



Segmentation of center brains and optic lobes in 3D confocal images of adult fruit fly brains

Shing Chun Benny Lam, Zongcai Ruan, Ting Zhao, Fuhui Long, Arnim Jenett, Julie Simpson, Eugene W. Myers, Hanchuan Peng*

Janelia Farm Research Campus, Howard Hughes Medical Institute, 19700 Helix Drive, Ashburn, VA 20147, USA

ARTICLE INFO

Article history:

Accepted 13 August 2009

Available online 19 August 2009

Keywords:

Segmentation
Deformable model
Optic lobe
Center brain
3D confocal image
Fruit fly
Drosophila melanogaster
Shortest path
Gradient descent
Energy function optimization

ABSTRACT

Automatic alignment (registration) of 3D images of adult fruit fly brains is often influenced by the significant displacement of the relative locations of the two optic lobes (OLs) and the center brain (CB). In one of our ongoing efforts to produce a better image alignment pipeline of adult fruit fly brains, we consider separating CB and OLs and align them independently. This paper reports our automatic method to segregate CB and OLs, in particular under conditions where the signal to noise ratio (SNR) is low, the variation of the image intensity is big, and the relative displacement of OLs and CB is substantial.

We design an algorithm to find a minimum-cost 3D surface in a 3D image stack to best separate an OL (of one side, either left or right) from CB. This surface is defined as an aggregation of the respective minimum-cost curves detected in each individual 2D image slice. Each curve is defined by a list of control points that best segregate OL and CB. To obtain the locations of these control points, we derive an energy function that includes an image energy term defined by local pixel intensities and two internal energy terms that constrain the curve's smoothness and length. Gradient descent method is used to optimize this energy function. To improve both the speed and robustness of the method, for each stack, the locations of optimized control points in a slice are taken as the initialization prior for the next slice. We have tested this approach on simulated and real 3D fly brain image stacks and demonstrated that this method can reasonably segregate OLs from CBs despite the aforementioned difficulties.

© 2009 Elsevier Inc. All rights reserved.

1. Introduction

The GAL4-enhancer trap technique, which can block or enhance selected populations of neurons in a nervous system, is very powerful to study the neuronal circuits in a fruit fly (*Drosophila melanogaster*) brain [1–4]. It has become increasingly common to use confocal and multi-photon laser scanning microscopy to directly acquire three-dimensional (3D) image stacks of the entire brain of a fly of a GAL4 line, and thus to study its structures and functions [5]. Automated alignment (registration) of 3D images of fruit fly brains is a critical technique for high-throughput analysis of neuronal patterns [6]. However, as shown in Fig. 1, the two optic lobes (OLs) are loosely connected with the center region of the fly brain via the neuronal bundles. This loose articulation of the two OLs and the center brain (CB) significantly changes the local context at the boundary between OLs and CB from brain to brain. Such variations present difficulties to our automated registration technique. This paper reports an automatic method to segregate CB and OLs, especially for the situation where the signal to noise ratio (SNR) is low,

the variation of the image intensity is big, and the relative displacement of OLs and CB is substantial.

Previous neuro-anatomical studies have relied on manual or semi-automated methods to identify and delineate brain structures [7–9]. However, these manual methods are tedious and error prone. There are many existing studies on segmentation of brain regions from 3D Magnetic Resonance (MR), Computed Tomography (CT) images and microscopy. Xu et al. [10], Pham et al. [11], Shapiro and Stockman [12], and Kolmogorov and Rother [13] provide comprehensive reviews on segmentation using different methods. There are also a few reports on segmenting brain structures for display from microscopy images [14,15]. Nevertheless, these methods do not fit our purpose. For example, one of the most prominent level set based techniques from Chan and Vese [16] could only work up to a certain extent. One of the best results we obtained using Chan–Vese method is through initialization for which we placed circles of 30 pixels radii one by one next to each other over the whole image as initialization (Fig. 2a) and setting the smoothness parameter as 0.1 on the slice shown in Fig. 1. The respective Chan–Vese segmentation result is shown in Fig. 2b. While the Chan–Vese method could generate reasonable segmentation result for OL for this image, it can be noticed that the

* Corresponding author. Fax: +1 571 209 4933.

E-mail address: pengh@janelia.hhmi.org (H. Peng).

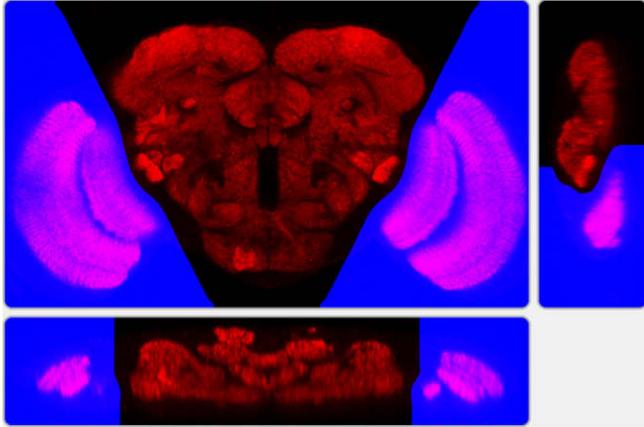


Fig. 1. Tri-view of a fruit fly brain and the 3D segmentation surface detected using our deformable model. Painted in blue regions are the segmented optic lobes. The remaining is the center brain. Top-left panel: frontal plane; top-right: sagittal plane; bottom-left: horizontal plane.

segmentation of CB is broken into several separate regions (over-segmentation). The problem still exists after morphological hole-filling procedure. Thus further image region grouping algorithms have to be developed to merge the broken regions. Differently, the work presented in this paper gives the first attempt to successfully automate the segmentation of articulated structures in an insect's brain image.

2. Image data

In this study we consider fruit fly GAL4 lines produced in the labs of Julie Simpson and Gerald Rubin. The synaptic neuropiles

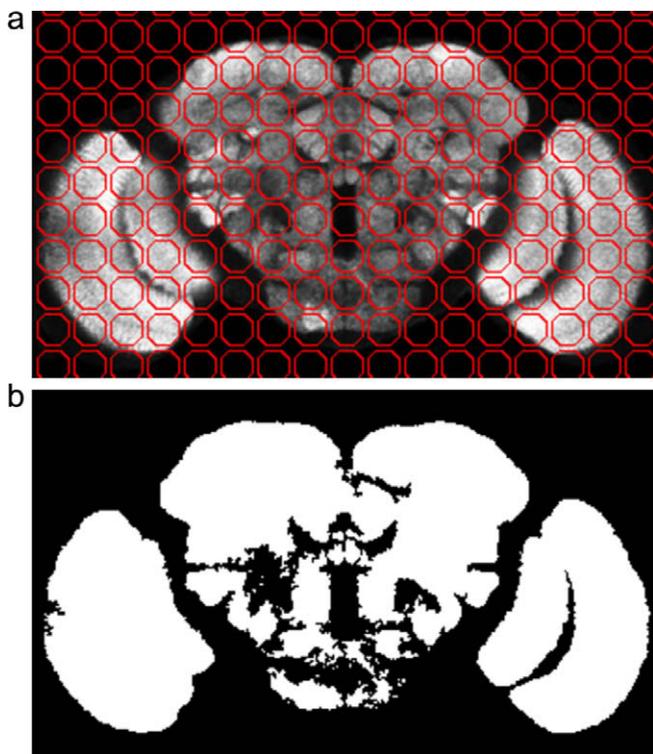


Fig. 2. (a) Initialization of Chan-Vese level set algorithm on the slice shown in Fig. 1. Circles of 30 pixels radii are placed one by one next to each other over the whole image as initialization. Smoothness parameter is 0.1. (b) Segmentation result on the slice shown in Fig. 1 using Chan-Vese level set algorithm after 500 iterations followed by morphological hole-filling procedure. The center brain is segmented as several sub-regions.

in the brain tissue are stained using monoclonal antibody nc82 [17], which in turn visualizes the entire shape of a brain. This nc82 neuropile stain is used as the reference channel in our Janelia 3D Brain Aligner (JBA) pipeline (Peng, et al., unpublished work). Since in this study we do not use other fluorescent channels in these data, the irrelevant specimen preparation details are skipped.

Whole-mount fly brains are scanned with a Zeiss 510 confocal laser scanning microscope with a Plan Apochromat[®] 20× NA 0.75 lens. Frontal series of entire brains are taken, with a typical optical voxel resolution $0.6 \times 0.6 \times 1 \mu\text{m}^3$. With these settings, image stacks of 200 ± 20 slices are obtained from complete scanning of whole-mount brains. Fluorochrome excitation of the nc82 stain is set to 568 nm; the respective emitted fluorescence signal is detected in the range of 607–701 nm.

All the images were first globally aligned to a “standard” target brain to have an “up-right” orientation (similar to Fig. 1) and a “standard” size of $1019 \times 601 \times 216$ voxels, using the automatic global affine alignment module in JBA. This global alignment step facilitates the following segmentation procedure to be fully automated.

3. Methods

3.1. Overview

In our method, segmentation is operated based on the analysis of the background (dark regions in-between OL and CB), because this region is usually homogenous throughout all slices. The objective is to find a minimum-cost 3D surface in this region that best separates an OL (either left or right) and CB. In the following we consider two methods, namely deformable model method and shortest path method.

Note in Fig. 1 that the separating surface/curve is optimized to go to the darkest image region but not to the bright part. To avoid possible confusion of the formulation below, we define a concept of “negative image”, which is to replace an 8-bit image voxel intensity value I using $255-I$. Thus for the negative image, the separating surface/curve goes to the bright image region.

3.2. Deformable model method

Given an input image $f(x, y, z)$, the segmentation problem is formulated as finding a deformable cutting surface defined in a 3D mesh, which is defined by a number of control points in the 3D coordinate system. Since the brain structures change gradually from one slice to another, the optimized control point locations in the present slice are taken as the prior for the next for initialization. This sequential operation reduces the optimization over a surface to a curve. The K control points of the deformable curve in each z -slice are defined as $c_k(x_k, y_k)$, $k = 1, 2, \dots, K$.

Let's consider separating one OL from CB at one time. For each slice, the segmenting curve is a collection of control points having the minimum energy defined by the energy function below:

$$E = \alpha E_{\text{image}} + \beta E_{\text{length}} + \gamma E_{\text{smoothness}}, \quad (1)$$

where E_{image} is the external energy term, and E_{length} and $E_{\text{smoothness}}$ are the two internal energy terms. The coefficients α , β , and γ are weighting parameters which control the relative contributions of the various energy terms to the final curve. E_{image} drives the control points to the region where the intensity is the minimum (i.e. maximum in the “negative image”), E_{length} constrains that the deformable curve is least bended, and $E_{\text{smoothness}}$ constrains the deformable curve to be smooth. The deformable curve evolves from the initialization curve devised from the previous slice. Fig. 3 illustrates the algorithm. Since a segmentation curve is obtained for

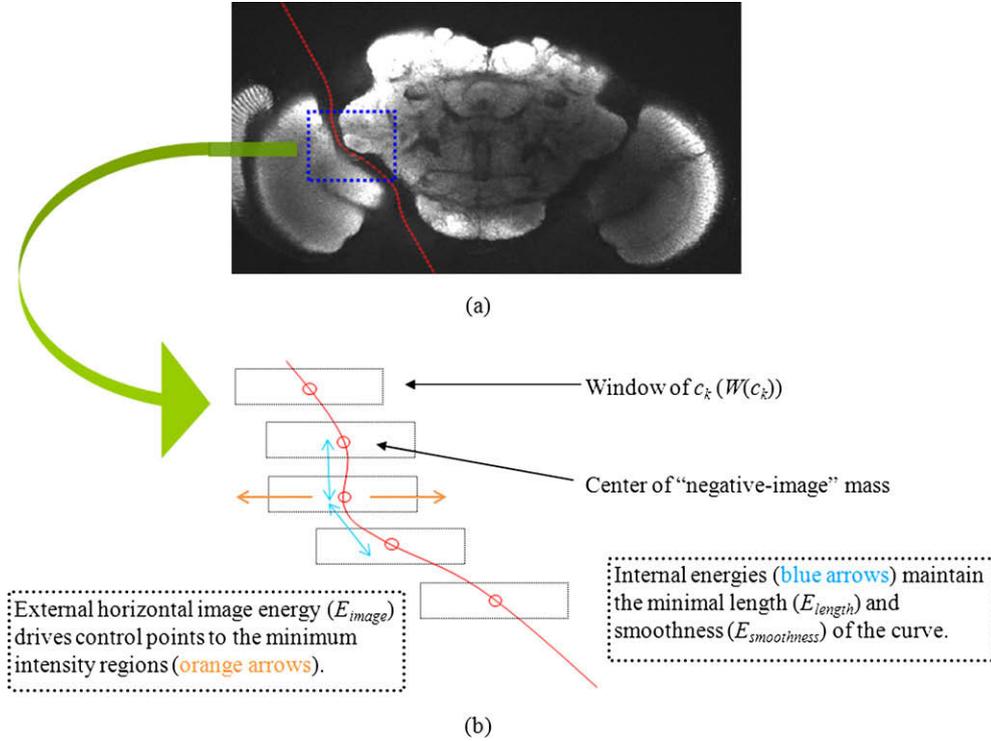


Fig. 3. Illustration of deformable model algorithm. The control points c_k (red circles) inside the blue box in (a) are magnified and shown in (b) as pointed by the green arrow. For each control point, the respective image energy and internal energy terms are defined (see main text for details).

each slice and the z resolution is high (e.g. $0.6 \mu\text{m}$), aggregation of these curves forms a smooth 3D segmentation surface. Note that this deformable model is modified from our earlier deformable curve model defined for straightening elongated and curved image objects [18]. The differences are explained in the following.

First, E_{image} is computed as

$$E_{image} = \sum_{k=1}^K \|c_k - m_k\|^2, \quad (2)$$

where $m_k = (m_{kx}, m_{ky})$ is center of mass of the squared pixel intensity in the window, calculated as $m_k = \sum_{i \in W(c_k)} f_i^2 r_i / \sum_{i \in W(c_k)} f_i^2$. f_i is the image intensity at pixel (x_i, y_i) and $r_i = (r_{ix}, r_{iy})$ is the offset between (x_i, y_i) and the origin of the coordinate system. A rectangular window $W(c_k)$ of size 41×21 (width \times height) in each slice is set for each c_k to calculate m_k . Different from [18], which considers Delaunay triangulation to define a local window around each control point, here we always consider a local rectangular window, and also constrain the control points to move horizontally.

Since we are considering a curve in 2D, the length constraint is formulated such that it favors small lengths between adjacent control points. Therefore, an energy term E_{length} is defined in Eq. (3) below, which is the sum of the squared distances between all consecutive pairs of control points.

$$E_{length} = \sum_{k=1}^{K-1} \|c_k - c_{k+1}\|^2. \quad (3)$$

Regarding the smoothness constraint, the energy term $E_{smoothness}$ is defined as

$$E_{smoothness} = \sum_{k=2}^{K-1} \|c_k - \frac{1}{2}(c_{k-1} + c_{k+1})\|^2, \quad (4)$$

which favors evenly spaced control points as $E_{smoothness}$ is 0 when $c_k = (c_{k-1} + c_{k+1})/2$ for every k .

Since the intensity profile inside W changes during every iteration of the minimization process, it is difficult to formulate the minimization as in classic snake/active contour [19,20]. Instead, we adopted the formulation described in [18] where, if E is minimal, the respective derivatives at the locations of control points must be 0:

$$\frac{\partial E}{\partial c_k} = \alpha \frac{\partial E_{image}}{\partial c_k} + \beta \frac{\partial E_{length}}{\partial c_k} + \gamma \frac{\partial E_{smoothness}}{\partial c_k} = 0. \quad (5)$$

Substituting the energy terms defined in Eqs. (2)–(4) into Eq. (5), it gives for any $k \in \{3, \dots, K-2\}$,

$$0 = \alpha(c_k - m_k) + \beta(2c_k - c_{k-1} - c_{k+1}) + \gamma(2c_k - \frac{3}{2}c_{k-1} - \frac{3}{2}c_{k+1} + \frac{1}{2}c_{k+2} + \frac{1}{2}c_{k-2}). \quad (6)$$

Equation (6) suggests one way to minimize the objective function in Eq. (1) is to iteratively adjust the locations of the control points by solving Eq. (6) for c_k :

$$c_k^{new} \leftarrow \frac{\alpha m_k + \beta(c_{k-1} + c_{k+1}) + \gamma(\frac{3}{2}c_{k-1} + \frac{3}{2}c_{k+1} - \frac{1}{2}c_{k-2} - \frac{1}{2}c_{k+2})}{\alpha + 2\beta + 2\gamma}. \quad (7)$$

For the boundary cases $k \in \{1, 2, K-1, K\}$, we just state the equations below as their derivation is straightforward:

$$c_k^{new} \leftarrow \begin{cases} m_1 & \text{for } k = 1 \\ \frac{\alpha m_2 + \beta(c_1 + c_3)}{\alpha + 2\beta} & \text{for } k = 2 \\ \frac{\alpha m_{K-1} + \beta(c_{K-2} + c_K)}{\alpha + 2\beta} & \text{for } k = K-1 \\ m_K & \text{for } k = K \end{cases}. \quad (8)$$

To improve the robustness of the algorithm, we further constrain a control point to move only along the direction of the first principal axis (i.e. left–right axis) of the brain. For brains that have

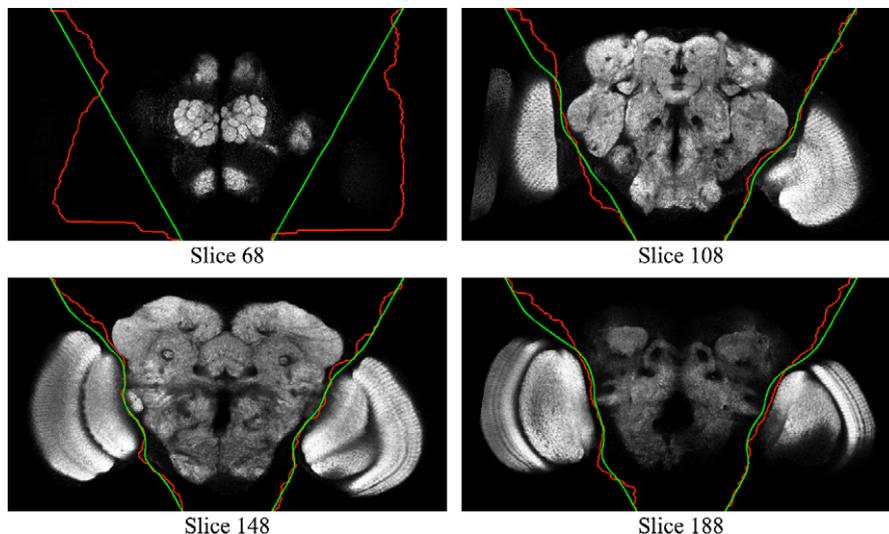


Fig. 4. Slice snapshots of a correctly segregated fly brain by (A) deformable model – green line, and (B) shortest path method – red line. Deformable model method correctly segments the OL and CB in all slices while shortest path method gives incorrect result in Slice 68.

been globally aligned so that the left–right axis is horizontal as shown in Fig. 1, we only need to adjust the x coordinate of c_k . This further simplifies the algorithm.

The algorithm starts from initialization of the deformable curve as a straight line, with control points evenly spaced in every 10 pixels in y -direction in the first slice of a 3D stack. Once the optimized curve for each slice is obtained, linear interpolation is carried out for each pair of control points (c_k and c_{k+1}) in the curve. For each y value, the corresponding x value is calculated as

$$x = \frac{y_{k+1} - y}{y_{k+1} - y_k} x_k + \frac{y - y_k}{y_{k+1} - y_k} x_{k+1}. \quad (9)$$

Fig. 4 shows the segmentation results of several 2D slices extracted from a 3D image. Green lines represent the resulting curves in extracted slices using deformable model method, while the red lines represent the results from shortest path method. The surface generated by aggregating all these curves is shown in Fig. 5.

We note that in the formulation of Eq. (1), a “prior energy” term can also be added to incorporate the prior information of how the deformable curve should look like.

3.3. Shortest path method

We also formulate another method to find the cutting curve based on the graph theory. An undirected graph is computed for each 2D slice, in which each pixel is a vertex $v = (x, y)$ and every pair of spatially adjacent pixels form an edge. The weight (ω) of each edge is defined as the product of the Euclidean distance, d , between its two vertices (v_i and v_j) and the square of the sum of their intensity values $I(v_i)$ and $I(v_j)$ as in Eq. (10),

$$E(v_i, v_j) = \omega(v_i, v_j) = (I(v_i) + I(v_j))^2 \times d \quad (10)$$

We then use Dijkstra’s algorithm [21] to find the lowest-weight path in the graph, which represents the curve that separates OL and CB. An example result is shown as the red lines in Fig. 4.

4. Experiments

Several experiments were done to test the accuracy and robustness of the proposed methods.

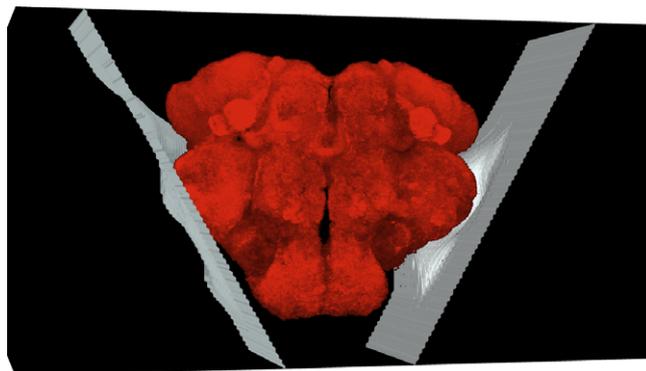


Fig. 5. Cutting surface in 3D obtained from deformable model method.

4.1. Response to brain surface–shape irregularity

In order to quantitatively assess the performance of the algorithms, we produced a simulated brain ($400 \times 350 \times 140$ voxels) which encapsulates most of the characteristic features of a real adult fruit fly brain which include shape peak, sandal regions, and surface with slow varying irregularity. Fig. 6a shows the 3D volume rendering of this simulated brain generated from a combination of predefined ellipsoids. Uniformly distributed random noises were added to the surface of these ellipsoids to simulate local shape irregularities. Noises of different levels ($s = \{0, 0.1, 0.3, 0.5\}$) were applied. Furthermore, zero-mean Gaussian image noise with variance of 0.1 was added to deteriorate the image quality of every image.

Fig. 6b shows a slice (slice 71) selected from the 3D volume in Fig. 6a, with different shape and image noises added. The areas covered by blue masks indicate the OL segmented by deformable model method with $\alpha = 1$, $\beta = \gamma = 0.3$. With shape noise of different levels added, the zoom-in figures of the region indicated by a yellow square in Fig. 6b are shown in Fig. 6c to illustrate the effect of different noise levels (s). The surface becomes fuzzier as s increases. Fig. 6d shows the results with different β values towards different levels of shape noises. The diamonds “◆” and crosses “×”, respectively, denote correct segmentations and major errors, determined by visual inspection of the results of various noise lev-

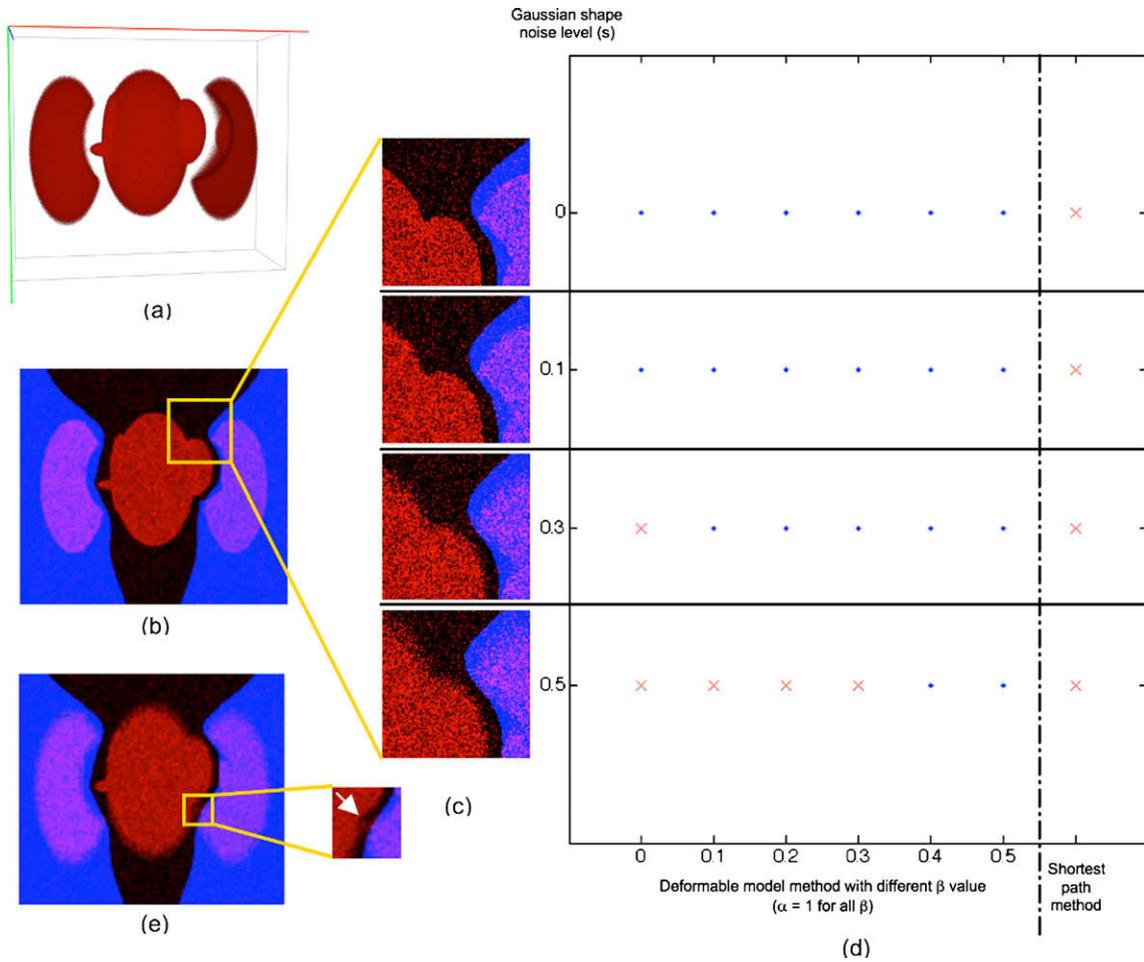


Fig. 6. Simulation study with different surface shape noise levels. (a) Simulated fly brain in 3D. (b) Slice 71 taken from (a). (c) Zoom in of (b) with different levels of shape surface noise (s) added (from top to bottom: $s = 0, 0.1, 0.3, 0.5$). Zero-mean Gaussian image noise with variance of 0.1 is added to all images. (d) Segmentation result of brains of different shape noise level using deformable model method with different β values and shortest path method. Diamonds “◆” denote correctly segmented result and crosses “X” denote major segmentations errors found at the corresponding noise level and β value by visual inspection. (e) A major segmentation error is found at the tip of the right OL as indicated by the white arrow.

els and β values. The deformable model method could successfully segment the OL up to shape noise level equaled to 0.3 with most β values. The maximum time to complete a segmentation of a 3D image was 60 s. When β increases to 0.4 and 0.5, image with $s = 0.5$ was also successfully segmented. However, the time required was much longer (>10 min). Fig. 6e shows a typical example of major error defined by our assessment criteria. The right OL was slightly wrongly segmented at the tip, indicated by a white arrow. Compared to the shortest path method that fails to produce correct

segmentation in all cases, the deformable model method is significantly more reliable. The shortest path method is also much slower: on average it took 400 s to process an image.

4.2. Response to variation in OL–CB distance and image noise

The area of the dark region between OL and CB varies from one brain to another. In order to test the robustness of the methods, another synthetic model having different simulated distances be-

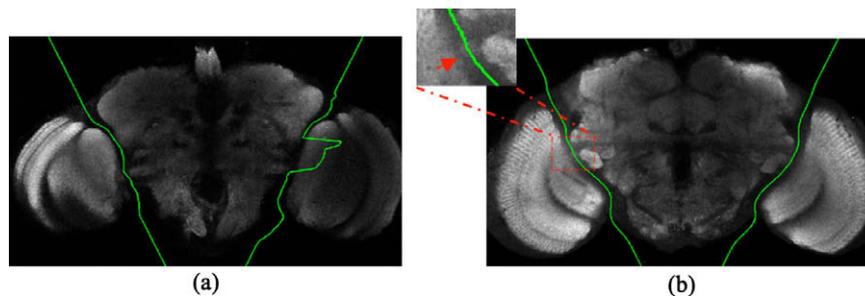


Fig. 7. Examples of major and minor errors. (a) Major error was found on the right side. The green curve cut into a significant region of the right OL. It occurred because several control points were trapped in the local minima in previous slices. (b) Minor error was found on the left side. The green curve cut on a small region belonged to CB as indicated by the red arrow. The error arises because CB is blurry and the gap between OL and CB is very small.

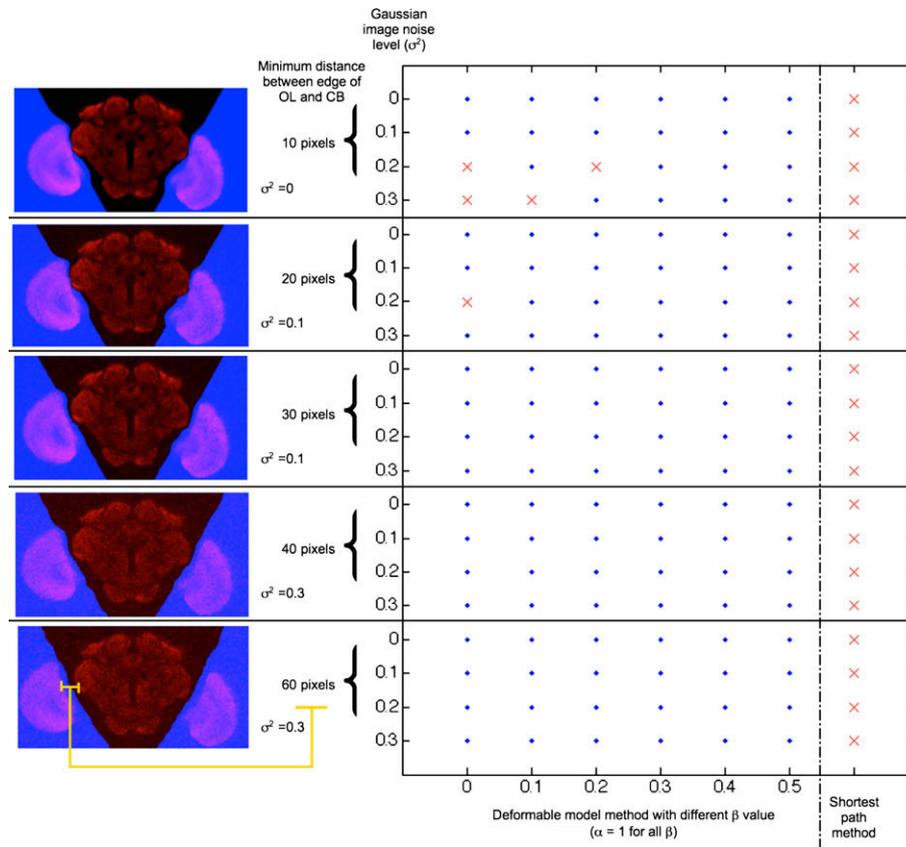


Fig. 8. Segmentation of model fruit fly brain of various image noise variances (σ^2) and distance between OL and CB. Results from deformable model method with different β values and shortest path method are presented. Diamonds “◆” denote correctly segmented results and crosses “✗” denote major segmentation errors found from the corresponding OL and CB separation, noise level and β value by visual inspection. Blue masks in the left column represent the segmented regions.

tween OL and CB was made from a real adult fruit fly brain. We generated these 3D volume images by adjusting the spacing between OL and CB to 10, 20, 30, 40, and 60 pixels. Zero-mean Gaussian image noise of different variances ($\sigma^2 = 0, 0.1, 0.2, 0.3$) was added to test the tolerance of the segmentation methods towards different image noise levels.

We visually inspected the segmentation results, and categorized them into three classes, *correct*, *major error*, and *minor error*. Major and minor errors were defined as the segmentation curve indicated in green, which cut into a large (right OL in Fig. 7a) and a small (left OL in Fig. 7b, arrow) region, respectively. A *major error* arose when control points were trapped into the local minimums where the pixel intensity was lower than that of the gap between the OL and CB. These major errors usually propagated from one slice to another. A *minor error* was usually due to blurry structures and extraordinary small gap between OL and CB.

Fig. 8 summarizes the results. Diamonds “◆” and crosses “✗” denote correctly segmented results and major segmentation errors found in the corresponding OL and CB separation, noise levels and β values. We say a brain has a segmentation error when an error was found from any one slice of the 3D image. For each simulated separation distance, a 2D slice extracted from the segmented 3D image is shown in the left column of Fig. 8. The noise level (σ^2) is shown at the lower-right corner of each figure. All these figures were obtained by deformable model method with $\beta = 0.3$. The segregated OLs are covered with blue masks. The deformable model method yielded errors when the separation between OL and CB is small (10 and 20 pixels) and image noise level was high ($\sigma^2 = 0.2, 0.3$). It provided correct segmentation in all other cases. On the contrary, the shortest path method failed to perform the segmentation in every case.

The processing of a 3D image stack using the deformable model method increased from 40 to 900 s with various β values. The average time to process an image with $\beta = 0.3$ was 50 s. The optimization took much longer to converge for larger β . For the shortest path method, the average processing time for an image was 500 s.

4.3. Segmenting real adult fruit fly brain images

We performed two sets of experiments for real brains. First, eight slices taken from 100 3D real fruit fly brain images (total $8 \times 100 = 800$ 2D images) of various image qualities were selected to identify the best parameter set for the deformable model method. The accuracy was determined based on the complete correctness of segmentation of both the left and right OLs in an image. As shown in Table 1, the combination of $\alpha = 1$ and $\beta = \gamma = 0.3$ resulted in the highest accuracy, about 92%.

Then, based on the result in Table 1, we took $\alpha = 1$ and $\beta = \gamma = 0.3$ for the deformable model method to compare the performance with shortest path method in the real adult fruit fly brain data. We first tested the two methods with real fly brain data produced in two different labs (100 brains from each). We categorized the segmentation results according to the errors that occur during the segmentation described in Fig. 7 and scored each of them by visual

Table 1
Segmentation results of real adult fruit fly brain images with different parameters.

α	1	1	1	1
β and γ	0	0.1	0.3	0.5
Accuracy (%)	36	69	92	76

Table 2

Comparison of the deformable model and the shortest path methods using real adult fruit fly brain images from two different labs. The segmentation results were regarded as having an error if any single slice in the whole 3D stack has an error.

N = 100		Whole brain correctly segmented	1 Minor error	More than 1 minor error	1 Or more major errors, or more than 3 minor errors
Simpson lab data	Deformable model	91	4	2	3
	Shortest path	56	4	7	33
Rubin lab data	Deformable model	80	0	0	20
	Shortest path	79	0	0	21

inspection. Table 2 shows the results obtained from the two methods. For the data set from the Simpson Lab, the deformable model method could achieve 91% correctness while the shortest path could only achieve 56%. For the Rubin Lab data, the deformable model method could achieve 80% of the brains correctly segmented. The shortest path could achieve 79% correctness. The errors normally happen when a brain is very dark.

The average computation time for a 3D volume image stack using the deformable model method ($\alpha = 1, \beta = \gamma = 0.3$) was 50 s, while the shortest path method took an average of 530 s to process. (image read/write time = 70 s for both methods).

5. Discussions and conclusions

This paper describes an accurate and robust scheme to segment the optic lobes from adult fruit fly brains. This approach provides a good pre-processing step for our ongoing high-throughput 3D volume registration pipeline. Based on the real adult fruit fly brain images segmentation results, the overall accuracy of the deformable model method is 85.5%. The majority of failure cases are found in poorly imaged or dissected brains which would not be further processed. An average computation time for a 3D volume image stack of a real brain using deformable model method ($\alpha = 1, \beta = \gamma = 0.3$) is about 50 s (on a Mac Pro with 2.8 GHz Intel Xeon CPU and Leopard operating system), showing that the speed is reasonable as an automated pre-processing step. When β and γ are small (<0.2), the contribution of length and smoothness constraint are less significant. The energy function is primarily driven by image intensity and thus more sensitive to noise. It explains why poorer results were obtained in these cases.

The deformable model method is a local searching method which can achieve good results for well-dissected and imaged brains. However, control points might be trapped into local minimum occasionally. Control points initialization plays an important role in this method as the accuracy of the result relies on it. Global alignment places every brain into a standard position so that the initialization can be fully automated. Local optimization significantly reduces the computation time. A prior model usually offers a good initialization for fast convergence but may also lead to incorrect result because of incorrect direction of convergence. However, it is rare in our dataset.

In contrast, the shortest path method can achieve a global minimum path for this formulation theoretically. Unfortunately, our experiments show that it is very sensitive to noise as it takes every pixel into account. In this case, the global minimum does not necessarily correspond to biologically meaningful results when this

method was used, it resulted in lower overall accuracy (67.5%) and took a much longer time to process.

The deformable-model based optic lobe segmentation method has been added as a function in the brain registration module of our image analysis platform V3D software [22].

Future work will require developing energy functions which incorporate the advantages of the two methods. We also anticipate that our current method will be applicable for the segmentation of other brain regions-of-interest in 3D volume images such as mushroom bodies, antennal lobes, protocerebrum, etc. These appropriate initialization models can be developed by experienced anatomists.

Acknowledgments

We thank Teri Ngo and Phuong Chung for generating the brain images used in this study and Gerry Rubin for support of this work, and Margaret Jefferies for help of text editing the manuscript.

References

- [1] A.H. Brand, N. Perrimon, *Development* 118 (1993) 401–415.
- [2] A.C. Groth, M. Fish, R. Nusse, M.P. Calos, *Genetics* 166 (2004) 1775–1782.
- [3] T. Lee, L. Luo, *Neuron* 22 (1999) 451–461.
- [4] H. Otsuna, K. Ito, *J. Comp. Neurol.* 497 (2006) 928–958.
- [5] B.D. Pfeiffer, A. Jenett, A.S. Hammonds, T.T. Ngo, S. Misra, C. Murphy, A. Scully, J.W. Carlson, K.H. Wan, T.R. Laverty, C. Mungall, R. Svirskas, J.T. Kadonaga, C.Q. Doe, M.B. Eisen, S.E. Celniker, G.M. Rubin, *PNAS* 105 (2008) 9715–9720.
- [6] H. Peng, *Bioinformatics* 24 (2008) 1827–1836.
- [7] R. Brandt, T. Rohlfing, J. Rybak, S. Krofczik, A. Maye, M. Westerhoff, H.C. Hege, R. Menzel, *J. Comp. Neurol.* 492 (2005) 1–19.
- [8] A. Jenett, J.E. Schindelin, M. Heisenberg, *BMC Bioinformatics* 29 (2006) 544.
- [9] A.E. Kurylas, T. Rohlfing, S. Krofczik, A. Jenett, U. Homberg, *Cell Tissue Res.* 333 (2008) 125–145.
- [10] C. Xu, D.L. Pham, J.L. Prince, in: J.M. Fitzpatrick, M. Sonka (Eds.), *Handbook of Medical Imaging*, vol. 2, Medical Image Processing and Analysis, SPIE Press, Bellingham, 2000, pp. 129–174.
- [11] D.L. Pham, C. Xu, J.L. Prince, *Annu. Rev. Biomed. Eng.* 2 (2000) 315–338.
- [12] L.G. Shapiro, G.C. Stockman, *Computer Vision*, Prentice-Hall, New Jersey, 2001, pp. 279–325.
- [13] V. Kolmogorov, C. Rother, *PAMI* 29 (2007) 1274–1279.
- [14] Y.C. Chen, Y.C. Chen, A.S. Chiang, *Comput. Methods Programs Biomed.* 89 (2008) 239–247.
- [15] J.R. Anderson, S.F. Barrett, M.J. Wilcox, *J. Electron. Imaging* 15 (2006) 043005.
- [16] T.F. Chan, L.A. Vese, *IEEE Trans. Image Process.* 10 (2001) 266–277.
- [17] A. Hofbauer, *Eine Bibliothek monoklonaler Antikörper gegen das Gehirn von Drosophila melanogaster*, Habilitation Thesis, University of Würzburg, Würzburg, Germany, 1991.
- [18] H. Peng, F. Long, X. Liu, S.K. Kim, E.W. Myers, *Bioinformatics* 24 (2008) 234–242.
- [19] M. Kass, A. Witkin, D. Terzopoulos, *Int. J. Comput. Vis.* 1 (1988) 321–331.
- [20] C. Xu, J.L. Prince, *IEEE Trans. Image Process.* 7 (1998) 359–369.
- [21] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, *Introduction to Algorithms*, MIT Press and McGraw-Hill, Cambridge, 2001, pp. 595–601.
- [22] V3D. Available form: <<http://penglab.janelia.org/proj/v3d>>.