國立交通大學

資訊科學與工程研究所

博 士 論 文

腦神經結構重建以及發育分析

Neuronal structure analysis: structure reconstruction and dynamic analysis

研 究 生：李秉璋

指導教授：荊宇泰博士
江安世博士

中華民國一百零一年八月十四日

腦神經結構重建以及發育分析

Neuronal structure analysis: structure reconstruction and dynamic analysis

研　究　生：李秉璋　　Student: Ping-Chang Lee

指導教授：荊宇泰博士　　Advisor: Dr. Yu-Tai Ching

江安世博士　　Dr. Ann-Shyn Chiang

國立交通大學

資訊科學與工程研究所

博士論文

A Dissertation

Submitted to Institute of Computer Science and Engineering

College of Computer Science

National Chiao Tung University

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Philosophy

in

Computer Science

August 2012

Hsinchu, Taiwan, Republic of China

中華民國一百零一年八月

腦神經結構重建以及發育分析

學生：李秉璋　　指導教授：荊宇泰博士
江安世博士

國立交通大學資訊科學與工程研究所博士班

摘　　要

神經的網路結構一直是生物學家感興趣的研究課題。研究者相信神經間的拓樸結構可以和神經傳導訊息的傳遞有關，並且也和主宰了那些功能有關。透過近代的基因標定技術以及光學儀器，我們可以清楚的看到單一個帶有螢光表現的神經細胞，使得我們有機會做更進一步的神經結構分析；不論是單一神經型態或是神經網路的拓樸。但是經由光學儀器取得的原始影像資料經常是十分龐大的，會帶給後續的分析很大的困擾，為了解決這個問題，本論文的第一部分將把重點放在如何萃取出神經的中線。當我們把一個神經的結構從影像轉為點和線的結構後，檔案的大小將從原來的數百萬位元組降至幾萬位元組，這得以加速後續的分析。我們同時對提出的方法進行強韌度及準確率測試。結果顯示我們的方法是可靠的。另外，我們也提出一個基於萃取出的中線結構的應用以證明這些萃取出來的中線可用於更一進步的結構分析。神經發育是另一個讓許多生物學家及神經科學家投入精力的課題。透過觀察神經的發育，我們可以理解究竟是甚麼樣的物質及環境使得神經退化或加速發育。神經的中線結構是科學家們觀察的神經發育的樣本，但傳統上為了觀察發育的現象只能靠生物學家曠日廢時的以人工標定每個細微處的變化，這大大的延宕了實驗的進度。本論文的第二部分提出一個監督式的半自動互動系統 (4D SPA)。我們實驗記錄了在 4D SPA 的輔助下，使用者需要花多少時間完成配對標定的工作，同時也採取了兩種在圖形辨識常用的策略來自動標定配對，並記錄了計算所需的時間。透過比較兩種不同方法，我們確認了相較於傳統的手動標定，4D SPA 使得標定神經發育這件工作更省時，相比於自動化的方法，我們的方法不僅省時也更穩健。

Neuronal structure analysis: structure reconstruction and dynamic analysis

Student: Ping-Chang Lee Advisor: Dr. Yu-Tai Ching
Dr. Ann-Shyn Chiang

Institute of Computer Science and Engineering
College of Computer Science
National Chiao Tung University

ABSTRACT

Biologists and neurologists have been interested in neural networks for a long time. They believe the network topology is highly related to the neural functions. Modern biotechnology and optical apparatus make it possible to observe a single neuron in micron scale. This gives us the chance to understand the neural circuits and neuronal morphology more. However, the size of the original image data is huge and this largely obstruct the further analysis. Hence, in the first part of this dissertation, we focus on the issue about how to reconstruct the 3D neuronal structures to sets of points and lines. When the data are reduced to points and lines, the data size becomes smaller, from hundreds megabytes to some megabytes. The robustness and accuracy test were performed and the results showed the proposed method is reliable. An application of the reconstructed results was also demonstrated. This showed the further analysis based on the reconstructed results is possible. In conclusion, the proposed method facilitates researchers to extend their study to the higher level of neural structures and the neural networks. The study of neuronal development is also a topic that attracts many biologists and neurologists. We can find the key factors that influence the neuronal development, neuronal dynamics, if we are firstly able to observe and quantify the development. The reconstructed 3D line structure is one of the popular neuronal representations used in the study of neuronal development. Nevertheless, conventionally, every tiny detail of the alternation of the neuron at two different time points is labelled by human experts manually. This extremely postpone the progress of this study. In the second part of this dissertation, we present a supervised 4D neuronal Structural Plasticity Analysis (4D SPA) computer method that helps the researchers to explore the

neuronal dynamics. The processing time for the user to complete the matching task was recorded; moreover, the time for two automatic methods to compute the matching results were recorded respectively, too. These records showed that with the help of 4D SPA, the consuming time for the exploration is greatly reduced, and furthermore the proposed method is more reliable.

Acknowledgement

It was a long journey to complete the Ph.D. program. Without the help from my family, friends and many kind co-workers, I doubt whether or not I could have made it.

First of all, I would like to thank to my advisors, Dr. Yu-Tai Ching and Dr. Ann-Shyn Chiang. For the past 6 years, both Dr. Ching and Dr. Chiang taught me a lot and gave me great opportunities to enlarge my horizon.

For the first part of my research, I would like to thank the Brain Research Center, NTHU for all the data; thank Dr. HM. Chang for his valuable suggestion; and thank Mr. Chao-Chun Chuang for the collaboration. For the second part of my research, I would like to thank Cold Spring Harbor Lab, Dr. Hollis T. Cline, Dr. Jeniffer E. Bestman, Dr. Hai-yan He, and Dr. Jian-li Li for their great suggestions and data support. Without them, it would have been impossible to complete this research.

Many thanks to my parents and my brother. There is no doubt that they give me the most support. Regardless of when or where I am, I know they will always back me up.

I would like to thank many past and current co-workers, 杰翰, 鴨老大, 大蛙教練, 阿北, 蟲先生, Yway, Rhys, 昆布, 猴捷, 科科, 小翁, Akito, 彥淳, 昌杰, 大威, 柏淳 and 炳輝. We have many valuable discussions and they also enriched my lab experiences.

Last but not least, I would like to thank a special person who ran into my life as the most wonderful surprise I have ever had. Thank you, Queena. At the final days of the program, your smile and inspiring words comforted me and cherished me.

Contents

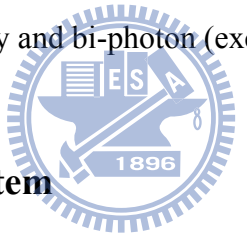
List of Tables

List of Figures

Chapter 1 Introduction

1.1 Background

Biomedical imaging technique has rapidly progressed in recent years. It is impossible that only the more and more sophisticated microscope that makes this happened. The combination of the discovery of *green fluorescent protein* (GFP), the maturity of transgenic technology, and the more sophisticated microscopy make today's enormously progress. Contemporary neurologist and biologist modify the gene of their observing target that cause the organ express the specific function and fluoresce at the same time. At the meanwhile, by the help of the confocal microscope, the tomograph of the fluorescing region can be derived. Below, we briefly introduce the the UAS-GAL4 system, labeling tachnique, confocal laser scanning microscopy and bi-photon (excitation) microscopy.



1.1.1 The UAS-GAL4 system

The UAS-GAL4 system [1] is a powerful technique for studying gene expression. The system has two parts: the GAL4 gene, encoding the yeast transcription activator protein GAL4, and the UAS (Upstream Activation Sequence), a short section of the promoter region, to which GAL4 specifically binds to activate gene transcription. The responder, the gene 'X' to be activated or over expressed, is under the control of UAS sequence. In order to activate transcription, the UAS gene 'X' carrier need to cross to the GAL4 gene carrier which will express in particular cells, then the resulting progeny will over express gene 'X'. Usually the GAL4 gene is placed under the driver gene, while the UAS controls expression of a target gene. Then, GAL4 is only expressed in cells where the driver gene is usually active. As a consequence, GAL4 should only activate gene transcription where a UAS has been introduced. For example, by fusing a gene encoding the GFP the expression pattern of the driver genes can be determined. An additional strength of the system arises from the ability to target expression of any responder in a variety of spatial and temporal fashions by mating it with distinct GAL4 drivers. So far the UAS-GAL4 system has been

applied on many species to study gene expression in organisms, such as fruit fly *Drosopila*, African clawed frog *Xenopus* and zebrafish.

1.1.2 Single-neuron labeling

There are many ways to visualize individual neuron with their native environment. In the 19th century Dr. Golgi had developed the classic Golgi staining method which was later largely extent by Dr. Ramón y Cajal. Both methods are still widely used today. From then on, many methods to observe single neuron have been developed including the genetic methods developed in current era. Single-neuron labeling using genetic methods can be divided into three categories: (1) using highly specific promoters, or insertions of transgenes in chance, to limit marker expression to a very small subset of isolated neuron; (2) using site-specific recombination within the same piece of DNA, such as ”flip-out” in *Drosophila* to limit marker expression in isolated neurons; and (3) using mitotic recombination to couple maker expression cell division. For more details about the Single-neuron labeling methods, please refer to Yuste et al, ch. 13 [2]. Labeling single neuron in their native environment can be used for the following applications:

- To trace the axonal projection and dendritic elaboration patterns of individual neurons.
- To study molecular mechanisms of dendritic and axonal development and plasticity with high anatomical resolution.
- To study the physiological functions of identified neurons in brain slice or *in vivo*.

Here, we would like to claim in advance. Our study object, neuron, in this dissertation, if without specification it means the single-neuron.

1.1.3 Confocal microscopy

Confocal laser scanning microscopy is a technique for obtaining high-resolution optical images with depth selectivity. The key feature of confocal microscopy is its ability to

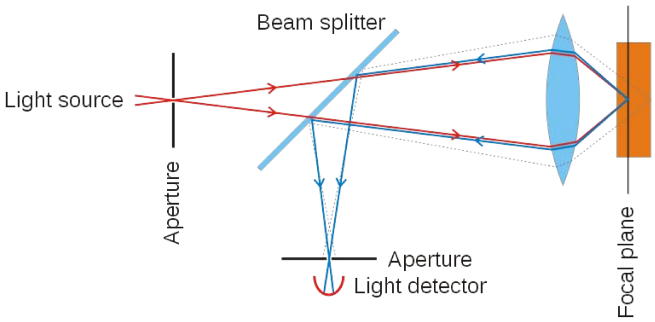


Figure 1.1: Principle of confocal microscopy. From *Wikipedia*.

acquire in-focus images from selected depths by using a spatial pinhole to eliminate out-of-focus light in specimens that are thicker than the focal plane. In a confocal laser scanning microscope, a laser beam passes through a light source aperture and then is focused by an objective lens into a small (ideally diffraction limited) focal volume within or on the surface of a specimen. A beam splitter separates off some portions of the light into the detection apparatus, which in fluorescence confocal microscopy will also have a filter that selectively passes the fluorescent wavelengths while blocking the original excitation wavelength (Fig. 1.1). After passing a pinhole, the light intensity is detected by a photodetection device, transforming the light signal into an electrical one that is recorded by a computer.

1.1.4 Bi-photon confocal microscopy

Multi-photon fluorescence microscopy has similarities to confocal laser scanning microscopy. Both use focused laser beams scanned in a raster pattern to generate images, and both have an optical sectioning effect. Unlike confocal microscopes, multi-photon microscopes do not contain pinhole apertures, which give confocal microscopes their optical sectioning quality. The optical sectioning produced by multi-photon microscopes is a result of the point spread function formed where the pulsed laser beams coincide. Two-photon excitation microscopy is a fluorescence imaging technique that allows imaging living tissue up to a depth of one millimeter. Two-photon excitation can be a superior alternative to confocal microscopy due to its deeper tissue penetration, efficient light detection and re-

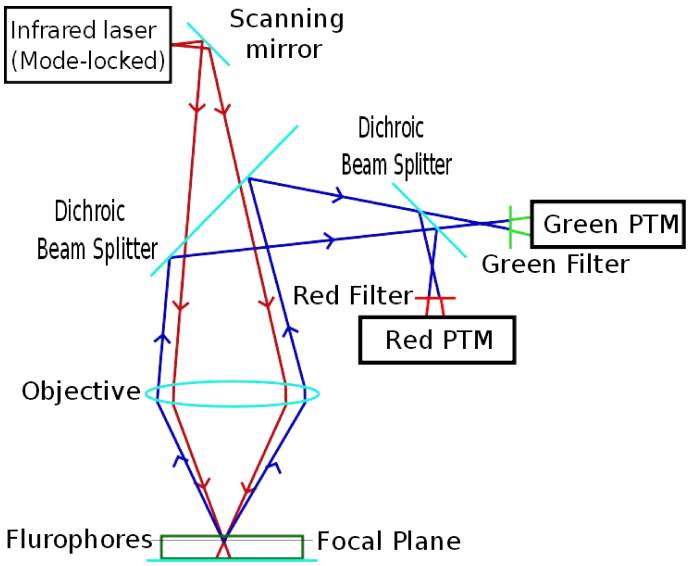


Figure 1.2: Scheme of Two-photon excitation microscopy. From *Wikipedia*.

duced phototoxicity. The most commonly used fluorophores have excitation spectra in the 400-500 nm range, whereas the laser used to excite the fluorophores lies in the 700-1000 nm (infrared) range. If the fluorophore absorbs two infrared photons simultaneously, it will absorb enough energy to be raised into the excited state. The fluorophore will then emit a single photon with a wavelength that depends on the type of fluorophore used. The use of infrared light to excite fluorophores in light-scattering tissue has added benefits. Longer wavelengths are scattered to a lesser degree than shorter ones, which is a benefit to high-resolution imaging. Figure 1.2 shows a brief scheme of Two-photon microscope.

1.2 Motivation

In neuroscience, there still are many questions that we cannot answer so far. However, we could have deeper understandings of them when we are able to derive the single neuron image. Below are two problems motivating me starting this research.

Neural networks. Neurons in animal brain form numerous distinct functional circuits. These circuits mediate the fundamental processes of sensation and motion. A combination of fundamental processes makes animals able to have complex actions. However,

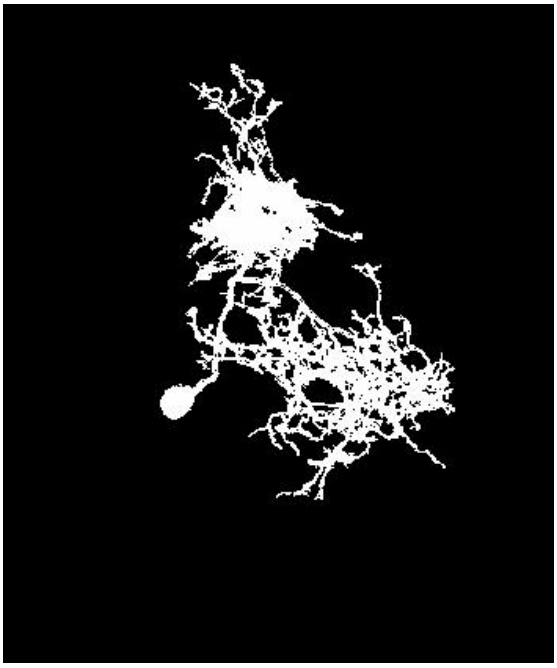


Figure 1.3: A MIP (Maximum Intensity Projection) neuron image. Even for a human expert, it is hard to reconstruct the neuronal structure manually.

the whole network connected by neurons and the message pathway are still not totally understood.

Neuronal development. Nerve cells reflect a highly complicated evolution governed by both intrinsic and extrinsic factors. In addition, growing cells influence one another spatially and physically. It is the interaction through time and space determines the evolution of neural shape, including synaptogenesis. But now, our understanding of dendritic development is still far from satisfactory.

Theoretically speaking, if an atlas of a brain is constructed and the information about a single neuron is acquirable, possibly we can answer the *Neural Network* question. Besides, the ability to derive images of a single neuron at different time point *in vivo* enable scientists to understand neuronal development further. However, performing quantitative analysis and exploring further information from image data are extremely time consuming. So mostly after a 3D neuron image volume is taken by either confocal microscope or bi-photon microscope, neurologist manually trace out the whole neuron structure via

helps of software. It works fine when the structure of the neuron is simple yet in many cases the neuron structure is complicate (Fig. 1.3). Even for a well-trained expert, it is hard to reconstruct the complicate neuronal structure. An efficient tool to reconstruct the neuron is needed. Nevertheless, although we have a method to reconstruct the neuronal structure efficiently, manually analyzing the development of a single neuron is another tedious work since the neuronal dynamics is complicated. It calls for a computer method to help scientist to speed up the process.

1.3 Related Studies

Drosophila melanogastor and *Xenopus laevis* are two commonly used model organisms. In our study, we used *Drosophila melanogster* neuron image data to develop a high-throughput computer method and the image sequences of *Xenopus laevis tadpole* taken *in vivo* were used to develop a computer aided system for 4D neuronal dynamic analysis. Later in this dissertation, *Drosophila melanogastor* is abbreviated as *Drosophila* and *Xenopus* means *Xenopus laevis*. Related studies on neuronal structures and neuronal dynamics are introduced in the following paragraphs.

1.3.1 Studies on 3D neuronal structure reconstruction

A single neuron in the *Drosophila* brain can be labeled by a green fluorescent protein [3]. Using the focus clear technique [4], one can employ a confocal microscope to acquire a clear whole brain image stack containing a labeled single neuron. This enables reconstructing the structure of neurons and studying the neural network of the fruit fly brain. Approximately 100,000 neurons are in the *Drosophila* brain.

Tracing neuron fibers is similar to tracking vasculature line structures in a 3D image volume. Previous studies on medical image processing presented related methods of tracking line structures. Bouix et al. proposed a method based on skeletonization and branch analysis [6]. Other approaches include the method based on enhancing line or edge properties and then chaining up the most likely pixels [7] and the method of attempting to find the

minimal paths [8]- [10]. Compared with medical images of blood vessels, neuron images often suffer from noises and uneven resolution in the x, y, and z directions, and a single neuron is usually discontinuous in the image stack. Directly applying those methods to reconstruct the neuronal structures is inadequate. Researchers have recently proposed methods to trace neurons or reconstruct the neuronal structure. Al-Kofahi et al. progressively fitted and matched the primitive template structures, spheres, ellipsoids, and cylinders in the image stack [11]. However, this method did not address the situation that a neuron is fragmented in an image space. Similar to [11], Zhao et al. investigated the morphological characteristics of neurons [12]. Both the Al-Kofahi and Zhao methods assumed that neuron fibers are spherical, ellipsoidal, or cylindrical, but usually neuron fibers in the image stack are not as regular as this assumption. Zhang et al. assembled many skeleton structures as a single neuronal structure [13]. However, this method only considered 2D neuron images. Peng et al. reduced the tracing problem as a variational problem by finding the geodesic shortest path [14] . T$\ddot{u}$retken et al. proposed a method based on optimizing a carefully designed energy function [19]. Nevertheless, none of these methods are specific for processing numerous sets of volume data automatically.

In our study, we present a high-throughput computer method of reconstructing the neuronal structure of the fruit fly brain. The design philosophy of the proposed method differs from those of previous methods. We proposed to compute 2D skeletons of a neuron in each slice of the image stack. The 3D neuronal structure was then established from the 2D skeletons.

1.3.2 Studies on 4D neuronal dynamics

Differences in neuronal structures that occur naturally, for instance during development, aging, as a result of circuit plasticity or disease, or under experimental conditions have been widely studied in an effort to understand how changes in neuronal structure affect neuronal and circuit function [16]- [18]. The technical advancements in *in vivo* neuronal labeling, imaging and acquisition of time-lapse data of changes in neuronal structure over time, which have been greatly facilitated by the development of *in vivo* microscopy techniques, such as confocal and multi-photon laser scanning, have demonstrated that dy-

namic changes in neuronal structure can occur over the time-course of minutes to days to months [17] [19]. Furthermore characterization of structural dynamics in neuronal arbors can often reveal fundamental mechanisms that underlie the difference between the end points [16]. Although 3 dimensional reconstruction of neuronal structure can be done with computer assistance, analysis of 4D image data sets remains a great challenge because of the difficulty of comparing two complex 3D neuronal arbors. Most 4D analysis of neuronal structural dynamics from time-lapse imaging data is done by manual comparison of 2D or 3D reconstructions. To analyze the changes between two 3D reconstructions of neurons manually takes an expert about 1-5 hours to align and match the two reconstructed 3D complete dendritic arbor structures. Manual identification of the numbers and distribution of dynamic branches, categorized as retracted, newly added, transient, and maintained, over a set of 3-5 images [20]- [22] is laborious and greatly slows down research in the field [16]. Importantly, manual analysis may also increase subjective errors and discrepancies between data from different individual researchers. A computer method that assists in comparing 3D neuronal structures would address the weaknesses of manual analysis. Comparing neuronal structures requires quantification of features of the neuronal morphology. Several parameters, including the total surface area, the volume, branch order, the number of bifurcations, the summed length of the neuronal branches , the topological asymmetry of the dendritic arbor, and the distribution of the tips, have been employed for this purpose.

As data demonstrating importance of the spatial segregation of inputs for synaptic integration and neuronal responses increase [23]- [25], it is clear that identification and quantification of the spatial distribution of dynamic branches within dendritic arbors will be valuable for understanding mechanisms governing neuronal plasticity. Recent work reported computer-assisted automatic analysis of time-lapse images could be achieved in cultured neurons [26]. This advance was facilitated by the 2D structure of cultured neurons and their relatively simple neuronal morphology.

To analyze detailed changes of 3D neuronal structure is a more difficult task, for instance, because cumulative changes in the locations of individual branches can occur as a result of modest 3D shifts in positions of lower order branches, or because minor differences in the position of the animal during *in vivo* imaging may shift the orientation of the neuron

in the image. An intuitive approach is to align the structure of a neuron from two time-points to compute the differences in structure. However, the growth speed of a neuron is not isotropic; furthermore, the neuronal shape and morphology usually change a lot from hours to hours in different stages. These two factors make the intuitive method fail. Another approach is to employ the graph algorithm technique; reconstructed neuronal structures from two time-points are used to generate two neuronal trees in 3 dimensions. Heumann et al [27] model the neuronal cells as labeled trees and use the tree-edit-distance as a measurement to compare two neurons however their work focuses only on 3D shape analysis.

In our study, we present a supervised 4D neuronal Structural Plasticity Analysis (4D SPA) computer method that computes precise changes in the positions and lengths of all neuronal branches in the arbor between two images or time-points and presents the data as an image superimposed on the 3D reconstruction of the neuron. We designed our method taking advantage of both computer algorithm and human expertise. With some help from human supervision, we were able to significantly reduce the processing time and increase the accuracy.

The remainder of this dissertation is organized as follows: the neuron reconstruction method is detailed in Chapter 2; A complete description of 4D SPA is presented in Chapter 3; Finally the conclusion is stated in Chapter 4.

Chapter 2 High-Throughput Computer Method for 3D Neuronal Structure Reconstruction

Drosophila melanogaster is a well-studied model organism, especially in the field of neurophysiology and neural circuits. Neurons in a fruit fly brain form numerous distinct functional circuits as in the mammalian brain. These circuits mediate the fundamental processes of vision, olfaction, locomotion, flight navigation, and complex behaviors such as feeding, learning, and memory. Neurotransmitters and molecular mechanisms mediating these behaviors and activities are highly similar to the higher organisms and humans. Although the brain of the *Drosophila* is small but complex and the image of a single neuron in the brain can be acquired using confocal microscopy. It is worth studying fruit fly brain as the initial step to understand functions of the neural network. Thus, in this study, we focus on reconstructing the neuron in *Drosophila* brain.

This study presents a high-throughput computer method of reconstructing the neuronal structure of the fruit fly brain. The design philosophy of the proposed method differs from those of previous methods. We proposed to compute 2D skeletons of a neuron in each slice of the image stack. The 3D neuronal structure was then established from the 2D skeletons. Biologists tend to use confocal microscopes for optimal images in a slice for human visualization. Furthermore, images in two consecutive slices contain overlapped information. Consequently, a spherical object becomes oval in the image stack; that is, neurons in the image stack do not reflect the true shape of the neuron. This is the main reason we chose not to work on the 3D volume directly.

The proposed method comprises two steps. The first is the image processing step, which involves computing a set of voxels that is a superset of the 3D centerlines of the neuron. The shortest path graph algorithm then computes the centerlines. The proposed method was applied to process more than 15,000 neurons. Using the large amount of reconstructions, this study also demonstrated a result derived from the reconstructed data by using the clustering technique.

The remainder of this chapter is organized as follows: The Methods section details the proposed method; The Results section presents the tracing results, reconstructed neuronal structures, an application using the reconstructed results and using the Olfactory Projection Fibers from the DIADEM test data set [28] to evaluate the accuracy of the proposed method; finally, we have a discussion in the Discussion section. Each image in the DIADEM test data set contains original image stacks and gold reconstructions created by experts. This study defines the distance between one reconstruction to the other, to evaluate the accuracy of the reconstruction. The Discussion section shows the accuracy analysis and robustness test.

2.1 Methods

2.1.1 Image acquisition

In this section , we briefly state the technique to label a single neuron for observing neurons using confocal microscopy [4]. A tiling heat-shock protocol in MARCM labeling was adapted to cover most neurons born at different times. Animals were kept in a 37°C water bath for 3 min to 60 min, depending on the Gal4 driver used, with 50% overlapping periods throughout the entire development from embryo to adult eclosion. In each case, GFP expression was controlled by a specific Gal4 driver with expression dependent on the stochastic removal of a *Gal80* repressor by heat-shock-induced expression of a flippase protein during mitotic recombination at cell birth. The Gal4 lines were driven by the promoter of an essential protein for synthesis or processing of one of the following neurotransmitters: acetylcholine (*Cha-Gal4*), dopamine (*TH-Gal4*), GABA (*Gad1-Gal4*), glutamate (*VGlut-Gal4*), octopamine (*Tdc2-Gal4*), or serotonin (*Trh-Gal4*). Thus, an individual neuron of putative birth time and neurotransmitter type was labeled.

Sample brains were imaged using a Zeiss LSM 510 confocal microscope with a $20\times$ C-Apochromat water-immersion objective lens (N.A. value 1.2, working distance 220 μm). The following settings were used in image acquisition: scanning speed 7, resolution 1024 $\times$ 1024, line average four times, zoom 0.7, and optical slice 1 μm for $20\times$ objectives. The

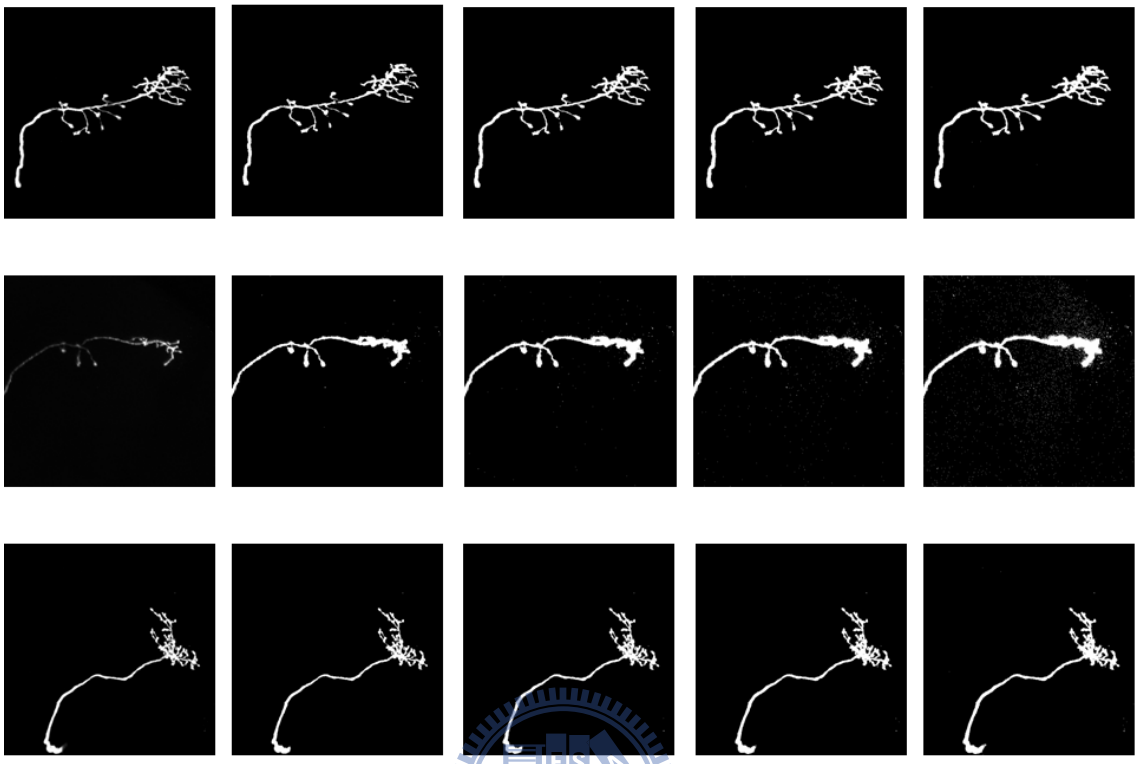


Figure 2.1: Three data of Olfactory Projection Neuron in the DIADEM data set are used to demonstrate the binarization method. The neurons are rendered by the maximum intensity projection (MIP) method. Every row shows a neuron undergoes different levels of binarization. From left to right, they are original neuron image followed by the binarized results. From the second column to the last column, we kept 50%, 60%, 70%, and 80% of the brightest visible voxels where the visible voxels are voxels having gray scale above 10. In the proposed method, we keep 70% of the brightest visible voxels which are shown in column 4.

voxel size of $x : y : z$ is $0.33 \times 0.33 \times 1\mu m$. The resolutions of the image stacks in the Olfactory Projection Fibers of the DIADEM data set were the same. All the data except those provided in the DIADEM data set were acquired by neural biologists in the Brain Research Center, National Tsing Hua University, Hsinchu, Taiwan.

2.1.2 Image preprocessing

A heuristic method was first used to binarize the volume data of an image stack containing a single neuron. The heuristic method was designed based on the observation that biologists tend to set up a confocal microscope for optimal human visualization of the neuron in a slice. The heuristic approach in this study segments the neuron by keeping 70% of the brightest "visible" voxels. An 8-bit grayscale voxel is "visible" if it has an intensity above 10. This method enables keeping sufficient voxels on the neuron and keeping the amount of noise small. Each row in Fig. 2.1 shows the images of a neuron undergoing different levels of binarization. The first column shows the original images, and the fourth column shows the results obtained by the proposed heuristic. Because the segmented result could still contain sparse noises, we remove noises by eliminating small-sized connected components in the volume. The 2D connected component analysis was applied to each slice first. All 2D 8-neighbor connected components consisting of less than 9 pixels were removed. The 3D 26-neighbor connected component analysis was then applied to the volume to remove all 3D 26-neighbor connected components consisting of less than 30 voxels. After removing the small connected components, the 2D morphological closing operator was applied to each slice in the image stack to smoothen the boundary of the neuron. This step is necessary; otherwise, the step computing the 2D skeletons produces unwanted small branches for rough boundary components.

The 2D skeletons of each slice were computed before reconstructing the 3D neuronal structure. Let B be the volume containing the binarized neuron and denote the set of non-zero voxels by B_N. The 2D Euclidean distance transform was applied to each image slice in B, and the 2D skeletons for each connected component were computed based on the transformed result [29]. We denote the set of points on the 2D skeletons by Q.

2.1.3 Fragmental line structures assembling

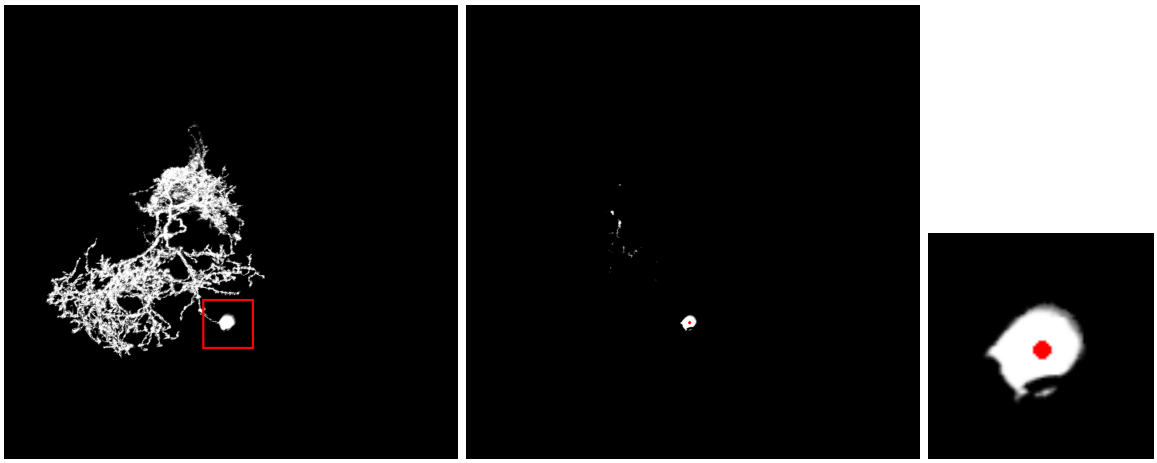
Because GFP labels only one neuron, Q should form a single 3D connected component. If Q does not form a 3D connected component (i.e., broken branches exist), then the Minimum Spanning Tree (MST) technique is applied to compute the connected component.

A weighted graph was constructed based on the 3D 26-neighbor connected components. Each vertex in this weighted graph represents a 3D connected component. An edge was present between a pair of vertices if the Euclidean distance in the image space between the closest points in two connected components was less than 5% of the largest image dimension. The edge weight is the distance between the pair of closest points. The MST of the graph was then computed using the Kruskal algorithm. Once an edge was selected during MST construction, points along the edge were sampled to make the distance between a pair of consecutive points to equal approximately 1. These sampled points were then added to the set Q. However, more than one connected component might be present when the process terminates. In this case, only the largest connected component is kept, and all the others are removed from Q. For each point in Q, we identify if it is a candidate 3D end point by examining nine digital planes in the 26-neighborhood [30]. The set of candidate 3D end points is denoted by V_E. The next step reconstructs the 3D neuronal structure from Q.

2.1.4 Reconstruction of the neuron branches

Another weighted graph was constructed from the point set Q. The shortest path algorithm employed in this study is the single-source shortest path algorithm. Because a source point in the graph should be given, this study defines the center of the soma as the source. In the image, the soma is a set of high-intensity voxels forming a spherical object. Geometrically, the Chebyshev center of a set is the point within the set that is the farthest from the boundary ([31] Ch. 8). An approximated Chebyshev center for the center of the soma served as the source vertex in the graph.

A good approximation for the Chebyshev center of the soma is a point in Q that is farthest from the boundary. The approximate Chebyshev center is computed iteratively as follows: For each point p in Q, we iteratively increase the radius, r_p, of the sphere centered at p until the sphere cannot enclose points totally in B_N. The center of the soma is the point c in Q that admits the largest sphere enclosing points totally in B_N and the largest sum of the intensity. Figure 2.2 shows an example of soma detection.



(a) (b) (c)

Figure 2.2: An example of the soma detection. (a) The MIP image of the original image stack. (b) Red dot indicates the center of the soma calculated by the soma detection procedure. (c) A close view of the detected soma position.

(a) (b)

Figure 2.3: The black nodes represent the points in Q

2.1.5 Reconstruction of the neuron branches

To push the 3D centerline toward the true center of the neuron, it is important to identify the candidate branch points. A 2D skeleton point is a candidate branch point if it has four or five neighboring points in Q and the arrangement of four is isomorphic to one of the patterns shown in Fig. 2.3. The set of candidate branch points is denoted by *Bps2D*.

A weighted graph $G = (V, E)$ was constructed from Q. V is a set of vertices where each vertex is a point in Q. E is a set of edges (p, q), where p and q are in Q and neighboring

to each other in the volume. The cost associated with the edge (p, q) is

$$f(p, q) = e^{-w(p,q)}. \tag{2.1}$$

where $w(p, q)$ is as shown in (2.2)

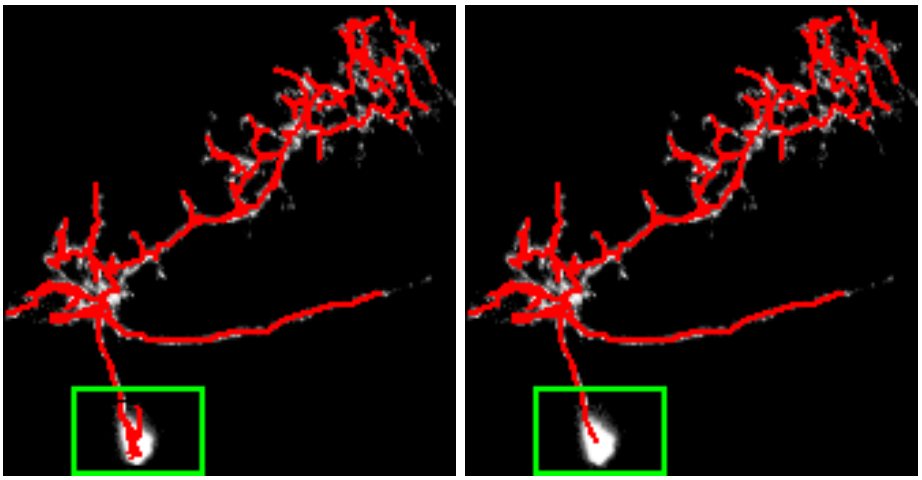
$$w(p, q) = w_d + w_b. \tag{2.2}$$

In (2.2), w_d is the Euclidean distance between p and q in the image space. The traced branches should intersect at a single branch point where a visual bifurcation point is present. To meet this purpose, an edge close to a candidate branch point has a large value for w_b. Let b_p be a candidate branch point. Consider the sphere centered at b_p with a radius $R = 1.0\mu m$. If (p, q) is enclosed in the sphere, then w_b of (p, q) is $\eta - d$, where $d = min\{||\overline{p, b_p}||, ||\overline{q, b_p}||\}$. Otherwise, $w_b = 0$. R is set to be $1.0\mu m$. This is because a sphere of the radius of $1.0\mu m$ is usually totally enclosed in the neuron in the bifurcation region. Because the resolutions in the x-,y-, and z-directions are $0.33\mu m$, $0.33\mu m$, and $1.0\mu m$, respectively, a $6 \times 6 \times 2$ box was used to approximate the sphere. To guarantee that w_b is positive, set $\eta = 10$.

In (2.2), the edges close to a branching point have a large $w_b = \eta - d$. Thus, these edges have a heavy weight $w(p, q)$ and light cost $f(p, q)$. When the shortest path algorithm is applied, these edges tend to become a part of the shortest path; thus, the constructed shortest paths tend to keep the appropriate branch points of the neuron branches.

Given the weighted graph and the source vertex c, the shortest paths from c to all the points in V_E can be computed by applying the Dijkstra algorithm. Each path from c to a vertex in V_E is called a branch. A branch should be removed if the ratio between its length and the length of the longest branch is less than 0.2. Such short branches are usually branches in the interior of the soma. The short branches enclosed by the green box in Fig. 2.4(a) should be removed. These branches in the soma are not really neuron fibers. The image on the right shows the result of removing short branches.

The neuronal structure can be reconstructed by iteratively selecting a branch obtained in



(a) (b)

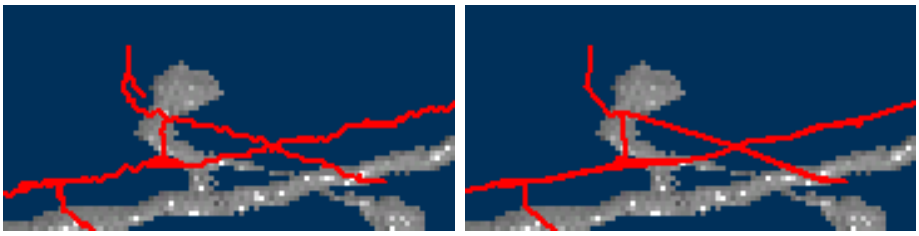
Figure 2.4: (a) There are short branches inside the soma. These short branches are produced because the soma is not perfectly spherical. These branches should be removed. (b) The traced result with all short branches in soma removed.

the previous step, followed by post-processing described in the following steps. Iterations stop when V_E is empty. A list (*BpList*) that stores the branch points formed during the reconstruction process is required.

1. Select the longest branch, P, which is a branch from c to a point p in V_E. Remove p from V_E. If *BpList* is not empty, compute the physical distances between the points on P and the points in *BpList*. Along the path from p to c, let b be the first point, such that $min_{q \in BpList} ||\overline{b,q}|| \leq 1.0 \mu m$. In this case, b is considered the same as q. Furthermore, P can be divided into two paths: from c to b, and from b to p. The path from c to q is a subpath of a previously reconstructed path, P'. In this case, the branch P is modified to be the path from c to q of P', and b in the path from b to p is replaced by q. If no such pair exists, look for another pair (b', q'), in which q' is from *Bps2D* and b' is from the path along p to c. Let b' be the first point, such that $min_{q' \in Bps2D} ||\overline{b', q'}|| \leq 0.75 \mu m$, and replace b' by q' on the branch.

2. If P shares a common subpath from c to b_p with a previously reconstructed branch and b_p is not in *BpList*, then b_p is inserted into *BpList*.

3. Because skeletons were computed in 2D slices, false candidate 3D end points exist.



(a) (b)

Figure 2.5: A zoom-in view of the traced result (a) Red lines show the tracing result; (b)The ϵ-approximation of the tracing result where $\epsilon = \sqrt{3}$.

This study considers the candidate 3D end points close to P to be false end points, and removes the branch if the distance between its end points and P is less than $\bar{r}$, where $\bar{r}$ is the average of r_p, $p \in Q$.

2.1.6 Polygonal path approximation

The 3D centerline obtained by applying the shortest path algorithm was not smooth because the graph is a grid graph (Fig. 2.5(a)). To construct smooth centerlines, an ϵ-approximated polygonal path was calculated to approximate the centerline computed by the shortest path algorithm. An ϵ-approximation of a polygonal path has fewer points on the path and within a small deviation, ϵ, from the original polygonal path (Fig. 2.5(b)). Given a polygonal path $S =< v_0, \ldots, v_m >$ and an error bound ϵ, we look for a polygonal path,$\tilde{S}$ that is an ϵ-approximation of S. $\tilde{S} =< u_, \ldots, u_m >$ optimally ϵ-approximates S if $\tilde{S}$ meets the following criteria.

1. Vertex set of $\tilde{S}$ is a subset of S.
2. Let $u_i = v_j$ and $u_{i+1} = v_k, i = 1, \ldots, m-1$, the distance between any vertex on the polygonal path $< v_j, \ldots, v_k >$ and the line segment $< u_i, u_{i+1} >$ is less than ϵ.
3. The number of the vertices on $\tilde{S}$ is the smallest possible.

This problem can be solved using the dynamic programming technique. We define the number of edges on $\tilde{S}(i, j)$ to be its cost. The lowest cost among all the ϵ-approximations

for S is the optimal cost denoted $c(i,j)$. For the boundary condition that $i = j$, we let $c(i,j) = 0$. If $j > i$, there are two cases for establishing the optimal ϵ-approximation path.

1. $\tilde{S}(i,j)$ is the line segment (v_i, v_j)
   This case occurs when all the distances between vertices v_k, $i \leq j \leq k$, to (v_i, v_j) are less than ϵ. (v_i, v_j) ϵ-approximates $< v_i, v_{i+1}, \ldots, v_j >$ and thus $c(i,j) = 1$.

2. $\tilde{S}(i,j)$ consists of two or more line segments.
   In this case, $\tilde{S}(i,j)$ can be divided into two sub-paths $\tilde{S}(i,k)$ and $\tilde{S}(k,j)$ where v_k is a vertex on $< v_i, \ldots, v_j >$. Note that both $\tilde{S}(i,k)$ and $tildeS(k,j)$ ϵ-approximate polygonal paths $< v_i, \ldots, v_k >$ and $< v_k, \ldots, v_j >$. The cost of optimal ϵ-approximation $c(i,j)$ is $min_{i<k<j}(c(i,k) + c(k,j)$.

Based on the above discussion, the optimal cost can be written in the recurrence

$$c(i,j) = \begin{cases} 0 & \text{if } i = j \\ 1 & \text{if } (vi, vj)\epsilon\text{-approximates} < v_i, \ldots, v_j > \\ min_{i<j<k} c(i,k) + c(k,j) & \text{otherwise.} \end{cases} \quad (2.3)$$

All experiments in this study adopted a value of $\sqrt{3}$ for ϵ.

For the convenience of reproducing our results, the parameters used in all the steps above are summarized in Table 2-1.

2.2 Results

In this section, we demonstrate two reconstructed results and one application of the reconstructed results. The reconstructed olfactory projection neuron is the first trial and the reconstructed projection neurons in optical lobes show that our method can handle the fragmented branches and wide range of intensity.

Table 2-1: Parameters used in the proposed method

	Image Preprocess	Fragments assembling	Reconstruction	Approximation
Initial threshold level	10	-	-	-
Threshold for the 2D 8-neighbor CC	9	-	-	-
Threshold for the 3D 26-neighbor CC	30	-	-	-
Maximum edge weight to connect two component	-	5% of dimension	-	-
Radius R	-	-	1.0 μm	-
η	-	-	10	-
Ratio for determining branches in the soma	-	-	0.2	-
Threshold for connecting to a point in BpList	-	-	1.0 μm	-
Threshold for connecting to a point in Bps2D	-	-	0.75 μm	-
ϵ for path approximation	-	-	-	$\sqrt{3}$

2.2.1 The reconstructed structure of olfactory projection neuron

The neural network system of the olfactory system of Drosophila has received considerable attention from neural science researchers. The experiment in this study traced the axon of the olfactory projection neuron. Compared with other neurons, the axon of the olfactory projection neuron is relatively simple because it usually does not have a complex arborization structure.

The datum used for the demonstration was retrieved from the Olfactory Projection Fibers in the DIADEM dataset. For comparative purposes, the traced result and the original image were rendered in the same image, but with slight distancing (Fig. 2.6).

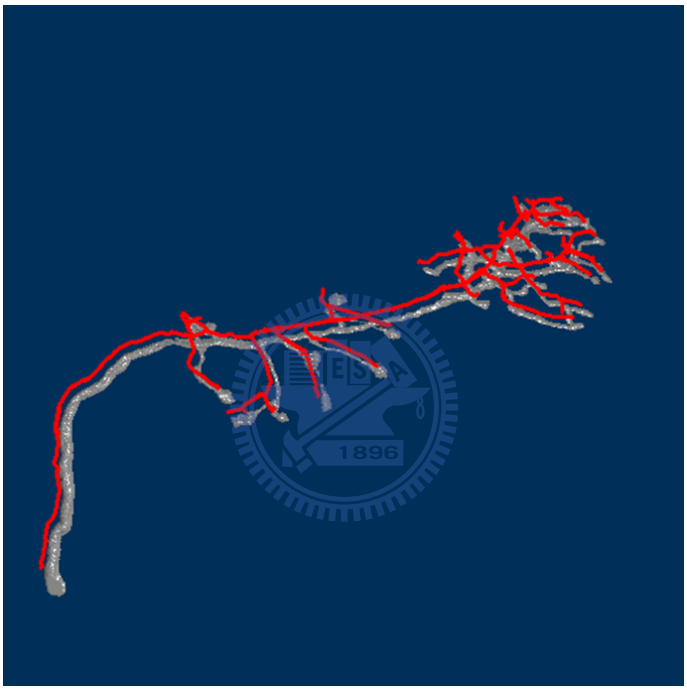


Figure 2.6: The traced result (*red*) of the image stack overlaps with the volume rendering of the original image stack. For the purpose of comparison, the result is translated a little from its original position.

2.2.2 Reconstructed structure of projection neuron connecting optical lobes

The neuronal structure is usually more complex than the olfactory projection neuron. An example of this is the projection neuron connecting two optical lobes. In this experiment, the intensity of the neuron image spreads widely, and broken branches emerge. The proposed method forms necessary corrections to produce satisfactory results. Figure 2.7 shows the traced results.

2.2.3 Tract discovery

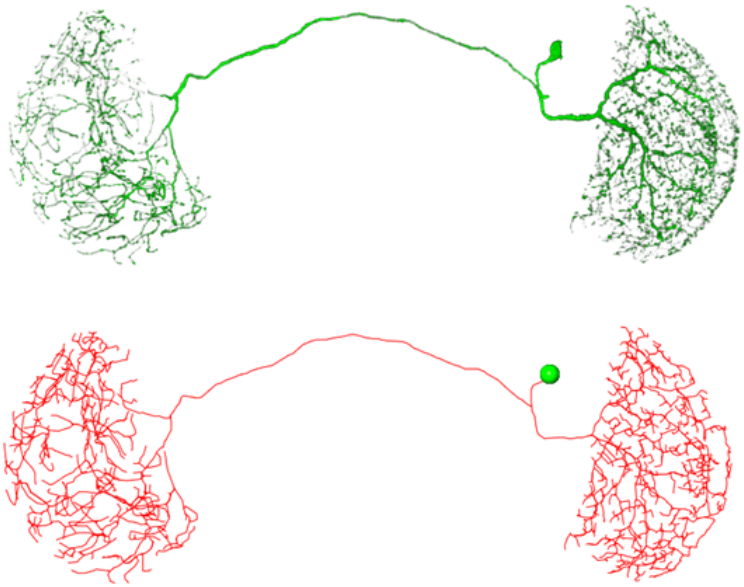
The traced results can be used to discover new information. Neuropils in the fruit fly brain are connected by neurons. Tracts are bundles of neurons and they are the major message channels connecting neuropils. Determining tracts in the fruit fly brain is an application. Below I will demonstrate a collaborative work with Mr. Chao-Chun Chuang. In this cowork, we used the reconstruction results and a computer method to determine the tracts. Here we take four neurons connecting ipsilateral medulla and contralateral lobula plate (Fig. 2.8a-d) as a simple example to demonstrate how to construct the neural tracts.

1. In each of the two neuropils, all terminals (brown spheres) and their average positions (blue spheres) were determined (Fig. 2.9a). The shortest path connecting the two selected terminals that are closest to the two average terminal positions was found (Fig. 2.8a).

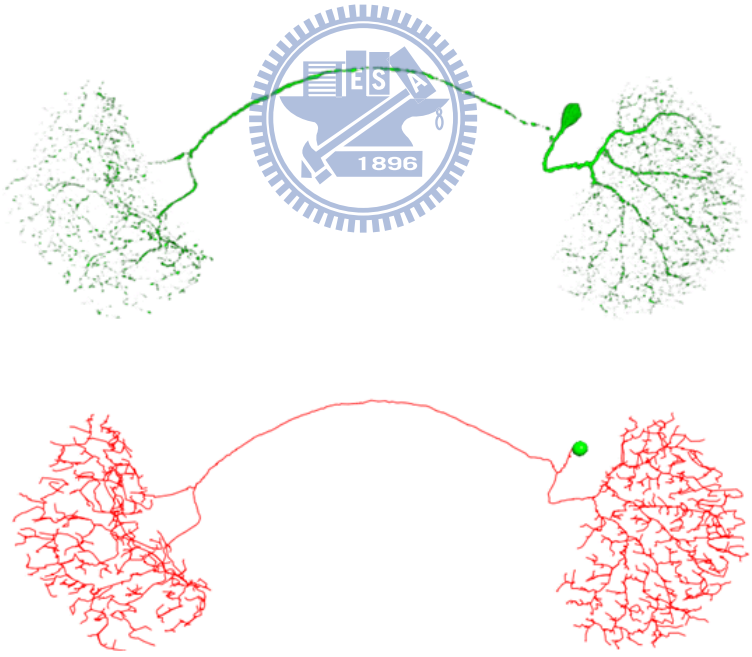
2. For simplicity, the cell body and internal paths in each of the two neuropils were removed (Fig. 2.9b-c).

3. Each of the 4 tracts was cut into 99 fragments of equal length. We then calculated the average distance of 100 points between tract pairs, and create a distance matrix table (Fig. 2.9d).

4. We used the function (Hclust) in R to calculate the matrix table. Finally, the tracts of neuron VGlut-F-300388 and VGlut-F-800007 were bundled into a single tract; the



(a)



(b)

Figure 2.7: Both (a) and (b) show the neurons (top) and the traced results (bottom). Both of the two neurons are optical nerves. The projection neuron connecting optical lobes has high density and complex morphology. In addition, the intensity of the neuron in (a) has a wide dynamic range. The proposed method can manage these situations and make necessary corrections. After the whole process, the reconstructions of both neurons are complete with high fidelity.

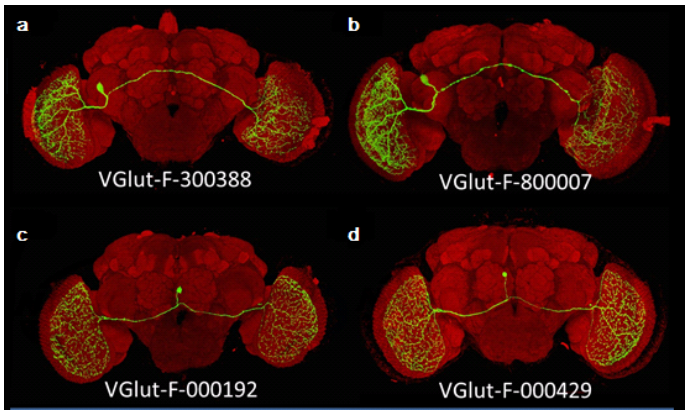


Figure 2.8: Four neuron tracts are used for the demonstration. Photo credit: Chao-Chun Chuang, Institute of bioinformatics and system biology, NCTU

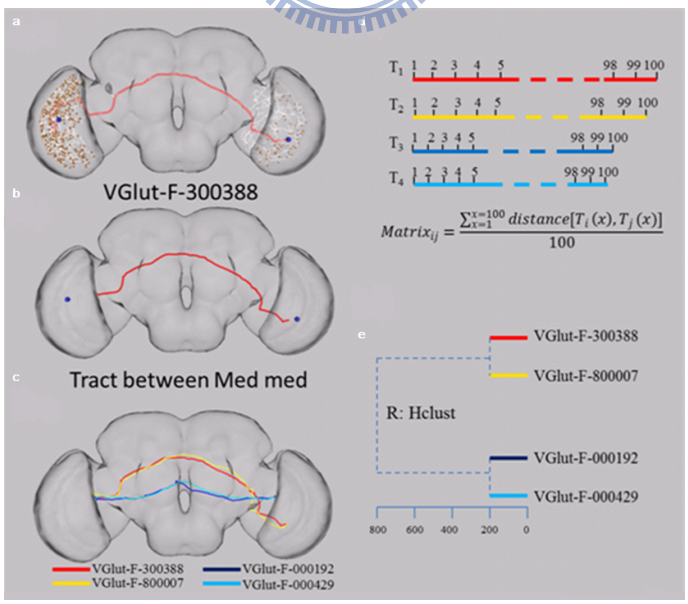


Figure 2.9: Steps for constructing neural tracts. Photo credit: Chao-Chun Chuang, Institute of bioinformatics and system biology, NCTU

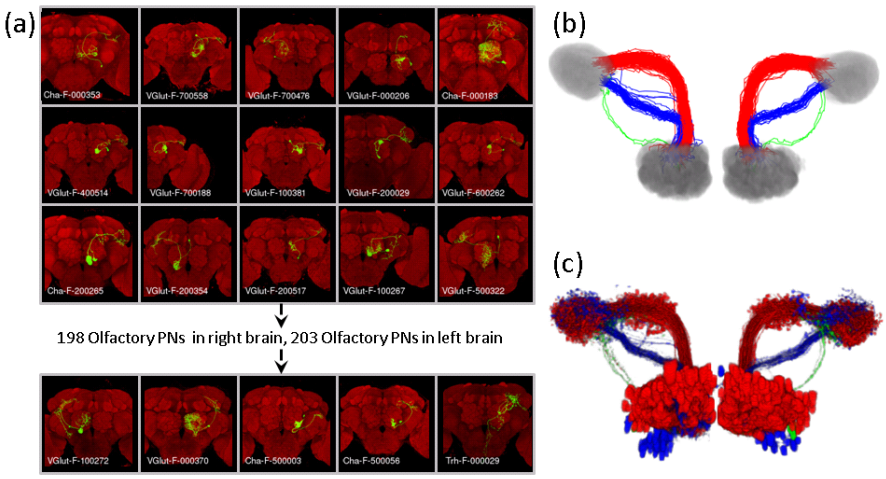


Figure 2.10: (a) 198 olfactory PNs in the right hemisphere and 203 olfactory PNs in the left hemisphere were selected. (b) Tract clusters: iACT (red), mACT (blue), and oACT (green). (c) Neuron image clusters overlap the tract clusters.

tracts of neuron VGlut-F-000192 and VGlut-F-000429 were bundled into another single tract.

Approximately 15,000 neurons were processed, and the reconstructed results were warped into a typical brain [32]. Consider the connection between two neuropils: the Antenna Lobe (AL) and the Lateral Horn (LH). Among the reconstructed neurons, 401 traced results were selected. Of 401 neurons, 198 neurons innervate both the LH and the AL in the right hemisphere without innervating the LH or the AL in the left hemisphere. The remaining 203 neurons innervate only both the LH and the AL in the left hemisphere. The paths connecting the AL and the LH were then extracted. Every path was evenly sampled, and hierarchical cluster analysis was applied to the sampled paths. The results show six clusters (Fig. 2.10), with three on each hemisphere.

The projection neurons ascending from the antennal lobe (AL) to the lateral horn (LH) form three tracts: inner antennocerebral tract (iACT), medial antennocerebral tract (mACT), and outer antennocerebral tract (oACT) [34] [35]. These three tracts are the major message channels from the AL to the LH. The computed six clusters (Fig. 2.10) show that three clusters are on both the left and right hemispheres. Comparing these clusters to the tracts previously observed from the image data shows that they are the same as the iACT,

Table 2-2: The processing time and Dis of four test data

Data	Size (voxel)	Time (sec.)	$Dis(N_2, N_1)$	$Dis(N_1, N_2)$
OP_1	512 x 512 x 60	11.297	1.413±0.746	1.341±1.091
OP_6	512 x 512 x 101	5.234	1.869±0.751	2.055±0.976
OP_7	512 x 512 x 71	6.359	1.747±0.849	1.801±1.100
OP_9	512 x 512 x 92	10.390	1.656±1.041	1.582±1.133

mACT, and oACT in both hemispheres. Thus, the data derived from the proposed method can be used for further analysis.

2.2.4 Processing time

Table 2-2 shows the processing time for every test datum chosen from the DIADEM test data. Except for the I/O time, it requires approximately 10 s for image preprocessing, tracing a single neuron and the ϵ-approximation procedure. The experiments in this study were performed using a PC with an Intel Core i7 920 processor and 8 GB memory. However, the actual memory usage was no more than 2 GB for all test data, including the optical nerves.

2.2.5 Accuracy analysis

With two reconstructed results, N_1 and N_2, from a set of volume data containing a single neuron, we define the distance from one reconstruction to the other. The distance from N_1 to N_2 is defined as follows. Let p be a point in N_1. The distance from p to N_2 is defined as

$$dis(p, N_2) = min_{q \in N_2} ||\overline{p, q}||_2 \tag{2.4}$$

In 2.4, $||\overline{p, q}||_2$ is the Euclidean distance in the image space. The distance from N_1 to N_2 is defined as

$$Dis(N_1, N_2) = \frac{1}{n}\sum_{i=1}^{n} dis(p_i, N_2) \tag{2.5}$$

where $p_i \in N_1$. This study computes the distance between the reconstruction and the ground truth to analyze the accuracy. Four data sets in olfactory projection fibers in the DIADEM test data were used.

Because the points on the neuron fibers of the ground truth are substantially denser compared to the reconstruction, points on the neuron branches of the reconstruction were sampled to produce the same density of points on the neuron fibers. Let N_1 and N_2 be the ground truth and our reconstruction respectively. We computed $Dis(N2, N1)$ for the four data sets, and the distances ranged from 1.4 to 1.87. Table 2-3 shows a summary of the experimental results.

This study also investigates the profile of $dis_{p \in N_2}(p, N_1)$ and $dis_{q \in N_1}(q, N_2)$. Figure 2.11 shows the histograms. Most of the points are within three voxels of the reference reconstruction. The maximum difference is larger when the automatically reconstructed result is the reference. This is likely because large nodular structures are in the projection neurons. Experts usually trace more points to the boundary of the nodular structure. However, the proposed method computes the centerline to represent the nodular structures by using fewer points.

Consequently, a branch in the reconstruction has a good chance to be shorter than the same branch in the ground truth. Figure 2.12 shows both the gold reconstruction in the DIADEM data set, and our reconstructed result overlapped with the original neuron image. Large distances appear in the green rectangular boxes in Fig. 2.12(b). In this experiment, the distance is measured in the number of voxels. Each voxel is considered a unit cube.

2.2.6 Robustness test

The robustness test was performed by first adding Gaussian random noise to the datum, and then applying the proposed method to compute the centerlines of the neuron. We tested five cases for different noise levels. The noise levels were determined by the standard

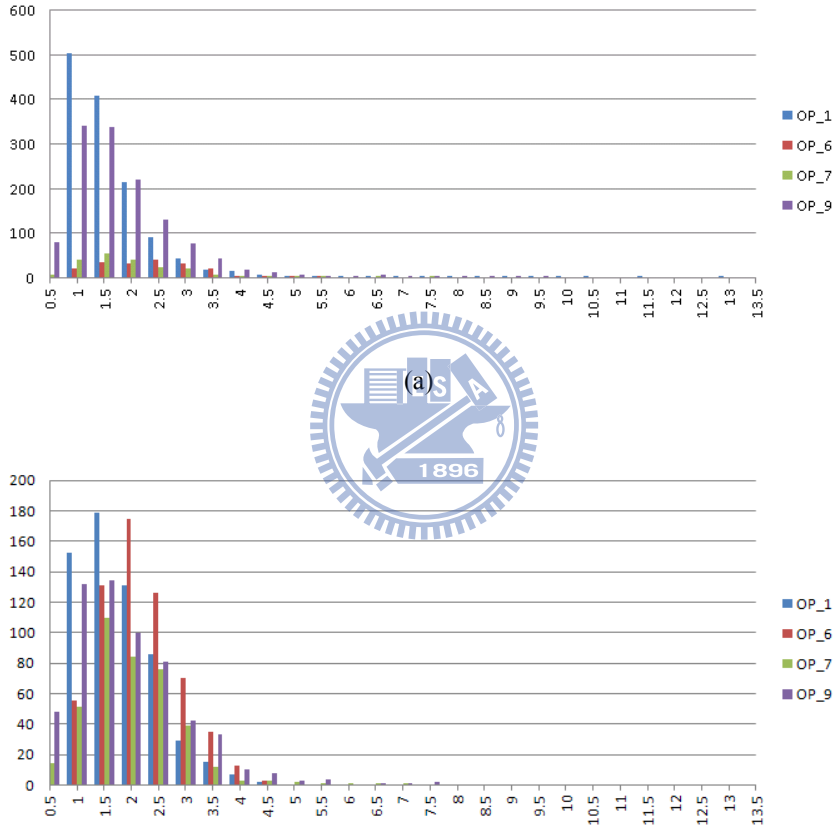
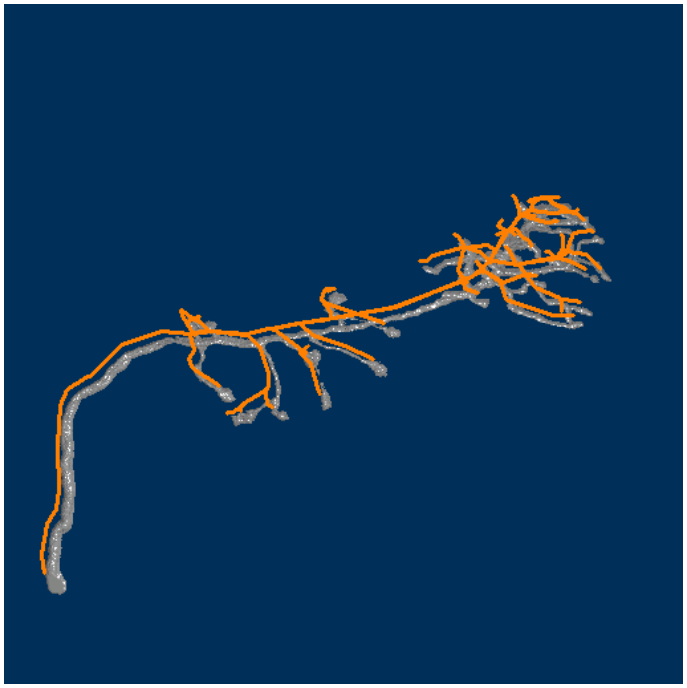
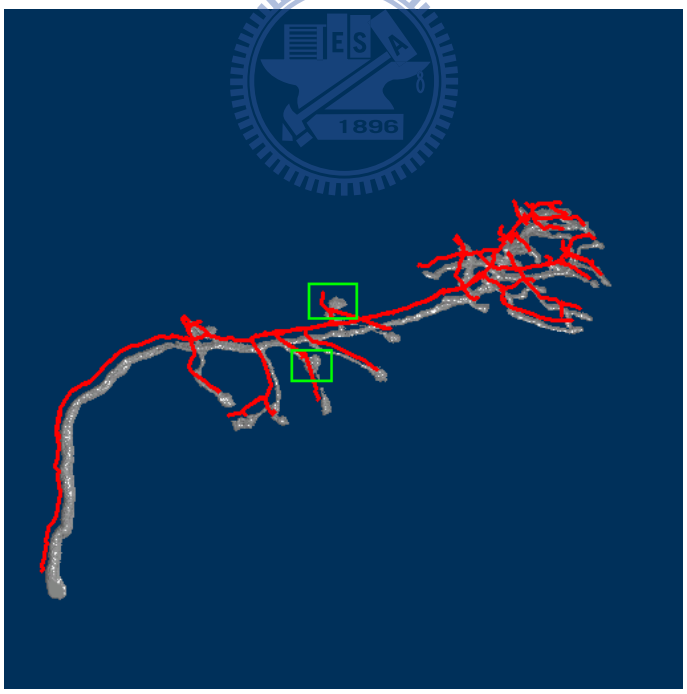


Figure 2.11: These two histograms show the distribution of the distances when the references are respectively our reconstruction (top) and ground truth reconstruction (bottom).



(a)



(b)

Figure 2.12: The gold reconstruction (*orange*) overlaps with the volume rendering of the original image stack. (b) Our tracing result (*red*) overlaps with the volume rendering of the original image stack. The green rectangles indicate the regions where the large distances happen.

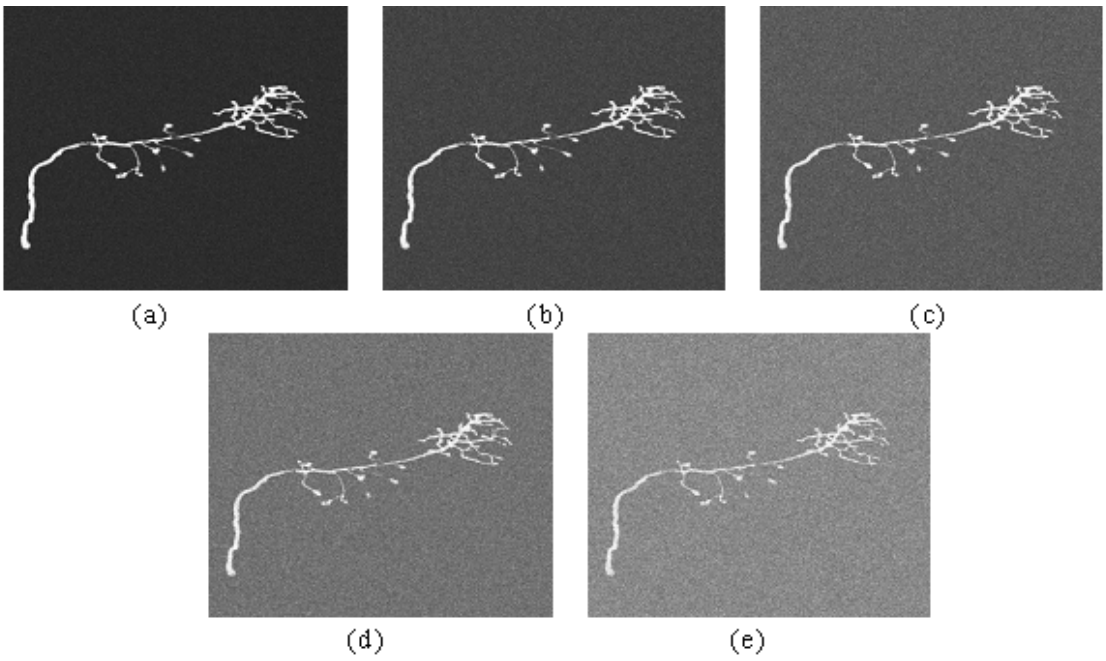


Figure 2.13: The images were contaminated by different levels of Gaussian noise: (a) σ = 20, (b) σ = 30, (c) σ = 40, (d) σ = 50, and (e) σ = 60.

deviation of the Gaussian distribution, to be σ = 20, σ = 30, σ = 40, σ = 50, and σ = 60. The intensity value ranged from 0 to 255. σ increases in conjunction with the number of visible voxels, but the percentage of visible voxels on the neuron decreases (Fig. 6), making neuron tracing more difficult.

Because the image is contaminated by noise, keeping 70% of the brightest visible voxels could include too many noisy voxels. The visible voxel is defined in the Methods section. Instead of keeping 70% brightest voxels, only 20% of the brightest voxels among the visible voxels were kept in the robustness test experiment. To quantitatively evaluate the robustness of the proposed method in noisy images, this study also calculates the distance between the reconstructed result and the gold reconstruction in the DIADEM data set. As Fig. 2.14 shows, the distances between the reconstructions and the ground truth are still manageable, despite the noise level being at 60. Figure 2.15 shows the traced results.

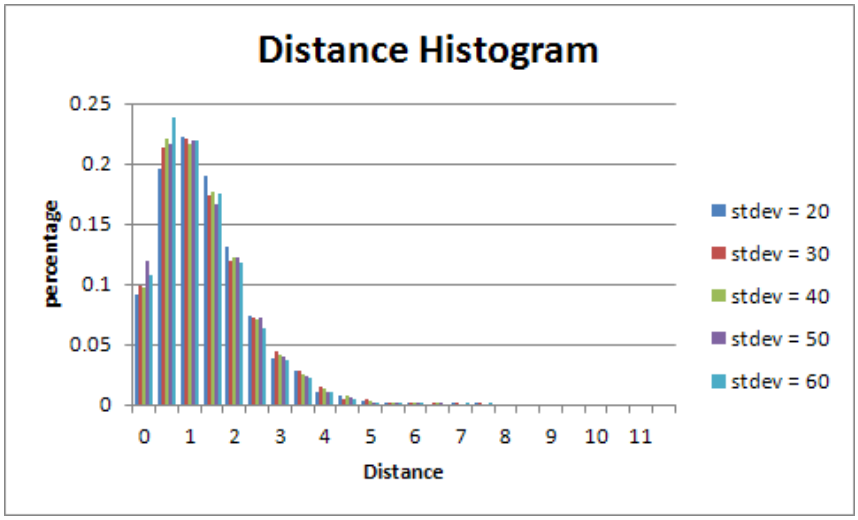


Figure 2.14: The histogram for all five different noise levels. We can find that the distributions of five noise levels are almost the same.

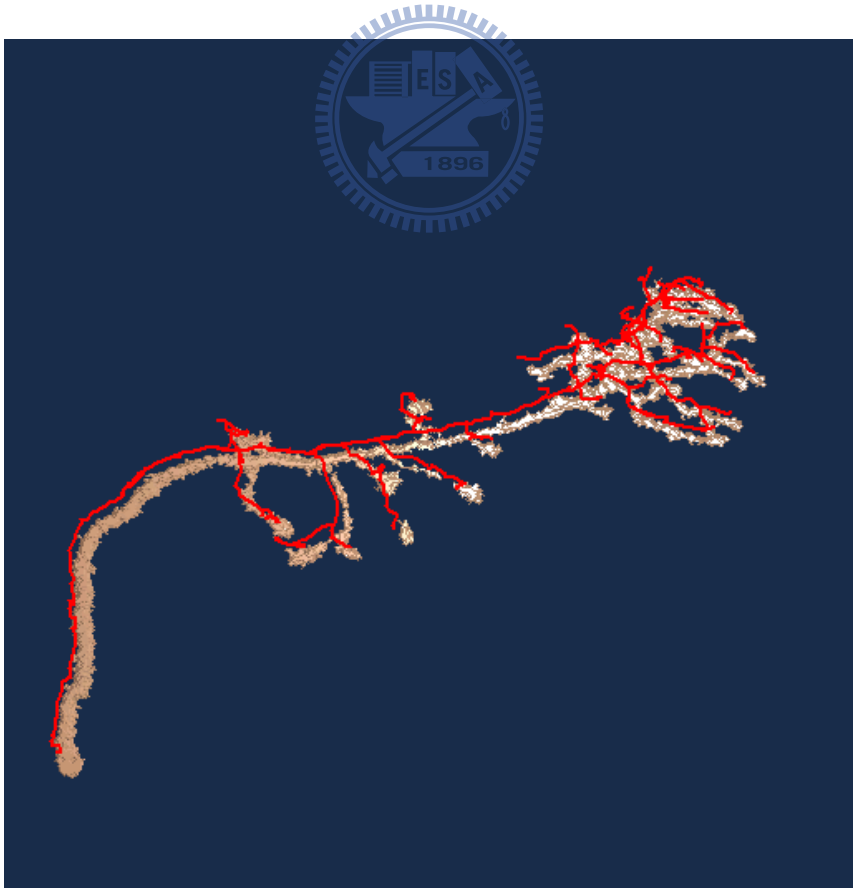


Figure 2.15: The traced result (*red*) of the contaminated, $\sigma = 60$ image stack overlaps with the voulme rendering of the image stack.

2.3 Discussion

In this section, we presented a computer method for reconstructing neuronal structure from an image stack. Based on the fact that biologists tend to use confocal microscopes for optimal images in a slice for human visualization, we proposed processing 2D slice images first. 3D neuronal structures were then constructed from the processed 2D images. Using this strategy, a high-throughput method was designed. More than 16,000 neurons were reconstructed and stored in the database [32]. A few of the reconstructed neurons were incorrect, mainly because the resolution of the optical microscope is not sufficient to distinguish dense branches.

The features used to design the weights of the edges (Eq. 2.2) were extracted from the 2D skeletons. One of the features is the branch point; however, the proposed template matching method is unable to detect all branch points. When a slice passes through the branch point, and the two branches are respectively above and below the slice, we are not able to detect the branch point. This problem may cause errors in the location of the branch point.

Another weakness of the proposed method is related to detecting delicate structures. The candidate 3D end points were obtained from the 2D end points, and the end points close to a branch were removed. Some small branches could therefore be considered as noise and ignored. Thus, the reconstruction by the proposed methods is inadequate for a study for which the details of a neuron are extremely important, e.g. the study of neuron dynamics [16].

Currently, the DIADEM [28] data set is widely used in the study of neuron reconstruction for accuracy evaluation. A scoring system is provided. We used the DIADEM data set as ground truth to evaluate the accuracy of the proposed method; however, reconstructions obtained by the proposed method did not achieve a good DIADEM score. The reasons are:

- The neuron branches obtained were based on the 2D skeletons in each slice. Thus, neuron branches obtained by the proposed method are shorter than those traced by

experts.

- Our approach tends to merge branch points in a small region to a single point. The proposed method could ignore some branch points in the ground truth. However, each branch point is highly weighted in the DIADEM metric so that our reconstructed neuron could receive a serious penalty.

- The DIADEM ground truth constructed by experts was obtained using NeuroLucida. The traced neurons were smoothed by spline interpolation. Coordinates of points on the neuron branches are real numbers. We used a polygonal line to approximate a neuron branch; coordinates of the points are integer numbers. Consequently, a large error could occur in estimating the radius of the neuron branches of our reconstruction.

Although neurons constructed by the proposed method cannot achieve a good DIADEM score, nevertheless, as shown in the Results section, the reconstructed results are suitable for further studies. In conclusion, the reconstructed neurons are sufficiently reliable to support the analysis of the neural network.

Chapter 3 A Computer Aided System for 4D Neuronal Dynamic Analysis

Detail changes of 3D filament structures of dendritic arbors between two time-points can be categorized as *retracted*, *newly added*, *transient*, or *remaining* branches. Many parameters for describing a neuron have been proposed [36] [37]. These parameters quantitatively describe the topological and geometrical relations of two neurons; however, when the chronological relation of structures of a single neuron is undefined, we cannot utilize these parameters to characterize the developmental process. Let N^1 and N^2 be two reconstructed neuronal trees at two consecutive time-points. Typically, for a branch b in N^1, the expert has to go over all un-matched branches in N^2 to confirm whether there is a match; this is extremely time-consuming. With a reliable list of suggested matching branch candidates, the user does not have to scan for a wide range of candidate matches in N^2, which would greatly reduce processing time. In addition, the suggestion list generated by the same algorithm helps to keep the matching criteria consistent and thus reduces human errors.

In this chapter, we present a supervised 4D neuronal Structural Plasticity Analysis (4D SPA) computer method that computes precise changes in the positions and lengths of all neuronal branches in the arbor between two images or time-points and presents the data as an image superimposed on the 3D reconstruction of the neuron. In the proposed method, an intrinsic feature, *significant point* of a neuron is defined. We decomposed a neuron into a set of branches. Geometrical similarity was used to define the distance between two branches.

In this study, we used *in vivo* time-lapse images of neurons from the optic tectum of stage 48 *Xenopus laevis* tadpoles to test and demonstrate the 4D SPA method. The details of the system is described in the Methods section; The Results section presents matching results; A comparison between the proposed method and two common strategies used for pattern recognition is shown in the Discuss section; finally, the operation manual of the proposed system is in the Appendix I. The Discussion section shows the consuming time

to align two neurons by applying automatic computer methods and by experts with the help of the system. This method takes advantage of both computer algorithm and human expertise. With some help from human supervision, we were able to significantly reduce the processing time and increase the accuracy.

3.1 Methods

3.1.1 Sample preparation

GFP (green fluorescent protein)-expression DNA constructs were electroporated into the tectum *Xenopus laevis* tadpoles. Animals were screened for brightly labeled well isolated single neurons. Time-lapse imaging was conducted using a custom-built two-photon laser-scanning system, which was modified from an Olympus Fluoview confocal system with an Olympus Fluoview confocal scan box mounted on an Olympus BX50WI microscope. GFP was excited by 910nm laser provided by a Tsunami femtosecond-pulsed Ti:sapphire laser pumped by a 10 W solid-state Millenia X laser. Images were acquired with the Olympus Fluoview acquisition software and an Olympus XLUM Plan FI 20X water immersion lens (NA 0.95, Japan) with 3X zoom and $1\mu m$ step size in the z-axis to capture the entire extent of dendritic branch arbor. The two images were collected 4 or 24 hours apart. Reconstruction of the entire dendritic arbor of the neuron was done by the computer-aided filament tracing function in the 3D/4D image analysis software Imaris©(Bitplane, USA) and the full filaments data was exported for dynamic analysis using the proposed computer aided comparison method.

3.1.2 Preliminary

The traced neuron, denoted N is presented as a tree rooted at the soma, s. For a pair of points u and v in N, if there is a directed path $b =< u, p_1, p_2, \ldots, pk, v >$, we use $< u, v >$ to denote the path b and the path length, $L(b)$ is defined as $L(b) = |\overline{u, p_1}| + \sum_{1 \leq j < k} |\overline{p_j, p_{j+1}}| + |\overline{p_k, v}|$. The points are categorized into three sets, the tips, the regular

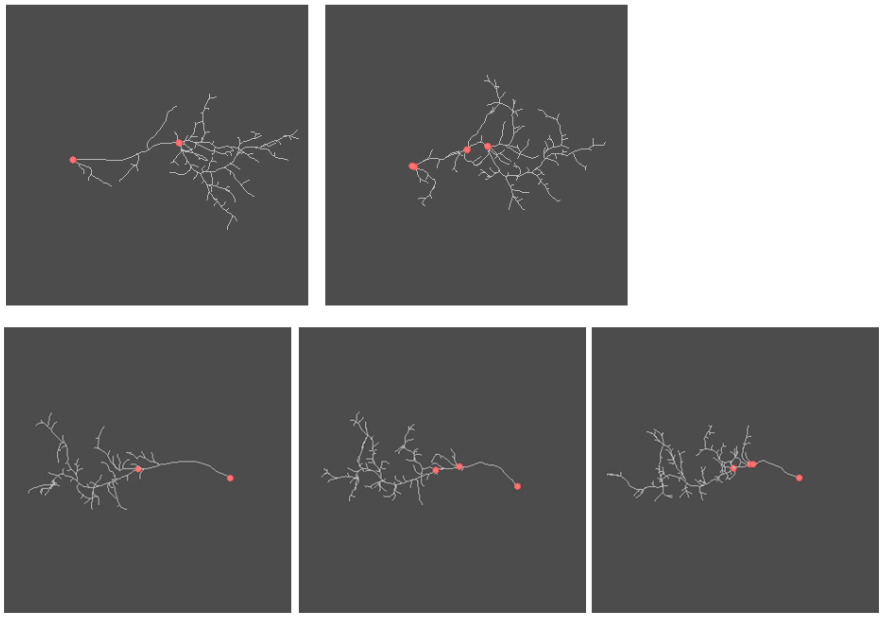


Figure 3.1: Two sets of data and their significant points are shown. They are both 24-hour lapsed. The first set of data contains two time-points neuron data (first row) and the second set of data contains three time-points neuron data. Both sets of data, from left to right, are arranged in increasing chronological order. Every pink sphere represents a significant branch point or the soma. Clearly, significant branch points last for sufficient long time.

points and the branch points. The tips are the leaves denoted t_i, $1 \leq i \leq n$, where n is the number of tips. The branch points are vertices that have out degree greater than 1. All the other vertices are regular points on the neuron fibers. They have out degree 1.

In a path $b =< u, v >$, if u is the soma or a branch point and v is a tip, then the path b is called a "branch" in N. For the purpose of computing the similarity between a pair of branches coming from two neurons, firstly we resample points on the path between two consecutive branch points so that the length between any two consecutive points on the path is approximately 1

To facilitate comparison of the neuronal trees acquired at different time points, we need to define 'significant' points. Within the neuronal tree, each branch point could be considered as the root of a subtree. The size of a subtree is defined as the number of branch tips in

that subtree. We denote $\Delta S_T(v)$ the minimal subtree size of the tree rooted at v. We also denote the average of the minimal subtree size of a neuron, N, $\Delta\overline{S_T(N)}$ where $\Delta\overline{S_T(N)}$ is calculated by dividing the sum of $\Delta S_T(v)$ within N by the total number of branch points. A branch point, v, is defined as a "*significant*" point if 1. $\Delta S_T(v) > \Delta\overline{S_T(N)}$; and 2. One of the following criteria is also satisfied:

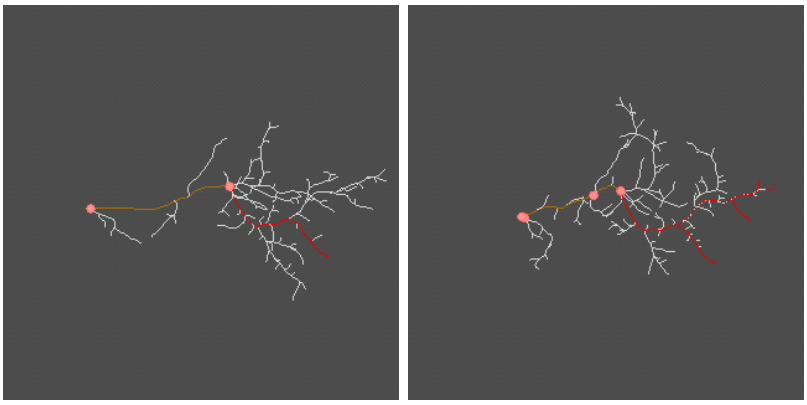
1. There are more than 2 subtrees rooted at v.
2. There are only two subtrees rooted at v and the sizes of both subtrees are greater than 3.

Empirically, significant points defined above are usually stable over time and would exist in the dendritic trees of the same neuron at two consecutive time-points, thus can be used as reference points for alignment of the arbors (Fig. 3.1). In the following matching process, we firstly consider the branches emanating from a significant branch point b_i to a tip t_i where b_i is the closest significant point to t_i on the path from t_i to the soma (Fig. 3.2). If there is no significant branch point along the path from t_i to the soma, then the soma is the starting point.

3.1.3 Branch attributes

Each branch $b =< b_i, t_i >$ has three attributes, the *length*, $L(b)$, the *parent*, and the *position*. The *parent* is either itself or another branch $b' =< b_j, t_j >$ that b attaches on it. The position of b is undefined if the parent is itself. If the parent of b is b', then the position = $L(< b_j, b_i >)/L(b')$. The parent and position of b are denoted $parent(b)$ and $position(b)$ respectively. Initially, parent of each branch is itself. These attributes are used to compare a pair of branches between two images in a pre-screening step. The shape similarity of the branches is taken into consideration only if the pair of branches pass the pre-screening. The similarity between two branches is defined as Eq. 3.1.

$$f(b_1, b_2) = \min_{R,v} \frac{1}{n} \sum_{x \in b_1, y \in b_2} ||Rx_i + v - y_j||_2 + \max\{\frac{L(b_1)}{L(b_2)}, \frac{L(b_2)}{L(b_1)}\} \qquad (3.1)$$



(a) (b)

Figure 3.2: The time lapse between (a) and (b) is 24 hours. All significant branch point are marked as pink sphere. The conjunction points of *orange* segments and *red* segments in both (a) and (b) are major branch points. If we consider only red segments in both (a) and (b), they are highly similar. But if we consider a branch starting from soma, i.e. a branch comprising both *orange* and *red* segments, then these two branches are less similar. All red branches in (b) are computed matching candidates.

where R is a rotation matrix and v is a translation vector. The number of points on b_1 is n. The value of $f(b1, b2)$ goes up when similarity goes down. The first term in Eq. 3.1 can be solved efficiently [38].

3.1.4 Pairwise matching analysis

For two reconstructions of the same neuron traced from image stacks acquired at different time-points, the first one is designated N^1 and the second one is designated N^2, in chronological order. During the matching process, branches in N^1 are categorized into 3 types, *remaining*, *retracted*, and *undefined*. A branch in N^1 is *remaining* if there is a matched branch in N^2. A branch is *retracted* if there is no matched branch in N^2. An *undefined* branch is a branch that has not been processed yet. Similarly, the branches in N^2 are also characterized into 3 types. The *remaining* and the *undefined* are the same as defined in N^1. There is no *retracted* branch category in N^2 instead, there are *newly added* branches, which refer to branches in N^2 that do not have a match in N^1.

The matching process is a greedy approach. Each iteration consists of the following steps:

1. The longest undefined branch, B in N^1 is chosen.

2. To save the computing time, we pre-screened all undefined branches in N^2 by checking the attributes of each of them and the attributes of B. For each undefined branch, $\tilde{b} =< b_j, t_j >$ in N^2, we check its attributes. $\tilde{b}$ is collected into the *candidate short list* if it satisfied all criteria listed below:

   (a) $parent(B)$ and $parent(\tilde{b})$ are matched pair or both B and $\tilde{b}$ are undefined,

   (b) $\max\{\frac{L(\tilde{b})}{L(B)}, \frac{L(B)}{L(\tilde{b})}\} < \alpha$,

   (c) $position(B) - position(\tilde{b}) < \beta$.

   α sets the upper limit of variations allowed in the branch length for the two branches in comparison. β accounts for the variations allowed for the branch position on the parent branch. Both parameters help to take into account of the difference caused by neuronal growth over the imaging time interval. The values are decided empirically and could be adjusted to optimize the program for specific applications. In all of our experiments we used the default setting set $\alpha = 2.5$ and $\beta = 0.4$.

   The similarity (Eq. 3.1) between B and all branches in the *candidate short list* are calculated. The default setting of the length of the list is limited to five so at most, the most similar five branches will be included in the list (Fig. 3.2).

3. The analyst then uses the suggestion list to assign a matched branch for B. If the analyst decides that there is no matched branch for B, then B is assigned to the 're-tracted' category. Based on the assignment of branches to categories, the attributes of some branches in both N^1 and N^2 are updated. The following is the update procedure for the branches in N^1 and N^2:

   (a) If $L(B) < \gamma$, then the procedure stops. Here, γ stands for the minimum length a branch must have to be used for morphology comparison. This is because very short branches usually do not have many unique morphological features and could be easily confused with other non-matching short branches. During the course of our experiments, we set $\gamma = 15\mu m$.

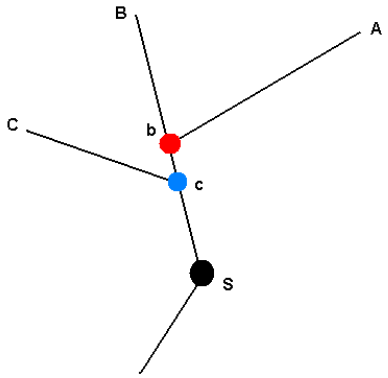


Figure 3.3: The original start points of branch A, B, and C are S. Assuming that branch A is identified as 'remaining' then the parents of both B and C become A, and the start points of B and C are updated to b and c, respectively.

(b) If $\tilde{b}$ is categorized as *remaining* or *retracted*, no further process is needed.

(c) Assuming an undefined branch, $\tilde{b} =< b_k, t_k >$ in N^1 has a common path with B and the conjunction point of B and $\tilde{b}$ is $\tilde{c}$. Then $\tilde{b}$'s parent and starting point become B and $\tilde{c}$ respectively. A schematic example is shown in Fig. 3.3.

4. If there were no more *undefined* branches in N^1, then all of the remaining *undefined* branches in N^2 were changed *newly added* and the process terminates. Otherwise, iteration restarted from step 1.

A GUI software system was implemented to facilitate the user to complete the matching process. Through the graphics user interface, user can decide the type of a branch and to set up parameters, α, β, and γ for the matching process.

3.1.5 Dynamics pairwise analysis

Pairwise analysis of sequential images in a longer time-lapse data set requires that branches have unique identifiers that are transmitted through the image series. Initially, the branches are assigned an identifier or an "index" according to their length in descending order. After the user confirms all the matching branch pairs in N^1 and N^2, the 4D SPA program adjusts the indices so that the matching branches are assigned with the same identifier. To combine

the multiple pairwise alignment results in data sets with more than 2 images, an extra step is added at the end of each comparison to synchronize the branch indices. This way, we can easily identify each individual branch in every time-point and analyze the dynamic changes on that particular branch over time.

3.2 Results

We used *in vivo* time-lapse images of neurons from the optic tectum of stage 48 *Xenopus laevis tadpoles* to test and demonstrate the 4D SPA method. Neurons were labelled by expression of GFP (green fluorescent protein) and time-lapse images were acquired with a two-photon laser-scanning microscope, at either 4 hour or 24 hour intervals. Reconstruction of the entire dendritic arbor of the neurons was done by the computer-aided filament tracing function in the 3D image analysis software Imaris and the full filament data set was exported for dynamic analysis using the 4D SPA method. Ten data sets were used in our experiments. We allowed the experimentalist to use two data sets to become familiar with the 4D SPA program and then recorded the time to analyze the remaining eight data sets.

3.2.1 Pairwise analysis

In this analysis, N^1 and N^2 are two reconstructed neuronal arbors imaged at consecutive time-points. Typically, to manually evaluate structural dynamics in two images, the analyst identifies a branch in N^1, reviews all un-matched branches in N^2, and selects the best matched one. This process is extremely time-consuming and subjective. Our program generates a reliable short-list of suggested matching branch candidates in N^2 for every branch in N^1 based on the algorithm described above, which should greatly reduce the processing time. In addition, use of the same algorithm to generate the short list of suggested matching branch candidates helps to maintain consistent criteria to identify candidate matches and reduces human errors. We quantified the reliability of the algorithm by calculating the hit ratio. For a branch in N^1, the 4D SPA method generates a list of

suggested matching branches in N^2. We get a hit if one of the following two criteria is met:

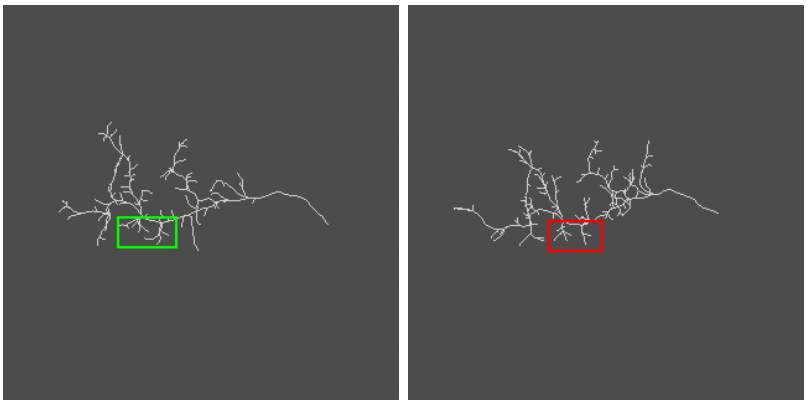
- The user decided there is no match, and the branch is identified as *retracted*, or
- The user selected a matched branch from the suggested short list.

The hit ratio is calculated as the percentage of hits relative to the total number of branches in N^1, which measures the accuracy of the suggested match list. The analysis result for each data set is listed in Table 1. The mean number of branches on the suggested match list was 2.78 (Table 3-1). The average hit ratio was $85.8 \pm 8.4\%$ (Table 3-1). A pair-wise alignment result of representative dataset number 8 is shown in Video S1 (`http://people.cs.nctu.edu.tw/~percycat/VideoS1.mpg`) In the supplementary video, each frame displays both N^1 and N^2, and only one matching pair, which had been confirmed by the analyst, is rendered in red. Among all the data sets, data set number 7 had the lowest hit ratio (70.3%) and the longest average list of the suggested matches. This is due to the high similarity between subtrees emanating from the same parent branch in N^1 and N^2 (Fig. 3.4). The combination of similarity in both subtree structure and the parent branch tends to mislead the algorithm.

The average time to align a pair of complete dendritic arbors was about 43 minutes (Table 3-1). In the worst case, it took 111 minutes to process a set of data. Without the help of 4D SPA, it usually takes hours for an expert analyst to process a paired data set, depending on the complexity of the arbor and the magnitude of change between the two neuronal structures [16].

3.2.2 Dynamic Branch Analysis

This method can be easily applied to analyze data sets consisting of multiple time-points in which N^1 is aligned with N^2, then N^2 is aligned with $N^3, \ldots$, and so on. Analysis of data sets with more than 2 time-points provides detailed analysis of branch dynamics, and often reveals important information about the underlying mechanisms of structural plasticity that are lost when long-interval data points are compared [20] [39] [40]. In



(a) (b)

Figure 3.4: (a) N^1 of the data set 7 (b) N^2 of the data set 7. The structures in the green rectangle are quite similar and the possible matching structures in N^2 (shown in the *red* rectangle) are also similar among themselves. In addition, the ratio positions of these structures do not vary much. Altogether, these confused the proposed method.

Table 3-1: The processing time and hit ratio

Data	Hit ratio	Time (min.)	size of N^1	size of N^2	Avg. list size
1	0.921	42	76	63	2.422
2	0.859	43	63	79	2.422
3	0.967	111	89	84	2.422
4	0.795	13	38	58	3.051
5	0.917	46	58	59	3.483
6	0.758	44	60	80	3.515
7	0.704	35	80	94	3.714
8	0.833	5	13	18	2.5
9	0.969	-	31	24	1.407
10	0.856	-	69	80	2.814
mean Hit ratio: 0.858					
stdev. Hit ratio: 0.084					
mean Time: 42.375					
mean Avg. list size: 2.78					

analysis of branch dynamics, branches are categorized as *stable*, *newly added*, *retracted* or *transient*. A 'stable' branch is a branch that is present at all time-points. A retracted branch is a branch that is present at the first time-point but is lost at a later time-point. A 'newly added' branch is a branch that does not exist at the first time-point but appears later and remains in place until the last time-point. The 'newly added' branch category includes those that appear at the last time point. A 'transient' branch is a branch that appears at a time-point after the first time-point and is then retracted by the last time-point. To better visualize the alignment results and branch dynamics, we implemented a color-coding system for the final display of the matching results. The *stable* branches are coded black, the *retracted* branches are blue, the *transient* branches are magenta, and the *newly added* branches are green. (Fig. 3.5). This implementation facilitates the identification of qualitative changes in branch dynamics as well as possible spatial patterns of dynamic changes in the dendritic arbor over time.

3.3 Discussion

We present a method to provide computer assisted 4D structural plasticity analysis (4D SPA) of neuronal dendritic arbors. Here, we describe a simple but efficient algorithm that takes advantage of the presence of relatively stable branch points within the neuronal arbor structure combined with their geometry to facilitate the analysis of dynamic changes in the neuronal structure. To precisely identify similar structures within the tree structure by computer algorithms is a very difficult problem, given the subtle difference the arbors could have at different time points, and artifactual differences caused by shifts in the position of the animal during imaging and other factors. To achieve the most reliable result in an expeditious manner, we combined the advantages of the computer algorithm and the expertise of human analyst. Instead of having the computer generate the final analysis, we use the computer algorithm to effectively narrow down the candidate list of the matched branches and let the human expert select the optimal match.

4D SPA significantly increases the rate at which structural dynamics can be quantified and mapped onto the neuronal structure, and effectively reduces errors in the analysis and

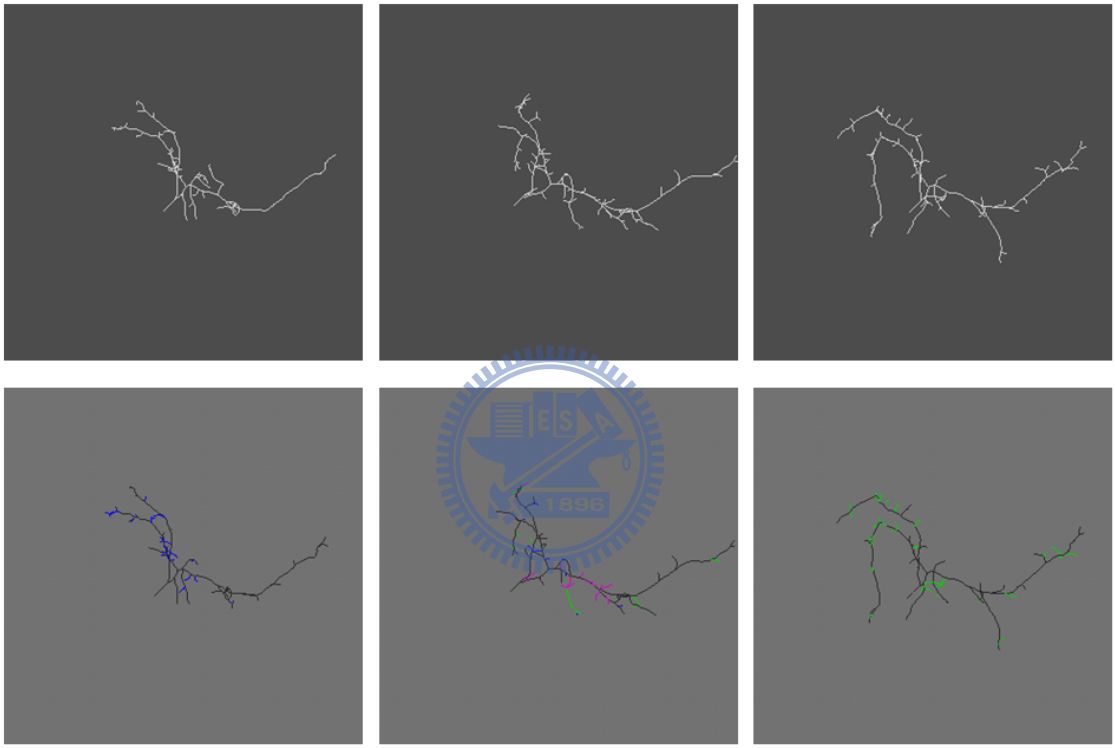


Figure 3.5: A set of color-coded results is demonstrated. The original data are shown on the first row and the corresponding color-coded results are shown in the second row. The 'retracted' branches are *blue*, the 'transient' branches are *magenta*, the 'stable' branches are *black*, and the 'newly added' branches are *green*.

discrepancies between experimenters. As described in the method section, there are three parameters, α, β, and γ used in the program that can be adjusted by the analyst to optimize the algorithm to particular applications and cell types. For instance, higher values of α and β can be used if the images are taken over a longer time interval and the neuron is expected to have changed a lot at the second time point. On the other hand, for relatively stable neuronal structure, smaller value would be preferred to reduce false positive suggested matching candidates. In order to avoid to branches become too fragmented, γ should not be set too small. The best setting for the parameters differs from one kind of neuron to another. However, the analyst could specify an optimal setting based on his background knowledge. This increase the flexibility of the proposed 4D SPA. The accuracy of the suggestion list decreases when the morphology of the subtrees attaching to the same parent branch are highly similar, for instance in in highly symmetric neuronal structures. In this case, extra care needs to be taken using the suggestion list. In the cases where the real matched branch is not included in the suggestion list, we have implemented a manual assignment function in the program, which allows the analyst to pick any branch in N^2 and assign it as the matched branch for the branch in question.

A common strategy used in the applications of pattern recognition is to explore the differences by a weighted-matching algorithms [41] and applications to biomedical data were also reported [42] [43]. Decomposing a traced neuron into a set of branches and using Eq. 3.1 as the weighting function, we compare the performance of the automatic weighted-matching method and the proposed method. In our implementation, the Hungarian algorithm is used to report the weighted match and thus, we can only apply the length ratio, α as matching criterion. Another strategy to find the matching is the greedy approach. The greedy matching process is the same as the process detailed in the section 3.1.4, instead that we chose the least weighting pair as a match in each iteration.

In practice, processing time is an important measurement to evaluate an automatic method. Using the data sets 1-8 listed in the Table 3-1, we recorded the processing time of both automatic methods. We performed all of the experiments on a PC with Intel i7 920 inside. The average consuming time for the automatic weighted-matching method is 91.71 minutes which is more than twice of the average processing time when the proposed method is used; besides, the proof-editing process is necessary before further analysis and this

Table 3-2: The processing time of two automatic approaches

Data	Time for optimal matching (min.)	Time for greedy matching (min.)
1	128.3	1.08
2	142.03	1.83
3	112.52	0.85
4	16.07	0.58
5	31.33	0.77
6	129.62	0.68
7	170.72	0.70
8	3.10	0.25
Avg. time	91.71	0.84

would take more time. As expected, the greedy approach is the fastest method to complete the matching task; the average consuming time is 0.84 minutes. However, the results are considerably unreliable. The processing time of each experiment of both methods was listed in Table 3-2. Exemplary video clips showing the results reported by the Hungarian algorithm and the results generated by the greedy approach could be found at the Video S2 (`(http://people.cs.nctu.edu.tw/~percycat/VideoS2.mpg)` and Video S3 (`(http://people.cs.nctu.edu.tw/~percycat/VideoS3.mpg)`, respectively.

Quantitative analysis of structural dynamics has been key to gaining insight into the mechanisms and function of dynamics in neurons. In conclusion, we anticipate that application of this method to a variety of biological systems will be valuable.

Chapter 4 Conclusions

I concentrated on two methods for improving fundamental neuroscience studies in my research. The first one focused on method to efficiently reconstruct 3D neuronal structures from an image stacks. Compare to the manual reconstruction, the proposed method required much less time to complete the job. However, if we considering the whole process of reconstruction, there still are improvements could be done. Occasionally, there are multiple neurons labelled in the same brain tissue and it relies on experts to separate them with the help of software. with no doubt, the more time could be saved if there was a automatic segmentation method.

The second method emphasizes on designing a tool to help neuroscientists aligning two neurons. In my design, the feature used to decompose the neuron was based on observation and all values of the parameters were set empirically. I trust there were some more genius descriptors of a neuron. In practice, the proposed tool really helps neuroscientists saving their precious time but frankly speaking, it still takes time. An automatic method is necessary.

In conclusion, we proposed two frameworks to solve two issues that many neuroscientists encountered in their researches. Both frameworks worked fine; nevertheless, we all clear that this is far away from perfect. However, I believe that what we had done is a stepping-stone for contemporary and future researchers to explore our understanding of Nature.

Bibliography

[1] A.H. Baron, N. Perrimon, "Targeted gene expression as a means of altering cell fates and generating dominant phenotypes," *Development* vol. 118 pp. 401–415. 1998

[2] R. Yuste and A. Konnerth, "Imaging in neuroscience and development," *CSHL press*, 2005

[3] R. J. Greenspan, "Fly Pushing," Cold Spring Harbor Laboratory Press, 2004

[4] A.S. Chiang, Y.C. Liu, S.H. Hu , C.Y. Huang, and CH. Hsieh, "Three-dimensional mapping of brain neuropils in the cockroach. Diploptera punctata," *J. Comp. Neurol* vol. 440, pp. 1-11, 2001

[5] H.H. Lin, J.S. Lai, A.L. Chin, Y.C. Chen, and A.S. Chiang, "A map of olfactory representation in the Drosophila mushroom body," *Cell*, vol. 128, pp. 1205-17, 2007

[6] S. Bouix, K. Siddiqi, and A. Tannenbaum, "Flux driven automatic centerline extraction," *Medical Image Analysis*, Vol. 9, pp. 209-221, 2005.

[7] M. Sonka, M. D. Winniford, X. Zhang, and S. M. Collins, "Lumen centerline detection in complex coronary angiogram," *IEEE Tran. Med. Imaging,* Vol. 41, pp. 520-528, 1994.

[8] T. Deschamps and Laurent D. Cohen, "Fast extraction of minimal paths in 3D images and application to virtual endoscopy," *Medical Image Analysis,* Vol. 5, pp. 281-299, 2001.

[9] I. Bitter, Arie E. Kaufman and M. Sato, "Penalized-distance volumetric skeleton algorithm," *IEEE Tran. Visualization and Computer Graphics*, Vol. 7, No. 3, pp. 195-205, Jul-Sep., 2001

[10] H. Li and A. Yezzi, "Vessels as 4-D curves: global minimal 4-D paths to extract 3-D tubular surfaces and centerlines," *IEEE Tran. Med. Imaging*, Vol. 26, No. 9, pp. 1213-1223, Sep., 2007

[11] K. A. Al-Kofahi, S. Lasek, D. H. Szarowski, C. J. Pace, G. Nagy, and J. N. Turner, "Rapid automated three-dimensional tracing of neurons from confocal image stacks," *IEEE Tran. Info. Tech. in Biomedicine,* Vol. 6, No.2, pp. 171-187, Jun. 2002.

[12] T. Zhao, J. Xie, F. Amat, N. Clack, P. Ahammad, H. Peng, F. Long, and E. Myers, "Automated reconstructions of neuronal morphology based on local geometrical and global structural models," *NeuroInform* vol. 9, pp. 247-261, 2011.

[13] Y. Zhang, X. Zhou, A. Degterev, M. Lipinski, D. Adjeroh, J. Yuan and S. T.C. Wong, "Automated neurite extraction using dynamic programming for high-throughput screening of neuron based assays," *NeuroImage* vol. 35, pp. 1502-1515, 2007.

[14] H. Peng, Z. Ruan, D. Atasoy, and S. Sternson, "Automatic reconstruction of 3D neuron structures using a graph-augmented deformable model," *Bioinformatics* vol. 26, pp. i38-i46, 2010.

[15] E. T*ü*retken, G. Gonzalez, C. Blum, and P. Fau, "Automated reconstruction of dendritic and axonal tress by global optimization with geometric priors," *NeuroInform* vol. 9, pp.279-302, 2011.

[16] H.Y. He and H. T. Cline, "Diadem X: Automated 4 Dimensional Analysis of Morphological Data," *Neuroinform.*, vol. 9, pp. 107-112, 2011.

[17] H. T. Cline, "Dendrite Development in Dendrites," *Oxford Univ. Press, London*, 1999.

[18] A. Holtmaat, and K. Svoboda, "Experience-dependent structural synaptic plasticity in the mammalian brain," *Nat Rev Neurosci.*, vol. 10, pp. 647-658, 2009.

[19] J.L. Chen JL, W.C.Lin, J.M. Cha, P.T. So, Y. Kubota, E. Nedivi, "Structural basis for the role of inhibition in facilitating adult brain plasticity," *Nat Neurosci.* , vol. 14,pp. 587-94, 2011.

[20] K. Haas, J. Li, H. T. Cline, "AMPA receptors regulate experience-dependent dendritic arbor growth *in vivo*," *PNAS.*, vol. 103 pp. 12127-31, 2006.

[21] H. T. Cline, K. Haas, "The regulation of dendritic arbor development and plasticity by glutamatergic synaptic input: a review of the synaptotrophic hypothesis," *J. Physiol.* vol. 586, pp.1509-17, 2008.

[22] K. Podgorski, D. Dunfield, and K. Haas, "Neuronal division of labor drives visual learning in an awake developing brain network," *PLoS Biology*, 2012.

[23] H. Makino, R. Malinow, "Compartmentalized versus global synaptic plasticity on dendrites controlled by experience," *Neuron*, vol. 72, pp. 1001-11, 2011.

[24] T. Kleindienst, J. Winnubst, C. Roth-Alpermann, T. Bonhoeffer, C. Lohmann, "Activity-dependent clustering of functional synaptic inputs on developing hippocampal dendrites," *Neuron*, vol. 72, pp. 1012-24, 2011.

[25] N. Takahashi, K. Kitamura, N. Matsuo, M. Mayford, M. Kano, N. Matsuki, Y. Ikegaya, "Locally synchronized synaptic inputs," *Science*, vol. 335, pp. 353-6, 2012.

[26] O. Al-Kofahi, R. J. Radke, B. Roysam, and G. Banker, "Automated semantic analysis of changes in image sequences of neurons in culture," *IEEE Trans. Biomed. Eng.* vol. 53, pp. 1109-23, 2006.

[27] H. Heumann and G. Wittum, "The tree-edit-distance, a measure for quantifying neuronal morphology," *Neuroinform.*, vol. 7 pp. 179-190, 2009.

[28] K. M. Brown, G. Barrionuevo, A. J. Canty, V. De Paola, J. A. Hirsch , G. S. Jefferis, J. Lu, M. Snippe, I. Sugihara, G. A. Ascoli, "The DIADEM data sets: representative light microscopy images of neuronal morphology to advance automation of digital reconstructions," *Neuroinform.*, vol. 9, pp. 143-57, 2011.

[29] C. W. Niblack, P. B. Gibbsons, and D. W. Capson DW, "Generating skeletons and centerlines from the distance transform," *CVGIP: Graphical Models And Image Processing* vol. 54, pp. 420-437, 1992.

[30] C. Pudney, "Distance-ordered homotopic thining: a skeletonization algorithm for 3D digital images," *Computer Vision and Image Understanding*, 72(3), pp. 404-413, 1998

[31] S. Boyd and L. Vandenberghe, "Convex Optimization," *Cambridge Univ. Press*, 2004.

[32] `http://www.flycircuit.tw`

[33] `http://stat.ethz.ch/R-manual/R-devel/library/stats/html/hclust.html`

[34] R. F. Stocker, M. C. Lienhard, A. Borst, and K. F. Fischbach, "Neural architecture of the antennal lobe in Drosophila melanogaster," *Cell Tissue*, vol. 262, pp. 9-34, 1990.

[35] R. F. Stocker, "The organization of the chemosensory system in Drosophila melanogaster:a review," *Cell Tissue Res.*, vol. 275, pp. 3-26, 1994.

[36] L. F. Costa, E. T. M. Manoel, F. Faucereau, J. Chelly, J. Pelt and G. Ramakers, "A shape analysis framework for neuromorphometry," *Network: Computation in Neural Systems*, Vol. 13, pp. 283-310, 2002

[37] H. B. M. Uylings and J. Pelt, "Measures for quantifying dendritic arborizations," *Network: Computation in Neural Systems*, Vol. 13, pp. 397-414, 2002

[38] P. J. Besl and N. D. MaKay, "A method for registration of 3-D shapes," *IEEE Trans. PAMI*, Vol. 14, pp.239-256, Feb. 1992

[39] J. E. Bestman, H. T. Cline, "The RNA binding protein CPEB regulates dendrite morphogenesis and neuronal circuit assembly *in vivo*," *PNAS*, Vol. 105, pp. 20494-9, 2008.

[40] S. L. Chiu, C. M. Chen, H. T. Cline, "Insulin receptor signaling regulates synapse number, dendritic plasticity, and circuit function *in vivo*," *Neuron*, Vol. 58, pp.708-19. 2008.

[41] D. Donte, F. Foggia, C. Sansone and M. Vento, "Thirty years of graph matching in pattern recognition," *IJPRAI*, Vol. 18, pp. 265-298, 2004.

[42] A.C. M. Dumay, R. J. Geest, J. J. Gerbrands, E. Jansen and J. H.C. Reiber JHC, "Consistent inexact graph matching applied to labeling coronary segments in arteriograms," *Proc.Int. Conf. Pattern Recognition*," pp. 439-442, 1992

[43] K. Haris, S. N. Efstradiatis, N. Maglaveras, C. Pappas, J. Gourassas and G. Louridas G, “Model-based morphological segmentation and labeling of coronary angiograms,” *IEEE Tran. Med, Imag.*, Vol. 18, pp. 1003-1015, 1999.

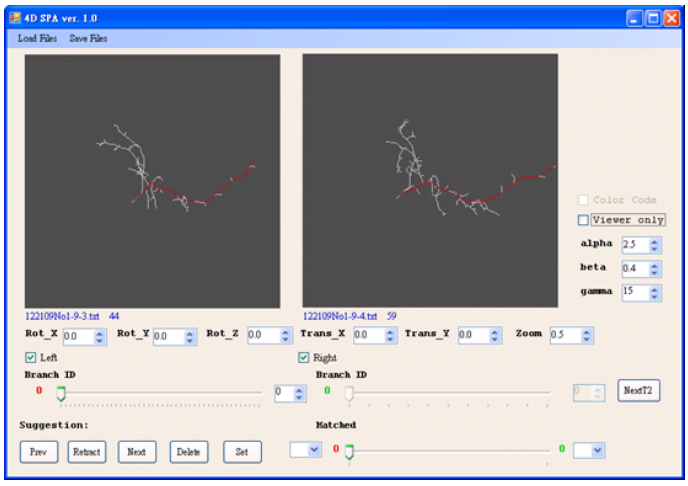


Figure I.1: The outlook of 4D SPA

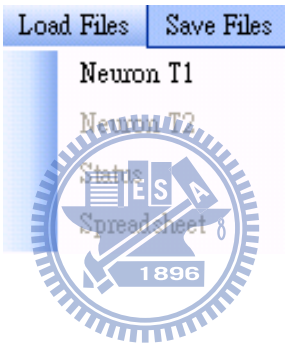


Figure I.2: The supported input file formats

Appendix I: Manual of 4D SPA

The operations of 4D SPA can be separated as three categories: **utility**, **transformation** and **main functions**. Below, we will introduce each category.

I.1 Utility

There are three types of data supported by 4D SPA: 1. Neuron data, 2. Status data, and 3. Spreadsheet. The input neuron data have to satisfy the following format:

@.IPT@.BCF //the first two lines are header.
3 0

Figure I.3: The supported output file formats

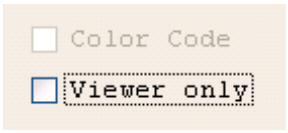


Figure I.4: Modes of 4D SPA

P index<int> n number_ of_ nodes<int>

63.4895 109.492 81.132 // the first line is always the soma position.

. . .

61.6 111.46 84 //the last line of a branch block is the tip position.

e //the terminal symbol of neuron data

The status file records the matching process so far. The function of saving 'Status' file is supported by 4D SPA. Whenever you want to continue your job, you can just load it and continue without lost any information. The spreadsheet file records the type of each branch. About how to generate the spreadsheet file please refer to Appendix II.

Three file types are supported. When the matching task is complete, the result could be output as a 'Report' for further analysis. The 'Report' is an ASCII text file. You can also record current matching state to a 'Status' file (.stat) and save the matching table to a 'Table' file (.tab). We strongly recommend the user NOT to modify the 'Status' file.

The default mode of 4D SPA is for pair-wise analysis. Three more modes, matching-pair observation, Color Code and Viewer only are supported. When the Viewer only is checked, all functions for matching are invalid but all transformation function to observe the neurons are still available. After a spreadsheet is loaded, the Color Code mode is

Figure I.5: Three parameters used in 4D SPA

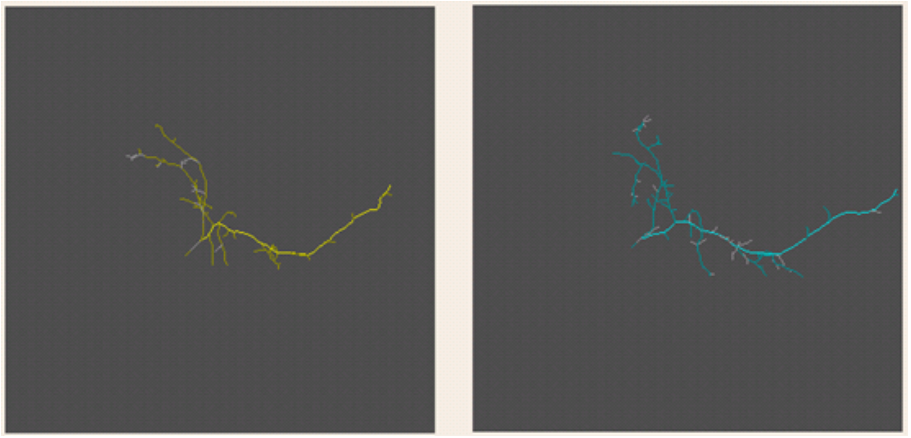


Figure I.6: Three parameters used in 4D SPA

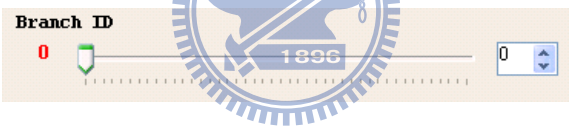


Figure I.7: Branch ID tracker

enabled. The neuron branches will be rendered in different color according to their categories when the Color Code mode is activated. The matching-pair mode will be described later.

When the user changes his/her focus on observing matched pairs, 4D SPA changes to the matching-pair mode. In this mode, all matched branches in Neuron T1 and Neuron T2 are rendered yellow and cyan, respectively, and the rest of branches are rendered light grey. The user could specify the focused matching pair by drawing the matched tracker, or changing the number of matching index of Neuron T1 or Neuron T2 in the number box.

To observe a specific branch in either Neuron T1 or Neuron T2, the user could indicate the branch by drawing the corresponding track or changing the number in the corresponding index box. If 4D SPA is in the default mode, matching mode, when the focused branch in

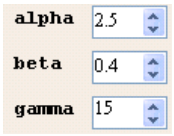


Figure I.8: Three parameters used in 4D SPA

Figure I.9: Transformation functions

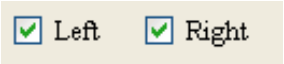


Figure I.10: Check box to indicate targets

Neuron T1 is changed, the suggestion list will be changed either. However, the changes of the focused index in Neuron T2 will not vary the suggestion list.

Three parameters used to adjust the 4D SPA. These parameters will affect the implemented similarity function. The user could adjust these parameters to optimize 4D SPA in his/her applications.

I.2 Transformation

In order to facilitate the user to observe the input neurons, rigid-body transformations and scaling function are supported. Furthermore, the user could indicate which neuron to be transformed by checking the corresponding check-box.

I.3 Main Funciton

5 functions are supported for matching process.

1. Prev: If the user made a wrong decision then he/she can restore the last matching

Figure I.11: Functions supported during the matching process

Figure I.12: Move to the next index in the suggestion list

status by pressing ‘Prev’ button.

2. Retract: Indicate a branch in Neuron T1 is retracted.
3. Next: Move to the currently second longest undefined branch.
4. Delete: To delete an already exist matching pair.
5. Set: To confirm a matching pair.

In the default mode, if the suggestion list is not empty, the user could circularly move the focused index in the suggestion list by pressing this button.

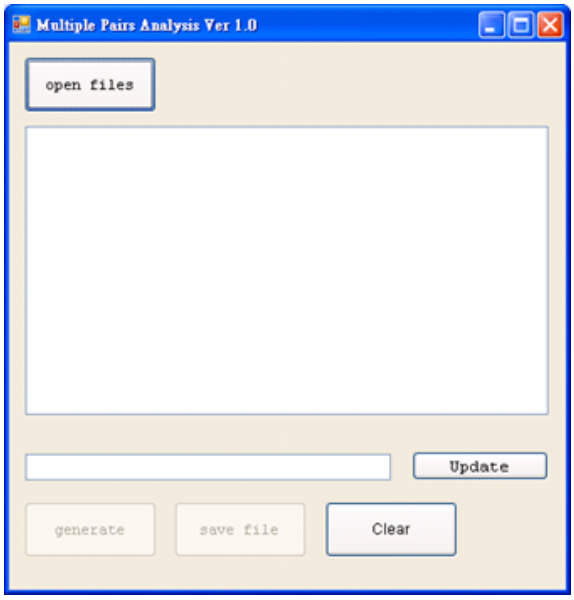


Figure II.1: The outlook of Multiple Pairs Analyzer

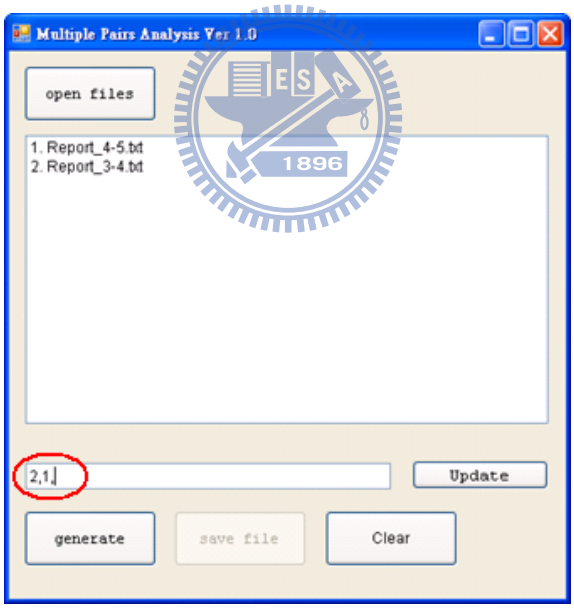


Figure II.2: Wrong input data order

Appendix II: Manual of Multiple Pairs Analyzer

The first step is pressing **open files** and loading files from indicated folder. Multiple files can be loaded at once.

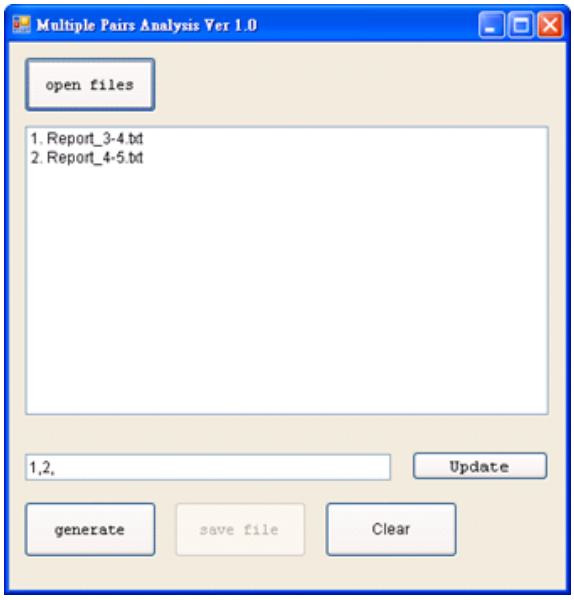


Figure II.3: Wrong input data order

The files will be loaded in the order displayed in the window. If the order is wrong, you could input the correct order, then **Update** it. Do not forget the last terminal symbol, ',' in the input sequence.

Once the file-loading order is confirmed, you could generate your report. The output file format is:

Total Time points <int>

BranchID Type T_0,. . ., T_n <double> //n equals Total Time points minus 1.

Each T_n represents the length of the BranchID branch at time point T_n.

Example:

3

0 Stable 131.658 129.849 121.516

1 Stable 130.953 128.289 120.19

. . .

94 Newly_ Added 0 12.8521 16.6947

Please **Clear** before loading another set of data, otherwise the later loaded files will be added into the already existed data set.

Publication List

Journal Paper

1. **Ping-Chang Lee**, Hsiu-Ming Chang, Chih-Yang Lin, Ann-Shyn Chiang, Yu-Tai Ching, "Constructing Neuronal Structure from 3D Confocal Microscopic Images," *Journal of Medical and Biological Engineering*, Vol. 29, No. 1, 2009
   IF: 0.731 **Rank:** 61/72 (84.5%, ENGINEERING, BIOMEDICAL)

2. Ann-Shyn Chiang, Chih-Yung Lin, Chao-Chun Chuang, Hsiu-Ming Chang, Chang-Huain Hsieh, Chang-Wei Yeh, Chi-Tin Shih, Jian-Jheng Wu, Guo-Tzau Wang; Yung-Chang Chen, Cheng-Chi Wu, Guan-Yu Chen, Yu-Tai Ching, **Ping-Chang Lee**, Chih-Yang Lin, Hui-Hao Lin, Chia-Chou Wu,, Hao-Wei Hsu, Yun-Ann Huang, Jing-Yi Chen, Hsin-Jung Chiang, Chun-Fang Lu, Ru-Fen Ni, Chao-Yuan Yeh, Jenn-Kang Hwang, "Three-dimensional reconstruction of brainwide wiring networks in Drosophila at single cell resolution," *Current Biology*, Vol. 21, pp. 1-11, 2011
   IF: 9.647 **5-Year Impact Factor:** 10.881
   Rank: 18/289 (6.2%, BIOCHEMISTRY & MOLECULAR BIOLOGY)

3. **Ping-Chang Lee**, Chao-Chun Chuang, Ann-Shyn Chiang, Yu-Tai Ching, "High-throughput Computer Method for 3D Neuronal Structure Reconstruction from the Image Stack of the Drosophila Brain and its Applications," *PLoS Computational Biology*, accepted, 2012.
   IF: 5.215 **5-Year Impact Factor:** 5.884
   Rank: 2/47 (4.3%, MATHEMATICAL & COMPUTATIONAL BIOLOGY)

4. **Ping-Chang Lee**, Hai-yan He, Yu-Tai Ching, Hollis T. Cline, "A Computer Aided System for 4D Dynamic Neuron Structural Analysis," *Neuroinformatics*, in press.
   IF: 2.973 **5-Year Impact Factor:** 2.560
   Rank: 13/99 (13.1%, COMPUTER SCIENCE, INTERDISCIPLINARY APPLICATIONS)

Conference Paper

1. **P. C. Lee**, Y. R. Huang, H. M. Chang, A. S. Chiang, and Y. T. Ching, "Computation of brain neurodegeneration in the Alzheimer's fly," *SPIE Medical Imaging*, 2006.

2. **Ping-Chang Lee**, Hsiu-Ming Cahng, Yu-Tai Ching, Ann-Shyn Chiang, "A semi-automatic method for neuron centerline extraction in confocal microscopic image stack," *ISBI*, 2008.

Patent

1. Ann-Shyn Chiang, **Ping-Chang Lee**, Yu-Tai Ching, Chao-Jun Zhuang, Hsiu-Ming Chang, Yung-Chang Chen, Chang-Huain Hsieh. "Image Processing System for the 3D Image Database Construction," **US 8,126,247 B2**, 2012.