JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 38, 253-271 (2022)
DOI: 10.6688/JISE.202201_38(1).0013

Drosophila Brain Aligner: Registering 3D Point Clouds in 2D Parameterization Domain*

HAO-CHIANG SHAO[1,+], LU-HUNG HSU[2], YUNG-CHANG CHEN[2],
YING-CHU HUANG[3] AND SHIH-TING HUANG[4]
[1]*Department of Statistics and Information Science*
Fu Jen Catholic University
New Taipei, 242 Taiwan
[2]*Department of Eletrical Engineering*
National Tsing Hua University
Hsinchu, 300044 Taiwan
[3]*Service Systems Technology Center*
Industrial Technology Research Institute
Hsinchu, 310401 Taiwan
[4]*Department of Statistics*
University of California, Irvine
Irvine, CA 92697, USA

In order to study brain functions and connectome, scientists need to register and warp source image data of some model organism, *e.g. Drosophila*, into a pre-defined standard atlas. Such registration and warping procedure is conventionally driven and constrained by point clouds extracted from contour edges of manually-segmented landmark tissues within a 2D+Z image volume. However, it is difficult to register two dense 3D point clouds. The fitness between spatial distributions of two point clouds cannot guarantee the matchness between 3D anatomical surfaces from which point clouds were sampled. Hence, to settle down this problem, we propose in this paper a strategy to register 3D point clouds of *Drosophila* brain in 2D parameterization domain. Our contributions are twofold. First, instead of registering point clouds directly, our method was designed to register two mesh surface models, each defined by a to-be-registered point clouds, so that the anatomical shape details described by the point cloud can be aligned after registration. Second, the proposed method performs registration/warping in a parameterization domain, and hence it no longer needs a rigid transformation to globally align and scale the input models. Experiments show that the surface-to-surface distance is reduced after registration and warping process. For models with an about 1100-voxel-long bounding box diagonal, the average surface-to-surface distance is reduced to about 0.1 voxel after registration. The proposed method is effective.

Keywords: point cloud registration, surface registration, parameterization, *Drosophila* brain, mesh surface model

1. INTRODUCTION

Since National Institutes of Health (NIH) launched Human Connectome Project (HCP) in 2009, systematic studies on both neuron-to-neuron and structure-to-function

Received February 27, 2020; revised December 6 & 28, 2020; accepted January 27, 2021.
Communicated by Jing-Ming Guo.
+ Corresponding author: shao.haochiang@gmail.com
* This project was sponsored by Ministry of Science and Technology, Taiwan under Grant No. MOST 107-2320B-030-012-MY3.

relationship in brain have caught public attentions. Because scientists need a 3D brain road-map, *i.e.*, a 3D brain standard model, to explore the labyrinth of neuron circuitry, several representative 3D standard brains of model organisms, *e.g.* sphinx moths [1], honey bees [2], and desert locusts [3], were already developed for this purpose. Backbone techniques for developing a standard brain are registration and warping methods, as mentioned in standard brain construction methods, such as ISA (iterative shape averaging) [4], VIB (virtual insect brain) [5], and Shao *et al.*'s method [6]. Both ISA and VIB are designed for constructing a *probabilistic* standard brain model, whereas Shao *et al.*'s method concentrates on the development of a *stereotypical* 3D standard brain model. In addition to standard brain construction strategies, scientists still need image atlasing methods to align microscope image volumes with a pre-defined standard reference model [7], and image atlasing methods also operate on the basis of registration techniques. The proposed method focuses on registering two 3D mesh surfaces defined by dense point clouds. It aims to derive a mapping relationship, *i.e.* the function Π described in Eq. (13), between two surfaces. Such mapping relationship will be considered in the future as boundary constraints for atlasing and warping images into a *stereotypical* standard brain model, *e.g.* the one derived by Shao *et al.*'s method [6]. Hence, the proposed method is different from probabilistic approaches, such as ISA and VIB, intrinsically.

Registration methods are driven by landmark correspondences. However, although the registration/warping accuracy within landmarks' local neighborhoods can be guaranteed, it is difficult to align regions that are distant from landmarks or not dense with landmarks. This fact leads a common artifact that while landmark points within the source are all well-aligned to those on the target, the entire 3D shape of the source might be not. In order to solve this issue, several surface-constrained registration framework were proposed by taking especially the point cloud on a 3D surface into account and exploiting 3D surface as a boundary condition in the registration process [8, 9]. Consequently, the accuracy of surface registration, or the point cloud registration equivalently, becomes a performance bottleneck in common biomedical image volume registration problems.

The point cloud registration problem is very similar to the surface registration issue. Due to the development of imaging technologies, organs and its surrounding tissues can all be represented as point clouds, in which each vertex is sampled from target surfaces via imaging devices [10], so point clouds are often converted to polygonal or triangular mesh modelsvia a meshing [11] or a surface reconstruction procedure [12] for visualization and animation purposes. Specifically speaking, by giving a point cloud $\mathbf{V}$ a graph structure (*i.e.*, the adjacency matrix) G via meshing or surface reconstruction techniques, a mesh surface model $\mathcal{M}(\mathbf{V},G)$ can be defined. Consequently, the point cloud registration issue in biomedical applications can be regarded as a mesh surface registration problem, *i.e.*, a graph-assisted cloud registration task in some sense.

In this paper, we proposed a parameterized point cloud registration strategy for aligning 3D brain surface of *Drosophila*, *i.e.*, fruit fly, one of the most proper model for neural connectivity researches [6, 13–18] because of its small enough brain yet rather sophisticated behavioral repertoire. Although the proposed method aims to solve a very special point cloud registration issue, it has following primary advantages and contributions. First, we formulate a complex 3D fly brain registration issue as a tractable 2D registration problem via a 2D parameterization domain for fly brain surface. This idea, similar to those stated in [6, 19], is to introduce an intermediate reference platform for data align-

ment and integration. Second, our registration process is triggered by correspondences between boundary points on the parameterization domain. Such design makes the proposed method can skip the global registration process and take surface boundaries as crest-lines [20], so even surface textures can be aligned well through our method.

The rest parts of this paper are organized as follows. In Section 2, several benchmark point cloud registration method will be reviewed; then, in Section 3, each stage of the proposed method will be elaborated. Finally, we demonstrate our experiment results in Section 4 and draw our conclusions in Section 5.

2. RELATED WORK

There have been many point set registration approaches. The most classical method is ICP (iterative closest point) algorithm [21]. Assuming each vertex in the source point cloud $\mathbf{V}_s$ corresponds with the closest vertex to it in the target point cloud $\mathbf{V}_t$, ICP performs rigid registration in an iterative way by updating source vertex coordinates via an estimated least-square rigid transformation of each iteration. Therefore, ICP can work best if the initial guess of the rigid transformation is given or if the initial pose of $\mathbf{V}_s$ is sufficiently close to that of $\mathbf{V}_t$, but it cannot handle local non-rigid deformations. Moreover, the NDT (normal distributions transform) [22, 23] uses standard optimization techniques, applied to statistical models of 3D points, to determine the most probable registration between two point clouds. NDT subdivides the space containing a point cloud into cells. To each cell, it assign a normal distribution that models locally the probability of measuring a point. Hence, NDT can derive a transformation with piecewise continuous and differentiable probability density, and NDT requires no explicit correspondences accordingly. Additionally, Myronenko *et al*. developed Coherent Point Drift (CPD) method for non-rigid point-cloud registration [24]. CPD also takes a probabilistic approach to align point sets. Unlike earlier approaches to non-rigid registration that exploit deterministic approaches, such as the thin-plate-spline (TPS) model [25–27], CPD assumes instead the point set represents the Gaussian mixture model (GMM) centroids. Consequently, when two point sets are best registered, the GMM posterior probability for a given data point should reach its maximum. Myronenko *et al*. called their method *coherent* because the GMM centroids, in their design, are forced to move coherently as a group to preserve the topological structure of the point sets. However, CPD is not so suitable for large scale point cloud registration due to computational complexity, as reported in [28], and the fact that the alignment in probability cannot guarantee the matchness for each vertex position.

The most relevant research work to our problem is Nain *et al*.'s routine [29]. Nain *et al*. derived a multiscale comparison of human hippocampus models, reconstructed from point clouds, by surface registration. To register those surface models, Nain *et al*. used conformal mapping to map input hippocampus models to a sphere at the beginning stage. Next, they remeshed the mapped models – that is, resampling the point cloud – to make their connectivity information identical. Finally, they registered each remeshed model and then applied spherical wavelet transform [30] to derive the multiscale comparison results. However, because the *Drosophila* brain model reconstructed from confocal microscope image volume is topologically a **torus** rather than a sphere, it is impossible to adopt Nain *et al*.'s method and map a *Drosophila* brain to a unit sphere. Consequently, we developed our parameterized non-rigid point cloud registration method.

3. REGISTERING 3D POINT CLOUDS IN 2D PARAMETERIZATION DOMAIN

The goal of the proposed method is to register two fly-brain point clouds, *i.e.*, $\mathbf{V}_s$ and $\mathbf{V}_t$, while preventing the relative relationship among vertices of each point cloud altered. In other words, the 3D surface structure described by each point clouds must be retained after registration. Because meshing algorithms capable of reconstructing a surface model $\mathcal{M}(\mathbf{V}, G)$ from a point cloud $\mathbf{V}$ are matured, *e.g.*, [11, 12, 31], and 3D scanning devices can generate a mesh surface model by scanning an object surface, we here assume all to-be-registered point clouds were transformed into mesh surface models in advance. As a result, we focus in this paper on the surface registration process, which can be regarded as a graph-assisted point cloud registration method, for fly brain models. The proposed method is therefore implemented in three stages, namely; (i) parameterization (preprocessing); (ii) thin-plate spline registration, and iii) surface reconstruction.

The fly brain model is topologically a 3D torus. Hence, we need several preprocessing sub-steps, including trimming, simplifying and smoothing, to map a 3D fly brain model onto a 2D parameterization domain. This first stage ends by recording the mapping relationship as $\Pi : \mathbb{R}^3 \rightarrow \mathbb{R}^2$. Next, the second stage aims to register two input models in the parameter domain, *i.e.*, $\Pi(\mathbf{V}_s)$ and $\Pi(\mathbf{V}_t)$. Here we run 2D TPS-based (thin-plate spline) registration method and take boundary vertices on the parameterization domain as necessary landmarks. Finally, we reconstruct a well-aligned 3D surface model, *i.e.*, 3D point cloud, in the third stage by transforming a deformed 2D parameterization result inversely back to vertex domain through Π^{-1}. Note that all interpolation required in our registration routine is computed according to the local neighborhood described by the graph structure G.

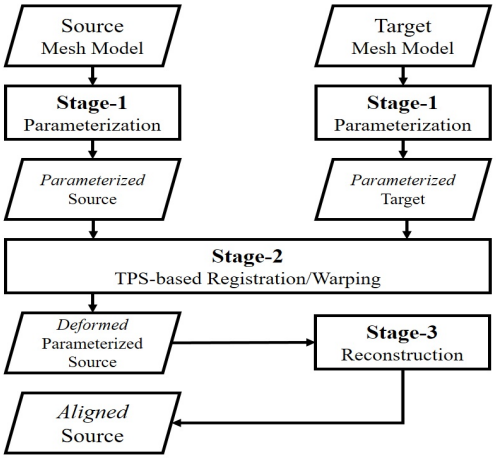


Fig. 1. Block diagram of the proposed method.

Fig. 1 shows the flow diagram of the proposed method, and each stage will be detailed in following subsections.

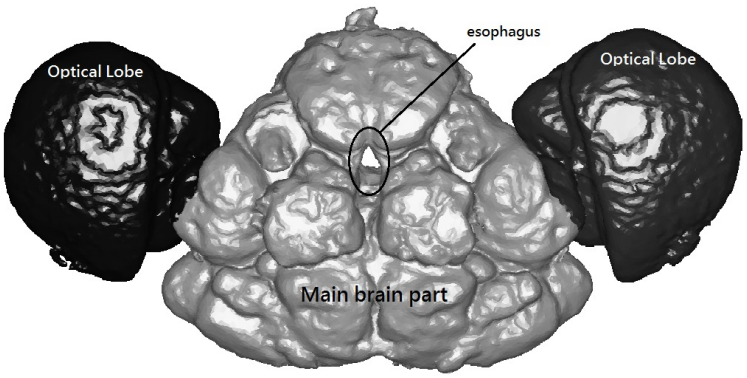


Fig. 2. Top view of a *Drosophila* brain model. Because of the esophagus, the *Drosophila* (fruit fly) brain model is topologically a torus model. Highligthted in black are two optical lobes. In this paper, we focus on the registration of the main brain part.

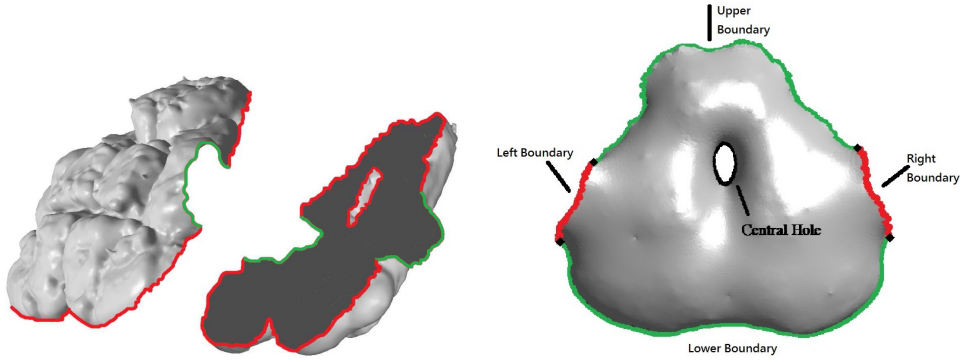
3.1 Parameterization

The parameterization stage aims to transform a 3D input mesh surface model $\mathcal{M}(\mathbf{V},G)$ into a 2D parameterization space, *i.e.*, $\mathcal{M}(\Pi(\mathbf{V}),G)$. Here, Π represents the mapping function obtained after the parameterization process; and, $\mathbf{V}$ and G denote the vertices and the graph recording the geometric relationship, *i.e.*, the connectivity, of the given mesh surface model $\mathcal{M}$, respectively.

The **esophagus** running through the fly head makes the *Drosophila* brain model a topological ***torus***, as illustrated in Fig. 2, and consequently conventional parameterization strategy that maps a to-be-processed mesh model onto a spherical domains is not applicable in our case. Therefore, we dissect a to-be-registered brain surface model into two hemi-surfaces and then map each of them onto a 2D rectangle space to achieve parameterization instead. Specifically speaking, our parameterization stage consists of four sub-stages, namely, 1) surface simplification, 2) model trimming and dissecting, 3) surface smoothing, and 4) model parameterization.

(A) ***Simplification***: The input surface models can be simplified (subsampled) by any classical benchmark mesh simplification algorithm, *e.g.* PM [32], QEM [33], MAPS [34], or other tools, *e.g.* MeshLab [35] and MATLAB built-in function **pcdownsample**, to reduce both computational time cost and memory cost. However, mesh simplification always accompanies with more or less distortions. We need to check if the distortion is tolerable after simplification. For a point cloud describing a complex shape like a fly brain or any tissue/neuropil within, we recommend not to reduce the number of points (vertices) to less than $10,000$, as we will discuss via Fig. 7 in Section 4. Note that this sub-stage can be optional.

(B) ***Trimming and dissecting***: The main purpose of this substage is to trim uninteresting regions, *e.g.* optical lobes, and then dissect the region of interest, *e.g.* main brain part, into two hemi-spheres, as illustrated in Figs. 2 and 3 (a). Because the main brain part contains more than a hundred thousand neurons inside, as demonstrated in neuron images on the FlyCircuit database [36], people would like to pay more attention to it in *Drosophila* brain surface model registration problems. Therefore, we regard the main brain part as



(a) (b)

Fig. 3. Trimmed and dissected half brain surface; (a) Example of half brain surfaces; (b) The central hole (esophagus) and the four boundaries.

the region of interest (ROI) in our registration procedure and disregard the optical lobes containing only neurons related to visual signals.

To dissect the brain surface model, we use the *Drosophila* brain coordinate system proposed in [37] to define the upper and lower half brain surfaces by using a brain bounding box. We dissect the bounding box of brain surface model into upper and lower halves, and each of the half bounding boxes defines a half-brain surface.

(C) ***Smoothing***: Because complex textures on brain surface may perplex the parameterization, the input brain surface has to be smoothened. We apply the average kernel used in Loop subdivision scheme [38, 39] on all vertices to smoothen the input mesh surface model. The Loop subdivision scheme was designed to send an interpolated vertex $\hat{v}$ to where it is supposed to be, *i.e.*, the target position v, on the limit surface [38] by following equations.

$$v = \frac{1}{\alpha(n)+n}\left(\hat{v} + \sum_{v_i \in \mathcal{N}(\hat{v})} v_i\right) \tag{1}$$

$$= \frac{\alpha(n)}{\alpha(n)+n}\hat{v} + \frac{n}{\alpha(n)+n}\left(\frac{1}{n}\sum_{v_i \in \mathcal{N}(\hat{v})}\right), \tag{2}$$

with

$$\alpha(n) = \frac{n(1-\beta(n))}{\beta(n)}, \text{ and} \tag{3}$$

$$\beta(n) = \frac{5}{4} - \frac{(3+2cos(2\pi/n))^2}{32}, \tag{4}$$

where $\mathcal{N}(\hat{v})$ denotes the one-ring-neighborhood of $\hat{v}$, and n is the cardinality of $\mathcal{N}$. The average kernel in Eq. (2) moves a vertex to some place between its original position and the berycenter of its connecting vertices, and therefore it is able to smoothen the surface

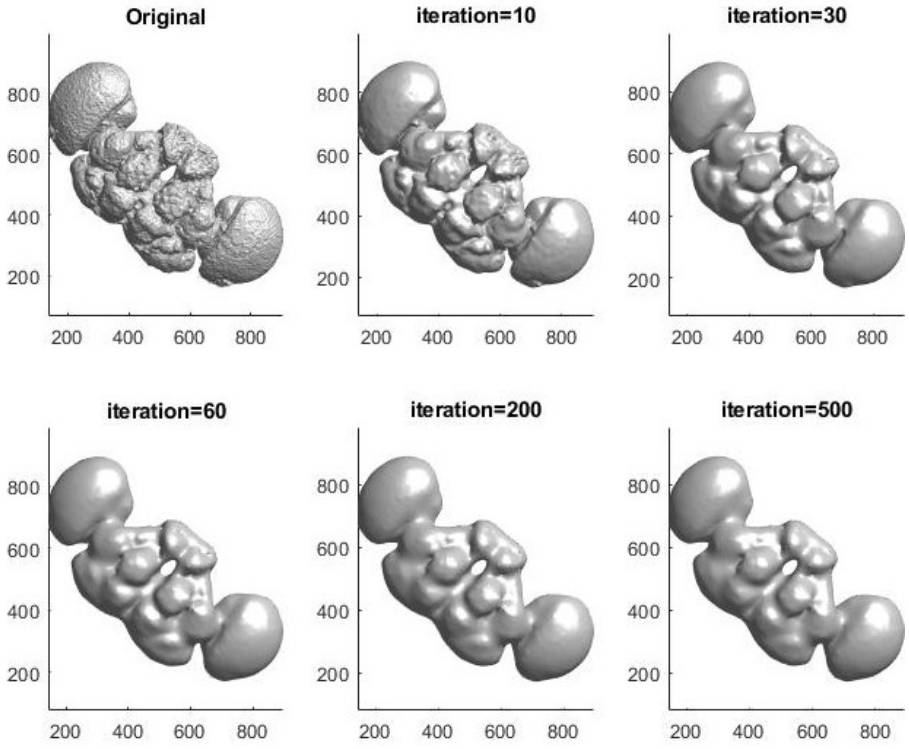


Fig. 4. Examples of smoothing results.

model, as demonstrated in Fig. 4.

(D) ***Parameterization***: In this sub-stage, each vertex $v_i \in \mathbf{V}$ in the processed half brain surface is mapped to its counterpart q_i in the rectangle parameterization domain. Another goal of this sub-stage is to record the mapping function Π describing the relationship $q_i = \Pi(v_i)$.

As illustrated in Figs. 3 (b) and 5, a half brain surface has five primary boundaries, namely (i) central hole; (ii) upper boundary; (iii) lower boundary; (iv) left boundary, and v) right boundary. To derive the parameterized result $\mathcal{M}(\Pi(\mathbf{V}), G)$, we first map each of the five boundaries to their counterparts in rectangle domain, and then we apply Loop average kernel on all q_i to make interior vertices moving toward to boundaries properly and uniformly-distributed in the rectangle (parameterization) space [1].

Finally, because we map all to-be-registered models onto an intermediate reference domain, our parameterization procedure can be regarded as the global alignment process used in common registration method. Hence, with the aid of the parameterization process, we can skip typical global rigid transformation and run TPS-registration non-rigid local transformation directly. Also, note that the mapping function Π that records the parameterization process and the correspondence between q_i and v_i will be used to inversely transform the parameterization domain to the 3D surface in the reconstruction stage.

[1] Based on our experiment results, it needs about 500 smoothing iterations.

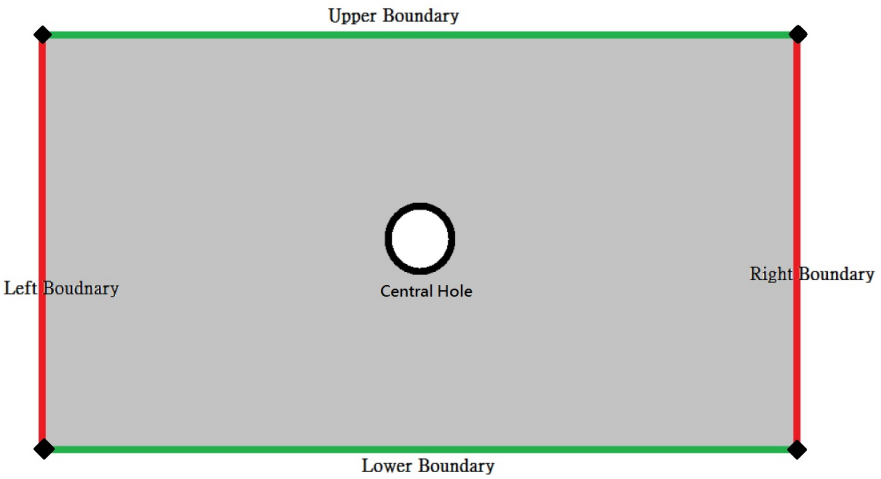


Fig. 5. 2D rectangle parameterization domain. The parameterization stage aims to map each vertex $\mathbf{v} \in \mathbb{R}^3$ on the 3d surface onto a point $p \in \mathbb{R}^2$ in this 2d rectanble domain.

3.2 TPS Registration

Given a n pairs of corresponding landmarks p_i and q_i, $i = 1, \cdots, n$ in spaces of dimension d, the TPS registration aims to find a continuous transformation $\mathcal{T} = \mathbb{R}^d \rightarrow \mathbb{R}^d$, which can minimize a given objective function and fulfill the interpolation conditions [25, 26]:

$$q_i = \mathcal{T}(p_i),\ i = 1, \cdots, n. \tag{5}$$

Based on the objective function and results given in [26, 40], the analytic solution for any interior point $\mathbf{x} = [x, y]^T$ could be approximated as

$$\mathcal{T}(\mathbf{x}) = \sum_{\nu=1}^{M} \alpha_\nu \phi_\nu(\mathbf{x}) + \sum_{i=1}^{n} w_i U(\mathbf{X}, p_i), \tag{6}$$

with basis functions $U(\mathbf{X}, p_i)$. Here, the basis functions $U(\mathbf{X}, p_i)$ span the n-dimensional space depending on landmarks p_i, and the nullspace is spanned by $\phi_1(\mathbf{X}) = 1$, $\phi_2(\mathbf{X}) = x$, and $\phi_3(\mathbf{X}) = y$. Meanwhile, coefficients α_ν and w_i in Eq. (6) can be obtained by solving the following system of linear equations:

$$\mathbf{\Psi}\alpha + \mathbf{K}\mathbf{w} = \hat{\mathbf{p}} \tag{7}$$

$$\mathbf{\Psi^T}\mathbf{w} = \mathbf{0}, \tag{8}$$

where $K_{i,j} = U(p_i, p_j)$, $P_{i,j} = \phi_j(p_i)$, and $\hat{p}$ is the column vector of one component of the coordinates of the landmarks q_i of the **to-be-registered source** data. The constraint $\mathbf{P}^T\mathbf{w} = 0$ represents the boundary conditions and ensures that the elastic part of the transformation is zero at infinity.

In our implementation, for two landmark point sets $\mathbf{V}_Q$ and $\mathbf{V}_P$, we determine the point correspondence (p_i, q_i) by $p_i = \arg\min_k |q_i - p_k|$, $\forall p_k \in \mathbf{V}_P$ and $\forall q_i \in \mathbf{V}_Q$. Again,

note that in our method both $\mathbf{V}_Q$ and $\mathbf{V}_P$ contain boundary vertices of the parameterization domains. Also, we adopt radial basis functions $U(\mathbf{x}, p_i)$ of following form:

$$U(p,q) = \|p-q\|^2 log\|p-q\|^2, \tag{9}$$

and we also introduce a user-specified diagonal matrix $\mathbf{W}^{-1}$ whose diagonal entries denote weights of landmarks. As a result, we modified Eq. (7) as

$$\mathbf{\Psi}\alpha + (\mathbf{K} + \lambda \mathbf{W}^{-1})\mathbf{w} = \hat{\mathbf{p}}, \tag{10}$$

where $\lambda \geq 0$ is a regularization term used to control the smoothness of the deformation field.

Finally, the proposed method interpolates the interior vertices by exploiting landmark points on both inner boundary (*i.e.*, central hole) and outer boundary of the parameterization domain. Hence, our method is triggered by boundary conditions and thus can retain boundary shapes better than conventional TPS-based approaches. Also, $\mathbf{W}^{-1}$ is designed to control the influences of different landmarks because biologists would like to give larger weights to landmarks near important organs, tissues or anatomical structures.

3.3 Surface Reconstruction

This stage aims to send *warped vertices* back to 3D surface by using the berycentric coordinate coefficients derived in parameterization domain. Briefly, any warped vertex $\mathcal{T}(\mathbf{x}_{source})$ in the 2D parameterization space must lie in one of the triangular patches, *i.e.*, $\Delta p_a p_b p_c$, belonging to $\mathcal{M}(\Pi(\mathbf{V}_{target}), G_{target})$. Therefore, based on berycentric coordinate, $\mathcal{T}(\mathbf{x}_{source})$ can be represented as the convex combination of p_a, p_b, and p_c. that is,

$$\mathcal{T}(\mathbf{x}) = \alpha\Pi(p_a) + \beta\Pi(p_b) + \gamma\Pi(p_c), \tag{11}$$

$$\text{with } \alpha+\beta+\gamma = 1 \text{ and } \alpha, \beta, \gamma > 0. \tag{12}$$

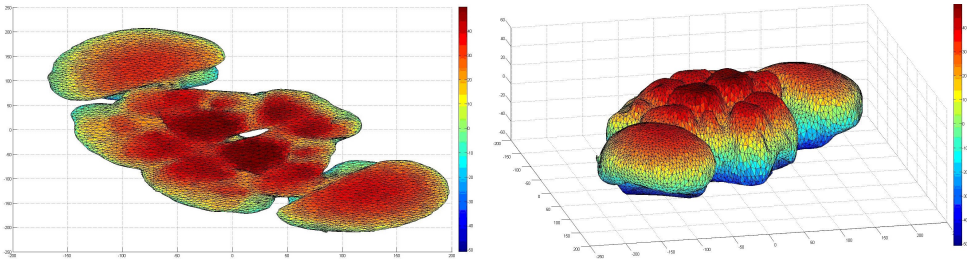
As a result, a registered and warped vertex in parameterization domain can be inverse-transformed to 3D surface $\mathcal{M}(\Pi^{-1}(\mathcal{T}(\mathbf{V}_{source})), G_{source})$ by

$$\Pi^{-1}(\mathcal{T}(x)) = \alpha p_a + \beta p_b + \gamma p_c. \tag{13}$$

4. EXPERIMENT RESULT

4.1 Dataset

The point clouds used in our experiments are the contour points of the manual segmentation results of image volumes acquired by a Zeiss LSM 510 confocal microscope. The sampling resolutions of the source image volume along x-, y- and z-direction are respectively 0.35, 0.35 and 1.0 μm; that is, the spatial size of each image voxel is $0.35 \times 0.35 \times 1 \mu m^3$. All 3D brain surface models were constructed by using a biomedical image processing software **Amira** [31]. Also, all distances/lengths, *e.g.* surface-to-surface distance and length of bounding-box diagonal (BBox diagonal), reported in this



(a) (b)

Fig. 6. Illustration of Brian-1 in wire-frame style. Here we show edges connecting vertices pairs. Because it diffi cult to illustrate a point could containing about 100,000 vertices, we show a point cloud $\mathbf{V}$ by phong-shading it surface model $\mathcal{M}(\mathbf{V},G)$ in this paper.

paper were measured in **voxel**. Finally, Table 1 lists detail information of meshed point clouds we used in this paper. We also show a wire-frame model in Fig. 6 to illustrate the anatomical shape of fly brain. Note that because the point cloud constituting the brain surface is too dense to be clearly demonstrated, we show only rendered results of all brain models later in this section instead.

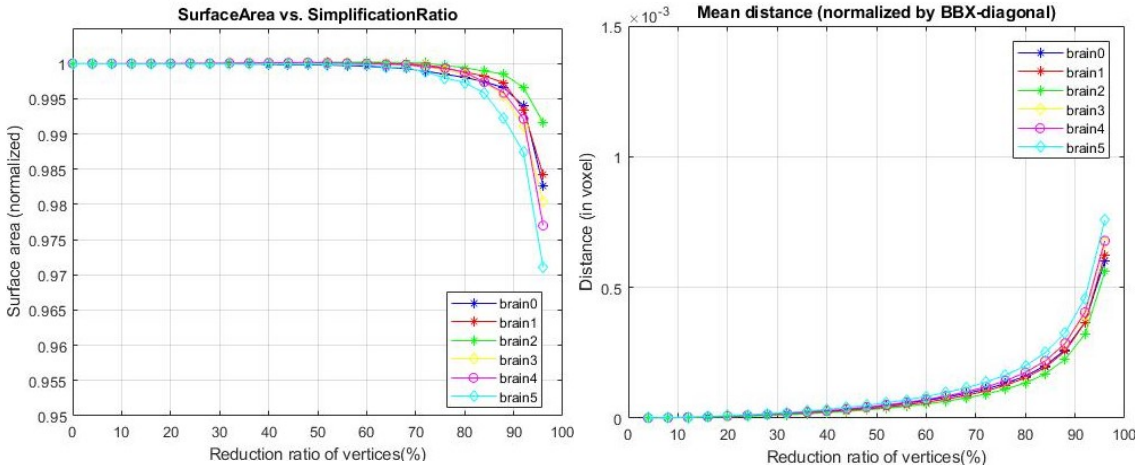
Table 1. Information of meshed pointcClouds $\mathcal{M}(\mathbf{V},G)$.

	# of Vertices	# of Patches	Surface Area	BBox Diagonal
Brain 0	100,023	122,338	623,932	1122.969
Brain 1	100,004	118,623	582,625	1058.247
Brain 2	99,996	120,963	764,043	960.011
Brain 3	100,018	131,711	675,160	1096.227
Brain 4	100,001	116,073	664.723	1114.798

4.2 Preprocessing for Parameterization

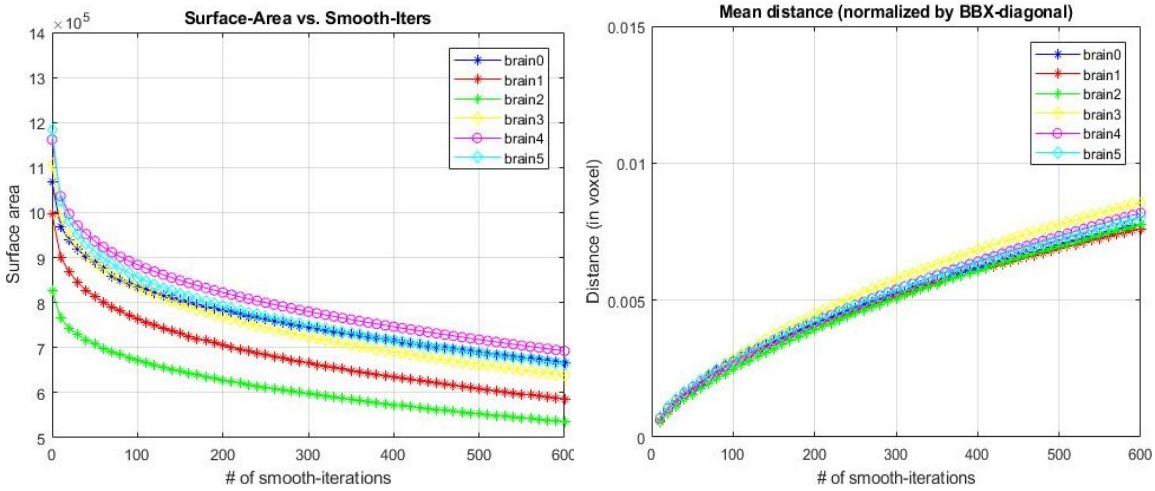
We stated in Subsection 3.1 (A) that we can simplify, *i.e.*, downsample, the brain surface model $\mathcal{M}(\mathbf{V},G)$ by any mesh simplification algorithm. However, notice that when we reduced the number of points (vertices) to less than 10,000, *i.e.*, about 10%, the surface area shrinks and the mean surface-to-surface distance increases significantly, as illustrated in Fig. 7. This implies that over-simplification distorts the shape of the downsampled point cloud obviously. As a result, the downsampling ratio for such point clouds should never be less than 0.2, empirically.

Next, we verify the number of iterations required for the smoothing process. Illustrated in Fig. 8 are curves showing how total area and average surface-to-surface distance vary during the smoothing process. Each data point in Fig. 8 is a measurement based on one smoothened surface model. As we can see, in general, both these two kinds of curves vary slowly after about the **400th** iteration of smoothing procedure. Hence, although we can use the difference between surface areas (or distances) of two consecutive, *i.e.*, the i^{th} and the $i+1^{th}$, iterations as an indicator to stop the iterations, it is efficient, and reasonable



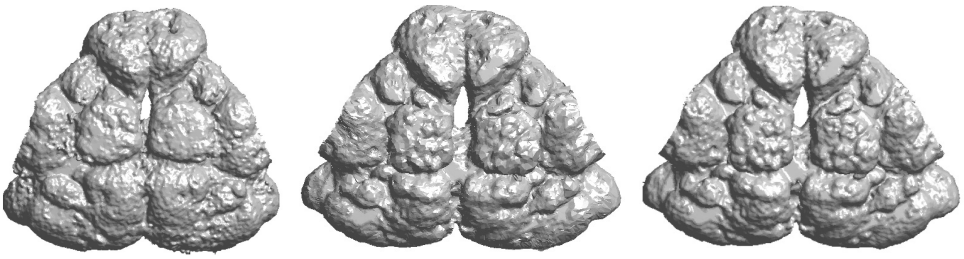
(a) (b)

Fig. 7. Distortion-vs-simplification curves; (a) Normalized surface area; (b) Normalized mean surface-to-surface distance. For a meshed point cloud with 100k vertices describing a complicated anatomical structure like fly brain, it can still retain shape details after reducing 80% vertices. Note that this phenomenon had been already reported in classical papers focusing on mesh simplification or remeshing, such as PM [32], QEM [33], and MAPS [34].



(a) (b)

Fig. 8. Distortion-vs-smoothing curves; (a) Total surface area; (b) Normalized mean surface-to-surface distance. These two curves show that after the 400^{th} iteration, both surface area and mean distance vary quite slowly.



(a) (b) (c)

Fig. 9. Top view of a to-be-aligned model pair; (a) Source model: **Brain-0**; (b) Aligned source model; and (c) Target model **Brain-1**.

as well, to terminate the smoothing iteration by about the **500th** iteration[2].

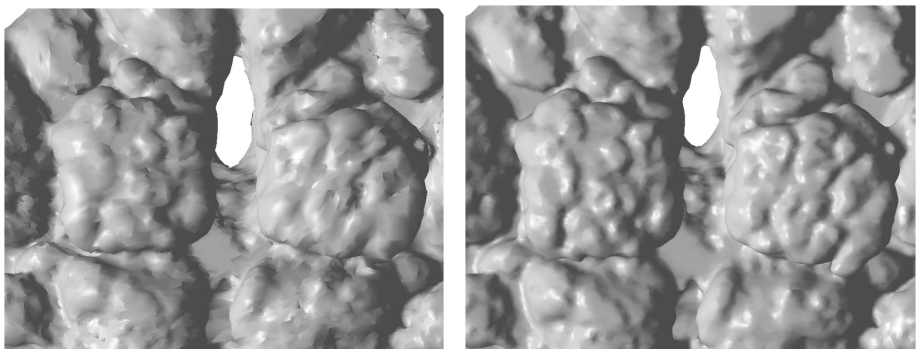
The average kernel of Loop subdivision scheme, described in Eqs. (1)–(4), moves every vertex v_i to a weighted average position of v_i and the adjacent vertices thereof. Consequently, if a to-be-smoothed mesh model is topologically a ball, the distortion value will not converge until i) the to-be-smoothed model shrinks into a ball, and ii) each vertex is co-planar with its adjacent vertices and locates on the position satisfying Eq. (1). However, this situation surely results in a smoothed yet distorted model, which is surely not similar to its original structure and shape. Therefore, we finally adopted an empirical setting and let the stop criterion to be the 500th iteration, although the distortion value does not begin to converge at that time.

4.3 Surface Alignment

To verify the effectiveness of the proposed method, we selected **Brain-0** to be the source mesh surface model, registered and warped it to other four brain models through different methods in turn, and then measure the distance between aligned surface and the target. Listed in Table 2 are the mean and root-mean-square surface-to-surface distances, which were all measured by **METRO** [41]. Table 2 shows that the proposed method outperforms other benchmark point cloud registration methods. These results also indicate that the accuracy of point cloud registration can be improved by using a pre-made graph information, *i.e.* the connectivity among vertices. By registering two mesh surface models in a parameterization domain, we can finish conventional global rigid transform and local non-rigid transform together. Moreover, one extra interesting observation based on Table 2 is that for large point clouds containing tens of thousands vertices to represent complicated anatomical structure and surface texture, a typical non-rigid registration method, *e.g.* CPD, may work not so good as the very conventional rigid registration method, *i.e.*, ICP, as is similar to what reported recently in [28].

Moreover, demonstrated in Fig. 9 are the half-surfaces of (a) source model, (b) deformed surface derived by our method, and (c) target model (**surface-4**); and, Fig. 10

[2]It takes about 60 seconds to measure the surface-to-surface once but takes only 0.5 seconds to run one smoothing iteration on a computer with an Intel i5-3470 CPU.



(a) (b)

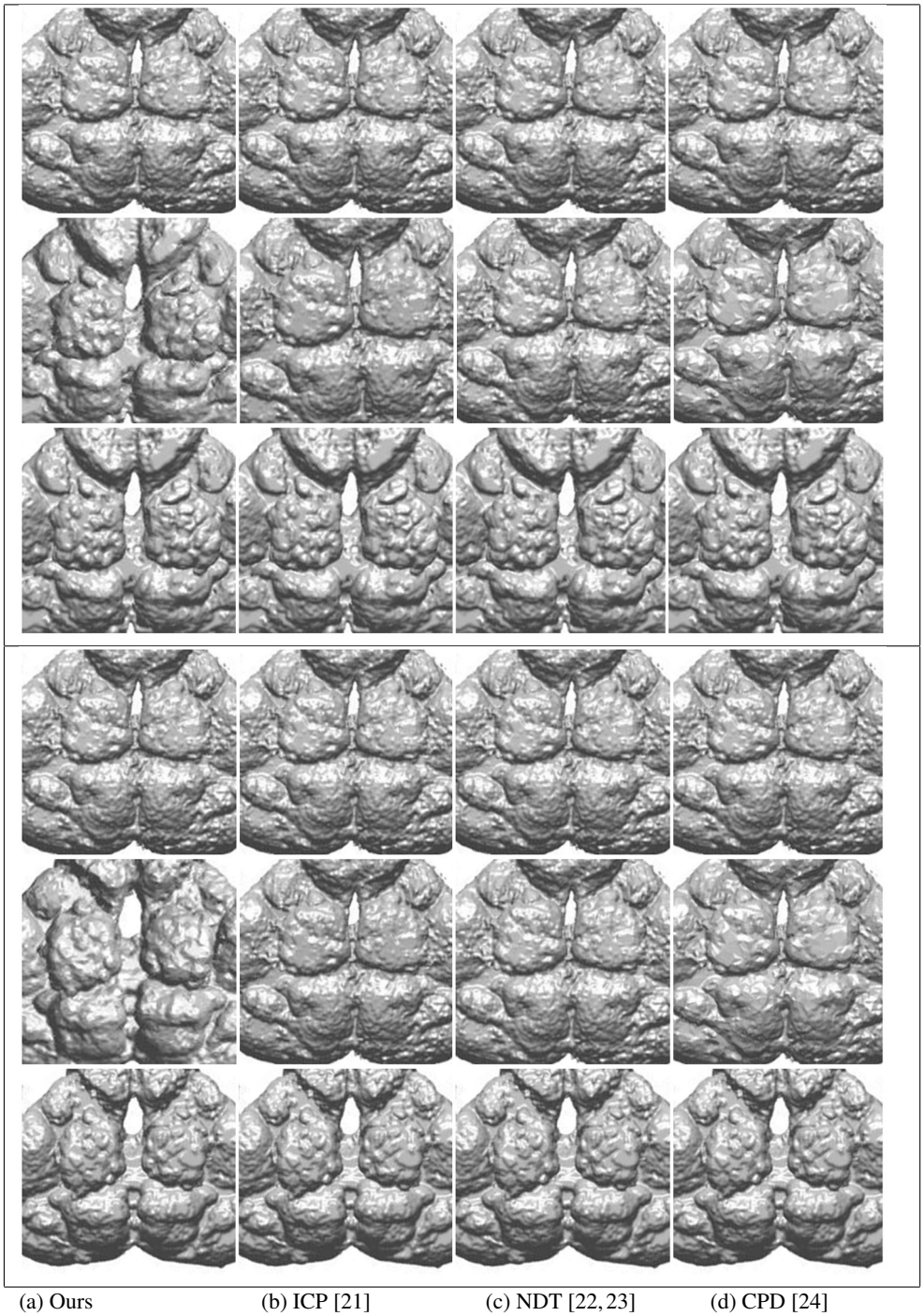
Fig. 10. Comparison between aligned source surface and target surface; (a) Registered and aligned source surface model; and (b) Target model. This figure shows the source model can be well aligned to the target.

shows zoom-in views of Figs. 9 (b) and (c). We can find that although the deformed surface is slightly rougher than the target one, the shape of the source model is well aligned to that of target surface.

Finally, to show the proposed method achieves better performance than conventional point cloud registration methods, such as iterative closest point registration (ICP), 3D normal distributions transform (NDT) and coherent point drift (CPD), we applied these methods to register point clouds $\mathbf{V}_i$ of aforementioned fly brain surface models $\mathcal{M}_i(\mathbf{V}_i, G_i)$ and then made comparisons. Illustrated in Fig. 11 are zoom-in views of aligned models derived by different methods. Sub-figures exhibited in the upper half of Fig. 11 are results of registering Brain-0 to Brain-1, whereas sub-figures in the lower half are those of registering Brain-0 to Brain-3. Similar to numerical values listed in Table 1, these images show our method is more advantageous than previous benchmark point cloud registration methods on fly brain datasets. Because it takes more than 74GB RAM to run the CPD algorithm on two full point clouds, each containing about $100,000$ vertices, we ran CPD on downsampled point clouds instead. Note that i) the reduced point clouds, each containing about $20,000$ vertices, for CPD experiments were derived by MATLAB built-in function **pcdownsample**, and ii) based on Fig. 7 (b), the mean surface-to-surface distance brought by point reduction is less than 0.25 voxels, and this value is neglectable compared with those shown in the CPD-column in Table 2.

Table 2. Comparison among different methods. All values are mean surface-to-surface distance in voxels.

	Our method		ICP		NDT [22, 23]		CPD [42]		w/o Registration	
	Mean	RMS	Mean	RMS	Mean	RMS	Mean	RMS	Mean	RMS
Surface 0	–	–	–	–	–	–	–	–	–	–
Surface 1	0.107	0.401	10.832	13.099	13.153	16.439	13.226	16.486	31.712	40.439
Surface 2	0.074	0.308	15.147	18.876	14.571	18.672	14.798	18.959	27.949	36.581
Surface 3	0.164	0.694	9.989	12.858	10.642	13.593	10.676	13.613	18.952	22.937
Surface 4	0.108	0.531	9.318	11.668	12.580	15.313	13.708	16.714	28.539	34.306



(a) Ours (b) ICP [21] (c) NDT [22, 23] (d) CPD [24]

Fig. 11. Comparison among registration results of different methods. **Upper half**: Zoom-in views of the experiment set of registering Brain-0 to Brain-1. **Lower half**: Zoom-in views of the experiment set of registering Brain-0 to Brain-3. For each experiment set, the first row shows source surface, the second row shows aligned source, and the third row shows the target.

5. CONCLUDING REMARKS

We propose in this paper a strategy to register 3D point clouds of *Drosophila* brain surface, topologically a torus, in a 2D parameterization domain. We regard the 3D point cloud registration problem as a surface registration issue, *i.e.*, a graph-assisted cloud registration task. Therefore, the proposed method integrates classical TPS-based method for 2D images and mesh processing techniques for 3D surface models. Through parameterization, a 3D registration issue can be solved by general 2D registration methods and no longer needs a global rigid-body transformation process. Our experiments demonstrate the effectiveness of the proposed method in registering dense point clouds depicting 3D anatomical structure of fly brain. Also, the proposed method can align shape details, including cortical textures on brain surface, and the mean surface-to-surface distance can be reduced to about 0.01% of bounding-box diagonal. The proposed method is effective.

REFERENCES

1. W. Huetteroth, B. E. Jundi, S. E. Jundi, and J. Schachtner, "3D-reconstructions and virtual 4D-visualization to study metamorphic brain development in the sphinx moth Manduca sexta," *Frontiers in Systems Neuroscience*, Vol. 4, 2010, p. 7.
2. R. Brandt, T. Rohlfing, J. Rybak, *et al*., "Three-dimensional average-shape atlas of the honeybee brain and its applications," *Journal of Comparative Neurology*, Vol. 492, 2005, pp. 1-19.
3. B. el Jundi, H. Basil, L. Stanley, *et al*., "The locust standard brain: a 3D standard of the central complex as a platform for neural network analysis," *Frontiers in Systems Neuroscience*, Vol. 3, 2010, p. 21.
4. T. Rohlfing, R. Brandt, C. R. Maurer Jr., and R. Menzel, "Bee brains, B-splines and computational democracy: generating an average shape atlas," in *Proceedings of IEEE Workshop on Mathematical Methods in Biomedical Image Analysis*, 2001, pp. 187-194.
5. A. Jenett, J. E. Schindelin, and M. Heisenberg, "The virtual insect brain protocol: creating and comparing standardized neuroanatomy," *BMC Bioinformatics*, Vol. 7, 2006, p. 544.
6. H.-C. Shao, C.-C. Wu, G.-Y. Chen, *et al*., "Developing a stereotypical *Drosophila* brain atlas," *IEEE Transactions on Biomedical Engineering*, Vol. 61, 2014, pp. 2848-2858.
7. H. Peng, P. Chung, F. Long, L. Qu, *et al*., "BrainAligner: 3D registration atlases of Drosophila brains," *Nature Methods*, Vol. 8, 2011, p. 493.
8. A. A. Joshi, D. W. Shattuck, P. M. Thompson, and R. M. Leahy, "Surface-constrained volumetric brain registration using harmonic mappings," *IEEE Transactions on Medical Imaging*, Vol. 26, 2007, pp. 1657-1669.
9. A. Klein, S. S. Ghosh, B. Avants, *et al*., "Evaluation of volume-based and surface-based brain image registration methods," *Neuroimage*, Vol. 51, 2010, pp. 214-220.
10. S. Chen, D. Tian, C. Feng, A. Vetro, and J. Kovačević, "Fast resampling of three-dimensional point clouds via graphs," *IEEE Transactions on Signal Processing*, Vol. 66, 2017, pp. 666-681.

11. W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3D surface construction algorithm," *ACM Siggraph Computer Graphics*, Vol. 21, 1987, pp. 163-169.
12. Y.-C. Chen, Y.-C. Chen, A.-S. Chiang, *et al*., "A reliable surface reconstruction system in biomedicine," *Computer methods and programs in biomedicine*, Vol. 86, 2007, pp. 141-152.
13. Z. Zheng, J. S. Lauritzen, E. Perlman, *et al*., "A complete electron microscopy volume of the brain of adult Drosophila melanogaster," *Cell*, Vol. 174, 2018, pp. 730-743.
14. R. Franconville, C. Beron, and V. Jayaraman, "Building a functional connectome of the Drosophila central complex," *Elife*, Vol. 7, 2018, p. e37017.
15. Y.-C. Huang, C.-T. Wang, T.-S. Su, *et al*., "A single-cell level and connectome-derived computational model of the Drosophila brain," *Frontiers in Neuroinformatics*, Vol. 12, 2019, p. 99.
16. C.-T. Shih, O. Sporns, S.-L. Yuan, *et al*., "Connectomics-based analysis of information flow in the Drosophila brain," *Current Biology*, Vol. 25, 2015, pp. 1249-1258.
17. H. C. Shao, W. Y. Cheng, Y. C. Chen, and W. L. Hwang, "Colored multi-neuron image processing for segmenting and tracing neural circuits," in *Proceedings of the 19th IEEE International Conference on Image Processing*, 2012, pp. 2025-2028.
18. A.-S. Chiang, C.-Y. Lin, C.-C. Chuang, *et al*., "Three-dimensional reconstruction of brain-wide wiring networks in Drosophila at single-cell resolution," *Current Biology*, Vol. 21, 2011, pp. 1-11.
19. H. C. Shao and W. L. Hwang, "Deriving 3D shape properties by using backward wavelet remesher," in *Proceeding of the 5th IEEE Global Conference on Signal and Information Processing*, 2017, pp. 191-195.
20. A. W. Toga, *Brain Warping*, Academic Press, 1998.
21. P. J. Bes, N. D. McKay, *et al*., "A method for registration of 3-D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, 1992, pp. 239-256.
22. P. Biber and W. Strasser, "The normal distributions transform: A new approach to laser scan matching," *Proceedings IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 3, 2003, pp. 2743-2748.
23. M. Magnusson, "The three-dimensional normal-distributions transform: an efficient representation for registration, surface analysis, and loop detection," Ph.D. dissertation, School of Science and Technology, Orebro Universitet, 2009.
24. A. Myronenko, X. Song, and M. A. Carreira-Perpinán, "Non-rigid point set registration: Coherent point drift," *Advances in Neural Information Processing Systems*, 2007, pp. 1009-1016.
25. F. L. Bookstein, "Principal warps: Thin-plate splines and the decomposition of deformations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, 1989, pp. 567-585.
26. K. Rohr, H. S. Stiehl, R. Sprengel, T. M. Buzug, J. Weese, and M. H. Kuhn, "Landmark-based elastic registration using approximating thin-plate splines," *IEEE Transactions on Medical Imaging*, Vol. 20, 2001, pp. 526-534.
27. H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understanding*, Vol. 89, 2003, pp. 114-141.

28. B. Combès and S. Prima, "A new efficient EM-ICP algorithm for non-linear registration of 3D point sets," *Computer Vision and Image Understanding*, Vol. 191, 2020, p. 102854.
29. D. Nain, S. Haker, A. Bobick, and A. Tannenbaum, "Multiscale 3-d shape representation and segmentation using spherical wavelets," *IEEE Transactions on Medical Imaging*, Vol. 26, 2007, pp. 598-618.
30. P. Schröder and W. Sweldens, "Spherical wavelets: Efficiently representing functions on the sphere," in *Proceedings of the 22nd ACM Annual Conference on Computer Graphics and Interactive Techniques*, 1995, pp. 161-172.
31. Amira, https://www.openinventor.com/en/about/user-testimonials/detail/amira-software for-life-sciences-thermo-fisher-scientific, 2010.
32. H. Hoppe, "Progressive meshes," in *Proceedings of the 23rd ACM Annual Conference on Computer Graphics and Interactive Techniques*, 1996, pp. 99-108.
33. M. Garland and P. S. Heckbert, "Surface simplification using quadric error metrics," in *Proceedings of the 24th ACM Annual Conference on Computer Graphics and Interactive Techniques*, 1997, pp. 209-216.
34. A. W. F. Lee, W. Sweldens, P. Schröder, L. Cowsar, and D. Dobkin, "MAPS: Multiresolution adaptive parameterization of surfaces," in *Proceedings of the 25th ACM Annual Conference on Computer Graphics and Interactive Techniques*, 1998, pp. 95-104.
35. "MeshLab," http://www.meshlab.net/.
36. "FlyCircuit," http://www.flycircuit.tw/.
37. C.-C. Wu, G. Y. Chen, Y.-C. Chen, H. C. Shao, C. C. Wu, H. M. Chang, A. S. Chiang, and Y. C. Chen, "Algorithm for the creation of the standard Drosophila brain model and its coordinate system," in *Proceedings of the 5th International Conference on Visual Information Engineering*, 2008, pp. 478-483.
38. C. Loop, "Smooth subdivision surfaces based on triangles," Master's Thesis, Department of Mathematics, University of Utah, 1987.
39. E. J. Stollnitz, T. D. DeRose, and D. H. Salesin, *Wavelets for Computer Graphics: Theory and Applications*, Morgan Kaufmann, 1996.
40. H. Chui and A. Rangarajan, "A new algorithm for non-rigid point matching," *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 2, 2000, pp. 44-51.
41. P. Cignoni, C. Rocchini, and R. Scopigno, "Metro: Measuring error on simplified surfaces," in *Proceedings of Computer Graphics Forum*, Vol. 17, 1998, pp. 167-174.
42. A. Myronenko and X. Song, "Point set registration: Coherent point drift," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 32, 2010, pp. 2262-2275.

Hao-Chiang Shao received his Ph.D. degree in Electrical Engineering from National Tsing Hua University, Taiwan, in 2012. He has been an Assistant Professor with the Department of Statistics and Information Science, Fu Jen Catholic University, Taiwan, since 2018. During 2012 to 2017, he was a Postdoctoral Researcher with the Institute of Information Science, Academia Sinica, involved in a series of *Drosophila* brain research projects; in 2017-2018, he was an R&D Engineer with the Computational Intelligence Technology Center, Industrial Technology Research Institute, Taiwan, taking charges of DNN-based automated optical inspection (AOI) projects. His research interests include 2D+Z image atlasing, 3D mesh processing, big industrial image data analysis, and machine learning.

Lu-Hung Hsu received his B.S. and M.S. degrees in Electrical Engineering from National Tsing Hua University, Taiwan, in 2010 and 2012, respectively. He is now an Engineer in MEDIATEC, Hsinchu, Taiwan.

Yung-Chang Chen received the B.S. and M.S. degrees in Electrical Engineering from the National Taiwan University, Taipei, Taiwan, in 1968 and 1970, respectively, and the Ph.D. (Dr.-Ing.) degree from the Technische Universitat Berlin, Berlin, Germany, in 1978. In 1978, he joined the Department of Electrical Engineering, National Tsing Hua University, Hsinchu, Taiwan, where he is currently a Professor. From 1980 to 1983, he was the Chair in the Department of Electrical Engineering, National Central University, Chungli, Taiwan. From 1992 to 1994, he was the Chair in the Department of Electrical Engineering, National Tsing Hua University. From 2002 to 2004, he was the Dean of the College of Engineering and a Professor in the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi, Taiwan. From 2009 to 2010, he was the Chair of Image Processing and Pattern Recognition Society. His research interests include multimedia signal processing, digital video processing, medical imaging, biomedical image processing, computer vision, and pattern recognition. He is an IEEE Life Fellow and now an Emeritus Professor in the Department of Electrical Engineering, National Tsing Hua University, Taiwan.

Ying-Chu Huang received her Master's degree from the Institute of Applied Statistics, Fu Jen Catholic University, Taiwan in 2019. Now she is a Deputy Engineer at Smart Channel & Logistics Department, Service Systems Technology Center, Industrial Technology Research Institute. Her research interests lie in machine learning and computer vision.

Shih-Ting Huang recieved her B.S. from the Department of Statistics and Information Science, Fu Jen Catholic University in 2019. She is currently working for Appier as an Operation Assistant in Campaign Management Team. Her research interest lies in object detection, bio-informatics and machine learning applications.