

TRACKING NEURON RECONSTRUCTION IN CALCIUM IMAGING USING MARKOV RANDOM FIELD

S. Gulyanon¹, W. D. Tracy², L. He², and G. Tsechpenakis¹

¹Computer and Information Science Department, Indiana University-Purdue University Indianapolis, USA;

²Department of Biology, Indiana University, USA

ABSTRACT

It is known that the locomotion of an organism is controlled by proprioceptive neurons but their relationship is unknown. To observe the link between the larval locomotion and activities in sensory neurons, calcium images of proprioceptive neurons were acquired. One main issue of calcium images is that they produce low responses when there are no neuronal activities. In this work, we propose the novel neuron centerline tracking method that handles the issue in calcium images. It incorporates image appearance, neuron shape, and movement model in the Markov random field (MRF) framework, which is optimized using quadratic pseudo-boolean optimization (QPBO) with α -expansion, which is also known as *fusion moves*. We compare our method with state-of-the-art optical flow technique and illustrate that our method can track the dendrites even when they disappear from the images.

Index Terms— MRF, neuron tracking, articulated body, calcium images.

1. INTRODUCTION

Sensory neurons beneath the epidermis of larval *Drosophila* provide important proprioceptive feedback to the brain. In the absence of sensory signals from these proprioceptors, larval locomotion is uncoordinated and slow. Although the signals of these sensory neuron is important to proper locomotion, it is unknown exactly what these signals are. For instance, what is the relationship of neuronal firing to the muscle contractions in a segment? Are the sensory neurons activated by stretch during muscle relaxation? Or are the sensory neurons activated during the contraction of the muscle within a segment.

To address these questions, we have developed an imaging process to visualize the calcium responses in the proprioceptive neurons during larval locomotion. Calcium imaging reveals the spatio-temporal information of activities in neurons at the single-cell level. Due to the large amount of volume sequences, the automated analysis system is required. The

main requirement of the system is the capability to track the location of individual neurons, including dendrites and soma, from time-lapse calcium images. One unique characteristic of calcium images is that they produce low responses when there are no activities on neurons so parts of dendrites become invisible. Previous tracking systems weren't designed for calcium images so they fail to detect neurons when dendrites disappear and the error will propagate to the rest of the image sequence.

Object tracking and detection have been one of the most studied topic in computer vision and many techniques have been proposed. One possible solution based on detection technique is to automatically trace neurons in each frame independently using neuron reconstruction methods like in [1, 2]. Then, the registration technique (e.g., [3]) can be applied to improve the temporal smoothness and obtain the trajectory of the neuron. Registration and tracing could be solved simultaneously using method in [4]. However, calcium image sequences contain frames with low intensity dendrites so these techniques will produce errors because they fail to detect and/or trace neurites.

An alternative is the optical flow based methods [5, 6]. They produce dense motion field over the entire image. Most optical flow methods follow the brightness constancy constraint equation [7], which assumes that pixel intensity remains constant during the displacement. However, this assumption is not true in the calcium images whose responses change with the activities on neurons. Moreover, tracking over background regions, which occupy most of the area, is misleading as the motion field of noises is regularized over the neurite regions.

Other methods [8, 9] integrate the domain knowledge in the neuron's tree-structure shape and motion like the kinematic structure of articulated body. Authors represented the human body by the loose-limbed model, which is the variation of the pictorial structure [10, 11]. The problem is formulated as the graphical model where parts are sites and their configuration contains the position and orientation of parts. Edges encode the position and angle relationships between adjacent parts. This model allows us to track only the regions of interests and estimate the morphology of the neuron, which is also the information we are interested in.

This work is supported by NSF/DBI [#1252597]: 'CAREER: Modeling the structure and dynamics of neuronal circuits in the *Drosophila* larvae using image analytics' awarded to G. Tsechpenakis.

The previous works on pose estimation and tracking articulated body are not effective on neuron tracking because neurons have higher degrees of freedom compared to human body model owing to larger number of branch points. Second, the number of degrees of freedom changes across neurons while it is fixed in human body. Neuron tracking problem has the massive search space due to larger degrees of freedom and unreliable image cues compared to self-occlusion and appearance variation from clothing in human-body tracking.

To address these problems, our method is based on the generalized pictorial structure for any tree structure. We tackle the unreliability of image intensity by incorporating both image appearance and domain knowledge, including the shape and motion models. Here we describe 5 factors that drive our tracking system - a) image feature, b) dynamics model, c) transition similarity, d) part-based model, and e) shape smoothness.

2. METHOD

Let the centerline of a neuron in each frame is represented by the pictorial structure model [10, 11], $\mathcal{G}^t = \{\mathbf{X}^t, \mathcal{E}_R\}$ (fig. 1). Let $\mathbf{X}^t = \{\mathbf{x}_i^t\}$ be a set of tree structure nodes $i \in \{1, \dots, n\}$ at time $t \in [0, T]$, and $\mathcal{E}_R = \{e_{ij}\}$ be a set of tree structure edges. Each node contains the Cartesian coordinate of the centerline, $\mathbf{x}_i^t = \{x, y\}$, where $x, y \in \mathbb{R}$. Let $K_n = \{\mathcal{V}, \mathcal{E}\}$ be the complete graph with n vertices and \mathcal{E}_G be a set of repulsive edges [12], which are not tree structure edges, $\mathcal{E}_G = \mathcal{E} \setminus \mathcal{E}_R$. Let $\mathbf{V}^t = \{\mathbf{v}_i^t\}$ be a set of mapping vectors that transforms \mathbf{X}^{t-1} to \mathbf{X}^t , so $\mathbf{v}_i^t = \mathbf{x}_i^t - \mathbf{x}_i^{t-1}$.

The objective function is formulated as the conditional distribution over the set of neuron's shape $\mathbf{S}^t = \{\mathbf{X}^t, \mathcal{E}\}$ for each frame, given the image sequence $\mathbf{I} = \{\mathbf{I}^t\}$ and previous neuron's shape $\mathbf{S}^{0:t-1}$ (fig. 1),

$$\begin{aligned} \hat{\mathbf{S}}_{MAP}^t &= \arg \max_{\mathbf{S}^t} P(\mathbf{S}^t | \mathbf{I}, \mathbf{S}^{0:t-1}) \\ &= \arg \max_{\mathbf{X}^t} P(\mathbf{X}^t | \mathbf{I}, \mathbf{S}^{0:t-1}) \end{aligned} \quad (1)$$

We assume that the topology of the neuron is fixed, so \mathcal{G}^t is isomorphic for every time t and only the local shape of the neuron deforms. So we optimize the objective function by modifying the position of the centerline \mathbf{X}^t . We modeled the probabilistic distribution of the centerline using the second-order multi-label MRF, where sites are nodes $\mathbf{x}_i^t \in \mathbf{X}^t$ and their configuration is every possible coordinate.

$$\begin{aligned} P(\mathbf{X}^t | \mathbf{I}, \mathbf{S}^{0:t-1}) &\propto \exp \left\{ - \sum_{\mathbf{x}_i^t \in \mathbf{X}^t} E(\mathbf{x}_i^t | \mathbf{I}, \mathbf{S}^{t-1}, \mathbf{S}^0) \right\} \\ E(\mathbf{x}_i^t | \mathbf{I}, \mathbf{S}^{t-1}, \mathbf{S}^0) &= A(\mathbf{x}_i^t | \mathbf{I}, \mathbf{S}^0) + \sum_{e_{ij} \in \mathcal{E}} I(\mathbf{x}_i^t, \mathbf{x}_j^t | \mathbf{S}^{t-1}) \\ &\quad + \sum_{e_{ip}, e_{iq} \in \mathcal{E}_R, p \neq q} H(\mathbf{x}_p^t, \mathbf{x}_i^t, \mathbf{x}_q^t) \end{aligned} \quad (2)$$

where \mathbf{S}^0 is the neuron trace initialization, e_{ij} is the edge incident to the node \mathbf{x}_i^t . A is the unary potential function (sec. 2.1), I is the

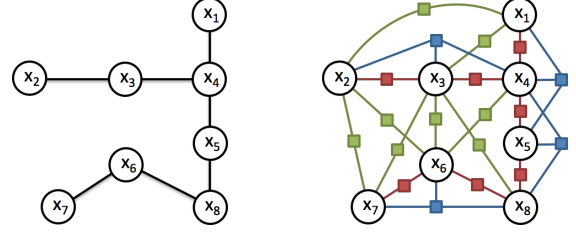


Fig. 1. shows the tree structure or pictorial structure of the neuron (left) and our corresponding graphical model (right). Red cliques \mathcal{E}_R form the tree structure of neuron. Green cliques \mathcal{E}_G represent the repulsive edges, and blue cliques are the tertiary potential. Some of green and blue cliques were left out for the sake of visualization.

pairwise smoothness (sec. 2.2), and H is the tertiary smoothness. Section 2.3 will show that the careful choice of the tertiary term allows us to decompose it into a pairwise functions without adding an auxiliary node; therefore, the graphical model can be described by the graph $\mathbf{S}^t = \{\mathbf{X}^t, \mathcal{E}\}$.

2.1. Unary Potential

The unary potential assigns the position of the centerline based on image feature and dynamics model.

$$A(\mathbf{x}_i^t | \mathbf{I}, \mathbf{S}^0) = A_{img}(\mathbf{x}_i^t | \mathbf{I}) + A_{model}(\mathbf{x}_i^t | \mathbf{I}, \mathbf{S}^0) \quad (3)$$

Image feature assigns the node's position based on the response from the Frangi filter [13] for dendrites and the Hessian-based function for branch points. Frangi filter is a vesselness-filter designed to detect only the line-like structure, so a different function is required for the branch point (fig. 2). In 2D, the branch point appears like a blob so we design a function similar to Frangi filter that has high value when both eigenvalues of Hessian matrix have high magnitude.

$$\begin{aligned} A_{img}(\mathbf{x}_i^t | \mathbf{I}) &= \\ &\begin{cases} \exp\left(-\frac{R_B^2}{2\beta^2}\right) \left(1 - \exp\left(-\frac{S^2}{2c^2}\right)\right) & , \text{ if } \deg(\mathbf{x}_i^t) \leq 2 \\ \left(1 - \exp\left(-\frac{R_A}{2\beta^2}\right)\right) \left(1 - \exp\left(-\frac{S^2}{2c^2}\right)\right) & , \text{ otherwise} \end{cases} \end{aligned} \quad (4)$$

where $\deg(\mathbf{x}_i^t)$ is the degree of the node \mathbf{x}_i^t , which is less than or equal to two if they are dendrite nodes, otherwise, they are branch points. $R_B = \lambda_1/\lambda_2$ measures the deviation from a blob-like to a line-like structures, where λ_k is the eigenvalue of the Hessian matrix at position \mathbf{x}_i^t with the k -th smallest magnitude. $R_A = \sqrt{\min(\lambda_1, 0) \cdot \min(\lambda_2, 0)}$ measures the blob-like structure similarity. For bright neurite over dark background, both eigenvalues have negative value at branch points. $S = \lambda_1^2 + \lambda_2^2$ distinguishes the neurite from the background. β and c are parameters controlling the sensitivity of R_A , R_B , and S [13].

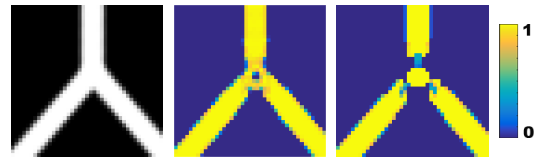


Fig. 2. shows the synthetic image of a branch point (left) with its corresponding Frangi filter (middle) and our Hessian-based function (right) from eq. 4. Frangi filter gives low response at the center of the branch point while our Hessian-based function gives high response at the branch point and low value to surroundings.

Dynamics model chooses the position of node \mathbf{x}_i^t that resembles to its position from our motion model, which is governed by the soma's location. Image feature alone is not enough to locate dendrites, which may disappear in calcium images. The dynamics model solves this problem by introducing the motion model or the initial guess of the centerline location, which can be seen as the prior, to complement the image feature. Although dendrites are difficult to detect, on the other hand, the soma is very clear and easy to locate. Given \mathbf{S}^0 , our motion model is obtained by shifting the initial trace so that the locations of soma agree,

$$A_{model}(\mathbf{x}_i^t | \mathbf{I}, \mathbf{S}^0) = \exp(\alpha_d \cdot \|\mathbf{x}_i^t - T_{0,t}(\mathbf{x}_i^0)\|) \quad (5)$$

where $\|\cdot\|$ is the Euclidean distance. α_d is the parameter with negative value that tunes the sensitivity of the dynamics model. T_{t_1, t_2} is the translation operator that shifts the soma's position at time t_1 to its position at time t_2 , so $T_{t_1, t_2} = \sum_{t=t_1+1}^{t_2} \mathbf{v}_0^t$, where $\mathbf{v}_0^t = \mathbf{x}_0^t - \mathbf{x}_0^{t-1}$ and \mathbf{x}_0^t are the mapping vector and position of the soma at time t , respectively. The positions of the soma is located by the Circle Hough Transform technique [14, 15] and tracked along the optical flow computed by the FullFlow method [5].

2.2. Pairwise Potential

The pairwise smoothness assigns the position of node such that the mapping function is smooth and the transformed neuron maintains the global shape. It consists of the transition similarity and the part-based model energies. These two types of energies are applied on two different types of edges, \mathcal{E}_R and \mathcal{E}_G ,

$$I(\mathbf{x}_i^t, \mathbf{x}_j^t | \mathbf{S}^{t-1}) = \begin{cases} \alpha_p \cdot \exp(\alpha_t \cdot \|\mathbf{v}_i^t - \mathbf{v}_j^t\|) & , \text{if } e_{ij} \in \mathcal{E}_R \\ \left[\|\mathbf{x}_i^t - \mathbf{x}_j^t\| > \langle \|\mathbf{x}_k^t - \mathbf{x}_l^t\| \rangle_{e_{kl} \in \mathcal{E}_R} \right]_{e_{ij} \in \mathcal{E}_G} & , \text{if } e_{ij} \in \mathcal{E}_G \text{ and } br(\mathbf{x}_i^t) \neq br(\mathbf{x}_j^t) \\ 0 & , \text{otherwise} \end{cases} \quad (6)$$

where $[\cdot]$ is the indicator function taking value 1 if a given predicate is true, otherwise, value 0. $\langle \cdot \rangle$ is the expectation value. $br(\mathbf{x}_i^t)$ is the branch number in which the node \mathbf{x}_i^t is. $\alpha_p = (2 \langle \|\mathbf{v}_i^t - \mathbf{v}_j^t\| \rangle)^{-1}$ scales small and large vector differences to a proper range [16], while α_t tunes the sensitivity of the exponential term.

Transition similarity enforces the smoothness of mapping vectors between adjacent nodes. It is defined by the function of the mapping vector difference over tree structure edges $e_{ij} \in \mathcal{E}_R$. It encourages adjacent nodes to have a similar mapping vector.

Part-based model constrains the morphology of the neuron using the repulsive edges [12]. It makes dendrites to repel each other and prevents them from creating any unnecessary crossover and deforming the morphology. Repulsive edges \mathcal{E}_G are added between nodes from different branches. They give higher energy when dendrites overlap than when they don't. It is computed by thresholding the distance between branches.



Fig. 3. shows that although the curvature and our decomposition are different, they share the same optimum when $\mathbf{x}_p^t \mathbf{x}_q^t = \mathbf{x}_p^t \mathbf{x}_i^t + \mathbf{x}_i^t \mathbf{x}_q^t$ (left), and discourage bending when $\mathbf{x}_p^t \mathbf{x}_q^t < \mathbf{x}_p^t \mathbf{x}_i^t + \mathbf{x}_i^t \mathbf{x}_q^t$ (right).

2.3. Tertiary Potential

Tertiary potential ensures that shape characteristics of neurons is inherited. It is decomposed into pairwise terms for efficient inference.

Shape smoothness ensures the smoothness of dendrites by encouraging low curvature and short length like in the active contour model [17]. Computing curvature requires the tertiary potential,

$$H(\mathbf{x}_p^t, \mathbf{x}_i^t, \mathbf{x}_q^t) = (\mathbf{x}_p^t - \mathbf{x}_i^t)^2 + (\mathbf{x}_p^t - 2\mathbf{x}_i^t + \mathbf{x}_q^t)^2 \quad (7)$$

Unlike [18], our decomposition avoids adding auxiliary nodes by using the triangle inequality, where the clique of three nodes $\{\mathbf{x}_p^t, \mathbf{x}_i^t, \mathbf{x}_q^t\}$ is broken down into three pairwise cliques with no extra auxiliary nodes - $\{\mathbf{x}_p^t, \mathbf{x}_i^t\}$, $\{\mathbf{x}_i^t, \mathbf{x}_q^t\}$, and $\{\mathbf{x}_p^t, \mathbf{x}_q^t\}$. Consider a triangle whose corners are defined by nodes \mathbf{x}_p^t , \mathbf{x}_i^t , and \mathbf{x}_q^t . The optimized curvature value is obtained when the triangle forms the straight line, $\mathbf{x}_p^t \mathbf{x}_q^t = \mathbf{x}_p^t \mathbf{x}_i^t + \mathbf{x}_i^t \mathbf{x}_q^t$, so the simplification has the same optimum as the potential in eq. 7 (fig. 3), which is defined as,

$$H(\mathbf{x}_p^t, \mathbf{x}_i^t, \mathbf{x}_q^t) = \alpha_s \cdot \left\{ \|\mathbf{x}_p^t - \mathbf{x}_i^t\|^2 + \|\mathbf{x}_i^t - \mathbf{x}_q^t\|^2 - \frac{\|\mathbf{x}_p^t - \mathbf{x}_q^t\|^2}{4} \right\} \quad (8)$$

where α_s tunes the sensitivity of shape smoothness. The quadratic terms encourage the short distance between nodes. This term encourages both rigidity and elasticity like the active contour model. The third term is divided by 4 to ensures that $H(\mathbf{x}_p^t, \mathbf{x}_i^t, \mathbf{x}_q^t) \leq 0$.

3. IMPLEMENTATION

The position of centerline \mathbf{X}^t is discretized so the problem becomes feasible. Algorithm 1 shows the summary of our method, given \mathbf{I} and \mathbf{S}^0 . In line 6, our energy function in eq. 2 is non-submodular so MRF is optimized using the α -expansion [19] with QPBO [20, 21], also known as *fusion moves* [22].

Algorithm 1 Neuron tracking

```

1: procedure TRACKING(  $\mathbf{I}, \mathbf{S}^0$  )
2:   for  $t = 1$  to  $T$  do
3:     Compute  $A(\mathbf{x}_i^t | \mathbf{I}, \mathbf{S}^0)$  using eq. 4 and 5
4:     Compute  $I(\mathbf{x}_i^t, \mathbf{x}_j^t | \mathbf{S}^{t-1})$  using eq. 6
5:     Compute  $H(\mathbf{x}_p^t, \mathbf{x}_i^t, \mathbf{x}_q^t)$  using eq. 8
6:      $\mathbf{S}^t \leftarrow \arg \max_{\mathbf{X}^t} P(\mathbf{X}^t | \mathbf{I}, \mathbf{S}^{0:t-1})$ 
7:   return  $\mathbf{S}$ 

```

For efficiency, the search space is limited by reducing the number of candidate positions and pruning the redundant edges from \mathbf{S}^t .

To limit candidate positions, we consider regions within 20 pixels around the starting location, which is the possible node location based on previous trace. In this work, the starting location of each node at time t is computed by applying the translation operator that shifts soma's location from previous time-step to the current position, $\tilde{\mathbf{x}}_i^t = T_{t-1, t}(\mathbf{x}_i^{t-1})$.

Redundant edges are repulsive edges linking nodes that are unlikely to overlap so removing these edges makes no difference to the model. \mathbf{S}^t is pruned such that there is at most one edge between a node and a branch, where the shortest edge is selected. Edges incident to other branch nodes are redundant. With these schemes, we limit the number of repulsive edges $|\mathcal{E}_G| = \mathcal{O}(N)$. Thus, the total the number of edges in our model is also $\mathcal{O}(N)$.

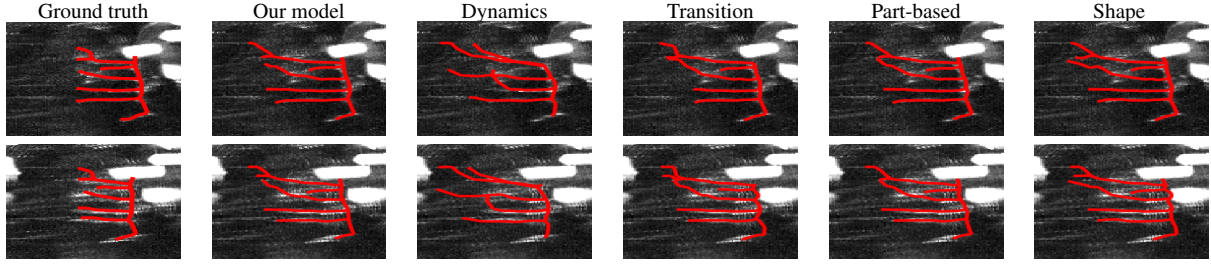


Fig. 4. Tracking results (in red color) on 2 adjacent frames in each row. From left to right, the ground truth, our results, and results excluding one of the factors - dynamics model, transition similarity, part-based model, and shape smoothness respectively.

4. RESULTS

For image acquisition, larvae expressing the genetically encoded calcium sensor GCaMP6.0F in the proprioceptors were observed in three dimensional confocal microscopy volumes over time. This is possible because the neurons are found directly beneath the cuticle and epidermis which are optically transparent. Data analyses of the neurons in these volumes presents specific challenges that are not present in typical calcium time series because the neurons of interest are moving in the X, Y, Z and time axes.

In this dataset, each sequence captures a few motions of neurons and the duration of each motion is around 10-20 frames. The ground truth is manual tracing of neurons during motions. The neuron reconstructions were produced using the software called *neuTube* [23]. Three dissimilarity metrics were used in this work - spatial distance (SD), substantial SD (SSD), and the percentage of sample points used in SSD (%SSD). $SD(A,B)$ is defined by the mean of the average Euclidean distance of nodes in A to the closest point in B and the other way round. Nodes in A and B are resampled such that the distance between adjacent nodes is one voxel to remove the bias from trace node sampling. To emphasize the magnitude of errors, SSD is defined as the SD over nodes that are apart from the other neuron at least 2 voxels, while %SSD indicates the inconsistency between two neurons, detailed in [24]. We compared our method with the optical flow based tracking called ‘FullFlow’ [5]. The experiment was carried out on Mac Pro (2×2.66 GHz 6-Core Intel Xeon, 20GB 1333MHz DDR3 ECC); the run time of our method was around one second per frame per neuron. The dimension of each frame is around 512×192 pixel. Tracking was done on 2D maximum projection of 3D images, albeit the availability of image stacks, to eliminate the color bleach effect that is more likely to occur over Z-axis due to image acquisition process, where image slices were taken in sequence.

Fig. 4 shows the importance of each factor by displaying results when one of the factors is excluded. Dynamics model is required to compensate for the unreliability of image intensity. Without this fac-

tor, nodes move towards the pixel with high intensity regardless of neuron’s shape. Transition similarity prevents rapid changes in adjacent nodes resulting in smooth mapping. Part-based model ensures dendrites avoid overlapping. Finally, shape smoothness smooths out the dendrites; otherwise, crooked curves are detected due to dip in intensity over neurites.

We evaluate our method and competitor on sequences with low and high signal-to-noise ratio (SNR) to illustrate the advantage of our method in fig. 5. The box plot of SD score in the first column shows that our results are comparable to the competitor when SNR is high but ours outperform the other method when SNR is low. This observation is supported by the qualitative results in fig. 6. It illustrates that optical flow based method fails to recover the motion field, resulting in poor tracked neuron centerline. While our method integrates shape and motion information with intensity to produce the robust tracking results. SSD and %SSD are computed for each frame in the second and third columns respectively. They show that our method gives high error when neurons move, especially %SSD that peaks at the middle of the motions. The error arises from the out-of-focus problem because the imaging system acquires blurry images when the organism moves. Although a large portion of nodes were tracked incorrectly, the magnitude of errors is low as shown by SSD scores, which is around 5 pixels on the testing data.

5. CONCLUSION

We introduced the novel neuron tracking method that allows the automated analysis of calcium image sequence at the single-cell level. Our tracking method is influenced by image appearance, neuron shape characteristics, and movement model. They are integrated under the non-submodular second-order multi-label MRF framework over the articulated body of neurons. Our simplification of tertiary potential enables efficient inference. We evaluated our method using noisy calcium image sequences of *Drosophila* sensory neurons, and we compared our results with optical flow based tracking.

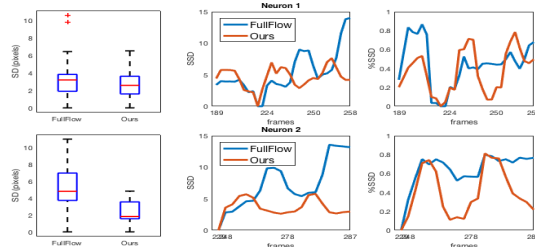


Fig. 5. shows three dissimilarity scores - SD, SSD, and %SSD, of FullFlow [5] and our method. Rows illustrate the tracking results when SNR is high (top) and low (bottom). SD is displayed in box plot, while SSD and %SSD are displayed for every frame.

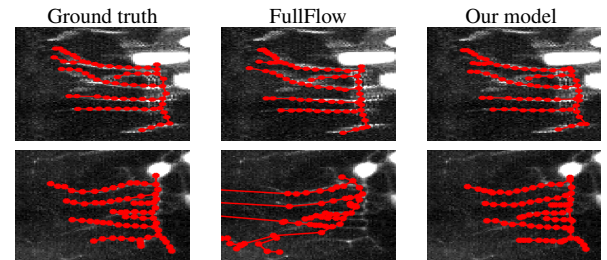


Fig. 6. shows the ground truth (left), results from FullFlow (middle), and ours (right) in red color. Rows show tracking results of a frame when SNR is high (top) and low (bottom).

6. REFERENCES

- [1] S. Gulyanov, N. Sharifai, S. Bleykhman, E. Kelly, M. D. Kim, A. Chiba, and G. Tsechpenakis, "Three-dimensional neurite tracing under globally varying contrast," *ISBI*, pp. 875–879, 2015.
- [2] S. Gulyanov, N. Sharifai, M. D. Kim, A. Chiba, and G. Tsechpenakis, "Crf formulation of active contour population for efficient three-dimensional neurite tracing," *ISBI*, pp. 593–597, 2016.
- [3] S. Farhand, S. Gulyanov, N. Sharifai, M. D. Kim, A. Chiba, and G. Tsechpenakis, "Temporal neurite registration using hierarchical alignments," *ISBI*, pp. 334–338, 2016.
- [4] P. Glowacki, M. A. Pinheiro, A. Mosinska, E. Turetken, D. Lebrecht, A. Holtmaat, R. Sznitman, J. Kybic, and P. Fua, "Reconstructing evolving tree structures in time lapse sequences by enforcing time-consistency," *TPAMI*, vol. PP, no. 99, pp. 1–1, 2017.
- [5] Q. Chen and V. Koltun, "Full flow: Optical flow estimation by global optimization over regular grids," *CVPR*, pp. 4706–4714, 2016.
- [6] C. Bailer, K. Varanasi, and D. Stricker, "Cnn-based patch matching for optical flow with thresholded hinge loss," *CVPR*, 2017.
- [7] D. Fortun, P. Bouthemy, and C. Kervrann, "Optical flow modeling and computation: A survey," *CVIU*, vol. 134, pp. 1–21, 2015.
- [8] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard, "Tracking loose-limbed people," *CVPR*, vol. 1, pp. I-421–I-428, 2004.
- [9] T. X. Han and T. S. Huang, *Articulated Body Tracking Using Dynamic Belief Propagation*, pp. 26–35, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.
- [10] M. A. Fischler and R. A. Elschlager, "The representation and matching of pictorial structures," *IEEE Trans. Comput.*, vol. C-22, no. 1, pp. 67–92, 1973.
- [11] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *IJCV*, vol. 61, no. 1, pp. 55–79, 2005.
- [12] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, "Pose search: Retrieving people using their pose," *CVPR*, pp. 1–8, 2009.
- [13] A. F. Frangi, W. J. Niessen, K. L. Vincken, and M. A. Viergever, "Multiscale vessel enhancement filtering," *MICCAI*, pp. 130–137, 1998.
- [14] R. O. Duda and P. E. Hart, "Use of the hough transformation to detect lines and curves in pictures," *Commun. ACM*, vol. 15, no. 1, pp. 11–15, 1972.
- [15] T. J. Atherton and D. J. Kerbyson, "Size invariant circle detection," *Image and Vision Computing*, vol. 17, no. 11, pp. 795–803, 1999.
- [16] Y. Y. Boykov and M. P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in n-d images," *ICCV*, vol. 1, pp. 105–112, 2001.
- [17] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *IJCV*, vol. 1, no. 4, pp. 321–331, 1988.
- [18] O. Woodford, P. Torr, I. Reid, and A. Fitzgibbon, "Global stereo reconstruction under second-order smoothness priors," *TPAMI*, vol. 31, no. 12, pp. 2115–2128, 2009.
- [19] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *TPAMI*, vol. 23, no. 11, pp. 1222–1239, 2001.
- [20] E. Boros and P. L. Hammer, "Pseudo-boolean optimization," *Discrete Appl. Math.*, vol. 123, no. 1-3, pp. 155–225, 2002.
- [21] V. Kolmogorov and C. Rother, "Minimizing nonsubmodular functions with graph cuts-a review," *TPAMI*, vol. 29, no. 7, pp. 1274–1279, 2007.
- [22] V. Lempitsky, C. Rother, S. Roth, and A. Blake, "Fusion moves for markov random field optimization," *TPAMI*, vol. 32, no. 8, pp. 1392–1405, 2010.
- [23] L. Feng, T. Zhao, and J. Kim, "neutube 1.0: a new design for efficient neuron reconstruction software based on the swc format," *eNeuro*, 2015.
- [24] H. Peng, Z. Ruan, F. Long, J. H. Simpson, and E. W. Myers, "V3d enables real-time 3d visualization and quantitative analysis of large-scale biological image data sets," *Nat Biotech*, vol. 28, no. 4, pp. 348–353, 2010.