https://doi.org/10.2352/ISSN.2470-1173.2017.17.COIMG-417

A Model Based Neuron Detection Approach using Sparse Location Priors

Soumendu Majee*[1]*, Dong Hye Ye*[1]*, Gregery T. Buzzard*[2] *and Charles A. Bouman*[1]*;
[1]***School of ECE, Purdue University, West Lafayette, IN, USA;***
[2]***Department of Mathematics, Purdue University, West Lafayette, IN, USA;***

Abstract

In order to accurately monitor neural activity in a living mouse brain, it is necessary to image each neuron at a high frame rate. Newly developed genetically encoded calcium indicators like GCaMP6 have fast kinetic response and can be used to target specific cell types for long duration. This enables neural activity imaging of neuron cells with high frame rate via fluorescence microscopy. In fluorescence microscopy, a laser scans the whole volume and the imaging time is proportional to the volume of the brain scanned. Scanning the whole brain volume is time consuming and fails to fully exploit the fast kinetic response of new calcium indicators. One way to increase the frame rate is to image only the sparse set of voxels containing the neurons. However, in order to do this, it is necessary to accurately detect and localize the position of each neuron during the data acquisition.

In this paper, we present a novel model-based neuron detection algorithm using sparse location priors. We formulate the neuron detection problem as an image reconstruction problem where we reconstruct an image that encodes the location of the neuron centers. We use a sparsity based prior model since the neuron centers are sparsely distributed in the 3D volume. The information about the shape of neurons is encoded in the forward model using the impulse response of a filter and is estimated from training data. Our method is robust to illumination variance and noise in the image. Furthermore, the cost function to minimize in our formulation is convex and hence is not dependent on good initialization. We test our method on GCaMP6 fluorescence neuron images and observe better performance than widely used methods.

Introduction

Mapping the functional connectivity in the brain continues to be a major challenge in neuroscience. An important component in developing such a map is the ability to measure the activity of many neurons accurately at a high frame rate. Fluorescence microscope imaging using recently developed fluorescent calcium indicators is a good candidate for making these measurements due to a combination of high spatial and temporal resolution. The foundation of this form of neural imaging is the increase in calcium ion concentration whenever a neuron is activated [1]; this increased concentration leads to fluorescence from the calcium indicator, which is captured by a fluorescence microscope [2] [3]. Recently developed calcium indicators like GCaMP6 respond to changes in calcium ion concentration quickly and thus provide the possibility of measuring neural activity at a high frame rate [4].

In conventional raster scan fluorescence microscopy, in which all the voxels in the volume are scanned equally often, the frame rate suffers markedly as the size of the brain volume to be imaged increases [5] [6]. However, for purposes of capturing neural activity, it is necessary to scan only the volume containing the neurons, which is typically a small fraction of the total volume. Thus, replacing a full raster scan for each frame with a scan of the much smaller neural volume can dramatically increase the scan frame rate. Of course, this strategy relies on accurate knowledge of the neural volume, which we propose to obtain using a single GCaMP6 scan of the full volume of interest.

The central challenge of this research is to create a method for accurately detecting and determining the location of each neuron in a GCaMP6 image so that the volume can be sparsely scanned at a high frame rate. This goal presents a number of difficult challenges. First, there can be large illumination variations across the image, along with high noise. Furthermore, there is often fluorescence from the dense network of axons and dendrites, referred to as neuropil. There are also cylindrical blood vessels in the images that have similarities with neurons, making the detection of neurons more challenging. Also, some neurons are saturated with fluorescence; this is an artifact of GCaMP overexpression that leads to a significantly different type of calcium fluorescent image relative to other neurons [1] [3].

Before describing our method, we briefly review related work on cell segmentation and tracking methods, including frame to frame association and active contour models.

Several algorithms [7] [8] track cells in a sequence of 2D images by detecting cells in each frame and then establishing frame to frame association based on several criteria. However, there are problems in the frame to frame association phase if there is large variation in illumination from frame to frame, a high density of cells, and/or a large number of cells entering or leaving the frame [9] [10].

Active contour models have also been proposed for segmenting and tracking cells in microscopy images [11] [12] [13] [14] [15]. Active contour based methods take a deformable contour with a certain energy function and minimize that energy to segment a cell. However a good initialization is often required to avoid getting stuck on a local minima [16]. Active contours require a data attachment term to guide the contour. Many methods use a data attachment term that depends solely on the edge map computed from the image gradient [17] [18] [19]. However, in images with high noise, gradient information is unreliable. Other methods use statistical intensity information computed for the region inside the contour [20] [21].

Active contour models have been extended to include strong prior shape information in [22] [23] [24] [25]. However it is difficult to detect an unknown number of objects [26]. To mitigate

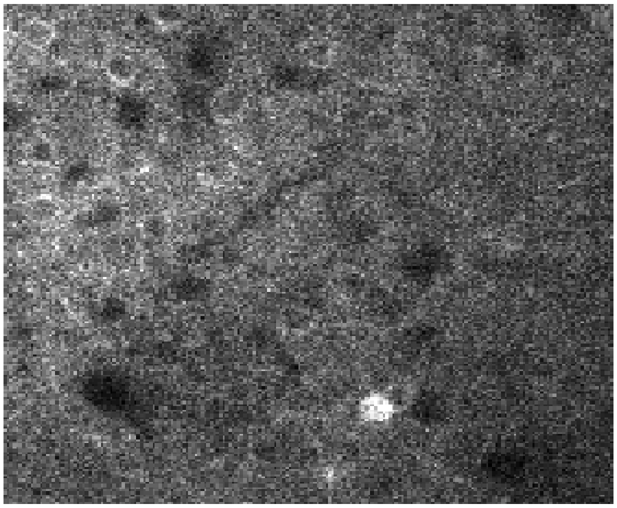


Figure 1. *Example of a GCaMP6 calcium fluorescence image used in this study. The bright circle is a neuron affected with GCaMP over-expression. The large black circles are blood vessels. The annular rings are normal neurons.*

this, variations of active contour with Marked Point process is proposed in [26] [27]. But these methods are formulated for 2D images and cannot be directly applied to 3D images.

In this paper, we solve the problem of detecting neuron centers in a 3D volume using a novel model-based neuron detection algorithm (MBND). We formulate this as an image reconstruction problem in which we construct a sparse location image that encodes the neuron centers. That is, a non-zero value in the location image corresponds to the center of a neuron. We model the observed neurons as a linear combination of shape models in the forward model and use separate shape models for normal neurons and abnormal over-expressed neurons; the coefficients in this linear combination are the nonzero entries in the location image. We use a sparsity based prior model for the construction of the location image since the neuron centers are sparsely distributed in the 3D volume. We model the background illumination variation in the image as a linear combination of low-frequency basis vectors of the Discrete Cosine Transform, and we model the presence of sparsely distributed bright dendrites as additive impulsive noise. We compute the MAP estimate of the location image by minimizing a convex cost function. Tests on GCaMP6 fluorescence neuron images show better accuracy than widely used methods.

Forward and Prior Model

To formulate the problem, let us denote the observed vectorized image as $Y \in \mathbb{R}^N$, where N is the total number of voxels in the 3D image. Let S denote the lattice in $\mathbb{R}^3$ representing the voxel locations in the image and let Ω map from the lattice points in S to an index i such that $1 \leq i \leq N$. Then $Y_{\Omega(s)}$ is the value of the image at the voxel location $s \in S$

Let $A^{(1)}, A^{(2)} \in \mathbb{R}^{N \times N}$ be matrices corresponding to linear space invariant filters. The impulse responses of $A^{(1)}, A^{(2)}$ are the shape model of normal and over-expressed neurons. $A^{(1)}, A^{(2)}$ perform convolution with the shape model as their convolution kernel.

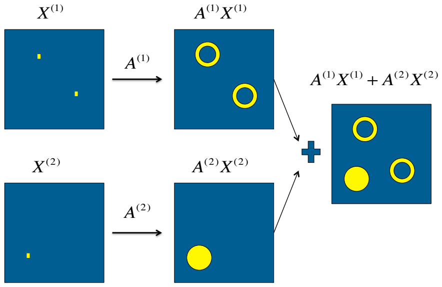


Figure 2. *Illustration of how $A^{(1)}X^{(1)} + A^{(2)}X^{(2)}$ model the neurons in the image. The ring shapes represent normal neurons and the disk shaped ones represent over-expressed neurons*

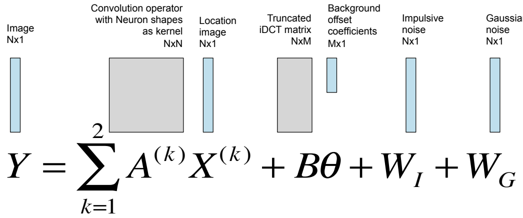


Figure 3. *Illustration of all the variables in the forward model and their sizes*

Let $X^{(1)}, X^{(2)} \in \mathbb{R}^N$ be sparse images which encode the location of the normal and over-expressed neuron centers respectively.We shall refer to them as location images. $X^{(1)}$ and $X^{(2)}$ represent the contribution of two neuron models to explain the data at each voxel location. $X^{(k)}$ is non-zero only at the center of neurons that are modeled by the shape model of $A^{(k)}$ and the non-zero value is proportional to the brightness of the neuron it models.

Therefore, $A^{(1)}X^{(1)} + A^{(2)}X^{(2)}$ models the neurons in the image Y. This is illustrated in figure 2.

Let the columns of the matrix $B \in \mathbb{R}^{N \times M}$ be the M basis vectors of the 3D Discrete Cosine Transform corresponding to the low-frequency components. Then given appropriate choice of the parameter vector $\theta \in \mathbb{R}^M$, $B\theta$ can be used to model the low-frequency background illumination variation in the image for an appropriate choice of $\theta \in \mathbb{R}^M$. We shall refer to $B\theta$ as the background offset and θ as the background offset coefficients. Using these, we model the image Y as

$$Y = \sum_{k=1}^{2} A^{(k)}X^{(k)} + B\theta + W_I + W_G, \qquad (1)$$

where W_I represents additive impulsive noise from dendrites, and W_G represents additive Gaussian noise. The variables in equation 1 and their sizes are illustrated in figure 3

We consider the Gaussian noise to be white so that the forward model (likelihood of the data, Y given unknown variables

$X^{(1)}$, $X^{(2)}$, W_I and θ) can be written as

$$p(Y|X^{(1)},X^{(2)},\theta,W_I) = \frac{1}{z_F} exp\Big(-\frac{1}{2\sigma^2_{W_G}}\|Y-\sum_{k=1}^{2}A^{(k)}X^{(k)}-B\theta-W_I\|_2^2\Big), \quad (2)$$

where z_F is the normalizing constant or the partition function of the forward model and $\sigma^2_{W_G}$ is the variance of the gaussian noise.

In the prior model formulation, we model the variables $X^{(1)}$, $X^{(2)}$, W_I as independent. Thus their joint prior probability $p(X^{(1)},X^{(2)},W_I)$ can be factored into the individual prior probabilities $p_1(X^{(1)})$, $p_2(X^{(2)})$ and $p_I(W_I)$ as:

$$p(X^{(1)},X^{(2)},W_I) = p_1(X^{(1)})p_2(X^{(2)})p_I(W_I) \quad (3)$$

Since, $X^{(1)}$, $X^{(2)}$ and W_I are sparse, we model their negative log prior to be proportional to their L1 norm. The L1 norm prior is known to induce sparsity in a convex fashion [28]. The individual prior probabilities are then given by:

$$p_k(X^{(k)}) = \frac{1}{z_k} exp\Big(-\frac{1}{\sigma_i}\|X^{(i)}\|_1\Big) \quad , k = 1,2 \quad (4)$$

and

$$p_I(W_I) = \frac{1}{z_{W_I}} exp\Big(-\frac{1}{\sigma_{W_I}}\|W_I\|_1\Big), \quad (5)$$

where z_1, z_2 , z_{W_I} are the corresponding partition functions and σ_1, σ_2 , σ_{W_I} are scale parameters.

Neuron Centers Estimation

Since the location of the neuron centers are encoded in the location images , $X^{(1)}$ and $X^{(2)}$, they need to be estimated first in order to calculate the neuron centers. The variables θ and W_I are not needed for calculating the neuron centers but they need to be estimated in order to estimate $X^{(1)}$ and $X^{(2)}$. For estimating the location images we use the maximum a posteriori probability (MAP) estimate. Since we also have the parameter vector θ , we find the joint MAP estimate of $X^{(1)}$,$X^{(2)}$, W_I and θ as:

$$\{X^{(1)^*},X^{(2)^*},\theta^*,W_I^*\} = \underset{X^{(1)},X^{(2)},\theta,W_I}{\text{argmin}} \Big\{-log(p(Y|X^{(1)},X^{(2)},\theta,W_I)) - log(p(X^{(1)},X^{(2)},W_I))\Big\}, \quad (6)$$

where $p(Y|X^{(1)},X^{(2)},\theta,W_I)$ is the likelihood of the image Y given $X^{(1)},X^{(2)},\theta,W_I$ and $p(X^{(1)},X^{(2)},W_I)$ is the prior probability of $X^{(1)}$, $X^{(2)}$ and W_I. Using equations 4 and 5 we can find the MAP estimate as:

$$\{X^{(1)^*},X^{(2)^*},\theta^*,W_I^*\} = \underset{X^{(1)},X^{(2)},\theta,W_I}{\text{argmin}} \Big\{\frac{1}{2\sigma^2_{W_G}}\|Y-\sum_{k=1}^{2}A^{(k)}X^{(k)}-B\theta-W_I\|_2^2 + \sum_{k=1}^{2}\frac{1}{\sigma_k}\|X^{(k)}\|_1 + \frac{1}{\sigma_{W_I}}\|W_I\|_1 \ . \quad (7)$$

```
function $[X^{(1)},X^{(2)},\theta,W_I] \leftarrow$ MAP_ESTIMATE($Y,A^{(1)},A^{(2)},B,$
$\sigma_1,\sigma_2,\sigma_{W_I},\sigma_{W_G}$)
    Initialize $X^{(1)} \leftarrow \mathbf{0}$
    Initialize $X^{(2)} \leftarrow \mathbf{0}$
    Initialize $W_{(I)} \leftarrow \mathbf{0}$
    Initialize $\theta \leftarrow B^T Y$
    Initialize $e \leftarrow Y - B\theta$
    while Stopping criterion is not met do
        for $i = 1$ to $N$ do
            for $k = 1$ to $2$ do
                $v \leftarrow X_i^{(k)}$
                $X_i^{(k)} \leftarrow Shrink\Big(v + \frac{e^T A^{(k)}_{*,i}}{\|A^{(k)}_{*,i}\|^2}, \frac{\sigma^2_{W_G}}{\sigma_k\|A^{(k)}_{*,i}\|^2}\Big)$
                $e \leftarrow e - A^{(k)}_{*,i}(X_i^{(k)} - v)$
            end for
            $v \leftarrow W_{I_i}$
            $W_{I_i} \leftarrow Shrink\Big(v + e_i, \frac{\sigma^2_{W_G}}{\sigma_k}\Big)$
            $e \leftarrow e - (W_{I_i} - v)$
        end for
        $\tilde{e} \leftarrow e + B\theta$
        $\theta \leftarrow B^T\tilde{e}$
        $e \leftarrow \tilde{e} - B\theta$
    end while
end function
```

Figure 4. *Pseudocode of MAP estimation*

```
function $\zeta \leftarrow$ MBND($Y,A^{(1)},A^{(2)},B,\sigma_1,\sigma_2,\sigma_{W_I},\sigma_{W_G},W$)
    Initialize $\zeta \leftarrow \{\}$
    $[X^{(1)},X^{(2)},\theta,W_I] \leftarrow$ MAP_ESTIMATE($Y,A^{(1)},A^{(2)},B,$
$\sigma_1,\sigma_2,\sigma_{W_I},\sigma_{W_G}$)
    for each $s \in S$ do
        for $k = 1$ to $2$ do
            if $X^{(k)}_{\Omega(s)} > 0$ and $X^{(k)}_{\Omega(s)} > X^{(k)}_{\Omega(r)} \forall r \in \delta s$ then
                $\zeta \leftarrow \zeta \cup \{s\}$
            end if
        end for
    end for
end function
```

Figure 5. *Pseudocode of MBND. Output, ζ is the set of all detected neuron centers. Input is the image Y, shape models $A^{(1)}$,$A^{(2)}$ and the parameters $B,\sigma_1,\sigma_2,\sigma_{W_I},\sigma_{W_G},W$. S is the set of all lattice points representing voxel locations. $\Omega()$ maps the lattice points to a linear index.*

The MAP estimate is computed by iteratively minimizing the cost function above. This is similar to the LASSO cost function [29]. We use the iterative co-ordinate descent method [30] to minimize the cost function in equation 7. We iteratively minimize the cost function in equation 7 until the change in forward model cost, $\frac{1}{2\sigma^2_{W_G}}\|Y-\sum_{k=1}^{2}A^{(k)}X^{(k)}-B\theta-W_I\|_2^2$, falls below a certain threshold ε. The pseudo-code for the minimization of the cost function in equation 7 is given in figure 4, where the shrinkage function $Shrink(x,\lambda)$ is defined as:

$$Shrink(x,\lambda) = sign(x)max(|x|-\lambda,0) \quad (8)$$

After the location images $X^{(1)}$ and $X^{(2)}$ have been estimated, we can find the neuron center locations by finding the location of its non-zero voxels. However, due to imperfect matching between our shape model and the shape of the neurons in the data, we might have consecutive non-zero voxel values in the estimated location images $X^{(1)}$ and $X^{(2)}$ leading to duplicate neuron centers.

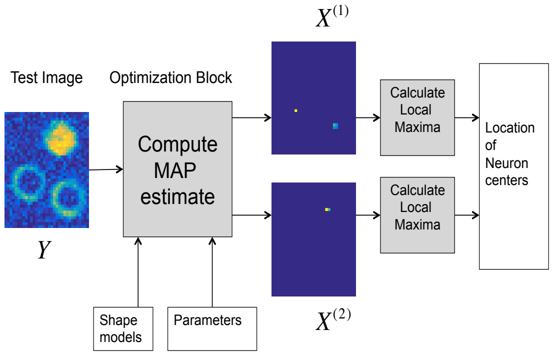


***Figure 6.** Overview of neuron center estimation in MBND*

To mitigate this, we compute the normal and over-expressed neuron centers as the local maxima of location images $X^{(1)}$ and $X^{(2)}$ respectively. In other words, we consider the location $s \in S$ to be a neuron center if $X^{(k)}_{\Omega(s)} > 0$ and $X^{(k)}_{\Omega(s)} > X^{(k)}_{\Omega(r)} \forall r \in \delta s$, $k = 1,2$, where the neighborhood δs of s is given by:

$$\delta s = \{r \in S \mid \|W^{-1}(s-r)\|_2 < 1, r \neq s\}, \tag{9}$$

where $W = diag(w_x, w_y, w_z)$ is a weighting matrix providing the size of the neighborhood.

The overview of the full process of neuron center estimation is shown in figure 6 and the pseudo-code is given in figure 5.

Shape Model Estimation

The impulse responses of $A^{(1)}$ and $A^{(2)}$ serve as shape models and are estimated from training data. The impulse responses of $A^{(1)}$ and $A^{(2)}$ are image patches that represent the shape of the typical neuron of normal and over-expressed type respectively as shown in the first images in figure 8(a)(b). Let the impulse response of $A^{(1)}$ and $A^{(2)}$ be $u^{(1)}$, $u^{(2)} \in \mathbb{R}^L$ in vectorized form. We extract neuron patches from the training volume to estimate $u^{(1)}$ and $u^{(2)}$. Let Z be the vectorized training volume. We manually select P_1 neuron centers of the normal type and P_2 neuron centers of the over-expressed type in the training volume and train the shape models using these. These neuron centers need not be exhaustive, i.e. there might be other neuron centers present in the training volume apart from these. To estimate the shape models that best fit the training volume, we minimize the minimum reconstruction error in equation 7 in the neuron region with respect to the shape models. To simplify the training, we find the background offset variable, θ by minimizing the reconstruction error with respect to θ to get $\theta = B^T Z$. We then compute the background offset subtracted training volume, $\widetilde{Z}$ as:

$$\widetilde{Z} = Z - BB^T Z \tag{10}$$

We then extract P_1 training patches of size L from $\widetilde{Z}$ centered around the P_1 neuron centers of the normal type and put them in the matrix $Y^{(1)} \in \mathbb{R}^{L\times P_1}$. Similarly, we extract P_2 training patches of size L from $\widetilde{Z}$ centered around the P_2 neuron centers of the over-expressed type and put them in the matrix $Y^{(2)} \in \mathbb{R}^{L\times P_2}$. The process of extracting training patches and estimating the shape models is illustrated in figure 7.

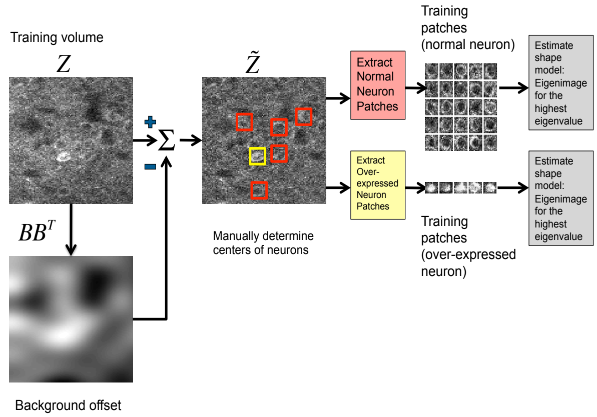

Training volume
Z
$\tilde{Z}$
+
−
Σ
BB^T
Background offset
Manually determine centers of neurons
Extract Normal Neuron Patches
Training patches (normal neuron)
Estimate shape model: Eigenimage for the highest eigenvalue
Extract Over-expressed Neuron Patches
Training patches (over-expressed neuron)
Estimate shape model: Eigenimage for the highest eigenvalue

***Figure 7.** Overview of the process of extracting training patches and estimating the shape models*

Then, to estimate $u^{(k)}$ for $k = 1,2$, we minimize the minimum reconstruction error with respect to $u^{(k)}$ as:

$$u^{(k)} = \underset{u\in\mathbb{R}^L}{\text{argmin}} \left\{ \sum_{i=1}^{P_k} \min_{\rho\in\mathbb{R}} \|Y^{(k)}_{*,i} - \rho u\|_2^2 \right\} \tag{11}$$

Define the sample correlation matrices $R^{(k)}_Y \in \mathbb{R}^{L\times L}$, for $k = 1,2$ as

$$R^{(k)}_Y = \frac{1}{P_k} Y^{(k)} (Y^{(k)})^T \ . \tag{12}$$

Solving the least squares problem in equation 11 gives

$$u^{(k)} = \underset{u\in\mathbb{R}^L}{\text{argmax}} \left(\frac{u^T R^{(k)}_Y u}{\|u\|_2^2} \right) . \tag{13}$$

Let the eigenvalues and eigenvectors of $R^{(k)}_Y$ be $\Lambda^{(k)}$ and $E^{(k)}$ for $k = 1,2$. So that

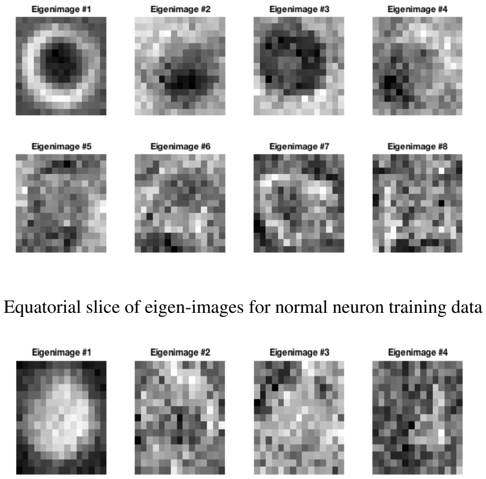
$$R^{(k)}_Y = E^{(k)} \Lambda^{(k)} (E^{(k)})^T \ . \tag{14}$$

The solution to equation 11 is well known: $u^{(k)}$ is the eigenvector of $R^{(k)}_Y$ corresponding to the highest eigenvalue. Since the solution to equation 13 is scale invariant, we set $\|u^{(k)}\|_2 = (\Lambda^{(k)}_{1,1})^{1/2}$ to capture the brightness scale of the neurons:

$$u^{(k)} = (\Lambda^{(k)}_{1,1})^{1/2} E^{(k)}_{*,1} \ . \tag{15}$$

Experimental Results

In this section we compare our Model Based Neuron Detection algorithm (MBND) with CellSegm [31]. CellSegm is a publicly available software for high throughput cell segmentation in fluorescence microscopy images. We use the segmct method within CellSegm toolbox which segments cells by iterative thresholding, hole filling and classification based on sizes of regions above threshold. We take the connected component centers of the final segmentation result to be the neuron centers.



(a) Equatorial slice of eigen-images for normal neuron training data

(b) Equatorial slice of eigen-images for over-expressed neuron training data

Figure 8. *The equatorial slices of the eigen-images for the (a) normal neuron data (b) neurons affected by GCaMP over-expression. Only the eigen-images corresponding to the highest eigenvalues are shown.*

Dataset Description

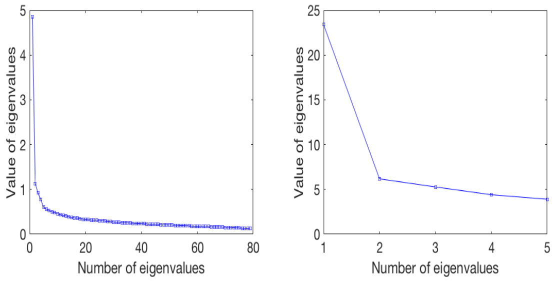
We use GCaMP6 labeled calcium imaging data of a mouse brain to evaluate our neuron detection algorithm. The data was provided by Prof. Meng Cui and Dr. Lingjie Kong at Purdue University. The full 3D data volume is of size 512x512x421 voxels along the x,y and z axes respectively. The inter-voxel distance is $1\mu m$. The neuron cells are approximately ellipsoids with diameter $10\mu m$ along each axis. Contrast stretching is done on the whole volume to improve the contrast and then the voxel values are normalized in the range 0-1. A section of the volume (after contrast stretching and normalization) is shown in figure 1 as an example.

We test our algorithm on a section of the full volume of size 104x101x21 that we will call test volume henceforth. We collect neuron patches for training from a separate section of size 351x201x51 from the whole volume. We will call this the training volume henceforth. We do not use the whole volume since collecting ground truth is time and labor intensive.

Training

We subtract the low-frequency background from the training volume and then extract 84 centered neuron patches from it, 79 of them normal neuron patches and 5 over-expressed ones. For background subtraction we construct B by using the first 6 , 5 , 11 basis vectors of the 1D Discrete cosine transforms along the x,y,z directions respectively so that we have $M = 6 \times 5 \times 11 = 330$ basis vectors of the 3D Discrete cosine transform in B. The patches extracted were of size 15x15x7. After extracting the training patches, the shape models $u^{(1)}$ and $u^{(2)}$ were estimated using solution to equation 15. The equatorial slices of the eigen-images for the normal and over-expressed neuron training data are shown in figure 8. We only show the eigen-images corresponding to the highest eigenvalues. The eigenvalues are shown in figure 9.

As a first order approximation, we use only the eigen-image of the highest eigen-value for the shape models.



(a) Eigenvalues of normal neuron training data

(b) Eigenvalues of over-expressed neuron training data

Figure 9. *Eigenvalues of the (a) normal neuron data (b) neurons affected by GCaMP over-expression*

Testing and Evaluation

We apply both the CellSegm method and MBND on the test volume. To determine the accuracy of the detected centers we rely on a ground truth: a list of actual neuron center co-ordinates. The ground truth was manually determined by looking at the 3D stack and was verified by an expert. We consider a detected neuron center to be a correct one if it differs from the true center by a certain tolerance value. Namely, if $c_1 = [x_1, y_1, z_1]^T$ is the location of a true neuron center, and $c_2 = [x_2, y_2, z_2]^T$ is the location of a detected neuron center, then c_2 is considered a true detection of c_1 if:

$$\|D^{-1}(c_1 - c_2)\|_2 < 1 \tag{16}$$

where, $D = diag(d_x, d_y, d_z)$ is a weighing matrix providing the tolerances along each of the x,y,z directions. The values of d_x, d_y, d_z should be small enough so that the ellipsoid with radii d_x, d_y, d_z is well inside the neuron and large enough that small deviations in center co-ordinates are still considered true detection. We take $d_x = 3$, $d_y = 3$, $d_z = 4$ since there are roughly half of the radii of the ellipsoidal neurons.

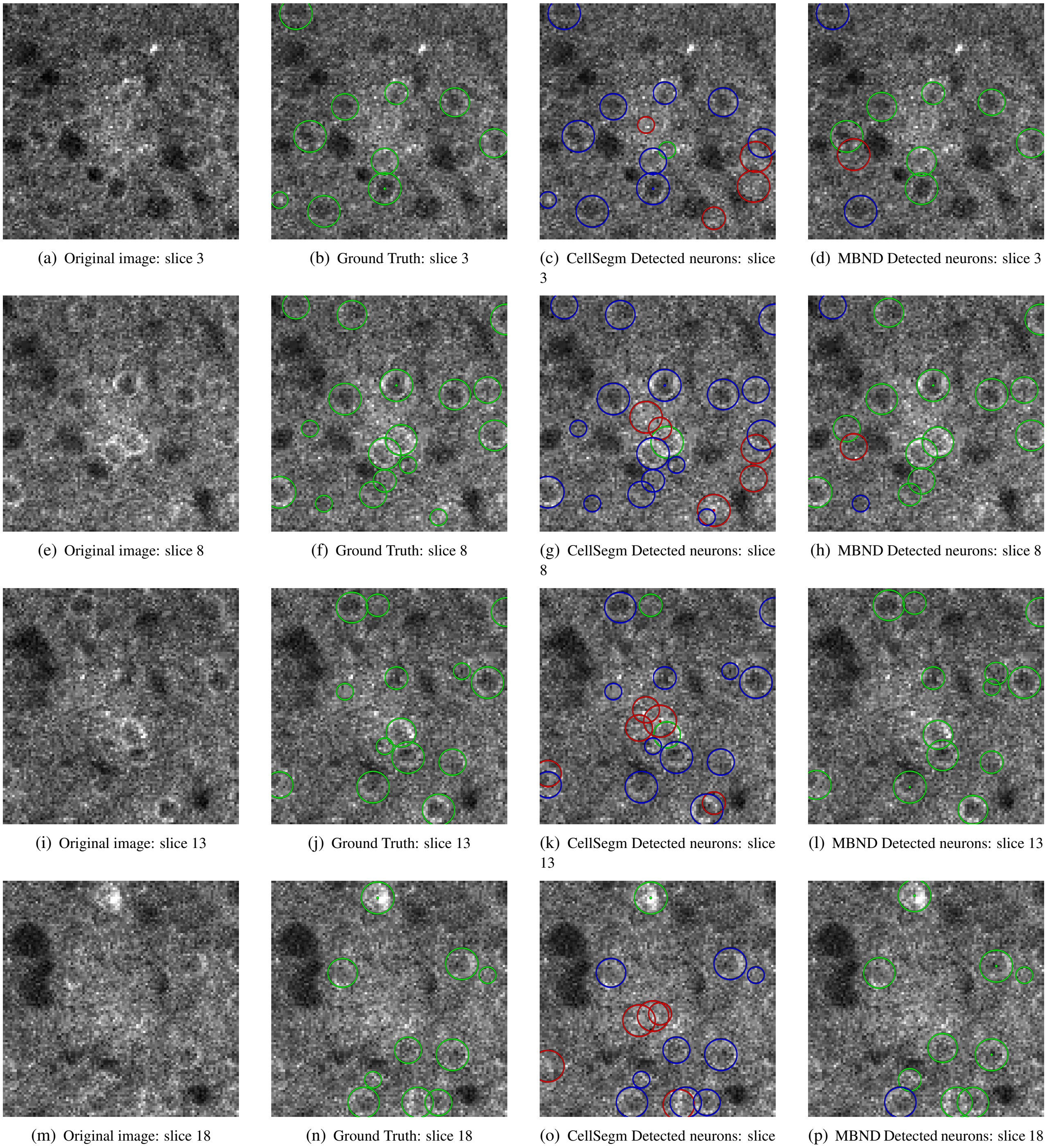
Each detected neuron center that is a true detection of some neuron in the ground truth is termed a true positive. If multiple neuron centers are detected within the tolerance limit of the actual neuron center, only one of them is considered a true positive and the rest are considered false positives. If a detected neuron center is not a true detection of any neurons in the ground truth then it is termed a false positive. All neuron centers in the ground truth that are not detected are termed false negatives. For a set of detected centers and ground truth, let the total number of true positives, false positives and false negatives be denoted as: N_{TP}, N_{FP} and N_{FN} respectively. We evaluate the list of detected centers by the precision and recall metric defined as:

$$precision = \frac{N_{TP}}{N_{TP} + N_{FP}} \tag{17}$$

$$recall = \frac{N_{TP}}{N_{TP} + N_{FN}} \tag{18}$$

Precision and recall values lie in the range $[0, 1]$ and high values for both are desirable. Combining both precision and recall, we define a metric F-score as:

$$F = \frac{2 \cdot precision \cdot recall}{precision + recall} \tag{19}$$



(a) Original image: slice 3

(b) Ground Truth: slice 3

(c) CellSegm Detected neurons: slice 3

(d) MBND Detected neurons: slice 3

(e) Original image: slice 8

(f) Ground Truth: slice 8

(g) CellSegm Detected neurons: slice 8

(h) MBND Detected neurons: slice 8

(i) Original image: slice 13

(j) Ground Truth: slice 13

(k) CellSegm Detected neurons: slice 13

(l) MBND Detected neurons: slice 13

(m) Original image: slice 18

(n) Ground Truth: slice 18

(o) CellSegm Detected neurons: slice 18

(p) MBND Detected neurons: slice 18

Figure 10. *Slice by slice comparison of detected neurons by our proposed method(MBND) with CellSegm Here (a), (e), (i), (m) show the original test image in slices 3,8,13,18 respectively. (b), (f), (j), (n) show the ground truth in slices 3,8,13,18 respectively. (c), (g), (k), (o) show the neurons detected by CellSegm in slices 3,8,13,18 respectively. (d), (h), (l), (p) show the neurons detected by our MBND in slices 3,8,13,18 respectively. To show the neurons, we draw a sphere around the neuron centers in 3D and we show the circular intersection of the surface of the sphere with the current slice in the figures. A dot in the center of the circle indicates that the neuron center is in the current slice. Ground truth and true detection are shown in green, false positives in red and missed detection in blue.*

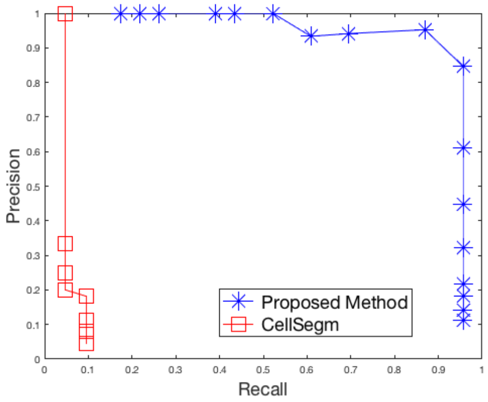


Figure 11. Precison recall plot of our proposed method(MBND) compared with CellSegm

For both MBND and CellSegm, we tune the parameters to get the best F-score on the test volume. We set the noise standard deviation $\sigma_{w_G} = 0.07$. For constructing B, use the first 6, 5, 11 basis vectors of the 1D Discrete cosine transforms along the x,y,z directions respectively so that we have $M = 6 \times 5 \times 11 = 330$ basis vectors of the 3D Discrete cosine transform in B. We set $w_x = w_y = w_z = 4$ for calculating local maxima of the location images. We set the parameters σ_{W_I}, σ_1, σ_2 so that a neuron of typical intensity and of the trained shape can be detected. Let there be a neuron $cu^{(k)}$, $(k = 1,2)$ embedded in the test image, where c is a scalar constant and $u^{(k)}$ is estimated from equation 15. Then to detect it we need to have a non-zero solution to the equation:

$$\rho^* = \underset{\rho \in \mathbb{R}}{\operatorname{argmin}} \frac{1}{2\sigma_{w_G}^2} \|cu^{(k)} - \rho u^{(k)}\|_2^2 + \frac{1}{\sigma_k} |\rho| \qquad (20)$$

It can be shown that a non-zero solution corresponds to $\sigma_k \geq c\sigma_{w_G}^2 / \|u^{(k)}\|$. This is the motivation behind setting $\sigma_k = c_k \sigma_{w_G}^2 / \|u^{(k)}\|$, $(k = 1,2)$ and $\sigma_{W_I} = c_I \sigma_{w_G}^2$, where c_1, c_2 and c_{W_I} are scalar constants. We set $c_1 = 3.3$, $c_2 = 2.7$ and $c_{W_I} = 2.5$.

For MBND, the best F-score obtained was 0.91 with corresponding $precision = 0.95$ and $recall = 0.87$. For CellSegm, the best F-score obtained was 0.13 with corresponding $precision = 0.18$ and $recall = 0.10$. The neuron detection results for MBND and CellSegm corresponding to their best F-score is displayed in figure 10. We also display the original image and ground truth to get a visual comparison. To show the neurons, we draw a sphere around the neuron centers in 3D and we show the circular intersection of the surface of the sphere with the current slice in the figures. A dot in the center of the circle indicates that the neuron center is in the current slice. Ground truth and true detection are shown in green, false positives in red and missed detection in blue.

Next, for MBND we vary the neuron regularizer σ_1 in equation 7 and keep the other parameters fixed to get a series of neuron detection results and the corresponding precision and recall values. For the CellSegm method, we vary the threshold and keeping the other parameters fixed to get a series of neuron detection results and the corresponding precision and recall values. We plot these precision recall values to generate a precision-recall plot. The precision recall plot of MBND is compared to that of CellSegm in figure 11.

From figures 10 and 11 we can see that MBND has minimal false positives and is able to detect many more neurons than CellSegm. From figures 10(o) and 10(g) we can see that CellSegm is able to detect only the very bright neurons. This is because iterative thresholding combined with other image-processing techniques are not good if there isn't a strong difference in illumination level between the object and the background. On the other hand, in MBND we train shape models for the neurons and thus MBND is able to detect neurons even when there isn't much illumination difference between the neuron and the background.

Conclusions and Future Work

In this paper we presented a novel model based neuron detection scheme(MBND) to detect neuron centers in fluorescence images of neurons. MBND is able to detect neurons at very low signal to noise ratios. Results show that MBND performs better than one widely used method.

ACKOWLEDGEMENTS

The authors acknowledge support from the National Science Foundation (Grant # 1318894). The authors also thank Prof. Meng Cui and Dr. Lingjie Kong, Purdue University for providing the GCaMP6 labeled Calcium imaging data used for evaluating our neuron detection algorithm.

References

[1] Lin Tian, S Andrew Hires, Tianyi Mao, Daniel Huber, M Eugenia Chiappe, Sreekanth H Chalasani, Leopoldo Petreanu, Jasper Akerboom, Sean A McKinney, Eric R Schreiter, et al. Imaging neural activity in worms, flies and mice with improved gcamp calcium indicators. *Nature methods*, 6(12):875–881, 2009.

[2] Jasper Akerboom, Nicole Carreras Calderón, Lin Tian, Sebastian Wabnig, Matthias Prigge, Johan Tolö, Andrew Gordus, Michael B Orger, Kristen E Severi, John J Macklin, et al. Genetically encoded calcium indicators for multi-color neural activity imaging and combination with optogenetics. *Frontiers in molecular neuroscience*, 6:2, 2013.

[3] Jasper Akerboom, Tsai-Wen Chen, Trevor J Wardill, Lin Tian, Jonathan S Marvin, Sevinç Mutlu, Nicole Carreras Calderón, Federico Esposti, Bart G Borghuis, Xiaonan Richard Sun, et al. Optimization of a gcamp calcium indicator for neural activity imaging. *The Journal of neuroscience*, 32(40):13819–13840, 2012.

[4] Tsai-Wen Chen, Trevor J Wardill, Yi Sun, Stefan R Pulver, Sabine L Renninger, Amy Baohan, Eric R Schreiter, Rex A Kerr, Michael B Orger, Vivek Jayaraman, et al. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature*, 499(7458):295–300, 2013.

[5] Alex C Kwan. What can population calcium imaging tell us about neural circuits? *Journal of neurophysiology*, 100(6):2977–2980, 2008.

[6] Michiel A Martens, Werend Boesmans, and Pieter Vanden Berghe. Calcium imaging at khz frame rates resolves millisecond timing in neuronal circuits and varicosities. *Biomedical optics express*, 5(8):2648–2661, 2014.

[7] Dirk Padfield, Jens Rittscher, and Badrinath Roysam. Spatio-temporal cell segmentation and tracking for automated screening. In *2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 376–379. IEEE, 2008.

[8] Xiaowei Chen, Xiaobo Zhou, and Stephen TC Wong. Automated segmentation, classification, and tracking of cancer cell nuclei in time-lapse microscopy. *IEEE Transactions on Biomedical Engineering*, 53(4):762–766, 2006.

[9] Zia Khan, Tucker Balch, and Frank Dellaert. Mcmc-based particle filtering for tracking a variable number of interacting targets. *IEEE transactions on pattern analysis and machine intelligence*, 27(11):1805–1819, 2005.

[10] Kevin Smith, Daniel Gatica-Perez, and J-M Odobez. Using particles to track varying numbers of interacting people. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 962–969. IEEE, 2005.

[11] Dipti Prasad Mukherjee, Nilanjan Ray, and Scott T Acton. Level set analysis for leukocyte detection and tracking. *IEEE Transactions on Image processing*, 13(4):562–572, 2004.

[12] Alexandre Dufour, Vasily Shinin, Shahragim Tajbakhsh, Nancy Guillén-Aghion, J-C Olivo-Marin, and Christophe Zimmer. Segmenting and tracking fluorescent cells in dynamic 3-d microscopy with coupled active surfaces. *IEEE Transactions on Image Processing*, 14(9):1396–1410, 2005.

[13] Dirk Padfield, Jens Rittscher, Nick Thomas, and Badrinath Roysam. Spatio-temporal cell cycle phase analysis using level sets and fast marching methods. *Medical image analysis*, 13(1):143–155, 2009.

[14] Dirk Dormann, Thorsten Libotte, Cornelis J Weijer, and Till Bretschneider. Simultaneous quantification of cell motility and protein-membrane-association using active contours. *Cell motility and the cytoskeleton*, 52(4):221–230, 2002.

[15] Oleh Dzyubachyk, Wiggert A van Cappellen, Jeroen Essers, Wiro J Niessen, and Erik Meijering. Advanced level-set-based cell tracking in time-lapse fluorescence microscopy. *IEEE transactions on medical imaging*, 29(3):852–867, 2010.

[16] Hailin Shen, Glyn Nelson, Stephnie Kennedy, David Nelson, James Johnson, David Spiller, Michael RH White, and Douglas B Kell. Automatic tracking of biological cells and compartments using particle filters and active contours. *Chemometrics and Intelligent Laboratory Systems*, 82(1):276–282, 2006.

[17] Michael Kass, Andrew Witkin, and Demetri Terzopoulos. Snakes: Active contour models. *International journal of computer vision*, 1(4):321–331, 1988.

[18] Ravi Malladi, James A Sethian, and Baba C Vemuri. Shape modeling with front propagation: A level set approach. *IEEE transactions on pattern analysis and machine intelligence*, 17(2):158–175, 1995.

[19] Vicent Caselles, Ron Kimmel, and Guillermo Sapiro. Geodesic active contours. *International journal of computer vision*, 22(1):61–79, 1997.

[20] Nikos Paragios and Rachid Deriche. Coupled geodesic active regions for image segmentation: A level set approach. In *European Conference on Computer Vision*, pages 224–240. Springer, 2000.

[21] Tony F Chan and Luminita A Vese. Active contours without edges. *IEEE Transactions on image processing*, 10(2):266–277, 2001.

[22] Daniel Cremers, Florian Tischhäuser, Joachim Weickert, and Christoph Schnörr. Diffusion snakes: Introducing statistical shape knowledge into the mumford-shah functional. *International journal of computer vision*, 50(3):295–313, 2002.

[23] Daniel Cremers, Stanley J Osher, and Stefano Soatto. Kernel density estimation and intrinsic alignment for shape priors in level set segmentation. *International Journal of Computer Vision*, 69(3):335–351, 2006.

[24] Shantanu H Joshi and Anuj Srivastava. Intrinsic bayesian active contours for extraction of object boundaries in images. *International journal of computer vision*, 81(3):331–355, 2009.

[25] Michael E Leventon, W Eric L Grimson, and Olivier Faugeras. Statistical shape influence in geodesic active contours. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 316–323. IEEE, 2000.

[26] Maria Kulikova, Ian Jermyn, Xavier Descombes, Josiane Zerubia, and Elena Zhizhina. A marked point process model with strong prior shape information for the extraction of multiple, arbitrarily-shaped objects. In *Signal-Image Technology & Internet-Based Systems (SITIS), 2009 Fifth International Conference on*, pages 180–186. IEEE, 2009.

[27] Maria Kulikova, Antoine Veillard, Ludovic Roux, and Daniel Racoceanu. Nuclei extraction from histopathological images using a marked point process approach. In *SPIE Medical Imaging*, pages 831428–831428. International Society for Optics and Photonics, 2012.

[28] Michael Elad. Sparse and redundant representation. 2010.

[29] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

[30] Charles A Bouman and Ken Sauer. A unified approach to statistical tomography using coordinate descent optimization. *IEEE Transactions on image processing*, 5(3):480–492, 1996.

[31] Erlend Hodneland, Tanja Kögel, Dominik Michael Frei, Hans-Hermann Gerdes, and Arvid Lundervold. Cellsegm-a matlab toolbox for high-throughput 3d cell segmentation. *Source code for biology and medicine*, 8(1):1, 2013.

Author Biography

Soumendu Majee is from Kolkata, India. He received his B.Tech in Electronics and Electrical Communication Engineering from the Indian Institute of Technology, Kharagpur in 2014. He is currently pursuing his PhD at Purdue University. His research advisors are Prof. Charles Bouman and Prof. Gregery Buzzard. His research interests include signal and image processing, pattern recognition and communication.