Segmentation fusion for connectomics

Segmentation Fusion for Connectomics

Amelio Vázquez-Reina,[1,2,3] Michael Gelbart,[4] Daniel Huang,[1]
Jeff Lichtman,[3] Eric Miller,[2] Hanspeter Pfister[1]

[1]School of Engineering and Applied Sciences, Harvard University, MA (USA)
[2]Department of Computer Science, Tufts University, MA (USA)
[3]Center for Brain Science, Harvard University, MA (USA)
[4]Graduate Program in Biophysics, Harvard University, MA (USA)
✉amelio@seas.harvard.edu

Abstract

We address the problem of automatic 3D segmentation of a stack of electron microscopy sections of brain tissue. Unlike previous efforts, where the reconstruction is usually done on a section-to-section basis, or by the agglomerative clustering of 2D segments, we leverage information from the entire volume to obtain a globally optimal 3D segmentation. To do this, we formulate the segmentation as the solution to a fusion problem. We first enumerate multiple possible 2D segmentations for each section in the stack, and a set of 3D links that may connect segments across consecutive sections. We then identify the fusion of segments and links that provide the most globally consistent segmentation of the stack. We show that this two-step approach of pre-enumeration and posterior fusion yields significant advantages and provides state-of-the-art reconstruction results. Finally, as part of this method, we also introduce a robust rotationally-invariant set of features that we use to learn and enumerate the above 2D segmentations. Our features outperform previous connectomic-specific descriptors without relying on a large set of heuristics or manually designed filter banks.

1. Introduction

Connectomics is an emerging branch of neuroscience that aims to physically map and resolve the neural circuitry of the nervous system [11]. With the help of recent advances in the preparation, sectioning, and imaging of brain tissue, biologists can now study neural connectivity and physiology from the visual inspection of image stacks at scales of only a few nanometers [7]. It is believed that the automatic, high-throughput analysis of such images can lead to a better understanding of the mammalian brain, as well as provide new insight into other important areas of neuroscience, such as the study of the biological basis of learning or memory [7]. Besides its scientific applications, connectomics also provides great opportunities for the computer vision community, as is evident in the fast growing vision literature addressing problems such as neuron segmentation [8, 9, 20, 5], sparse tomographic reconstruction [19], and the design of features for connectomics [12].

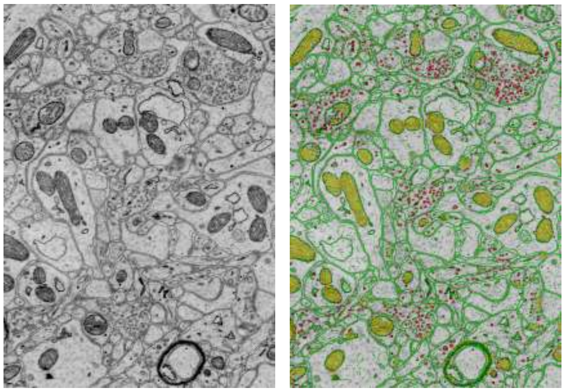


Figure 1: *Automatic labeling (right) of cellular structures of a 2D section of a volume of brain tissue (left) using our features. Each pixel is labeled by a random forest classifier trained with the features we introduce in Section 4. Mitochondria are shown in yellow, vesicles in red and cellular boundaries in green. The rest of pixels are labeled as cellular background (shown with a transparent label).*

In this paper, we focus on the problem of 3D segmentation of spatially irregular volumes of brain tissue. Such volumes are often obtained in neurobiology from serial-section electron microscopy (ssEM) of brain tissue, and their segmentation poses important challenges from a computer vision perspective. The 2D physical resolution of each stack section is usually one order of magnitude finer than the average distance between sections. This anisotropic sampling results in irregular spatial discontinuities when moving between sections, even after image registration, thereby significantly complicating the problem of tracking continuous structures across sections [11]. Neurons can also branch, merge, originate or terminate anywhere within the volume, and some neural processes such as spines can be as thin as the distance between sections or even move parallel between them. Finally, inconsistencies in the staining and cutting of the tissue, along with imaging artifacts, can make it challenging for even a neurobiologist to distinguish neural organelles or membranes without comparing images from several sections.

The problem of automatic segmentation of ssEM datasets has been addressed recently by several authors. These efforts have been devoted to a variety of problems, such as the labeling of cellular cross-sections [9, 5], the detection of specific cellular structures [12] such as mitochondria [13] or cell membranes [8, 9, 5], or the co-segmentation of pairs of adjacent sections of brain tissue [20].

A growing number of efforts have been dedicated recently to the problem of 3D labeling individual neurons from full stacks [8, 9, 20, 14]. Some of these solutions are based on Markov Random Fields (MRFs) at the pixel level. Given the anisotropic nature of ssEM volumes, they tend to be quite sensitive to the alignment between sections and cannot fully exploit the redundancy between the images in the stack [8].

A common approach in addressing this problem is to first obtain a 2D segmentation of each section, and then to match or agglomeratively cluster these 2D segments across sections in the stack [9, 20, 14]. However, such methods rely on the assumption that the initial 2D segmentation of each section is good enough for the posterior grouping, or that every neuron has been oversegmented [20, 9]. In addition, some of these methods need indirect penalties to prevent trivial yet incorrect clusterings [20], or require setting stopping conditions or manually-designed rules to converge to the right solution [9, 14]. Finally, a number of methods rely on greedy segment-merging strategies [9, 14] and cannot guarantee global clustering optimality.

It is also worth noting that there have been a number of efforts addressing semi-automatic 3D segmentation and tracing of individual neurons [18, 6]. However, in this paper, we restrict ourselves to fully automatic solutions.

Contributions. We introduce the notion of segmentation fusion, the global fusion of 2D segments and 3D links for the problem of 3D neuron segmentation. Our method compares multiple possible 2D segmentations and linking choices across sections to find the best fusion of segments and links that together form each neuron. Our fusion framework is flexible and does not require full over-segmentation of each section *a priori*, or the use of indirect penalties or stopping conditions to handle both small and large structures.

We also present a novel set of features for the classification of cellular structures that provide an accurate, rotationally-invariant summary of connectomic patches. We use this classifier to obtain 2D segmentations for the fusion. An example highlighting the discriminative power of our features is shown in Fig. 1. In contrast to previous work, our features do not require hand-tuning of a large set of heuristics or filter banks and efficiently identify cellular structures of different scales and morphologies.

2. Segmentation Fusion for Neural Reconstruction

We provide a high-level picture of our fusion framework in Fig. 2. We model each neuron as a group of 2D segments (neuron cross-sections) connected by a sequence of 3D links. We then formulate the problem of identifying the neurons in the stack as the problem of finding the fusion of segments and links that form each neuron.

We start by training a pixel classifier to label individual cellular structures including mitochondria, neurotransmitters, and cellular boundaries in each 2D section. The classifier takes a patch centered around each pixel, computes a compact feature descriptor, and outputs the probability of the center pixel belonging to each class (an example of the labeling is shown in Fig. 1). We provide details about the feature descriptor we use for classification in Section 4.

Once the pixel classifier has assigned a probability per class to each pixel, we enumerate a set of possible 2D cross-sections of neurons in each section. To do this, we apply multiple watershed transforms on the probability map for the boundary class obtained on each section (one watershed for each height in the map). Each watershed outputs a 2D partitioning of the section into several possible cell cross-sections. Together, all the watersheds provide a large set of 2D segments that may be used to identify cross-sections of neurons in each section.

We then enumerate a set of 3D links that connect pairs of the 2D segments obtained from the watersheds across consecutive sections. We enumerate those links that connect pairs that spatially overlap in XY (the image plane), but that belong to two different, consecutive sections in Z (see Fig. 3).

Once we have a set of candidate segments and links between them, we formulate our fusion problem. As mentioned earlier, we consider each neuron to consist of a sequence of segments (cross-sections) and links, and our goal is to find the fusion that forms each neuron. We formulate the solution to the fusion problem as the maximum a posteriori (MAP) labeling of an MRF, subject to a set of predefined clustering constraints. The MRF provides a convenient factorization of the variables in the problem, while the

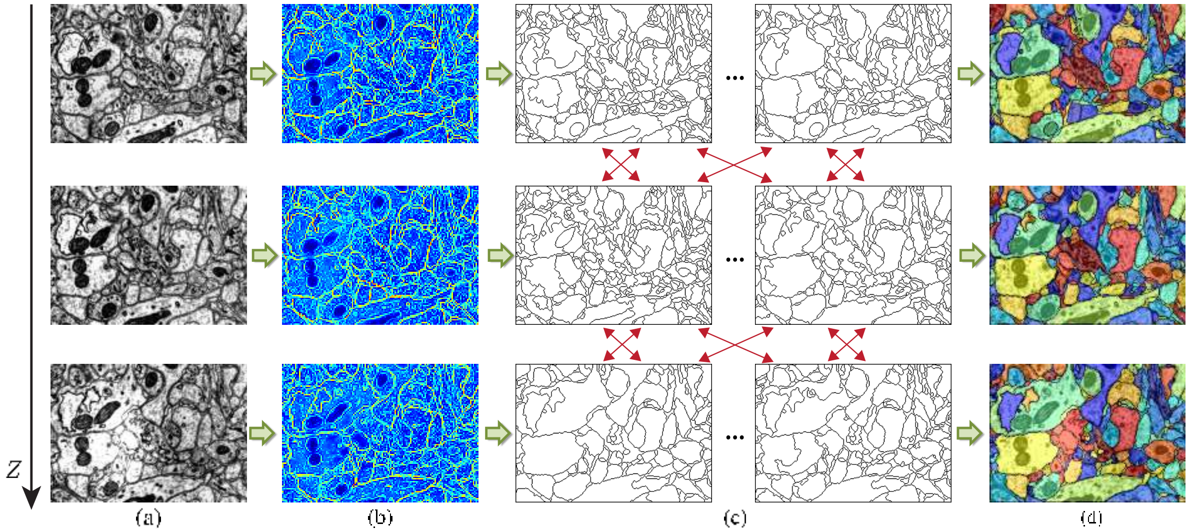


Figure 2: *Each section from an input stack of electron microscopy images is represented by each row in the figure, from the original image (column* ***(a)****), we compute the probability of each pixel belonging to the cell boundary (column* ***(b)****). We obtain this probability from the output of a random forest classifier applied to our rotationally-invariant features (described in Section 4) that was previously trained on a set of manually annotated images. We then apply a sequence of watershed transformations at different heights to the probability maps in each section. The outputs of the watershed transformations provide a set of possible 2D segmentations for each section (column* ***(c)****). We then enumerate a set of possible 3D links that may be used to connect segments in 3D (represented by the arrows in red). Finally, we determine the segmentation fusion of segments and links that identifies each neuron in the stack (column* ***(d)****).*

constraints help us to add prior knowledge about the relationship between segments and links. As we explain later, we use constraints to prevent, among other things, the selection of two segments for the final fusion that overlap in the same section (because we know that one pixel can only belong to one cell), and to make sure that, if a 3D link connecting two 2D segments from two consecutive sections is selected, the 2D segments are also selected. We formalize this in the next section.

3. Modeling Fusion with MAP-MRF

We start by associating indicator variables (binary variables) s_i and l_j with each 2D segment and 3D link enumerated before. A segment is assumed selected for the final segmentation if its indicator variable is activated (*e.g.* $s_i = 1$), and similarly for a 3D link (*e.g.*, $l_j = 1$). This way, a 3D segmentation of the data is simply specified by a labeling of the indicator variables.

With these definition, we model the solution to the 3D segmentation problem as the MAP selection of segments and links given the image data subject to the fusion constraints. For simplicity, we formulate the posterior probability with the following factorization:

$$P\left(\mathbf{s}, \mathbf{l} | \text{data}\right) = \frac{1}{Z} \psi_{\text{SAT}}\left(\mathbf{s}, \mathbf{l}\right) \prod_{i,j=1}^{S,L} \psi_s(s_i) \psi_l(l_j), \quad (1)$$

where Z is the partition function [21], $\mathbf{s}$ and $\mathbf{l}$ represent the vectors of indicator variables, S and L are the number of 2D segments and 3D links in the enumeration, and $\psi_s\left(s_i\right) = \exp\left(\theta_{s_i} s_i\right)$ and $\psi_l\left(l_j\right) = \exp\left(\theta_{l_j} l_j\right)$, and ψ_{SAT} are compatibility functions of an MRF defined over the indicator variables [21].

The parameters θ_{s_i} and θ_{l_j} measure the relative strength of a 2D segment s_i and a 3D link l_j in the posterior probability, respectively. Dropping the indices i and j for the moment, in our experiments we set each θ_s from the output of our 2D pixel classifier by measuring the strength of the membrane probability on the boundary. As for θ_l, we measure the cross-correlation and displacement between the pair of segments that are connected by the link in question. The more similar the segments are, and the closer they are to each other, the more likely they are presumed to be connected. However, other choices for θ_s and θ_l are possible.

Both θ_s and θ_l should take into account the relative size of the segments being considered. Specifically they should guarantee that, any large segment, aside from the image data, can compete equally against a set of much smaller

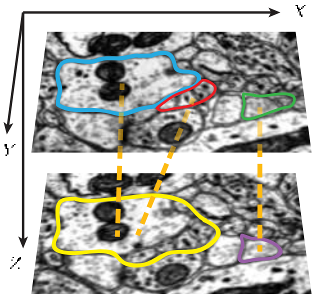


Figure 3: *We enumerate links between every pair of segments that belong to two consecutive sections and that overlap in XY. Above, the yellow segment overlaps in XY with the blue and red segments, and not with the green segment.*

segments that may be eligible to cover the same 2D area. In our experiments, we choose to achieve this calibration by setting θ_s and θ_l as follows: $\theta_s = \overline{\theta_s}\,\mathrm{size}\,(s)$ and $\theta_l = \overline{\theta_l}\,\mathrm{size}\,(l)$, where $\mathrm{size}\,(s)$ measures the size of a segment (*e.g.* in pixels), and $\mathrm{size}\,(l) = \mathrm{size}\,(s_a) + \mathrm{size}\,(s_b)$ measures the size of a link connecting two segments s_a and s_b. The parameters $\overline{\theta_s}, \overline{\theta_l} \in [0,1]$ measure the normalized weight of a segment s and a link l, respectively, as described before.

As mentioned earlier, some fusions of segments and links are physically unrealistic or undesirable given our knowledge about the nature of the problem. We use a special compatibility function, ψ_{SAT}, to give such configurations zero-mass in the MRF (effectively assigning them zero probability of being chosen as a MAP solution), *i.e.*:

$$\psi_{\mathrm{SAT}}\left(\mathbf{s},\mathbf{l}\right) = \begin{cases} 1 : \text{if } \mathbf{s} \text{ and } \mathbf{l} \text{ satisfy fusion conditions} \\ 0 : \text{else} \end{cases} \tag{2}$$

We next enumerate our fusion conditions. The first one prevents undesired solutions, while the second one helps obtain an overall better 3D segmentation. We provide an illustration explaining them in Fig. 4.

1. Preventing overlaps between segments:
As indicated earlier, we model each neuron as a group of 2D segments connected by a sequence of 3D links, with each segment representing the cross-section of a neuron. We also know that each pixel can only belong to one neuron at a time. These two facts imply that, for our modeling to be consistent, we cannot tolerate segment overlaps.

One possible way of avoiding overlaps is to require the segmentation to provide a tessellation of the stack (a partitioning without gaps nor overlaps), effectively forcing every region to be covered by exactly one segment. We can achieve this by defining a constraint that requires that $\sum_{i \in \mathrm{o_k}} s_i = 1$ for every set of overlapping segments $\mathrm{o_k}$. However, in our experiments, we noticed that such constraint can be too restrictive. Depending on the quality of the data, it is sometimes the case that a small 2D region in the stack can only choose from a set of segments that may not correspond to anything meaningful in the data. This can cause neighboring regions that may otherwise be able to choose a good segment, be covered by the same wrong choice. To avoid this problem, we choose to relax the previous constraint by simply letting some regions not be covered by any segment, *i.e.*, $\sum_{i \in \mathrm{o_k}} s_i \leq 1$, which still prevents overlaps between segments.

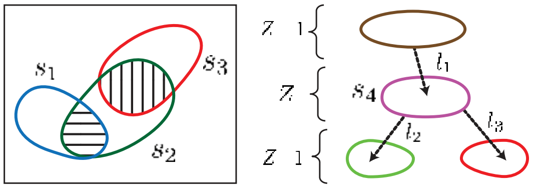


Figure 4: ***Left:*** *Three possible neural cross-sections overlapping on the same section. Each of them comes from a different 2D segmentation of the section. Our first fusion condition (see text) requires the MRF to avoid overlaps when deciding on a segmentation for the section. This is enforced by requiring that the indicator variables satisfy $s_1 + s_2 \leq 1$, and $s_2 + s_3 \leq 1$.* ***Right:*** *Our fusion framework requires the MRF to choose those segments that provide the best continuity in 3D (those that are connected by the best links). The selection of segments depends on the selection of links via our second fusion condition. In this case, s_4 must be activated if either l_1, l_2 or l_3 are activated (see Eq. 3).*

2. Rewarding good 3D continuity across the stack:
One way of leveraging information from other sections when choosing a segment for a section is to encourage the selection of segments that yield good 3D continuity in the stack. The idea is to make decisions in each section depend on decisions made in the rest of the stack. We achieve this by making the selection of segments and links dependent on each other, and rewarding the selection of segments that are connected by strong links. To do this, we set up the following constraints for every candidate segment s_i:

$$\sum_{j \in \mathrm{TOP_i}} l_j \leq s_i, \quad \sum_{j \in \mathrm{BOT_i}} l_j \leq s_i \tag{3}$$

where $\mathrm{TOP_i}$ specifies the set of links connecting the segment s_i from the top and $\mathrm{BOT_i}$ the ones connecting it from the bottom (in Fig 4, $\mathrm{BOT_4}$ for segment s_4 would include the links l_2 and l_4, and $\mathrm{TOP_4}$ the link l_1). To see how this condition works, notice that a link necessarily connects one segment from the top, and one segment from the bottom, so from Eq. 3, its activation requires the activation of the corresponding segments.

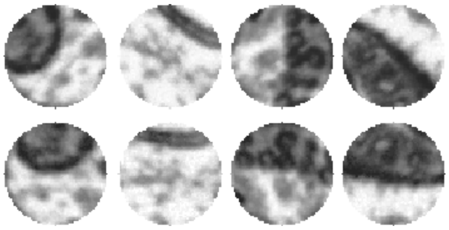


Figure 5: ***Top:*** *Exemplar circular patches of brain tissue.* ***Bottom:*** *Inverse discrete Zernike transfrom [15] applied to the rotation normalization of the patches provided by our features. Given a patch, the correct classification of its center pixel is invariant to orientation (but not necessarily to changes in scale or translation). Our features automatically normalize each patch to a referential orientation (in the examples, the disks are automatically rotated to move lower intensities towards the top of the disk). This rotation normalization can help reduce the sample complexity and the size of the training set for the classification of the center pixel of the patch.*

Solving the segmentation fusion. As stated earlier, we determine the solution to our fusion problem as the MAP assignment to our vectors of the indicator variables $\mathbf{s}$ and $\mathbf{l}$. Such an assignment is obtained by solving $\arg\max_{\mathbf{s},\mathbf{l}} P\left(\mathbf{s},\mathbf{l}|\text{data}\right)$, with the posterior probability given by Eq. 1, which yields the following binary linear programming problem:

$$\begin{aligned} \arg\max_{\mathbf{s},\mathbf{l}} \; & \textstyle\sum_{i=1}^{S} \theta_{s_i} s_i + \sum_{j=1}^{L} \theta_{l_j} l_j \\ \text{s.t.} \;\; & s_i, l_j \in \{0,1\}, \\ & \psi_{\text{SAT}}\left(\mathbf{s},\mathbf{l}\right) = 1 \end{aligned} \tag{4}$$

We solve Eq. 4 using a general-purpose binary linear programming solver. We discuss implementation details and running times in Section 5.

4. Adapted Zernike Features

In this section, we introduce an orthogonal set of rotationally-invariant features for labeling cellular structures in stacks of brain tissue based on the recently introduced Discrete Zernike Transform (DZT) [15]. We use these features to first label each pixel as belonging to either mitochondria, cellular boundary, neurotransmitter, or cellular space (inside and outside) based on a surrounding patch; and second, to enumerate the possible 2D segments (neural cross-sections) that are needed for the fusion.

Rotational invariance is a desired quality when labeling or classifying image patches in connectomics. The 2D orientation of a patch of brain tissue is presumed irrelevant for its classification, and should not condition the labeling or segmentation of the patch, or the labeling of its center pixel (see Fig. 5).

The design of rotationally invariant patch descriptors for applications such as texture classification or image categorization is well discussed in the literature [16, 1]. However, most of these descriptors also provide invariance to general affine transformations such as scale or translation, which is not necessarily desired in connectomics. For example, the correct label of the center pixel of a patch may depend on which cellular structure the center pixel lies on within the patch (*i.e.*, a translation of the patch).

Recently, several authors have addressed the design of features that target specific cellular structures such as mitochondria [13, 12], cell boundaries [4], or synapses [10]. Most of these descriptors require the adjustment of a set of parameters depending on the morphology or geometry of the object of interest. Moreover, they often rely on a set of heuristics and filter banks chosen by hand, and are not guaranteed to provide an orthogonal (non-redundant) basis for encoding the original patch.

Our features build upon the DZT [15] to provide a rotationally-invariant and orthogonal descriptor of circular patches. Zernike polynomials have long been used to build image moments with affine invariance [3], but a common problem in their application has been that they do not form a complete basis on the sampled disk [15]. As a consequence, the computation of moments usually requires redundant disk sampling and least-squares fitting, which leads to numerical difficulties [15].

The DZT avoids these problems, providing an orthonormal basis via non-redundant sampling of the unit disk and an orthogonal factorization of sampled Zernike polynomials [15]. This new disk decomposition offers other important benefits, such as a finite spectrum that allows the design of compact representations of input patches.

In what follows, we show how to build an orthogonal rotationally-invariant feature set from the DZT of an input patch. We refer the reader to the paper by Rafael *et al.* [15] for specific details about the DZT.

We define the DZT decomposition of a patch I as:

$$I\left(\rho,\theta\right) = \sum_{m,n} c_{m,n} Z_{m,n}\left(\rho,\theta\right), \tag{5}$$

where $c_{m,n}$ are the coefficients of the DZT, (ρ,θ) are polar coordinates indexing the image patch within the unit disk, and Z is the set of Zernike polynomials. The integers m and n index the radial and angular frequencies on the unit disk, with $n \in [0,1,2,...]$, $m \in [-n,n]$, $(n-m)$ even, and the total number of harmonics determined by the number of samples on the input disk [15].

The forward and inverse DZT can then be computed as $\mathbf{c} = \mathbf{Q^T i}$ and $\mathbf{i} = \mathbf{Qc}$, respectively, where $\mathbf{c}$ represents the vector of DZT coefficients, $\mathbf{i}$ the vector of input sam-

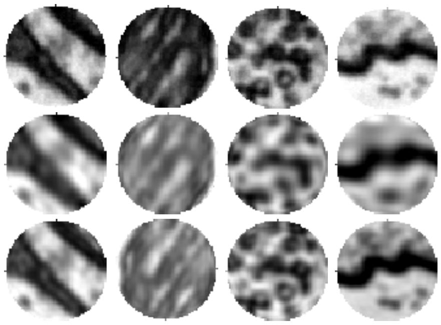


Figure 6: *Disk reconstructions from the inverse discrete Zernike transform to* $\mathrm{DZT_k}$ *for different values of* k. ***First row:*** *Original disks.* ***Second row:*** *Reconstructions for* $k = 10$. ***Third row:*** *Reconstructions for* $k = 15$. *Our Adapted Zernike Features retain the large visual cues from the original patch within only a few bands. The highest possible value of* k *in all these images is 73.*

ples, and $\mathbf{Q}$ the orthogonal basis that results from the QR decomposition of Z.

Grouping each radial band (*i.e.*, grouping the harmonics by their corresponding radial frequencies m), we define $\mathrm{DZT_k}$ as the low-pass filtering of the DZT spectrum that keeps only the first k radial bands of the discrete Zernike spectrum. Such filtering builds on the assumption that the low-frequency bands of the DZT spectrum are usually sufficient for classification [3] (see Fig. 6 for some reconstruction examples). Finally, since the phase of the DZT harmonics is linear with respect to rotations of the input patch [3], we can compute the rotation-normalized spectrum $\overline{\mathrm{DZT_k}}$ of $\mathrm{DZT_k}$ by shifting the phase of all of its harmonics by the phase of a low-frequency strong harmonic [3]. This allows us to obtain the same descriptor for rotated instances of the same patch by normalizing with respect to the orientation of large visual cues, which tend to be robust to noise and small differences between patches of the same class. We search for this harmonic by first starting with low-radial and low-angular frequencies and then moving into higher frequencies. There are several search strategies for identifying such harmonics [3], but choosing the first harmonic above a small threshold (50%) worked well in practice. Assuming that $|A_{\hat{m},\hat{n}}|e^{-j\phi_{\hat{m},\hat{n}}}$ is a strong normalizing harmonic (in polar form) we normalize each harmonic $A_{m,n}$ in $\mathrm{DZT_k}$ as $\overline{A_{m,n}} = A_{m,n}e^{-j\phi_{\hat{m},\hat{n}}}$.

Note that the QR decomposition is only computed once for all patches (*i.e.*, not for every patch). This way, we only need to compute $\mathbf{c} = \mathbf{Q^T i}$, prune the harmonics beyond a radial band k defined *a priori*, and apply the rotation normalization to obtain our descriptor for a given patch $\mathbf{i}$. In Section 5, we comment on the selection of the radial band k that we used for our experiments.

5. Experimental Results

In this section, we provide experimental evaluation of our fusion framework and our feature descriptor. We first evaluate our fusion method against previous solutions for neuron segmentation and general segment clustering. We then compare separately our Adapted Zernike Features against other connectomic-specific features and more general rotationally-invariant patch descriptors.

For all our tests we used four ssEM stacks of brain tissue from the somatosensory cortex of an adult mouse. Each stack is 1K x 1K pixels x 9 sections deep. The sections have an estimated thickness of 20 nm, and each pixel has an estimated physical dimension of 4 nm x 4 nm.

Evaluation of our fusion framework. We use IBM's CPLEX [1] for solving the binary integer programming problem of Eq. 4. In our experiments, we had an average of 20K variables per volume evaluated, and the solver took barely 5 seconds to determine the solution on a desktop PC.

We compare our fusion framework with two recently developed segment-clustering methods for connectomics: LP-R, a method for image co-clustering based on linear programming relaxation [20], and the agglomerative clustering approach employed in [9], which we refer to as AC. We also test against MHVS, a method that was developed recently to combine multiple pre-segmentations for video segmentation [17]. Finally, as a baseline and to evaluate the benefit of working with multiple segmentations, we include the results from running our fusion framework when only one 2D segmentation per section is provided (which we refer to as Fusion-1). For this last comparison, we obtain the 2D segmentation for each section by labeling each pixel with our pixel classifier.

In order to give all the methods the same initial advantage, we used the same pixel classifier (ours) to provide the initial 2D pre-segmentations. For those methods that required an initial 2D over-segmentation (e.g. LP-R and AC), we run our pixel classifier at higher boundary detection rates until every cell was initially over-segmented.

Fig. 7 shows the average percentage of split and merge errors made by each method with respect to a set of ground truth skeletons provided by an expert neuroscientist. Our fusion framework achieved the lowest error rate in our tests, with errors resulting mostly from false splits (*i.e.*, loss of continuity) along the Z axis and the creation of spurious small segments in areas between cells. The majority of neural cross-sections were reconstructured correctly, as we show in the examples in Fig. 9.

Evaluation of our features. We compare our Adapted Zernike Features with two connectomics-specific descriptors, the Ray features [13] and the Radon-like features [12]. In contrast to our features, these descriptors were designed

[1] ibm.com/software/integration/optimization/cplex-optimizer/

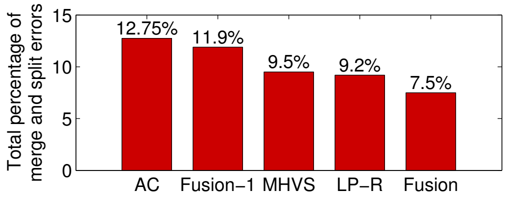


Figure 7: *Average percentage of merge and split errors in four ssSEM stacks with respect to the ground truth provided by an expert neuroscientist. Fusion outperforms other state-of-the-art methods demonstrating the benefit of explicitly enumerating possible segments and links for the final segmentation over solutions based on region clustering.*

to classify specific types of cellular structures such as mitochondria [13] and require adjusting several parameters depending on the structure of interest [12]. Since our features were designed with the goal of providing a robust rotation-invariant descriptor, we also test against more general rotation-invariant descriptors and filter banks such as MR8 [16], and the feature set LBP-HF [1].

We provide the precision-recall curves from the pixel labeling of mitochondria, cellular boundaries, vesicles, and cellular space in Fig. 8. In all comparisons, we used a random forest classifier with 300 trees [2]. For training, we used expert pixel annotations on the first four images of each stack. We tested on the rest of images in the stack. For our adapted Zernike features, we used the first 18 radial bands of the DZT spectrum and an input disk of 60 pixels in diameter after downsampling each section by a factor of two. In our tests, computing the feature descriptor and classifying every pixel of each testing stack takes less than 3 hours if run on a modern PC, and less than 10 min. if run on a grid with 25 machines. For MR8 and LBP-HF, we used the image patch that encloses such disk. For the other features, we used the parameters originally reported by the corresponding authors. In all our tests, our adapted Zernike features outperformed the competing descriptors.

6. Conclusions

We addressed the problem of automatic 3D segmentation of ssEM image stacks. We presented a framework that leverages information from multiple sections and considers several segmentation choices in each section to determine the final global partitioning. Unlike previous efforts that perform segmentation by clustering regions from an initial oversegmentation, our method is able to directly evaluate candidate segments that may be used for the final partitioning of each section.

Currently our framework does not handle neural mergers and splits, but are exploring several strategies to cope with them and hope to include them in future work.

Finally, we have also presented a highly discriminative set of rotationally-invariant features for connectomics. Our features can target cellular structures of different scales and morphologies and only require adjusting two parameters; the size of the input disk, and the number of radial bands in the DZT. Together, and individually, our features and fusion method gave the best segmentation and reconstruction results in our comparisons.

7. Acknowledgments

This work was supported in part by NSF Grant PHY-0835713, the Gatsby Charitable Trust, the Center for Brain Science at Harvard University, a Collaborative Innovation Award (X. Zhuang PI) from HHMI and the NIH. We also thank Shiv Vitaladevuni, Justo Arines and Verena Kaynig for providing us access to their code.

References

[1] T. Ahonen, J. Matas, C. He, and M. Piettikainen. Rotation invariant image description with local binary pattern histogram fourier features. In *SCIA*, 09. 5, 7

[2] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. 7

[3] J. Flusser, B. Zitova, and T. Suk. *Moments and Moment Invariants in Pattern Recognition*. Wiley Publishing, 2009. 5, 6

[4] G. González, F. Fleuret, and P. Fua. Learning rotational features for filament detection. In *CVPR*, 2009. 5

[5] V. Jain, B. Bollmann, M. Richardson, D. Berger, M. Helmstaedter, K. Briggman, W. Denk, J. Bowden, J. Mendenhall, W. Abraham, K. Harris, N. Kasthuri, K. Hayworth, R. Schalek, J. Tapia, J. Lichtman, and H. Seung. Boundary learning by optimization with topological constraints. In *CVPR*, 10. 1, 2

[6] E. Jurrus, M. Hardy, T. Tasdizen, P. Fletcher, P. Koshevoy, C. Chien, W. Denk, and R. Whitaker. Axon tracking in serial block-face scanning electron microscopy. *Medical Image Analysis*, 13(1):180–188, Feb. 2009. 2

[7] N. Kasthuri and J. W. Lichtman. Neurocartography. *Neuropsychopharmacology*, 35:342–343, Jan 2010. 1

[8] V. Kaynig, T. Fuchs, and J. Buhmann. Neuron geometry extraction by perceptual grouping in sstem images. In *CVPR*, 10. 1, 2

[9] V. Kaynig, T. J. Fuchs, and J. M. Buhmann. Geometrical consistent 3d tracing of neuronal processes in sstem data. In *MICCAI*, 10. 1, 2, 6

[10] A. Kreshuk, C. Straehle, C. Sommer, U. Koethe, G. Knott, and F. Hamprecht. Automated segmentation of synapses in 3d em data. In *ISBI*, 10. 5

[11] Y. Kubota, S. Hatada, and Y. Kawaguchi. Important factors for the three-dimensional reconstruction of neuronal structures from serial ultrathin sections. *Frontiers in Neural Circuits*, (0):12, 2009. 1, 2

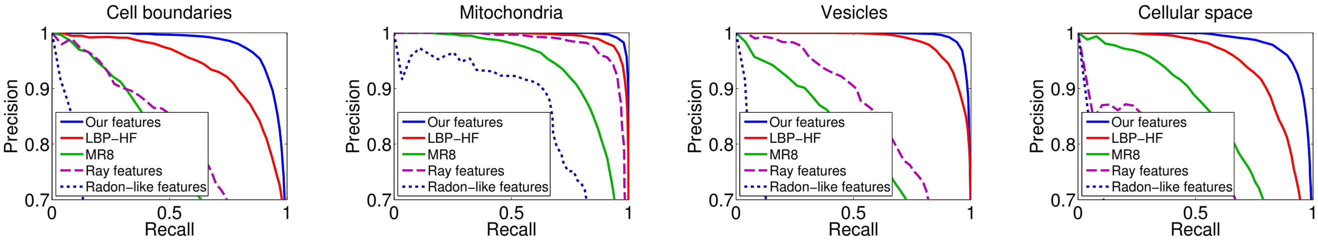
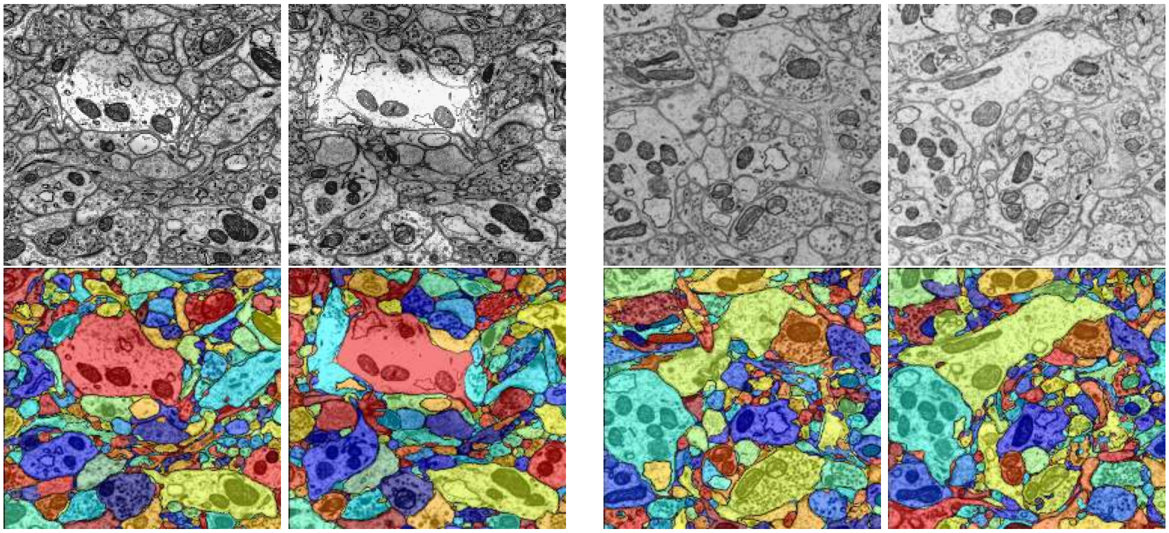


Figure 8: *Precision and recall curves for our Adapted Zernike Features and other competing features when labeling cellular structures. Our features are highly discriminative even when dealing with structures of different scales and morphologies, outperforming all the other descriptors on every class evaluated.*



(a) (b) (c) (d)

Figure 9: *Results of the fusion applied to two stacks of images. The pair* (a)-(b) *comes from one stack, and the pair* (c)-(d) *from a different stack. The images within each pair are nine sections apart. Our fusion framework correctly grouped most of the neurons in the stack, even in highly saturated regions of the images.*

[12] R. Kumar, A. Vázquez-Reina, and H. Pfister. Radon-like features and their application to connectomics. In *CVPRW*, 10. 1, 2, 5, 6, 7

[13] A. Lucchi, K. Smith, R. Achanta, V. Lepetit, and P. Fua. A Fully Automated Approach to Segmentation of Irregularly Shaped Cellular Structures in EM Images. *In MICCAI 10*. 2, 5, 6, 7

[14] Y. Mishchenko. Automation of 3D reconstruction of neural tissue from large volume of conventional serial section transmission electron micrographs. *Journal of Neuroscience Methods*, 176(2):276–289, Jan. 2009. 2

[15] R. Navarro, J. Arines, and R. Rivera. Direct and inverse discrete zernike transform. *Opt. Express*, 17(26):24269–24281, Dec 2009. 5

[16] M. Varma and A. Zisserman. A statistical approach to material classification using image patch exemplars. *PAMI*, 31(11):2032 –2047, 2009. 5, 7

[17] A. Vazquez-Reina, S. Avidan, H. Pfister, and E. Miller. Multiple hypothesis video segmentation from superpixel flows. In *ECCV*, 10. 6

[18] A. Vazquez-Reina, E. Miller, and H. Pfister. Multiphase geometric couplings for the segmentation of neural processes. In *CVPR*, 09. 2

[19] A. Veeraraghavan, A. Genkin, S. Vitaladevuni, L. Scheffer, S. Xu, H. Hess, R. Fetter, M. Cantoni, G. Knott, and D. Chklovskii. Increasing depth resolution of electron microscopy of neural circuits using sparse tomographic reconstruction. In *CVPR*, 10. 1

[20] S. Vitaladevuni and R. Basri. Co-clustering of image segments using convex optimization applied to em neuronal reconstruction. In *CVPR*, 10. 1, 2, 6

[21] M. J. Wainwright and M. I. Jordan. *Graphical Models, Exponential Families, and Variational Inference*. Now Publishers Inc., 2008. 3