Segmentation and Tracking of 3D Neuron Microscopy Images Using a PDE Based Method and Connected Component Labeling algorithm

Yalin Wang[1] Ilhwan Jo[1] Stephen Wong[2] Shing-Tung Yau[3] Tony F. Chan[1]

[1]Department of Mathematics
University of California, Los Angeles
{ylwang,ilhwanjo,chan}@math.ucla.edu

[2]Center for Bioinformatics
Harvard Medical School
stephen_wong@hms.harvard.edu

[3]Department of Mathematics
Harvard University
yau@math.harvard.edu

***Abstract*— In this paper we introduce our preliminary research results for segmentation and labeling of 3-dimensional microscopy neuron image. We segment each of stacked 2-dimensional image slices using a Partial Differential Equation (PDE) based algorithm and project previous slice segmentation result to the next slide as an initialization condition. Then we label neurons using an efficient connected component labeling algorithm. We show sample results obtained from real neuron image data.**

I. INTRODUCTION

To study morphology of neurons in three dimensions can help neuroscientists understand neuronal development [1]. Three dimensional image stacks of neurons such as motor neurons can be obtained by confocal microscopy [2]. In a recent paper [3], Cai et. al. use slice-wise segmentation of 3D microscopy image stacks. Their work is based on GVF snake model for segmentation of 2D images. In this paper we propose a segmentation-labeling model for 3D neuron image based on tracking with Chan-Vese algorithm and connected component labeling algorithm. Some preliminary results are presented.

II. TRACKING WITH CHAN-VESE ALGORITHM AND CONNECTED COMPONENTS LABELING

A. Chan-Vese Segmentation Algorithm

The Chan-Vese algorithm is a region-based segmentation model which is based on the active contour model, the Mumford-Shah functional and the Osher-Sethian level set method [4]. Given a grayscale image $I : \Omega \subset R^p \rightarrow R_+$ $(p = 2, 3)$, the Chan-Vese algorithm finds a curve C that represents a partition of Ω into two regions Ω_{in} and Ω_{out} so that they give an optimal piecewise constant approximation of the image. The contour C minimizes the following energy

$$E(C, c_1, c_2) = \lambda_1 \int_{\Omega_{in}} (I(x) - c_1)^2 dx$$
$$+\lambda_2 \int_{\Omega_{out}} (I(x) - c_2)^2 dx + \mu \text{length}(C)$$

where c_1, c_2 are the average intensities in Ω_{in} and Ω_{out} respectively and λ_i, μ are parameters.
This model can detect objects with edges that are not necessarily defined by gradient or with smooth boundaries. Its initial contour can be placed anywhere in the image and topological change of contour is allowed. It is also robust to noise.

B. Tracking objects with the CV Algorithm

Moelich and Chan proposed an algorithm for tracking objects in video sequences [5]. The algorithm can be described as a sequential segmentation in which the final segmentation contour of a frame is used as the initial contour for the segmentation of the next frame. In order for the tracking with CV algorithm to work properly, it is required that initial contour be in contact with the object to be detected [5]. Moelich et. al. added a modification based on target intensities to overcome this problem. In our work we only employ the following algorithm.

```
C_0 = initial contour

for I = 1 to N
{
        C_k = Chan-Vese(C_{k-1}, I_k)
        draw contour on image
        output frame
}
```

Fig. 1. The sequential tracking algorithm.

C. Connected Components Algorithm

Once a binary image is obtained by segmentation, pixels in the image can be grouped based on maximal connectivity by applying a connected component operator [6]. We extended the Lumia, Shapiro and Zuniga algorithm to 3D volume image. With defined "neighbors" for each pixel, the original algorithm can label all the connected components in a 2D image by finding all the equivalent classes in two passes. We extend this algorithm into 3D volume data. Specifically, we label all the connected components in each 2D slice. With 3D neighborhood definition, we find all equivalent classes of the foreground pixels in two passes along the z direction (here we assume each slice has x and y directions). The algorithm is very efficient to label 3D neuron data.

1-4244-0278-6/06/$20.00 ©2006 IEEE

D. Neuron Segmentation and Labeling

We segment the first 2D image slice using the Chan-Vese algorithm, apply tracking with CV algorithm to the image stack. We use the segmentation result as an initial contour for the next 2D image and sequentially apply this process to the whole stacked 2D images. In practice, we can use a manual segmentation by medical experts for the first 2D image and we can also start with any 2D image in the stack. Once all the 2D images of the stack is segmented, we apply connected component operator to the stack to obtain labeling of objects.

III. SAMPLE RESULTS

We tested our method on a 3D real neuron image data. The tested image is of size $512 \times 42 \times 256$ and contains seven neurons. The results are shown in Figure 2 and 3. As shown in Figure 2, With previous segmentation result (Figure 2 (a)) as the initial condition for the segmentation in the next slide. Figure 2 shows the sequence of the segmentation on the next slide. Figure 2 demonstrates that our algorithm can segment the 3D neuron image volume with Chan-Vese model.

Figure 3 shows 3D rendering segmentation results and the ground truth data. Figure 3 (a) show three different views of tracked neurons with original image intensity values. (b) is the ground truth data of the given image. Compare (a) and (b), we can see most of neurons are correctly segmented and labeled.

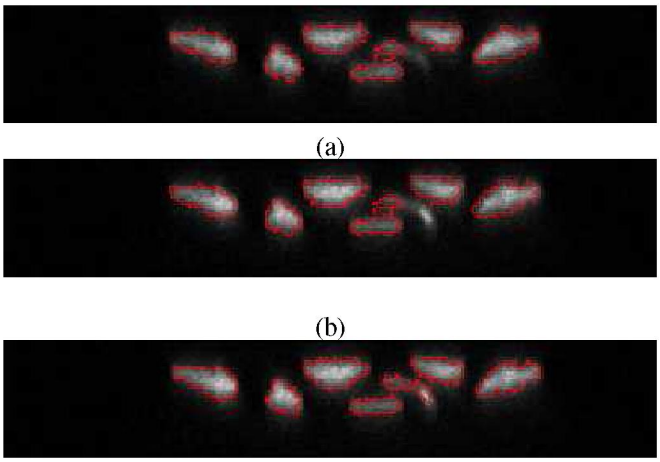

Fig. 2. Final segmentation contour (a) of a 2D image slice is used as the initial contour for the next, (b). (c) shows the segmentation of the next 2D image.

IV. CONCLUSIONS AND FUTURE WORK

One drawback of the tracking with CV algorithm is the inability to distinguish between objects with similar intensities that are close to each other [5]. Neurons in microscopy data have similar intensities and if two neurons are close enough to each other or the boundary between them is weak, contours may merge with each other. Our future work will focus on new algorithm development to distinguish close neurons.

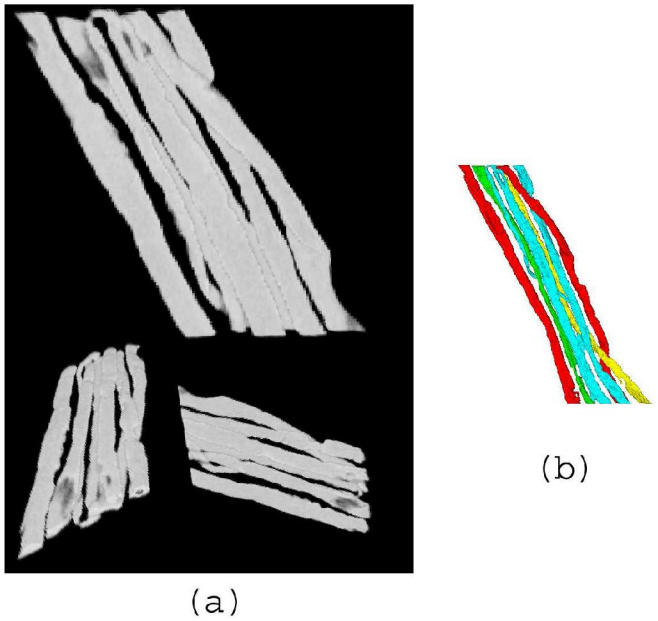

Fig. 3. 3D rendering of tracked neuron data compared with ground truth data. (a) 3D rendering of tracked neuron data (in three different view directions with original intensity values); (b) ground truth data of the given neuron image in which 7 neurons are labeled in different colors.

REFERENCES

[1] N. Kasthuri and J. W. Lichtman, "The role of neuronal identify in synaptic competition", *Nature*, vol. 424, no. 6974, pp. 426-430, 2003.

[2] G. Feng, R. H. Mellor, M. Bernstein, C. Keller-Peck, Q. T. Nguyen, M. Wallace, J. M. Nerbonne, J. W. Lichtman and J. R. Sanes, "Imaging neuronal subsets in transgenic mice expressing multiple spectral variants of GFP", *Neuron*, vol. 28, no. 1, pp. 41-51, Oct. 2000

[3] H. Cai, X. Xu, J. Lu, J. Lichtman, S.P. Yung, and S.T.C Wong, "Shape-constrained repulsive snake method to segment and track neurons in 3D microscopy images", *ISBI 2006*, pp. 538-541.

[4] T. Chan and L. Vese, "An active contour model without edges, "*Int. Conf. Scale-Space Theories in Computer Vision*, 16(2):266-277, 1999.

[5] M. Moelich and T. Chan, "Tracking Objects with the Chan-Vese Algorithm, " *UCLA CAM Report 03-14*.

[6] R. M. Haralick and L. G. Shapiro, "Computer and Robot Vision Volume I, " *Addison Wesley*, vol 1, 1992.