganda-Carreras, Srinivas C Turaga, Daniel R Berger, Dan Cireşan, Alessandro Giusti, Luca M Gambardella, Jürgen Schmidhuber, Dmitry Laptev, Sarvesh Dwivedi, Joachim M Buhmann, et al. Crowdsourcing the creation of image segmentation algorithms for connectomics. *Frontiers in neuroanatomy*, 9:142, 2015.

[6] Thorsten Beier, Constantin Pape, Nasim Rahaman, Timo Prange, Stuart Berg, Davi D Bock, Albert Cardona, Graham W Knott, Stephen M Plaza, Louis K Scheffer, et al. Multicut brings automated neurite segmentation closer to human performance. *Nature Methods*, 14(2):101–102, 2017.

[7] Manuel Berning, Kevin M Boergens, and Moritz Helmstaedter. SegEM: efficient image analysis for high-resolution connectomics. *Neuron*, 87(6):1193–1206, 2015.

[8] Ryoma Bise, Takeo Kanade, Zhaozheng Yin, and Seung-il Huh. Automatic cell tracking applied to analysis of cell migration in wound healing assay. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6174–6179. IEEE, 2011.

[9] Dan Ciresan, Alessandro Giusti, Luca M Gambardella, and Jürgen Schmidhuber. Deep neural networks segment neuronal membranes in electron microscopy images. In *Advances in neural information processing systems*, pages 2843–2851, 2012.

[10] Qi Dou, Lequan Yu, Hao Chen, Yueming Jin, Xin Yang, Jing Qin, and Pheng-Ann Heng. 3d deeply supervised network for automated segmentation of volumetric medical images. *Medical image analysis*, 41:40–54, 2017.

[11] Michal Drozdzal, Gabriel Chartrand, Eugene Vorontsov, Mahsa Shakeri, Lisa Di Jorio, An Tang, Adriana Romero, Yoshua Bengio, Chris Pal, and Samuel Kadoury. Learning normalized inputs for iterative estimation in medical image segmentation. *Medical image analysis*, 44:1–13, 2018.

[12] Loïc Fagot-Bouquet, Romaric Audigier, Yoann Dhome, and Frédéric Lerasle. Improving multi-frame data association with sparse representations for robust near-online multi-object tracking. In *European Conference on Computer Vision*, pages 774–790. Springer, 2016.

[13] Jialue Fan, Wei Xu, Ying Wu, and Yihong Gong. Human tracking using convolutional neural networks. *IEEE Transactions on Neural Networks*, 21(10):1610–1623, 2010.

[14] Rohit Girdhar, Georgia Gkioxari, Lorenzo Torresani, Manohar Paluri, and Du Tran. Detect-and-track: Efficient pose estimation in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 350–359, 2018.

[15] Kostas Haris, Serafim N Efstratiadis, Nikolaos Maglaveras, and Aggelos K Katsaggelos. Hybrid image segmentation using watersheds and fast region merging. *IEEE Transactions on image processing*, 7(12):1684–1699, 1998.

[16] David Held, Sebastian Thrun, and Silvio Savarese. Learning to track at 100 fps with deep regression networks. In *European Conference on Computer Vision*, pages 749–765. Springer, 2016.

[17] Moritz Helmstaedter. Cellular-resolution connectomics: challenges of dense neural circuit reconstruction. *Nature methods*, 10(6):501–507, 2013.

[18] Moritz Helmstaedter, Kevin L Briggman, Srinivas C Turaga, Viren Jain, H Sebastian Seung, and Winfried Denk. Connectomic reconstruction of the inner plexiform layer in the mouse retina. *Nature*, 500(7461):168, 2013.

[19] Nathaniel Huebsch, Peter Loskill, Mohammad A Mandegar, Natalie C Marks, Alice S Sheehan, Zhen Ma, Anurag Mathur, Trieu N Nguyen, Jennie C Yoo, Luke M Judge, et al. Automated video-based analysis of contractility and calcium flux in human-induced pluripotent stem cell-derived cardiomyocytes cultured over different spatial scales. *Tissue Engineering Part C: Methods*, 21(5):467–479, 2015.

[20] Viren Jain, Joseph F Murray, Fabian Roth, Srinivas Turaga, Valentin Zhigulin, Kevin L Briggman, Moritz N Helmstaedter, Winfried Denk, and H Sebastian Seung. Supervised learning of image restoration with convolutional networks. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[21] Michał Januszewski, Jörgen Kornfeld, Peter H Li, Art Pope, Tim Blakely, Larry Lindsey, Jeremy Maitin-Shepard, Mike Tyka, Winfried Denk, and Viren Jain. High-precision automated reconstruction of neurons with flood-filling networks. *Nature methods*, 15(8):605, 2018.

[22] Alexandr A Kalinin, Ari Allyn-Feuer, Alex Ade, Gordon-Victor Fon, Walter Meixner, David Dilworth, Jeffrey R De Wet, Gerald A Higgins, Gen Zheng, Amy Creekmore, et al. 3d cell nuclear morphology: microscopy imaging dataset and voxel-based morphometry classification results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 2272–2280, 2018.

[23] Kai Kang, Wanli Ouyang, Hongsheng Li, and Xiaogang Wang. Object detection from video tubelets with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 817–825, 2016.

[24] Narayanan Kasthuri, Kenneth Jeffrey Hayworth, Daniel Raimund Berger, Richard Lee Schalek, José Angel Conchello, Seymour Knowles-Barley, Dongil Lee, Amelio Vázquez-Reina, Verena Kaynig, Thouis Raymond Jones, et al. Saturated reconstruction of a volume of neocortex. *Cell*, 162(3):648–661, 2015.

[25] Verena Kaynig, Amelio Vazquez-Reina, Seymour Knowles-Barley, Mike Roberts, Thouis R Jones, Narayanan Kasthuri, Eric Miller, Jeff Lichtman, and Hanspeter Pfister. Large-scale automatic reconstruction of neuronal processes from electron microscopy images. *Medical image analysis*, 22(1):77–88, 2015.

[26] Jinseop S Kim, Matthew J Greene, Aleksandar Zlateski, Kisuk Lee, Mark Richardson, Srinivas C Turaga, Michael Purcaro, Matthew Balkam, Amy Robinson, Bardia F Be-

habadi, et al. Space–time wiring specificity supports direction selectivity in the retina. *Nature*, 509(7500):331, 2014.

[27] Seymour Knowles-Barley, Verena Kaynig, Thouis Ray Jones, Alyssa Wilson, Joshua Morgan, Dongil Lee, Daniel Berger, Narayanan Kasthuri, Jeff W Lichtman, and Hanspeter Pfister. RhoanaNet pipeline: Dense automatic neural annotation. *arXiv preprint arXiv:1611.06973*, 2016.

[28] Victor Kulikov, Victor Yurchenko, and Victor Lempitsky. Instance segmentation by deep coloring. *arXiv preprint arXiv:1807.10007*, 2018.

[29] Avisek Lahiri, Kumar Ayush, Prabir Kumar Biswas, and Pabitra Mitra. Generative adversarial learning for reducing manual annotation in semantic segmentation on large scale miscroscopy images: Automated vessel segmentation in retinal fundus image as test case. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 42–48, 2017.

[30] June-Goo Lee, Sanghoon Jun, Young-Won Cho, Hyunna Lee, Guk Bae Kim, Joon Beom Seo, and Namkug Kim. Deep learning in medical imaging: general overview. *Korean journal of radiology*, 18(4):570–584, 2017.

[31] Kisuk Lee, Jonathan Zung, Peter Li, Viren Jain, and H Sebastian Seung. Superhuman accuracy on the SNEMI3D connectomics challenge. *arXiv preprint arXiv:1706.00120*, 2017.

[32] Jeff W Lichtman and Winfried Denk. The big and the small: challenges of imaging the brains circuits. *Science*, 334(6056):618–623, 2011.

[33] Jeff W Lichtman, Hanspeter Pfister, and Nir Shavit. The big data challenges of connectomics. *Nature neuroscience*, 17(11):1448–1454, 2014.

[34] Brian Matejek, Daniel Haehn, Haidong Zhu, Donglai Wei, Toufiq Parag, and Hanspeter Pfister. Biologically-constrained graphs for global connectomics reconstruction. *CVPR*, 2019.

[35] Alexander Matveev, Yaron Meirovitch, Hayk Saribekyan, Wiktor Jakubiuk, Tim Kaler, Gergely Odor, David Budden, Aleksandar Zlateski, and Nir Shavit. A multicore path to connectomics-on-demand. In *Proceedings of the 22Nd ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming*, PPoPP '17, pages 267–281, New York, NY, USA, 2017. ACM.

[36] Marina Meilă. Comparing clusterings. *Journal of multivariate analysis*, 98(5):873–895, 2007.

[37] Yaron Meirovitch, Alexander Matveev, Hayk Saribekyan, David Budden, David Rolnick, Gergely Odor, Seymour Knowles-Barley Thouis Raymond Jones, Hanspeter Pfister, Jeff William Lichtman, and Nir Shavit. A multi-pass approach to large-scale connectomics. *arXiv preprint arXiv:1612.02120*, 2016.

[38] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571. IEEE, 2016.

[39] Laurent Najman and Michel Schmitt. Geodesic saliency of watershed contours and hierarchical segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 18(12):1163–1173, 1996.

[40] Juan Nunez-Iglesias, Ryan Kennedy, Toufiq Parag, Jianbo Shi, and Dmitri B Chklovskii. Machine learning of hierarchical clustering to segment 2D and 3D images. *PloS one*, 8(8):e71715, 2013.

[41] Toufiq Parag, Anirban Chakraborty, Stephen Plaza, and Louis Scheffer. A context-aware delayed agglomeration framework for electron microscopy segmentation. *PloS one*, 10(5):e0125825, 2015.

[42] Toufiq Parag, Fabian Tschopp, William Grisaitis, Srinivas C Turaga, Xuewen Zhang, Brian Matejek, Lee Kamentsky, Jeff W Lichtman, and Hanspeter Pfister. Anisotropic EM segmentation by 3D affinity learning and agglomeration. *arXiv preprint arXiv:1707.08935*, 2017.

[43] AG Amitha Perera, Chukka Srinivas, Anthony Hoogs, Glen Brooksby, and Wensheng Hu. Multi-object tracking through simultaneous long occlusions and split-merge conditions. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 1, pages 666–673. IEEE, 2006.

[44] Stephen M Plaza and Stuart E Berg. Large-scale electron microscopy image segmentation in Spark. *arXiv preprint arXiv:1604.00385*, 2016.

[45] Stephen M Plaza and Jan Funke. Analyzing image segmentation for connectomics. *Frontiers in Neural Circuits*, 12:102, 2018.

[46] Elizabeth P Reilly, Jeffrey S Garretson, William R Gray Roncal, Dean M Kleissas, Brock A Wester, Mark A Chevillet, and Matthew J Roos. Neural reconstruction integrity: A metric for assessing the connectivity accuracy of reconstructed neural networks. *Frontiers in Neuroinformatics*, 12:74, 2018.

[47] Xiaofeng Ren and Jitendra Malik. Learning a classification model for segmentation. In *null*, page 10. IEEE, 2003.

[48] Aurélien Rizk, Grégory Paul, Pietro Incardona, Milica Bugarski, Maysam Mansouri, Axel Niemann, Urs Ziegler, Philipp Berger, and Ivo F Sbalzarini. Segmentation and quantification of subcellular structures in fluorescence microscopy images using squassh. *Nature protocols*, 9(3):586, 2014.

[49] David Rolnick, Yaron Meirovitch, Toufiq Parag, Hanspeter Pfister, Viren Jain, Jeff W Lichtman, Edward S Boyden, and Nir Shavit. Morphological error detection in 3d segmentations. *arXiv preprint arXiv:1705.10882*, 2017.

[50] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015*, pages 234–241. Springer, 2015.

[51] Shin-ya Takemura, C Shan Xu, Zhiyuan Lu, Patricia K Rivlin, Toufiq Parag, Donald J Olbris, Stephen Plaza, Ting Zhao, William T Katz, Lowell Umayam, et al. Synaptic circuits and their variations within different columns in the visual system of drosophila. *Proceedings of the National Academy of Sciences*, 112(44):13711–13716, 2015.

[52] Srinivas C Turaga, Joseph F Murray, Viren Jain, Fabian Roth, Moritz Helmstaedter, Kevin Briggman, Winfried Denk, and H Sebastian Seung. Convolutional networks can learn to generate affinity graphs for image segmentation. *Neural computation*, 22(2):511–538, 2010.

[53] Ranjith Unnikrishnan, Caroline Pantofaru, and Martial Hebert. Toward objective evaluation of image segmentation algorithms. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, (6):929–944, 2007.

[54] Mengmeng Wang, Yong Liu, and Zeyi Huang. Large margin object tracking with circulant feature maps. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4021–4029, 2017.

[55] Yu Xiang, Alexandre Alahi, and Silvio Savarese. Learning to track: Online multi-object tracking by decision making. In *Proceedings of the IEEE international conference on computer vision*, pages 4705–4713, 2015.

[56] SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810, 2015.

[57] SYJCY Yoo, Kimin Yun, Jin Young Choi, K Yun, and JY Choi. Action-decision networks for visual tracking with deep reinforcement learning. CVPR, 2017.

[58] Li Zhang, Yuan Li, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[59] Tianzhu Zhang, Changsheng Xu, and Ming-Hsuan Yang. Multi-task correlation particle filter for robust object tracking. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4335–4343, 2017.

[60] Jonathan Zung, Ignacio Tartavull, Kisuk Lee, and H Sebastian Seung. An error detection and correction framework for connectomics. In *Advances in Neural Information Processing Systems*, pages 6818–6829, 2017.