GEOMETRIC REPRESENTATION OF NEUROANATOMICAL DATA OBSERVED IN MOUSE BRAIN AT CELLULAR AND GROSS LEVELS

A Dissertation

by

WONRYULL KOH

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2007

Major Subject: Computer Science

GEOMETRIC REPRESENTATION OF NEUROANATOMICAL DATA OBSERVED IN MOUSE BRAIN AT CELLULAR AND GROSS LEVELS

A Dissertation

by

WONRYULL KOH

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee, Bruce H. McCormick
Committee Members, Louise C. Abbott
Nancy M. Amato
John C. Keyser
Head of Department, Valerie E. Taylor

May 2007

Major Subject: Computer Science

ABSTRACT

Geometric Representation of Neuroanatomical Data Observed

in Mouse Brain at Cellular and Gross Levels. (May 2007)

Wonryull Koh, B.S., The University of Texas at Austin;

M.S., Texas A&M University

Chair of Advisory Committee: Dr. Bruce H. McCormick

This dissertation studies two problems related to geometric representation of neuroanatomical data: (i) spatial representation and organization of individual neurons, and (ii) reconstruction of three-dimensional neuroanatomical regions from sparse two-dimensional drawings. This work has been motivated by nearby development of new technology, Knife-Edge Scanning Microscopy (KESM), that images a whole mouse brain at cellular level in less than a month.

A method is introduced to represent neuronal data observed in the mammalian brain at the cellular level using geometric primitives and spatial indexing. A data representation scheme is defined that captures the geometry of individual neurons using traditional geometric primitives, points and cross-sectional areas along a trajectory. This representation captures inferred synapses as directed links between primitives and spatially indexes observed neurons based on the locations of their cell bodies. This method provides a set of rules for acquisition, representation, and indexing of KESM-generated data.

Neuroanatomical data observed at the gross level provides the underlying regional framework for neuronal circuits. Accumulated expert knowledge on neuroanatomical

organization is usually given as a series of sparse two-dimensional contours. A data structure and an algorithm are described to reconstruct separating surfaces among multiple regions from these sparse cross-sectional contours. A topology graph is defined for each region that describes the topological skeleton of the region's boundary surface and that shows between which contours the surface patches should be generated. A graph-directed triangulation algorithm is provided to reconstruct surface patches between contours. This graph-directed triangulation algorithm combined together with a piecewise parametric curve fitting technique ensures that abutting or shared surface patches are precisely coincident. This method overcomes limitations in i) traditional surfaces-from-contours algorithms that assume binary, not multiple, regionalization of space, and in ii) few existing separating surfaces algorithms that assume conversion of input into a regular volumetric grid, which is not possible with sparse inter-planar resolution.

ACKNOWLEDGMENTS

I would like to express my gratitude to my advisor, Dr. Bruce H. McCormick, who has provided the motivation, guidance, insights, and encouragements for this research. His vision, invention, steadfast conviction and optimism, generosity, and patience have shaped and nurtured this dissertation. I am grateful to Dr. Louise Abbott who has provided me with vast and valuable neuroanatomical knowledge both through her expertise and through her supply of *en bloc* stained mouse brains. I am grateful to Dr. Nancy Amato who has taught by example throughout my graduate studies how to be an excellent and generous researcher, mentor, and role model. I am grateful to Dr. John Keyser who first suggested Morse theory and has shown great interest throughout this research.

I would like to acknowledge all past and present participants of the Knife-Edge Scanning Microscopy project. I would like to thank Dr. Molly McCormick who has inculcated the importance of writing and writing well. I acknowledge the Department of Computer Science at Texas A&M University, the Texas Higher Education Coordinating Board (ATP-000512-0146-2001, J. Keyser, PI), and the National Institute of Neurological Disorders and Stroke (R01-NS54252, Y. Choe, PI) for their financial support.

Last but not least, I thank my parents and sister for their love and support.

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF FIGURES

Page

Page

CHAPTER I

INTRODUCTION

This dissertation studies two problems in the field of neuroscience that involve the representation of data having an intrinsic geometric structure. These problems are: (i) representation and spatial organization of neuronal geometry observed in histological sections at cellular resolution, and (ii) boundary surface approximation of neuroanatomical structures described by a series of their cross-sectional diagrams. Both problems arise in the study of brain due to the brain's unique structural complexity; the first problem emerges from the need to extract relevant geometric information from a massive cellular-level data set, whereas the second problem emerges from the need to interpolate missing geometric information from a sparse data set. Our solutions to both problems describe how to convert complex data into visual forms so that we can discern patterns and trends and to facilitate further analysis.

A. Motivation

In 1950, Alan Turing [1], credited with inventing modern computer science, postulated that machines could be created that would closely mimic the cognitive processes of the human brain. In 1966, participants in an NIH-sponsored workshop on image processing in biological science first proposed that computers could be used to display, manipulate, and manage three-dimensional images of the brain. In 1985, some of the participants from the NIH workshop (Clark, McCormick, Vastola, Waxman) convened another

---

This dissertation follows the style of *IEEE Transactions on Medical Imaging*.

workshop at Texas A&M University, and "the conversations and presentations in that later workshop stirred imaginations to dream of things to come of the power of computers in neuroscience" [2]. In 1995, Larry Swanson noted that the future of brain mapping lies in computer science [3]. In 2000, *Nature* reported that neuroscientists worldwide, studying everything from individual molecules to complex behaviors in species from nematode worms to humans, have created one of the largest, most unwieldy datasets in science and that they are turning to computers for help [4].

Since 1950, computers have become an important tool to study the brain, but not without initial skepticism from the neuroscience community. Critics considered the initial idea of using computational tools to study the structure and function of brain technically absurd and went on to say that even if imaging the whole brain were possible, the cost of equipment to do so would be excessive and impossible to justify. In the mid-1960s it was "only the most immoderate visionary who could imagine the exponential growth in computer power and the precipitous fall in the cost of computation and storage" to even consider the role of computers in the study of brain and its functions, recalls Jerome Cox in his forward to *Neuroinformatics: An Overview of the Human Brain Project* [2].

This dissertation has been motivated by one such "immoderate visionary" and his invention to image the whole brain, *Knife-Edge Scanning Microscopy*, that have since proven the critics wrong. The Knife-Edge Scanning Microscope (KESM) enables for the first time modeling of fundamental cellular network organization in mammalian brains by generating aligned stacks of histological sections at cellular resolution (see Section III.A). The problems studied in this dissertation originated to support acquisition, representation, organization, and understanding of neurons and their circuitry observed in KESM-generated data of a mouse brain.

B. Neuronal Geometry

Neurons, the basic functional units of our nervous system, are shaped such that each has a central cell body and a number of thin tube-like projections (called axons and dendrites) extending from the cell body. There are about 100 billion neurons in the human brain, and neuronal cell bodies together with axonal/dendritic projections make a tangled mess in human and other mammalian brains. For example, each cubic millimeter volume of mouse cortex packs on average 92,000 neuronal cell bodies, 4 kilometers of axons, 450 meters of dendrites, and 700 million synapses (connections between two neurons via axons and dendrites) [5]. The complex spatial characteristics of neuronal data pose an interesting problem in computer graphics and database design, particularly because inter-neuronal connections are commonly non-local.

To understand the brain activity, Santiago Ramon y Cajal noted in 1899, it is necessary to understand the molecular and connectional changes of neurons, not to mention the exact histology of each cortical area and all of their pathways [6]. Although the 302 neuron nervous system of the nematode worm, C. elegans, and its wiring data were mapped in 1987 [7], [8], there has not been a study of the complete structure of individual neurons and their interconnections in a mammalian brain.

In this dissertation, we present a data representation method that captures geometry of individual neurons using traditional geometric primitives—points and cross-sectional areas along a trajectory—and that captures observed synapses as directed links between the primitives. To improve storage and querying efficiency, we also present a spatial indexing method based on the location of individual neuron's cell body. Traditional spatial indexing methods—such as k-d trees [9], quad/oct-trees [10], [11], R-trees [12], and their variants [13], [14]—result in inefficiencies in querying and storage when there are large overlaps among data objects or among their bounding boxes/volumes. With

these traditional methods, a neuron's bounding volume, filled with cell bodies, axons, and dendrites of many other neurons, does not provide efficient data access.

C. Surface Approximation of Neuroanatomical Structures

Although it might seem that one could simply connect neurons together by means of synapses and make networks that mediate behavior, a general principle of biology is that any given behavior of an organism depends on a hierarchy of levels of organization, with spatial and temporal scales spanning many orders of magnitude. This is nowhere more apparent than in the construction of the brain. As applied to neuronal circuits, it means that one needs to identify the main levels of neuroanatomical organization in order to provide a framework for understanding the principles underlying their construction and function [15].

The neuroanatomical organization of brain has been studied for more than a thousand years, and more than a thousand neuroanatomical structures have been identified for humans and rodents alike. This accumulated knowledge is usually given in book form as a series of cross-sectional diagrams showing graphical outlines of standard structures and their nomenclature [16]-[18]. We introduce a surface approximation method to reconstruct the unknown object surfaces of neuroanatomical structures from a series of their cross-sectional contours. Recovering an unknown shape from a series of two-dimensional contours is an ill-posed, NP-hard problem [19], but many acceptable approximations have been proposed [20]-[31]. What warrants our new approach to this problem is that neuroanatomical structures are tightly packaged in a small space; they abut each other and tend to share the surfaces separating them. This leads to multiple regionalization [32] of space with non-manifold surfaces, distinct from the binary regionalization that traditional surfaces-from-contours algorithms expect as input.

D. Overview of Dissertation and Contribution

This dissertation makes the following contributions:

Neuronal geometry representation

This portion of the dissertation shows how to represent geometry of individual neurons using traditional geometric primitives, how to represent observed synapses as directed links between the primitives, and how to index neuronal geometry data based on the location of individual neuron's cell body.

Surface approximation of neuroanatomical structures

This portion of the dissertation shows how to reconstruct precisely coincident, geometrically consistent surfaces, from sparse cross-sectional diagrams, that separate multiple neuroanatomical regions in space.

The dissertation has three major parts:

(1) Chapter II provides a brief survey of existing whole brain imaging methods, three-dimensional models of neuroanatomical structures, and anatomically-oriented neuroscience databases.

(2) Chapter III and Chapter IV describe methods developed to support KESM in the acquisition of cellular-level data. Chapter III prescribes the KESM data acquisition protocol, and establishes a three-dimensional coordinate system for data organization and indexing. Chapter IV defines a set of terminology and rules needed to model neuronal geometry, and shows how to apply this set to represent cellular-level data observed using different stains.

(3) Chapter V and Chapter VI address neuroanatomical data at a more gross anatomical level. Chapter V introduces a topology-based data structure for neuroanatomical regions and an algorithm to reconstruct their separating surfaces from cross-sectional contours using parametric curve-fitting technique

and triangulation. Using these methods, Chapter VI presents experimental results for the boundary surface approximation of anatomical structures in mouse brain olfactory bulb, and application of our results to align a KESM-generated histological volume.

Chapter VII gives a summary and discussion of future work.

CHAPTER II

SURVEY OF THREE-DIMENSIONAL HISTOLOGY, NEUROSCIENCE DATABASES, AND THREE-DIMENSIONAL MODELS OF NEUROANATOMY

In this chapter, we present a brief survey of three-dimensional histology, neuroscience databases, and three-dimensional models of neuroanatomy. The survey reflects our attempt to find available neuroscience resources to efficiently manage KESM-generated data sets of a whole brain. Instead, we found a wealth of resources that tend to focus on studying individual or small ensembles of neurons in great detail. Thus, they are not yet directly applicable to our problem related to KESM, but are nevertheless related because we need to rely on their domain expert knowledge in order to gain insight from KESM-generated data sets.

Our survey is not meant to be a comprehensive review. It is also largely independent of the remaining chapters in this dissertation although some of the materials surveyed here appear as prior work in later chapters.

A. Three-Dimensional Histology

Traditionally, postmortem histological analysis of a thick tissue specimen requires sectioning the tissue, examining the cut sections under the microscope, and extrapolating meaningful three-dimensional information from the 2D sections. An alternative is cryoplaning with digital image capture, that produces an aligned series of cryosectioned specimen surface images at approximately 200 μm spatial and axial resolution [33]. Imaging the newly planed blockface was soon extended to the construction of 3D mouse maps, typically at 10-20 μm resolution. The invention of confocal microscopy [34] introduced three-dimensional imaging capability, the ability

to form a three-dimensional image of an object by producing a series of optical sections. The main drawback of confocal microscopy has been its wasteful use of excitation, which is particularly problematic in fluorescence microscopy due to photobleaching and photodamage of the tissue being examined [35]. This problem has been solved by 2-photon laser scanning microscopy [36], that limits the fluorescence excitation to the focal slice. Using confocal and multi-photon laser scanning microscopy, tissue sections that are more than 100 μm thick [37] can be imaged at submicron spatial and axial resolution [38]. Presently, electron microscopy provides the highest spatial resolution at less than 5 nm, and when combined with electron tomography, produces a single or double axis tilt series of tomograms from a 0.5-4 μm thick section [39], [40].

Although electron tomography and confocal and multiphoton laser scanning microscopy provide three-dimensional imaging capability, when studying a whole mouse brain, their relatively thin imaging depth—approximately 4 μm for electron tomography and 25-100 μm for confocal and multiphoton laser scanning microscopy, respectively—requires sectioning and registration of cut sections. Currently, there are two techniques that enable imaging an entire mouse brain in three dimensions and obviate the need to register the cut sections—knife-edge scanning [41] and all-optical histology using ultrashort laser pulses [42].

A brief introduction to *Knife-Edge Scanning Microscopy* is given in Section III.A.

The all-optical histology technique provides diffraction-limited volumetric data that are used to reconstruct the architectonics of labeled cells of microvasculature. It makes use of successive iterations of imaging with two-photon laser scanning microscope (TPLSM) and tissue ablation with ultrashort pulses of infrared laser light. The sequence repeats serially until the desired volume of tissue has been analyzed. This leads to a digitized block of optical sections from the labeled tissue.

B. Neuroscience Databases

The success of experimental neuroscience has brought with it a problem that the quantity and complexity of relevant data, and their dispersion through an extensive literature, make it very difficult to derive reliable conclusions about the information they collectively bear about the nervous system. These complex and numerous data require extensive analysis in order to develop and substantiate hypotheses about the organization and possible structure-function relationships in the brain. Before analysis can begin, however, relevant data must be brought together into an empirically faithful but tractable form. Hence, computer-based collation, management, and analysis of neurobiological data—an approach known as 'neuroinformatics' by direct analogy to 'bioinformatics'—is a necessary step to make the complex data more tractable, leading to the development and appropriate testing of better-informed hypotheses [43].

The current scope of neuroscience databases ranges from data inventories for personal use and specialized data collations by a sub-community to large-scale database projects of general interest [44]. The neuroscience databases reviewed in this section are divided into five classes: (1) collections of structural and functional data, (2) image databases, (3) ontologies of neuroscientific objects, (4) morphology databases, and (5) databases of computational models and their components. Each class of neuroscience databases is briefly reviewed below.

1. Collections of structural and functional data

A first generation of connectivity databases has been constructed in the last fifteen years for the macaque monkey [45], [46], cat [47], [48] and rat [49]. In describing possible structure-function relationships in the brain, there are two fundamentally different approaches to describing the localization of brain data [44]. The first approach references neuroanatomical entities such as neuronal cell types, columns, layers, or

areas. The alternative approach specifies a three-dimensional reference system. These two concurrent approaches are exemplified by the connectivity databases CoCoMac [50] and XANAT [51], respectively.

Two of the attempts in human brain mapping are BrainMap [52], [53]—a software environment for meta-analysis of the human functional brain-mapping literature, and the European Computerised Human Brain Database (ECHBD) [54]—a 3D computerized database for relating function to microstructure of the cerebral cortex of humans. For examining volumes of brain tissue at nanometer resolutions, the Cell Centered Database (CCDB) [40], [55] houses high resolution 3D light and electron microscopic reconstructions spanning the dimensional range from $5nm^3$ to $50um^3$; the SynapseWeb [56], [57] provides an interface for interactively examining volumes of brain tissue to study synaptic connections and supporting structures in the gray matter that can be fully visualized only through 3D electron microscopy. The Biomedical Informatics Research Network (BIRN) is a distributed information technology infrastructure initiative to enable researchers to collaborate on large-scale studies of human disease with multi-resolution tools [58]

2. Image databases

The fMRI Data Center serves as a repository for imaging data which underlies peer-reviewed, published fMRI studies [59]. The Mouse Brain Library (MBL) consists of uniformly processed section images (3060 x 2036 pixels, 25 μm per pixel) and databases of brains from many genetically characterized strains of mice [60].

3. Ontologies of neuroscientific objects

Three active projects for ontologies of neuroscientific objects include NeuroML [61]—a markup language effort for neurosciences, BrainML [62]—an open and non-formal

functional ontology for neuroscience for interoperability among neuroscience resources, and Common Data Model [63] for neuroscience and biophysical data archiving and exchange.

4. Morphology databases

Two types of morphology databases are available: biologically observed and virtually generated. An on-line archive of neuronal geometry [64] houses full three-dimensional representations of 87 neurons from the rat hippocampus, obtained following intracellular staining with biocytin and reconstruction using Neurolucida [65]. Ascoli et al [66] have virtually generated anatomically plausible neurons for several morphological classes, including cerebellar Purkinje cells, hippocampal pyramidal cells and interneurons, and spinal cord motor neurons.

5. Databases of computational models and their components

There are two widely used neural simulation packages: GENESIS and NEURON. GENESIS (GEneral NEural SImulation System) and its reimplementation in C++, MOOSE (Messaging Object Oriented Simulation Environment), [67] are a general purpose simulation platform that supports the simulation of neural systems ranging from complex models of single neurons to simulations of large networks made up of more abstract neuronal components. NEURON [68] is also a simulation environment for neurons and networks of neurons. It is particularly well-suited to problems where cable properties of cells play an important role, and where cell membrane properties are complex, involving many ion-specific channels, ion accumulation, and second messengers.

C. Cartography of Brain in Three-Dimensions

The use of atlases within experimental neuroscience provides an essential global neuroanatomical framework where data from different experiments may be brought into register [69]. One of the earliest of digital brain atlases was BRAIN BROWSER [70] for the rat brain, that was distributed as a Macintosh HyperCard stack acting as a non-centralized desktop application. Digital atlases of brain have since been developed for several species. In this section, digital atlases are divided into three classes: (1) surface based atlases, (2) volumetric atlases of mouse brain, and (3) multimodal atlases.

1. Surface-based atlases

Surface Management System (SuMS) [71] provides surface-based atlases for cerebral and cerebellar cortex of human, macaque, rat, and mouse. The atlas data sets include a variety of partitioning schemes for macaque and human cortex, fMRI activation patterns, and Talairach stereotaxic foci. SuMS also includes CARET (Computerized Anatomical Reconstruction and Editing Toolkit) [72] for viewing, manipulating, and analyzing surface reconstructions of the cerebral and cerebellar cortex.

2. Volumetric atlases of mouse brain

In volumetric atlases, each voxel is assigned a unique label usually by color- or intensity-encoding; segmented anatomical structures are represented as a set of color-coded voxels. The High Resolution Mouse Brain Atlas [73] is based on *The Atlas of the Mouse Brain and Spinal Cord* [74], and groups segmented anatomic structures by color-encoded voxels. Kovacevic et al [75]'s minimal deformation atlas was produced by averaging MRM images of nine mouse brains. Ma et al [76] constructed a database based on the MRM images of 10 mouse brains, that offer three types of digital atlases—individualized, minimal deformation, and probabilistic. The mouse brain atlas from

Allen Institute for Brain Science [77] is also a color-coded digital brain atlas constructed to map gene expression.

3. Multimodal atlases

Multimodal atlases combine data acquired from different imaging modalities. The International Consortium for Brain Mapping (ICBM) [78]'s probabilistic atlas of human brain combines multi-spectral MRI studies from its 5800 subjects, and functional imaging studies employing functional MRI, PET and event-related potentials from a subset of subjects. Mackenzie-Graham et al [79] report their group's development of a multi-modal imaging brain atlas [80] that co-registers histologically processed and annotated sections with magnetic resonance microscopy (MRM) images of the same mouse brain both *in vivo* and post-mortem.

CHAPTER III

A PROTOCOL FOR KESM DATA ACQUISITION AND ORGANIZATION

In this chapter, we present a data acquisition and organization protocol for the capture of aligned image stacks from a specimen using KESM [41], a new technology that allows imaging of a large volume of tissue, e.g., a whole mouse brain, at cellular resolution. Imaging a whole mouse brain at cellular resolution has been difficult due to two reasons. First, the Nyquist sampling theorem [81] states that to be able to resolve a region of interest (ROI) in a sampled image, the sampling resolution must be half of ROI size or less. Second, the size of the field of view (FoV) of a microscope objective at a magnification necessary to obtain images at cellular resolution is only a small fraction of the dimension of the whole mouse brain. Thus, multiple images need to be collected from a brain specimen and then put back together to produce a histological volume corresponding to the specimen. KESM generates multiple, aligned images at submicron resolution with 0.625 mm effective FoV across a brain specimen that has 15 mm x 9 mm x 6 mm dimensions. Our protocol relies on the translated coordinates from the high precision positioning stage of KESM, and prescribes how to collect and organize adjacent images to form a corresponding composite mosaic volume of an imaged specimen.

A. Knife-Edge Scanning Microscopy

The Knife-Edge Scanning Microscope (KESM) [41], [82]-[84] consists of three components: precision positioning stage, microscope/microtome assembly, and imaging system.

The *precision positioning stage* has 20 nm encoder resolution for X- and Y-axes and 25 nm encoder resolution for Z-axis. It provides accurate translation of a mounted specimen and thus ensures alignment of sectioned images.

The *microscope/microtome assembly* consists of a custom-made diamond knife, a white light source, and a microscope objective. As the diamond knife sections a thin strip of tissue, the white light source illuminates the tissue at the diamond knife tip. The microscope objective, aligned perpendicular to the top facet of the knife, images the light reflection from the knife-edge, as illustrated in Figure 1.

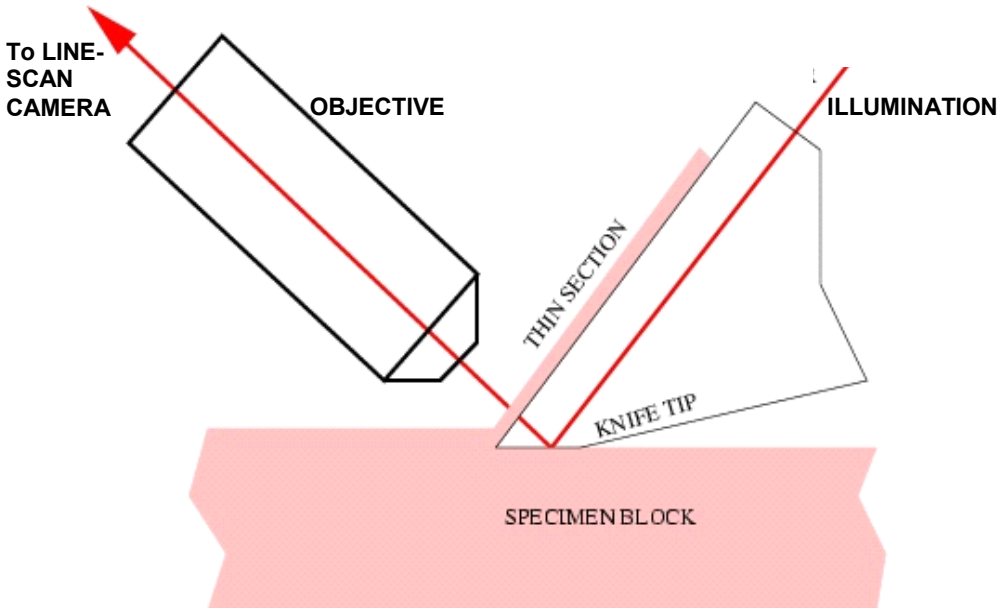


Fig. 1. Specimen undergoing knife-edge scanning.

Newly cut tissue is imaged with the microscope objective (the thickness of thin section is exaggerated). Tissue is illuminated using diamond knife as collimator for illumination from the white light source.

The *imaging system* consists of a high-sensitivity line camera operating at 45kHz that captures the image of the newly-cut section just beyond the knife-edge, prior to its subsequent deformation.

B. Data Acquisition and Organization

A specimen to be imaged by KESM is *en bloc* stained [85] and then embedded in plastic prior to sectioning. The effective FoV of the microscope objective determines the width of thin sections that are concurrently cut and imaged. KESM uses repeated knife-edge scanning to create a sequence of nested ascending or descending stair steps in specimen (see Figure 2) so as not to damage adjacent tissue lying on the same sectioning plane but not covered by the current FoV.

Our method collects and organizes adjacent images as a set of image stacks. Image stacks are indexed by their relative X, Y, Z positions within the specimen, and images in each image stack are in turn indexed by their relative depth within the image stack. We record the relative X, Y, Z positions during data acquisition from the translation coordinates of the positioning stage. Our protocol consists of a set of terminology, coordinate systems for stage and specimen, scanning conventions, and image naming and indexing rules.

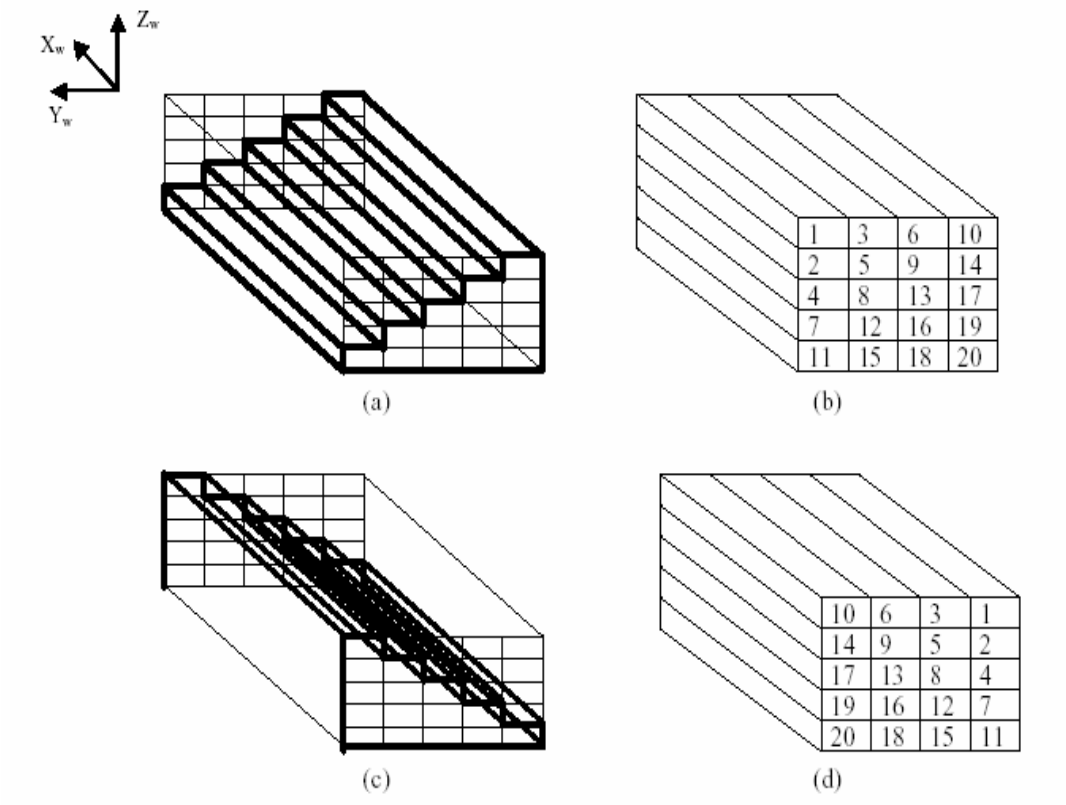


Fig. 2. Stair-step cutting.

(a) Ascending staircase constructed by removal of planks; (b) Sequence for plank cutting and scanning for ascending staircase; (c) Descending staircase constructed by removal of planks; (d) Sequence for plank cutting and scanning for descending staircase.

1. Terminology

Specimen

Specimen refers to a whole or part of *en bloc* stained and then plastic-embedded tissue molded on a specimen mount. Current specimen size is limited to 15 mm in length (X-axis), 9 mm in width (Y-axis) and 6 mm in height (Z-axis). A specimen fits within the well of a mold that binds the specimen to a detachable specimen ring, which keys the

mount to the specimen tray atop the positioning stage of KESM. This arrangement allows removal and replacement of the mount without loss of registration.

Plank, plank width, plank depth

Plank is a stack of images cut as one stair step during the stair-step cutting process (see Figure 3).

Physically, *plank width* is determined by an effective FoV of the microscope objective. With the current 40X objective, the plank width is approximately 0.625 mm; with the 10X objective, approximately 2.5 mm. *Plank depth* is physically computed by multiplying the number of consecutive sections forming the plank by their uniform section thickness. With the current 40X objective, sections are cut at approximately 0.5μm thickness; with the 10X objective, at 1μm thickness.

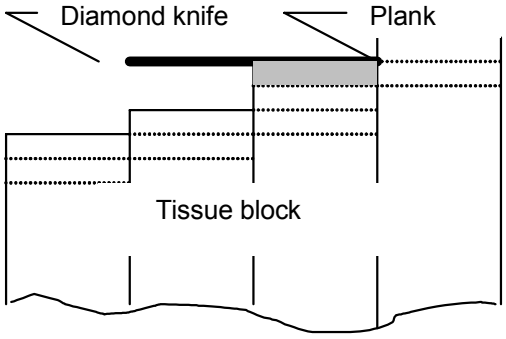


Fig. 3. A plank during stair-step cutting.
A plank with plank width of half knife-width being cut during stair-step cutting is shown in gray.

As a stack of sampled images, the *plank width* is determined by the resolution of line-scan camera. In the current KESM configuration, with monochrome camera, plank width is 4096 pixels; with color camera, 2048 pixels. All images of a plank start and stop at the same X- and Y-positions within the positioning stage, and have equal length, the full length of the specimen.

Block

Block is a stack of square images created by uniformly dividing the length of each plank by the plank width. For example, a plank that is 15mm long and 0.625mm wide is partitioned into 24 blocks.

When the plank depth matches plank width, a block forms a nearly cubical histological volume. A conceptual illustration of blocks covering a specimen is shown in Figure 4.

A 15mm x 9mm x 6mm specimen imaged by the current color camera with 0.625 mm effective FoV and 0.625 mm plank depth can be partitioned into 3600 (24 x 15 x 10) blocks. Each block, a $(0.625\text{mm})^3$ cubical volume, can contain up to 15.7 GB of data (1250 of 2048 x 2048 RGB images)—some blocks may not contain meaningful information as they fall outside the range of tissue within the specimen. The total data size of a composite volume of all blocks corresponding to a specimen, then, can in theory be as large as 57 TB. Our protocol described in this chapter guides this massive data set to be organized into manageable units.

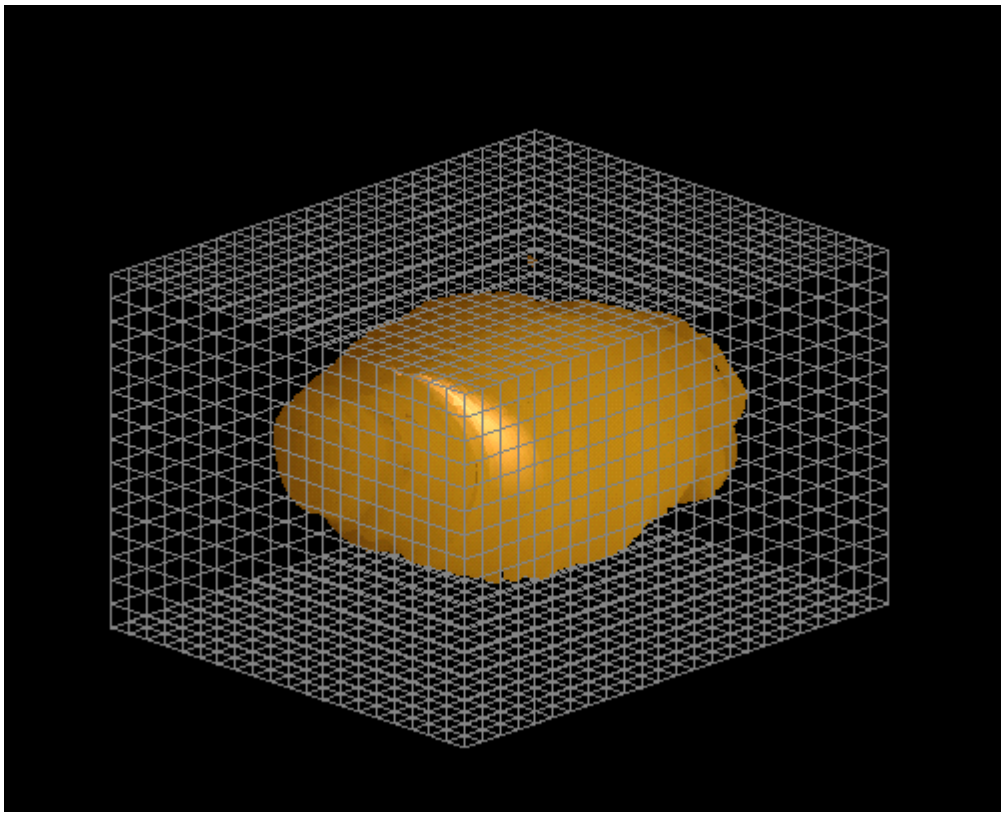


Fig. 4. Blocks covering a specimen.

2. Stage coordinate system

The precision stage conventions for home positions and directions of travel were set at the time of manufacture. The conventions (see Figure 5) are:

- X_s-axis stage: X_s-axis homes to the left end of the stage and positive motion is to the right.
- Y_s-axis stage: Y_s-axis homes to the front and positive motion is toward the rear.
- Z_s-axis stage: Z_s-axis homes at the bottom and positive motion is upward.

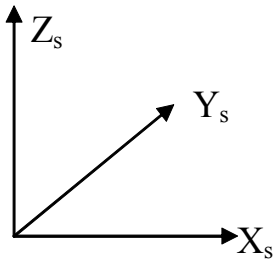


Fig. 5. Stage coordinates.

3. Specimen coordinate system

Computer graphics conventions are used to set the world coordinates for the specimen (see Figure 6): the three-dimensional workpiece is viewed as if by a camera positioned above the workspace, looking downward along the stage Z_s-axis. The world coordinate system is right-handed and the object (workpiece) is considered to reside in the negative-Z half-space. We use X_w, Y_w, and Z_w for the coordinates of the volume data set generated from the workpiece:

- X_w-axis: home is at workpiece right, increasing to workpiece left
- Y_w-axis: home is at workpiece rear, increasing toward the workpiece front
- Z_w-axis: home is at workpiece top, increasing upward (workpiece is in negative Z half-space)

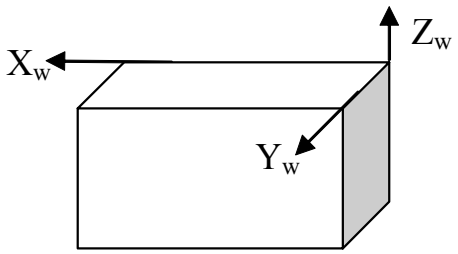


Fig. 6. Specimen coordinates.

4. Scanning conventions

A specimen is scanned along the positive X_w-axis as the air-bearing X_s-axis stage moves left-to-right. Scanning conventions for mapping stage coordinates to specimen coordinates are:

- X-axis scanning: increment X_s-axis position of stage. Map X_s-coordinate to X_w by translation
- Y-axis scanning: increment Y_s-axis position of stage. Map Y_s-coordinate to Y_w by translation
- Z-axis scanning: increment Z_s-axis position of stage. Map Z_s-coordinate to Z_w by inversion and translation.

Figures 7 and 8 show the mouse brain orientations for coronal and sagittal sectioning.

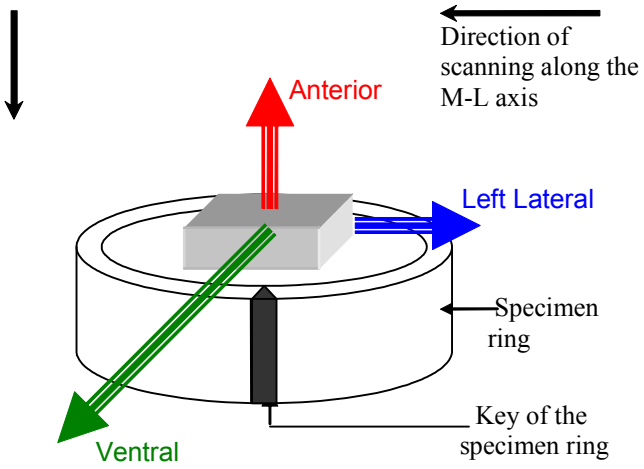


Fig. 7. Mouse brain orientation for coronal sectioning.

Drawn as seen from standing in front of the positioning stage, and not drawn to scale.

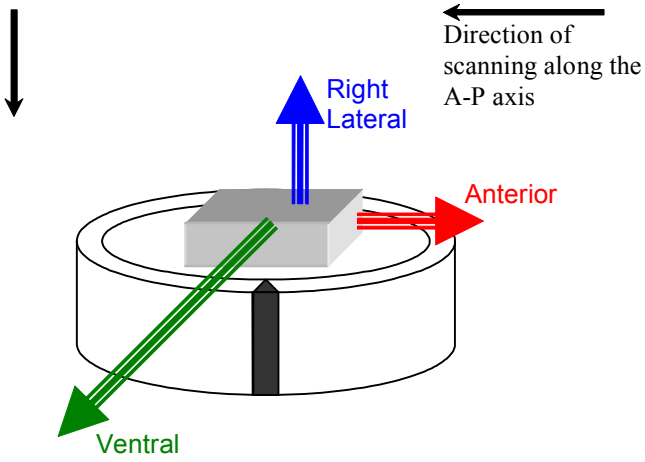


Fig. 8. Mouse brain orientation for sagittal sectioning.

Drawn as seen from standing in front of the positioning stage, and not drawn to scale.

5. Naming rules

We partition a specimen into P x Q x R blocks where

$$P = \frac{\text{specimen length}}{\text{plank width}},$$

$$Q = \frac{\text{specimen width}}{\text{plank width}}, \text{ and}$$

$$R = \frac{\text{specimen depth}}{\text{plank depth}}$$

A block within a specimen is uniquely identified by a three-dimensional (p, q, r)-index, where $0 \le p < P$, $0 \le q < Q$, $0 \le r < R$, and p, q, and r increase along the directions shown in Figure 9. The X-index increases from right to left along the positive X_w-axis, and the Z-index increases from top to bottom along the negative Z_w-axis. The Y-index increases from front to back along the negative Y_w-axis for ascending stair-step cutting, and from back to front along the positive Y_w-axis for descending stair-step cutting.

Each image in a block is also uniquely identified according to its Z-position, z, within the specimen by a (p, q, r, n)-index where

$$n = \frac{|z|}{\text{section thickness}} \% \frac{\text{plank depth}}{\text{section thickness}}$$

We name each image from a specimen I-p-q-r-n.xxx where I is an identifier referring to the specimen and xxx is a placeholder for image file format suffix. We also prescribe that images sharing same I-p-q-r prefixes be stored together within a same image directory to facilitate later parallel processing of images and block-based spatial queries.

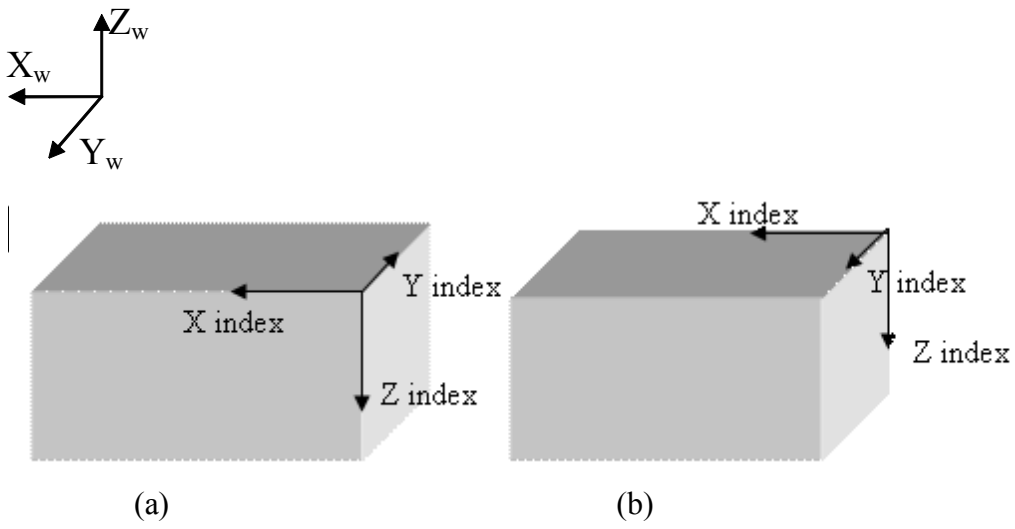


Fig. 9. Indexing directions for a specimen.

(a) shows the directions for ascending stair-step cutting, and (b) for descending stair-step cutting.

CHAPTER IV

REPRESENTATION OF NEURONAL GEOMETRY

A. Introduction

Neurons, the basic units of our nervous system, are shaped such that each has a central cell body and a number of thin tube-like projections (called axons and dendrites) extending from the cell body. Neurons receive and integrate a diverse array of incoming information, mostly from other neurons, via chemical receptors located mostly on their soma and small protrusions of their dendritic surfaces, called spines. Dendrites rarely extend more than a few millimeters from the cell body [86].

Information flows from the tertiary and secondary dendrites into the main dendrites and then to the cell body in a neuron. When activated, the cell body relays this information to other neurons via the axon by firing of an action potential. The action potential traveling on the axon enables the neuron to communicate rapidly with other neurons over sizable distances, sometimes more than a meter away depending on the size of the individual [86]. The axon branches into a series of terminals that form connections, called synapses, with the dendrites and cell bodies of other neurons, and occasionally with other axons.

A cubic millimeter volume of mouse cortex contains on average 92,000 neuronal cell bodies, 4 kilometers of axons, 450 meters of dendrites, and 700 million synapses; each neuron in mouse cortex is pre-synaptic to 7,000–8,000 neurons and postsynaptic to 6,000–10,000 neurons, and multiple synapses between the same two neurons are rare [5]. Although there is consensus about these numbers among neuroscientists, the numbers come from sparse statistical estimates. KESM enables cellular-level observation and exploration of such complex data.

In this chapter, we study the representation of neuronal geometry observed in KESM-generated histological sections of a brain specimen. We first define a set of terminology and then show how to apply this set to represent neuronal geometry data observed using different stains.

B. Terminology

We assume that histological sections are obtained following the data acquisition and organization protocol given in Chapter III. We also use terminology defined in the Chapter III without reintroducing it here.

Point

A *point* represents a cross-section of a neuronal *segment* (described below) sampled along its trajectory. A point is represented by $\begin{cases} C = (C_x, C_y, C_z) \\ \vec{n} = (n_x, n_y, n_z) \\ \vec{m} = (m_x, m_{y,} m_z) \\ \{(r_1, \theta_1), \cdots, (r_m, \theta_m)\} \\ B = (i, j, k) \end{cases}$

where

- C is the x-, y-, and z-coordinates of its center,
- $\vec{n}$ is its unit normal vector,
- $\vec{m}$ is its unit semi-major axis vector,
- r_i is the radius of its cross-sectional extent in the direction of $R_{\vec{n}}(\theta_i) \cdot \vec{m}$ where $R_{\vec{n}}(\theta_i)$ represents a rotation about $\vec{n}$ by a positive angle θ_i, $i = 1,...,m$, and $\theta_i < \theta_{i+1}$.
- B is the index of the block it is sampled from.

Segment

A *segment* is part of a traced neuron entirely within a *block*. A segment can be a neuronal cell body, or a part of a neuronal cell body or a neuronal process. We represent a segment by

$$\begin{cases} P = \{p_1, p_2, ... p_n\} \\ T, \quad T \in \{soma, dendrite, axon, unknown\} \\ n\_sb \\ \{(x_i, y_i, z_i)\}, \quad 0 \le i \le n\_sb \end{cases}$$

where

- P is the set of two or more sampled *points*, $\{p_1, p_2, ..., p_n\}$, ordered from one end of the segment to the other end,
- T is the segment's morphological type,
- n_sb is the number of observed dendritic spines (if the segment's type is dendrite), or the number of observed boutons (if the segment's type is axon).
- $\{(x_i, y_i, z_i)\}$ is the set of (x,y,z)-locations of observed spines if the segment's type is dendrite, and the set of (x,y,z)-locations of observed boutons if the segment's type is axon.

Vertex

A *vertex* is a *point* that is also an endpoint of a *segment*. We represent a vertex by

$$\begin{cases} p \\ Seg \\ boundary\_flag \end{cases}$$

where

- p is a pointer to its point representation,
- Seg is a pointer to the segment whose one endpoint is the vertex, and
- boundary_flag is a Boolean flag indicating whether the vertex lies at the boundary of a *block*.

Near-collision

A *near-collision* refers to a probable synapse between two *segments* based on their proximity. A near-collision is represented by $\begin{cases} S = \{S_x, S_y, S_z\} \\ Seg_1, Seg_2 \\ T \end{cases}$

where

- S is the x-, y-, and z-coordinates of its observed location,
- Seg_1 and Seg_2 are the two segments participating in the near-collision, and
- T is the type of synapse, e.g., axo-dendritic, axo-axonic, dendro-dendritic, etc., determined by w and also by the morphological types of Seg_1 and Seg_2.

Inter-block processing

During *inter-block processing*, the *vertices* from two neighboring *blocks*, lying on or close to block's bounding faces are matched to discover connected *segments* between the blocks.

C. Representation of Neuronal Geometry

Freshly prepared brain tissue has a uniform appearance under a microscope yielding no differences in pigmentation to enable an observer to resolve different cellular structures. Thus, neuroscientists stain brain tissue to selectively color some, but not all, parts of the cells. Two widely used stains are Nissl stain and Golgi stain. The Nissl stain, introduced by Franz Nissl in the late nineteenth century, stains the DNA in nuclei of all cells and clumps of RNA surrounding the nuclei of neurons [87]. The Golgi stain, introduced by Camillo Golgi also in the late nineteenth century, stains a small percentage of neurons in their entirety [87]. In this section, we show how to represent neuronal geometry observed in Nissl-stained and Golgi-stained brains using our terminology defined in the previous section.

A Nissl-stained specimen data set yields the full morphology of neuronal cell bodies, but not their processes. We represent the geometry of a Nissl-stained neuron as a set of *points*, *segments*, and *vertices* representing the cell body. This representation also provides the index of the block within which a cell body resides, and thus the location of neuron within the brain volume (see Figure 10).

A Golgi-stained specimen data set yields selective neurons in their full morphology. We represent the geometry of a Golgi-stained neuron as a set of *points*, *segments*, *vertices*, and *near-collisions*. This representation provides neuronal morphology information, location of neuron's cell body if ascertained, and location of observed synapses (see Figure 10).

Inter-block processing applies to Nissl-stained and Golgi-stained data sets to discover connected components of neuronal geometry (see Figure 10).

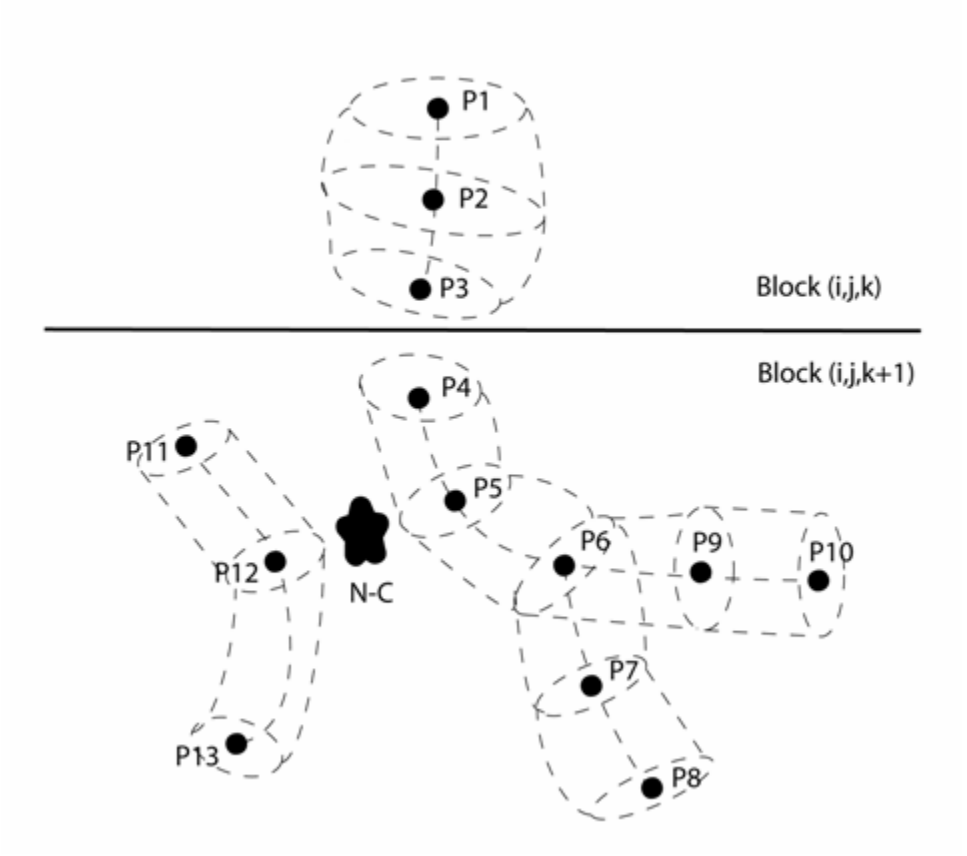


Fig. 10. Conceptual illustration of neuronal geometry representation.

The geometry of a Nissl-stained neuron can be represented as {{P1, P2, P3}, {S1}, {P1, P3}} where the segment S1 = {{P1, P2, P3}, *soma*, 0, ϕ}.

The geometry of a partial Golgi-stained neuron shown on bottom right can be represented as {{P4, P5, … , P10}, {S2, S3, S4}, {P4, P6, P7, P8, P9, P10}, {N-C}} where S2 = {{P4, P5, P6}, *dendrite*, 0, ϕ}, S3 = {{P7, P8}, *dendrite*, 0, ϕ}, and S4 = {{P9, P10}, *dendrite*, 0, ϕ}.

An observed near-collision N-C can be represented as {(N-Cx, N-Cy, N-Cz), S2, S5, *axo-dendritic*} where S5 = {{P11, P12, P13}, *axon*, 0, ϕ}.

During the inter-block processing between block (i,j,k) and block (i,j,k+1), two vertices, P3 and P4 can be matched to determine that S1 and S2 are connected components of a same neuron, yielding a 'merged' geometry of a partial Golgi-stained neuron represented by {{P1, P2, … , P10}, {S1, S2, S3, S4}, {P1, P3, P4, P6, P7, P8, P9, P10}, {N-C}}.

CHAPTER V

APPROXIMATION OF SEPARATING SURFACES FROM CROSS-SECTIONAL CONTOURS

A. Introduction

The problem of reconstructing object surfaces from cross-sectional contours has been studied for more than thirty years, and has many well established solutions (see Section V.B). These solutions, however, place strong assumptions on the input. These solutions assume that each contour from a cross-section is simple and closed, and that the contour divides a two-dimensional cross-section into two regions: inside and outside of the contour. The reconstructed surface, by tiling the contours on successive planes, forms a boundary of a manifold that divides a three-dimensional volume into two regions: inside and outside of the manifold.

The problem addressed in this chapter is reconstruction of separating surfaces from sparse cross-sectional line drawings or diagrams depicting multiple regions. Our solution places no assumptions on the input. We assume that the curves that make up each cross-sectional diagram can be i) simple and open, ii) not-simple and closed, or iii) not-simple and open as well as iv) simple and closed (see Figure 11). Our solution consists of four steps:

(1) break the input curves into simple contours such that each simple contour lies on a boundary between exactly two regions.
(2) construct a topology graph from the contours obtained after step (1).
(3) apply a curve fitting algorithm to convert the contours into parametric B-splines.
(4) generate triangulated surfaces based on our topology graph and on the evaluated points of the B-spline contours.

Steps (1) and (2) are user-guided manual processes; steps (3) and (4) are automated processes.

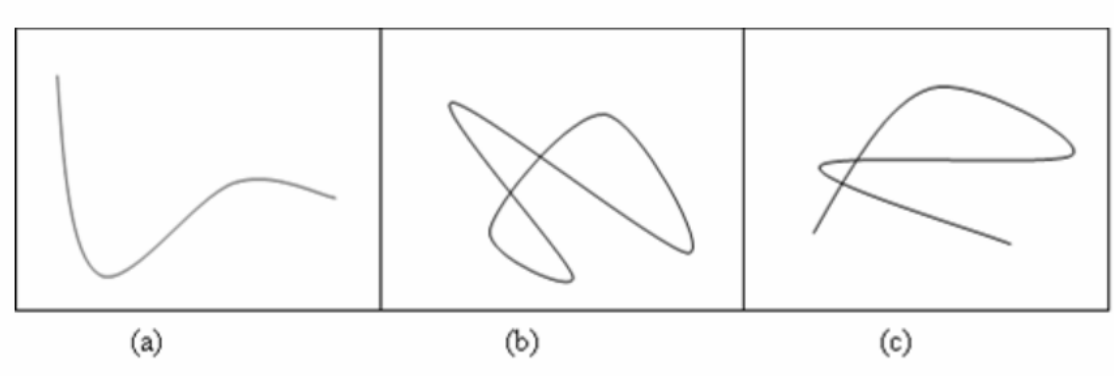


Fig. 11. Acceptable input curves.
Our solution can handle (a) open simple curves, (b) closed non-simple curves, (c) open non-simple curves, as well as closed simple curves.

The advantage of our solution is that it results in geometrically consistent separating surfaces between regions because our topology graph maintains a consistent number of geometric points used during triangulation as well as directing which contours are to be triangulated together. This is not possible with traditional surfaces-from-contours algorithms because traditional algorithms produce independent triangulations for each closed region and thereby result in interpenetrating surfaces that are not precisely coincident when the boundary surfaces of two or more regions abut. A disadvantage of our solution is that our topology graph has to be constructed manually. However, it has been shown that no fully automatic conversion from a graphical sketch to a polyhedral model can exist [88], and also that in dealing with complicated input, it requires very little extra effort from an expert to specify the continuation of objects from one section

to next [89]. A strategy for possible automation of topology graph construction process is discussed in Chapter VII.

We describe a data structure and an algorithm for computing triangulated surfaces which separate regions of different types. We assume that we have a collection of contours C_{ij}, $i = 1,...,n$, $j = 1,...,nc_i$ where n is the number of cross-sections and nc_i is the number of contours in the i-th cross-section. Our goal is to produce triangulated surfaces that separate the components of different regions R_k, $k = 1,...,M$. We assume that each contour has been classified to one or more of several classes B_k, $k = 1,...,M$ denoting the boundary of R_k.

These separating surfaces can be viewed as a generalization of the polyhedral surfaces often associated with the surfaces-from-contours algorithm (see [23]-[30]). In the context of the surfaces-from-contours algorithm the closed simple contours on each cross-section represent the boundaries between two possible classes: inside or outside of a structure. The reconstructed polyhedral surface then distinguishes these two regions. In the more general situation where there are several possible classes for internally partitioned structures or substructures, the separating surface is defined as S = $\bigcup_{i,j=1,...,M,i\neq j}(R_i \cap R_j)$. This more general separating surface is fundamentally different from a polyhedral surface from a surfaces-from-contours algorithm in that it may contain regions where three or more surface segments join (see [90]). This means that the data structure used for the representation of surface segments must allow for three or more surface segments to share a common contour. This is not necessary for the results of a valid surfaces-from-contours type algorithm, which can be represented with a concatenation of two polyhedral surfaces joined at one common simple closed contour.

B. Previous Work

The problem of reconstructing a surface from a set of scattered sample points arises in a variety of applications in the fields of reverse engineering, computer graphics, computer vision, and medical image segmentation. Surface reconstruction from scattered samples is an ill-posed problem in that there is no unique solution [91]. Although a number of surface reconstruction techniques have been proposed, most existing algorithms make certain strong assumptions on the original surface and its sample points. For example, many algorithms necessitate a dense sampling to be able to capture drastic topological or geometric change in a small region; some approaches need additional knowledge such as surface normal or interior/exterior information; some algorithms are not tolerant of noise and corrupted data.

The problem of reconstructing separating surfaces arises in a variety of biomedical applications and in multi-fluid dynamics calculations. This problem, an extension of the problem of reconstructing a manifold surface from a set of scattered points, is also an ill-posed problem. Moreover, the multiple regionalization [32] posed in this problem leads to reconstruction of non-manifold surfaces that are difficult to model in conventional solid modeling [90]. Although a small number of separating surfaces reconstruction techniques have been proposed, all algorithms necessitate a conversion of input data into a regular volumetric grid. These algorithms require that intra-planar and inter-planar resolutions be sufficiently similar in their input.

1. Surface reconstruction from unorganized points

Boissonnat [92] describes a method for Delaunay triangulation of a set of points in 3-space, that progressively eliminates tetrahedra from the Delaunay triangulation based on their circumspheres. Edelsbrunner and Mucke [93] use α-shapes to build a polyhedral shape with an unorganized set of points. Hoppe et al [94] generate a signed distance

function from the input points, and then polygonalize its zero-set by the marching cubes algorithm [95]. Curless and Levoy [96] derive a continuous volumetric function from the sample points and store it on a voxel grid. Amenta et al [97] produce a topologically correct output mesh that interpolates the input points based on the three-dimensional Voronoi diagram and Delaunay triangulation. Adamson and Alexa [98], [99] Aelxa et al [100], Amenta and Kil [101], Fleishman et al [102] , Levin [103], Xie et al [91], and Zwicker et al [104] use the MLS surface or its variants for point-set modeling and rendering. Edelsbrunner and Harer [105] describe how to extract Jacobi surfaces from Morse functions.

A major advantage of the unorganized points algorithms is that they do not make any prior assumptions about connectivity of points. In the absence of range images or contours to provide connectivity cues, these algorithms are the only recourse [96]. However, although these algorithms behave well in smooth regions of surfaces, they are not robust in regions of high curvature unless some sampling density criteria are met [96], [97]. They often require dense sampling of points, or additional data such as surface normals or interior/exterior information of the sample points [91].

2. Surface reconstruction from contours

The problem of reconstructing surfaces from contours is generally broken into three subproblems: correspondence, tiling, and branching [20], [29]. The correspondence problem deals with determining the topological adjacency relationships between the contours. To determine the correspondence, Meyers et al [29] first approximate the contours by ellipses and assemble them into cylinders. Soroka [106] uses the concept of generalized cylinders. Jones and Chen [27] use contour overlap. Bajaj et al [20] require that, given a fine enough inner-slice spacing, the reconstructed surface between adjacent slices can have at most one intersection with any line orthogonal to the slices. The tiling problem is concerned with generating the “best” topological adjacency

relationships between the points on pairs of contours by constructing a triangular mesh from their points. Keppel [28] first reduces this problem to a graph search problem, and tries to maximize the volume enclosed by convex polygons to find an optimal triangulation. Fuchs et al [25] find an optimal triangulation by minimizing the surface area using a divide-and-conquer approach in an Euler tour of a toroidal graph in $O(n^2)$ time where n is the number of data points in each contour. Christiansen and Sederberg [24] use the selection of shortest slice chords as their optimization criterion. The branching problem arises when an object is represented by a different number of contours in adjacent sections. Christiansen and Sederberg [24], and Jones and Chen [27] rely on user interaction to guide the solution. Boissonnat [92] adds extra vertices before Delaunay triangulation and builds a branched structure from the two Delaunay triangulations. Geiger [26] combines the external Voronoi skeleton and the Delaunay triangulation, splits the merging contour into several regions corresponding to each of the branching contours, and constructs a tetrahedron between these corresponding regions.

3. Separating surfaces

The first algorithm specifically designed for separating surface extraction was developed by Nielson and Franke [107], and also independently by Muller [108]. Their algorithm assumes that it is given a collection of three-dimensional rectilinear grid points each of which has been classified into one of several possible classes. Their marching cubes type [95] algorithm first transforms voxels into tetrahedra, and then marches through each tetrahedron, building triangles for the separating surfaces by evaluating the classifications of the tetrahedron's nodes against a mask and a case table. Using this method, Weinstein [109] generates separating surfaces from cross-sectional contours that partition each cross-section into multiple materials by first voxelizing each cross-section onto a uniform grid and then contouring the regular voxel grid. Similarly, Ju et al [110] first projects the contours from two planes orthogonally onto a common

plane, creates a volume graph based on the intersection points formed on the common plane, transforms the volume graph into a volumetric grid, and finally generates separating surfaces using a multimaterial contouring method. Bonnell et al [111], [112] generate separating surfaces by constructing material interfaces for a grid where each grid vertex has an associated barycentric coordinate representing the fractional parts of each material at the vertex.

C. Data Structure: Topology Graph

Traditionally, solid modeling systems based on boundary representations have employed the *winged edge* data structure or its variants for storing topology. The disadvantage of the winged-edge-based data structure is that it is simply unable to represent many of the non-manifold conditions that arise in the construction of complex shapes [113].

We introduce a data structure to represent the topological skeleton of an object surface and to direct separating surface generation between contours. The idea of using a topological object description to direct the process of reconstructing 3D surfaces from a stack of sections is not new. Kaneda et al [114], Giersten et al [89], Shinagawa and Kunii [115], and Biasotti [116] proposed a topological graph structure based on the Reeb graph for this purpose. However, the Reeb graph is also defined on a manifold. Although our data structure can be considered as an extension of the Reeb graph and its variants, it is specifically designed to represent the topology and geometry of reconstructed separating surfaces that are non-manifold. Furthermore, what is novel in our approach is that we use a topological skeleton to *build* an unknown shape from insufficient data. Previous approaches have used a topological skeleton (i.e., the Reeb graph) to *extract* a simplified shape description from huge data sets representing a digital model of three-dimensional object [116].

1. Reeb graph and Morse theory

Morse theory can be thought of as a generalization of the classical theory of critical points (maxima, minima and saddle points) of smooth functions on Euclidean spaces. Morse theory states that for a generic function defined on a closed compact manifold the nature of its critical points determines the topology of the manifold. Morse functions are generic functions for which all the critical points are non-degenerate, that is, the Hessian matrix of the function at a critical point is non-singular [117]. For a Morse function, the critical points determine the homology groups—a set of points for which the function is less than a given value—of the manifold, and these sets fully describe the topology of the manifold. The way the manifold is embedded in the three-dimensional space can be coded using the Reeb graph, which is a skeleton graph that encodes the evolution and the arrangement of the homology groups. A Reeb graph represents the configuration of critical points and their relationship, and provides a way to understand the intrinsic topological structure of a shape [117].

Formally, the Reeb graph is defined as follows [115]:
Let $f : M \to R$ be a real valued function on a compact manifold M. The Reeb graph of f is the quotient space of the graph of f in $M \times R$ by the equivalence relation ~ given below:

$(X_1, f(X_1)) \sim (X_2, f(X_2))$

if and only if

$f(X_1) = f(X_2)$

and

X_1, X_2 are in the same connected component of $f^{-1}(f(X_1))$.

The Reeb graph of the height function on an object surface is the quotient space of the graph in R^3. This quotient space identifies (x_1, y_1, z_1) and (x_2, y_2, z_2) if $z_1=z_2=z$, and if these two points are in the same connected component on the cross section of the

surface at the height z. The Reeb graph represents the contours on each plane as nodes, and shows the topological relations between contours in two successive cross sections (Figure 12).

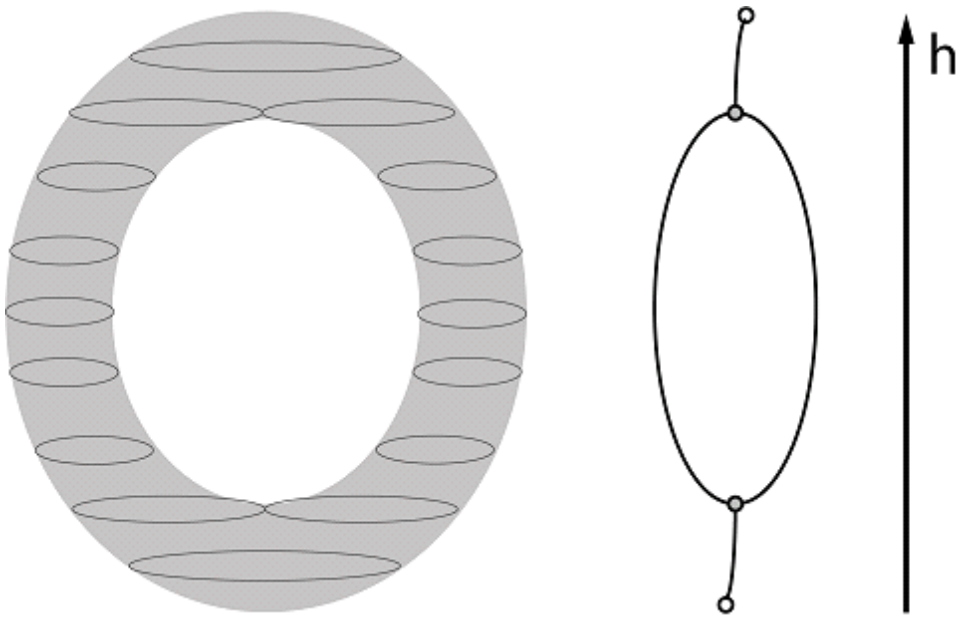


Fig. 12. Reeb graph of a torus.
(a) shows a torus with its cross sections. (b) shows its Reeb graph. (c) shows the direction of the height function.

Given a shape represented as a triangle mesh, its Reeb graph under the height map can be efficiently extracted by cutting the mesh into parallel slices, orthogonal to the considered height direction, and by studying the properties of the resulting mesh strips

delimited by two adjacent levels [118], [119]. The topological connectivity of critical points can be reconstructed using edge-based adjacency among contours. The Reeb graph structure has been applied to model representation [120], shape understanding [118], [121], similarity estimation [122], and database retrieval [123].

2. Topology graph

Recall that our goal is to reconstruct separating surfaces from sparse cross-sectional line drawings or diagrams depicting multiple regions, and that the input curves that make up each cross-sectional diagram can be open and not-simple. Our input, a series of cross-sectional line-drawings gives us visual information, but not topological information that we need to recover surfaces between them. Our solution consists of four steps (see Section V.A). In this section, we describe steps (1) and (2) to convert our input curves to a topology graph that represents the topological skeleton of each region's boundary surface and that shows between which contours the surface patches should be generated.

Step (1) Break the input curves into simple contours such that each simple contour lies on a boundary between exactly two regions, and merge any connected contours that have identical class sets

A cross-sectional diagram gives visual information on the boundaries of multiple regions present in the cross-section. For example, Figure 13(a) shows a cross-sectional diagram with six regions which are labeled in Figure 13(b). Our goal in this step is, given such a diagram, to break the curves into simple contours as shown in Figure 13(c) so that we can describe the boundary of a region using these contours. The curves are manually broken into simple contours to satisfy the condition that a simple contour lies between (i) exactly two regions, or (ii) one region and the background. After all the simple contours are determined, each contour is manually assigned (i) two classes denoting the boundaries of two regions or (ii) one class denoting the boundary of one region. If two contours are connected, and have identical class sets, they are merged.

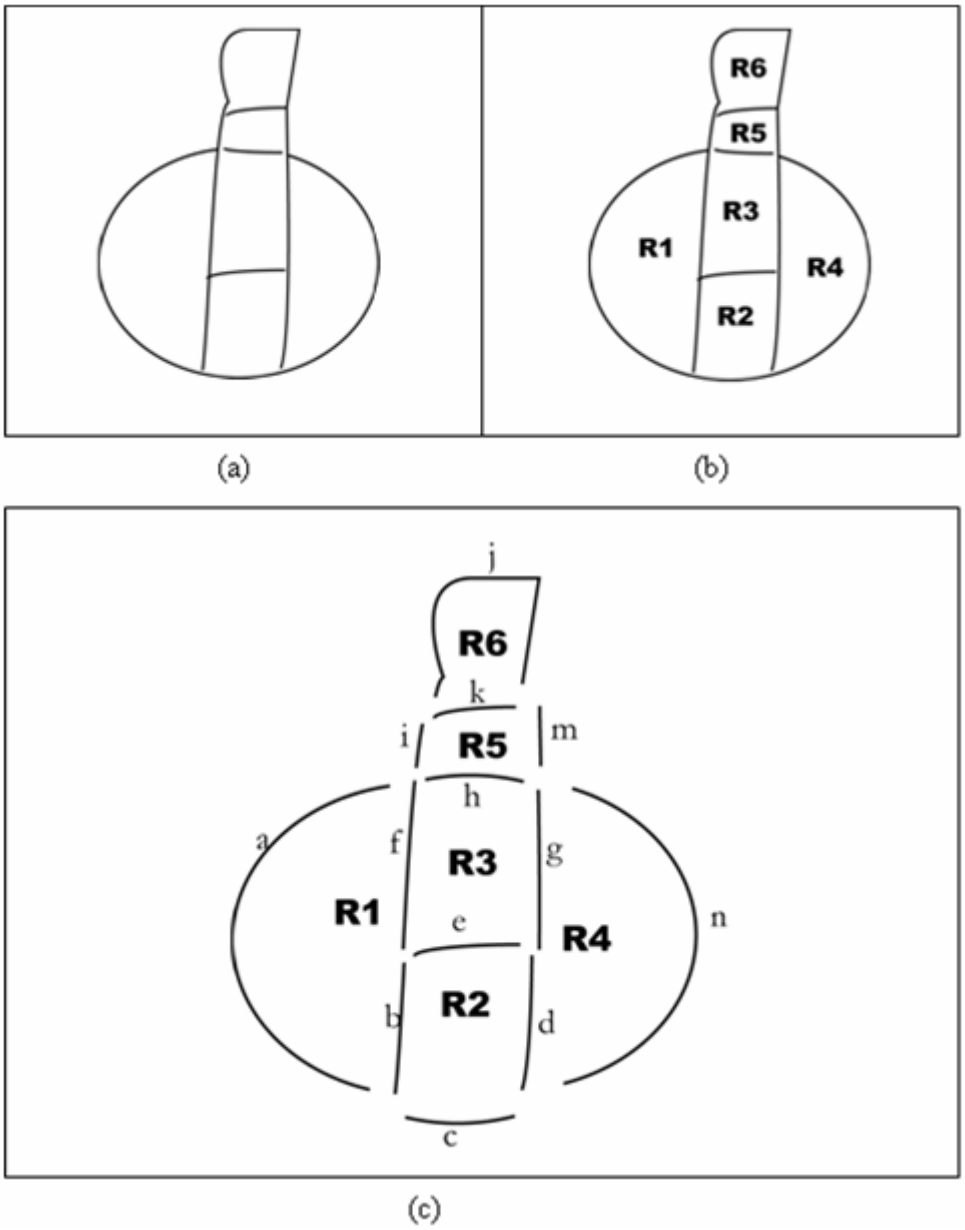


Fig. 13. From a diagram to a collection of simple contours.

(a) shows a cross-sectional diagram. (b) shows six regions; each region is labeled as "R#". (c) shows simple contours, labeled with lower case characters, broken up from the curves in the diagram.

For example in Figure 13(c), the contours, *b, d, e, f, g, h,* and *k*, lie between exactly two regions, whereas the contours, *a, c, i, j, m*, and *n*, lie between one region and the background. The contour, *a,* is assigned one class, $\{B_1\}$, denoting that *a* lies between region1 (R1) and the background, the contour, *b,* is assigned two classes $\{B_1, B_2\}$, denoting that *b* lies between region1 (R1) and region2 (R2).

Step (2) Construct a topology graph from the contours obtained from step (1)

For this step, our goal is to construct a topology graph to produce triangulated surfaces in later steps (3) and (4), that separate the surface components of different regions, R_k, $k = 1,...M$. We are given, after step (1) described above, a collection of simple contours C_{ij}, $i = 1,...,n$, $j = 1,...,nc_i$ where each contour has been classified to one or more of several classes B_k, $k = 1,...M$, denoting the boundary of R_k. Similar to the Reeb graph, we define for each region R_k a topology graph that represents the topological skeleton of the region's boundary surface and that shows between which contours the surface patches should be generated. Each contour, C_{ij}, is a vertex in our topology graph. The connectivity relations between the contours are specified first within a cross-section and then between two successive sections (see Figure 14). The topology of the objects to be reconstructed is then defined as a set of such relations throughout the complete stack of sections.

We assume that $C_{ij} = \{p_1, p_2, ..., p_{n_{ij}}\}$, i.e., a simple contour C_{ij} is defined by an ordered sequence of n_{ij} three-dimensional points.

For each R_k, $k = 1,...M$, we manually construct its topology graph $G_k(V_k, E_k)$ by constructing two subgraphs, first $Gp_k(V_k, Ep_k)$ for the topology of contours within cross-sectional planes and second $Gs_k(V_k, Es_k)$ for the topology of contours between successive cross-sections as follows:

for i = 1…n (where n is the number of cross-sections)

1. insert C_{ij} to V_k if $B_k \in class(C_{ij})$ where $j = 1,...,nc_i$, and nc_i is the number of contours in the i-th cross-section.
2. insert (C_{ij}, C_{il}) to Ep_k if $C_{ij} \in V_k$, $C_{il} \in V_k$, and C_{ij} and C_{il} are connected.

for i = 1…n-1

1. insert $(C_{(i+1)j}, C_{il})$ to Es_k if $surface(C_{(i+1)j}, C_{il}) \in R_k$ where $1 \le j \le nc_{i+1}$, $1 \le l \le nc_i$, and $surface(C_{(i+1)j}, C_{il})$ is a user-determined surface between two contours $C_{(i+1)j}$ and C_{il} .

$G_k(V_k, E_k)$ is then union of its two subgraphs, $Gp_k(V_k, Ep_k) \cup Gs_k(V_k, Es_k)$.

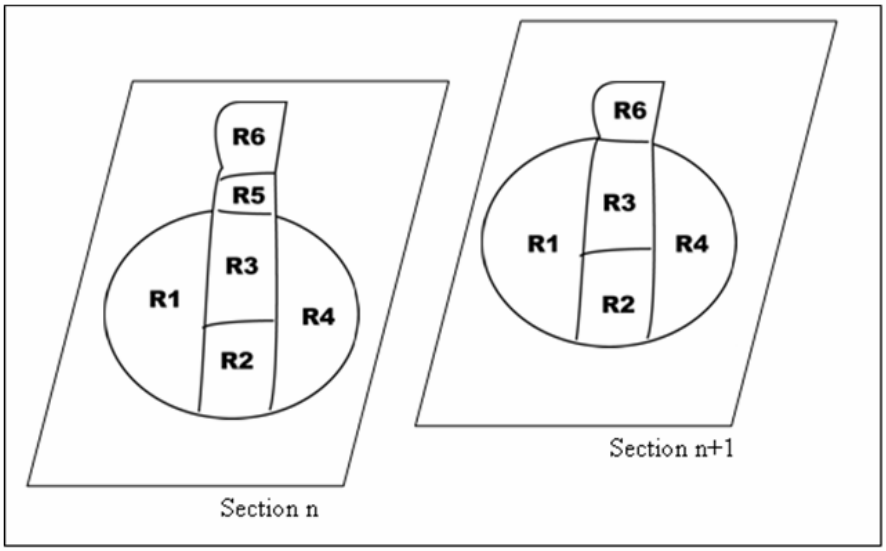


Fig. 14. Topology graph for a region within and between two cross-sections.
(a) shows two cross-sectional diagrams. (b) illustrates (shown on next page) a topology graph construction for region R1 within and between two sections.

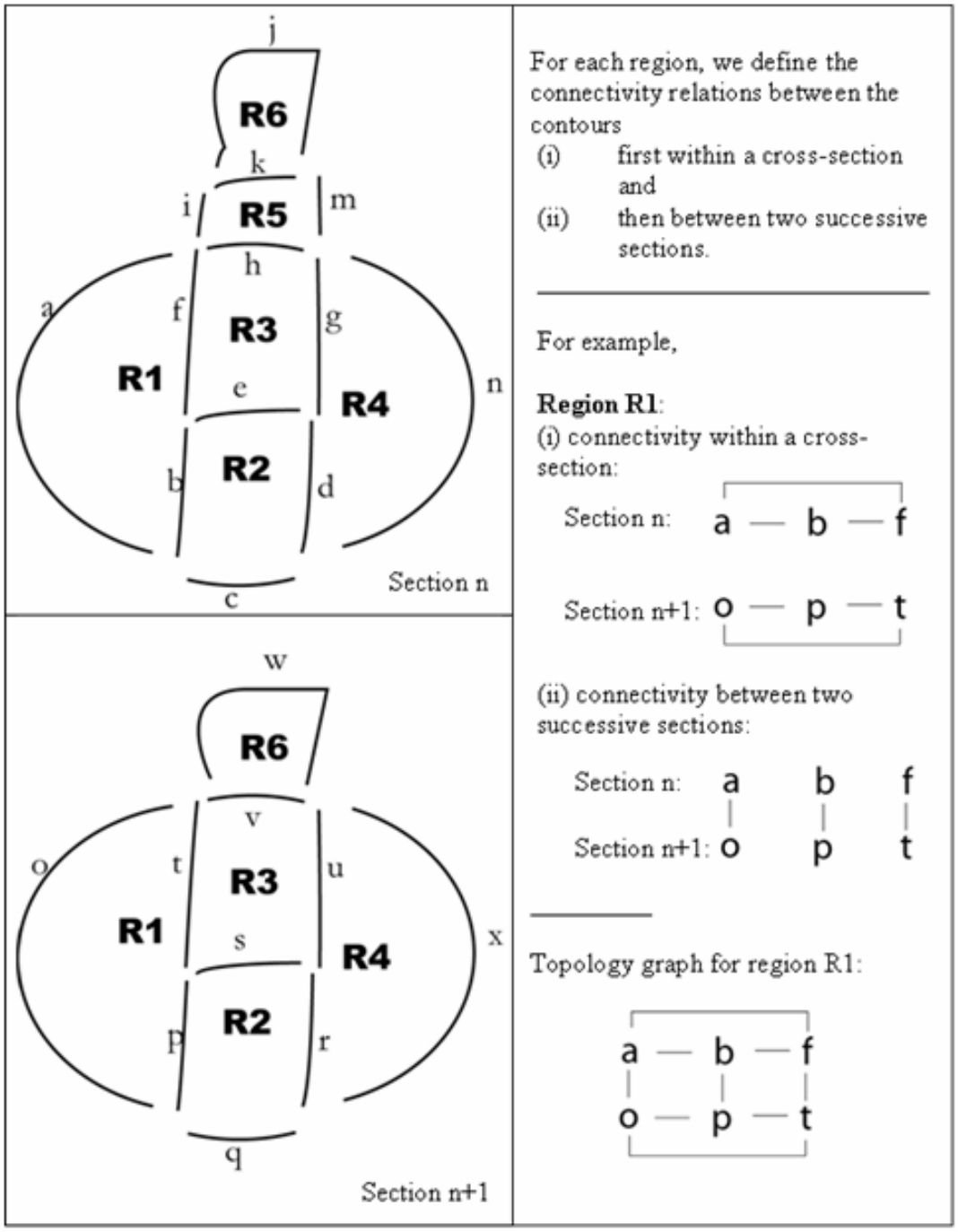


Fig. 14. Continued.

The edges Es_k of topology graph G_k describe the object surface of region R_k. $\bigcup_{k=1,...,M}(G_k(V_k, E_k))$ describes the boundary surface representation and cross-sectional boundaries for all regions R_k. An edge, $(C_{(i+1)j}, C_{il})$ where $1 \le i \le n-1$, $1 \le j \le nc_{i+1}$, and $1 \le l \le nc_i$, that is shared among two or more topology graphs describes a surface patch that separates two or more abutting regions.

Figure 15(a) and (b) illustrate $G_k(V_k, E_k)$, $k = 1,...,6$, and $\bigcup_{k=1,...,6}(G_k(V_k, E_k))$, respectively, corresponding to the six regions depicted in two diagrams shown in Figure 14(a). In Figure 15, the boundary classification information for each contour, as assigned in step (1) and shown in (a), is not shown in (b). Using the boundary class information, we can recover individual $G_k(V_k, E_k)$, $k = 1,...,6$ shown in Figure 15(a) from $\bigcup_{k=1,...,6}(G_k(V_k, E_k))$ shown in Figure 15(b).

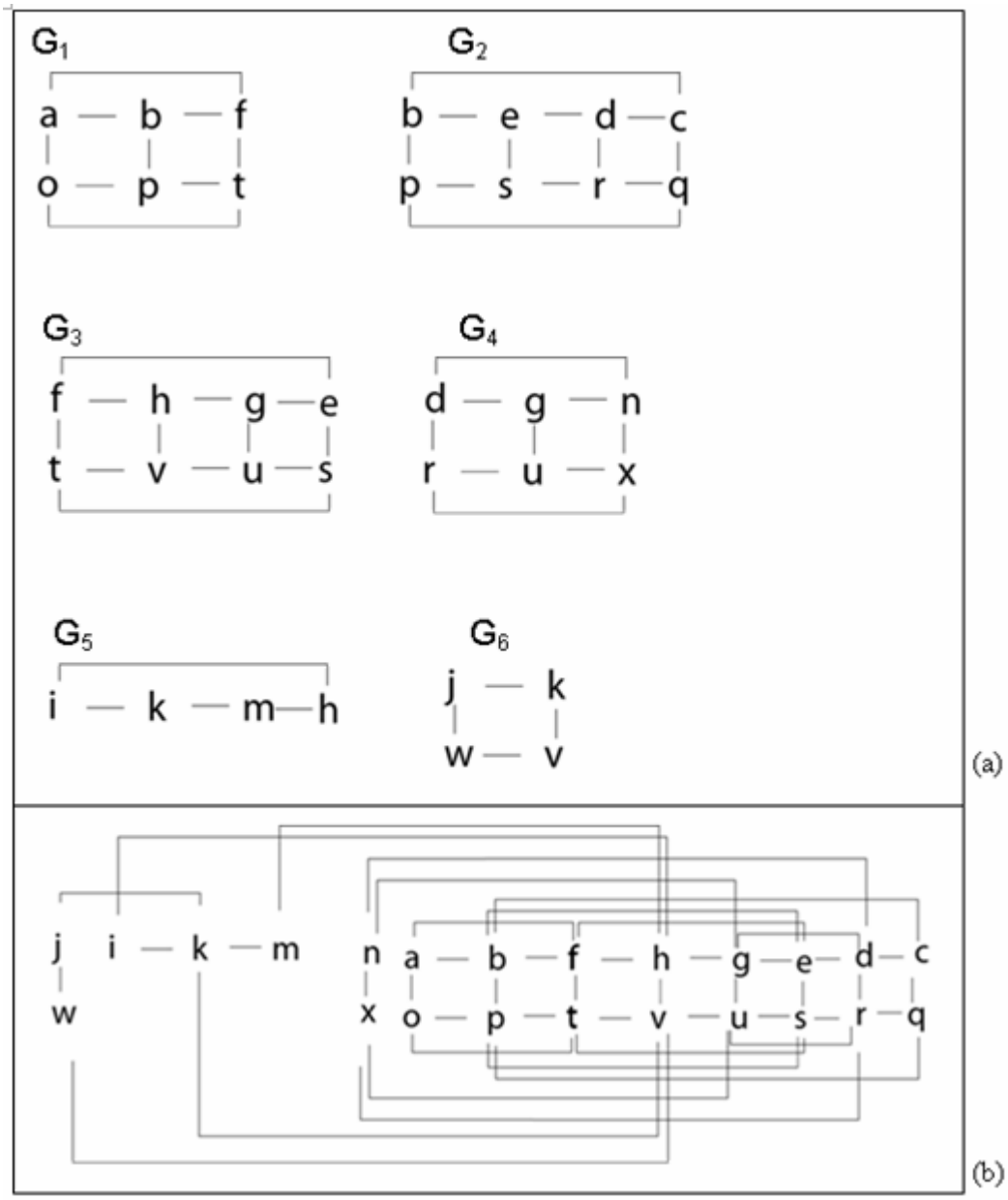


Fig. 15. Topology graphs.

(a) shows $G_k(V_k, E_k)$, $k = 1,...,6$, corresponding to regions 1-6 shown in Figure 14(a).

(b) shows $\bigcup_{k=1,...,6}(G_k(V_k, E_k))$.

D. Algorithm: Curve Fitting and Graph-Directed Triangulation

The contours C_{ij} define the intersections of the object surfaces at successive cross-sections. Our goal is to utilize these cross-sectional contours to recover the three-dimensional surfaces of the objects for visualization as well as geometric analysis. Shape from cross sections is an important problem in diverse fields of science, and it has been studied extensively [19], [20], [23], [25]-[31], [124], [125]. Most of these methods, however, suffer from correspondence, tiling, and branching problems.

The advantage of having a topology graph is that it allows reconstruction of three-dimensional surfaces automatically independent of shape complexity, i.e., branching, objects contained within objects, and objects abutting other objects. Another advantage is that it automatically solves the correspondence problem and the branching problem since topological descriptions are used to avoid ambiguities. We describe a new method to define and triangulate surfaces that are non-manifold or manifold with boundary. Our method, based on piecewise parametric curve-fitting technique and graph-directed triangulation, differs from conventional polygonalization methods in that it permits multiple, rather than binary, regions of space.

1. Correspondence, tiling, and branching problems

The correspondence problem (Figure 16) arises when there are multiple contours in a section because the contours must be organized into groups representing individual objects. Due to the underconstrained nature of the problem, automatic solution of the correspondence problem is difficult. Therefore, assumptions about the nature of the objects to be reconstructed are often used to help constrain the problem, allowing a *reasonable* solution [29]. To help reduce the exponential complexity of the problem, Bresler et al [126] advocated the use of domain knowledge to group contours into feasible objects.

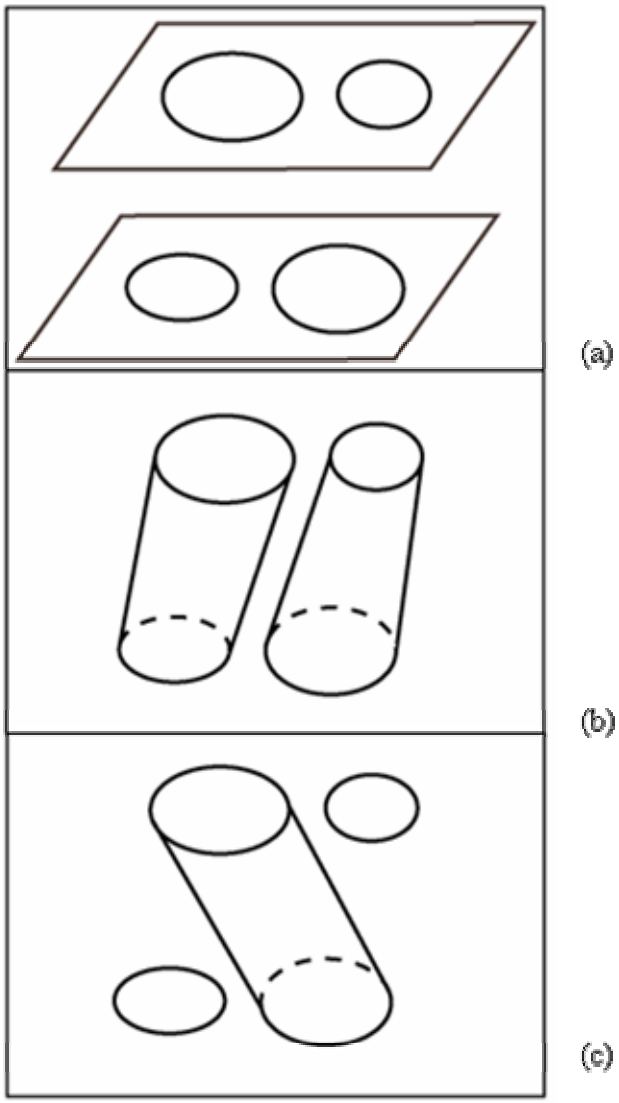


Fig. 16. Correspondence problem.

(a) shows two cross-sections each of which has two contours. (b) and (c) show two possible groupings between the same contours.

The tiling problem (Figure 17) is concerned with generating the "best" topological adjacency relationships between the points on pairs of contours by constructing a

triangular mesh from their points. The tiling problem has been the subject of most of the previous work on reconstructing surfaces from contours. Keppel [28] has shown that between two contour lines consisting of n and m points respectively, there are T(m,n) possible triangle arrangements where $T(m,n) = \frac{(m+n)!}{(m-1)!(n-1)!}$. For example, when n = m = 12, there are about 10^7 triangle combinations, and this combinatorial aspect precludes an exhaustive search for the optimal triangulation. Keppel [28] first reduced the problem of matching points in successive contours to a search problem on a toroidal graph. Fuchs et al [25] provided an extensive analysis of the search problem and developed an efficient search method. Sloan and Painter [30], [31] addressed the choice of metric for the graph cost function and described several improvements to the divide-and-conquer algorithm. The tiling problem is underconstrained, and many surfaces could give rise to the observed contours in cross-sections. In choosing one "correct" surface, the chosen surface should capture some notion of what a good surface is and should be easy to compute [29].

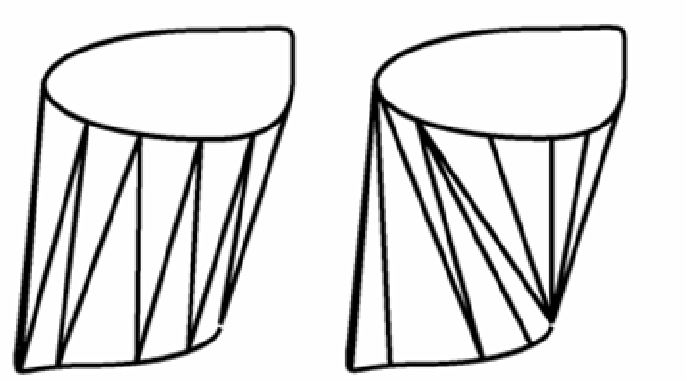


Fig. 17. Tiling problem.
Two possible acceptable tilings between two contours are shown.

A branching problem (Figure 18) exists when an object represented by m contours in one section is represented by n contours in an adjacent section, and m and n are distinct. Previous approaches to the branching problem have tried to form composite contours, adding fabricated vertices between the adjacent contours to model the saddle surface implied by the contours [24], [92]. In most cases however, these approaches resort to user interaction to guide a solution [20], [24].

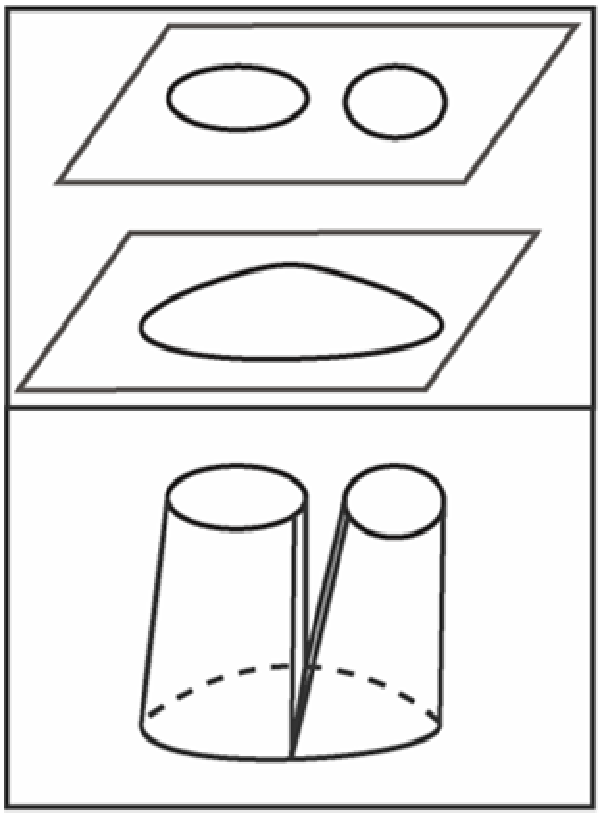


Fig. 18. Branching problem.

2. Curve fitting

We state our curve-fitting problem as follows: Given a set of data points (x_r, y_r), r = 1,… ,m, that describes an arbitrary curve in a cross-section, determine a spline $s(u)$ on [a, b] of degree k (k=3), with knots a= λ_0, λ_1, …, λ_g, λ_{g+1}=b, as the solution of a constrained optimization problem. Our goal is to find $s(u)$ with parametric representation

$$s(u) = \begin{cases} x = s_x(u), \\ y = s_y(u), \end{cases} \qquad a \le u \le b$$

that satisfies

$$(s_x(u_r), s_y(u_r)) \cong (x_r, y_r), \text{ r} = 1,\ldots,\text{m}.$$

Thus, our objective is to select the knots and to determine the B-spline coefficients to satisfy two criteria: the least squares criterion and the smoothing criterion. The least squares criterion determines the closeness of fit whereas the smoothing criterion determines the smoothness of fit (see Figure 19).

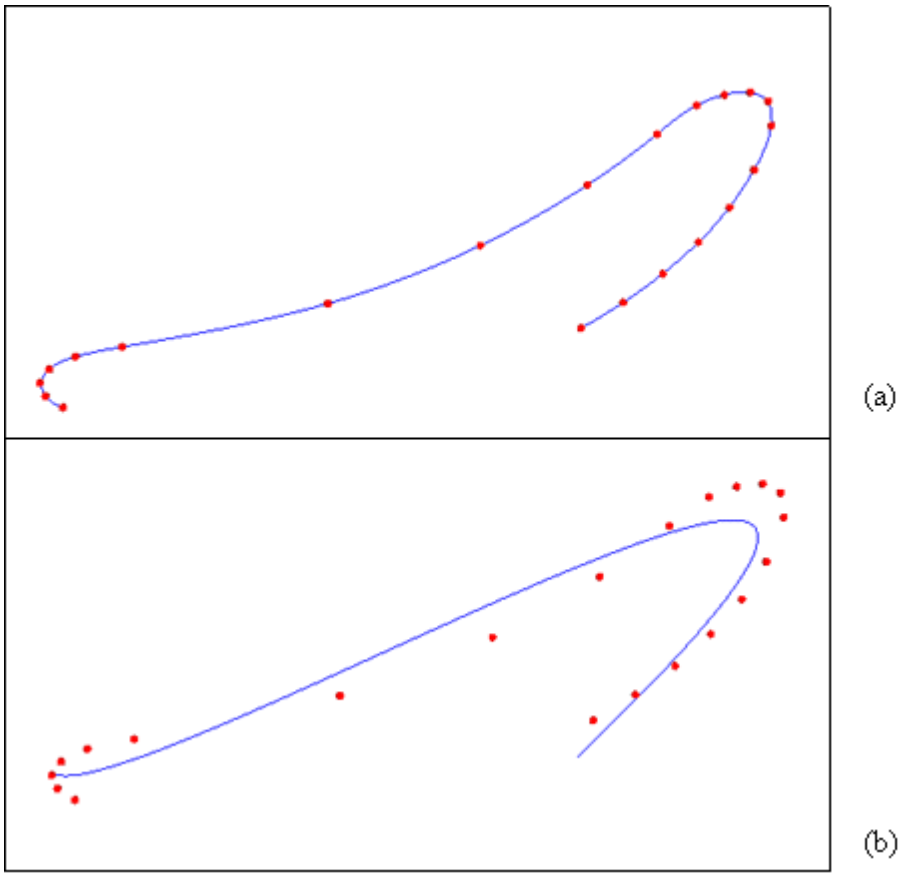


Fig. 19. Curve fitting.

A blue curve is fit to red points. (a) illustrates closeness of fit; (b) illustrates smoothness of fit.

B-spline representation

With every point (x_r, y_r), we associate a u_r-value such that $u_r \leq u_{r+1}$. We then determine for an interval [a, b] two cubic spline functions $s_x(u)$ and $s_y(u)$ with common knots, λ_i, i = 0, 1,…, g, g+1 (λ_0=a, λ_{g+1}=b), where

$$s_x(u) = \sum_{i=-k}^{g} c_{xi} N_{i,k+1}(u) \text{ and } s_y(u) = \sum_{i=-k}^{g} c_{yi} N_{i,k+1}(u).$$

The coefficients, c_{xi} and c_{yi}, are called *B-spline coefficients* of s_x(u) and s_y(u), respectively, and $N_{i,k+1}(u)$ are *B-spline basis functions* defined as

$$N_{i,k+1}(u) = (\lambda_{i+k+1} - \lambda_i)\Delta_t^{k+1}(\lambda_i,...,\lambda_{i+k+1})(t-u)_+^k$$

where $(t-u)_+^k = \begin{cases} (t-u)^k, & \text{if } t \geq x \\ 0, & \text{if } t < x \end{cases}$.

The least-squares criterion

The least-squares criterion determines the desired spline by minimizing

$$\delta = \sum_{r=1}^{m} w_r^2 \left((x_r - s_x(u_r))^2 + (y_r - s_y(u_r))^2 \right),$$

where w_r, which are called *weights*, allow account to be taken of differing accuracies of how closely s_x(u) and s_y(u) fit the data points.

The smoothing criterion

The smoothing criterion minimizes η defined as

$$\sum_{i=1}^{g} \left(s_x^{(k)}(\lambda_i+) - s_x^{(k)}(\lambda_i-) \right)^2 + \left(s_y^{(k)}(\lambda_i+) - s_y^{(k)}(\lambda_i-) \right)^2$$

subject to the constraint such that

$$\delta = \sum_{r=1}^{m} w_r^2 \left((x_r - s_x(u_r))^2 + (y_r - s_y(u_r))^2 \right) < \text{S}.$$

The smoothing factor S is a user-specified, positive number that controls the extent of smoothing.

Solution to curve fitting

Following the approach taken by Dierckx [127] and de Boor [128], we treat our curve fitting problem as a minimization problem. Given a set of contour points, $\{cp_1, cp_2, ...cp_m\}$, we obtain the B-spline coefficients and the knots as the solution of the following problem:

Minimize

$$\eta = \sum_{q=1}^{g} \left\| s^{(k)}(\lambda_q +) - s^{(k)}(\lambda_q -) \right\|^2$$

subject to constraint

$$\delta = \sum_{r=1}^{m} \left\| w_r^2 (cp_r - s\ (u_r)) \right\|^2 \le S\,.$$

We compute the B-spline coefficients $c_{i,l}$ of n splines $s_{l,p}(u)$, defined for positive values of p and $l = 1,...,n$ as the least-squares solution of the n systems of equations ,

$$w_r \sum_{i=-k}^{g} c_{i,l} N_{i,k+1}(u_r) = w_r cp_{r,l}, \quad \text{r = 1, …, m,}$$

$$\frac{1}{\sqrt{p}} \sum_{i=-k}^{g} c_{i,l} a_{i,q} = 0\,, \quad \text{q = 1, …, g,}$$

where p is given the value of the positive root of F(p) = S,

$$F(p) = \sum_{r=1}^{m} \left\| w_r (cp_r - s_p(u_r)) \right\|^2,$$

and

$$a_{i,q} = \begin{cases} 0, & \text{if } i < q-k-1 \text{ or } i > q \\ \dfrac{(-1)^{k+1} k! (\lambda_{i+k+1} - \lambda_i)}{\prod_{j=i, j\neq q}^{i+k+1} (\lambda_q - \lambda_j)} & \text{if } q-k-1 \le i \le q \end{cases}.$$

We select the knots

a= λ_0, λ_1, …, λ_g, λ_{g+1}=b

first by determining the weighted least-squares polynomial which is simply the weighted least-squares spline $S_0(u)$. If $F(0) \le S$, this is a solution to our problem. If $F(0) > S$, then we determine successive least-squares splines $S_{g_j}(u)$ with an increasing number of knots g_j, until the condition

$$F_g(\infty) \le S < F(0)$$

is satisfied. At each iteration we increase the number of knots g_j by Δg_j where

$$\Delta g_j = \begin{cases} 1, & j = 0 \\ \min\{\Delta_4, \Delta_3, \max\{1, \Delta_1, \Delta_2\}\} & j = 1,2,... \end{cases},$$

$$\Delta_1 = \left\lceil \frac{F_{g_j}(\infty) - S}{F_{g_{j-1}}(\infty) - F_{g_j}(\infty)} \Delta g_{j-1} \right\rceil,$$

$$\Delta_2 = \left\lceil \frac{\Delta g_{j-1}}{2} \right\rceil,$$

$\Delta_3 = 2\Delta g_{j-1}$, and

$\Delta_4 = m - k - 1 - g_j$.

The additional knots are then located inside the intervals $[\lambda_i, \lambda_{i+1}]$ with the largest δ_i. As previously mentioned, our solution to the curve fitting problem is based on two books, by Dierckx [127] and de Boor [128], and they provide detailed explanations and proofs which are not repeated here.

3. Graph-directed triangulation

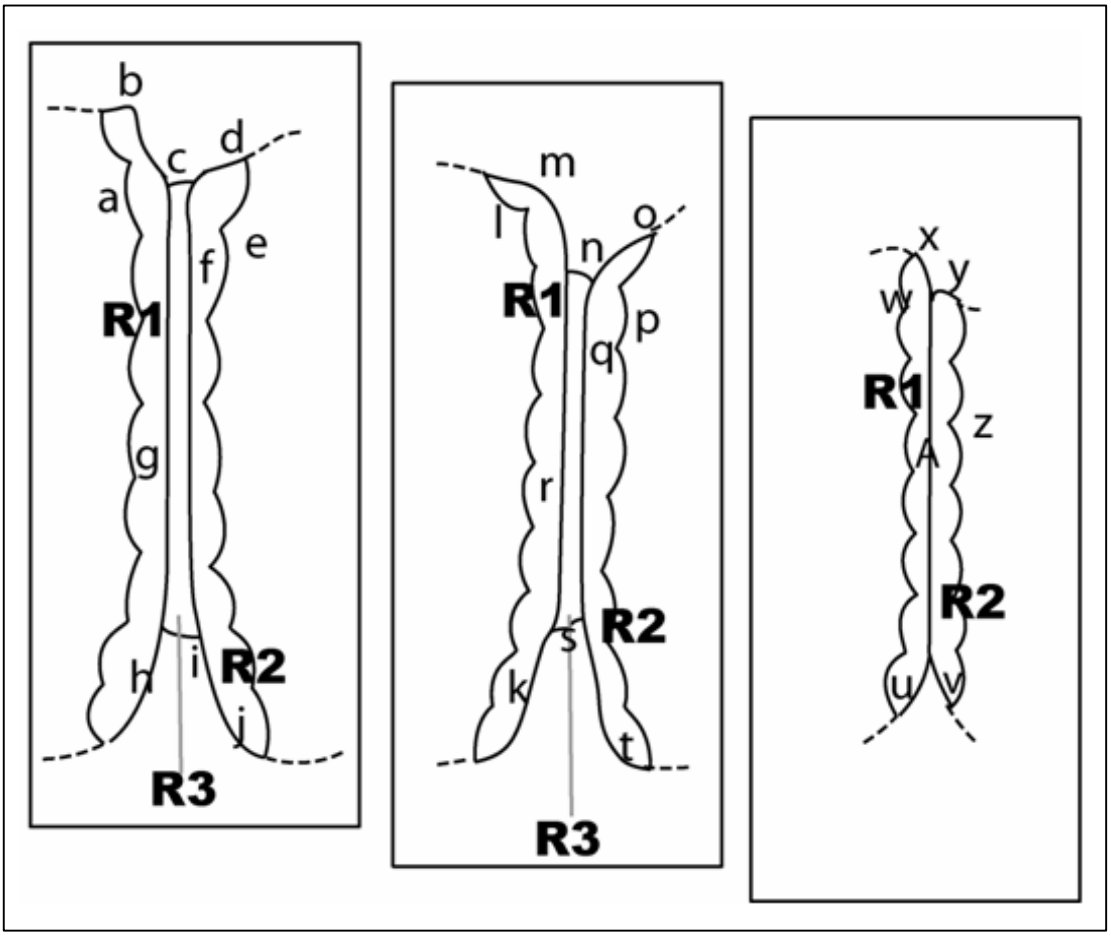
Recall that our goal is to produce triangulated surfaces that separate the components of different regions R_i, $i = 1,...,M$, given a set of topology graphs $\bigcup_{i=1,...,M}(G_i(V_i, E_i))$ as defined in Section V.C.2. Our manual construction of topology graphs solves the correspondence problem by prescribing which contours are to be grouped together

between two sections; it also solves the branching problem because each grouping of two contours from adjacent sections is one-to-one, not m-to-n where m and n are distinct (see Section V.D.1). Similar to the traditional methods, we generate surfaces by tiling two contours from two adjacent cross-sections, given by edges $(C_{(k+1)j}, C_{kl})$ in $\bigcup_{i=1,...,M}(Gs_i(V_i, Es_i))$ where $1 \le k \le n-1$, $1 \le j \le nc_{k+1}$, and $1 \le l \le nc_k$. The novelty in our method is that our method generates consistent triangulations across boundaries of multiple objects. This consistency is not possible with traditional methods since they reconstruct each object surface independently and thus result in different tilings for different objects, leading to interpenetrations or gaps between the surfaces shared by abutting objects.

To generate a tiled surface corresponding to an edge $(C_{(k+1)j}, C_{kl})$ in $\bigcup_{i=1,...,M}(Gs_i(V_i, Es_i))$, we first fit each contour with a piecewise parametic B-spline function as described in the previous section. We then evaluate each piecewise B-spline function, that satisfies both the least-squares and the smoothing criteria, at a fixed number P of intervals, and use these evaluated points to produce a triangulation. We assign P such that the spline functions corresponding to contours that are topologically connected in $\bigcup_{i=1,...,M}(Gs_i(V_i, Es_i))$ are evaluated at the same number of points, and our method is independent of similarities in shapes or parameterizations between two contours.

Given $Gs(V, Es) = \bigcup_{i=1,...,M}(Gs_i(V_i, Es_i))$ and an arbitrary vertex $C_{ij} \in V$, where $i = 1,...,n$; $j = 1,...,nc_i$, we build a breadth-first-tree from source C_{ij}, $BFT(C_{ij}) = Gs(V_\pi(C_{ij}), Es_\pi(C_{ij}))$. $BFT(C_{ij})$ is a subgraph of $Gs(V, Es)$ consisting of vertices and edges reachable from C_{ij} during a breadth-first-search on $Gs(V, Es)$ from C_{ij} (see

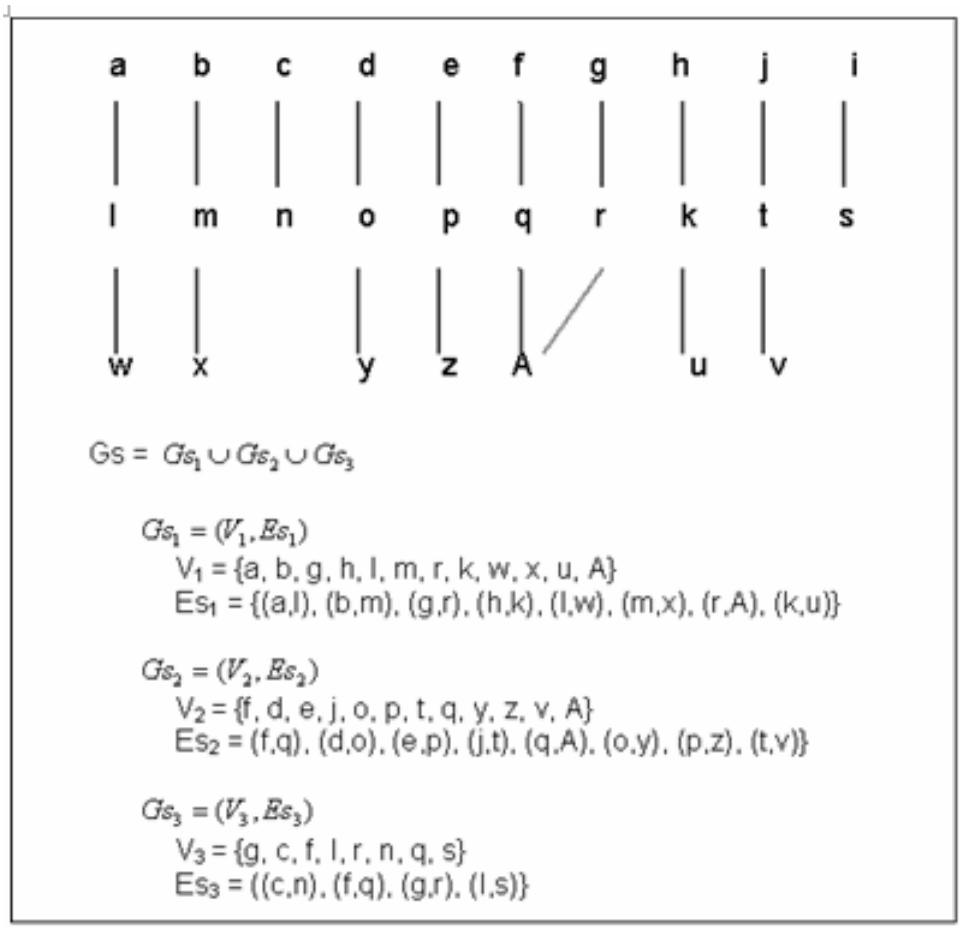
Figure 20). Therefore, $BFT(C_{ij})$ describes a complete set of contours (vertices) and surface patches (edges) that are topologically connected to C_{ij} in $Gs(V, Es)$; we evaluate spline functions for contours in $V_{\pi}(C_{ij})$ at the same number of points before triangulation. We then repeat the process from source $C_{st} \in V - V_{\pi}(C_{ij})$, and so on until the union of all breadth-first trees constructed is equivalent to $Gs(V, Es)$.



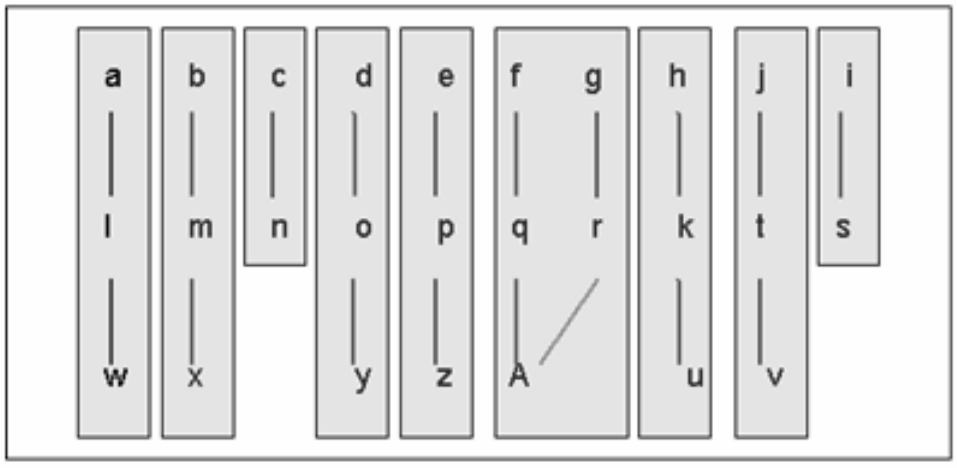
(a)

Fig. 20. Breadth-first trees of a topology graph.

(a) shows input contours (named a, b, …z, and A) and regions (R1, R2, and R3) in three consecutive cross-sections. (b) shows Gs. (c) shows all breadth-first trees (shaded in gray) found in Gs. (b) and (c) are shown on next page.



(b)



(c)

Fig. 20. Continued.

In addition to directing which contours are to be tiled together, our topology graph also maintains the number of geometric points used during triangulation to be consistent. This, in turn, ensures that our abutting or shared surface patches are precisely coincident. Thus, our topology graph directed triangulation results in separating surfaces that are geometrically consistent.

CHAPTER VI

BOUNDARY SURFACE APPROXIMATION OF ANATOMICAL STRUCTURES IN MOUSE BRAIN OLFACTORY BULB

A. Introduction

Reconstruction of three-dimensional objects from serial two-dimensional images can contribute to the understanding of many biological structures. In this chapter, we apply our method for the reconstruction of separating surfaces, as described in the previous chapter, to approximate the boundary surfaces of the anatomical structures in mouse brain olfactory bulb. We then apply our surface approximations to direct image alignment.

Given a stack of sparse contours denoting abutting and shared boundaries of multiple anatomical regions in mouse brain olfactory bulb, our problem is to approximate the topology and geometry of the reconstructed surfaces of these regions. This problem arises from an unknown three-dimensional object that has to be reconstructed from a sequence of two-dimensional images. The only information about the object consists of the intersections of its surface with a finite number of specified parallel planes. These intersections are assumed to be curved contours; each contour is defined by a finite number of points. Thus, the input to the problem is a set of finite sequences of points encountered while traversing each of the contours. The desired output is a piecewise planar approximation to the original object surface constructed in such a way that its intersections with the parallel planes are assured to be nearly identical to the original curves lying on them. A negative result by Gitlin et al [19] has shown that, in general, two polygonal curves cannot be joined by a non-self-intersecting surface with only those vertices, and even deciding its possibility is NP-hard [129]. Our solution to the

problem tries to find a "good" separating surface reconstruction based on a topology graph, piecewise parametric curve-fitting, and piecewise triangulation.

Our solution is fully applicable to reconstruction of separating surfaces of other anatomical regions in mouse brain or of multiple regions in other entities. Our current bottleneck is the manual construction of topology graphs. As mentioned in Chapter V, after the topology graphs are constructed, our method solves the curve-fitting and graph-directed triangulation automatically.

B. Boundary Surface Approximation of Anatomical Structures in Mouse Brain Olfactory Bulb

1. Previous work on three-dimensional reconstruction of anatomical structures in mouse brain

Traditional atlases for adult mouse brain, consisting of a series of two-dimensional images of stained sections together with nomenclature and graphical outlines of standard anatomical structures, are provided in book form or as printed manuscripts [16]-[18]. Three-dimensional models try to overcome the limitations of traditional atlases by being able to navigate and computationally section the brain structures at arbitrary angles, or to view a structure independently or in conjunction with other structures to understand better their relationships with one another [130]. Although in high demand, digital three-dimensional mouse brain models are still sparsely available [76]. Sidman et al [73], [74] are developing a voxel-based atlas where segmented anatomical structures are grouped by color-encoded voxels. Similarly, the Allen Institute for Brain Science developed a voxel-based atlas for their database of gene expression patterns [77]. Mackenzie-Graham et al [79] report their group's development of a multi-modal imaging brain atlas [80] that co-registers histologically processed and annotated sections with magnetic resonance microscopy (MRM) images

of the same mouse brain, both in vivo and post-mortem. Kovacevic et al [75]'s minimal deformation atlas is produced by averaging MRM images of nine mouse brains. Ju [131] developed a polygonal atlas based on 350 histological tissue sections, which serves as a spatial database of gene expressions over the mouse brain. Ma et al [76] constructed a database based on the MRM images of 10 mouse brains, that offer three types of digital atlases—individualized, minimal deformation, and probabilistic.

Current three-dimensional models can be grouped into two categories based on how they represent the segmented anatomical structures: volumetric and surface-modeling. In the volumetric atlases [73], [75], [79], [80], after segmentation, each voxel of the atlas volume is assigned a unique anatomical label, usually by color- or intensity-encoding. Each segmented anatomical structure is visualized and manipulated as a set of color-coded voxels. The surface-modeling [76], [131] atlases model the boundaries of anatomical structures as reconstructed polygonal surfaces. The reconstructed surfaces of these two atlases are extracted from a volumetric grid after in-house segmentation.

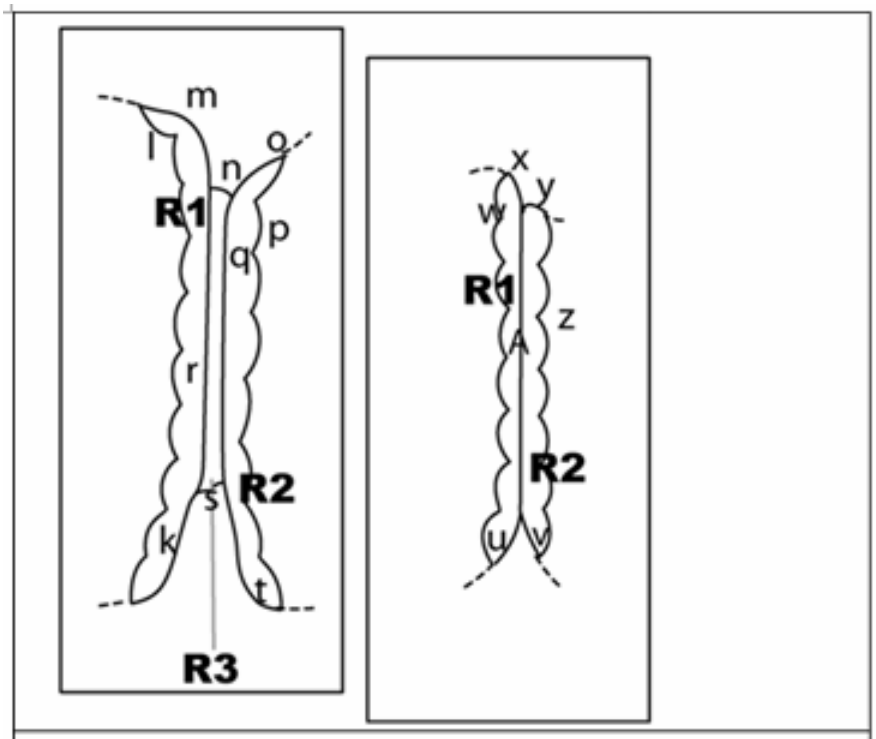
Our work presented in this chapter is distinct from previous and ongoing bodies of work on three-dimensional reconstruction of mouse brain anatomical structures in three ways. First, compared with the volumetric atlases, our reconstructions represent anatomical structures using boundary surfaces, not as a collection of voxels. Second, our work focuses on approximating the boundary surfaces of anatomical structures in the olfactory bulb. The surface-modeling atlases by Ju [131] and Ma et al [76] contain 17 and 20 segmented and reconstructed anatomical structures, respectively, of a whole mouse brain. Although they both include a reconstructed olfactory bulb, the anatomical structures inside the olfactory bulb are not reconstructed. Third, our work further differs from Ju [131] and Ma et al [76] in that we approximate the boundary surfaces of anatomical structures based on a surface-based, not a volume-based, approach to the problem of reconstructing a three-dimensional separating surface from a collection of

planar contours. Our boundary surfaces are reconstructed from a data set that defines the intersection of a surface and a plane of sectioning, whereas the volume-based approaches reconstruct surfaces from data that are available as a three-dimensional grid.

2. Topology graph construction

The ability to measure and understand the rich complexity and variability of brain structure and function often requires comparison against some index, standard or alternative representation. This anatomical framework may be in the form of a map to relate the name and location of structures within a coordinate system or a template with complete shape descriptions of structures [132]. Our reconstruction extends an existing two-dimensional anatomical framework, a stereotaxic mouse brain atlas by Paxinos and Franklin [18], whose anatomical delineation and nomenclature serve as a reference atlas in the neuroscience community, into three-dimensional surface approximations. Their anatomical delineations shown in curved line drawings are contours in our topology graph. The nomenclature attached to each delineated region is its class. Given this collection of contours and the set of classes, we construct a topology graph for each delineated region. An example of a topology graph constructed following the algorithm given in the previous chapter is shown in Figure 21.

Although our topology graph shows the topological skeleton of each structure's boundary surface, the task of visualizing and following the intrinsically three-dimensional anatomical structures is left to the user's imagination, and can be difficult. From the topology graph, we generate triangulated surfaces between contours from adjacent sections to demonstrate structural details and relationships that are otherwise difficult to discern or may be obscured in two-dimensional sections.



Topology graph G = $\bigcup_{i=1,2,3} (G_i(V_i, E_i))$.

where $G_i(V_i, E_i) = Gp_i(V_i, Ep_i) \cup Gs_i(V_i, Es_i)$

G_1:

V_1 = {l, m, r, k, w, x, u, A}
Ep_1 = {(l,m), (m,r), (r,k), (k,l), (w,x), (w,A), (A,u), (u,w)}
Es_1 = {(l,w), (m,x), (r,A), (k,u)}

G_2:

V_2 = {o, p, t, q, y, z, v, A}
Ep_2 = {(q,o), (o,p), (p,t), (t,q), (A,y), (y,z), (z,v), (v,A)}
Es_2 = ((q,A), (o,y), (p,z), (t,v)}

G_3:

V_3 = {n,s,r,q}
Ep_3 = {(r,s), (s,q), (q,n), (n,r)}
Es_3 = ϕ

Fig. 21. Topology graph example.

Mathematically, a contour is an intersection of an arbitrary surface and a plane. Contour lines of irregular and abutting structures do not easily lend themselves to exact polynomial representation, or other attempts at precise mathematical description [24]. In such cases, a convenient numerical description of a contour line can be approximated by a sequence of straight line segments. This description of a digitized contour line provides two pieces of information: point coordinates and connectivity of points implied by the sequence in which the points are listed. From this inherent connectivity in contour data, we can extract a surface definition between pairs of adjacent contour lines using triangulation. There are two obvious rules in triangulation for obtaining an acceptable surface [25] that reduce the number of triangle combinations: (1) If two points of the same contour are to be defined as points of the same triangle, they must neighbor each other on their contour line, and (2) no more than two vertices of any triangle may be recruited from the same contour [24]. Using these rules, we triangulate between the evaluated points of two curve-fit contours. Figure 22 shows the triangulated surfaces corresponding to the topology graph shown in Figure 21.

Figure 23 shows our reconstructed surfaces for anatomical structures in mouse brain olfactory bulb as delineated by Paxinos and Franklin [18]. The contours from sparse coronal sections are not equally spaced, and many contours are shared by multiple abutting structures.

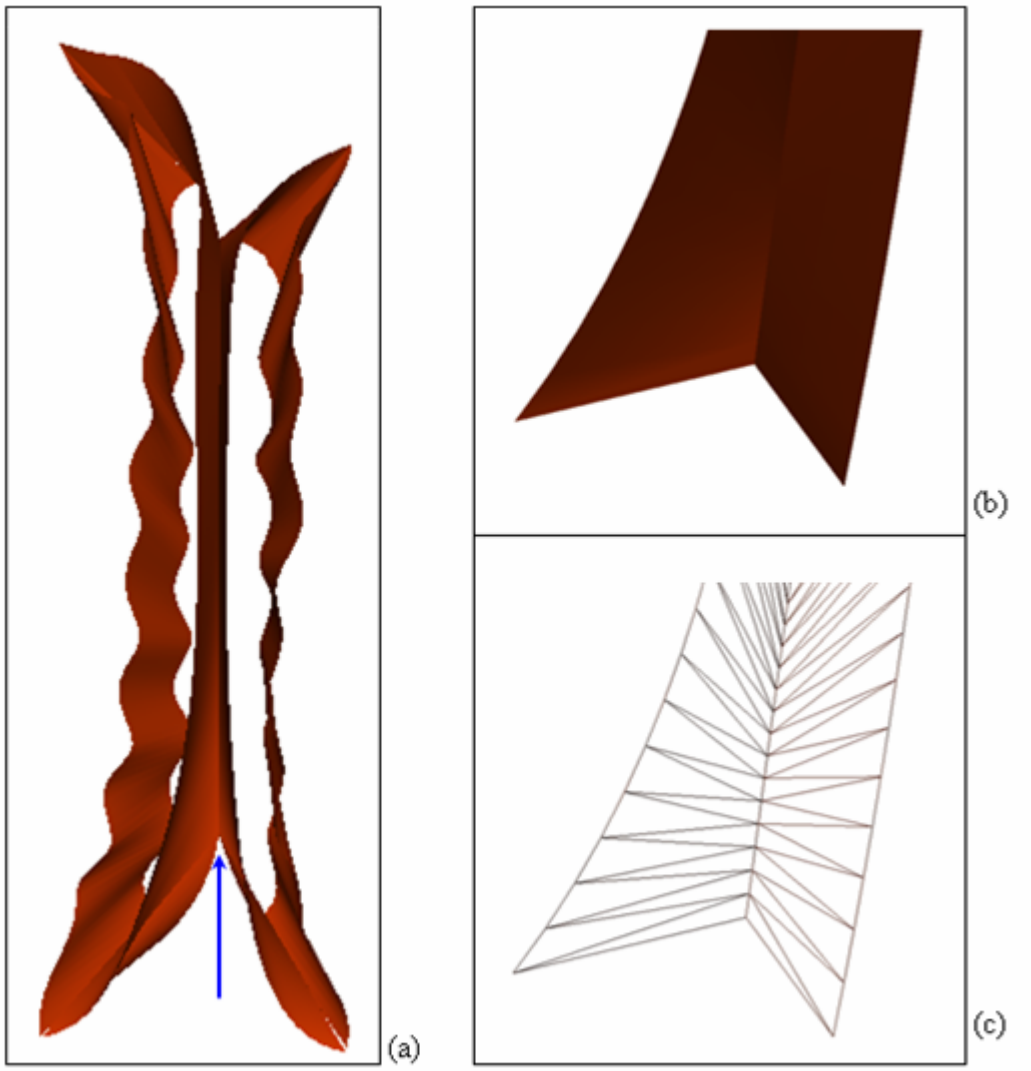
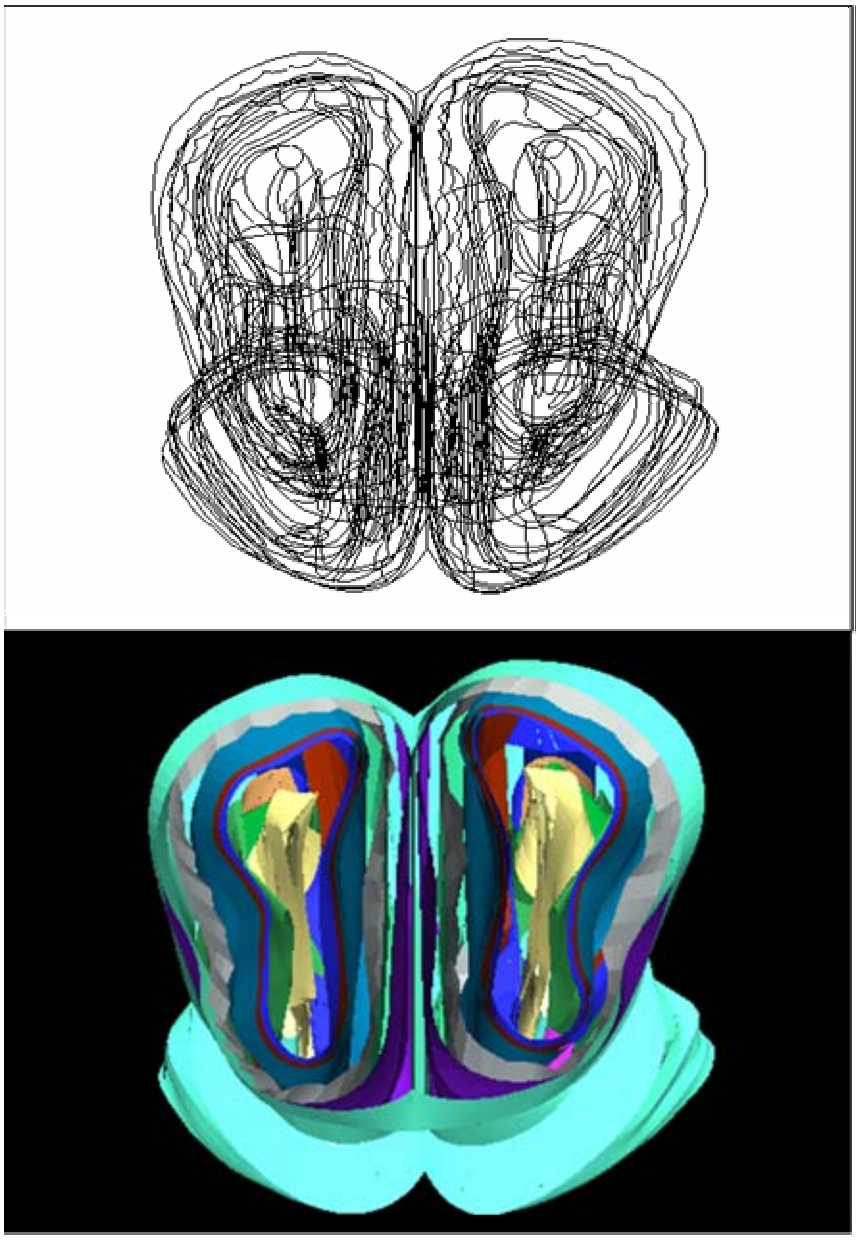


Fig. 22. Graph-directed triangulation.

(a) shows the triangulated surfaces corresponding to the topology graph from Figure 21. (b) shows a magnified view of two surfaces, (r,A) and (q,A) in Figure 21, around the blue arrow shown in (a). (c) illustrates the coincident triangulations around a shared contour between the two surfaces.



(a)

(b)

Fig. 23. Approximated surfaces of mouse brain olfactory bulb structures from coronal diagrams.

(a) and (b) show overlaid coronal diagrams and their approximated surfaces. (c) and (d) show overlaid contours and approximated surfaces of lateral olfactory tract. (d) The outer boundaries of the glomerular layer of the olfactory bulb are shown on left; contours for the olfactory nerve layer are shown on right. (e) is a ventral view of the same contours shown in (d). (f) shows surface approximations of contours shown in (e). (g) is an overlaid view of two structures shown in (f).

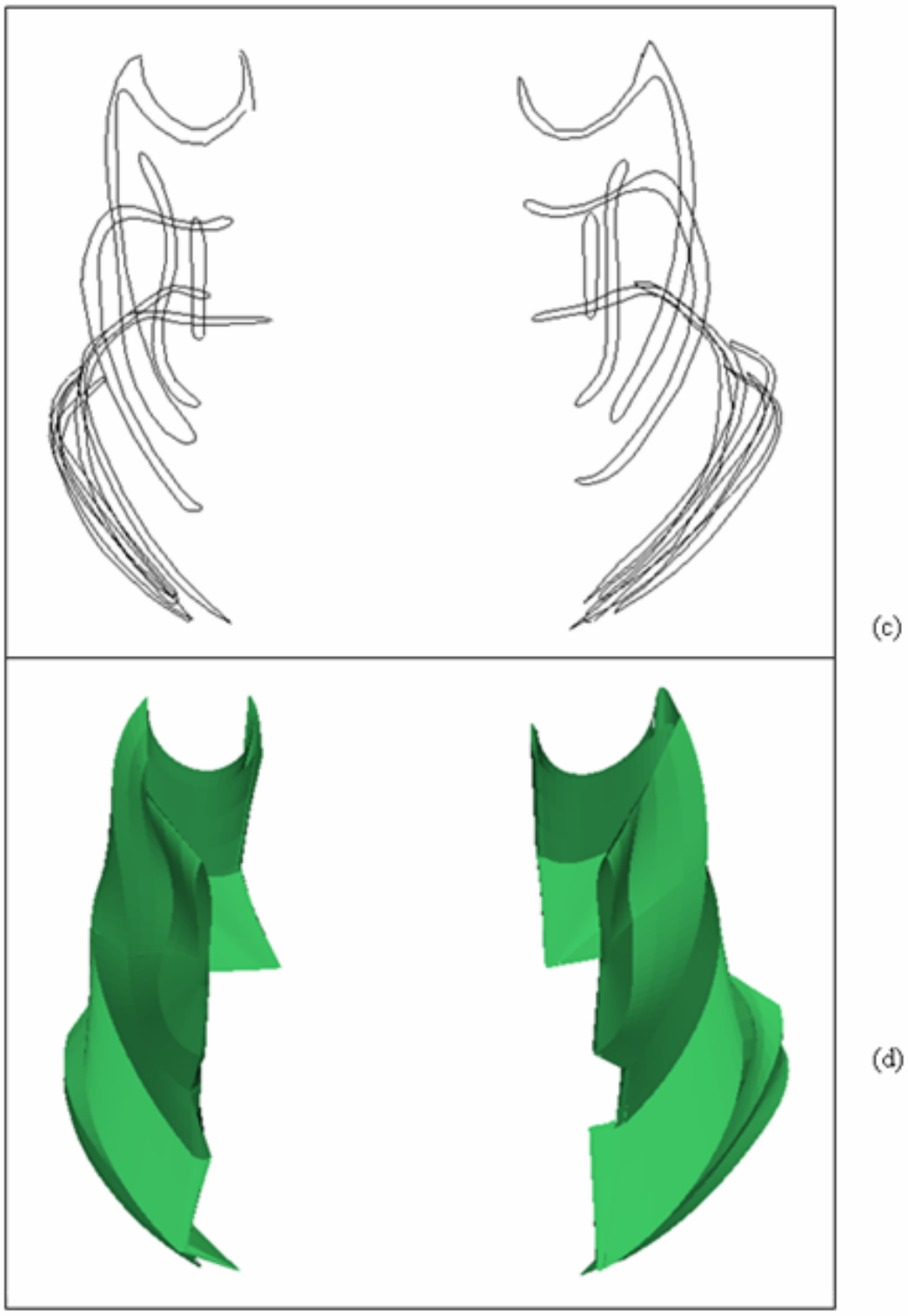


Fig. 23. Continued

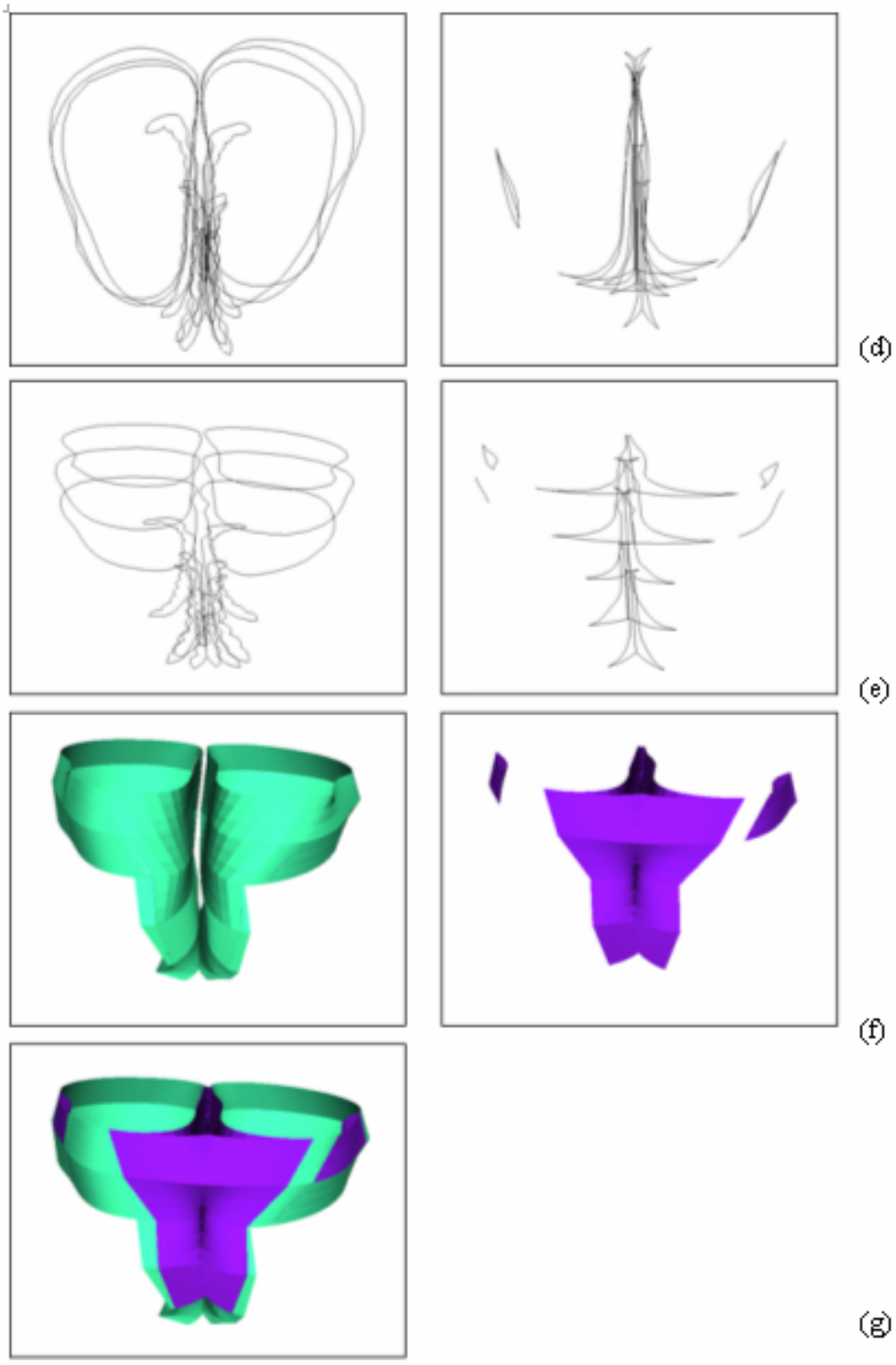


Fig. 23. Continued

3. Interpolated sections

Standard sections in traditional two-dimensional atlases are sparse with gaps between them. Thus, it is difficult to find a match when an experimental section falls between these gaps. To determine a histological slice's position and orientation, neuroscientists use its visible anatomical structures as landmarks, and compare an image of a given slice with a corresponding standard section, such as those provided in the stereotaxic atlas by Paxinos and Franklin [18]: first, a standard section that seems closest to the experimental section is located, and then two sections are superimposed to discern a number of anatomical structures. We use our surface reconstructions to interactively generate and visualize interpolated sections that are implied but missing in the set of two-dimensional standard sections (see Figure 24).

4. Image alignment

Between two consecutive histological slices, two problems in morphological shape differences arise: there may be anatomical changes from section to section, and there may be shape changes due to distortions induced by the sectioning process. To minimize tissue distortions, a high-precision motorized stage can be used during sectioning process, care can be taken during the data collection steps to reduce histological imperfections, and sections can be cut in a stereotaxic fashion by using appropriate orientation devices [133]. When a stack of experimental sections have comparable X-, Y-, and Z-resolutions, they are often formed into a three-dimensional histological block so that morphological and anatomical structures can be visualized in three-dimensions based on intensity variations. As with the individual two-dimensional slices, a histological volume can be superimposed with one or more standard two-dimensional sections to determine its position, orientation, and discernable anatomical structures.

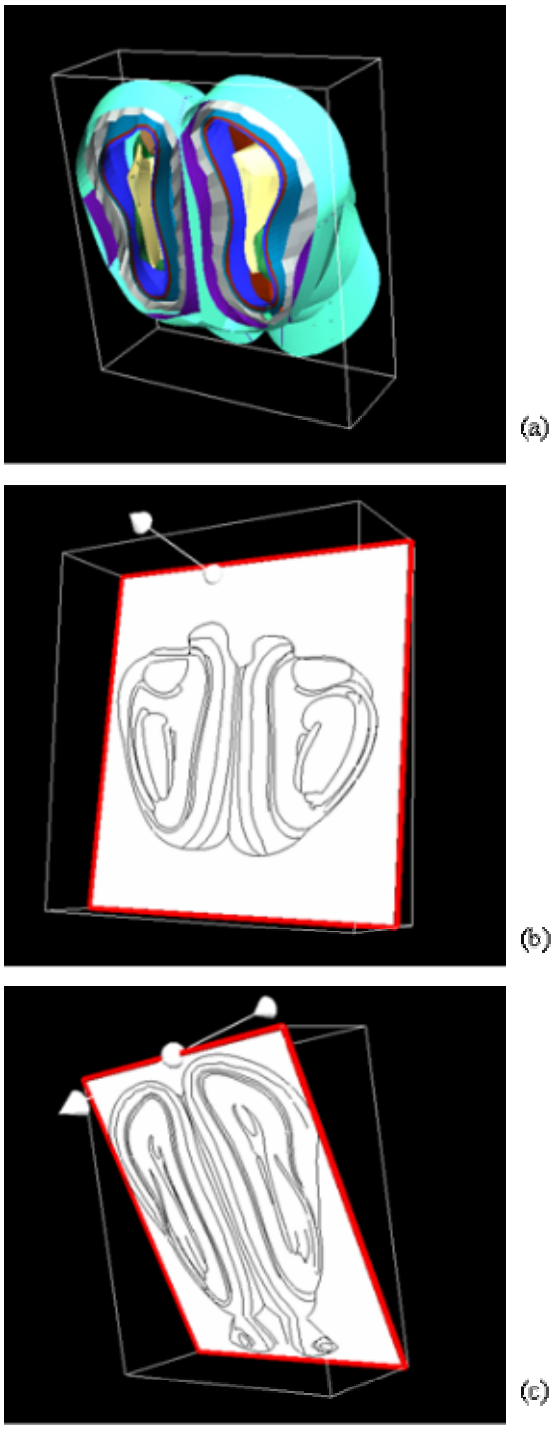


Fig. 24. Interpolated sections.

From our surface reconstructions shown in (a), we can interactively generate interpolated sections at arbitrary planes as shown in (b) and (c).

One of the first steps required in successful fusion of histology and standard sections in three-dimensions is the alignment of histological slices in a stereotaxic coordinate system. We use our surface reconstructions to guide alignment in a stack of histological slices, in which inter-slice resolution closely matches intra-slice resolution and is uniform. When the gap between two consecutive histological sections is small, it is assumed that shape differences in anatomy and morphology from one slice to the next differ only slightly [134]. In such cases, rigid transformations [135], [136] are recommended over elastic transformations [137]-[140] until anatomical and morphological changes from section to section can be clearly distinguished from local distortions due to the histological processing. Although elastic deformations may correct some of the observed morphological changes, they may also introduce false deformations that can compromise the overall data fusion [133]. We use translation, rotation, and non-uniform scale transformations. For the experimental results shown in Figure 25, we apply our transformations to our surface approximations rather than to the stack of images to improve efficiency, but we can easily apply an inverse transform to the stack of images to achieve the same results. For results shown Figure 25, we applied our surface approximations from the previous section to align 800 KESM-generated coronal histological sections of a mouse brain olfactory bulb specimen. The 800 images amount to a 1.5 mm x 2.2 mm x 0.8 mm volume of tissue and were scanned at 0.6 μm x 0.6 μm x 1 μm resolution.

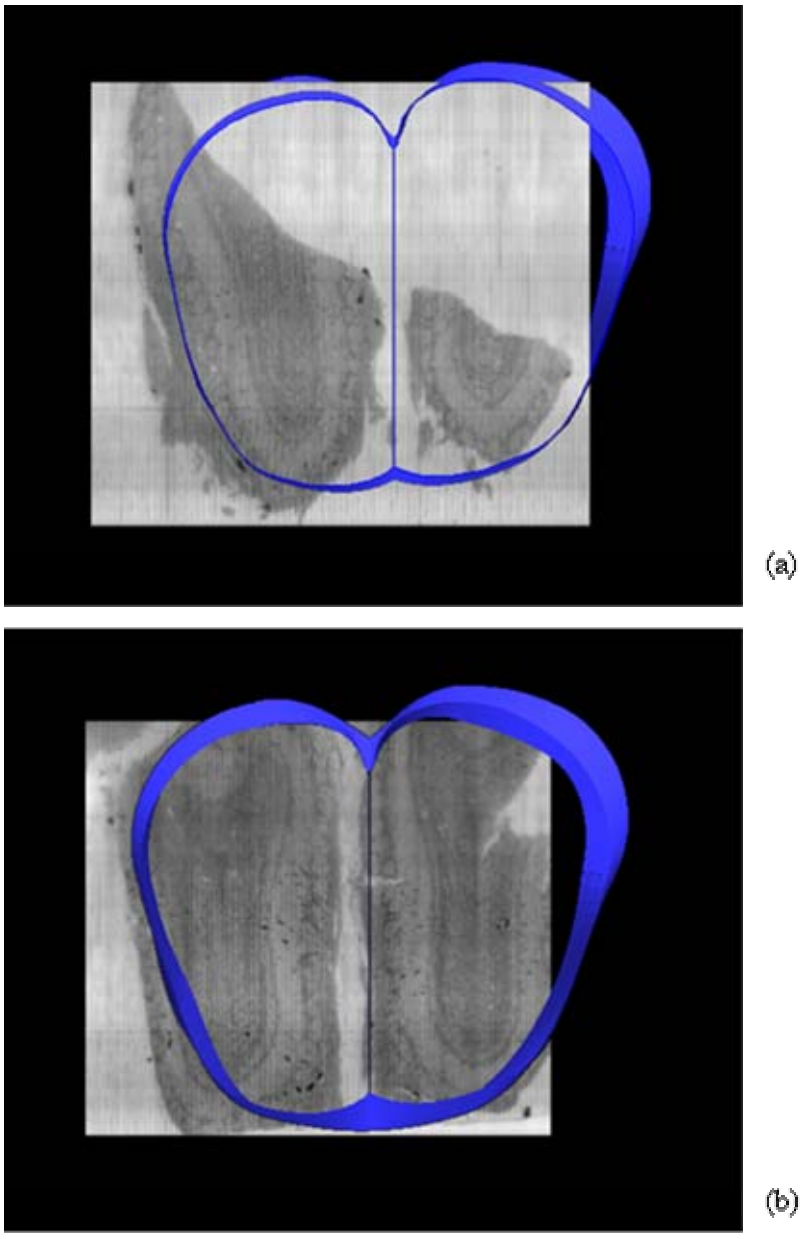
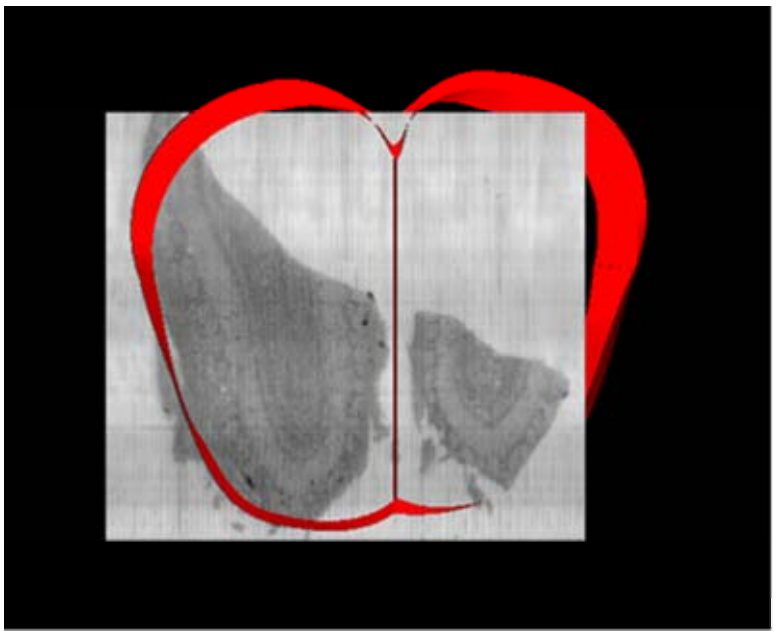
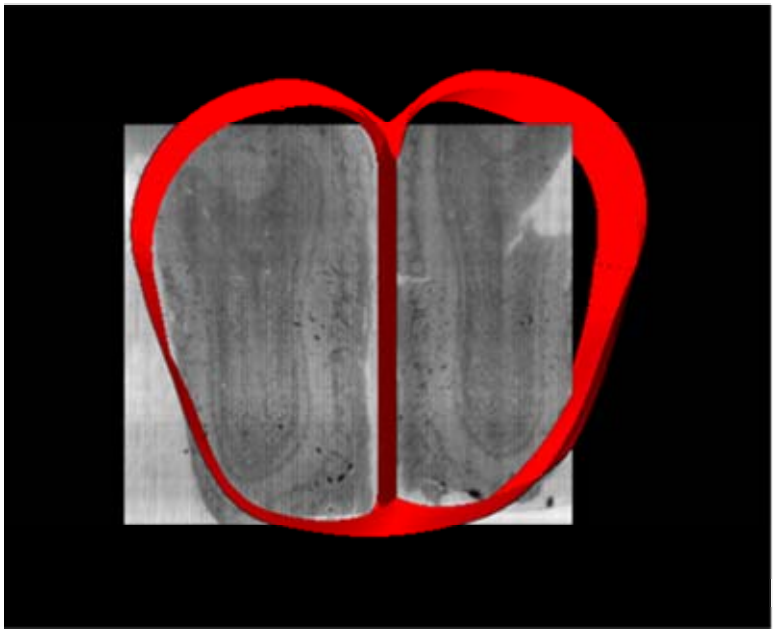


Fig. 25. Image alignment.

(a) The first of 800 images is aligned with our surface approximation via translation. (b) The same translation, applied to the last of the 800 images, indicates that the images are not in stereotaxic alignment. In (c) and (d), we apply a scale and rotation to our surface approximations to correct the misalignment detected between (a) and (b). (e) and (f) show the 400th image overlaid with the surface approximations before and after the scale and rotation, respectively. The images from KESM were acquired by David Mayerich and Jaerock Kwon.

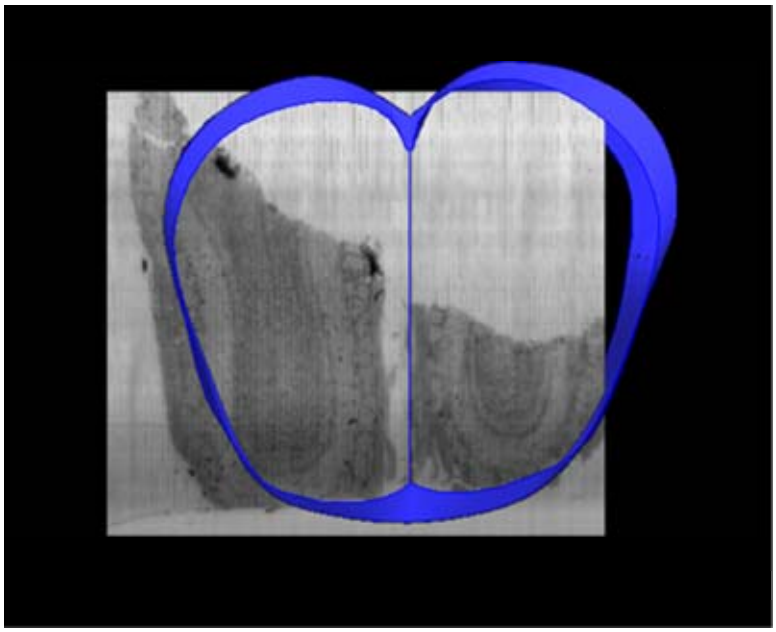


(c)

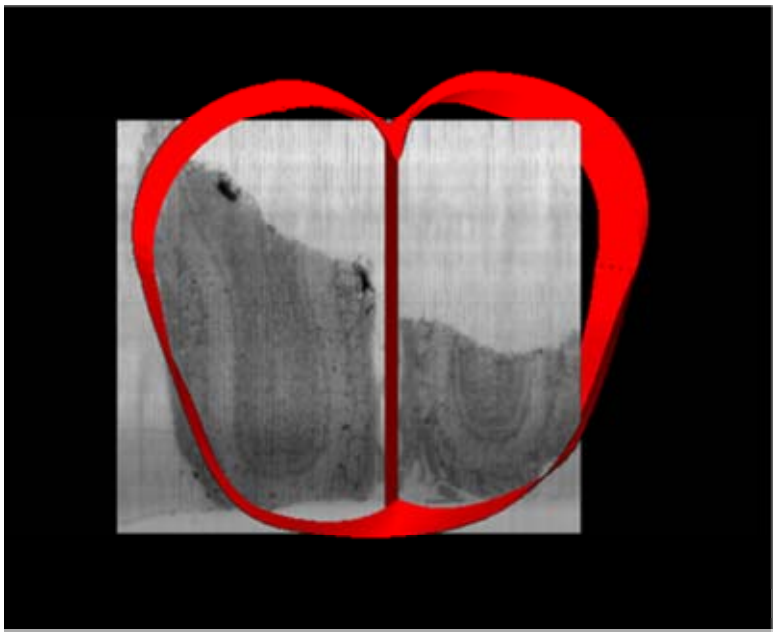


(d)

Fig. 25. Continued.



(e)



(f)

Fig. 25. Continued

CHAPTER VII

CONCLUSION AND FUTURE WORK

In this dissertation, we have investigated problems related to the geometric representation of neuronal and neuroanatomical data. In Chapters III and IV we studied the problem of the spatial representation and organization of neuronal geometry, as extracted from dense histological sections. In Chapters V and VI we studied the problem of reconstructing separating surfaces interpolated from sparse two-dimensional contours and its application to approximating surfaces of neuroanatomical structures.

A. Neuronal Geometry

The morphological complexity of neurons in mammalian brains has long been observed although mostly by focusing on a small group of neurons within an isolated brain area. Adding to the difficulty of describing neuronal geometry in the whole brain is that the cell bodies of interconnected neurons may not be bounded by a common small volume of interest, but bounded only by the extent of the entire brain. In this dissertation, we have presented a neuronal geometry representation method based on geometric primitives, and described how to organize spatial data across a brain specimen into an indexed composite volume. Representing observed synapses as directed links between geometric primitives captures the tangled and linked web-like structure of the neuronal interconnection data in literal fashion: as mapped from local or micro scale to potentially global scale.

The initial objective of our work on neuronal geometry was to design a schema for a *Mouse Brain Web* that archives the morphologies of individual neurons and putative synapses between the observed neurons of a mouse brain from a data set generated by KESM. After initial prototype design [141]-[143], we decided to focus on representing

neuronal geometry because it forms the basis for describing neuronal morphology and for inferring synapses based on geometric proximity. Figure 26 shows a conceptual sketch of how a *Mouse Brain Web* from KESM-generated data set can lead to anatomically correct modeling of mouse brain networks, and to subsequently allow the mapping of anatomically correct networks to physiologically correct network simulation.

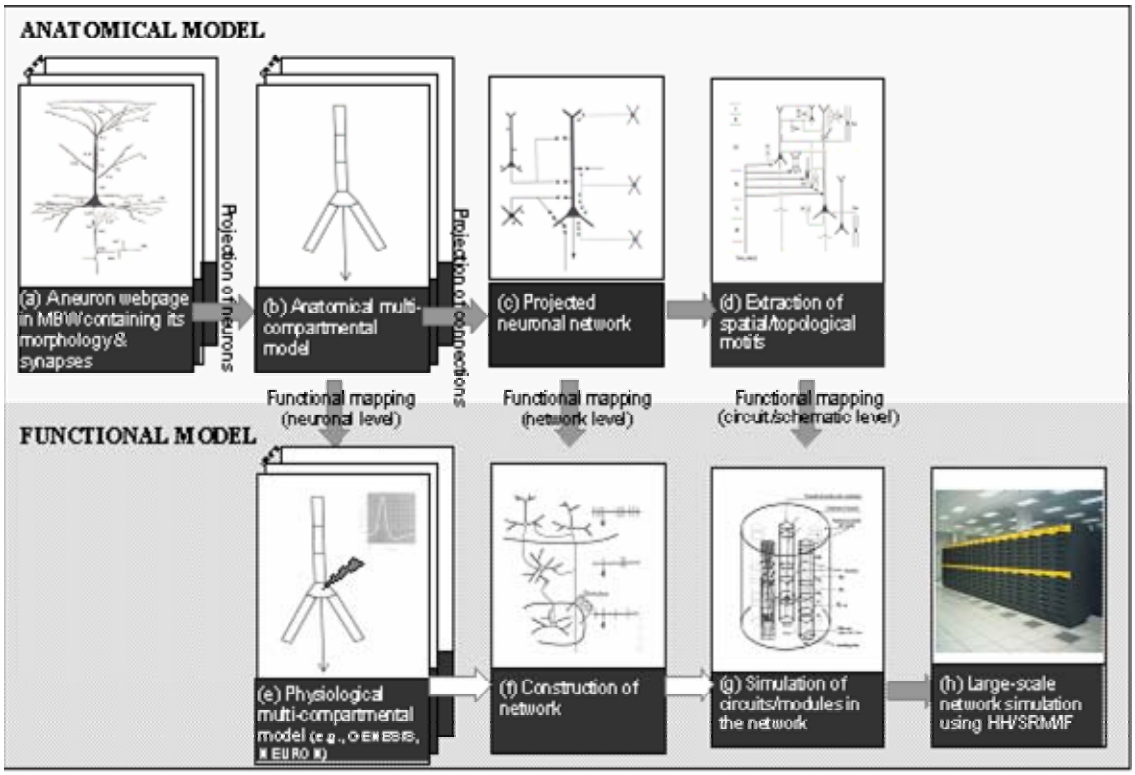


Fig. 26. The *Mouse Brain Web* and its anatomical and functional use.

Data generation, compression, segmentation, reconstruction, editing, visualization, and storage are some of the myriad issues we face before a *Mouse Brain Web* can be realized. KESM's unique ability to provide data for modeling fundamental network

organization in mammalian brains is revolutionary, but the sheer volume and unique nature of brain's structural data necessitate development of new computational tools to manage and gain insights from this complex information. Since anatomically correct modeling relies on geometric models of individual neurons, our work forms a common denominator for each stage of anatomical and functional simulation shown in Figure 26. Our representation can facilitate navigating across a composite volume, and is also extensible to incorporate and provide links to supplementary physiological data related to neuronal morphology, such as ion channel densities. As a *Mouse Brain Web* develops, our work will need to be adapted and extended to accommodate new technological developments.

B. Reconstruction of Neuroanatomical Structures

Surface approximation of neuroanatomical structures from sparse cross-sectional contours has proven to be difficult due to the presence of abutting, shared boundary surfaces that are not handled by traditional boundary-representation (B-rep) data structures and surfaces-from-contours algorithms (see Section V.B). We presented a topology-based data structure, the topology graph, and an algorithm based on parametric curve fitting and triangulation to model surfaces separating multiple regions in space. We also applied our algorithm to approximate the boundary surfaces of anatomical surfaces in mouse brain olfactory bulb. Improved knowledge of central olfactory circuit anatomy and physiology is essential to understand olfactory coding, and may provide fundamental insights about the underlying computational features that have driven the selective expansion of cortical structures in the evolution of the mammalian brain [144]. Our work currently provides recovery and visualization of geometric data in three-dimensions that are missing in a series of two-dimensional maps of mouse brain olfactory bulb.

The advantage of having a topology graph is that it allows reconstruction of three-dimensional surfaces automatically, independent of shape complexity and that it avoids ambiguities by solving the correspondence problem. Another advantage of our method is that by combining parametric curve fitting and graph-directed triangulation, it ensures that our abutting or shared surface patches are precisely coincident, resulting in separating surfaces that are geometrically consistent. The disadvantage is that our topology graph has to be built manually relying on domain expert knowledge present in two-dimensional cross-sectional diagrams. This manual process is similar to what early CAD package designers faced, namely that they had to focus their efforts on "the most tedious, time-consuming, and unrewarding aspect of conventional design: the process of converting a designer's sketches and notes into finished engineering drawings—the drafting process" [113]. However, this drafting process has proven to be necessary and justified because, although only modest gains in efficiency were possible in creating the first version of a design, tremendous gains were realized when design modifications were needed [113].

Our proposed future work on the reconstruction of neuroanatomical structures falls into three broad categories. First is to apply our method to model various anatomical regions in mouse and other mammalian brains, that is, to continue on with the 'drafting process' of converting standard two-dimensional brain maps into digital models. The bulk of this work lies in construction of topology graphs, as our current solution provides a fully automated, geometrically consistent reconstruction of separating surfaces once a topology graph is given. Thus, second is to semi-automate the topology graph construction process while maintaining its two purposes: representing the topological skeleton of object surfaces, and directing separating surface generation between contours. Different anatomical structures exhibit distinct textures in histological cross-sections, and a texture-based region classification scheme may partially automate the construction of topology graphs. Given regions classified by texture, if a fully labeled set of boundary contours is available in digital form, the intra-

cross-sectional topology graphs for those regions can be automatically constructed. When such a set is not available, the boundary classification of contours may still require manual intervention, based on the similarity or near-symmetry of shape and texture information. However, constructing the inter-cross-sectional topology graphs may still need domain expert consultation, visual validation, and manual intervention since to our knowledge, there is no known method to automatically solve the correspondence problem between cross-sectional contours that depict multiple regions, and to generate precisely coincident, geometrically consistent surfaces separating different regions. Third is to apply our digital reconstructions to reap similar benefits gained by early CAD designers when design modifications were needed after the initial drafting process. Our benefits will come not from modifying a design but from providing a flexible new template to aid the map-making process for individual brains and to perform brain-to-brain comparisons. Performing brain-to-brain comparisons, or warping one brain to the other, has been extensively studied [145]. Most existing approaches are based on intensity patterns or anatomical landmarks placed on two- or three-dimensional grids. Several approaches based on elastic deformations have been proposed [137]-[139], [146], [147], but are mostly limited to a one-to-one transformation that deforms one source curve to one target curve. We foresee a need for new approaches which transform surfaces separating multiple neuroanatomical regions to comparable brain data from different imaging modalities.

REFERENCES

[1] A. M. Turing, "Computing machinery and intelligence," *Mind,* vol. 59, pp. 433-460, 1950.

[2] J. R. Cox, "Foreword," in *Neuroinformatics: An Overview of the Human Brain Project*, S. H. Koslow and M. F. Huerta, Eds. Mahwah, NJ: Lawrence Erlbaum Associates, 1997, pp. vii-ix.

[3] L. W. Swanson, "Mapping the human brain: past, present, and future," *Trends Neurosci.,* vol. 18, pp. 471-474, 1995.

[4] M. Chicurel, "Databasing the brain," *Nature,* vol. 406, pp. 822-825, 2000.

[5] V. Braitenberg and A. Schuz, *Cortex: Statistics and Geometry of Neuronal Connectivity*, 2nd ed. Berlin: Springer-Verlag, 1998.

[6] S. Ramon y Cajal, *Histology of the Nervous System of Man and Vertebrates* vol. 1. New York: Oxford University Press, 1995.

[7] R. M. Durbin, "Studies on the development and organization of the nervous system of caenorhabditis elegans", Ph.D. dissertation, King's College, University of Cambridge, Cambridge, U.K.,1987.

[8] J. G. White, E. Southgate, J. N. Thomson, and S. Brenner, "The structure of the nervous system of the nematode caenorhabditis elegans," *Phil. Trans. Royal Soc. London, Series B, Biol. Sci.,* vol. 314, pp. 1-340, 1986.

[9] J. T. Robinson, "The K-D-B-tree: a search structure for large multidimensional dynamic indexes " in *Proc. 1981 ACM SIGMOD Int. Conf. Management of Data*, 1981, pp. 10-18.

[10] H. Samet, "The quadtree and related hierarchical data structures," *ACM Comput. Surv. ,* vol. 16, pp. 187-260, 1984.

[11] M. Tamminen and H. Samet, "Efficient octree conversion by connectivity labeling " in *Proc. 11th Conf. Computer Graphics and Interactive Techniques*, 1984, pp. 43-51

[12] A. Guttman, "R-trees: a dynamic index structure for spatial searching," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1984, pp. 47-57.

[13] N. Roussopoulos and D. Leifker, "Direct spatial search on pictorial databases using packed R-trees," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1985, pp. 17-31.

[14] N. Beckman, H. P. Kriegel, R. Schneider, and B. Seeger, "The R*-tree: an efficient and robust access method for points and rectangles," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 1990, pp. 332-331.

[15] G. M. Shepherd, *The Synaptic Organization of the Brain*, 5th ed. New York: Oxford University Press, 2004.

[16] P. R. Hof, W. G. Young, F. E. Bloom, P. V. Belichenko, and M. R. Celio, *Comparative Cytoarchitectonic Atlas of the C57BL/6 and 129/Sv Mouse Brains*. Amsterdam: Elsevier Science Health Science Div, 2000.

[17] D. M. Jacabowitz and L. C. Abbott, *Chemoarchitectonic Atlas of the Developing Mouse Brain*. Boca Raton, FL: CRC Press, 1997.

[18] G. Paxinos and K. B. J. Franklin, *The Mouse Brain in Stereotaxic Coordinates*, 2nd ed. San Diego: Academic Press, 2001.

[19] C. Gitlin, J. O'Rourke, and V. Subramanian, "On reconstructing polyhedra from parallel slices," *Internat. J. Comput. Geom. Appl.,* vol. 6, pp. 103-122, 1996.

[20] C. Bajaj, E. Coyle, and K. Lin, "Arbitrary topology shape reconstruction from planar cross sections," *Graph. Models Image Process.,* vol. 58, pp. 524-543, 1996.

[21] G. Barequet, M. T. Goodrich, A. Levi-Steiner, and D. Steiner, "Straight-skeleton based contour interpolation," in *Proc. 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, 2003, pp. 119-127.

[22] S. Batnitzky, H. I. Price, N. Cook, N. T. Cook, and S. J. Dwyer III, "Three-dimensional computer reconstruction from surface contours for head CT examinations," *J. Comput. Assist. Tomogr.,* vol. 5, pp. 60-67, 1981.

[23] J.-D. Boissonnat, "Shape reconstruction from planar cross sections," *Comput. Vision Graph. Image Process.,* vol. 44, pp. 1-29, 1988.

[24] H. N. Christiansen and T. W. Sederberg, "Conversion of complex contour line definitions into polygonal element mosaics," *Comput. Graph.,* vol. 13, pp. 187-192, 1978.

[25] H. Fuchs, Z. M. Kedem, and S. P. Uselton, "Optimal surface reconstruction from planar contours," *Commun. ACM,* vol. 20, pp. 693- 702, 1977.

[26] B. Geiger, "Three-dimensional modeling of human organs and its application to diagnosis and surgical planning", Ph.D. dissertation, INRIA Sophia Antipolis, Sophia Antipolis, France, 1993.

[27] M. W. Jones and M. Chen, "A new approach to the construction of surfaces from contour data," *Comput. Graph. Forum,* vol. 13, pp. C75-C84, 1994.

[28] E. Keppel, "Approximating complex surfaces by triangulation of contour lines," *IBM J. Res. Develop.,* vol. 19, pp. 2-11, 1975.

[29] D. Meyers, S. Skinner, and K. Sloan, "Surface from contours," *ACM Trans. Graphics,* vol. 11, pp. 228-258, 1992.

[30] K. R. Sloan Jr. and J. Painter, "From contours to surfaces: testbed and initial results," in *SIGCHI/GI Conf. Human Factors in Computing Systems and Graphics Interface*, 1986, pp. 115-120.

[31] K. R. Sloan Jr. and J. Painter, "Pessimal guesses may be optimal: a counter-intuitive search result," *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 10, pp. 949-955, 1988.

[32] J. R. Rossignac and A. A. G. Requicha, "Constructive non-regularized geometry," *Comput. Aided Des.,* vol. 23, pp. 21-32, 1991.

[33] A. W. Toga, K. Ambach, B. Quinn, M. Hutchin, and J. S. Burton, "Postmortem anatomy from cyrosectioned whole human brain," *J. Neurosci. Methods,* vol. 54, pp. 239-252, 1994.

[34] M. Minsky, *Microscopy Apparatus*, U.S. Pat. No. 3,013,467, 1961.

[35] W. Denk and K. Svoboda, "Photon upmanship: why multiphoton imaging is more than a gimmick," *Neuron,* vol. 18, pp. 351-357, 1997.

[36] W. Denk, J. H. Strickler, and W. W. Webb, "Two-photon laser scanning fluorescence microscopy," *Science,* vol. 248, pp. 73-76, 1990.

[37] R. Fernandez-Gonzalez, A. Jones, E. Garcia-Rodriguez, P. Y. Chen, A. Idica, S. J. Lockett, M. H. Barcellos-Hoff, and C. Oritiz-De-Solorzano, "System for combined three-dimensional morphological and molecular analysis of thick tissue specimens," *Microsc. Res. Tech.,* vol. 59, pp. 522-530, 2002.

[38] A. Egner, S. Jakobs, and S. W. Hell, "Fast 100-nm resolution three-dimensional microscope reveals structural plasticity of mitochondria in live yeast," *PNAS,* vol. 99, pp. 3370-3375, 2002.

[39] F. Capani, M. E. Martone, T. J. Deerinck, and M. H. Ellisman, "Selective localization of high concentrations of F-actin in subpopulations of dendritic spines in rat central nervous system: a three-dimensional electron microscopic study," *J. Comp. Neurol.,* vol. 435, pp. 156-170, 2001.

[40] M. E. Martone, A. Gupta, M. Wong, X. Qian, G. Sosinsky, B. Ludascher, and M. H. Ellisman, "A cell-centered database for electron tomographic data," *J. Struct. Biol.,* vol. 138, pp. 145-155, 2002.

[41] B. H. McCormick, *System and Method for Imaging an Object*, U.S. Pat. No. 6,744,572, 2004.

[42] P. S. Tsai, B. Friedman, A. I. Ifrraguerri, B. D. Thompson, V. Lev-Ram, C. B. Schaffer, Q. Ziong, R. Y. Tsien, J. A. Squier, and D. Kleinfeld, "All-optical histology using ultrashort laser pulses," *Neuron,* vol. 39, pp. 27-41, 2003.

[43] M. P. Young and J. W. Scannell, "Brain structure-function relationships: advances from neuroinformatics," *Philos. Trans. R. Soc. Lond. B Biol. Sci.,* vol. 355, pp. 3-6, 2000.

[44] R. Kotter, "Neuroscience databases: tools for exploring brain structure-function relationships," *Philos. Trans. R. Soc. Lond. B Biol. Sci.,* vol. 356, pp. 1111-1120, 2001.

[45] D. J. Felleman and D. C. van Essen, "Distributed hierarchical processing in the primate cerebral cortex," *Cereb. Cortex,* vol. 1, pp. 1-47, 1991.

[46] M. P. Young, "The organization of neural systems in the primate cerebral cortex," *Proc. Royal Soc. Lond. B. Biol. Sci.,* vol. 252, pp. 13-18, 1993.

[47] J. W. Scannell, C. Blakemore, and M. P. Young, "Analysis of connectivity in the cat cerebral cortex," *J. Neurosci.,* vol. 15, pp. 1463-1483, 1995.

[48] J. W. Scannell, G. A. Burns, C. C. Hilgetag, M. A. O'Neil, and M. P. Young, "The connectional organization of the cortico-thalamic system of the cat," *Cereb. Cortex,* vol. 9, pp. 277-299, 1999.

[49] G. A. Burns and M. P. Young, "Analysis of the connectional organization of neural systems associated with the hippocampus in rats," *Philos. Trans. Royal Soc. Lond. B. Biol. Sci.,* vol. 356, pp. 1187-1208, 2000.

[50] K. E. Stephan, L. Kamper, A. Bozkurt, G. A. Burns, M. P. Young, and L. Kobbelt, "Advanced database methodology of Connectivity data on the Macaque brain (CoCoMac)," *Philos. Trans. R. Soc. Lond. B Biol. Sci.,* vol. 356, pp. 1159-1186, 2001.

[51] W. A. Press, B. A. Olshausen, and D. C. van Essen, "A graphical anatomical database of neural connectivity," *Philos. Trans. R. Soc. Lond. B Biol. Sci.,* vol. 356, pp. 1147-1157, 2001.

[52] P. T. Fox and J. L. Lancaster, "Neuroscience on the net," *Science,* vol. 266, pp. 994-996, 1994.

[53] P. T. Fox, S. Mikiten, G. Davis, and J. L. Lancaster, "BrainMap: a database of human functional brain mapping," in *Functional Neuroimaging: Technical Foundations*, R. W. Thatcher, M. Hallett, T. Zeffiro, and E. Y. John, Eds. New York: Academic Press, 1994, pp. 95-105.

[54] P. E. Roland and K. Zilles, "The developing European computerized human brain database for all imaging modalities," *Neuroimage,* vol. 4, pp. S39-S47, 1996.

[55] Cell Centered Database, http://ncmir.ucsd.edu/CCDB/, accessed November, 2006.

[56] J. C. Fiala and K. M. Harris, "Extending unbiased stereology of brain ultrastructure to three-dimensional volumes," *J. Am. Med. Inform. Assoc.,* vol. 8, pp. 1-16, 2001.

[57] Synapse Web, http://synapse-web.org/, accessed November, 2006.

[58] Biomedical Informatics Research Network, http://www.nbirn.net/index.htm, accessed November, 2006.

[59] J. D. van Horn, J. B. Woodward, G. Simonds, B. Vance, J. S. Grethe, M. Montague, J. Aslam, D. Rus, D. Rockmore, and M. S. Gazzaniga, "The fMRI center: software tools for neuroimaging data management, inspection, and sharing," in *Neuroscience Databases: a Practical Guide*, R. Kotter, Ed. Boston: Kluwer Academic Publishers, 2003, pp. 221-235.

[60] Mouse Brain Library, http://www.mbl.org/, accessed November, 2006.

[61] N. H. Goddard, M. Hucka, F. Howell, H. Cornelis, K. Shankar, and D. Beeman, "Towards NeuroML: model description methods for collaborative modeling in neuroscience," *Philos. Trans. R. Soc. Lond. B Biol. Sci.,* vol. 356, pp. 1209-1228, 2001.

[62] BrainML, http://www.brainml.org/, accessed November, 2006.

[63] D. Gardner, K. H. Knuth, M. Abato, S. M. Erde, T. White, R. DeBellis, and E. P. Gardner, "Common data model for neuroscience data and data model exchange," *J. Am. Inform. Assoc.,* vol. 8, pp. 17-33, 2001.

[64] R. C. Cannon, D. A. Turner, G. K. Pyapali, and H. V. Wheal, "An on-line archive of reconstructed hippocampal neurons," *J. Neurosci. Methods,* vol. 84, pp. 49-54, 1998.

[65] Neurolucida, http://www.microbrightfield.com, accessed November, 2006.

[66] G. A. Ascoli, J. L. Krichmar, S. J. Nasuto, and S. L. Senft, "Generation, description and storage of dendritic morphology data," *Philos. Trans. R. Soc. Lond. B Biol. Sci.,* vol. 356, pp. 1131-1145, 2001.

[67] J. M. Bower, D. Beeman, and M. Hucka, "GENESIS simulation system," in *The Handbook of Brain Theory and Neural Networks*, 2nd ed, M. A. Arbib, Ed. Cambridge, MA: MIT Press, 2003, pp. 475-478.

[68] M. L. Hines and N. T. Carnevale, "The NEURON simulation environment," *Neural Comput.,* vol. 9, pp. 1179-1209, 1997.

[69] G. A. Burns, "Knowledge management of the neuroscientific literature: the data model and underlying strategy of the NeuroScholar system," *Philos. Trans. R. Soc. Lond. B Biol. Sci.,* vol. 356, pp. 1187-1208, 2001.

[70] F. E. Bloom, W. G. Young, and Y. M. Kim, *Brain Browser*. San Diego: Academic Press, 1990.

[71] D. C. van Essen and H. A. Drury, "Structural and functional analyses of human cerebral cortex using a surface-based atlas," *J. Neurosci.,* vol. 17, pp. 7079-7102, 1997.

[72] D. C. van Essen, H. A. Drury, J. Dickson, J. Harwell, D. Hanlon, and C. H. Anderson, "An integrated software suite for surface-based analyses of cerebral cortex," *J. Am. Med. Inform. Assoc.,* vol. 8, pp. 443-459, 2001.

[73] High Resolution Mouse Brain Atlas, http://www.hms.harvard.edu/research/brain/, accessed November, 2006.

[74] R. L. Sidman, *Atlas of the Mouse Brain and Spinal Cord*. Cambridge, MA: Harvard University Press, 1971.

[75] N. Kovacevic, J. T. Henderson, E. Chan, N. Lifshitz, J. Bishop, A. C. Evans, R. M. Henkelman, and X. J. Chen, "A three-dimensional MRI atlas of the mouse brain with estimates of the average and variability," *Cereb. Cortex,* vol. 15, pp. 639-645, 2005.

[76] Y. Ma, P. R. Hof, S. C. Grant, S. J. Blankband, R. Bennett, L. Slatest, M. D. Mcguigan, and H. Benveniste, "A three dimensional digital atlas database of the adult C57BL/6J mouse brain by magnetic resonance microscopy," *Neuroscience,* vol. 135, pp. 1203-1215, 2005.

[77] Allen Institute for Brain Science, http://www.brainatlas.org, accessed November, 2006.

[78] J. C. Mazziotta, A. W. Toga, A. C. Evans, P. T. Fox, J. Lancaster, K. Zilles, R. P. Woods, T. Paus, G. Simpson, B. Pike, C. J. Holmes, D. L. Collins, P. M. Thompson, D. MacDonald, M. Iacoboni, T. Schormann, K. Amunts, N. Palomero-Gallagher, S. Geyer, L. Parsons, K. L. Narr, N. Kabani, G. Le Goualher, D. Boomsma, T. Cannon, R. Kawashima, and B. Mazoyer, "A probabilistic atlas and reference system for the human brain: International Consortium for Brain Mapping (ICBM)," *Philos. Trans. R. Soc. Lond. B Biol. Sci.,* vol. 356, pp. 1293-1322, 2001.

[79] E. MacKenzie-Graham, F. Lee, I. D. Dinov, M. Bota, D. W. Shattuck, S. Ruffins, H. Yuan, F. Konstantinidis, A. Pitiot, Y. Ding, G. Hu, R. Jacobs, and A. W. Toga, "A multimodal, multidimensional atlas of the C57BL/6J mouse brain," *J. Anat.,* vol. 204, pp. 93-102, 2004.

[80] Mouse Atlas Project, http://www.loni.ucla.edu/MAP/, accessed November, 2006.

[81] H. Nyquist, "Certain topics in telegraph transmission theory," *Trans. AIEE,* vol. 47, pp. 617-644, 1928.

[82] B. H. McCormick, "Development of the brain tissue scanner", Tech. report, Dept. Comput. Sci., Texas A&M Univ., College Station, TX, 2002.

[83] B. H. McCormick, "Brain tissue scanner enables brain microstructure surveys," *Neurocomputing,* vol. 44-46, pp. 1113-1118, 2002.

[84] B. H. McCormick and D. M. Mayerich, "Three-dimensional imaging using knife-edge scanning microscope," *Microsc. Microanal.,* vol. 10 (Suppl 2), pp. 1466-1467, 2004.

[85] L. C. Abbott and C. Sotelo, "Ultrastructural analysis of catecholaminergic innervation in weaver and normal mouse cerebellar cortices," *J. Comp. Neurol.,* vol. 426, pp. 316-329, 2000.

[86] J. M. Allman, *Evolving Brains*. New York: Scientific American Library, 1999.

[87] M. F. Bear, B. W. Connors, and M. A. Paradiso, *Neuroscience: Exploring the Brain*, 2nd ed. Baltimore, MD: Lippincott Williams & Wilkins, 2001.

[88] M. Mantyla, *An Introduction to Solid Modeling*. Rockville, MD: Computer Science Press, 1988.

[89] C. Giertsen, A. Halvorsen, and P. R. Flood, "Graph-directed modelling from serial sections," *Visual Comput.,* vol. 6, pp. 284-290, 1990.

[90] J. Bloomenthal and K. Ferguson, "Polygonization of non-manifold implicit surfaces," in *Proc. 22nd Conf. Computer Graphics and Interactive Techniques*, 1995, pp. 309-316

[91] H. Xie, J. Wang, J. Hua, H. Qin, and A. Kaufman, "Piecewise C^1 continuous surface reconstruction of noisy point clouds via local implicit quadric regression," in *Proc. IEEE Visualization '03*, 2003, pp. 91-98.

[92] J.-D. Boissonnat, "Geometric structures for three-dimensional shape representation," *ACM Trans. Graphics,* vol. 3, pp. 266-286, 1984.

[93] H. Edelsbrunner and E. P. Mucke, "Three-dimensional alpha shapes," *ACM Trans. Graphics,* vol. 13, pp. 43-72, 1994.

[94] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle, "Surface reconstruction from unorganized points," in *Proc. 19th Conf. Computer Graphics and Interactive Techniques*, 1992, pp. 71-78.

[95] W. E. Lorensen and H. E. Cline, "Marching cubes: a high resolution 3D surface construction algorithm," in *Proc. 14th Conf. Computer Graphics and Interactive Techniques*, 1987, pp. 163-169.

[96] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proc. 23rd Conf. Computer Graphics and Interactive Techniques*, 1996, pp. 303-312.

[97] N. Amenta, M. Bern, and M. Kamvysselis, "A new Voronoi-based surface reconstruction algorithm," in *Proc. 25th Conf. Computer Graphics and Interactive Techniques*, 1998, pp. 415-421.

[98] A. Adamson and M. Alexa, "Approximating and intersecting surfaces from points," in *Proc. EG Symposium on Geometry Processing*, 2003, pp. 245-254.

[99] A. Adamson and M. Alexa, "Ray tracing point set surfaces," in *Proc. Shape Modeling International*, 2003, pp. 272-279.

[100] M. Alexa, D. Behr, D. Cohen-Or, S. Fleishman, D. Levin, and C. T. Silva, "Computing and rendering point set surfaces," *IEEE Trans. Vis. Comput. Graph.,* vol. 9, pp. 3-15, 2003.

[101] N. Amenta and Y. Kil, "Defining point set surfaces," *ACM Trans. Graphics,* vol. 23, pp. 264-270, 2004.

[102] S. Fleishman, D. Cohen-Or, M. Alexa, and C. T. Silva, "Progressive point set surfaces," *ACM Trans. Graphics,* vol. 22, pp. 997-1011, 2003.

[103] D. Levin, "Mesh-independent surface interpolation," in *Geometric Modeling for Scientific Visualization*, G. Brunnett, B. Hamann, K. Mueller, and L. Linsen, Eds. Heidelberg: Springer-Verlag, 2003, pp. 37-49.

[104] M. Zwicker, M. Pauly, O. Knoll, and M. Gross, "Pointshop 3d: An interactive system for point-based surface editing," in *Proc. 29th Conf. Computer Graphics and Interactive Techniques*, 2002, pp. 322-329.

[105] H. Edelsbrunner and J. Harer, "Jacobi sets of multiple Morse functions," in *Foundations of Computational Mathematics*, F. Cucker, R. DeVore, P. Olver, and E. Sueli, Eds. London: Cambridge Univ. Press, 2002, pp. 37-57.

[106] B. I. Soroka, "Generalized cones from serial sections," *Comput. Graph. Image Process.,* vol. 15, pp. 154-166, 1981.

[107] G. M. Nielson and R. Franke, "Computing the separating surface for segmented data," in *Proc. IEEE Visualization '97*, 1997, pp. 229-233.

[108] H. Muller, "Boundary extraction for rasterized motion planning", Tech. report, Universitat Dortmund, Dortmund, Germany, 1995.

[109] D. Weinstein, "Scanline surfacing: building separating surfaces from planar contours," in *Proc. IEEE Visualization '00*, 2000, pp. 283-289.

[110] T. Ju, J. Warren, J. Carson, G. Eichele, C. Thaller, W. Chiu, M. Bello, and I. Kakadiaris, "Building 3D surface networks from 2D curve networks with application to anatomical modeling," *Visual Comput.,* vol. 21, pp. 764-773, 2005.

[111] K. S. Bonnell, M. A. Duchaineau, D. R. Schikore, B. Harmann, and K. I. Joy, "Material interface reconstruction," *IEEE Trans. Vis. Comput. Graph.,* vol. 9, pp. 500-511, 2003.

[112] K. S. Bonnell, D. R. Schikore, K. I. Joy, M. Duchaineau, and B. Harmann, "Constructing material interfaces from data sets with volume-fraction information," in *Proc. IEEE Visualization '00*, 2000, pp. 367-372.

[113] M. J. Muuss and L. A. Butler, "Combinatorial solid geometry, boundary representations, and non-manifold geometry," in *State of the Art in Computer Graphics: Visualization and Modeling*, D. F. Rogers and R. A. Earnshaw, Eds. New York: Springer-Verlag, 1991, pp. 185-223.

[114] K. Kaneda, K. Harada, E. Nakamae, M. Yasuda, and A. G. Sato, "Reconstruction and semi-transparent display method for observing inner structure of an object consisting of multiple surfaces," *Visual Comput.,* vol. 3, pp. 137-144, 1987.

[115] Y. Shinagawa and T. L. Kunii, "Constructing a Reeb graph automatically from cross sections," *IEEE Comput. Graph. Appl.,* vol. 11, pp. 44-51, 1991.

[116] S. Biasotti, "Reeb graph representation of surfaces with boundary," in *Proc. Shape Modeling International*, 2004, pp. 371-374.

[117] Y. Xiao, P. Siebert, and N. Werghi, "A discrete Reeb graph approach for the segmentation of human body scans," in *Proc. 4th Int. Conf. 3D Digital Imaging and Modeling*, 2003, pp. 378-385.

[118] M. Attene, S. Biasotti, and M. Spagnuolo, "Shape understanding by contour-driven retiling," *Visual Comput.,* vol. 19, pp. 127-138, 2003.

[119] K. Cole-McLaughlin, H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci, "Loops in Reeb graphs of 2-manifolds," in *Proc. 19th ACM Symposium on Computational Geometry*, 2003, pp. 344-350.

[120] Z. J. Wood, M. Desburn, P. Schroder, and D. Breen, "Semi-regular mesh extraction from volumes," in *Proc. IEEE Visualization '00*, 2000, pp. 275-282, 567.

[121] A. Verroust and F. Lazarus, "Extracting skeletal curves from 3D scattered data," *Visual Comput.,* vol. 16, pp. 15-25, 2000.

[122] M. Hilaga, Y. Shinagawa, T. Komura, and T. I. Kunii, "Topology matching for fully automatic similarity estimation of 3D shapes," in *Proc. 28th Conf. Computer Graphics and Interactive Techniques*, 2001, pp. 203-212.

[123] J. W. H. Tangelder and R. C. Veltkamp, "A survey of content based 3D shape retrieval methods," in *Proc. Shape Modeling Applications*, 2004, pp. 145-156.

[124] G. Cong and B. Parvin, "An algebraic solution to surface recovery from cross-sectional contours," *Graph. Models Image Process.,* vol. 61, pp. 222-243, 1999.

[125] Y. Shinagawa and T. I. Kunii, "The homotopy model: a generalized model for smooth surface generation from cross sectional data," *Visual Comput.,* vol. 7, pp. 72-86, 1991.

[126] Y. Bresler, J. E. Fessler, and A. Macovski, "A Bayesian approach to reconstruction from incomplete projections of multiple object 3D domain," *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 11, pp. 840-858, 1989.

[127] P. Dierckx, *Curve and Surface Fitting with Splines*. New York: Oxford University Press, 1993.

[128] C. de Boor, *A Practical Guide to Splines*. New York: Springer-Verlag, 2001.

[129] T. K. Dey, "Curve and surface reconstruction," in *Handbook of Discrete and Computational Geometry*, 2nd ed, J. E. Goodman and J. O'Rourke, Eds. Boca Raton, FL: Chapman & Hall, 2004, pp. 677-692.

[130] C. Gustafson, O. Tretiak, L. Bertrand, and J. Nissanov, "Design and implementation of software for assembly and browsing of 3D brain atlases," *Comput. Methods Programs Biomed.,* vol. 74, pp. 53-61, 2004.

[131] T. Ju, "Building a 3D atlas of the mouse brain", Ph.D. dissertation, Dept. Comput. Sci., Rice University, Houston, TX, 2005.

[132] A. W. Toga and P. Thompson, "An introduction to brain warping," in *Brain Warping*, A. W. Toga, Ed. San Diego: Academic Press, 1999, pp. 1-26.

[133] J. P. Carson, C. Thaller, and G. Eichele, "A transcriptome atlas of the mouse brain at cellular resolution," *Curr. Opin. Neurobiol.,* vol. 12, pp. 562-565, 2002.

[134] T. Ju, J. Warren, J. Carson, M. Bello, I. Kakadiaris, W. Chiu, C. Thaller, and G. Eichele, "3D volume reconstruction of a mouse brain from histological sections using warp filtering," *J. Neurosci. Methods,* vol. 156, pp. 84-100, 2006.

[135] R. P. Woods, J. C. Mazziotta, and S. R. Cherry, "MRI-PET registration with automated algorithm," *J. Comput. Assist. Tomogr.,* vol. 1993, pp. 536-546, 1993.

[136] R. P. Woods, S. T. Grafton, C. J. Holmes, S. R. Cherry, and J. C. Mazziotta, "Automated image registration: I. general methods and intrasubject, intramodality validation," *J. Comput. Assist. Tomogr.,* vol. 22, pp. 139-152, 1998.

[137] R. Bajcsy and S. Kovacic, "Multiresolution elastic matching," *Comput. Vision Graph. Image Process.,* vol. 46, pp. 1-21, 1989.

[138] J. C. Gee, M. Reivich, and R. Bajcsy, "Elastically deforming an atlas to match anatomical brain images," *J. Comput. Assist. Tomogr.,* vol. 17, pp. 225-236, 1993.

[139] F. Solina and R. Bajcsy, "Recovery of parametric models from range images: the case for superquadrics with global deformations," *IEEE Trans. Pattern Anal. Mach. Intell.,* vol. 12, pp. 131-147, 1990.

[140] D. Terzopoulos, A. Witkin, and M. Kass, "Energy constraints on deformable models: recovering shape and non-rigid motion," in *Proc. AAAI*, 1987, pp. 755-760.

[141] W. Koh and B. H. McCormick, "Organization and visualization of brain tissue volume data," *Neurocomputing,* vol. 38-40, pp. 1643-1650, 2001.

[142] W. Koh and B. H. McCormick, "Distributed brain microstructure database system: an exoskeleton to a 3D reconstruction and modeling," *Neurocomputing,* vol. 44-46, pp. 1099-1105, 2002.

[143] B. H. McCormick, W. Koh, Y. Choe, L. C. Abbott, J. Keyser, D. M. Mayerich, Z. Melek, and P. Doddapaneni, "Construction of anatomically correct models of mouse brain networks," *Neurocomputing,* vol. 58-60, pp. 379-386, 2004.

[144] M. T. Shipley, M. Ennis, and A. C. Puche, "Olfactory systems," in *The Rat Nervous System*, 3rd ed, G. Paxinos, Ed. New York: Elsevier Science & Technology Books, 2004, pp. 923-964.

[145] A. W. Toga, *Brain Warping*. San Diego: Academic Press, 1999.

[146] G. E. Christensen, R. D. Rabbitt, and M. I. Miller, "Deformable templates using large deformation kinematics," *IEEE Trans. Image Proc.,* vol. 5, pp. 1435-1447, 1996.

[147] H. J. Johnson and G. E. Christensen, "Consistent landmark and intensity-based image registration," *IEEE. Trans. Med. Imaging,* vol. 21, pp. 450-461, 2002.

VITA

Name: Wonryull Koh

Address: Department of Computer Science, c/o Dr. Bruce McCormick, Texas A&M University, College Station, TX 77843-3112

Email Address: wkoh@cs.tamu.edu

Education: B.S., Computer Science and Mathematics, The University of Texas at Austin, 1996
M.S., Computer Science, Texas A&M University, 2000