

NUMERICAL MODELING OF NASAL CAVITIES AND AIR FLOW SIMULATION

Except where reference is made to the work of others, the work described in this dissertation is my own or was done in collaboration with my advisory committee. This dissertation does not include proprietary or classified information.

Kezhou Wang

Certificate of Approval:

Vitaly J. Vodyanoy
Professor
Anatomy, Physiology and Pharmacology

Thomas S. Denney Jr., Chair
Associate Professor
Electrical and Computer Engineering

Edward E. Morrison
Professor
Anatomy, Physiology and Pharmacology

Amnon J. Meir
Professor
Mathematics and Statistics

Stephen L. McFarland
Acting Dean
Graduate School

NUMERICAL MODELING OF NASAL CAVITIES AND AIR FLOW SIMULATION

Kezhou Wang

A Dissertation
Submitted to
the Graduate Faculty of
Auburn University
in Partial Fulfillment of the
Requirements for the
Degree of
Doctor of Philosophy

Auburn, Alabama
May 11, 2006

NUMERICAL MODELING OF NASAL CAVITIES AND AIR FLOW SIMULATION

Kezhou Wang

Permission is granted to Auburn University to make copies of this dissertation at its discretion, upon the request of individuals or institutions and at their expense. The author reserves all publication rights.

Signature of Author

Date

VITA

Kezhou Wang, son of Shixun Wang and Yulan Ma, was born on October 20, 1969, in Wushan, Gansu, China. He received his B.E. degree in Mechanical Engineering from Wuhan University of Technology, Wuhan, Hubei, China in 1992 and M.E. degree in Computational Mechanics from Dalian University of Technology, Dalian, Liaoning, China in 1997. Mr. Wang entered the Ph.D. program in Electrical and Computer Engineering at Auburn University in August 2001. He married Liping Guo, daughter of Zongyang Guo and Shuxian Xu on August 8, 2000, in Beijing, China.

DISSERTATION ABSTRACT

NUMERICAL MODELING OF NASAL CAVITIES AND AIR FLOW SIMULATION

Kezhou Wang

Doctor of Philosophy, May 11, 2006

(M.E., Dalian University of Technology, June, 1997)

(B.E., Wuhan University of Technology, July, 1992)

146 Typed Pages

Directed by Thomas S. Denney Jr.

Computational fluid dynamics (CFD) has many applications in biomedical engineering, such as simulating air dynamics in nasal cavities and lungs, blood flow in vessels, and blood flow in hearts. To perform CFD simulations, numerical models of anatomic structures have to be constructed. The models may be developed from tomographic slices of anatomic structures acquired by medical imaging modalities, such as computed tomography (CT) and magnetic resonance imaging (MRI). However, anatomic structures usually are highly irregular in shape. A mesh with large number of elements is needed to construct an accurate model of an anatomic structure. Manually constructing models would be tedious and error prone. An automatic geometric modeling method is highly desired.

In this dissertation, an automatic numerical modeling technique for nasal cavities and a mathematical model for the shape of the electro-olfactogram (EOG) are developed. Two issues are addressed for numerical nasal cavity modeling. The first issue is that the slice thickness of CT or MRI is usually much larger than the imaging plane resolution, and significant differences are observed between adjacent slices, making it difficult to

construct accurate 3D models directly from acquired image slices. This problem is addressed by introducing a hierarchical spline-based image registration method to perform slice interpolation. The second issue is how to automatically generate 3D finite element CFD mesh from the segmented data. This issue is addressed by the development of an automatic mesh generation algorithm, called marching volume elements (MVE). The algorithm is able to generate three-dimensional (3-D) finite element mesh from volume data. Six human nasal cavity models and a dog model were developed with the numerical modeling technique, and air flow simulations were conducted with the developed models. The mathematical model for modeling the shape of electrical responses of olfactory epithelium to odorant stimuli is a linear input-output model. The model is able to predict the shape of the responses to different odorant concentrations for a fixed duration of stimuli. This model has the potential to evaluate olfactory electrical responses and to estimate kinetics of G-protein cascade within the olfactory receptor neuron.

ACKNOWLEDGMENTS

I would like to thank my Ph. D. advisor, Professor Thomas S. Denney Jr., for his guidance, encouragement, and friendship during the course of my study. He has been a valuable source of ideas and stimulating discussions on both technical and nontechnical topics. I am especially grateful for his efforts on the improvement of my writing and presentation skills. I would like to thank Dr. Denney for providing continuous financial support for my graduate studies. I would also like to express special thanks to the FAA RG and USARMACCN for their generous financial support of this research program.

I would like to thank Professor Vitaly J. Vodyanoy, Professor Edward E. Morrison and Professor Amnon J. Meir for serving on my committee and giving me helpful comments on my dissertation.

I would like to thank my parents Shixun Wang and Yulan Ma, and my parents-in-law Zongyang Guo and Shuxian Xu. I could not have got this far without their great love and support.

Finally I would like to thank my wife Liping Guo, whose love, support and patience have made this work possible. This dissertation is dedicated to her.

Style manual or journal used Journal of Approximation Theory (together with the style known as “aums”). Bibliography follows van Leunen’s *A Handbook for Scholars*.

Computer software used The document preparation package T_EX (specifically L^AT_EX) together with the departmental style-file `aums.sty`.

TABLE OF CONTENTS

| | | |
|-----------------|---|-----|
| LIST OF FIGURES | | xi |
| LIST OF TABLES | | xix |
| 1 | INTRODUCTION AND BACKGROUND | 1 |
| 1.1 | Background Information | 2 |
| 1.2 | Anatomy of Nasal Cavities | 4 |
| 1.2.1 | Anatomy of Human Nasal Cavity | 4 |
| 1.2.2 | Anatomy of Dog Nasal Cavity | 5 |
| 1.3 | Modeling of Nasal Cavity | 7 |
| 1.3.1 | Overview of Human Nasal Cavity Models | 8 |
| 1.3.2 | Geometric Modeling and Mesh Generation | 10 |
| 1.4 | Numerical Simulation of Air flow dynamics in nasal cavities | 11 |
| 1.5 | Olfactory Response | 12 |
| 1.6 | Dissertation Overview and Contributions | 15 |
| 1.6.1 | Slice Interpolation with Hierarchical Spline-Based Registration | 15 |
| 1.6.2 | Finite Element Mesh Generation with Marching Volume Element | 16 |
| 1.6.3 | Air Flow Simulation within Nasal Cavities | 16 |
| 1.6.4 | Olfactory Epithelium Shape Modeling with System Identification | 17 |
| 1.7 | Dissertation Organization | 17 |
| 2 | HIERARCHICAL SPLINE-BASED INTERPOLATION | 19 |
| 2.1 | Review of Interpolation Methods | 19 |
| 2.2 | Hierarchical Spline-Based Interpolation Algorithm | 22 |
| 2.2.1 | General Equations | 23 |
| 2.2.2 | Spline motion model | 25 |
| 2.2.3 | Displacement Field Estimation | 26 |
| 2.2.4 | Hierarchical image structure | 28 |
| 2.2.5 | Implementation considerations | 30 |
| 2.3 | Experimental Results | 32 |
| 2.4 | Conclusion | 39 |
| 3 | MARCHING VOLUME ELEMENT ALGORITHM | 40 |
| 3.1 | Introduction | 40 |
| 3.2 | Marching Volume Elements Algorithm | 43 |
| 3.2.1 | Major Cases of the MVE algorithm | 44 |
| 3.2.2 | Ambiguous Face Problem and Solution | 45 |
| 3.2.3 | Face Tessellation Rules | 51 |

| | | |
|-------|---|-----|
| 3.2.4 | Volume Mesh Generation | 52 |
| 3.3 | Mesh Quality Improvement | 57 |
| 3.4 | Experiments | 59 |
| 3.5 | Conclusion | 63 |
| 4 | NUMERICAL AIRFLOW SIMULATION | 72 |
| 4.1 | Introduction | 72 |
| 4.2 | Simulation Methods | 73 |
| 4.2.1 | Image Acquisition | 74 |
| 4.2.2 | Image Interpolation and Segmentation | 75 |
| 4.2.3 | Geometry Modeling and Mesh Generation | 77 |
| 4.2.4 | Governing Equations and Boundary Conditions | 78 |
| 4.2.5 | Solution Method | 83 |
| 4.3 | Results | 83 |
| 4.3.1 | Air Flow Simulation in Human Nasal Cavities | 84 |
| 4.3.2 | Air Flow Simulation in Dog Nasal Cavity | 92 |
| 4.4 | Discussion | 96 |
| 5 | MODELING ELECTRO-OLFACTOGRAM RESPONSE SHAPES USING SYSTEM IDENTIFICATION TECHNIQUES | 102 |
| 5.1 | Experiments and Data Collection | 102 |
| 5.2 | System Identification | 104 |
| 5.3 | Fitting Eog Shape with OE Model | 105 |
| 5.4 | Experimental Results | 106 |
| 5.5 | Discussion | 109 |
| 6 | CONCLUSION | 112 |
| 6.1 | Hierarchical Spline-Based Interpolation | 112 |
| 6.1.1 | Main Results | 113 |
| 6.1.2 | Future Work | 113 |
| 6.2 | Marching Volume Element Algorithm | 114 |
| 6.2.1 | Main Results | 114 |
| 6.2.2 | Future Work | 115 |
| 6.3 | Numerical Airflow Simulation | 116 |
| 6.3.1 | Main Results | 116 |
| 6.3.2 | Future Work | 117 |
| 6.4 | Modeling Electro-Olfactogram Response Shapes Using System Identification Techniques | 117 |
| 6.4.1 | Main Results | 118 |
| 6.4.2 | Future Work | 119 |
| | BIBLIOGRAPHY | 120 |

LIST OF FIGURES

| | | |
|------|---|----|
| 1.1 | Human nasal cavity CT slice and the segmented right cavity | 5 |
| 1.2 | A cropped CT slice of dog nasal cavities | 7 |
| 1.3 | Signal transduction in olfactory sensory neurons (reprinted from [25]). . . | 14 |
| 2.1 | 3D surface of a human left cavity constructed from 31 CT slices without slice interpolation. (a) left side of view, (b) right side of view. | 20 |
| 2.2 | Artificial square blocks interpolated with the linear interpolation algorithm. 22 | 22 |
| 2.3 | An ideal slice interpolation of artificial square blocks. | 22 |
| 2.4 | Artificial square blocks interpolated with a shape-based interpolation algorithm. | 22 |
| 2.5 | Artificial square blocks interpolated with the adaptive control grid interpolation (ACGI) algorithm. | 23 |
| 2.6 | Displacement field for square images. (a) The target image. (b) The estimated displacement field. (c) The source image | 23 |
| 2.7 | Displacements on coarser grid are calculated. | 25 |
| 2.8 | The structure of Gaussian pyramid. | 29 |
| 2.9 | An image pyramid of a human nose CT slice. | 30 |
| 2.10 | Normalized execution time and mean square error vs. block-size for the HSR algorithm. The computation time normalized by the maximum computation time. The mean square error is normalized by the maximum error. The results are based on two cut CT the images with a size of 257×257 | 31 |
| 2.11 | Binary square images are interpolated with different methods. The images on the most left side and the most right side are the reference images. The other images are created at 1/3,1/2 and 2/3 distance from the reference images respectively by different methods. (a) linear interpolation. (b) shape-based interpolation. (c) ACGI interpolation. (d) HSR interpolation. 33 | 33 |

| | | |
|------|---|----|
| 2.12 | Displacement field for square images. (a) The target image. (b) The estimated displacement field by the ACGI algorithm. (c) The source image | 34 |
| 2.13 | Displacement field for square images. (a) The target image. (b) The estimated displacement field estimations by the HSR algorithm. | 34 |
| 2.14 | Mean squared differences vs. iteration times for the HSR algorithm and the ACGI algorithm. 4×4 block-sizes are used for each case. For the HSR algorithm, three level pyramids are used, where level 1 is the finest, and level 3 is the coarsest. | 34 |
| 2.15 | Linear interpolation on human head CT slices. (a) and (c) are two adjacent human head CT slices. (b) is the intermediate slices at half distance by the linear interpolation. | 35 |
| 2.16 | Shape-based interpolation on human head CT slices. (a) and (c) are segmented images. (b) is the intermediate slice at half distance by shape-based interpolation. | 36 |
| 2.17 | ACGI interpolation on human head CT slices. (a) and (c) are two adjacent human head CT slices. (b) is the intermediate slices at half distance by the ACGI interpolation. | 36 |
| 2.18 | HSR interpolation on CT slices. (a) and (c) are two adjacent human head CT slices. (b) is the intermediate slices at half distance by the HSR interpolation. | 36 |
| 2.19 | Mean Squared Differences vs. Iteration for the HSR algorithm and the ACGI algorithm with human nose CT slices. A block size of 12×12 is used in the calculations. The time cost is also listed. The algorithms are run on a PC which has a Intel Pentium III 549MHz processor and 1.0 GB of RAM. For the HSR algorithm, six level pyramids are used, where level 1 is the finest, and level 6 is the coarsest. | 37 |
| 2.20 | The HSR algorithm is used to construct 3D surface of a human cavity with 31 CT slices. 5 intermediate slices are interpolated between each two adjacent CT slices. (a) left side of view, (b) right side of view. | 38 |
| 2.21 | The ACGI algorithm is used to construct 3D surface of a human left cavity from 31 CT slices. 5 intermediate slices are interplated between each two adjacent CT slices. (a) left side of view, (b) right side of view. | 38 |
| 3.1 | Two ways to tessellate a cube into 5 tetrahedra. | 42 |

| | | |
|-----|--|----|
| 3.2 | 21 major cases of the marching volume element algorithm. Letter C denotes a complementary case. For example, Case 1 and Case 1C are complementary cases of each other. One type of triangularization (thin lines) is shown for each case. | 45 |
| 3.3 | An ambiguous face with two black diagonal vertices and two white diagonal vertices. (a) Two white vertices are separated. (b) Two white vertices are connected. | 46 |
| 3.4 | Example of discontinuous surface created by the MC algorithm. Cube (a) and (b) share a common face. The connection of the common face of cube (a) and (b) are different. The two white vertices on the common face of cube (a) are separated. The two white vertices on the common face of the cube (b) are connected. This is the fundamental reason of the discontinuous surface problem of the MV algorithm. | 46 |
| 3.5 | Two subcases for Case 3 (3A and 3B) in Figure 3.2. | 47 |
| 3.6 | Subcases for the Case 10 in Figure 3.2. (a) white vertices on two ambiguous faces are separated. (b) white vertices on one ambiguous face are separated (bottom face), while white vertices on the other ambiguous face are connected (top face). (c) white vertices on one ambiguous face are separated (top face), while the white vertices on the other ambiguous face are connected (bottom face). (d) white vertices on two ambiguous faces are connected. Note (b) and (c) are equivalent by rotation. | 48 |

| | | |
|------|---|----|
| 3.7 | Subcases for the Case 13 in Figure 3.2. (a) White vertices on all ambiguous faces are connected. (b) White vertices on five ambiguous faces are connected, while white vertices on one ambiguous face are separated (front face). (c) White vertices on two parallel ambiguous faces are separated (front and back faces), while the white vertices on the other ambiguous faces are connected. (d) White vertices on two adjacent ambiguous faces are separated (front and right faces), while the white vertices on the other ambiguous faces are connected. (e) White vertices on three adjacent ambiguous faces are separated (front, right and left faces, notice the three faces share two edges), while the white vertices on the other ambiguous faces are connected. (f) White vertices on three adjacent ambiguous faces are separated (front, right and bottom faces, notice the three faces share three edges), while the white vertices on the other ambiguous faces are connected. (g) White vertices on three adjacent ambiguous faces are separated (front, top and bottom faces, notice the three faces share two edges), while the white vertices on the other ambiguous faces are connected. (h) White vertices on two parallel ambiguous faces are connected (front and back faces), while the white vertices on the other four ambiguous faces are separated. (i) White vertices on two adjacent ambiguous faces are connected (front and right faces), while the white vertices on the other four ambiguous faces are separated. (j) White vertices on one ambiguous face are connected (front face), while white vertices on other ambiguous faces are separated. (k) White vertices on all ambiguous faces are separated. | 49 |
| 3.8 | Bilinear interpolation. | 50 |
| 3.9 | Tessellation rules for different cube face configurations. | 52 |
| 3.10 | Marching volume element notation. Numbers 1-8 represent eight cube vertices. Letter O represents the center point of the cube. Letters A-L represent isopoints on edges. The centers of faces are represented by letters P-U. | 53 |
| 3.11 | Volume mesh decomposition for Case 1C. | 54 |
| 3.12 | Volume mesh decomposition for Case 2. | 54 |
| 3.13 | Volume mesh decomposition for Case 2C. | 54 |
| 3.14 | A 2-D example of mesh quality improvement: (a) before improvement. (b) after improvement. | 58 |
| 3.15 | A 3-D surface mesh of a mortar generated by MVE from a 16x16x8 pixel simulated CT data set. | 60 |

| | | |
|------|---|----|
| 3.16 | Mesh computation time versus resolution for both MVE (line only) and GAMBIT (line and circle). MVE algorithm is able to mesh all data sets, while GAMBIT can not mesh the two higher resolution data sets. | 60 |
| 3.17 | Edge ratio histograms for the 128x128x64 mortar mesh constructed by (a) MVE before improvement (b) MVE after improvement (c) GAMBIT. The MVE mesh contained 887,504 elements before improvement, 884,624 elements after improvement, and the GAMBIT mesh contained 3,134,488 elements. | 62 |
| 3.18 | Equi-angle skew histograms for the 128x128x64 mortar mesh constructed by (a) MVE before improvement (b) MVE after improvement (c) GAMBIT. Note the vertical axes of plots (a) and (b) have been cropped to improve the detail in the higher equi-angle skew values. The bar at zero equi-angle skew extends to 50% in plot (a) and (b). The MVE mesh contained 887,504 elements before improvement, 884,624 elements after improvement, and the GAMBIT mesh contained 3,134,488 elements. . . . | 63 |
| 3.19 | Right nasal cavity surface constructed with the MVE algorithm. The marked region is mainly contains olfactory epithelium. | 64 |
| 3.20 | Edge ratio histograms for the nasal cavity model constructed with MVE before (a) and after (b) improvement. | 64 |
| 3.21 | Equi-angle skew histograms for the nasal cavity model constructed with MVE before (a) and after (b) improvement. | 64 |
| 4.1 | A cropped CT slice of human nasal cavities | 73 |
| 4.2 | A cropped CT slice of dog nasal cavities | 74 |
| 4.3 | Six CT slices of the dog scans | 79 |
| 4.4 | Dog nasal cavity surface reconstructed from 70 CT slices. Underneath the surface is the volume mesh. Red colored region are the sinues, magenta colored regions are the nasal cavity airways, and the blue colored region is the olfactory region. | 80 |
| 4.5 | Right nasal cavity surface of study 4. The blue region is mainly covered with olfactory epithelium. The numbered lines indicate the position of planes where simulation results are displayed. | 81 |

| | | |
|------|---|----|
| 4.6 | Secondary velocity vector and axial velocity contours on cross-section planes 1. The Maximum velocity is 4.865 m/s. The maximum axial velocity is 4.805 m/s. | 85 |
| 4.7 | Secondary velocity vector and axial velocity contours on cross-section planes 2. The Maximum velocity on this plane is 2.865 m/s. The maximum axial velocity is 2.785 m/s. | 86 |
| 4.8 | Secondary velocity vector and axial velocity contours for plane 3. The Maximum velocity on this plane is 2.676 m/s. The maximum axial velocity is 2.675 m/s. | 86 |
| 4.9 | Secondary velocity vector and axial velocity contours for plane 4. The Maximum velocity on this plane is 2.529 m/s. The maximum axial velocity is 2.506 m/s. | 86 |
| 4.10 | Secondary velocity vector and axial velocity contours for plane 5. The Maximum velocity on this plane is 2.194 m/s. The maximum axial velocity is 2.178 m/s. | 87 |
| 4.11 | Secondary velocity vector and axial velocity contours for plane 6. The Maximum velocity on this plane is 1.966 m/s. The maximum axial velocity is 1.943 m/s. | 87 |
| 4.12 | Secondary velocity vectors at the plane 5 in Figure 4.5 for each of the six human studies | 88 |
| 4.13 | Velocity magnitude iso-contours at plane 5 in Figure 4.5 for each of the six human studies | 89 |
| 4.14 | Path routes simulation for group 1 particles for study 4. The particles were released about 1 mm away from the ventral tip of the naris wall. . . | 93 |
| 4.15 | Group 1 particle passing positions on cross-sectional planes 1 to 6 as shown in Figure 4.5. The particles were released about 1 mm away from the ventral tip of the naris wall. | 93 |
| 4.16 | path routes simulation for group 2 particles. The Particles were released at middle line of the external naris. | 94 |
| 4.17 | Group 2 particle passing positions on cross-sectional planes 1 to 6 as shown in Figure 4.5. The Particles were released at middle line of the external naris. | 94 |

| | | |
|------|--|-----|
| 4.18 | Path routes simulation for group 3 particles. The particles were released about 1 mm away from the dorsal tip of the naris wall. | 95 |
| 4.19 | Group 3 particle passing positions on cross-sectional planes 1 to 6 as shown in Figure 4.5. The particles were released about 1 mm away from the dorsal tip of the naris wall. | 95 |
| 4.20 | The velocity vectors at the No. 15 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 15 CT slice of the dog(counted from the naris to the Nasopharyngeal) | 96 |
| 4.21 | The velocity vectors at the No. 24 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 24 CT slice of the dog(counted from the naris to the Nasopharyngeal) | 97 |
| 4.22 | The velocity vectors at the No. 32 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 32 CT slice of the dog(counted from the naris to the Nasopharyngeal) | 97 |
| 4.23 | The velocity vectors at the No. 40 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 40 CT slice of the dog(counted from the naris to the Nasopharyngeal) | 98 |
| 4.24 | The velocity vectors at the No. 48 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 48 CT slice of the dog(counted from the naris to the Nasopharyngeal) | 98 |
| 4.25 | The velocity vectors at the No. 53 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 53 CT slice of the dog(counted from the naris to the Nasopharyngeal) | 99 |
| 4.26 | The velocity vectors at the No. 66 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 66 CT slice of the dog(counted from the naris to the Nasopharyngeal) | 99 |
| 4.27 | The velocity vectors at the No. 74 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 74 CT slice of the dog(counted from the naris to the Nasopharyngeal) | 100 |
| 5.1 | System identification input-output configuration | 105 |

| | | |
|-----|---|-----|
| 5.2 | Normalized measured (solid line) and normalized simulated (dashed line) responses from a rat single-cell (a), rat EOG (b), and dog EOG (c) to a 16mM stimulus. The measured responses are the validation data. The arrow denotes the start of the stimulus. The stimulation duration was (a) 500ms, (b) 200ms, (c) 100ms. | 108 |
| 5.3 | Normalized measured (solid line) and normalized simulated (dashed line) responses: (a) rat EOG response to a 200ms, 1mM stimulus; (b) rat EOG response to a 200ms, 4mM stimulus; (c) dog EOG response to a 500ms, 16mM stimulus. The arrow denotes the start of the stimulus. | 108 |

LIST OF TABLES

| | | |
|-----|--|-----|
| 3.1 | Subcases of Ambiguous Cases | 47 |
| 3.2 | Mesh Table for Marching Volume Elements Algorithm | 66 |
| 3.3 | Cube Rotation Table | 68 |
| 4.1 | Resolution of CT Scans and Slice Numbers | 75 |
| 4.2 | The Number of Elements for Each Model for Human Studies 1-6 and the Dog Study | 78 |
| 4.3 | The area and the perimeter on cross-section planes and positions of the planes relative to the naris | 85 |
| 5.1 | Coefficients for the rat single-cell, rat EOG, and dog EOG models. Blank entries denote a coefficient of zero. The a_j coefficients have no units. The rat single-cell b_j coefficients have units of mM^{-1} . Rat and dog EOG b_j coefficients have units of μM^{-1} | 108 |

CHAPTER 1

INTRODUCTION AND BACKGROUND

Knowledge of the airflow field in nasal cavities is essential to understand the basic functions of the nose, such as air transportation and odorant sensation. With the advanced medical imaging modalities, such as magnetic resonance imaging (MRI) and computed tomography, detailed anatomic structure information may be obtained non-invasively, making it possible to develop geometrical and numerical models of anatomic structures for computational fluid dynamics (CFD) simulations. Two issues have to be addressed in the construction of 3D geometric models and computational meshes for CFD. The first issue is that the slice thickness is usually much larger than the imaging plane resolution. Large differences are observed between adjacent slices, making it difficult to construct accurate 3D models directly from image slices. The second issue is how to automatically generate 3D finite element CFD mesh from the segmented data. Biological structures often have highly irregular shapes and the mesh can contain a large number of elements. Manual mesh generation is tedious and error prone. This dissertation will address these two issues. The first issue is solved by introducing a hierarchical spline-based image registration method to perform interpolation between slices of large thickness, which is presented in detail in Chapter 2. The second issue is addressed by development of an automatic mesh generation algorithm, called marching volume elements (MVE). The algorithm is able to generate three-dimensional (3-D) finite element mesh from volume data. The MVE algorithm is presented in Chapter 3. With the image interpolation scheme and the MVE algorithm, six 3D geometric models of human and

one dog nasal cavity models were constructed from CT scans, and steady state air flow simulations were conducted on those models with FiDAP (Fluent, Inc. Lebanon, New Hampshire). The simulation results are presented in Chapter 4.

In addition, a mathematical model is proposed for the shape of electrical responses of olfactory epithelium or olfactory neurons to odorant stimuli in Chapter 5 in this dissertation. The model may be used to predict the shapes of the responses to different odorant concentrations for a fixed duration of stimuli.

The main contributions of this dissertation are the development of the 3-D automatic mesh generation and the slice interpolation algorithm for geometrical modeling of anatomic structures for medical image data, the simulation of air flow in human and dog nasal cavities, and the EOG shape modeling of the olfactory epithelium.

In this chapter, a brief overview of background information, nasal anatomy, modeling of anatomic structures and review of related work are presented.

1.1 Background Information

Noses are very important organs for human and animals. It is the front part of the respiratory system which has many important physiological functions. It provides a channel for respiratory system inspiration of fresh ambient air and expiration of the deoxygenated air. Through the nose, the dust, bacteria and other environmental pollutants are filtered before entering the lower respiratory tract, and the inspired air is also warmed and humidified before reaching lungs to prevent damage [71].

Although the nose plays vital physiological functions, some pathological conditions may develop. The filtration of environmental pollutants can cause infections and other lesions in the nasal mucosa. The distribution of these lesions has been shown to relate to

airflow patterns that determine regional uptake of the pollutions on airway walls [44, 45]. In addition, various anatomical deformaties, such as a deviated septum and nasal polyps, can obstruct the nasal airway and increase the resistance to flow [46]. Nasal obstructions and pattern of airflow have also been shown to affect olfactory perception [53].

The knowledge of air flow in nasal cavity facilitates understanding of the structure and function relationship of the nose. The air flow in nasal cavity may be simulated with computational fluid dynamics (CFD). To study air flow in nasal cavities with CFD, geometric models of the anatomic structures must be constructed. Accurate three-dimensional (3D) geometric models may be constructed from image slices acquired from tomographic imaging modalities, such as computed tomography (CT) or magnetic resonance imaging (MRI). There are three basic steps in the development of a geometric model of anatomic structure:

1. Segmentation the airway channels from the medical image.
2. Extraction of the surface of nasal cavity from the segmented data.
3. Mesh generation.

To simulate air flow with CFD, a computational domain needs to be descritized into small elements, called mesh. Since the anatomic structures usually are complex and highly irregular, large number of elements are required to present the anatomic structure precisely. The problem is how to convert a stack of image slices into a computational mesh. This is a bottleneck in the application of computational methods to biological structures. An automatic 3-D mesh geneartion algorithm, which can create 3-D finite element mesh from tomography data, is highly desired.

1.2 Anatomy of Nasal Cavities

In general, the nose of mammal is separated medially by the nasal septum into two cavities of equal size. Each half of the nasal cavity can be divided into three regions: vestibule, respiratory and olfactory regions. Although the functionalities of the noses are about the same among mammals, different terminologies are used for human and animals. In this section, the nomenclatures of human and dog nasal cavity anatomy are presented, and will be used throughout this dissertation.

1.2.1 Anatomy of Human Nasal Cavity

The nomenclature of the human nasal anatomy used in this dissertation is the same as Proctor [71]. The human nasal cavity extends from the nostrils to the turbinates. The cavity is divided into right and left halves by the nasal septum. Each half has a roof, a floor, medial and lateral walls. The roof is formed by nasal cartilages, nasal and frontal bones, cribriform plate of the ethmoid, and body of the sphenoid. The floor is formed by the palatine process of the maxilla and the horizontal plate of the palatine bone. The medial wall is the nasal septum. the lateral wall presents three medial projections termed nasal chonchae, formerly known as the inferior, middle and superior turbinates. The nasal cavity can be divided into three portions: vestibule, respiratory and olfactory regions. The vestibule region is a funnel shaped dilated region. The funnel leads to a region referred to as the nasal valve, where the airway is shaped like a narrow slit. The posterior part of the vestibule surface is covered by mucosa. Beyond the vestibule is the respiratory region which comprises most of the nasal cavity, which is covered by pseudostratified ciliated columnar epithelium. The main airway in this region is divided

into different parts by the turbinates. The airway channels under the turbinates are named after the turbinates above them, which are the inferior, middle, and superior meatuses respectively shown in Figure 1.1. The slit-shaped region in the superior part between the nasal septum and the lateral wall of the main nasal passage is the olfactory airway, where the surface is covered with olfactory epithelium.

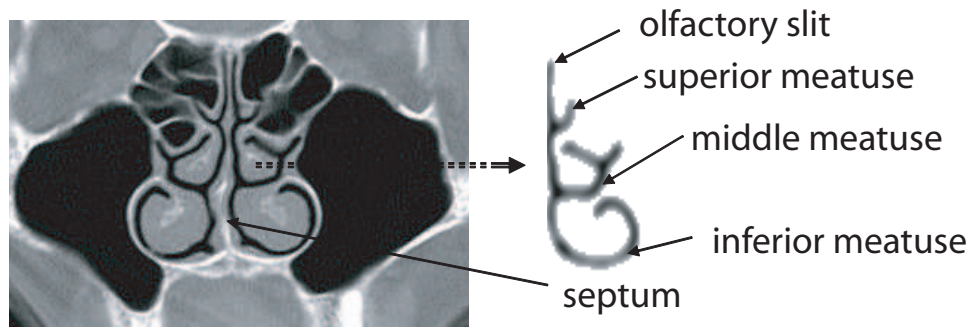


Figure 1.1: Human nasal cavity CT slice and the segmented right cavity

1.2.2 Anatomy of Dog Nasal Cavity

The nomenclature of the dog nasal anatomy used in this dissertation references to Evans [40]. The dog nasal cavity is significantly different from the human. Firstly, nasal vestibule of dogs is not empty, as it is in man. It is largely obliterated by the large bulbous end of the alar fold. Secondly, the major portion of each half of the nasal cavity is occupied by the nasal turbinates, called dorsal, ventral and ethmoidal turbinates, and the most obvious difference is the ventral nasal turbinate which is a tightly folded series of scrolls, as shown in Figure 1.2. Finally, unlike humans, dogs have ethmoturbinates with both scrolled main portion and a number of smaller pockets and sinuses with narrow

openings to the main portion, and the ethmoturbinates fill the caudal part of the nasal cavity. The thin membranous nature of the turbinate region makes it more effective than that of the human in removing many of the particles that penetrate the naris. In addition, the large surface area of the turbinates of the dog should be more effective in absorbing soluble gases than the relative simple human turbinates. Due to presentation of the delicate scrolled turbinates, the respiratory and olfactory region of dog nasal cavity is much more complex than human. The passage airway between the dorsal nasal turbinate and the ventral surface of the nasal bone is called the dorsal nasal meatus. The dorsal nasal meatus confluent with the common nasal meatus. Similar to human, the middle nasal meatus lies between the dorsal nasal turbinate and the dorsal part of the ventral nasal turbinate. The middle portion of the middle meatus is about 1mm wide. It dilates at its rostral end, and laterally the caudal part of the middle nasal meatus is divided by the scrolls of the ethmoid turbinate into several air passages that lie between these scrolls. The ventral nasal meatus is located between the ventral nasal turbinate and the dorsal surface of the hard maxilla palate. It gradually widens from the nasal vestibule, and attains a width of about 1 *cm* at the large nasomaxillary opening. The ventral meatus continues ventral to the floor plate of the ethmoid bone as the nasopharyngeal meatus, where the middle, ventral, and common nasal meatuses converge. The common nasal meatus is a longitudinal narrow slit on either side of the nasal septum. Laterally, it is bounded by the ventral nasal turbinate and the nasal bone. The olfactory region is located primarily on the ethmoturbinates, the caudal half of the nasal septum, and a good portion of the roof and lateral walls of the nasal cavity. The surface of the olfactory region is covered by the sensory olfactory epithelium. The olfactory receptor neurons are located in the olfactory epithelium. Axons from these olfactory receptor cells enter

the skull through the cribriform plate of the ethmoid bone to reach the olfactory bulbs of the brain where they form first order synapse [40].

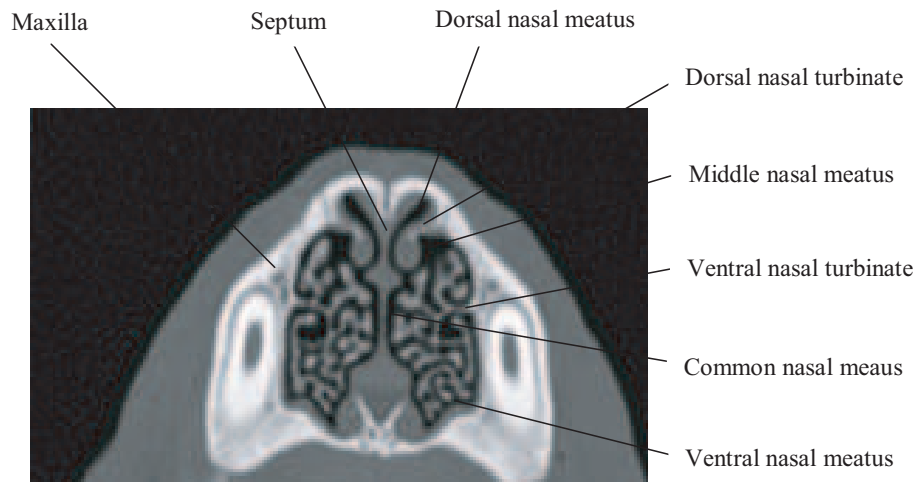


Figure 1.2: A cropped CT slice of dog nasal cavities

1.3 Modeling of Nasal Cavity

To study air flow in nasal cavities, geometric models of anatomic structures have to be developed. Many experimental models and numerical models of human nasal cavities have been developed to study air flow. But to our knowledge, the study of air flow in the dog nasal cavity has not been reported.

1.3.1 Overview of Human Nasal Cavity Models

Experimental Models of Human Nasal Cavity

Previously researches used human cadavers to construct their models. For visual observation, smoke, dyed water or xenon were used as the flow medium, and the non-planar septum was replaced by a clear flat plastic plate [59, 72, 73, 87, 29, 39]. Swift and Proctor used a pitot device which was introduced into the flow through the nasal septum to measure the velocity field [87]. Girardin et al. model measured the velocity field with a laser Doppler velocimetry [29]. Hornung et al. used xenon to image the airflow pattern in their model [39]. The radioactive gas was infused at different sites in the nostril, and the distribution of radioactivity was imaged in the sagittal plane with a scintillation camera.

With medical imaging technology, researchers developed their models non-invasively based on tomographic slices of anatomic structure scanned with modalities, such as computed tomography (CT) or magnetic resonance imaging (MRI). Although anatomically accurate physical models can be reconstructed from the image slices, it is extremely difficult to obtain detailed flow field due to the small and complicated nasal structures. The resolution of the measurements was poor because of the small size of the cast model. To overcome these difficulties, scaled-up physical models were developed [79, 31, 32]. Schreck et al. studied airflow with a three times enlarged plastic model of a half-nasal cavity based on MRI data [79]. Dyed water was used as the flow medium. Hahn et al. used a 20 times enlarged nasal cavity model constructed from CT scans to study airflow patterns [31, 32]. Air velocities were measured with hot-filament anemometer probes which were inserted through measurement holes on the experiment model.

Although experimental models can be used to simulate air flow, it is difficult to set boundary conditions. Moreover, development of an experimental model is time consuming, and it is hard to be modified once it is constructed. Numerical models can be used to overcome these drawbacks.

Numerical Models of Human Nasal Cavity

Numerical methods have many advantages over physical models. First, complete fluid field information can be obtained with numerical models, which is impractical in physical models. Second, it is easier to make modifications with the numerical model. With this feature, it is possible to study the air flow changes in the cavities with the change of anatomic structures. The results can be used by physicians to assess or develop optimal surgical plans. Finally, it is more efficient to develop a numerical model than a physical model if an automatic modeling method is available.

Computational fluid dynamics (CFD) has proven to be an attractive method to solve complex fluid flow problems. With computers becoming more powerful and less expensive, simulation of fluid flow with numerical methods become affordable and desirable. Recently, numerical models were developed to study the fluid flow in the nasal cavities [31, 43, 96, 14]. A 2-D steady laminar flow in the nasal valve was simulated in [96]. Elad et al. simulated steady laminar flow with a 3-D simplified model [14]. Using CT data and graph paper, Keyhani et al. developed a 3-D numerical model of human nasal cavity and steady state air flow simulations were conducted [43].

All the models are either simplified or manually constructed. But human nasal cavities are very complex and highly irregular. It is very difficult to simplify or idealize

the 3-D complex structure. Manual construction of 3-D nasal cavity model would be tedious and error prone. An automatic model reconstruction method is highly desired.

1.3.2 Geometric Modeling and Mesh Generation

To conduct numerical simulations of air flow in nasal cavities, Geometric models of anatomic structures have to be developed. Anatomic structures, which are often imaged with computed tomography (CT) or magnetic resonance imaging (MRI), are needed to convert into geometric models and computational meshes. The conversion algorithms are usually referred to as mesh generation algorithms. Biological structures often have highly irregular shapes. To represent the nasal cavity structure precisely, a large number of elements are required. For these reasons an automatic algorithm is needed for constructing a volume mesh from a CT imaging scan of a biological structure. Since usually large data sets are needed for processing, the algorithm should be memory efficient as well.

Many meshing algorithms have been developed over the years, such as Octree techniques, Delaunay-based techniques and advancing front methods [80, 101, 58, 100, 13, 26, 60, 81, 55, 56, 75, 11, 41]. In octree techniques [80, 101, 58], cubes containing geometric models are recursively subdivided until desired resolution is reached. The problem with this technique is that element sizes of adjacent cubes may change dramatically, making it difficult to ensure topological consistency. Delaunay-based techniques [100, 13, 26, 60, 81] are based on the Delaunay criterion, which states that any node of an element must not be contained within the circumspheres of any other elements. This criterion connects a set of existing points in space. The main drawback of Delaunay techniques is that objects must be convex and have no interior holes. This usually can not be guaranteed

for biological structures. If the objects are concave or have holes, the concave parts will be eliminated and holes will be meshed as objects. Therefore, the mesh created by Delaunay techniques may not satisfy the geometric consistency requirement. In the advancing front method [55, 56, 75, 11, 41], tetrahedra are built progressively inward from a triangulated surface. An active front is maintained at where new tetrahedra are formed. For each triangular facet on the front, an ideal location for a new fourth node is computed. The algorithm selects either the new fourth node or an existing node to form the new tetrahedron. The algorithm can start from an arbitrary surface. One disadvantage of the advancing front method is that it is computationally extensive because it must continuously track and calculate the new front. Another drawback is that it does not take advantage of the structure of gridded data. In addition, the Delaunay-based and advancing front based algorithm require surfaces of the objects provided. For anatomic structure represented with tomographic slices, the surface extraction itself is a difficult task.

1.4 Numerical Simulation of Air flow dynamics in nasal cavities

Simulations of air dynamics in nasal cavities are studied in this research. The air flow in nasal cavity may be simulated with Computational fluid dynamics (CFD) using numerical models. There are many kinds of fluids flow in biological bodies, such as blood and air flow in hearts and lungs. Biological bodies depend on these fluids to fulfill important physiological functions. In vivo measurement of these fluid flow is very difficult if it is not impossible. CFD can play a very important role in simulation of these fluid flows.

The knowledge of air flow in the nasal cavity may help us understand the underlying biological functions. By developing nasal cavity models and conducting air flow simulations, we can obtain air flow field information and patterns in nasal cavity. The information of air flow in nasal cavity may have lots of applications. It can be used to study how the odorants are transported to the olfactory epithelium and how the air flow pattern affects the olfaction. In addition, to relieve nasal blockage, surgery is often performed without quantitative information about the airflow pattern. With numerical simulation methods, an optimal surgery plan may be conducted, and it is possible to estimate the surgery results by performing simulations after a surgery.

1.5 Olfactory Response

The nose is not only an important component of respiratory system, but also where the olfactory system resides in. Human beings and animals rely on olfactory systems to smell. Human beings view smell as an aesthetic sense, yet for most animals smell is the primary sense to survive. Many animals depend on olfactory system to identify food, predators and mates. Each individual has a unique, genetically determined scent. Humans can recognize approximately 10,000 odorant chemicals, ranging from pleasant scent of freshly cut flowers to aversive smell of an angry skunk. Many animals have an even greater sensitivity to odors than human beings do. For example, blood-hounds are legendary for their extraordinary ability to discriminate scents [77].

The olfactory region in the nasal cavity is mainly covered by the mammalian main olfactory epithelium (MOE). Inspired air carries odorant molecules to the olfactory receptor neuron dendritic knobs, which are exposed to the environment on the MOE surface,

where odorants bind G-protein-linked transmembrane receptors. This binding initiates the intracellular signal cascade that results in depolarization and action potential.

Ottoson[66] developed the electro-olfactogram (EOG) to measure early responses to odorants in tissue fragments and discovered the slow potential, which is a gradual change in the potential difference, typically negative, across the MOE surface relative to the recording and reference electrodes. Among the related phenomena, Takagi et al. described positive transient changes to either and chloroform [35, 82, 92, 93, 94, 95]. Gesteland et al. reported a purely positive transient with methanol, but in the same study, also described the interactions of two slow potentials generated after the tissue is exposed to a succession of two odorants [27]. The latter observation indicates that a given sensory neuron, which expresses a single odorant receptor type, responds differently to different odorants, and that an odorant, if it has any effect on an olfactory receptor neuron, can either excite or inhibit that cell. Therefore, the slow potentials measured by EOG are the sum of individual sensory cell responses and are composed of at least two opposing processes: one that drives the potential in the positive direction and one that drives it in the negative direction. These two antagonistic processes are known to be based on so called G-protein cascade, the cascade of internal enzymes that control ion currents of olfactory neurons [51, 17].

Fig.1.3 shows how the chemical binding signal is converted into electrical signal. When odorant molecules are bound on the olfactory receptors (G-protein-coupled receptors, a cascade of events happen. the receptor activates a G-protein(Golf), which in turn activates an AC III type adenylyl cyclases, so called the second messenger. The cyclase converts the abundant intracellular molecule ATP into cyclic AMP (cAMP). The cAMP binds to the intracellular face of a cyclic nucleotide-gated (CNG) channel which

is an ion channel, enabling the channel to conduct cations such as Na^+ (sodium) and Ca^{2+} (calcium). If enough channels are open for enough time, causing the membrane potential to become less negative from the resting membrane voltage, the cell reaches threshold and generates an action potential, which is an electrical signal. The action potential is then propagated along the axon, into the olfactory bulb.

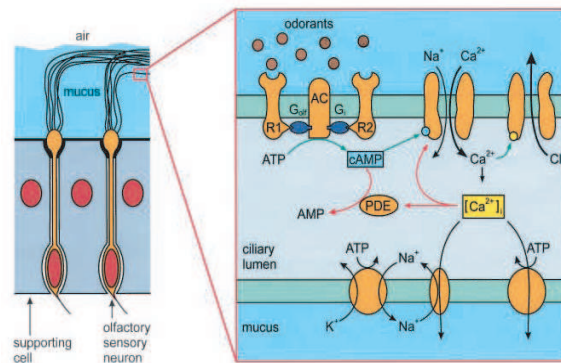


Figure 1.3: Signal transduction in olfactory sensory neurons (reprinted from [25]).

A number of theoretical models for olfaction have been proposed. Nachber and Morton [61] constructed a theoretical model that describes time response of surface concentrations of odorant to various step function presentations, adaptation, and dose-response functions that can be found in olfactory experiments. Hahn [30] proposed a theoretical model of olfaction involving many of the major mechanisms in a mass transport of odorant molecules from inspired air to the olfactory receptors. This model involves the major mechanisms that are found in the odorant transportation. The mechanisms include bulk flow, lateral transport, sorption, diffusion of odorant, and interaction of odorant molecules with the olfactory receptors. The model determined that flow rate of the odorant, length of the olfactory mucus surface, and solubility of odorant molecules play important roles in determination of the odor intensity. This model is consistent with experimental results

of Hornung and Mozel[38]. These models share, to varying degrees, difficulty in application because several parameters and rate constants required by the kinetic equations remain unknown.

1.6 Dissertation Overview and Contributions

In this research, we develop a new technique for construction of numerical models of nasal cavities. Numerical models of human and canine nasal cavities are developed with this technique, and the models are used to conduct air flow simulations with computational fluid dynamics software. The model construction technique includes two parts: the first part is a slice interpolation algorithm called hierarchical spline-based registration which is able to deal with large dissimilarities between adjacent image slices; the second part is a 3-D automatic mesh generation algorithm called marching volume elements, which is able to generate geometric models and computational mesh from 3-D volume data set. In addition, a mathematical model of the EOG shape is proposed.

1.6.1 Slice Interpolation with Hierarchical Spline-Based Registration

The first contribution is that a hierarchical spline-based registration algorithm is proposed to perform slice interpolation in the construction of nasal cavity models. This interpolation algorithm is a combination of optical flow technique and block matching method. The algorithm works in a coarse-to-fine fashion on hierarchical image pyramids which are constructed by recursive decimation of original images. The algorithm computes a displacement field which can warp one image into another image, and the intermediate slices can be obtained by interpolating the computed displacement field. The algorithm has the best performance compared to the other slice interpolation techniques

tested. Satisfying nasal cavity models have been developed based on this interpolation technique. Part of this work has been published in [98].

1.6.2 Finite Element Mesh Generation with Marching Volume Element

The second contribution is the development of a new 3-D mesh generation algorithm, called marching volume elements (MVE). This algorithm is able to extract surface information and create finite element mesh from 3-D volume data defined on a 3-D regular grid. The algorithm decomposes the 3-D volume domain into small cubes of the same size first, then the surface triangles and volume mesh for each cube are found by looking up a predefined look-up table. The algorithm works on the cubes one by one. Marching through all the cubes in the volume data will generate desired finite element mesh. The algorithm has been successfully used to develop numerical models of nasal cavities from CT data. Part of this work has been published [99, 97].

1.6.3 Air Flow Simulation within Nasal Cavities

The third contribution is the air flow simulations with nasal cavity models generated by the MVE algorithm. The MVE algorithm was validated on human studies first. After validation on human subjects, a dog nasal cavity model was constructed with the MVE algorithm, and air flow was simulated with the model. Six human and one dog nasal cavity models were developed from CT scans of human and dog heads. Steady state air flow is simulated with FiDAP (Fluent, Inc. Lebanon, New Hampshire). Air flow in human nasal cavities is steady or quasi-steady state flow for quiet breath [14, 43, 71]. The governing equations for steady state air flow are the Navier-Stokes equations [15]. To solve the full set of governing equations, a no-slip and stiff boundary

condition was applied on the surface of the cavity. For simulation of inspiratory flow, a stress free boundary condition (Neuman boundary) was assigned at the outlet (posterior of the cavity), and a uniform velocity field was imposed at the external naris. The simulation results were consistent with other researchers' work. A comparison study on the six human models was conducted. To our knowledge this is the largest study on air flow simulation of human nasal cavities. Since no experimental data, and no published material are available at this time for the dog study, steady state air flow is simulated in the dog model, though the flow may be highly turbulent. Part of this work has been published [97].

1.6.4 Olfactory Epithelium Shape Modeling with System Identification

The fourth contribution is that a mathematical model for the shape of the olfactory epithelium or olfactory neurons to odorant stimuli experimentally measured in rats and dogs is proposed. The linear input-output model combines equations for the odorant stimulus input, the disturbance-free output, the measured output, and the measurement noise. The model predicts the shape of the responses to different odorant concentration for a fixed duration of stimuli. The model has the potential to be used to evaluate olfactory electrical responses and to estimate kinetics of G-protein cascade within the olfactory receptor neuron.

1.7 Dissertation Organization

This dissertation is organized as follows. In Chapter 2 the hierarchical spline-based interpolation for nasal cavity reconstruction is introduced. In Chapter 3 the marching volume element (MVE) algorithm is developed. In Chapter 4 air flow simulation results

on both human and dog nasal cavity models are presented. In Chapter 5 the mathematical EOG shape model with system identification techniques is developed. Finally, we conclude the dissertation in Chapter 6 with the summary and a discussion of future research.

CHAPTER 2

HIERARCHICAL SPLINE-BASED INTERPOLATION

Human nasal cavity images created by computed tomography (CT) present large offset and have high dissimilarities between adjacent slices because of thickness of tomographic slices is much larger than the imaging plane resolution. Large differences are observed between adjacent slices of human nasal cavity. It is very difficult to construct an accurate 3D geometric model, which is intended to be used in computational fluid dynamics (CFD) analysis, directly from the obtained image slices. Interpolation techniques must be used to recover the lost information between adjacent slices. In our experiments, several algorithms, including linear, shape-based and adaptive control grid interpolation(ACGI) algorithms, failed to perform this task, only the hierarchical spline-based image registration method was able to produce satisfactory slice interpolation. The algorithm is capable of handling large offsets and high dissimilarities among images. With this interpolation algorithm, a series of smooth human nasal cavity were reconstructed.

2.1 Review of Interpolation Methods

Extracting and rendering anatomical objects from computed tomography (CT) or magnetic resonance imaging (MRI) data has many applications in science and medicine. CT and MRI, however, typically image a three-dimensional (3-D) sample as a sequence of thick slices, where the slice thickness can be three or more times the pixel size in the image plane. Consequently, before a surface can be extracted from the data using

an algorithm such as marching cubes [57], slices must be interpolated between acquired slices to make the voxel size approximately equal in all three directions. Without slice interpolation, the surface reconstructed directly from the acquired image slices will not be smooth and have block artifacts. As shown in Figure 2.1, the surface of a human nasal cavity reconstructed from 31 CT slices without slice interpolation is not smooth and has block artifacts. Interpolation algorithms are needed to solve the problems.

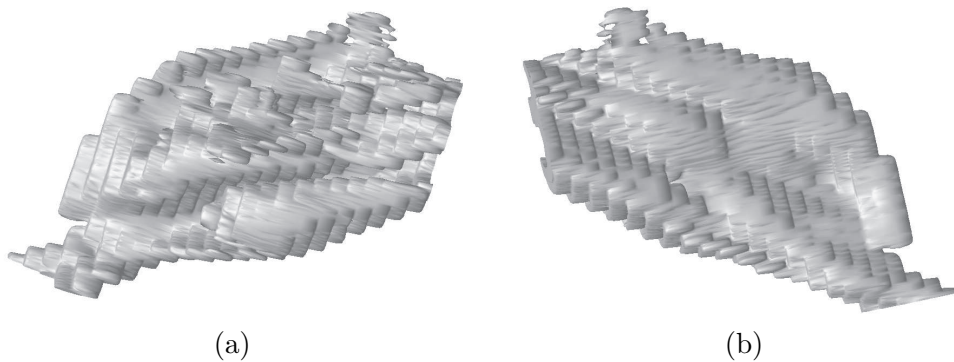


Figure 2.1: 3D surface of a human left cavity constructed from 31 CT slices without slice interpolation. (a) left side of view, (b) right side of view.

Many algorithms for slice interpolation have been developed over the years. Linear interpolation is the simplest one, where the value of each pixel in the intermediate slice is linearly interpolated from the corresponding pixels in each neighboring slice. The problem with linear interpolation is that if the object in the slices have large offset or small overlap, the interpolated images will be blurred as shown in Figure 2.2. In the figure, the most left and the most right images are the two artificial original images, the image in between is obtained with the linear interpolation method. An ideal interpolation would create a smooth interpolation between the original images, as shown in Figure 2.3.

In contour-based [9, 74, 1, 10] approaches, each acquired slice is segmented first, then the borders surrounding regions of interest are linked with a specified function.

This function approximates the location of borders in the interpolated image. In shape-based algorithms [5, 28, 33, 34, 36, 52, 76], each acquired slice is segmented and converted into a binary image. From each binary image, a distance image is computed, where each pixel value is the shortest distance from the pixel to the nearest object boundary. Pixels inside an object are assigned positive distances and pixels outside an object are assigned negative distances. Intermediate images are obtained by linearly interpolating the two calculated distance images. The drawback of the shaped-based algorithm is that if images are poorly segmented, errors will be magnified during interpolation process. In addition, since the algorithm relies on local image information for interpolation, performance can be poor when the offsets between adjacent slices are large. As shown in Figure 2.4, the most left and the most right images are the two artificial original images, and the image in between is obtained with a shape-based interpolation method. If the objects on two images do not have an overlap at all, the performance of the shape-based algorithms will be worse, and an empty intermediate image will be produced. Adaptive controlled grid interpolation (ACGI) [21] is a block-based method for slice interpolation. The ACGI algorithm uses block-matching with adaptive block size and optical flow techniques to compute a displacement field that warps one image to fit an adjacent image. The intermediate images are found by linearly interpolating the displacement field. ACGI has been successfully used for interpolating tomographic slices of blood vessels, but it does not perform well when there is a large difference between adjacent acquired slices as shown in Figure 2.5. The most left and the most right images are the two artificial original images, and the image in between is obtained with the ACGI interpolation method.

In this research, the hierarchical spline registration (HSR) algorithm for slice interpolation is proposed [4, 90]. The HSR algorithm uses block-matching and optical flow techniques as in ACGI, but HSR uses an image pyramid structure [7], and runs in a coarse-to-fine fashion on the image pyramid structures. This algorithm is able to handle large differences between two images.



Figure 2.2: Artificial square blocks interpolated with the linear interpolation algorithm.



Figure 2.3: An ideal slice interpolation of artificial square blocks.



Figure 2.4: Artificial square blocks interpolated with a shape-based interpolation algorithm.

2.2 Hierarchical Spline-Based Interpolation Algorithm

As in the ACGI algorithm, the HSR algorithm is also a combination of the optical flow and block matching method. But there are two major differences between the HSR and ACGI algorithms. The first difference is that the HSR works in a coarse-to-fine fashion on image pyramids, while the ACGI works on the original image. The second difference is that a fixed block size is used in the HSR algorithm, while the block size of the ACGI algorithm is determined by the estimated error in a local region.



Figure 2.5: Artificial square blocks interpolated with the adaptive control grid interpolation (ACGI) algorithm.

In the HSR algorithm, optical flow technique is used to find the connection or displacement field between two adjacent slices. Intermediate slices can be found by interpolating the estimated displacement field. The basic concept of the method is shown in Figure 2.6. The optical flow can find the pixel correspondence in two images. The use of block matching can reduce the computational cost, and an image pyramid structure is adopted to accelerated the optimal searching.

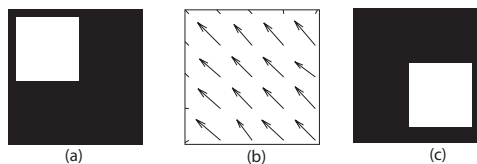


Figure 2.6: Displacement field for square images. (a) The target image. (b) The estimated displacement field. (c) The source image

Details of the algorithm will be presented in following sections. In section 2.2.1, the general image registration equation is given. In section 2.2.2, a spline-motion model is introduced to approximate the displacement field in a block. In section 2.2.3 the minimization problem is solved with the Levenberg-Marquardt iteration method [70]. The image pyramid structure of the HSR algorithm is presented in Section 2.2.4.

2.2.1 General Equations

The optical flow technique is traditionally used in motion analysis in computer vision and video compression [3, 2, 37, 69, 90, 89, 91]. The basic assumption of the

optical flow is that each pixel travels along a unique path from one frame (image) to another frame (image). The brightness of each pixel should remain the same, which is shown in equation (2.1), where $I_1(x + u, y + v)$ and $I_2(x, y)$ are two adjacent images. u and v are the displacement fields.

$$I_1(x + u, y + v) = I_2(x, y) \quad (2.1)$$

In this research, the slice direction in space domain is treated as an analog of time in the original optical flow equation. The problem of optical flow technique is to find u and v for all pixels simultaneously, so that the first image will match the second image. The maximum likelihood solution for this problem is to minimize the squared error shown in (2.2).

$$E(u, v) = \iint_{\Omega} [I_1(x + u, y + v) - I_2(x, y)]^2 dx dy \quad (2.2)$$

For digital image data, the integral can be replaced by the summations over all pixels (x_i, y_i) over the entire image shown in (2.3), where u_i and v_i are the displacements of pixel i in x and y directions respectively, and x_i and y_i are the coordinates of pixel i .

$$E(\{u_i, v_i\}) = \sum_i [I_1(x_i + u_i, y_i + v_i) - I_2(x_i, y_i)]^2 \quad (2.3)$$

The equation (2.3) is called the sum of squared differences (SSD) [1]. The minimization problem typically has many local minima. Hierarchical technique can speed up the optimal search and avoid local minima.

2.2.2 Spline motion model

The minimization problem in (2.3) is difficult to solve for ordinary medical images, since the two displacements, u_i and v_i , for each pixel need to be estimated. For an image size of 512, there will be $512 \times 512 \times 2 = 524,288$ unknowns to be estimated. Solving a problem of this size requires large memory and computational power. Blocking matching is used to reduce the size of the problem. Instead of finding the displacements for each pixel, displacements at coarser grids are calculated, as shown in Figure 2.7. The displacements of other pixels are defined by a spline-based motion model controlled by displacements on the coarser grids.

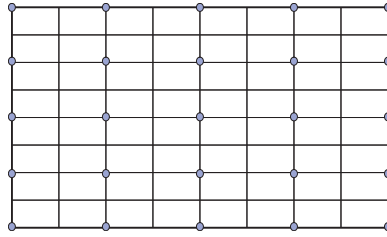


Figure 2.7: Displacements on coarser grid are calculated.

The motion between the coarse grids is assumed to be continuous. Any motion model which produces a continuous displacement field may be used. In a motion model, the displacement fields (u, v) are represented by two-dimensional splines which are controlled by a smaller number of displacement estimates \hat{u}_j and \hat{v}_j on the coarser grid. The displacement at pixel i can be written as shown in (2.4) and (2.5), where $\Phi_j(x, y)$ is the basis function at pixel j . The basis function is only non-zero over a small region (finite support). w_{ij} is the value for $\Phi_j(x, y)$ at pixel i . Equation (2.4) and (2.5) state that the

displacement (u_i, v_i) at each pixel is a weighted combination of (\hat{u}_j, \hat{v}_j) .

$$u(x_i, y_i) = \sum_j \hat{u}_j \Phi_j(x, y) = \sum_j \hat{u}_j w_{ij} \quad (2.4)$$

$$v(x_i, y_i) = \sum_j \hat{v}_j \Phi_j(x, y) = \sum_j \hat{v}_j w_{ij} \quad (2.5)$$

Any coarse grid can be used in the block matching. In this work, a regular sub-sampling of the pixel grid is used as the spline control grid, and the basis functions are spatially shifted versions of each other. In this research, the bilinear interpolation is used as the basis function $\Phi_j(x, y)$, whose value is one at the spline control grid j , and zero on all other control grid points (finite support). As described in equation (2.4) and (2.5), the spline-based model ensures a continuous displacement field.

2.2.3 Displacement Field Estimation

With the spline-based motion model in Equation (2.4) and (2.5), the minimization problem in (2.3) becomes a minimization problem in (2.6) with respect to displacements $\{(\hat{u}_j, \hat{v}_j)\}$ at the coarse grids.

$$E(\{\hat{u}_j, \hat{v}_j\}) = \sum_i \left[I(x_i + \sum_j \hat{u}_j w_{ij}, y_i + \sum_j \hat{v}_j w_{ij}) - I(x_i, y_i) \right]^2 \quad (2.6)$$

Minimization of the least square problem in (2.6) is equivalent to solving the linear equation (2.7), where H is the Hessian matrix, g is the gradient vector, u is the displacement vector.

$$Hu = -g \quad (2.7)$$

The gradient g can be calculated in by equation (2.8), where $e_i = I_1(x_i + u_i, y_i + v_i) - I_2(x_i, y_i)$ is the intensity error at pixel i . $G_i = \nabla I_1(x_i + u_i, y_i + v_i)$ is the intensity gradient of I_1 at the displaced position for pixel i .

$$g_j = \left[\frac{\partial E}{\partial \hat{u}_j} \quad \frac{\partial E}{\partial \hat{v}_j} \right]^T = 2 \sum_i e_i G_i w_{ij} \quad (2.8)$$

The Hessian matrix is composed of 2×2 blocks, which are given by

$$H_{jk} = \frac{\partial^2 E}{\partial \hat{u}_j \partial \hat{u}_k} = 2 \sum_i \left[G_i G_i^T w_{ij} w_{ik} + e_i \frac{\partial^2 I_1(x_i + u_i, y_i + v_i)}{\partial \hat{u}_j \partial \hat{u}_k} \right] \approx 2 \sum_i \left[G_i G_i^T w_{ij} w_{ik} \right] \quad (2.9)$$

The second term in (2.9) is ignored because it is negligible compared to the first term.

If the Hessian matrix is a small matrix and is invertible, Equation (2.7) can be solved directly. However, since the Hessian matrix usually is large matrix, it is too expensive to solve the equation (2.7) directly. A variant of Levenberg-Marquardt iterative minimization technique [70] is used to solve the problem. In this technique, the increment Δu of the current displacement estimation u is calculated by equation (2.10), where λ is a stabilization factor [89] which varies over time, Δu is an increment vector of the displacement estimates \hat{u}_j and \hat{v}_j .

$$(H + \lambda I) \Delta u = -g \quad (2.10)$$

Δu is estimated with the preconditioned gradient descent method in equation (2.11) [89], where α is the step size in the current direction, and \hat{g} is the preconditional gradient.

$$\Delta u = -\alpha \hat{g} \quad (2.11)$$

The preconditional gradient \hat{g} is calculated by equation (2.12):

$$\hat{g} = [\text{block_diag}(H) + \lambda I]^{-1} g \quad (2.12)$$

In equation (2.12), $\text{block_diag}(H)$ is a set of 2×2 block diagonal matrices of H , and I is the identity matrix. The optimal value of α is chosen to minimize the E of (2.6) in the current direction \hat{g} . Since the difference of E in current direction can be calculated by equation (2.13), by setting $\Delta E = 0$, we obtain $\alpha = (\hat{g}^T g) / (\hat{g}^T H \hat{g})$.

$$\Delta E = E(x - \alpha \hat{g}) - E(x) \approx -\alpha \hat{g}^T g + \frac{\alpha^2}{2} \hat{g}^T H \hat{g} \quad (2.13)$$

Replacing α and \hat{g} in equation (2.11), we can obtain the increment vector in current iteration.

After each iterative step, the displacement field is updated by $u_{\text{current}} = u_{\text{previous}} + \Delta u$, and the Hessian and gradient are updated by the new displacement. The criterion to terminate the iteration can be either the maximum amount of iterations reached or the minimum SSD obtained.

2.2.4 Hierarchical image structure

The HSR algorithm uses image pyramids to accelerate optimal search and convergence. An image pyramid is constructed for each CT slice, and the displacement fields are found by a coarse-to-fine fashion on the image pyramids. The flow field estimation algorithm depicted above in section 2.2.3 is used on these image pyramids. The displacement fields are estimated on coarsest images (the top images on the pyramids) first. This displacement field is then used as initial estimate of the displacement field for the next

higher resolution images. This process continues until the bottom image in the pyramid is reached, which is the original image. The intermediate images are obtained by interpolating the final displacement fields. The hierarchical image pyramid used is the Gaussian

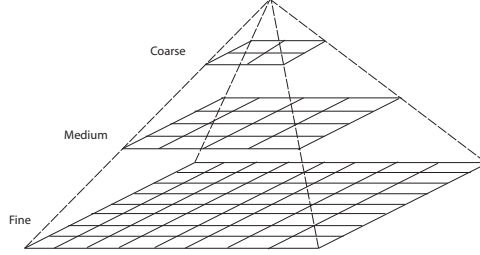


Figure 2.8: The structure of Gaussian pyramid.

image pyramid [7], which is constructed with a separable 3 point filter. The kernel of the filter is a Gaussian-like low-pass filter [88]. In the hierarchical image pyramid, a higher level image is a decimated version of its adjacent lower level image. Figure 2.8 shows the structure of the hierarchical pyramid, and Figure 2.9 shows a 5 level Gaussian pyramid of a human nose CT slice. The bottom image in the pyramid is the original image. To ensure continuity of the displacement field, adjacent blocks overlap each other by one pixel, so that the original image should have a size of $(m \times 2^L + 1, n \times 2^L + 1)$, which is the finest image in the Gaussian pyramid. L is the level of the pyramid. The image size at next level will be $(m \times 2^{L-1} + 1, n \times 2^{L-1} + 1)$. The size of the coarsest image is $(m+1, n+1)$. Denoting $I_l(i, j)$ as the level l image, where $L \geq l \geq 1$, the image in the next level can be formed by (2.14).

$$I_{l+1}(i, j) = \sum_{r=-1}^1 \sum_{c=-1}^1 w(r, c) I_l(2i + r, 2j + c) \quad (2.14)$$

where w is the Gaussian filter given by (2.15).

$$w = \begin{bmatrix} 1/4 & 1/2 & 1/4 \end{bmatrix}^T \begin{bmatrix} 1/4 & 1/2 & 1/4 \end{bmatrix} \quad (2.15)$$

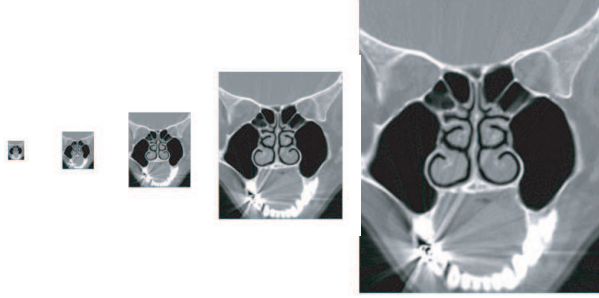


Figure 2.9: An image pyramid of a human nose CT slice.

2.2.5 Implementation considerations

Several parameters need to be determined in the implementation of the HSR algorithm. First, a maximum displacement in each iteration step is set to 1 pixel in our current algorithm implementation. Too large displacement in one step may prevent the algorithm from finding the optimal value. The introduction of limited increment displacement can reduce the possibility of the overstep size in the minimum search procedure. Since the HSR works on Gaussian image pyramids [7] in a coarse-to-fine fashion, and the displacement at a higher level is used as the initial displacement estimation in a lower level in the Gaussian pyramids, small additional displacements are expected at each level. Second, block size has to be chosen. The selection of block size can affect both computational cost and the final results. Usually, larger block size can result in faster convergence with larger SSD, while small block size can result in slower convergence

with smaller SSD. Too small block size will increase the computation cost dramatically. Moreover, noise may become significant. Therefore, there is a tradeoff between computation cost and selection of block-size. From our experiments, for regular CT images, a block size from 6×6 to 16×16 could be chosen. In our implementation, the mean square difference (MSD) was used instead of SSD, because SSD may be very large and can overflow. The effects of block size are illustrated in Figure 2.10, where two 257×257 cut CT images are used, and the computation times and the minimum MSDs are normalized with their maximum values respectively. Finally, a criterion to stop the iteration on each level needs to be selected. It can be either a minimum SSD or a maximum iteration times. In our current implementation, a fixed number of iterations on each level was adopted.

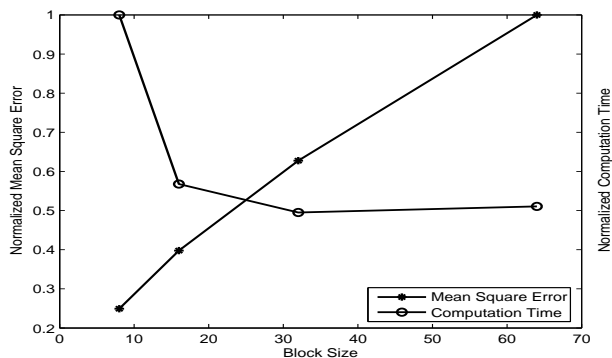


Figure 2.10: Normalized execution time and mean square error vs. block-size for the HSR algorithm. The computation time normalized by the maximum computation time. The mean square error is normalized by the maximum error. The results are based on two cut CT the images with a size of 257×257 .

2.3 Experimental Results

The HSR algorithm was used to interpolate an artificial data set and real human nasal-cavity CT slices, and was compared with three other algorithms.

The most left and right columns in Figure 2.11, are two artificial images which are 17×17 binary images with a 8×8 square in each of them. The squares have a large offset in the images. Assuming these two images are cross-section images of a bar at different positions, we are trying to estimate the cross-section images at $1/3$, $1/2$ and $2/3$ of distance between the two original images with interpolation algorithms. The two artificial images were interpolated with four algorithms, including linear, shape-based, ACGI and HSR interpolation algorithms. For comparison purposes, the original images are duplicated in each case. Linear interpolation creates blurred images, as shown in Figure 2.11(a). The shape-based algorithm is not able to produce satisfying interpolation either, as shown in Figure 2.11(b). The results of ACGI algorithm are shown in Figure 2.11(c), which is not good as well. Only the HSR algorithm produces desirable results, as shown in Figure 2.11(d). In the calculations, a fixed block size of 4×4 was used in both the ACGI and the HSR algorithms. The displacement fields for the ACGI and the HSR algorithms are shown in Figure 2.12 and Figure 2.13 respectively. The mean square difference (MSD) with respect to iteration time of the ACGI and the HSR algorithms are shown in Figure 2.14. For ACGI algorithm, the minimum MSD was 0.0119, and it did not decrease after 20 iterations. The final MSD for the HSR algorithm was less than 1.0×10^{-7} . Three level pyramids were used for the HSR algorithm. The level 1 image was the original image which was the finest image on the pyramid. The image on the level 2 was a decimated version of the original image, and the image on level 3

was a decimated version of the image on the level 2, and was the coarsest image on the pyramid. The displacement field was estimated on the level 3 first, and the calculation of the displacement on level 2 was started from the estimated displacement on the level 3, and this process was repeated on the level 1. The intermediate images were obtained by interpolating the displacement field on level 1. The MSD on each level shown in Figure 2.14 was computed based on the images on the level in the pyramids.

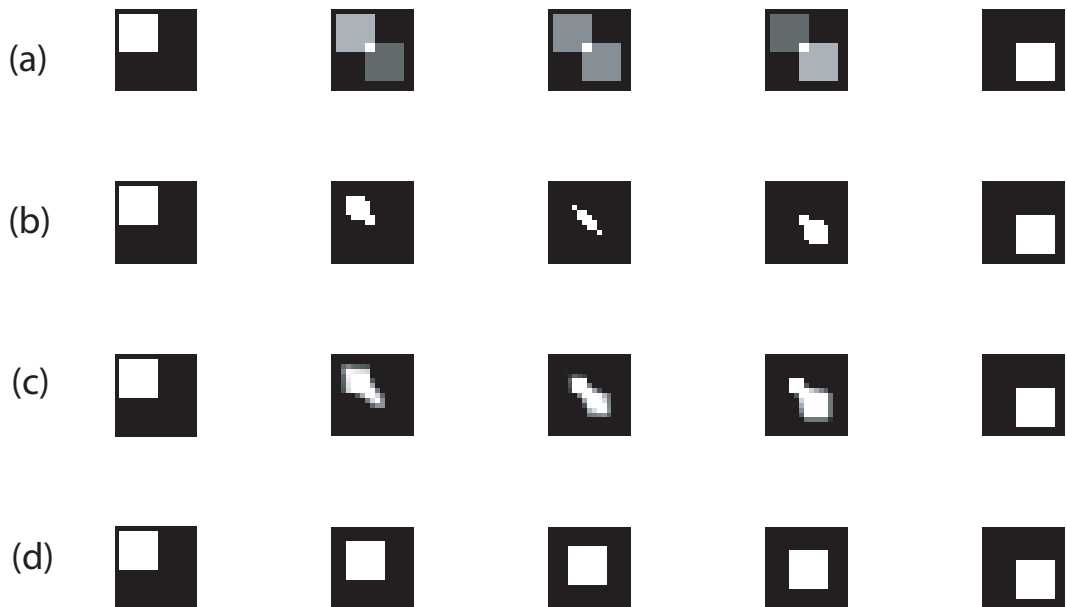


Figure 2.11: Binary square images are interpolated with different methods. The images on the most left side and the most right side are the reference images. The other images are created at $1/3, 1/2$ and $2/3$ distance from the reference images respectively by different methods. (a) linear interpolation. (b) shape-based interpolation. (c) ACGI interpolation. (d) HSR interpolation.

The HSR algorithm was also used to interpolate human nose CT slices. The results were compared with the other three interpolation algorithms. As expected, the HSR algorithm produced the best results. Linear interpolation produced blurred intermediate images, shown in Figure 2.15, where Figure(a) and (c) are two adjacent CT slices, and

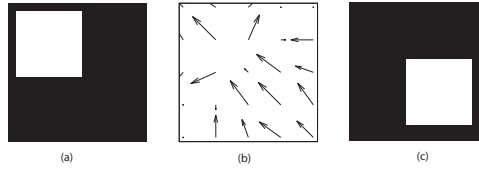


Figure 2.12: Displacement field for square images. (a) The target image. (b) The estimated displacement field by the ACGI algorithm. (c) The source image

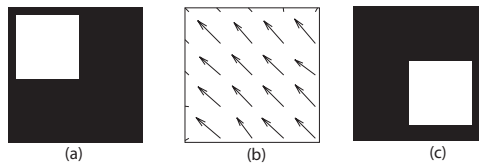


Figure 2.13: Displacement field for square images. (a) The target image. (b) The estimated displacement field estimations by the HSR algorithm.

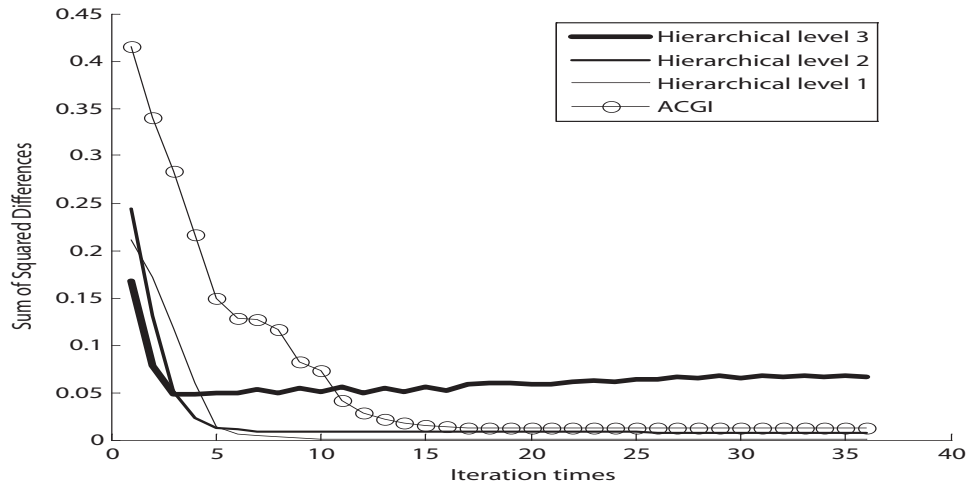


Figure 2.14: Mean squared differences vs. iteration times for the HSR algorithm and the ACGI algorithm. 4×4 block-sizes are used for each case. For the HSR algorithm, three level pyramids are used, where level 1 is the finest, and level 3 is the coarsest.

Figure(b) shows the intermediate slices obtained by linear interpolation at half distance. Shape-based algorithm created disconnected intermediate images, as shown in Figure 2.16, where Figure(a) and (c) are segmented images from Figure 2.15(a) and (c) respectively, and Figure 2.16(b) is interpolated at half distance. ACGI created much better interpolation than linear and shape-based algorithms as shown in Figure 2.17, where (a) and (c) are the same as Figure 2.15(a) and (c), and Figure 2.17(b) is intermediate slices at half distance. The HSR algorithm produced the most satisfying intermediate slice as shown in Figure 2.18(b). The performances of the ACGI and the HSR estimation for CT slices were compared in Figure 2.19. For the same fixed block size, the HSR algorithm converged to a lower MSD than the ACGI algorithm without adding much computation time. Six level pyramids were used for the HSR algorithm. Level 1 was the finest level, and level 6 was the coarsest level.



Figure 2.15: Linear interpolation on human head CT slices. (a) and (c) are two adjacent human head CT slices. (b) is the intermediate slices at half distance by the linear interpolation.

By using the HSR algorithm, a human left nasal cavity was reconstructed based on 31 CT slices (The resolutions of the CT scan were: $0.3125mm \times 0.3125mm$ pixel size, $3mm$ slice thickness. The regions of interest were found with a semiautomatic

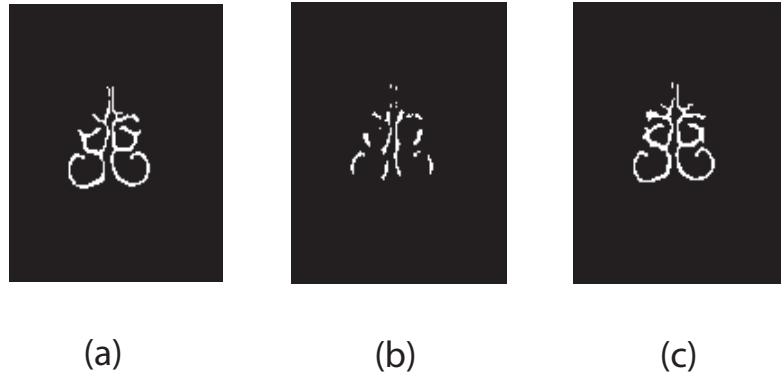


Figure 2.16: Shape-based interpolation on human head CT slices. (a) and (c) are segmented images. (b) is the intermediate slice at half distance by shape-based interpolation.

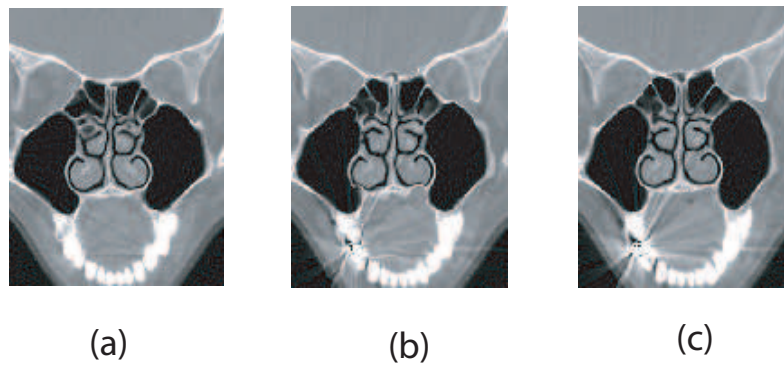


Figure 2.17: ACGI interpolation on human head CT slices. (a) and (c) are two adjacent human head CT slices. (b) is the intermediate slices at half distance by the ACGI interpolation.

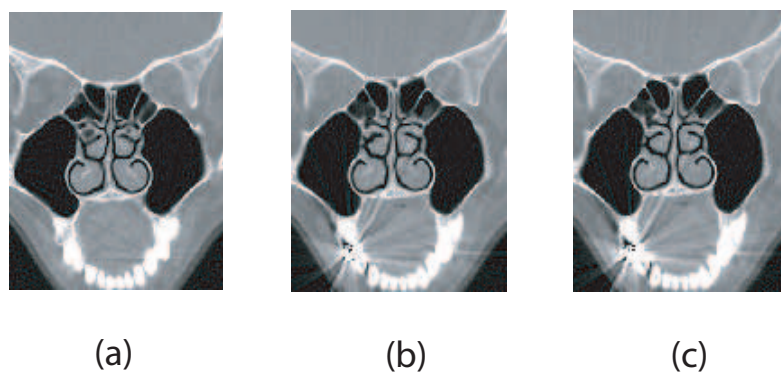


Figure 2.18: HSR interpolation on CT slices. (a) and (c) are two adjacent human head CT slices. (b) is the intermediate slices at half distance by the HSR interpolation.

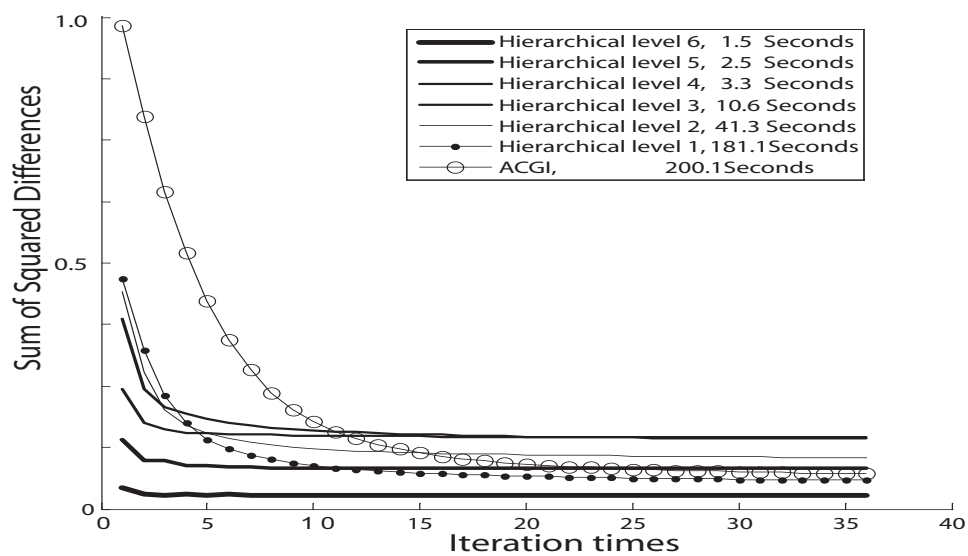


Figure 2.19: Mean Squared Differences vs. Iteration for the HSR algorithm and the ACGI algorithm with human nose CT slices. A block size of 12×12 is used in the calculations. The time cost is also listed. The algorithms are run on a PC which has a Intel Pentium III 549MHz processor and 1.0 GB of RAM. For the HSR algorithm, six level pyramids are used, where level 1 is the finest, and level 6 is the coarsest.

segmentation algorithm. Then, the segmented images were used to create 3D surface and finite element mesh for CFD. Figure 2.20 shows the 3D surface of the left side nasal cavity. The surface is significantly better than the surface constructed with the ACGI algorithm, shown in Figure 2.21.



Figure 2.20: The HSR algorithm is used to construct 3D surface of a human cavity with 31 CT slices. 5 intermediate slices are interpolated between each two adjacent CT slices. (a) left side of view, (b) right side of view.

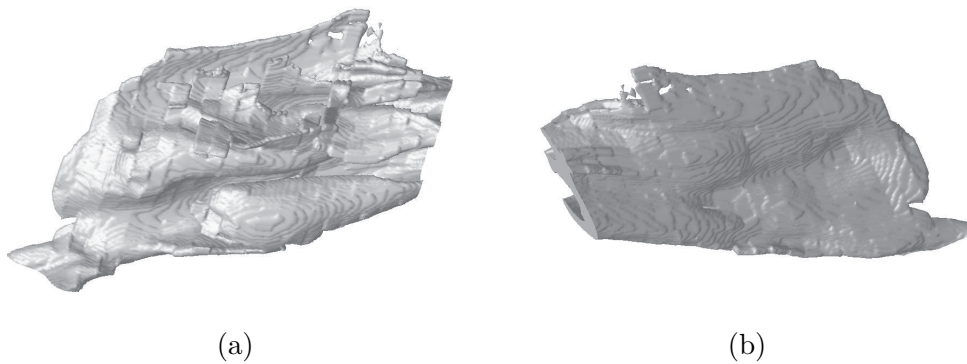


Figure 2.21: The ACGI algorithm is used to construct 3D surface of a human left cavity from 31 CT slices. 5 intermediate slices are interpolated between each two adjacent CT slices. (a) left side of view, (b) right side of view.

2.4 Conclusion

In conclusion, the hierarchical spline-based interpolation was employed to perform human nasal cavity reconstruction in this research. This algorithm has better performance in this application compared to other algorithms tested. The good performance of the HSR algorithm results from its unique pyramidal structures. The matching algorithm starts at the coarsest image on the Gaussian pyramid, where global displacements can be estimated, and the estimate on a higher level propagates to a lower level. Therefore this algorithm is able to cope with large offsets and high dissimilarities between images, and the technique is able to increase the optimal search and avoid local minima.

CHAPTER 3

MARCHING VOLUME ELEMENT ALGORITHM

In this chapter, a new 3-D finite element mesh generation algorithm, called marching volume elements (MVE), will be introduced. The algorithm is able to generate finite element mesh from 3-D gridded data. The algorithm is based on the marching cubes (MC) algorithm which is an isosurface generation algorithm. The MVE algorithm extends the capability of the MC algorithm to create a volume mesh. The mesh generated from the MVE algorithm is composed of pyramids, cubes and tetrahedra.

3.1 Introduction

Computational fluid dynamics (CFD) has many application in biomedical engineering, such as simulating air dynamics in nasal cavities and lungs, blood flow in vessels and the heart. To perform CFD simulations, geometric models and computational meshes have to be constructed. The models and meshes may be developed from tomographic slices of anatomic structures acquired by medical imaging modalities, such as computed tomography (CT) and magnetic resonance imaging (MRI). However, anatomic structures usually have highly irregular shapes. To construct an accurate model of an anatomic structure, a mesh with a large number of elements is needed. Manual construction of models would be tedious and error prone. Because of these reasons an automatic algorithm is highly desired in constructing a volume mesh from CT or MRI data.

Any valid mesh must adhere to geometric and topological consistency requirements [23, 24]. Geometric consistency requires that the final mesh must be an acceptable

approximation to the original geometric representation of the model. Topological consistency on the other hand, refers to the adjacent relations between elements, which requires that adjacent elements share an identical face without overlapping, crossing or penetrating each other.

Many meshing algorithms have been developed over the years. In octree techniques [80, 101, 58], cubes containing geometric models are recursively subdivided until desired resolution is obtained. The problem with this technique is that the element sizes of adjacent cubes may change dramatically, making it difficult to ensure topological consistency. Delaunay based techniques [100, 13, 26, 60, 81] are based on the Delaunay criterion, which says that any node of an element must not be contained within the circumspheres of any other elements. This criterion connects a set of existing points in space. The main drawback of Delaunay techniques is that objects must be convex and have no interior holes. This usually can not be guaranteed for biological structures. If the objects are concave or have interior holes, the concave parts will be eliminated and holes will be meshed as the objects. Therefore, the mesh created by Delaunay techniques may not satisfy the geometric consistency requirement. In the advancing front method [55, 56, 75, 11, 41], tetrahedra are built progressively inward from a triangulated surface. An active front is maintained at where new tetrahedra are formed. For each triangular facet on the front, an ideal location for a new fourth node is computed. The algorithm selects either the new fourth node or an existing node to form the new tetrahedron. The algorithm can start from an arbitrary surface. One disadvantage of the advancing front method is that it is computationally extensive because it must continuously track and calculate the new front. Another drawback is that it does not take advantage of the structure of gridded data.

An algorithm closely related to this work is the volume tetrahedrization (VT) algorithm [64], which utilizes the pixel-based data structure. The algorithm first divides volume data into cubes of the same size. Each cube is then tessellated into 5 tetrahedra first. Then the 5 tetrahedra are decomposed into smaller computational mesh. The problem of this approach is geometric inconsistency [65]. Since there are two ways to tessellate a cube into 5 tetrahedra, as shown in Figure 3.1(a) and (b), the two ways have to be used alternatively on adjacent cubes to keep a shared cube face meshed in the same way. Once the tessellation method is selected for the first cube, the tessellations on all other cubes are defined. However, there is no rule in the VT algorithm to decide which tessellation method is used on the first cube, and different choice for the first cube may result in different meshes with different geometries for a given data set, as a result, obtained mesh may not be geometrically consistent.

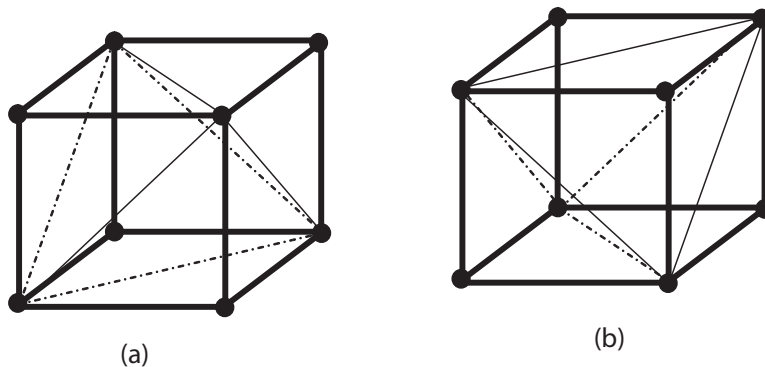


Figure 3.1: Two ways to tessellate a cube into 5 tetrahedra.

In this research, a new volume mesh algorithm, called Marching Volume Element (MVE) algorithm, is presented. The MVE algorithm is based upon the Marching Cubes (MC) algorithm [57] which is an isosurface generation algorithm for 3-D grided data. The MVE algorithm extends the capability of the MC algorithm to create volume mesh as

well. The MVE algorithm takes advantage of the inherent structure of gridded data. It creates a unique, geometrically and topologically consistent mesh, and can handle arbitrarily-shaped regions. Another advantage of the MVE algorithm is that the MVE algorithm is memory efficient. The MVE algorithm operates on local data, and each cube is meshed individually in such a way that topological consistency with its neighbors is ensured. As a result, the MVE is memory-efficient and can construct volume meshes from large data sets that may not be practical to mesh with other techniques.

The rest of this section is organized as follows. The MVE algorithm is presented in Section 3.2. A mesh quality improvement scheme is presented in Section 3.3. The effectiveness of the algorithm is demonstrated with a synthetic data set and a CT scan of a human nose in Section 3.4. Conclusions are presented in Section 3.5.

3.2 Marching Volume Elements Algorithm

The MVE algorithm is an automatic 3-D finite element mesh generation algorithm for intensity data defined on a regularly-spaced 3-D grid. The algorithm is based on the MC algorithm. The MVE can extract the surface of an object, and mesh the interior domain into tetrahedra, pyramids and cubes as well. The tetrahedral elements form the boundary layer of the object, cubes occupy the interior of the computational domain, and pyramids act as transitions from the tetrahedra to cubes. In this section, a brief introduction of MC algorithm will be given first. Then the ambiguous face problem of MC algorithm is solved with a bilinear interpolation method. The ambiguous face problem will be explained in Section 3.2.2. Next, cube face tessellation rules used in the MVE algorithm will be presented. Finally, detailed 3-D mesh scheme of the MVE algorithm will be presented.

3.2.1 Major Cases of the MVE algorithm

In this section, the MC algorithm is overviewed first, then the MVE is presented.

MC algorithm decomposes volume data into cubes, where each cube has 8 vertices and each vertex is a voxel. It is assumed that there is a threshold such that voxel intensities on one side of the threshold are considered to be inside the structure to be meshed and intensities on the other side of the threshold are outside the structure. By comparing a vertex value with a threshold, it can be identified whether the vertex is inside or outside of an object. Inside vertex is marked as white, and outside vertex is marked as black, or vice-versa. Each cube vertex can be inside or outside. After thresholding, each cube containing eight nearest-neighbor voxels has its vertices classified as either inside or outside the structure for a total of 256 different configurations. Using rotational and complementary symmetries, the MC algorithm reduces the 256 cases to 15 basic cases. The triangularization of the 15 basic cases are defined in MC. Two cases are complementary symmetry if the vertices have opposite colors. For example, a cube with one vertex outside and seven vertices inside is complementary symmetric to another cube with 7 outside and 1 inside. The rotational symmetry means cubes which have the same topology will be regarded as one case. For example, cubes which have only one white vertex are topologically the same.

In contrast to the MC algorithm, complementary symmetry cannot be used to construct a volume mesh in MVE, since the volumes of two complementary symmetric cases are not the same. Using rotational symmetry alone, however, the 256 configurations can be reduced to 23 major cases. These cases are shown in Figure 3.2, except for the cases of all inside or outside vertices.

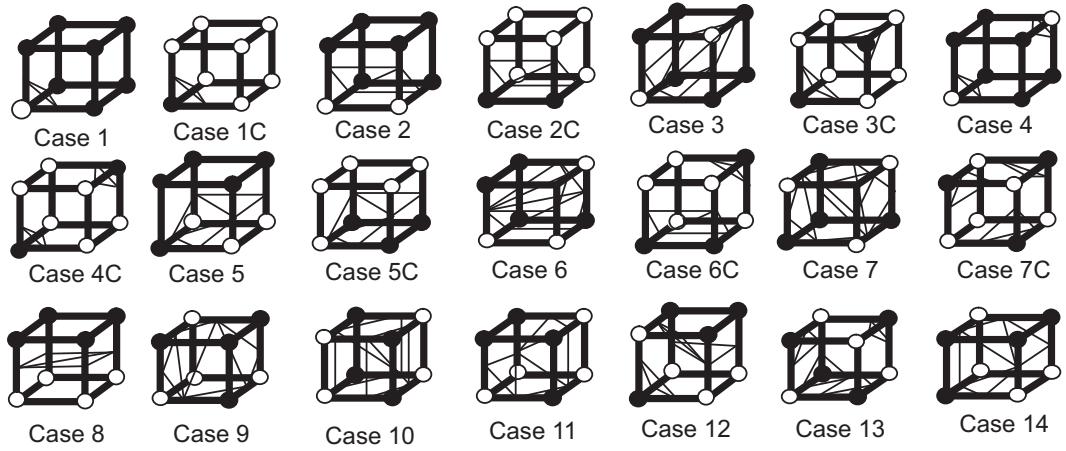


Figure 3.2: 21 major cases of the marching volume element algorithm. Letter C denotes a complementary case. For example, Case 1 and Case 1C are complementary cases of each other. One type of triangularization (thin lines) is shown for each case.

3.2.2 Ambiguous Face Problem and Solution

Without modifications, MC algorithm has the so called discontinuous surface problem [63]. The fundamental reason of the discontinuous surface problem is the ambiguous face problem. The ambiguous face problem happens when two diagonal vertices have different color with the other two diagonal vertices on the same cube face, as shown in Figure 3.3. The two white vertices can be separated as shown in Figure 3.3(a), or they are connected as shown in Figure 3.3(b). A choice has to be made on an ambiguous face. The MC algorithm makes the decision randomly. Consequently, the generated surface may not be continuous. Figure 3.4 shows such a situation. Different connections are made by the MC algorithm on a common ambiguous face of the two adjacent cubes. The two white vertices on the common face are separated in Figure 3.4(a), but they are connected in Figure 3.4(b), such that the isosurface on the two cubes are not continuous. This is the root of the discontinuous surface problem in the MC algorithm.

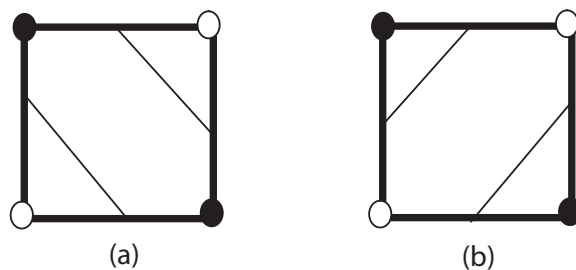


Figure 3.3: An ambiguous face with two black diagonal vertices and two white diagonal vertices. (a) Two white vertices are separated. (b) Two white vertices are connected.

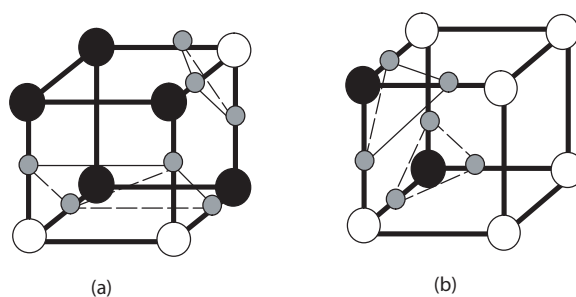


Figure 3.4: Example of discontinuous surface created by the MC algorithm. Cube (a) and (b) share a common face. The connection of the common face of cube (a) and (b) are different. The two white vertices on the common face of cube (a) are separated. The two white vertices on the common face of the cube (b) are connected. This is the fundamental reason of the discontinuous surface problem of the MV algorithm.

Cases 3, 3C, 6, 6C, 7,7C, 10, 12 and 13 in Figure 3.2 are topologically ambiguous, since one or more ambiguous faces exist in these cases. To solve the ambiguous problem, multiple subcases are introduced for each of these ambiguous cases. For example, Case 3 in Figure 3.2 has one ambiguous face, and there are two possible subcases as shown in Figure 3.5(a) and (b). The number of subcases depends on the number of ambiguous faces. The total number of subcases may be reduced by rotation. For example, Case 10 has two ambiguous faces, so there should be 4 possible subcases, as shown in Figure 3.6. But Figure 3.6(b) and (c) are equivalent to each other by rotation, so the 4 subcases reduce to 3 subcases. All six faces of the Case 13 are ambiguous, so the total combination should be 64 subcases. However, by rotation symmetry, only 11 subcases are left, as shown in Figure 3.7. Table 3.1 summarizes the number of subcases for all ambiguous cases.

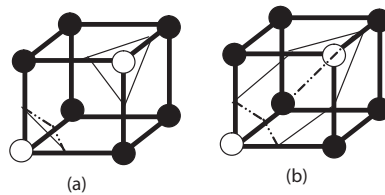


Figure 3.5: Two subcases for Case 3 (3A and 3B) in Figure 3.2.

Table 3.1: Subcases of Ambiguous Cases

| Cube Case | Number of Ambiguous Faces | Number of Possible Subcases | Number of Remaining Subcases |
|-----------|---------------------------|-----------------------------|------------------------------|
| 3,3C,6,6C | 1 | 2 | 2 |
| 10 | 2 | 4 | 3 |
| 12 | 2 | 4 | 4 |
| 7,7C | 3 | 8 | 4 |
| 13 | 6 | 64 | 11 |

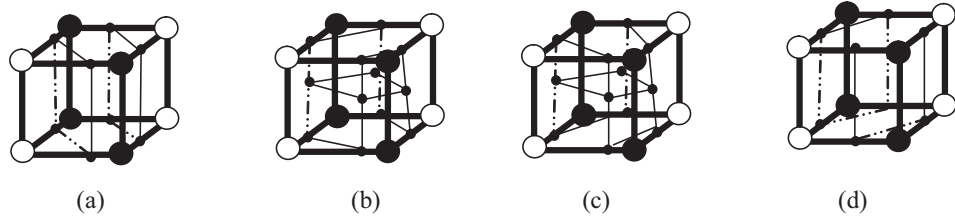


Figure 3.6: Subcases for the Case 10 in Figure 3.2. (a) white vertices on two ambiguous faces are separated. (b) white vertices on one ambiguous face are separated (bottom face), while white vertices on the other ambiguous face are connected (top face). (c) white vertices on one ambiguous face are separated (top face), while the white vertices on the other ambiguous face are connected (bottom face). (d) white vertices on two ambiguous faces are connected. Note (b) and (c) are equivalent by rotation.

The next problem is how to select whether two inside vertices on an ambiguous face are connected or not. The MVE uses a bilinear method [63] to make the decision and choose the correct subcase. This method is based upon the four vertex values of the ambiguous face, and makes a unique connection decision. It is assumed that the intensity has a bilinear variation over a cube face, so the intensity at each point $\{(s, t) : 0 \leq s \leq 1, 0 \leq t \leq 1\}$ inside the square can be calculated by bilinear interpolation of intensities on the four corner points as in (3.1), where I_{00} , I_{01} , I_{10} and I_{11} are the intensity values of the four corner points.

$$I(s, t) = \begin{bmatrix} 1 - s & s \end{bmatrix} \begin{bmatrix} I_{00} & I_{01} \\ I_{10} & I_{11} \end{bmatrix} \begin{bmatrix} 1 - t \\ t \end{bmatrix} \quad (3.1)$$

The bilinear function is totally determined by the four vertex values. For an ambiguous face, two diagonal vertices have larger intensity values than the other two diagonal vertex values. Figure 3.8 shows the typical shape of the bilinear function on an ambiguous face, where the intensity value is used as the third dimension. The intensity values at the four vertices are $\{5, 1, 0, 4\}$ in this figure. Intersecting the surface by a plane which

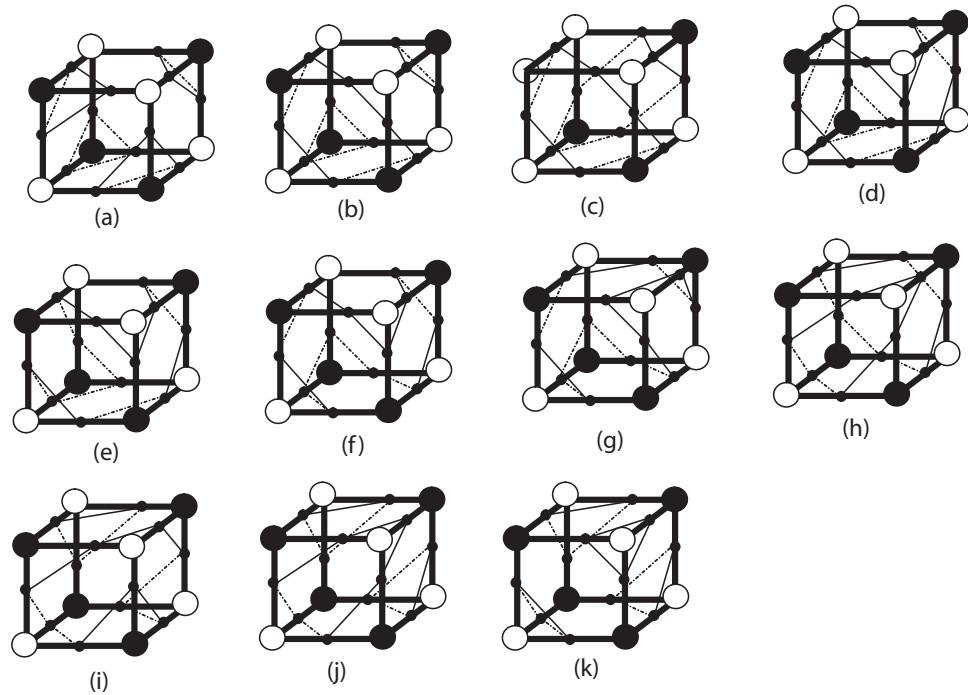


Figure 3.7: Subcases for the Case 13 in Figure 3.2. (a) White vertices on all ambiguous faces are connected. (b) White vertices on five ambiguous faces are connected, while white vertices on one ambiguous face are separated (front face). (c) White vertices on two parallel ambiguous faces are separated (front and back faces), while the white vertices on the other ambiguous faces are connected. (d) White vertices on two adjacent ambiguous faces are separated (front and right faces), while the white vertices on the other ambiguous faces are connected. (e) White vertices on three adjacent ambiguous faces are separated (front, right and left faces, notice the three faces share two edges), while the white vertices on the other ambiguous faces are connected. (f) White vertices on three adjacent ambiguous faces are separated (front, right and bottom faces, notice the three faces share three edges), while the white vertices on the other ambiguous faces are connected. (g) White vertices on three adjacent ambiguous faces are separated (front, top and bottom faces, notice the three faces share two edges), while the white vertices on the other ambiguous faces are connected. (h) White vertices on two parallel ambiguous faces are connected (front and back faces), while the white vertices on the other four ambiguous faces are separated. (i) White vertices on two adjacent ambiguous faces are connected (front and right faces), while the white vertices on the other four ambiguous faces are separated. (j) White vertices on one ambiguous face are connected (front face), while white vertices on other ambiguous faces are separated. (k) White vertices on all ambiguous faces are separated.

has a constant intensity value can determine the connections on the face. As shown in Figure 3.8, two planes intersect the surface. The first plane has an intensity value of 3.5. The second plane has an intensity value of 1.5. The intersections of the first plane with the surface show that the two vertices with higher intensity values should be separated at the threshold of 3.5. The intersections of the second plane and the surface indicate that the two vertices with higher intensity values should be separated for the threshold of 1.5. The connection on the surface can be uniquely determined by a threshold and the bilinear function, except for the saddle point. The saddle point is a point on the surface where the connection changes direction. When a threshold equals the value at the saddle point, the two high intensity vertices can be either connected or disconnected. As long as a consistent choice is made, the result will be topologically correct.

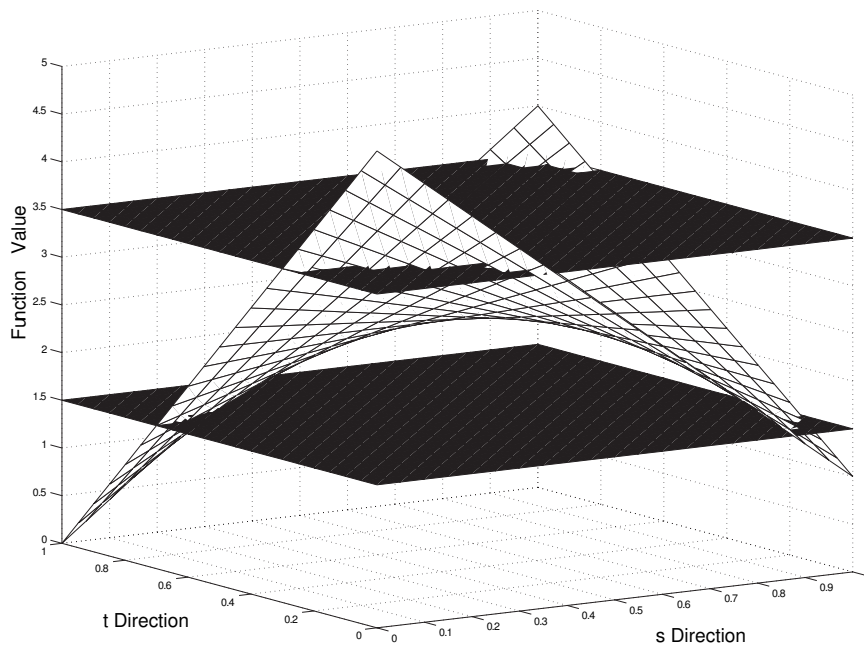


Figure 3.8: Bilinear interpolation.

The value of the saddle point can be calculated with equation (3.1) by replacing the position of the saddle point (s_{saddle}, t_{saddle}) . The position of the saddle point can be found by setting partial derivatives of $I(s, t)$ equal to zeros and solving the equation set (3.2).

$$\begin{cases} \frac{\partial I(s,t)}{\partial s} = 0 \\ \frac{\partial I(s,t)}{\partial t} = 0 \end{cases} \quad (3.2)$$

The solution is given by (3.3).

$$\begin{cases} s_{saddle} = \frac{I_{00} - I_{01}}{I_{00} + I_{11} - I_{01} - I_{10}} \\ t_{saddle} = \frac{I_{00} - I_{10}}{I_{00} + I_{11} - I_{01} - I_{10}} \end{cases} \quad (3.3)$$

Therefore, the value of the saddle point can be calculated using the equation (3.1). The result is shown in (3.4).

$$I_{saddle} = \frac{I_{00}I_{11} + I_{01}I_{10}}{I_{00} + I_{11} - I_{01} - I_{10}} \quad (3.4)$$

3.2.3 Face Tessellation Rules

To ensure a topologically consistent mesh, adjacent cubes must have the same tessellation in their adjoining face. To guarantee the consistency, the MVE algorithm uses the following rules to determine a cube face tessellation.

Face tessellation rules:

- Rule 1: If four vertices are white on a cube face, no tessellation is necessary as in Figure 3.9(a).
- Rule 2: If there are three white vertices on a cube face, split the face into three triangles as in Figure 3.9(b). The three triangles are 1a4, 12b and 1ba.

- Rule 3: If there are two white vertices on a cube face, and they are on the same edge, mesh the face into three triangles as shown in Figure 3.9(c). The three triangles are 1ca, 12c and 2bc, and the position of c is the midpoint of the line connecting a and b.
- Rule 4: If there are two diagonal white vertices on a cube face as shown in Figure 3.9(d), two possible solutions exist. This face is tessellated into two triangles (1ad, 3cb) and one quadrilateral (abcd), or two triangles (1ad and 3cb) depending on the decision made by the bilinear method.
- Rule 5: When there is only one white vertex on a cube face, the surface can be meshed as a triangle 1ab, which is shown in Figure 3.9(e).

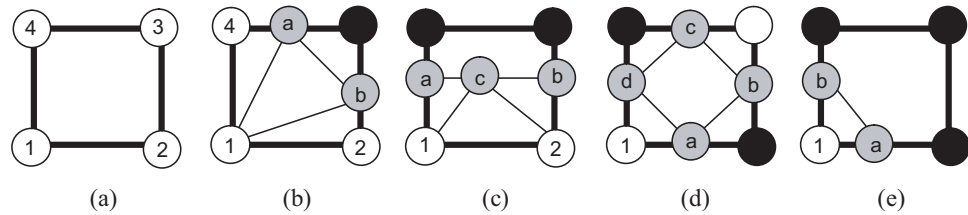


Figure 3.9: Tessellation rules for different cube face configurations.

3.2.4 Volume Mesh Generation

Adding all the unambiguous cases and the subcases for each ambiguous case, there are a total of 48 configurations. The notation used to define each configuration is introduced in Figure 3.10. Numbers from 1 to 8 are used to represent the eight cube vertices. Letters A-L represent the isopoints on edges as shown by small black points. An isopoint exists on an edge if the two edge vertices have different colors. In Figure 3.10, the isopoints are shown in the middle of edges, while the actual position of an isopoint is

calculated by linearly interpolating the two edge vertex intensities. Letter P-U represent the six cube face points. The positions of the face points are defined by Rule 3 in the previous section. The letter O represents a point inside the cube which is shown by the small gray point.

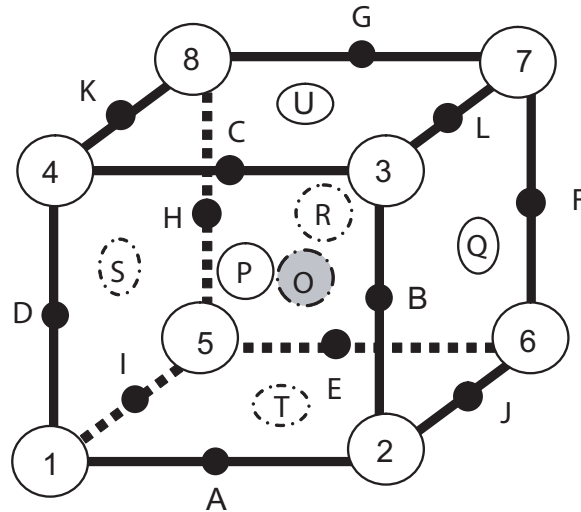


Figure 3.10: Marching volume element notation. Numbers 1-8 represent eight cube vertices. Letter O represents the center point of the cube. Letters A-L represent isopoints on edges. The centers of faces are represented by letters P-U.

Three typical cases are tessellated step by step in Figure 3.11, 3.12 and 3.13 for Case 1C, Case 2 and Case 2C, respectively. In each case, the original topology is displayed first, then an arrow points to the next topology in which the volume outside the isosurface has been cut off, and the isosurface is shown with thin lines. The inside volume is then meshed into elements step-by-step. In each step the volume is decomposed into either tetrahedra or a pyramid and the remaining volume.

Figure 3.11 shows the tessellation process of Case 1C. The final mesh elements for this case include four tetrahedra {A37D, I78D, DAI7, A6I7}, three pyramids {4837D, 2637A, 5687I}, and the boundary surface is one triangle {DIA}. Figure 3.12 shows

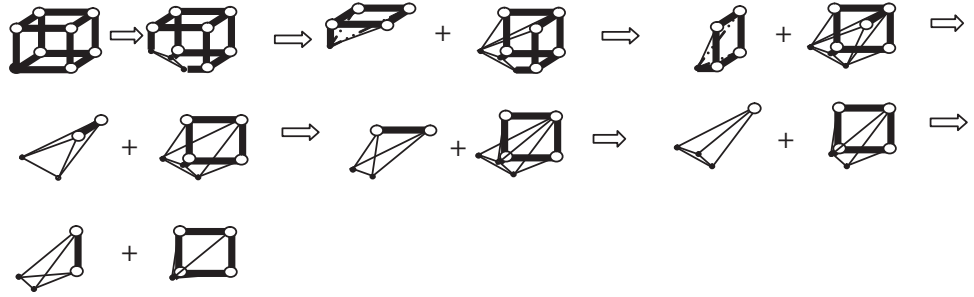


Figure 3.11: Volume mesh decomposition for Case 1C.

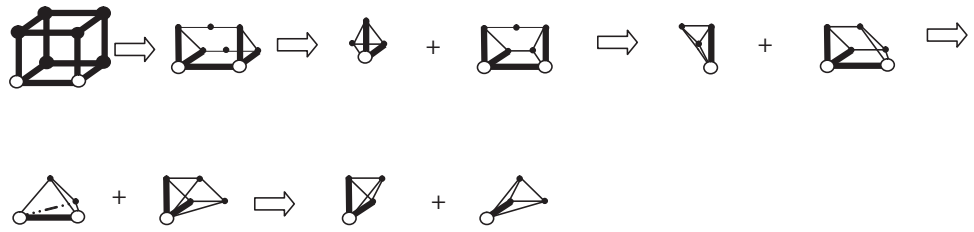


Figure 3.12: Volume mesh decomposition for Case 2.

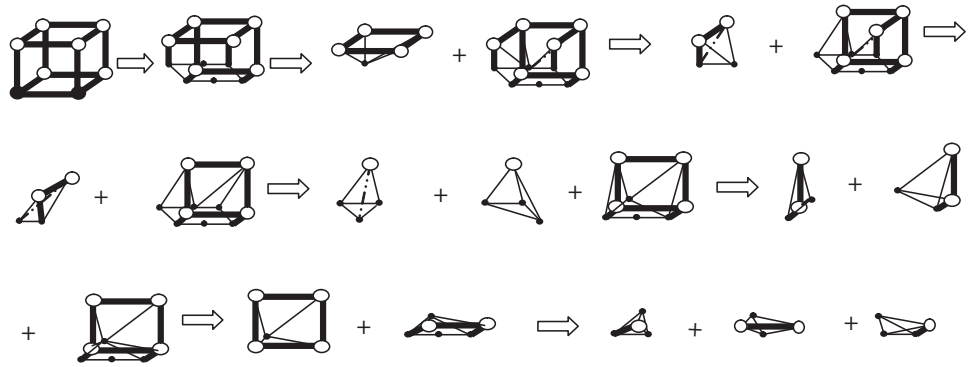


Figure 3.13: Volume mesh decomposition for Case 2C.

the tessellation process of Case 2. The final mesh elements for this case include five tetrahedra {2JTB, P2TB, 12TP, 1PID, 1TIP}, and the boundary surface is formed by four triangles {BJT, BTP, PTI, PID}. The tessellation process for Case 2C is demonstrated in Figure 3.13, where the final mesh elements include 9 tetrahedra {DP84, PB73, DPI8, PBJ7, IP58, PJ67, IT5P, T65P, TJ6P} and 2 pyramids {4387P, 5687P}, and the boundary surface is formed by four triangles {DIP, ITP, PJB, PTJ}.

The remaining cases are tessellated in a similar manner. The tessellation results for all 48 configurations can be organized into a two-level table, shown in Table 3.2 at the end of this chapter. The first level of the table contains entries for the 23 major cases (the 21 cases in Figure 3.2 plus the all black and all white cases). The second level contains the subcases for each ambiguous case. With this table, an automatic 3D mesh program can be easily implemented and a valid finite element mesh can be produced efficiently.

The pseudo code for the automatic volume mesh procedure is shown below.

```

Begin (process)
{
Read in 3D data
Divide the data set into small cubes of eight voxels each
Classify each cube vertex as inside or outside volume
(by comparing the vertex value with threshold)
For each cube
  If (all black)
    Continue
  Elseif (all white)
    The cube is a cube element
  Else
    Determine the major case
  If (the major case is ambiguous)
    Using the bilinear interpolation method to find the subcase
    Lookup surface triangles and volume mesh in the meshing table
Else
  Lookup surface triangles and volume mesh in the meshing table

```

```

    End
  End
End For
Export Boundary (isosurface)
Export Volume Mesh
}
End (process)

```

In the implementation of the algorithm, a rotation table (Table 3.3) is used to determine the major case for each cube configuration. We set the major cases listed in Figure 3.2 as standard forms, and the tessellation of the standard cube forms are defined in the Table 3.2. The tessellation for all equivalent configurations is the same as their standard form. Table 3.3 lists rotations needed for all the configurations. In the table, “Cube ID” is the identification number of a cube configuration which is the value of a binary number of the cube by setting white vertices to “1”s and black vertices to “0”s. For example, with the vertex notation introduced in Figure 3.10, if only the vertex 4 is white, the binary number will be “00001000”, and the value is 8 in decimal. The “Cube ID” of this configuration will be 8. Since this configuration is the same topology as the “Cube ID” of 1 which is the standard form, this configuration needs to be rotated to the standard form. The original vertex notation of all cube configurations is “87654321”, and the rotated vertex notation is “56218734” for the “Cube ID” of 8, which is listed in the “Rotation” column in Table 3.3. This table lists the rotations for all cube configurations. The corresponding major case is listed in the “Major Case” column, and it is Case 1 for this example. The column “Comp” shows whether a configuration has more than four white vertices. It is useful to indicate whether the cube is a complementary case in the original MC algorithm. It is “0” if the number of white vertices is more than 4, and it is “1” otherwise. The “Comp” of this example is 1.

3.3 Mesh Quality Improvement

Although the MVE algorithm can generate a geometrically and topologically consistent mesh, the resulting mesh may contain some poor quality elements, which can cause numerical difficulties in the solution process [23]. An ideal mesh for computational analysis is composed of all equilateral shaped elements, which is extremely difficult or impossible to obtain for complicated structures. In practice, mesh improvement algorithms are used to make a mesh as close to an ideal mesh as possible. Many algorithms have been developed to improve mesh quality in the literature. Points are inserted or deleted to improve the mesh quality [68]. The mesh quality is improved by local reconnection or face swapping [23, 42]. Grid points are relocated to improve mesh quality without changing mesh topology [8]. A method is introduced to find the optimal positions for each nodes [22]. All these mesh improve algorithms are proposed to improve the quality of tetrahedron mesh.

In MVE, only tetrahedron and pyramid elements need to be improved. The quality of cube elements do not need to be improved because they are only used to mesh cubes with all eight vertices inside the volume, and the vertices are at voxel locations in the data grid. Consequently, these elements have maximum quality because all edge angles are exactly 90 degrees and, if the data has been interpolated to isotropic resolution, all edge lengths are equal.

Two steps described below are used to improve the quality of tetrahedral and pyramidal elements. In the first step, the surface is improved by removing very small polygons (triangles and quadrilaterals) and very short edges. A minimum edge length is used for all surface polygons. If the lengths of all edges of a polygon are less than the threshold,

the polygon is deleted and replaced by the centroid of the polygon. If only one edge length is less than the threshold, the short edge is reduced to the middle point of the edge. The process improves the quality of surface polygons without losing geometric consistency. After this process, the volume elements which share the deleted elements are updated. Figure 3.14 shows an example in 2-D where the mesh quality in Figure 3.14(a) is improved by removing the small triangle, as shown in Figure 3.14(b)

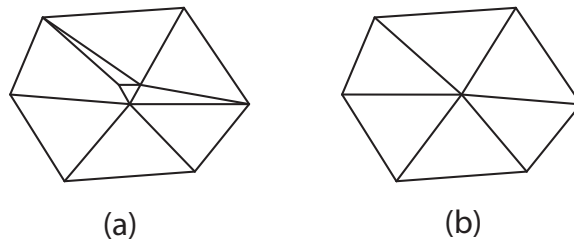


Figure 3.14: A 2-D example of mesh quality improvement: (a) before improvement. (b) after improvement.

The second step is to improve the quality of volume elements. A minimum edge length is used to eliminate the poor quality elements. Tetrahedra which have very short edges are eliminated, and the short edges shrink to points.

Special care has to be taken with tetrahedron and pyramid elements. If a tetrahedron to be removed shares a face with a pyramid or a triangle to be deleted is a face of a pyramid, the tetrahedron or the triangle can not be removed. If an edge to be removed is an edge of a pyramid and is an edge of the quadrilateral face of the pyramid, the pyramid will reduce to a tetrahedron. If an edge to be removed is an edge of a pyramid and is not an edge of the quadrilateral face of the pyramid, the edge will not be removed to keep topological consistency.

3.4 Experiments

In this section, the MVE algorithm is validated with two sets of experiments. In the first set, quantitative measures of mesh quality for the MVE mesh are calculated before and after mesh improvement and, where possible, compared with a mesh generated by GAMBIT (Fluent Inc., Lebanon, New Hampshire). GAMBIT is a commercially available software package that generates a tetrahedral mesh using Delaunay and advancing front method. GAMBIT cannot directly generate a mesh from 3D medical image data, so the isosurface generated by MVE algorithm was input to the GAMBIT volume mesh generation algorithm. In the second set of experiments, GAMBIT is not able to mesh the data set. Only mesh quality for the MVE mesh before and after mesh improvement were compared.

The MVE was used to mesh both a simulated object and a human nasal cavity from CT data. Six CT data sets of a mortar (a solid cylinder with a hemisphere-shaped region removed from the top) were simulated with resolutions of $16 \times 16 \times 8$, $32 \times 32 \times 16$, $64 \times 64 \times 32$, $128 \times 128 \times 64$, $192 \times 192 \times 96$ and $256 \times 256 \times 128$ voxels. A volume mesh was constructed from each data set with both the MVE algorithm and GAMBIT. Figure 3.15 shows the mortar mesh from the $16 \times 16 \times 8$ data set. The computation time for MVE is compared with GAMBIT in Figure 3.16. For all resolutions, MVE is faster than GAMBIT. In addition, GAMBIT was not able to mesh the two highest resolution data sets of the mortar and the human nasal cavity on a Sun Microsystems Ultra 80 with 2Gb memory and a Sparcv9+vis CPU.

To compare mesh quality, the edge ratio and the equi-angle skew of each element in the mesh are computed. The edge ratio is defined as the ratio of the longest edge to

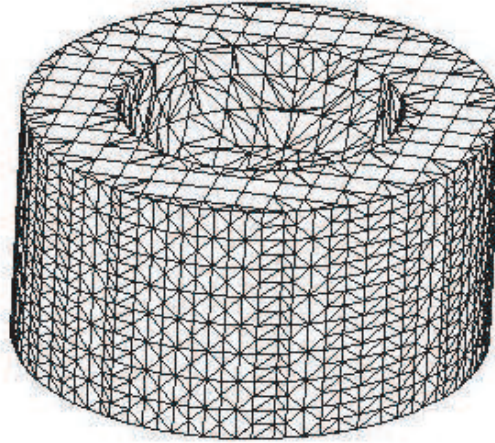


Figure 3.15: A 3-D surface mesh of a mortar generated by MVE from a 16x16x8 pixel simulated CT data set.

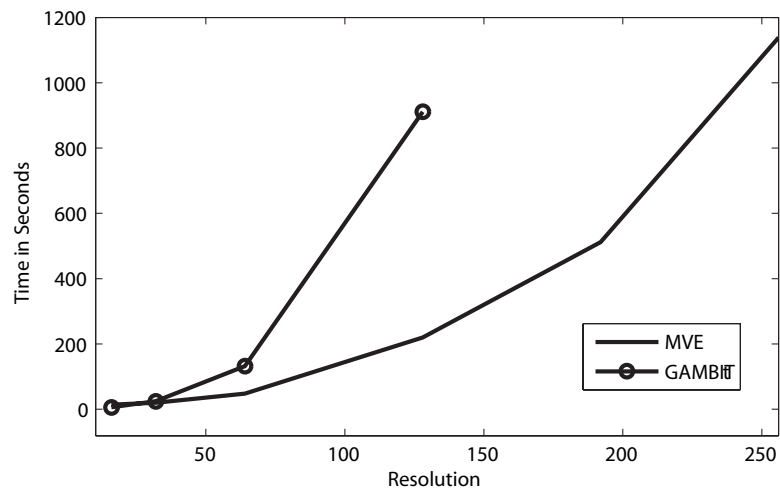


Figure 3.16: Mesh computation time versus resolution for both MVE (line only) and GAMBIT (line and circle). MVE algorithm is able to mesh all data sets, while GAMBIT can not mesh the two higher resolution data sets.

the shortest edge in an element. The edge ratio ranges from 1 to infinity. The highest quality element is an equilateral element, which has an edge ratio of 1. The equi-angle skew is defined as (3.5), where θ_{\max} and θ_{\min} are the maximum and minimum angles between the edges of the faces of an element, and θ_{eq} is the equi-angle value for the type of face, $\theta_{eq} = 60^\circ$ for a triangle and 90° for quadrilateral. By definition, the equi-angle skew ranges from 0 to 1. The equi-angle skew is 0 for an ideal element, and poor quality elements have values close to 1.

$$Q_{equi_angle_skew} = \max \left\{ \frac{\theta_{\max} - \theta_{eq}}{180 - \theta_{eq}}, \frac{\theta_{eq} - \theta_{\min}}{\theta_{eq}} \right\} \quad (3.5)$$

Figure 3.17 and 3.18 show histograms of the edge ratio and the equi-angle skew of the 128x128x64 mortar respectively. Figure 3.17(a) shows the MVE edge ratio before mesh improvement. Figure 3.17(b) shows the MVE edge ratio after mesh improvement. Figure 3.17(c) shows the edge ratio for the GAMBIT mesh. Figure 3.18(a) and (b) show the equi-angle skew of the MVE mesh before and after mesh improvement. Figure 3.18(c) shows the equi-angle skew of the GAMBIT mesh. Note that in Figure 3.17(b) and 3.18(b), most of poor quality elements, which have very high edge ratio and equi-angle skew close to one, have been removed. The GAMBIT mesh has a lower number of poor-quality elements, but most of its elements have an edge ratio around 1.5 and an equi-angle skew around 0.3. In contrast, most of the elements in the MVE mesh are cubes, which have ideal edge ratios and equi-angle skews. In addition, GAMBIT was not able to mesh the two data sets with higher resolution. For the 192x192x96 data set, GAMBIT ran several days without results. For the 256x256x128 data set, GAMBIT ran out of memory on the system described above.

The MVE algorithm was also used to mesh a left human nasal cavity from 31 CT slices. The CT data had a slice thickness of 3mm and an in-plane pixel size of 0.31mm x 0.31mm. Before running the MVE algorithm, five slices were interpolated between acquired slices using HSR introduced in Chapter 2. All slices (both acquired and interpolated) were decimated by a factor of 2 in both in-plane directions to obtain approximately isotropic resolution. Figure 3.19 shows the cavity surface. The mesh quality of human nasal cavity model is shown in Figure 3.20 and Figure 3.21. Figure 3.20(a) and (b) show the edge ratio histograms of mesh before and after the mesh improvement respectively. Figure 3.21(a) and (b) show the equi-angle skew histograms of the mesh before and after the mesh improvement respectively. The edge ratio was highly improved with large edge ratio elements removed. The equi-angle skew was also improved. Note that most poor equality elements whose equi-angle skews are close to one were removed.

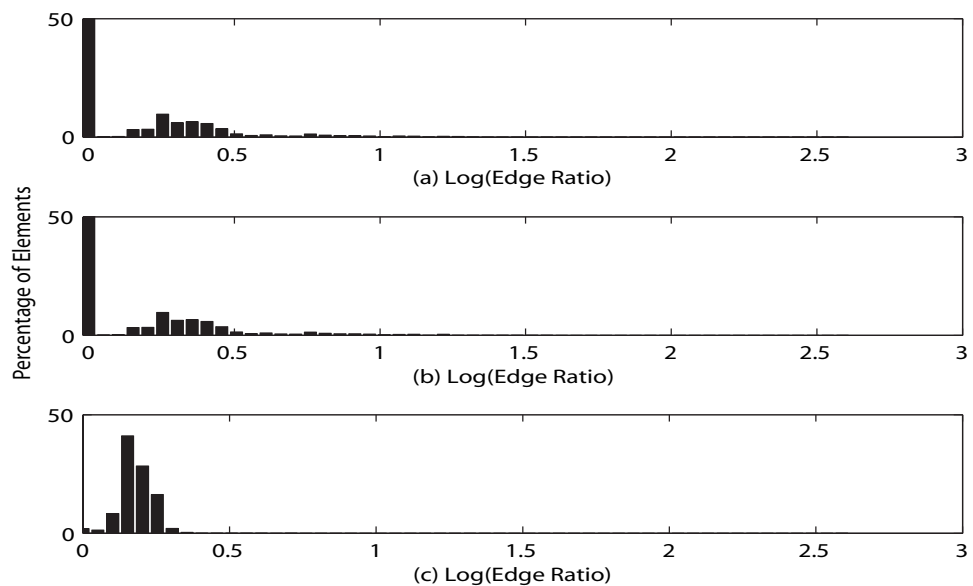


Figure 3.17: Edge ratio histograms for the 128x128x64 mortar mesh constructed by (a) MVE before improvement (b) MVE after improvement (c) GAMBIT. The MVE mesh contained 887,504 elements before improvement, 884,624 elements after improvement, and the GAMBIT mesh contained 3,134,488 elements.

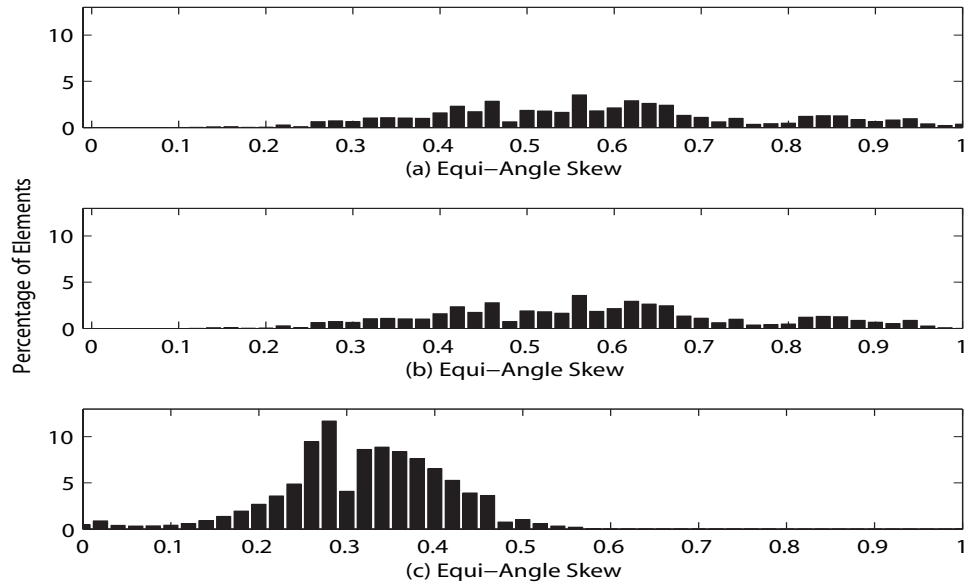


Figure 3.18: Equi-angle skew histograms for the 128x128x64 mortar mesh constructed by (a) MVE before improvement (b) MVE after improvement (c) GAMBIT. Note the vertical axes of plots (a) and (b) have been cropped to improve the detail in the higher equi-angle skew values. The bar at zero equi-angle skew extends to 50% in plot (a) and (b). The MVE mesh contained 887,504 elements before improvement, 884,624 elements after improvement, and the GAMBIT mesh contained 3,134,488 elements.

3.5 Conclusion

In this section, the new 3-D mesh generation algorithm, marching volume element(MVE), was presented. This algorithm is able to automatically construct a volume mesh from three-dimensional (3-D) data defined on a regularly-spaced grid. The resulting mesh is composed of tetrahedra, pyramids, and cubes. The MVE takes advantage of the inherent structure of gridded data by using cubes to mesh interior regions. Tetrahedra are used to mesh the boundary layer, and pyramids are used for transition between tetrahedra and cubes.



Figure 3.19: Right nasal cavity surface constructed with the MVE algorithm. The marked region is mainly contains olfactory epithelium.

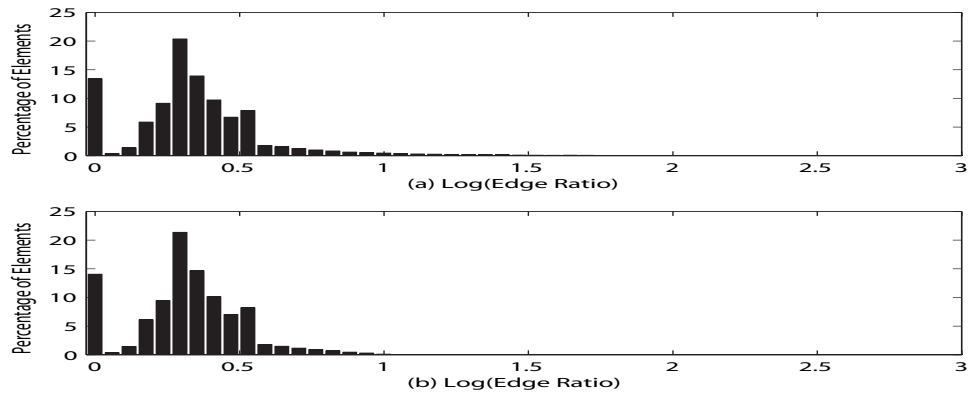


Figure 3.20: Edge ratio histograms for the nasal cavity model constructed with MVE before (a) and after (b) improvement.

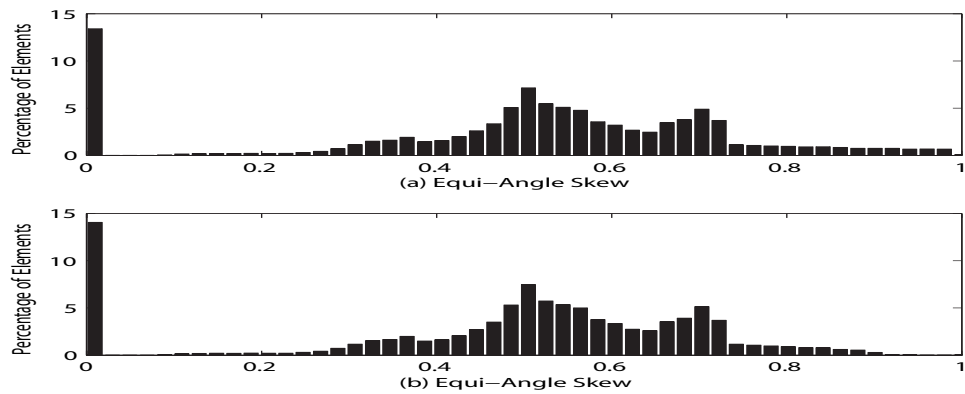


Figure 3.21: Equi-angle skew histograms for the nasal cavity model constructed with MVE before (a) and after (b) improvement.

The MVE algorithm operates locally on the data. Each cube is meshed individually in such a way that topological consistency with its neighbors is ensured. As a result, the MVE is memory efficient and can construct volume meshes from large data sets that may not be practical to mesh with other techniques. In addition, a relatively simple algorithm was used to improve the mesh quality.

The MVE algorithm is validated on a synthetic object and demonstrated on a computed tomography (CT) scan of a human nasal cavity. The volume mesh generated by the MVE algorithm is suitable for computational fluid dynamics simulations. In Chapter 4, air dynamics simulations will be conducted with human nasal cavity models constructed with the MVE algorithm.

Table 3.2: Mesh Table for Marching Volume Elements Algorithm

| CASE | SURFACE TRIANGLES | TETRAHEDRONS | PYRAMIDS |
|------|-------------------------|---|--|
| 1 | AID | 1AID | |
| 1C | DIA | A37D,I78D,DAI7,A6I7 | 4837D,2637A,5687I |
| 2 | BJT,BTP,PTI,PID | 2JTB,P2TB,12TP,1PID,1TIP | |
| 2C | DIP,ITP,PJB,PTJ | DP84,PB73,DPI8,PBJ7,IP58,PJ67,IT5P,T65P,TJ6P | 4387P,5687P |
| 3 | A | AID,CLB | 1AID,CBL3 |
| | B | IDL,DCL,ILA,BAL | 1DAI,BC3L,ADLI |
| 3C | A | ICD,ILC,IBL,IAB | DC84,A26B,DCI8,A6IB,ICL8,IB6L,IL58,I65L |
| | B | BLC,ADI | DC84,A26B,DCI8,A6IB,ICB8,I65B,IB58,BCL8,5BL8,B56L |
| 4 | AID,LGF | 1AID,LFG7 | |
| 4C | LFG,DIA | D3L4,A2L3,DAL3,A2FL,A26F,DAIL,AFIL,A6IF,DLG4,DG84,ILGD,IFGL,DGI8,IG58,IF5G,I65F | |
| 5 | QFR,HBR,BQR,IBH,IAB | Q6RF,A2QB,A26Q,IBRH,IR5H,IAQB,IBQR,IA6Q,I6RQ,I65R | |
| 5C | QRF,HRB,BRQ,IHB,IBA | 1A14,IAB4,IBH4,RQF7,BRH4,43BR,BQR3,RQ37,4RH8 | 4387R |
| 6 | A | LGF,JTB,BTP,PTD,DTI | LFG7,2JTB,P2TB,12TP,1TID,1PTD |
| | B | DGI,DLG,DPL,PBL,IGF,IFT,TFJ | LFG7,DLIG,ILFG,DTLP,DTIL,ITFL,PBTL,2JTB,P2TB,12TP,1TID,1TDP |
| 6C | A | BGL,BJG,JFG,ITD,TPD,PJB,TJP | DP84,PBL3,P43L,PLG4,PG84,PBLG,DTI8,DPT8,PBJG,PJTG,PGT8,GFJT,TJ6F,IT58,5G8T,5TFG,T65F |
| | B | LFG,TJB,TBP,TPD,TDI | DP84,PBL3,P3L4,PLG4,PG84,TBLP,TJ6F,DTI8,DPT8,TG8P,TLGP,TFGL,IT58,5TG8,5TFG,T65F |
| 7 | A | DKC,LGF,ABJ | DCK4,LFG7,A2JB |
| | B | LGF,CBK,BJK,DKA,AKJ | LFG7,DCK4,A2JB,ABJK |
| | C | OKC,OCB,OBL,OLG,OGF,OFJ,OJA,OAD,ODK | CDK4,A2JB,DCOK,ABJO,LFG7,OLFG |
| | D | CBL,KAD,KJA,KGJ,GFJ | DCK4,A2JB,LFG7,AJKB,KJGB |
| 7C | A | CLB,DAK,KAJ,JGK,GJF | CBL3,1A5D,DA5K,5AJK,5G8K,5J6F,5JGK,5JFG, |
| | B | ODC,OCL,OLB,OBA,OAJ,OJF,OFG,OGK,OKD | CBL3,CBOL,1A5D,DO5K,DA5O,AJ5O,5JFO,5J6F,KG58,5OGK,5FGO |
| | C | OKD,ODC,OCL,OLF,OFG,OGK,AJB | CBL3,CBOL,ABJO,DO5K,1A5D,DA5K,5AJO,5JFO,5J6F,KG58,5OGK,5OFG |
| | D | DCK,AJB,FGL | CBL3,1A5D,D5KC,AJ5B,5J6F,5LFG,5G8K,5BLC |
| 8 | DPS,PBQ,QFR,SRH,SPR,PQR | 1PSD,5RHS,QF6R,P2QB,26RQ,PRQ2,SRP1,12RP | 1256R |
| 9 | DGK,DAG,AFG,AJF | 1A5D,J65F,5G8K,5GKD,DA5G,5AFG,5AJF | |

| CASE | | SURFACE RIANGLES | TETRAHEDRONS | PYRAMIDS |
|------|---|--|--|-------------------------------------|
| 10 | A | SKC,PSC,LGQ,GRQ,AIS,ASP,QRE,EJQ | 4CSK,4PSC,Q7GL,RQ7G,1PS4,Q6R7,1AIS,1PAS,E6RQ,J6EQ | |
| | B | AIS,ASP,QRE,EJQ,SRQ,SQP,CPQ,CQL,SGK,SRG | SPQC,SCQK,SQ GK,SQRG,4CSK,4PSC,Q7GL,RQ7G,1PS4,Q6R7,1AIS,1PAS,E6RQ,J6EQ | CLKGQ |
| | C | CQL,CPQ,KGS,SGR,AQP,AJQ,SRE,SEI | 4CSK,4PSC,GQ7L,1PS4,RQ7G,SPQC,SQKC,SQ GK,SQRG,Q6R7,1ASP,1AIS,J6EQ,EQ6R,SAQP,IAQS,SERQ,IESQ | CLKGQ,AJIEQ |
| 11 | | HLG,HAL,SAH,DAS,AQL,AJQ | 1ASD,1A5S,SA5H,GQ7L,HQGL,AQHL,5AQH,5AJQ,5J6Q,5Q6H,H6GQ,GQ67 | |
| 12 | A | DKC,RQF,HQR,HBQ,HIB,ABI | DCK4,Q6RF,A2QB,A26Q,5RHI,I65R,IBQH,IQRH,IAQB,IA6Q,I6RQ | |
| | B | DKA,AKF,AFL,IFR,IRH,CFK,CQF,CBQ | DCK4,BQKC,QFKC,AQKB,AFKQ,A2QB,A26Q,A6FQ,A6IF,IR5H,I65R,I6RF | ABDCK |
| | C | RQF,ADI,BHC,CHK,HQR,HBQ | DCK4,Q6RF,IBHC,A2QB,A26Q,IR5H,I65R,IBQH,IQRH,IAQB,IA6Q,I6RQ | DIKHC,ABDCI |
| | D | FRK,RHK,FKC,DIC,IFC,IAF,AQF,ABQ | DCK4,HCRK,CRFK,IRCH,IFCR,A2QB,A26Q,A6FQ,A6IF,IR5H,I65R,I6RF | DIKHC |
| 13 | A | CKD,AJB,EIH,FGL | 1AID,CBL3,6JFE,8HGK,HBLC,HIBC,EBLH,IBEH | DIKHC,ABDCI,CKLGH,EFHGL,AJIEB,BJLFE |
| | B | EIH,FGL,BKJ,BCK,JKD,AJD | 1AID,3LBC,6JFE,8HGK,DJEK,EJBK,KBLG,KBEL,HLGK,HELK | AJIED,DIKHE,BJLFE,CLKGB,EFHGL, |
| | C | OAJ,OJB,OBC,OCK,OKD,ODA,OEI,OIH,OHG,OGI,OLF,OFE | 1AID,3LBC,6JFE,8HGK,DAIO,EJFO,HOGK,OBLC | DIKHO,AJIEO,BJLFO,CLKGO |
| | D | OCK,OKD,ODA,OAJ,OJF,OFG,OGI,OLB,OBC,EIH | 1AID,3LBC,6JFE,8HGK,DAIO,OBLC,EJFO,IEHO,HOGK | DIKHO,AJIEO,CLKGO,EFHGO |
| | E | OEI,OID,ODA,OAJ,OJF,OFG,OGI,OLB,OBC,OCK,OKH,OHE | 1AID,3LBC,6JFE,8HGK,DAIO,OBLC,EJFO,HOGK | AJIEO,CLKGO,EFHGO |
| | F | OBC,OCK,OKD,ODA,OAI,OIH,OHE,OEJ,OJF,OFG,OGI,OLB | 1AID,3LBC,6JFE,8HGK,DAIO,OBLC,EJFO,HOGK | DIKHO,EFHGO,CLKGO |
| | G | OKD,ODA,OAI,OIH,OHE,OEJ,OJB,OBC,OCL,OLF,OFG,OGK | 1AID,BLC3,DAIO,OBLC,EJFO,EJ6F,KGH8,HOGK | DIKHO,BJLFO,EFHGO |
| | H | OCL,OLB,OBA,OAI,OID,ODC,OGK,OKH,OHE,OEJ,OJF,OFG, | 1AID,3LBC,6JFE,8HGK,DAIO,OBLC,EJFO,HOGK | ABDCO, BJLFO |
| | I | HGK,OID,ODC,OCL,OLF,OFE,OEJ,OJB,OBA,OAI, | 1AID,3LBC,6JFE,8HGK,DAIO,OBLC,EJFO | ABDCI |
| | J | HGK,EJF,CID,CLI,LBI,BAI | 1AID,3LBC,6JFE,8HGK,CLBI | |
| | K | AID,HGK,BCL,EJF | 1AID,3LBC,6JFE,8HGK | |
| 14 | | IAS,SAK,ABK,KBG,BQG,QFG | KBSG,SB8K,SABK,SB58,ISB5,IABS,A2QB,5BG8,A26Q,5BQG,5QFG,56FQ,IBQ5,IAQB,I65Q,IA6Q, | |

Table 3.3: Cube Rotation Table

| Cube ID | Rotation | Major Case | Comp | | Cube ID | Rotation | Major Case | Comp |
|---------|----------|------------|------|--|---------|----------|------------|------|
| 0 | 87654321 | 0 | 1 | | 40 | 76583214 | 4 | 1 |
| 1 | 87654321 | 1 | 1 | | 41 | 76583214 | 6 | 1 |
| 2 | 48513762 | 1 | 1 | | 42 | 84375126 | 6 | 1 |
| 3 | 87654321 | 2 | 1 | | 43 | 56218734 | 11 | 1 |
| 4 | 15624873 | 1 | 1 | | 44 | 56218734 | 6 | 1 |
| 5 | 87654321 | 3 | 1 | | 45 | 78436512 | 12 | 1 |
| 6 | 85147623 | 2 | 1 | | 46 | 67325841 | 14 | 1 |
| 7 | 56218734 | 5 | 1 | | 47 | 14852376 | 5 | 0 |
| 8 | 56218734 | 1 | 1 | | 48 | 43781265 | 2 | 1 |
| 9 | 76583214 | 2 | 1 | | 49 | 48513762 | 5 | 1 |
| 10 | 76583214 | 3 | 1 | | 50 | 87654321 | 5 | 1 |
| 11 | 85147623 | 5 | 1 | | 51 | 87654321 | 8 | 1 |
| 12 | 56218734 | 2 | 1 | | 52 | 43781265 | 6 | 1 |
| 13 | 78436512 | 5 | 1 | | 53 | 48513762 | 12 | 1 |
| 14 | 67325841 | 5 | 1 | | 54 | 73268415 | 11 | 1 |
| 15 | 56218734 | 8 | 1 | | 55 | 65872143 | 5 | 0 |
| 16 | 23416785 | 1 | 1 | | 56 | 34127856 | 6 | 1 |
| 17 | 73268415 | 2 | 1 | | 57 | 62157348 | 14 | 1 |
| 18 | 78436512 | 3 | 1 | | 58 | 87654321 | 12 | 1 |
| 19 | 34127856 | 5 | 1 | | 59 | 26731584 | 5 | 0 |
| 20 | 15624873 | 4 | 1 | | 60 | 15624873 | 10 | 1 |
| 21 | 73268415 | 6 | 1 | | 61 | 12345678 | 6 | 0 |
| 22 | 85147623 | 6 | 1 | | 62 | 21563487 | 6 | 0 |
| 23 | 56218734 | 14 | 1 | | 63 | 12345678 | 2 | 0 |
| 24 | 26731584 | 3 | 1 | | 64 | 41238567 | 1 | 1 |
| 25 | 62157348 | 5 | 1 | | 65 | 87654321 | 4 | 1 |
| 26 | 32674158 | 7 | 1 | | 66 | 48513762 | 3 | 1 |
| 27 | 48513762 | 9 | 1 | | 67 | 87654321 | 6 | 1 |
| 28 | 65872143 | 6 | 1 | | 68 | 15624873 | 2 | 1 |
| 29 | 23416785 | 11 | 1 | | 69 | 51486237 | 6 | 1 |
| 30 | 67325841 | 12 | 1 | | 70 | 84375126 | 5 | 1 |
| 31 | 43781265 | 5 | 0 | | 71 | 67325841 | 11 | 1 |
| 32 | 34127856 | 1 | 1 | | 72 | 56218734 | 3 | 1 |
| 33 | 84375126 | 3 | 1 | | 73 | 67325841 | 6 | 1 |
| 34 | 84375126 | 2 | 1 | | 74 | 87654321 | 7 | 1 |
| 35 | 73268415 | 5 | 1 | | 75 | 85147623 | 12 | 1 |
| 36 | 85147623 | 3 | 1 | | 76 | 12345678 | 5 | 1 |
| 37 | 76583214 | 7 | 1 | | 77 | 78436512 | 14 | 1 |
| 38 | 41238567 | 5 | 1 | | 78 | 78436512 | 9 | 1 |
| 39 | 67325841 | 9 | 1 | | 79 | 21563487 | 5 | 0 |

| Cube ID | Rotation | Major Case | Comp | | Cube ID | Rotation | Major Case | Comp |
|---------|----------|------------|------|--|---------|----------|------------|------|
| 80 | 23416785 | 3 | 1 | | 120 | 32674158 | 12 | 1 |
| 81 | 37842651 | 6 | 1 | | 121 | 58761432 | 6 | 0 |
| 82 | 12345678 | 7 | 1 | | 122 | 73268415 | 7 | 0 |
| 83 | 34127856 | 12 | 1 | | 123 | 62157348 | 3 | 0 |
| 84 | 15624873 | 6 | 1 | | 124 | 78436512 | 6 | 0 |
| 85 | 43781265 | 10 | 1 | | 125 | 78436512 | 4 | 0 |
| 86 | 84375126 | 12 | 1 | | 126 | 67325841 | 3 | 0 |
| 87 | 26731584 | 6 | 0 | | 127 | 62157348 | 1 | 0 |
| 88 | 15624873 | 7 | 1 | | 128 | 62157348 | 1 | 1 |
| 89 | 62157348 | 12 | 1 | | 129 | 67325841 | 3 | 1 |
| 90 | 58761432 | 13 | 1 | | 130 | 78436512 | 4 | 1 |
| 91 | 48513762 | 7 | 0 | | 131 | 78436512 | 6 | 1 |
| 92 | 12345678 | 12 | 1 | | 132 | 62157348 | 3 | 1 |
| 93 | 48513762 | 6 | 0 | | 133 | 73268415 | 7 | 1 |
| 94 | 41238567 | 7 | 0 | | 134 | 58761432 | 6 | 1 |
| 95 | 12345678 | 3 | 0 | | 135 | 56218734 | 12 | 1 |
| 96 | 41238567 | 2 | 1 | | 136 | 62157348 | 2 | 1 |
| 97 | 41238567 | 6 | 1 | | 137 | 23416785 | 5 | 1 |
| 98 | 15624873 | 5 | 1 | | 138 | 62157348 | 6 | 1 |
| 99 | 73268415 | 14 | 1 | | 139 | 85147623 | 14 | 1 |
| 100 | 58761432 | 5 | 1 | | 140 | 51486237 | 5 | 1 |
| 101 | 58761432 | 12 | 1 | | 141 | 85147623 | 9 | 1 |
| 102 | 58761432 | 8 | 1 | | 142 | 78436512 | 11 | 1 |
| 103 | 37842651 | 5 | 0 | | 143 | 32674158 | 5 | 0 |
| 104 | 14852376 | 6 | 1 | | 144 | 23416785 | 2 | 1 |
| 105 | 87654321 | 10 | 1 | | 145 | 76583214 | 5 | 1 |
| 106 | 15624873 | 12 | 1 | | 146 | 32674158 | 6 | 1 |
| 107 | 23416785 | 6 | 0 | | 147 | 48513762 | 11 | 1 |
| 108 | 26731584 | 11 | 1 | | 148 | 23416785 | 6 | 1 |
| 109 | 32674158 | 6 | 0 | | 149 | 76583214 | 12 | 1 |
| 110 | 76583214 | 5 | 0 | | 150 | 48513762 | 10 | 1 |
| 111 | 23416785 | 2 | 0 | | 151 | 14852376 | 6 | 0 |
| 112 | 32674158 | 5 | 1 | | 152 | 37842651 | 5 | 1 |
| 113 | 87654321 | 11 | 1 | | 153 | 76583214 | 8 | 1 |
| 114 | 21563487 | 9 | 1 | | 154 | 37842651 | 12 | 1 |
| 115 | 51486237 | 5 | 0 | | 155 | 58761432 | 5 | 0 |
| 116 | 32674158 | 14 | 1 | | 156 | 51486237 | 14 | 1 |
| 117 | 62157348 | 6 | 0 | | 157 | 15624873 | 5 | 0 |
| 118 | 23416785 | 5 | 0 | | 158 | 41238567 | 6 | 0 |
| 119 | 62157348 | 2 | 0 | | 159 | 41238567 | 2 | 0 |

| CubeID | Rotation | Major Case | Comp | Cube ID | Rotation | Major Case | Comp |
|--------|----------|------------|------|---------|----------|------------|------|
| 160 | 12345678 | 3 | 1 | 200 | 65872143 | 5 | 1 |
| 161 | 41238567 | 7 | 1 | 201 | 37842651 | 11 | 1 |
| 162 | 48513762 | 6 | 1 | 202 | 65872143 | 12 | 1 |
| 163 | 73268415 | 12 | 1 | 203 | 43781265 | 6 | 0 |
| 164 | 48513762 | 7 | 1 | 204 | 12345678 | 8 | 1 |
| 165 | 87654321 | 13 | 1 | 205 | 87654321 | 5 | 0 |
| 166 | 41238567 | 12 | 1 | 206 | 48513762 | 5 | 0 |
| 167 | 15624873 | 7 | 0 | 207 | 43781265 | 2 | 0 |
| 168 | 26731584 | 6 | 1 | 208 | 14852376 | 5 | 1 |
| 169 | 23416785 | 12 | 1 | 209 | 14852376 | 14 | 1 |
| 170 | 14852376 | 10 | 1 | 210 | 14852376 | 12 | 1 |
| 171 | 15624873 | 6 | 0 | 211 | 56218734 | 6 | 0 |
| 172 | 51486237 | 12 | 1 | 212 | 43781265 | 11 | 1 |
| 173 | 12345678 | 7 | 0 | 213 | 84375126 | 6 | 0 |
| 174 | 37842651 | 6 | 0 | 214 | 76583214 | 6 | 0 |
| 175 | 23416785 | 3 | 0 | 215 | 76583214 | 4 | 0 |
| 176 | 21563487 | 5 | 1 | 216 | 43781265 | 9 | 1 |
| 177 | 87654321 | 9 | 1 | 217 | 41238567 | 5 | 0 |
| 178 | 87654321 | 14 | 1 | 218 | 76583214 | 7 | 0 |
| 179 | 12345678 | 5 | 0 | 219 | 85147623 | 3 | 0 |
| 180 | 21563487 | 12 | 1 | 220 | 73268415 | 5 | 0 |
| 181 | 87654321 | 7 | 0 | 221 | 84375126 | 2 | 0 |
| 182 | 67325841 | 6 | 0 | 222 | 84375126 | 3 | 0 |
| 183 | 56218734 | 3 | 0 | 223 | 34127856 | 1 | 0 |
| 184 | 14852376 | 11 | 1 | 224 | 43781265 | 5 | 1 |
| 185 | 84375126 | 5 | 0 | 225 | 43781265 | 12 | 1 |
| 186 | 51486237 | 6 | 0 | 226 | 58761432 | 11 | 1 |
| 187 | 15624873 | 2 | 0 | 227 | 65872143 | 6 | 0 |
| 188 | 87654321 | 6 | 0 | 228 | 32674158 | 9 | 1 |
| 189 | 48513762 | 3 | 0 | 229 | 32674158 | 7 | 0 |
| 190 | 87654321 | 4 | 0 | 230 | 62157348 | 5 | 0 |
| 191 | 41238567 | 1 | 0 | 231 | 26731584 | 3 | 0 |
| 192 | 12345678 | 2 | 1 | 232 | 43781265 | 14 | 1 |
| 193 | 21563487 | 6 | 1 | 233 | 85147623 | 6 | 0 |
| 194 | 12345678 | 6 | 1 | 234 | 73268415 | 6 | 0 |
| 195 | 58761432 | 10 | 1 | 235 | 15624873 | 4 | 0 |
| 196 | 26731584 | 5 | 1 | 236 | 34127856 | 5 | 0 |
| 197 | 26731584 | 12 | 1 | 237 | 78436512 | 3 | 0 |
| 198 | 84375126 | 14 | 1 | 238 | 73268415 | 2 | 0 |
| 199 | 34127856 | 6 | 0 | 239 | 23416785 | 1 | 0 |

| CubeID | Rotation | Major Case | Comp |
|--------|----------|------------|------|
| 240 | 43781265 | 8 | 1 |
| 241 | 67325841 | 5 | 0 |
| 242 | 78436512 | 5 | 0 |
| 243 | 56218734 | 2 | 0 |
| 244 | 85147623 | 5 | 0 |
| 245 | 76583214 | 3 | 0 |
| 246 | 76583214 | 2 | 0 |
| 247 | 56218734 | 1 | 0 |
| 248 | 56218734 | 5 | 0 |
| 249 | 85147623 | 2 | 0 |
| 250 | 87654321 | 3 | 0 |
| 251 | 15624873 | 1 | 0 |
| 252 | 87654321 | 2 | 0 |
| 253 | 48513762 | 1 | 0 |
| 254 | 87654321 | 1 | 0 |
| 255 | 87654321 | 0 | 0 |

CHAPTER 4
NUMERICAL AIRFLOW SIMULATION

4.1 Introduction

The study of air flow in human nasal cavities has been an interesting topic for many years. Experimental models and numerical models have been developed from human cadavers [59, 72, 73, 87, 29, 39], CT [31] or MRI [79]. As discussed in Chapter 1, numerical models have many advantages over experimental models. Human nasal cavities are very complex. Figure 4.1 shows a typical CT slice of human nasal cavity at the posterior region. It is very difficult to simplify or idealize such a complex structure. Obviously 2-D models can not represent the 3-D complex structure, and manual construction of 3-D nasal cavity models would be tedious and error prone. Dog nasal cavities are much more complex than human's. A CT slice of dog nasal cavities at the olfactory region is shown in Figure 4.2. The dog nasal cavities can not be simplified as a 2-D model as well. An accurate 3-D model of dog nasal cavities needs to be constructed for air flow simulations.

In this research, geometrical models of human and dog nasal cavities were constructed, and air flow was simulated in models with computational fluid dynamics (CFD). The MVE algorithm developed in Chapter 3 was used to develop nasal cavity models. The MVE algorithm was validated on human subjects first. Then it was used to develop a numerical model of dog nasal cavities. Six human models were constructed from CT scans of six human volunteers and steady state air flow simulations were conducted on the models. A comparison study was performed. To our knowledge, this is the largest



Figure 4.1: A cropped CT slice of human nasal cavities

number of subjects used in a study of air flow through human nasal cavities. After the MVE algorithm was successfully applied to model human nasal cavities, a dog nasal cavity model was constructed, and a steady state air flow simulation was performed with the model. For human studies, the results were consistent with other researchers' work for similar anatomic structures. The same technique was used to construct a dog nasal cavity model and study steady state air flow in the nasal cavity model. This is the first numerical air flow simulation study on a dog to our knowledge. The validation of the results are pending until an experimental model is developed.

4.2 Simulation Methods

To develop numerical models from CT data, the following basic steps are needed:

- 1) The CT image data is interpolated in the slice direction to match slice thickness with the imaging plane resolution. The HSR algorithm was introduced in Chapter 2.
- 2) The

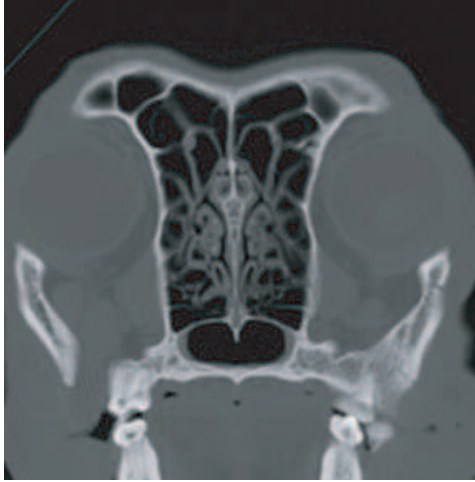


Figure 4.2: A cropped CT slice of dog nasal cavities

nasal airway is segmented using a region-growing algorithm. 3) Input the segmented airway data into the MVE algorithm to generate a 3-D volume mesh. 4) Elements in the mesh are adjusted or deleted to improve mesh quality. Finally, the constructed computational meshes were used to simulate air flow with FiDAP (Fluent, Inc. Lebanon, New Hampshire). FiDAP is a commercially-available computational fluid dynamics (CFD) software package. In the above steps, human interaction is needed in segmentation step to correct the region-growing algorithm near narrow boundaries.

4.2.1 Image Acquisition

The nasal cavities of six human subjects and one dog subject were scanned with CT. The CT slices of human were acquired by an Picker CT scanner, while the CT slices of the dog were acquired by a GE 9800 CT scanner. For human scans, the slice thickness was 3mm, and the in-plane resolution was either $0.27\text{ mm}\times 0.27\text{ mm}$ (human studies 1, 5, and 6) or $0.31\text{ mm}\times 0.31\text{ mm}$ (human studies 2, 3, and 4). The dog scan has a slice thickness of 1.5 mm, and the in-plane resolution was $0.49\text{ mm}\times 0.49\text{ mm}$.

Table 4.1 summarizes these information. In addition, the number of CT slices used to construct models are listed in Table 4.1. The CT scans of human studies 1, 2 and 5 were incomplete, because naris tip sections of the studies were missing.

Table 4.1: Resolution of CT Scans and Slice Numbers

| Subject | Slice Thickness | Resolution | Number of Slices |
|---------|-----------------|--------------------------|------------------|
| Human 1 | 3 mm | 0.27 mm \times 0.27 mm | 22 |
| Human 2 | 3 mm | 0.31 mm \times 0.31 mm | 24 |
| Human3 | 3 mm | 0.31 mm \times 0.31 mm | 27 |
| Human 4 | 3 mm | 0.31 mm \times 0.31 mm | 28 |
| Human 5 | 3 mm | 0.27 mm \times 0.27 mm | 23 |
| Human 6 | 3 mm | 0.27 mm \times 0.27 mm | 31 |
| dog | 1.5 mm | 0.49 mm \times 0.49 mm | 82 |

4.2.2 Image Interpolation and Segmentation

In imaging modalities such as CT and MRI, a 3D object is scanned with a sequence of thick slices, where slice thickness is usually three or more times of the in-plane resolution. To obtain a data set with approximately isotropic resolution, slices were interpolated between acquired slices using the hierarchical spline interpolation (HSI) algorithm introduced in Chapter 2.

The interpolated image data was then segmented with a region-growing algorithm to extract the airway. The region-growing algorithm uses the 26-connectivity test. The region-growing algorithm started from one or more manually placed seed points within the airway lumen. A threshold value compared with surrounding 26 pixels of a seed point in the volume data, if any point in the 26 connections was less than the threshold, the point was added into the airway. The growing algorithm continued from the new added points. The region-growing algorithm stopped when the airway failed to increase

in size. In some cases, particular in narrow airway regions, the region-growing algorithm steps over the airway and grows into the wrong region. In these cases, user interaction is required to correct segmentation errors.

Threshold Selection

To segment the airway from medical image slices, a threshold is needed. The segmentation result is determined by the selection of the threshold. Since CT data is quantitative data in HU (Hounsfield Units), we can determine the composition of a pixel based on the HU value of that pixel. For example: air has the lowest value of -1000 HU, water is 0 HU, tissues are between air and water, and bones have higher HU values. We are interested in segmentation of the air way. If the CT data is perfect, we can choose a threshold of -1000HU and finish the segmentation. However, we are not able to obtain ideal CT slices in reality. Noise is always present. Noise may be introduced by many factors, such as device limitations, image reconstruction algorithms, and so on. To solve this problem, the histogram of the CT scan was used to select the optimal threshold for each study. The optimal threshold is chosen as a local minimum near -1000 HU on each histogram of the CT scan. The threshold values of -750, -850, -700, -700, -820 and -720 HU were chosen for human study 1 to 6 respectively. The CT data of the dog study was not calibrated. A threshold of 113 was chosen in the segmentation. It should be pointed out that there is a common problem, called the partial volume effect (PVE), associated with threshold-based segmentation methods. The PVE is a phenomenon existing at tissue boundaries due to volume averaging. The PVE often results in misclassification of tissue near the boundary to a different type from those of neighboring tissues. But the

PVE is not a severe problem in this research, since the difference between HU values of tissue and air is large.

4.2.3 Geometry Modeling and Mesh Generation

Once the nasal airway was segmented, the nasal cavity surface was extracted with the marching volume element (MVE) algorithm [99]. The MVE algorithm is able to extract surface and generate volume mesh for an object defined by 3-D gridded data. The surface is presented by triangles, while the volume mesh is composed of tetrahedra, pyramids, and cuboids. After the mesh was constructed, it was run through a mesh quality improvement algorithm, see Section 3.3 for details about mesh quality improvement.

Six models of right side nasal cavities were constructed from six CT scans of healthy human volunteers. The following steps were performed in the model construction. First, 5 intermediate slices were interpolated between each two CT slices with the HSI algorithm discussed above. Then, the image data were decimated by 2 in plane to match the in-plane resolution with the inter-plane resolution. This operation also reduces the number of the final elements and the effects of noise as well. After decimation, the 3D data was rotated by 30 degrees in sagittal plane with respect to normal direction of the sagittal plane. This operation is needed due to the medical hardware limitation, since CT slices of nose were acquired at about 30 degrees from normal standing-up position. After rotation, the resulting image data was segmented with the region-growing algorithm discussed in the previous section. Finally, the segmented data was meshed with the MVE algorithm. The number of elements and the node numbers of the six cavity meshes are listed in Table 4.2. Figure 4.5 shows right side of human nasal cavity surface of study 4 developed from 28 coronal CT slices. The marked region is the olfactory region which is

Table 4.2: The Number of Elements for Each Model for Human Studies 1-6 and the Dog Study

| Study | 1 | 2 | 3 | 4 | 5 | 6 | dog |
|------------|--------|--------|--------|--------|--------|--------|--------|
| Tetrahedra | 484962 | 299408 | 485915 | 404109 | 349134 | 315216 | 855480 |
| Pyramids | 70773 | 41840 | 71384 | 54648 | 47816 | 42778 | 95088 |
| Cubes | 157291 | 40724 | 115141 | 88082 | 59764 | 60683 | 54034 |
| Nodes | 333595 | 149894 | 290799 | 236641 | 187720 | 176188 | 356784 |

covered by olfactory epithelium. The simulation will be presented on the cross-section planes indicated as numbered lines (from 1 to 6) in this figure.

Following the same procedure, a dog nasal cavity model was developed from coronal CT scan of a dog. Only 70 of 82 coronal CT slices were used to construct the numerical model, since it is difficult to model the nasal cavity from the first 12 CT slices. Figure 4.3 shows every other slice of the first 12 slices, there are no enclosed air way boundaries on most of these slices. This makes it difficult to determine the boundary locations and set the boundary conditions, so the first 12 slices were removed in the modeling process. Figure 4.4 shows the dog nasal cavity surface reconstructed from 70 slices. Compared to surface of the human nasal cavity in Figure 4.5, dog's nasal cavities are much more complex. Different colors are used on the surface to indicate different regions in the dog nasal cavity. The red regions are the sinuses, the magenta regions are the nasal cavity airways, and the blue region is the olfactory region. To our knowledge, this is the first realistic 3-D surface image of a dog nasal cavity.

4.2.4 Governing Equations and Boundary Conditions

Steady state air flow was simulated in both human and dog nasal cavities. From other researchers' work, the air flow in human nasal cavities is steady or quasi-steady state flow for low flow rates [43, 96, 14]. The air flow in dog nasal cavities may be

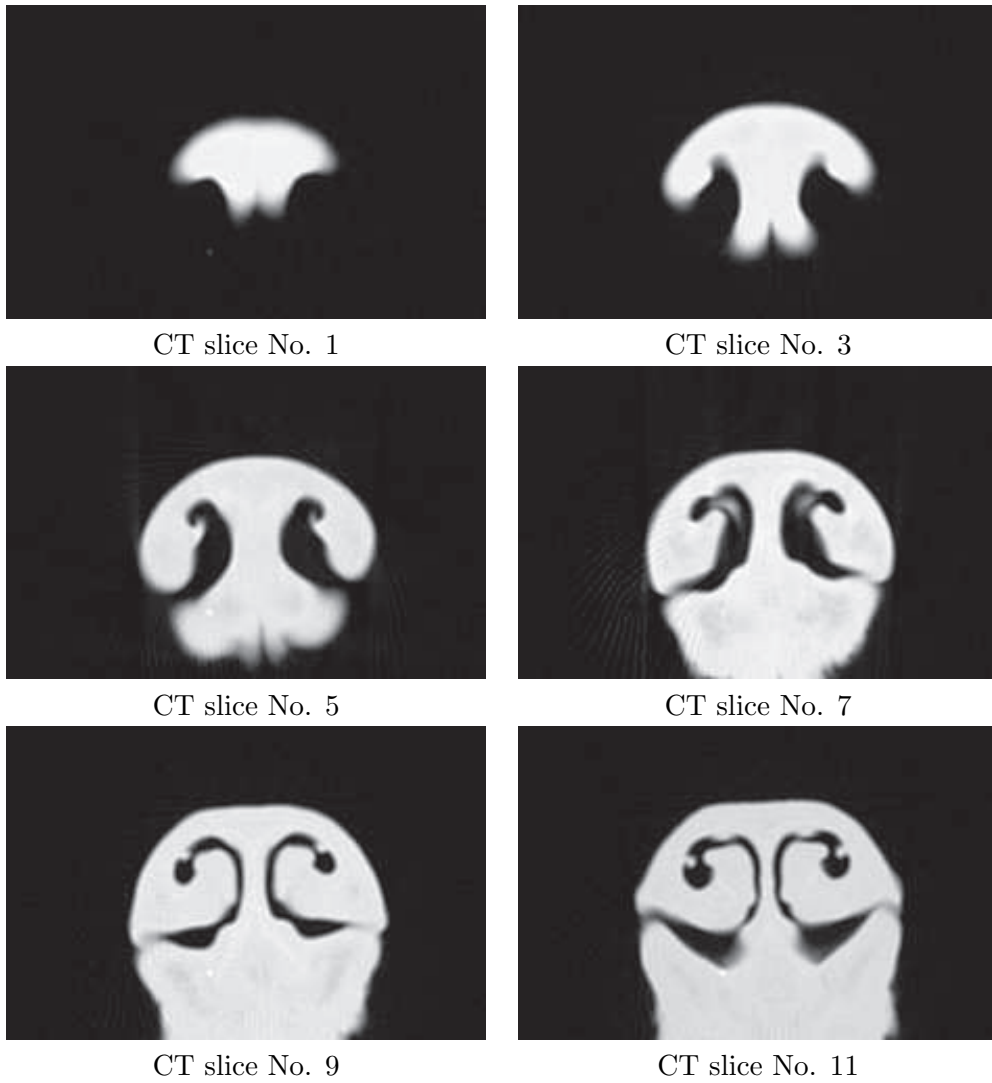


Figure 4.3: Six CT slices of the dog scans

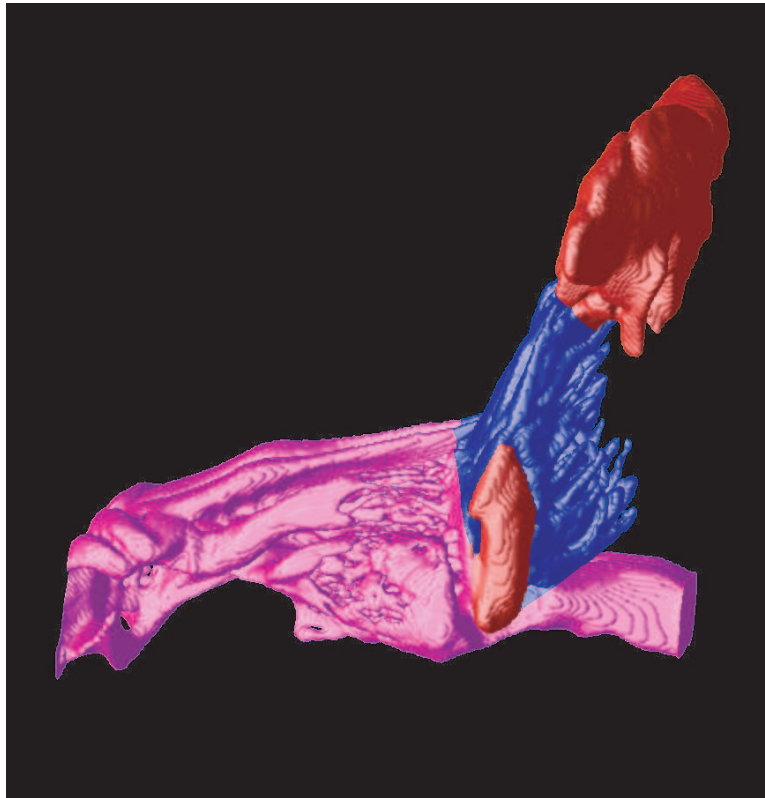


Figure 4.4: Dog nasal cavity surface reconstructed from 70 CT slices. Underneath the surface is the volume mesh. Red colored region are the sinues, magenta colored regions are the nasal cavity airways, and the blue colored region is the olfactory region.

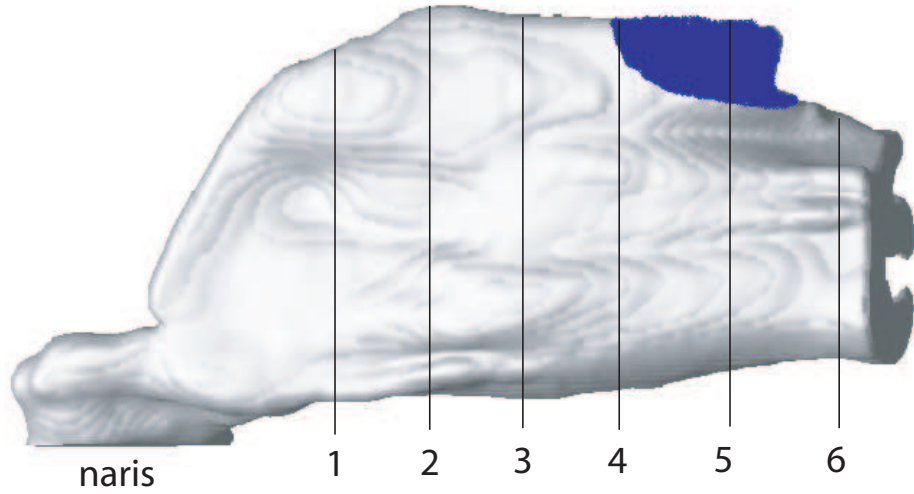


Figure 4.5: Right nasal cavity surface of study 4. The blue region is mainly covered with olfactory epithelium. The numbered lines indicate the position of planes where simulation results are displayed.

turbulent. Since many parameters for turbulent flow simulations are not available at this time, only steady state air flow is simulated in current study. The parameters for turbulent air flow in dog nasal cavity have to be obtained from experiments or physical models. Once the parameters are available, turbulent air flow may be simulated with FiDAP (Fluent, Inc. Lebanon, New Hampshire) as well.

In steady state, the air flow is governed by the Navier-Stokes equations [15] given by (4.1) and (4.2), where u is velocity, p is pressure, μ is the kinematical viscosity of air, and ρ is the density for air.

$$\nabla \cdot u = 0 \quad (4.1)$$

$$\rho u \cdot \nabla u = -\nabla p + \mu \nabla^2 u \quad (4.2)$$

Equation (4.1) is the governing equation for the conservation of mass, and equation (4.2) is the governing equation for the conservation of momentum.

Replacing the governing equations (4.2) and (4.1) with the dimensionless variables u^* , x^* and p^* as shown in (4.3), (4.4) and (4.5). The dimensionless form of governing equations (4.6) and (4.7) are obtained, where U_{in} is the average air velocity at the inlet (naris), d_{in} is the hydraulic diameter of the nares ($4 * area/perimeter$), and $Re = U_{in}d_{in}/\nu$ is Reynolds number.

$$u^* = u/U_{in} \quad (4.3)$$

$$x^* = x/d_{in} \quad (4.4)$$

$$p^* = p/(\rho U_{in}^2) \quad (4.5)$$

$$\nabla \cdot u^* = 0 \quad (4.6)$$

$$u^* \cdot \nabla u^* = -\nabla p^* + \frac{1}{Re} \nabla^2 u^* \quad (4.7)$$

It is clear from the equations (4.6) and (4.7) that the flow is characterized by the Reynolds number. The usage of dimensionless parameters has many advantageous: (1) Reducing the potentially large differences in orders of magnitude that may occur between terms in the field equations, (2) Providing a measurement of the relative importance of the terms in the equations (dominant physical phenomena), and (3) providing an estimate of the order of difficulty of the problem [15].

These dimensionless equations are solved using FiDAP (Fluent, Inc. Lebanon, New Hampshire). To solve the full set of governing equations, proper boundary conditions have to be assigned. No-slip and rigid boundary conditions were applied on the cavity surface of each model [43]. For simulation of inspiratory flow, a uniform velocity field was imposed at the external naris, and stress free boundary condition (Neuman boundary) was assigned at the outlet (posterior of the cavity).

4.2.5 Solution Method

The airflow was simulated using FiDAP (Fluent, Inc. Lebanon, New Hampshire). The segregated solver provided by FiDAP was employed to solve the nonlinear fluid equations, since the algorithm requires substantially less memory than direct solvers [15]. Using this approach, the governing equations are solved sequentially. Because the governing equations are non-linear, iterative method is used to obtain a converged solution. Since the problem is highly nonlinear, the Reynolds number was incrementally increased to obtain convergence. To obtain comparable results with [43] for human studies, the maximum Reynolds number or 610 was used for all simulations. This number is based upon the hydraulic diameter ($D = 4*area/perimeter$) of the external naris. Since we did not have velocity measurement of sniff volume, and no published quantity value was available for the dog study, a uniform velocity of $1m/s$ was used in the simulation.

4.3 Results

Six human nasal cavity models and one dog nasal cavity model were developed, and steady state air flow was simulated with FiDAP. In this section, the simulation results

in human studies will be presented first, followed by the simulation results of the dog study.

4.3.1 Air Flow Simulation in Human Nasal Cavities

Velocity Profile

The complete air flow field for human study 4 will be presented in this section, followed by the comparison study of the six human studies. As indicated by other investigators, at lower flow rates or when Reynolds numbers less than 610, the air flow is laminar [31, 43]. Steady state air flow is simulated with a Reynolds number of 610, which corresponds to an average velocity of about 1.6, 0.7, 1.0, 0.6, 1.3 and 1.7 m/s at the external naris for studies 1, 2, 3, 4, 5 and 6, respectively.

The simulated velocity fields of human study 4 are shown in Figure 4.6, 4.7, 4.8, 4.9, 4.10 and 4.11 for the cross-section planes 1, 2, 3, 4, 5 and 6 respectively. The locations of the cross-section planes are indicated on Figure 2, and the parameters of the airway on each plane are listed in Table 4.3. On each plane, both the secondary motion vectors (in plane) and the axial velocity contours are presented. Color is used to show the velocity magnitude on the planes. Blue color is the lowest speed, red color represents the highest speed, the other colors are in between. Fifteen contour lines are plotted on each plane. Each contour line represents an equal increment in speed from zero at the boundaries to the maximum value on the plane. The maximum speed and the normalized maximum axial velocity on each plane are given in the figure caption. The magnitude of the velocity is calculated by $\|u\|_2 = (u_x^2 + u_y^2 + u_z^2)^{1/2}$. The secondary velocity is $\vec{u}_x + \vec{u}_y$.

The simulation results of the six human nasal cavity models are compared at about the same anatomic position. Figure 4.12 and Figure 4.13 show the secondary velocity

Table 4.3: The area and the perimeter on cross-section planes and positions of the planes relative to the naris

| | Distance from anterior Tip(mm) | Cross-Sectional Area(mm^2) | Cross-Sectional Perimeter(mm) |
|---------|--------------------------------|--------------------------------|-------------------------------|
| Naris | | 164.0 | 46 |
| Plane 1 | 33.5 | 152.0 | 76 |
| Plane 2 | 44.0 | 197.0 | 141 |
| Plane 3 | 54.5 | 213.5 | 167 |
| Plane 4 | 65.0 | 214.5 | 161 |
| Plane 5 | 75.5 | 219 | 141 |
| Plane 6 | 85.0 | 220 | 88 |

vectors (u_x, u_y) and velocity magnitude iso-contours for the six human studies at the cross-section plane 5. The cross-section plane 5 is located at the posterior region of the nasal cavity shown in Figure 4.5. In the contour plots, fifteen contour lines are plotted on each plane. Each contour line represents an equal increment in speed from zero at the boundaries to the maximum value on the plane.

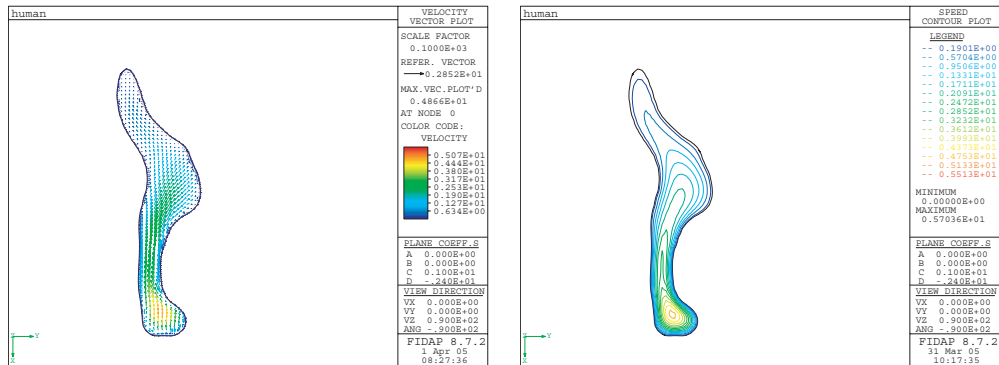


Figure 4.6: Secondary velocity vector and axial velocity contours on cross-section planes 1. The Maximum velocity is 4.865 m/s. The maximum axial velocity is 4.805 m/s.

As shown in Figure 4.12 and 4.13, the anatomic structures of the human nasal cavities and the air flow fields in the cavity models were different from one study to another. Although the nasal cavities in all six studies share a similar overall structure,

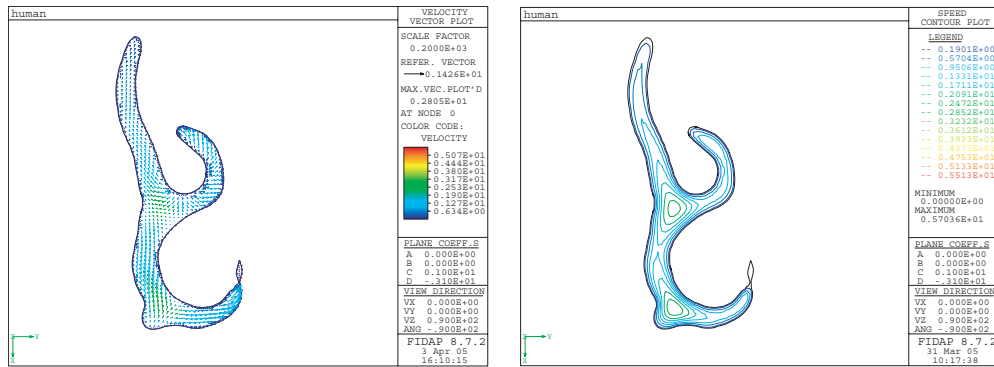


Figure 4.7: Secondary velocity vector and axial velocity contours on cross-section planes 2. The Maximum velocity on this plane is 2.865 m/s. The maximum axial velocity is 2.785 m/s.

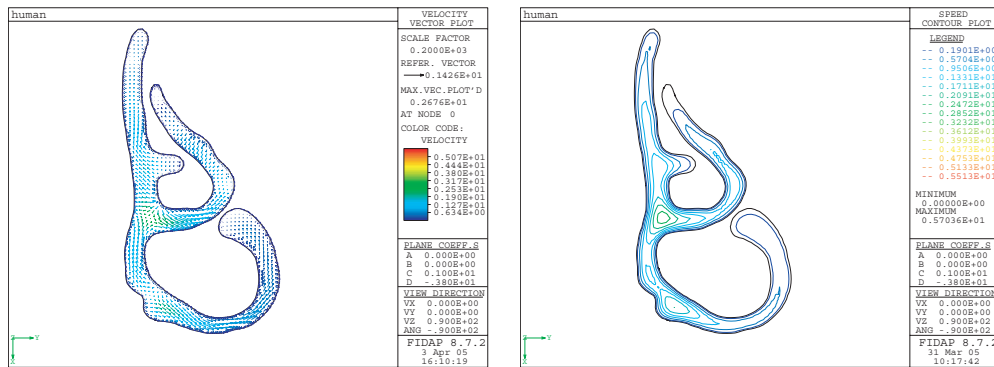


Figure 4.8: Secondary velocity vector and axial velocity contours for plane 3. The Maximum velocity on this plane is 2.676 m/s. The maximum axial velocity is 2.675 m/s.

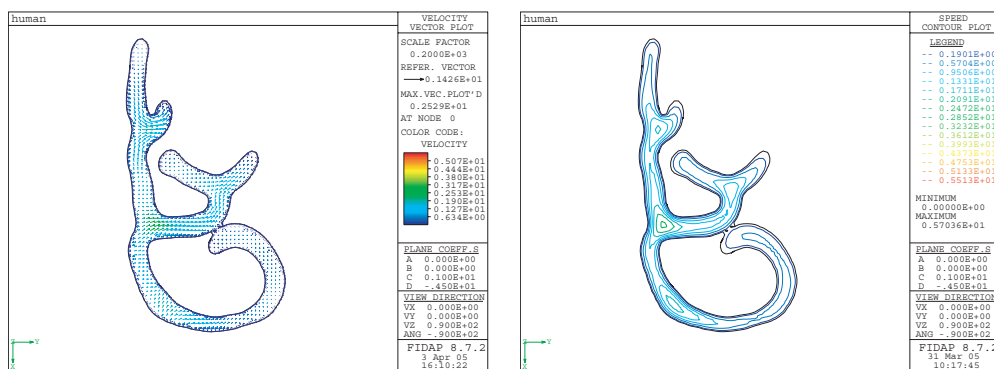


Figure 4.9: Secondary velocity vector and axial velocity contours for plane 4. The Maximum velocity on this plane is 2.529 m/s. The maximum axial velocity is 2.506 m/s.

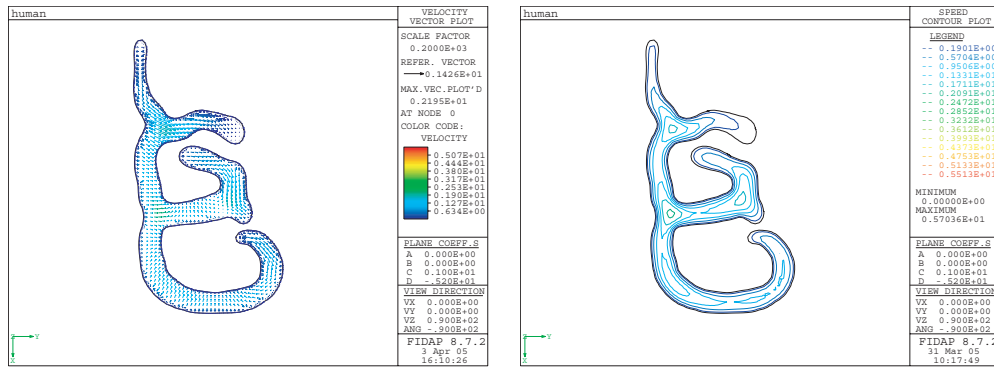


Figure 4.10: Secondary velocity vector and axial velocity contours for plane 5. The Maximum velocity on this plane is 2.194 m/s. The maximum axial velocity is 2.178 m/s.

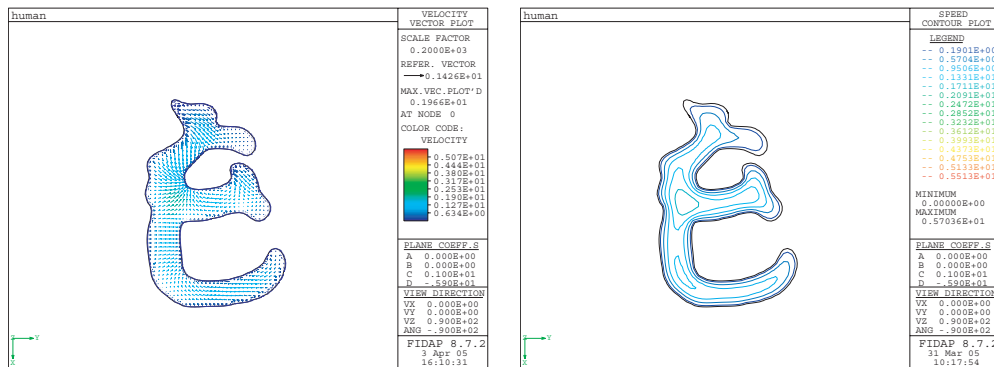


Figure 4.11: Secondary velocity vector and axial velocity contours for plane 6. The Maximum velocity on this plane is 1.966 m/s. The maximum axial velocity is 1.943 m/s.

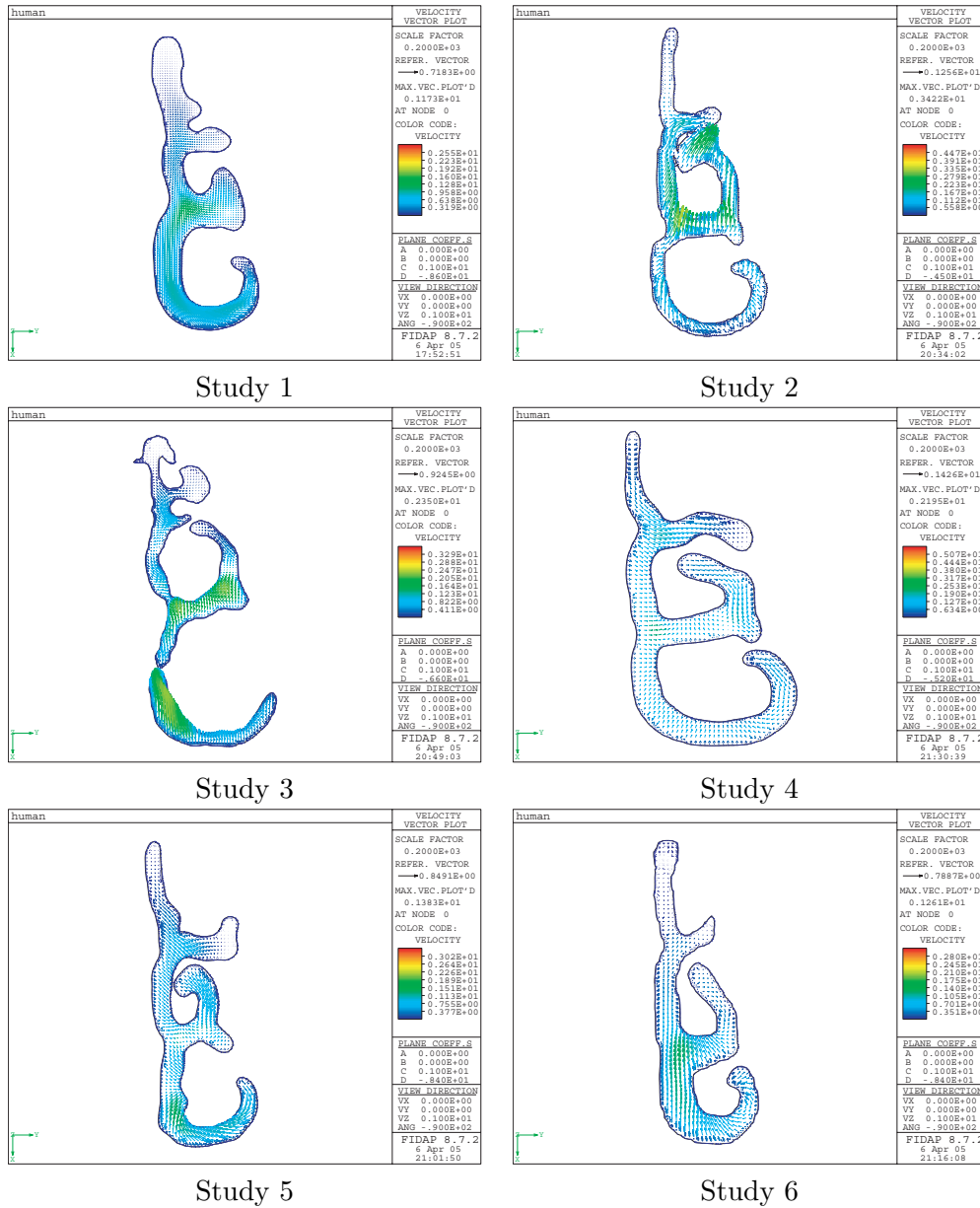


Figure 4.12: Secondary velocity vectors at the plane 5 in Figure 4.5 for each of the six human studies

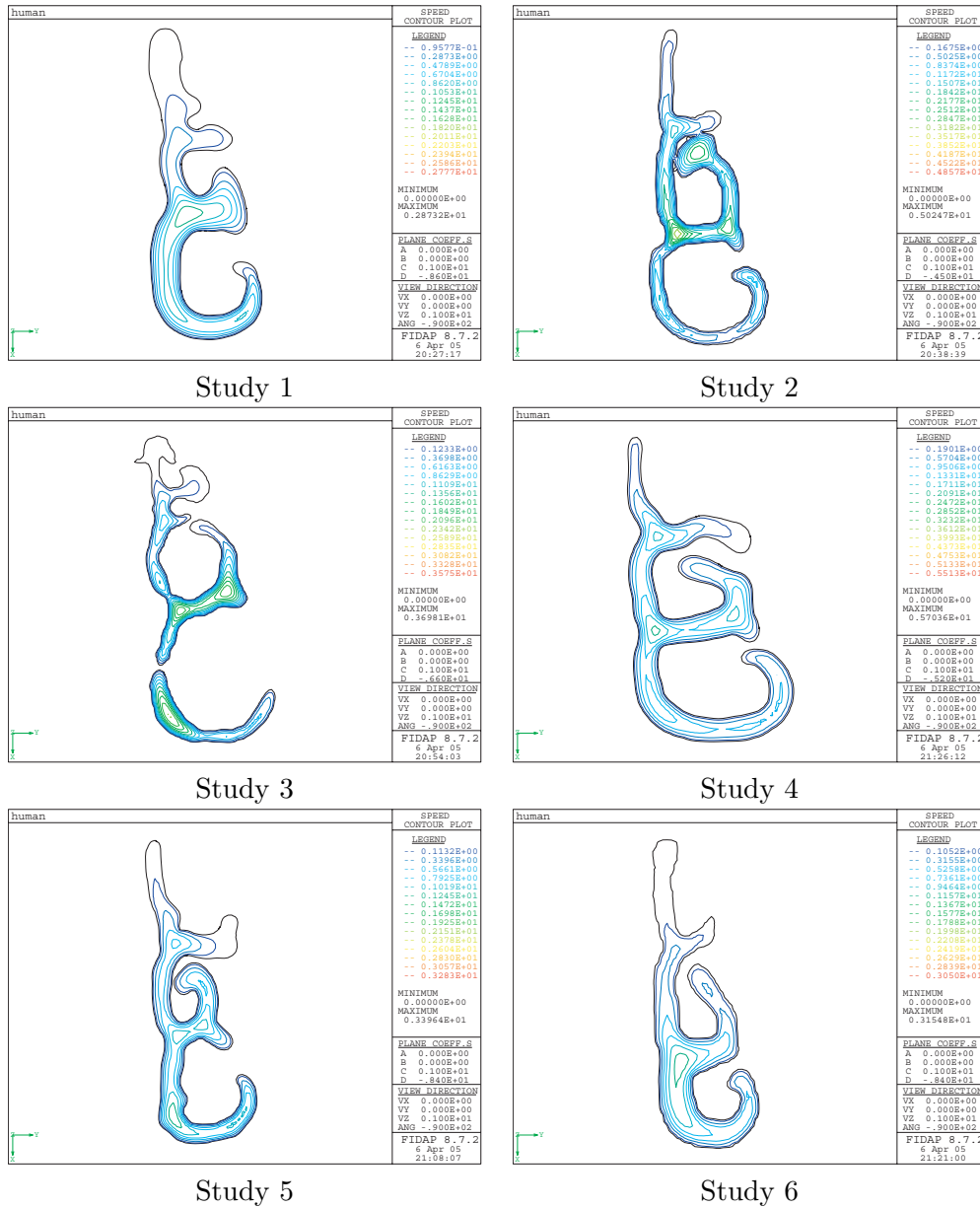


Figure 4.13: Velocity magnitude iso-contours at plane 5 in Figure 4.5 for each of the six human studies

signification differences exist between the models, such as the size and the shape of meatuses and the major airway channels. We thought all the studies were from normal volunteers, but study 3 had an obstruction in the common meatus which separated the air way at that location.

Due to anatomic differences, the air flow fields were significantly different from each other. The air flow field was controlled by the anatomic structure. The airflow was similar in the anterior part of the main nasal passage, because the variation is small in this region. However, the airflow was different in the posterior regions among the six studies due to large variations in shape there. In the anterior region, the highest air speeds are along or near the nasal floors, and the second lower peak occurs in the middle airways for all models. This is consistent with the results from other researchers [79, 31, 43, 14]. At the posterior region in general, as the size of inferior and middle meatuses increase, air flows towards tip of meatuses, but different air flow distributions were observed for the six studies. In study 1,2,4 and 6, the highest air speeds appear near the roots of the middle meatuses, and the middle meatus acts as the primary path of airway, as shown in Figure 4.12 and Figure 4.13, which is consistent with the observation of [79, 83]. For study 3 and 5, the highest speeds are near the floors of the inferior meatuses, and the inferior meatus acts as the main air channel. This is similar to the results of [31, 43]. The lowest air speeds appear in the superior regions in all studies, and the air speeds in the olfactory airway decrease from the inferior to the superior ends, which are the same as the results of [31, 43].

The axial component of velocity is much larger than the secondary component through most of the main nasal passages, which is consistent with the results of other investigators [31, 43]. The secondary motions of flow are induced by curvature, directional

changes and rapid expansion of the airway [43]. Due to complexity of the anatomical structure, the secondary flow is very complex. There are several rapid anatomy changes from the naris to the posterior of the nasal cavity. Anterior to the turbinates, where the inferior to superior length of the airway increases, the flow turns towards the horizontal and the secondary velocities point almost upwards towards the nasal roof, which is consistent with other investigators [31, 43]. In the anterior part of the main nasal passages, the size of the inferior and middle meatuses increase dramatically, where the pressure drops and the fluid flows towards the lateral side of the airway, into the superior parts of meatuses. Near the end of the main nasal passages, there is abrupt downward turn in the nasal roof, which results in a downward motion of the fluid. The inferior and middle meatuses decrease in size, which forces the fluid flows towards septum in these meatuses. These results are compatible with [31, 43].

Particle Path Simulation

Simulating the transportation of odorant particles through the nasal cavity to the olfactory epithelium is important to understand olfactory processes. The flow pattern in the olfactory slit is of special significance, since it affects the transportation of odorant molecules to the olfactory epithelium. With the obtained air flow field, we performed odorant particle path simulations on study 4, which was close in shape to the nasal cavity model in [43]. Three groups of particles were simulated. In these simulations, neutrally buoyant point particles were released at the external naris plane, and their motion through the nasal cavity was tracked with FiDAP (Fluent, Inc. Lebanon, New Hampshire). It is assumed that the introduced particles do not change the fluid field, which is a valid assumption if the odorant particles are very small. The first group is

released near the ventral tip, about 1 mm away from the wall. The second group is released at the midline of the naris. The third group is at the dorsal tip about 1 mm away from the lateral wall. The particle passing streamlines are shown in Figure 4.14, 4.16 and 4.16 for group 1, 2 and 3 respectively. The exact passing position at cross-section planes 1 to 6 for each particle is shown in Figure 4.15, 4.17 and 4.19 for group 1, 2 and 3 respectively. As shown in Figure 4.15, some particles of this group reached the olfactory region, where they might be detected by the olfactory system. All other particles of group 2 were concentrated in the common and inferior meatuses, no particle reached the olfactory slit, which is shown in Figure 4.17. All particles of group 3 flew through the inferior and middle meatuses shown in Figure 4.19. No particle reached the olfactory region. This indicates that only particles released at particular positions within the naris may reach the olfactory system. This result is consistent with other researchers' observation [43].

4.3.2 Air Flow Simulation in Dog Nasal Cavity

After applied the modeling technique with human subjects, the same method is used to a study for dogs. Steady state air flow was simulated in a dog nasal cavity model constructed from 70 CT slices (No. 13 to No. 82 CT slices). Since the air flow in sinuses should be very small, the effect of the sinuses to the overall velocity field is expected to be small. Removing the sinuses can reduce the size of the problem dramatically in terms of mesh size. The model with the sinuses removed was used in the simulation. The simulation results are presented in this section. The secondary air flow vectors will be shown in eight cross-section planes of the airway, and the corresponding CT slices will be shown as well.

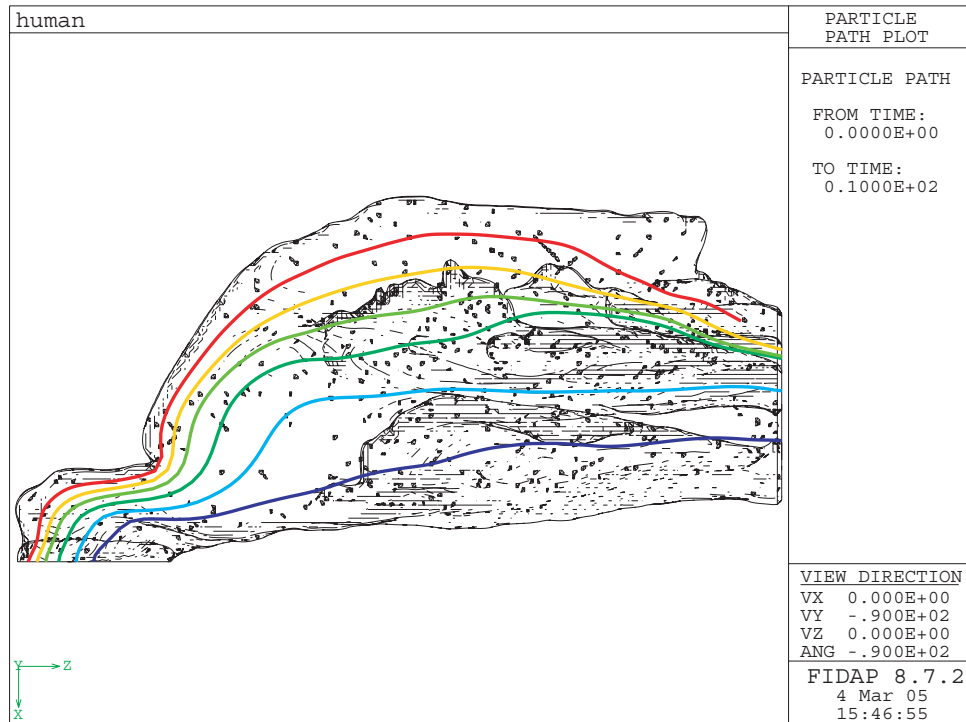


Figure 4.14: Path routes simulation for group 1 particles for study 4. The particles were released about 1 mm away from the ventral tip of the naris wall.

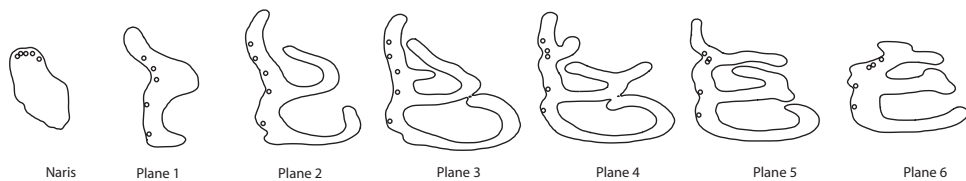


Figure 4.15: Group 1 particle passing positions on cross-sectional planes 1 to 6 as shown in Figure 4.5. The particles were released about 1 mm away from the ventral tip of the naris wall.

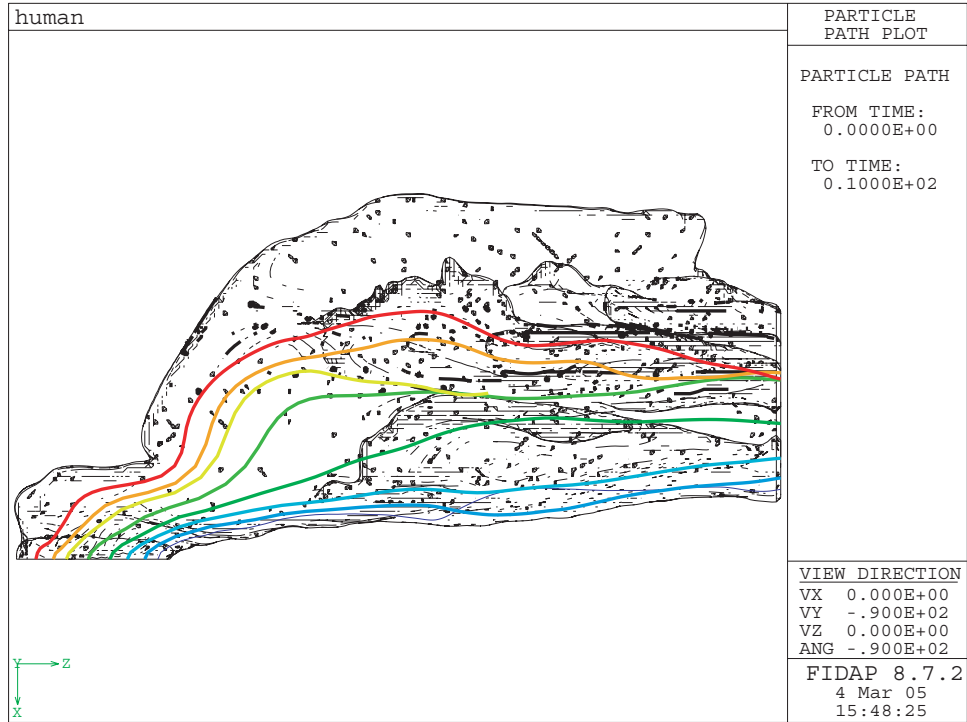


Figure 4.16: path routes simulation for group 2 particles. The Particles were released at middle line of the external naris.

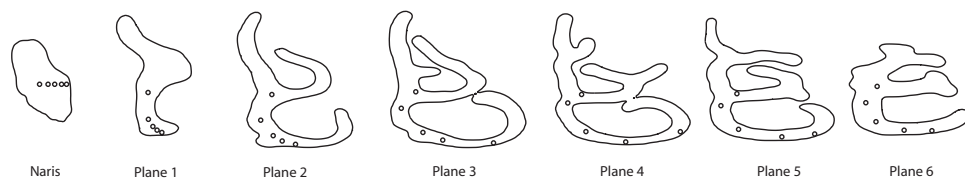


Figure 4.17: Group 2 particle passing positions on cross-sectional planes 1 to 6 as shown in Figure 4.5. The Particles were released at middle line of the external naris.

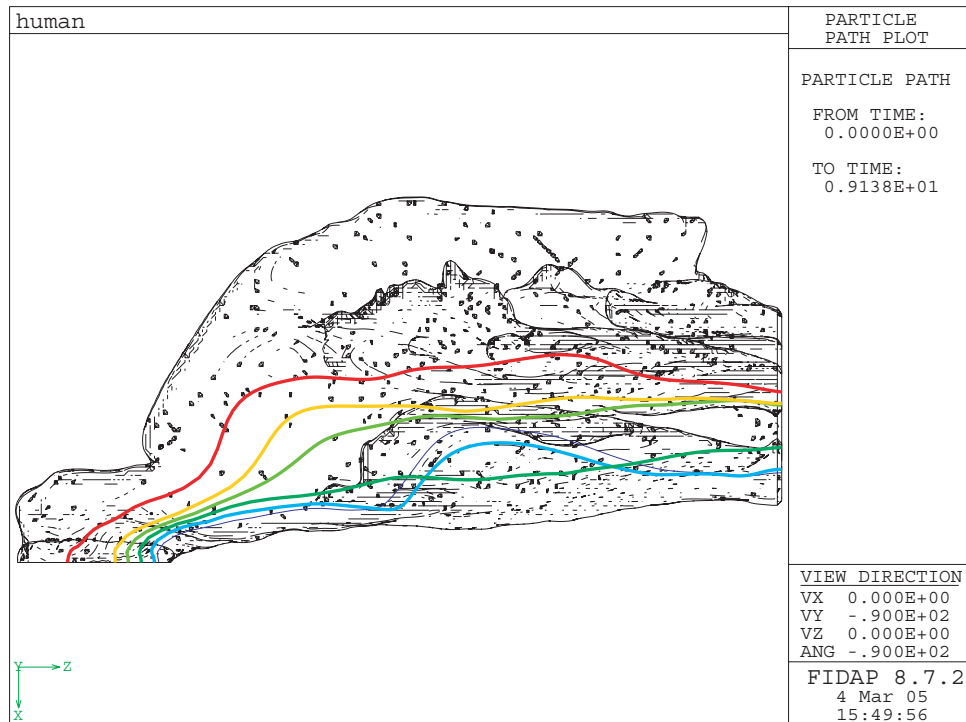


Figure 4.18: Path routes simulation for group 3 particles. The particles were released about 1 mm away from the dorsal tip of the naris wall.

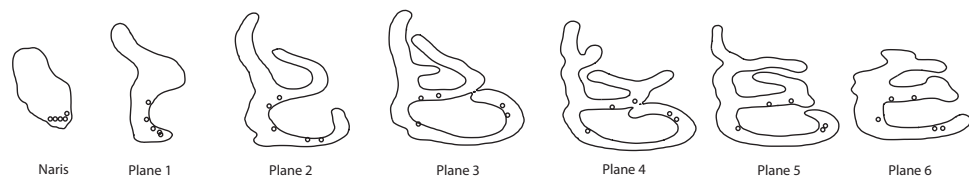


Figure 4.19: Group 3 particle passing positions on cross-sectional planes 1 to 6 as shown in Figure 4.5. The particles were released about 1 mm away from the dorsal tip of the naris wall.

The geometric structures of nasal cavities determine the air flow field. At the anterior section, the two airway channels were relatively simple and almost symmetric. The simulated air flow field were similar, as shown in Figure 4.3.2, 4.3.2 and 4.3.2. The air flow in the right cavity is much more uniform than in the left cavity at the same cross-section plane. As the geometric difference became larger towards posterior regions, air flow in the two channels were significantly different, as shown in Figure 4.3.2. The air flow became more complex as the geometry became more complex towards the posterior region, shown in Figure 4.3.2, 4.3.2, 4.3.2 and 4.3.2. It becomes uniform in the nasopharyngeal region, as shown in Figure 4.3.2.

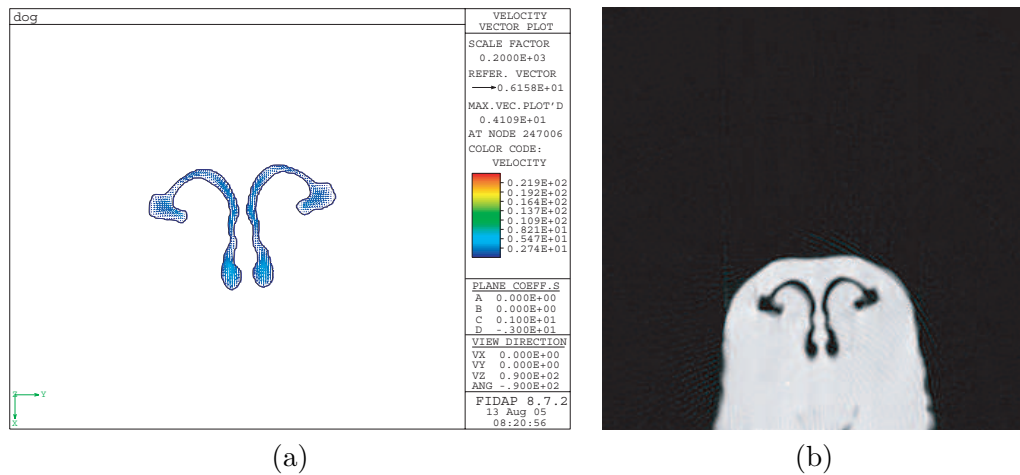


Figure 4.20: The velocity vectors at the No. 15 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 15 CT slice of the dog(counted from the naris to the Nasopharyngeal)

4.4 Discussion

With the automatic mesh generation algorithm, multiple numerical models of nasal cavities were developed very efficiently. The model constructing time is about 4 hours for each model. On the other hand, it could take several weeks or months to develop a

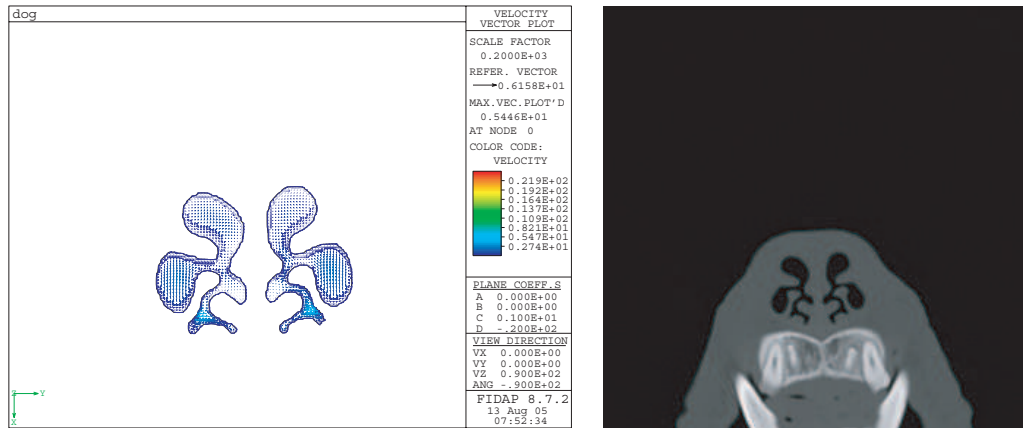


Figure 4.21: The velocity vectors at the No. 24 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 24 CT slice of the dog(counted from the naris to the Nasopharyngeal)



Figure 4.22: The velocity vectors at the No. 32 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 32 CT slice of the dog(counted from the naris to the Nasopharyngeal)

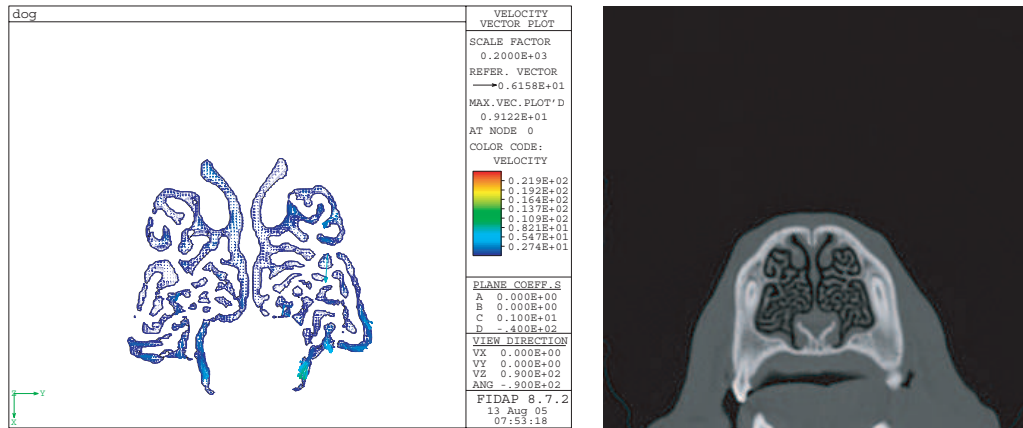


Figure 4.23: The velocity vectors at the No. 40 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 40 CT slice of the dog(counted from the naris to the Nasopharyngeal)

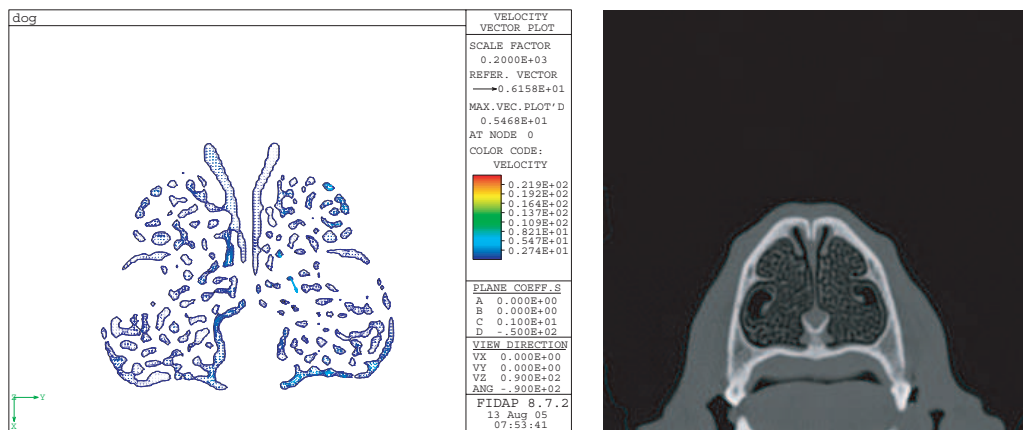


Figure 4.24: The velocity vectors at the No. 48 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 48 CT slice of the dog(counted from the naris to the Nasopharyngeal)

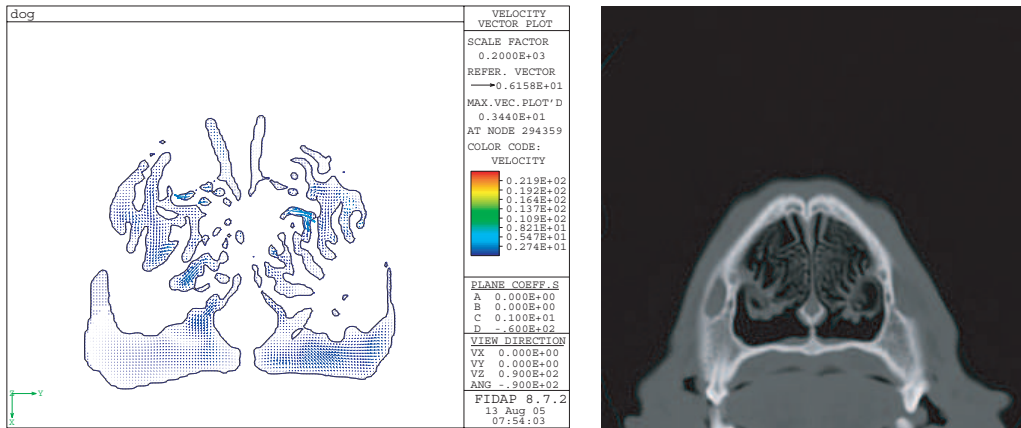


Figure 4.25: The velocity vectors at the No. 53 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 53 CT slice of the dog(counted from the naris to the Nasopharyngeal)

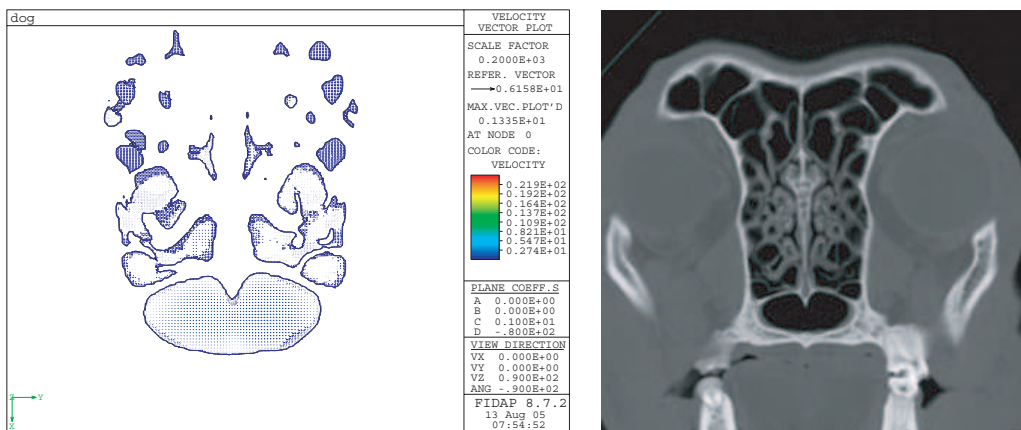


Figure 4.26: The velocity vectors at the No. 66 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 66 CT slice of the dog(counted from the naris to the Nasopharyngeal)

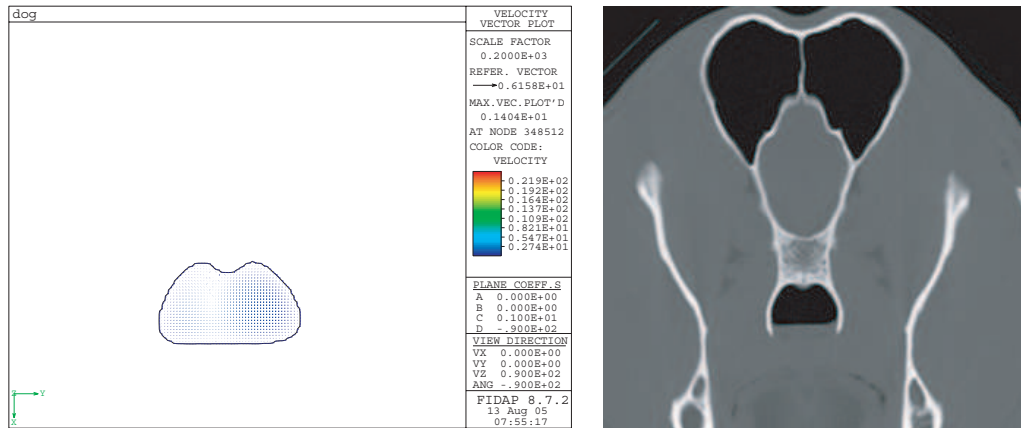


Figure 4.27: The velocity vectors at the No. 74 CT slice of dog nasal cavities. (a) the in-plane velocity vectors. (b) the No. 74 CT slice of the dog(counted from the naris to the Nasopharyngeal)

physical model or construct a numerical model manually. With the six human models, we conducted comparison study over different models with similar boundary conditions. To our knowledge, this is the largest study on air flow in human nasal cavities. From our models and simulation results, it is concluded that although the main characteristics of the human nasal cavities are similar, larger differences exist between individuals. The anatomic differences can significantly affect the flow patterns.

Since the air flow field is highly dependent on the anatomic structure, simulation results of one study may not be true for another. In addition, various anatomical deformities can affect the nasal airway, such as a deviated septum and airway pathologies. Other factors that cause anatomy differences include race, gender and age etc [43]. This indicates that subject-specific airway models should be developed and simulations need to be done on the specific model. With the techniques introduced in this dissertation, it is practical to develop numerical models and perform CFD airflow simulations for a large numbers of studies.

However, there are some limitations for the current work. To simplify simulations of the complex air flow in the nasal cavities, assumptions and simplifications were made. The air flow inside the nasal cavity was assumed to be steady, and a no-slip and rigid boundary condition were applied. This is a valid assumption only for normal quiet breathing rates [31, 43]. For vigorous sniffs, the flow should be modeled as unsteady flow, the nasal wall is not rigid as well, and the cavity shape does not remain constant either. In reality, there is a mucus layer in the nasal cavity. The mucus flow on the airway surface results in non-zero velocity. Since the mucus layer is very thin, about 10 microns [43], the effects of the mucus layer are negligible in the current study. According to other investigators [31, 43], the sinus and nasal hair have little effect on the general airflow in the cavity, therefore the sinus and the nasal hair are not included in our model.

While a constant inspiration is used in our simulation, human respiration is a time varying process. The effect of a time varying sniff pattern on the olfactory airway should be studied further. In addition, as reported by experimental observations and measurements of other investigators [31, 83, 12], the air flow during sniffs in the human nose is semi-turbulent or turbulent. In the future, turbulent flow through the nasal cavities needs to be simulated. The parameters of turbulent flow will need to be obtained from experimental models.

CHAPTER 5

MODELING ELECTRO-OLFACTOGRAM RESPONSE SHAPES USING SYSTEM IDENTIFICATION TECHNIQUES

In this chapter, a mathematical model is proposed for the shape of electrical responses of olfactory epithelium or olfactory neurons to odorant stimuli experimentally measured in the rat and dog. This chapter begins by introducing the experiment setting and data collection. Then, system identification techniques are presented, and an output error (OE) model is used to develop the shape fitting models. Finally, the developed models were used to predict the shapes of the responses to different odorant concentrations for a fixed duration of stimuli. The model has the potential to evaluate olfactory electrical responses and to estimate kinetics of G-protein cascade within the olfactory receptor neuron.

5.1 Experiments and Data Collection

All procedures for animal experiments conform to the NIH animal use guidelines and were approved by the Institutional Animal Care and Use Committee at Auburn University. Electro-olfactogram (EOG) and single-cell voltage-clamp measurements were performed on freshly harvested olfactory tissue. Rat (n=10) and dog (n=5) septal olfactory mucosa were dissected out and placed in Hank's Balanced Salt Solution (Sigma Chemical Co., St. Louis, MO) minutes prior to each experiment. The tissue was placed in a perfusion chamber (Warner Instrument Corp., Hamden, CT) such that the basal portions were immersed in physiological buffer (containing 137 mM NaCl, 5.3 mM KCl,

4.2 mM NaHCO₃, 0.4 mM KH₂PO₄, 3.4 mM Na₂HPO₄, and 5.6 mM D-Glucose at pH 7.4), while the epithelial surface with olfactory cilia was exposed to air.

In EOG experiments, glass patch electrodes (World Precision Instruments, Inc., Sarasota, FL) of approximately 24 μm tip opening were filled with the same physiological buffer and connected to a patch-clamp amplifier in order to detect EOG responses from the olfactory epithelium. Once the contact between the electrode tip and the surface of the olfactory epithelium was formed, air puffs of an odorant mixture containing equal parts of ethyl butyrate, eugenol, (+) and (-) carvone (Aldrich Chemical Company, Inc., Milwaukee, WI) were applied at various concentrations and stimulus durations.

In single-cell voltage-clamp experiments, the patch electrode (resistance of 8 - 16MΩ) was filled with a solution containing 110 mM KCl, 4 mM NaCl, 2 mM NaHCO₃, 1 mM MgCl₂, 0.1 mM CaCl₂, and 2 mM MOPS at pH 7.4. After a stable contact with an olfactory neuron was made, the surface of the tissue was stimulated with various stimulus durations (air puffs) of odorant mixture containing equal parts of ethyl butyrate, eugenol, (+) and (-) carvone in vapor above a 16 mM water solution (Aldrich Chemical Company, Inc., Milwaukee, WI) [84].

In both experiments, odor responses over the time duration of several minutes were amplified by a MultiClamp 700A patch-clamp amplifier (Axon Instruments Inc., Union City, CA), filtered at 0.1 to 1 kHz, and subsequently recorded on a computer for storage and analysis. The responses were analyzed by the pCLAMP data analysis program (Axon Instruments Inc., Union City, CA) and exported in ASCII format for the purpose of olfactory response modeling. Over 200 single-cell and 100 EOG recordings were obtained.

5.2 System Identification

System identification is a method to build mathematical models of a dynamical system based on measured data. The most common models are difference equations descriptions, such as autoregressive (AR) and moving average (MA) models, as well as all types of linear state-space models. These models are constructed by adjusting parameters within a given model until its output coincides with the measured output.

The basic concepts of system identification can be depicted in Figure 5.1. u , y and e denote the input, output and disturbance (noise) respectively. The input signal is usually known, and the output can be measured. The disturbance e is essentially unmeasurable, but its properties can be expressed in terms of power spectrum. The only unknown is the system itself. By assuming that the signals are related by a linear system, the relationship can be set up based on different model structures. There are several types of linear input-output models [54]. The disturbance-free input-output dynamics can be either AR, MA, or both (ARMA). Linear models also differ in the way of how random disturbances enter the model. Random disturbances are affected by the model and the disturbance-free input-output dynamics. In the ARX model, the relationship is written in discrete-time form given by (5.1). The explicit form is given by (5.2), where $u(k)$ is the input, $y(k)$ is the measured output, and $e(k)$ is the measurement noise. The parameter n_d is the amount of delay in samples from input to output in the model. The parameters n_a , n_b , and n_d and the coefficients a_1, a_2, \dots, a_{n_a} and b_1, b_2, \dots, b_{n_b} are chosen to match the data.

$$A(q)y(k) = B(q)u(k - n_d) + e(k) \quad (5.1)$$

$$y(k) + a_1y(k-1) + \dots + a_{n_a}y(k-n_a) = b_1u(k-n_d) + \dots + b_{n_b}u(k-n_d-n_b+1) + e(k) \quad (5.2)$$

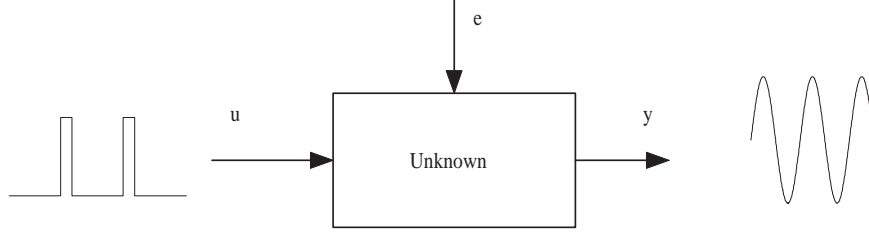


Figure 5.1: System identification input-output configuration

By rearranging Equation 5.2, a model known as output error (OE) [54] model can be obtained. This model is a combination of a disturbance-free response model and white measurement noise at output. The OE model is described by the following pair of linear difference equation given by (5.3), where $u(k)$ is the input, $y_{df}(k)$ is the disturbance-free output, $y(k)$ is the measured output, and $e(k)$ is the measurement noise. The parameter n_d is the amount of delay in samples from input to output in the model. The first equation relates $y_{df}(k)$ at time instant k to n_a past outputs $y_{df}(k-j)$ and n_b inputs $u(k-n_d-j)$. The parameters n_a , n_b , and n_d and the coefficients a_1, a_2, \dots, a_{n_a} and b_1, b_2, \dots, b_{n_b} are chosen to match the data.

$$\begin{aligned} y_{df}(k) &= -a_1y_{df}(k-1) - \dots - a_{n_a}y_{df}(k-n_a) + b_1u(k-n_d) + \dots + b_{n_b}u(k-n_d-n_b+1) \\ y(k) &= y_{df}(k) + e(k) \end{aligned} \quad (5.3)$$

5.3 Fitting Eog Shape with OE Model

The EOG and single-cell responses were modeled by the OE model discussed in the previous section. OE models were fit to the rat measured EOG, rat single-cell, and

dog EOG responses using the following procedures. First, for each of the three response types, two sets of stimuli and measured responses were selected. These responses were typical responses chosen from the single-cell and EOG recordings. One set was called the fit data set and the other was called the validation data set. The stimulus concentration and duration were the same for both the fit and validation data sets. The responses were filtered by a low-pass Bessel Filter with cutoff frequency at 100 Hz and downsampled to obtain a sampling rate of 200 Hz. Next, for each response type, models with a range of orders ($n_a = 1-10$, $n_b = 1-10$, $n_a \geq n_b$, and $n_d = 1-60$) were chosen to the fit data using prediction error method [54]. Each model was then simulated. The response was compared to the validation data set. The parameters n_a , n_b , and n_d were chosen to minimize the mean-square error between the model response and the validation data.

5.4 Experimental Results

Results of the EOG in vivo experiments revealed characteristic odorant-elicited responses marked by a negative change in measured voltage (on-response) that was followed by an upward recovery (off-response). The negative change in voltage was rapid and started soon after application of the odorant, while the recovery (off-response) was significantly slower. The delay in the negative voltage change, also called the latent period, is due to diffusion of the odorant through the olfactory mucus as well as binding of odorant molecules to receptors and conformational changes of proteins of the primary enzymatic cascade. These characteristics, which are typical in all EOG responses, were observed in both the rat, and dog EOGs.

Single-cell voltage-clamp experiments on rat olfactory tissue revealed characteristic odorant-elicited membrane depolarization and repolarization. The depolarizations,

which are observed upon odorant application, reflect the rapid negative current. The current is measured as the inside of the neuron becomes more positive due to the net movement of ions across the membrane. The repolarization reflects the slower recovery of the neuron back to a resting state where it can once again be stimulated.

Models were fit to typical rat single-cell ($n_a=5$, $n_b=1$, $n_d=52$), rat EOG ($n_a=8$, $n_b=2$, $n_d=50$), and dog EOG ($n_a=9$, $n_b=4$, $n_d=11$) responses. The coefficients of these models are shown in Table 5.1. Figure 5.2 shows, for all three models, the measured response from the validation data set and the response obtained by simulating the linear model to the same input. The input signal for each case is a square pulse with an amplitude of 16mM and duration of 500ms, 200ms, and 100ms for the rat single-cell, rat EOG, and dog EOG respectively. The arrow in each plot designates the start of the input pulse. There can be significant changes in response amplitude from experiment to experiment due to distribution of olfactory neurons, state of viability, and individual variability of the specimen. Predicting these changes is beyond the capabilities of the linear model, but the model can predict the shape of the response. As a result, both the measured and simulated responses are normalized so that the peak response is -1.

To assess the ability of the models to predict the response to other inputs, the response of the rat EOG model was computed for stimulus concentrations of 1mM and 4mM and a duration of 200ms. The measured and simulated responses are normalized, and are shown in Figure 5.3(a) and Figure 5.3(b). The response of the dog EOG model to a 16mM stimulus of 500ms duration is shown in Figure 5.3(c).

Table 5.1: Coefficients for the rat single-cell, rat EOG, and dog EOG models. Blank entries denote a coefficient of zero. The a_j coefficients have no units. The rat single-cell b_j coefficients have units of mM^{-1} . Rat and dog EOG b_j coefficients have units of μM^{-1}

| j | Rat Single Cell | | Rat EOG | | Dog EOG | |
|-----|-----------------|---------|---------|---------|---------|---------|
| | a_j | b_j | a_j | b_j | a_j | b_j |
| 1 | -0.2264 | -0.2392 | -2.0612 | -0.0629 | -1.8212 | -0.0087 |
| 2 | -0.7116 | | 1.2455 | 0.0515 | -0.3223 | 0.0306 |
| 3 | -0.7817 | | -1.8707 | | 0.4379 | 0.0297 |
| 4 | 0.5265 | | 3.2592 | | 3.2789 | -0.0794 |
| 5 | 0.1942 | | -1.7635 | | -2.4238 | |
| 6 | | | 1.3074 | | -0.8460 | |
| 7 | | | -2.0357 | | -0.3558 | |
| 8 | | | 0.9192 | | 1.7538 | |
| 9 | | | | | -0.7016 | |

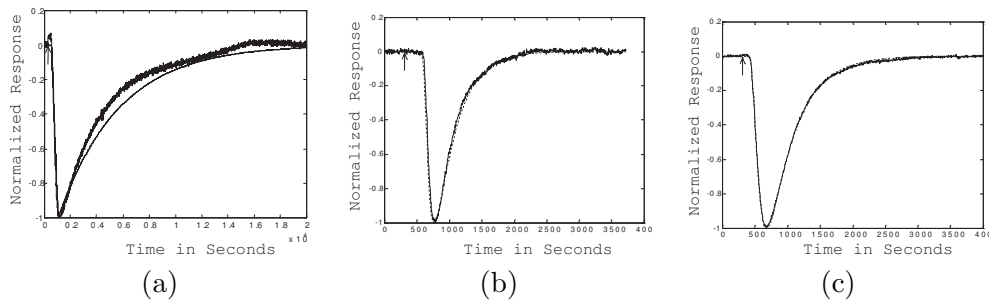


Figure 5.2: Normalized measured (solid line) and normalized simulated (dashed line) responses from a rat single-cell (a), rat EOG (b), and dog EOG (c) to a 16mM stimulus. The measured responses are the validation data. The arrow denotes the start of the stimulus. The stimulation duration was (a) 500ms, (b) 200ms, (c) 100ms.

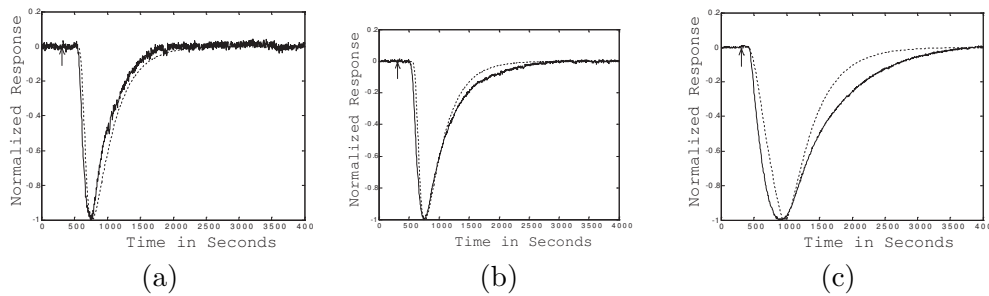


Figure 5.3: Normalized measured (solid line) and normalized simulated (dashed line) responses: (a) rat EOG response to a 200ms, 1mM stimulus; (b) rat EOG response to a 200ms, 4mM stimulus; (c) dog EOG response to a 500ms, 16mM stimulus. The arrow denotes the start of the stimulus.

5.5 Discussion

Linear, time invariant models were used to fit EOG and single-cell data which are measured from rats and dogs. The fit accuracy for the dog EOG was better than that of rat EOG and rat single-cell. One factor that affects the fit quality is the stimulus duration. The dog EOG has the shortest stimulus duration followed by the rat EOG and rat single-cell. From a system identification standpoint, the best stimulus would be an impulse (Dirac delta function) since this stimulus would excite all modes in the system [86]. This signal does not exist in nature, but, in general, the closer a stimulus signal is to an impulse, the better the model fit will be. Consequently, the dog EOG fit accuracy is the better than others since the stimulus duration used in the experiment is the shortest.

The OE models we fit to the EOG and single-cell responses assume that the system dynamics are linear and time-invariant, which is a reasonable first step toward the development of more realistic models. The actual responses, however, are nonlinear and time-varying. In particular, the OE models cannot model phenomena such as nonlinear dose-dependency [85] and adaptation [48]. These effects can be observed to a certain extent in Figure 5.3(c). The dog EOG model, which was fit using a 100ms stimulus, is simulated with a 500ms stimulus. Increasing the stimulus duration effectively increases the concentration of the stimulus (dose). Nonlinear dose dependency as well as short-range adaptation effects in the longer duration stimulus cause differences in the shapes of the simulated and actual responses. Our experimental results in Figure 5.3(a) and Figure 5.3(b) suggest, however, that the OE models can predict the shape of the response to different odorant concentrations when the stimulus duration is not changed.

The properties of the odorant-evoked current responses observed in our experiments correspond well to those obtained in slices of olfactory epithelium and isolated olfactory neurons [19, 20, 49, 18]. It is well established that, in many vertebrate species, odorants evoke a rapid formation of second messengers (cAMP) in olfactory transduction [6], and activation of sensory currents in olfactory receptor neurons depends on a G protein-mediated cAMP second messenger system [62, 16, 17]. A rise in internal cAMP causes opening of cAMP-gated cation channels [62] resulting in inward cation current. Calcium influx through the cation selective channels triggers an additional component of inward current carried by chloride ions through the Ca^{2+} -activated Cl^- channels [47, 50]. Thus, odorant stimulation leads to a depolarization of the olfactory receptor neuron provided by two inward current components. This fast depolarization component corresponds to the initial rapid decrease of the neuron current shown in Figure 5.2(a). After currents reach their peak, they decline due to hydrolysis of second messengers [6, 51]. Thus kinetics of the decreased and increased components of the odorant-current response reflects kinetics of synthesis and hydrolysis of second messengers. Making a model based on kinetic properties of all components of olfactory enzymatic cascade including activation of receptor, G-protein, adenylyl cyclase, phosphodiesterase, and second messenger gated ion channel, would be an enormously difficult task simply due to lack of experimental parameters of the olfactory cascade [51].

By using the model described in this work, one can quantify experimental odorant responses, estimate kinetics of synthesis and degradation of second messengers that are primarily responsible for the shape of the olfactory response. The model can also describe a latent period in the olfactory response, on-response, and off response [67]. The model described here does not describe odor discrimination and adaptation but can be used

to evaluate shapes of olfactory electrical responses and to estimate kinetics of G-protein cascade.

CHAPTER 6

CONCLUSION

In this dissertation, a numerical technique for modeling of nasal cavities and a mathematical model for the shape of the electro-olfactogram (EOG) were developed. The nasal cavity modeling technique includes two major parts: the first part is the hierarchical spline-based registration algorithm which is able to handle large differences between adjacent image slices of nasal cavities; the second part is a new 3-D finite element mesh generation algorithm, called marching volume elements (MVE), which has the capability of constructing volume mesh from volume data defined on 3-D regular grid. Six human nasal cavity models and a dog model were developed, and air flow simulations were conducted with the developed models. In addition, we developed a mathematical model to estimate the shape of electrical responses of olfactory epithelium to odorant stimuli. In this chapter, the main contributions of these algorithms are summarized and some directions for future work are suggested.

6.1 Hierarchical Spline-Based Interpolation

In Chapter 2, a hierarchical spline-based registration (HSR) algorithm is proposed to perform slice interpolation for reconstruction of nasal cavity models from computed tomography (CT) image slices. This interpolation algorithm is a combination of the optical flow technique and the block matching method. The optical flow technique is traditionally used in motion analysis in computer vision and video compression. This technique is able to find the correspondences of pixels in sequence of images. The use

of the block matching method reduced the computation time of the problem. The HSR algorithm works in a coarse-to-fine fashion on hierarchical image pyramids constructed by recursive decimation of original images. The basic idea of the algorithm is trying to find a displacement field which can warp one image into another image, and the intermediate slices can be obtained by interpolating the displacement field. The algorithm converges fast and is able to avoid local minima.

6.1.1 Main Results

1. The performance of the HSR algorithm was compared with other interpolation techniques on two binary artificial images. The results showed that the HSR algorithm produced the best result.
2. The performance of the HSR algorithm was compared with other interpolation techniques on CT slices of human nasal cavities. The algorithm is able to handle large offsets and differences between adjacent CT slices.
3. The human nasal cavity surfaces reconstructed from the HSR interpolated data were much smoother than the results of the other techniques tested.

6.1.2 Future Work

As far as computation cost is concerned, the HSR algorithm is much more expensive than linear interpolation or shape-based interpolations. The accuracy and the performance of the HSR algorithm are highly dependent on the chosen block size. Smaller

block size results in lower estimation error, while too small block will increase the computation cost dramatically and the algorithm becomes sensitive to noise. Alternatives to block matching need to be investigated.

6.2 Marching Volume Element Algorithm

In Chapter 3, a new 3-D mesh generation algorithm is developed. This algorithm is able to extract surface information and create finite element mesh from 3-D grided data set. The algorithm decomposes a 3-D volume domain into small cubes of the same size first, then the surface triangulation and volume tessellation for each cube can be found by looking up a predefined look-up table. Marching through all the cubes in the volume data will generate desired boundary triangles and 3-D finite element mesh. Because the algorithm only works on local data set, it is memory efficient, and is able to manage large data set.

6.2.1 Main Results

1. The performance of the MVE algorithm was compared with GAMBIT (Fluent, Inc. Lebanon, New Hampshire) on an artificial mortar which was a solid cylinder with a hemisphere shaped region removed from the top. Six different resolutions ($16 \times 16 \times 8$, $32 \times 32 \times 16$, $64 \times 64 \times 32$, $128 \times 128 \times 64$, $192 \times 192 \times 96$ and $256 \times 256 \times 128$) of the mortar were meshed with both the MVE and GAMBIT. The MVE algorithm was able to mesh all the data sets, but GAMBIT couldn't mesh the two largest data sets on our computer (Sun Microsystems Ultra 80, 2G memory, Sparcv9+vis CPU).

2. By removing very small elements or the elements which have very short edges, the mesh quality is improved. Experiments on both artificial data and CT data showed that the worst elements in the mesh were removed, thus the mesh quality was improved.
3. Numerical models of six human and one dog nasal cavities were developed from CT scans with the MVE algorithm. Air flow simulations were conducted with the models.

6.2.2 Future Work

1. The finite element mesh generated by the MVE algorithm may contain bad quality elements. The existence of bad quality elements may cause computation difficulties in the solution stage, so the mesh quality needs to be improved before numerical simulations. The mesh improvement scheme currently used can remove some bad quality elements, however, the improvement is not optimal. Future work should employ mesh improvement algorithms to obtain the optimal mesh, such as finding the best position for each node, changing connections in adjacent elements and so on.
2. More accurate results can be obtained with a fine mesh. However, a fine mesh also results in high computation cost. Since the computer resource has limitations, a compromise has to be made. The optimal goal is to maximize the computational precision with the lowest computation cost. Adaptive mesh may be a good solution. In the adaptive mesh, smaller elements are used at where higher gradient fields are expected, and larger elements are used at where lower gradient fields are expected.

Making the MVE algorithm adaptive or developing a new adaptive mesh algorithm is a topic for future work.

6.3 Numerical Airflow Simulation

In Chapter 4, numerical models of six human and dog nasal cavities were constructed with the MVE algorithm from CT scans, and steady state air flow were simulated with FiDAP (Fluent, Inc. Lebanon, New Hampshire). We validated our method on human subjects first. Once a valid technique was obtained on human studies, we applied the same technique to the dog study. In the simulations, a stress free boundary condition (Neuman boundary) was assigned at the outlet (posterior of the cavity), and a uniform velocity field was imposed at the external naris. The simulation results on human studies were consistent with other researchers' work. A comparative study on six human models was conducted. To our knowledge this is the largest study on air flow simulation of human nasal cavities. The study of the air flow simulation in dog nasal cavity was the first numerical air flow simulation in an anatomically correct nasal cavity model. Since no experimental data and published material for dog study are available at this time, the justification of our result is left to future researchers when experimental results are obtained.

6.3.1 Main Results

1. Six anatomically correct human nasal cavity models were developed from CT scans of six human volunteers. The models were successfully used to simulate air flow with CFD. The results showed that air flow in nasal cavity is highly dependent on

the anatomic structure. This indicates that the study of air flow in nasal cavity should be subject-specific. The result of one study may not be true for another.

2. A dog nasal cavity model was constructed from CT slices of dog nose, and steady state air flow was simulated with the model. To our knowledge, this is the first realistic dog nasal cavity model.

6.3.2 Future Work

1. In our current work, we set a constant air flow at the naris in the simulations. This can not present the reality sniff practice which is a complex dynamic process. But the dynamic simulation of air flow with a large mesh requires huge memory and storage space. With advance in technology, dynamic simulations of air flow in nasal cavity models may be possible in the future.
2. Some researchers [31, 83, 12] pointed out that the air flow during sniffs in the human nose is semi-turbulent or turbulent, particularly for vigorous sniff. Future's work should simulate turbulent flow through the nasal cavities, but the parameters required for turbulent flow simulations must be obtained with experimental methods.

6.4 Modeling Electro-Olfactogram Response Shapes Using System Identification Techniques

In Chapter 5, a mathematical model for the shape of the olfactory epithelium or olfactory neurons to odorant stimuli experimentally measured in the rat and dog is proposed. This is a linear input-output model which describes the relationship among the

odorant stimulus input, the disturbance-free output, the measured output, and the measurement noise. The parameters of the model were obtained by fitting a general equation to the measured Electro-Olfactogram (EOG) response. The model predicts the shape of the responses to different odorant concentrations for a fixed duration of stimuli. The model has the potential to evaluate olfactory electrical responses and to estimate kinetics of G-protein cascade within the olfactory receptor neuron.

6.4.1 Main Results

1. Electro-olfactogram (EOG) and single-cell voltage-clamp measurements were performed on freshly harvested olfactory tissue of dog and rat. Odor responses over the time course of several minutes were amplified by a MultiClamp 700A patch-clamp amplifier (Axon Instruments Inc., Union City, CA), filtered at 0.1 to 1 kHz, and subsequently recorded on a computer for storage and later analysis.
2. A general mathematic model is obtained to estimate the shape of the Electro-olfactogram (EOG) to odorant stimulus. The EOG is the overall response of the olfactory cascades of neurons to stimulus. The olfactory cascades include activation of receptor, G-protein, adenylyl cyclase, and the second messenger gated ion channel. Development of a model including all these processes would be a very difficult task. Our approach circumvents this difficulty by fitting general equations to the measured response data instead of modeling individual processes involved in these responses. The advantage of this model is that all olfactory cascade processes are implicitly included in the model, although the processes themselves may be unknown.

6.4.2 Future Work

1. The EOG model developed in this research simplified the olfactory response as a linear model, which is valid only for a small region of concentration and fixed duration of odorant stimulations. The linear model has limitations to estimate the nonlinearities of olfaction, such as odorant adaptation, inhibition, and etc. In the future, nonlinear models should be developed to describe the nonlinear processes.
2. In this research, the transportation of odorant particles in the nasal cavities was tracked with simulated flow field. In the future, we may simulate the odorant particles transportation as multi-phase flows in which characteristics of odorant particles, such as particle size and water solubility, are included. With this air-odorant flow model, it is possible to estimate the interaction of the odorants with the olfactory epithelium for various sniff patterns and ambient odorants.
3. In the future, we should combine the olfactory response model with the air-flow model. The combination of the air-odorant flow model with the nonlinear olfactory model can be used to estimate olfactory response to the ambient odorant concentration.

BIBLIOGRAPHY

- [1] Anandan, P., "A Computational Framework and an Algorithm for the Measurement of Visual-Motion," *International Journal of Computer Vision*, 1989. 2(3): p. 283-310.
- [2] BainbridgeSmith, A. and R.G. Lane, "Determining Optical Flow Using A Differential Method," *Image and Vision Computing*, 1997. 15(1): p. 11-22.
- [3] Barron, J.L., D.J. Fleet, and S.S. Beauchemin, "Performance of Optical-Flow Techniques," *International Journal of Computer Vision*, 1994. 12(1): p. 43-77.
- [4] Bergen, J.R., et al., "Hierarchical Model-Based Motion Estimation," *Lecture Notes in Computer Science*, 1992. 588: p. 237-252.
- [5] Bors, A.G., L. Kechagias, and I. Pitas, "Binary Morphological Shape-Based Interpolation Applied to 3-D Tooth Reconstruction," *IEEE Transactions on Medical Imaging*, 2002. 21(2): p. 100-108.
- [6] Breer, H., I. Boekhoff, and E. Tareilus, "Rapid Kinetics of 2nd Messenger Formation in Olfactory Transduction," *Nature*, 1990. 345, 6270, 65-68.
- [7] Burt, P.J. and E.H. Adelson, "The Laplacian Pyramid as A Compact Image Code," *IEEE Transactions on Communications*, 1983. 31(4): p. 532-540.
- [8] Canann, S., M. Stephenson and T. Blacker, "Optismoothing: An Optimization-Driven Approach to Mesh Smoothing", *Finite Elements in Analysis and Design*, 13 (1972), pp. 185-190.
- [9] Chen, S.Y., et al., "Improvement on Dynamic Elastic Interpolation Technique for Reconstructing 3-D Objects from Serial Cross-Sections," *IEEE Transactions on Medical Imaging*, 1990. 9(1): p. 71-83.
- [10] Choi, W.P., K.M. Lam, and W.C. Siu, "An Adaptive Active Contour Model for Highly Irregular Boundaries," *Pattern Recognition*, 2001. 34(2): p. 323-331.
- [11] Choi, W. Y., D. Y. Kwak, I. H. Son and Y. T. Im, "Tetrahedral Mesh Generation Based on Advancing Front Technique and Optimization Scheme," *International Journal for Numerical Methods in Engineering*, 58 (2003), pp. 1857-1872.
- [12] Churchill, S., et al., "Morphological Variation and Airflow Dynamics in the Human Nose," *AM J HUM BIOL*, vol. 16(6), pp. 625-638, 2004

- [13] Defloriani, L. and E. Puppo, "An Online Algorithm for Constrained Delaunay Triangulation, Cvgip-Graphical Models and Image Processing," 54 (1992), pp. 290-300.
- [14] Elad, D., et al., "Analysis of Air Flow Patterns in the Human Nose", *Med Biol Eng Comput.*, vol. 31, pp. 585-592, 1993.
- [15] *FIDAP Theory Manual*, version 8.0, Fluent Inc.
- [16] Firestein, S., B. Darrow, and G. M. Shepherd, "Activation of the Sensory Current in Salamander Olfactory Receptor Neurons Depends on a G Protein Mediated Camp 2nd-Messenger System," *Neuron*, 1991. 6, 5, 825-835.
- [17] Firestein, S. and G.M. Shepherd, "A kinetic-Model of the Odor Response in Single Olfactory Receptor Neurons," *Journal of Steriod Biochemistry and Molecular Biology*, 1991. 39(4B): p.615-620.
- [18] Firestein, S., G. M. Shepherd, and F. S. Werblin, "Time Course of the Membrane Current Underlying Sensory Transduction in Salamander Olfactory Receptor Neurons," *J. Physiol.-London*, 1990. 430, 135-158.
- [19] Firestein, S. and F. Werblin, "Odor-Induced Membrane Currents in Vertebrate-Olfactory Receptor Neurons," *Science*, 1989. 244, 4900, 79-82.
- [20] Firestein, S. and F. Werblin, "Receptor Events and Transduction in Taste and Olfaction," In J. G. Brand, J. H. Teeter, R. H. Cagan and M. R. Kare (ed.), *Chemical Senses 1*. Marcel Dekker, New York, 1989. pp. 449-467.
- [21] Frakes, D.H., et al., "Application of An Adaptive Control Grid Interpolation Technique to Morphological Vascular Reconstruction," *IEEE Transactions on Biomedical Engineering*, 2003. 50(2): p. 197-206.
- [22] Freitag, L. A. and P. M. Knupp, "Tetrahedral Mesh Improvement via Optimization of the Element Condition Number," *International Journal for Numerical Methods in Engineering*, 53 (2002), pp. 1377-1391.
- [23] Freitag, L. A. and C. OllivierGooch, "Tetrahedral Mesh Improvement using Swapping and Smoothing," *International Journal for Numerical Methods in Engineering*, 40 (1997), pp. 3979-4002.
- [24] Fried, I., "Condition of Finite Element Matrices Generated from Nonuniform Meshes," *AIAA Journal*, 10 (1972), pp. 219-.
- [25] Frings, S., "Chemolectrical Signal Transduction in Olfactory Sensory Neurons of Air-Breathing Vertebrates," *Cellular and Molecular Life Sciences*, 2001. 58(2001): p. 510-519.
- [26] George, P. L. "Improvements on Delaunay-based Three-Dimensional Automatic Mesh Generator," *Finite Elements in Analysis and Design*, 25 (1997), pp. 297-317.

- [27] Gesteland, R.C., J.Y. Lettvin, and W.H. Pitts, "Chemical Transmission in the Nose of the Frog. *Journal of Physiology*," 1965. 181: p. 525-559.
- [28] Grevera, G.J. and J.K. Udupa, "Shape-Based Interpolation of Multidimensional Grey-Level Images," *IEEE Transactions on Medical Imaging*, 1996. 15(6): p. 881-892.
- [29] Girardin, M., E. Bilgen, P. Arbour, "Experimental Study of Velocity Fields in a Human Nasal Fossa by Laser Anemometry," *Prog. Neurobiol*, 1984. 23: p. 317-345.
- [30] Hahn, I., P.W. Scherer, and M.M. Mozell, "A Mass Transport Model of Olfaction," *Journal of Theoretical Biology*, 1994. 167: p. 115-128.
- [31] Hahn, I., P.W. Scherer, and M.M. Mozell, "Velocity Profiles Measured for Airflow Through a Large Scale Model of the Human Nasal Cavity", *Bull. Math. Biol.*, vol. 75(5), pp. 2273-2287,1993.
- [32] Hahn, I., P.W. Scherer, and M.M. Mozell, "Velocity Profiles Measured for Airflow Through a Large Scale Model of the Human Nasal Cavity," *Bull. Math. Biol.*, 1993. 75(5): p. 2273-2287.
- [33] Herman, G.T. and C.A. Bucholtz, "Shape-Based Interpolation Using a Chamfer Distance," *Lecture Notes in Computer Science*, 1991. 511: p. 314-325.
- [34] Herman, G.T., J.S. Zheng, and C.A. Bucholtz, "Shape-Based Interpolation," *IEEE Computer Graphics and Applications*, 1992. 12(3): p. 69-79.
- [35] Higashino, S. and S.F. Takagi, "The Effect of Electrotonus on the Olfactory Epithelium," *J.gen. Physiol.*, 1964. vol. 48: p. 323-335.
- [36] Higgins, W.E., C. Morice, and E.L. Ritman, "Shape-Based Interpolation of Tree-Like Structures in 3-Dimensional Images," *IEEE Transactions on Medical Imaging*, 1993. 12(3): p. 439-450.
- [37] Horn, B.K.P. and B.G. Schunck, "Determining Optical-Flow," *Artificial Intelligence*, 1981. 17(1-3): p. 185-203.
- [38] Hornung, D.E. and M.M. Mozell, "Accessibility of Odorant Molecules to the Receptors," *Biochemistry of Taste and Olfaction*, 1981(New York: Academic Press): p. 33-45.
- [39] Hornung, D.E., et al., "Airflow Patterns in a Human Nasal Model," *Arch Otolaryngol Head Neck Surg*, 1987. 113(2): p. 169-72.
- [40] Evans, Howard E., "Miller's Anatomy of the Dog," third edition(Philadelphia, Pa. : W.B. Saunders), 1993.

- [41] Jin, H. and R. I. Tanner, "Generation of Unstructured Tetrahedral Meshes by Advancing Front Technique," *International Journal for Numerical Methods in Engineering*, 36 (1993), pp. 1805-1823.
- [42] Joe, B., "Three-Dimensional Triangulations from Local Transformations," *SIAM Journal on Scientific and Statistical Computing*, 10 (1989), pp. 718 - 741.
- [43] Keyhani, K., P.W. Scherer, and M.M. Mozell, "Numerical Simulation of Air Flow in the Human Nasal Cavity", *Journal of Biomechanical Engineering*, vol. 117, pp. 429-441,1995.
- [44] Kimbell, J.S., M. E. Andersen, K. T. Morgan, "The Role of Airflow in Nasal Pathology," *Biofluid Mechanics 3: Proceedings of the Third Mid-Atlantic Conference on Biofluid Mechanics*, University Press, New York, 1990. pp. 3-12.
- [45] Kimbell, J.S., E. A. Gross, D. R. Joyner, "Application of Computational Fluid Dynamics to Regional Dosimetry of Inhaled Chemicals in the Upper Respiratory Tract of the Rat," *Toxicol. Appl. Pharmacol.*, Vol. 121 1993, pp. 253-263.
- [46] Kimmelman, C. P. "The Problem of Nasal Obstruction," *Otolaryngologic Clinics of N. Amer.*, Vol. 22 1989, pp. 265-278.
- [47] Kleene, S. J. and R. C. Gesteland, "Calcium-Activated Chloride Conductance in Frog Olfactory Cilia," *J. Neurosci.*, 1991. 11, 11, 3624-3629.
- [48] Kurahashi, T. and A. Menini, "Mechanism of Odorant Adaptation in the Olfactory Receptor Cell," *Nature*, 1997. 385, 6618, 725-729.
- [49] Kurahashi, T. and T. Shibuya, "Membrane Responses and Permeability Changes to Odorants in the Solitary Olfactory Receptor-Cells of Newt," *Zool. Sci.*, 1989. 6, 1, 19-30.
- [50] Kurahashi, T. and K. W. Yau, "Coexistence of Cationic and Chloride Components in Odorant-Induced Current of Vertebrate Olfactory Receptor-Cells," *Nature*, 1993. 363, 6424, 71-74.
- [51] Lamb, T. D. and E. N. Pugh, "G-Protein Cascades - Gain and Kinetics. *Trends Neurosci.*" 1992. 15, 8, 291-298.
- [52] Lee, T.Y. and W.H. Wang, "Morphology-Based Three-Dimensional Interpolation," *IEEE Transactions on Medical Imaging*, 2000. 19(7): p. 711-721.
- [53] Leopold, D. A., "The Relationship Between Nasal Anatomy and Human Olfaction," *Laryngoscope*, Vol. 98, 1988, pp. 1232-1238.
- [54] Ljung, L., "System Identification - Theory for the User," 1999, 2nd, Prentice-Hall, Upper Saddle River, NJ.

- [55] Lohner, R., "Automatic Unstructured Grid Generators," *Finite Elements in Analysis and Design*, 25 (1997), pp. 111-134.
- [56] Lohner, R., "Progress in Grid Generation via the Advancing Front Technique," *Engineering with Computers*, 12 (1996), pp. 186-210.
- [57] Lorensen, W. and H. Cliene, "Marching Cubes: A High Resolution 3-D Surface Construction Algorithm," *Computer Graphics*, 21 (1987), pp. 163-169.
- [58] Marchand R., S. Riad and A. ElshabiniRiad, "A Combined Octant/Delaunay Method for Fully automatic 3D Mesh Generation with Multiple Level Octant Differences," *Communications in Numerical Methods in Engineering*, 12 (1996), pp. 343-349.
- [59] Masing, H., "Investigations about the Course of Flow in the Nose Model," *Arch. Klin. Exp. Ohr. Nas. Kehlkopf.*, Vol. 189, 1967, pp. 371-381.
- [60] Maur, P. and I. Kolingerova, "The Employment of Regular Triangulation for Constrained Delaunay Triangulation," *Computational Science and Its Applications - Iccsa 2004, Pt 3*, 2004, pp. 198-206.
- [61] Nachber, R. B. and T. H. Morton, "A Gas Chromatographic Model for the Sense of Smell. Variation of Olfactory Sensitivity with Conditions of Stimulation," *J. Theor. Biol.*, 1981. 89, 387-407.
- [62] Nakamura, T. and G. H. Gold, "A Cyclic Nucleotide-Gated Conductance in Olfactory Receptor Cilia," *Nature*, 1987. 325, 6103, 442-444.
- [63] Nielson, G. M. and B. Hamann, "The Asymptotic Decider: Resolving the Ambiguity in Marching Cubes," *IEEE Visualization* (1991), pp. 83-91.
- [64] Nielson, G. M. and J. Sung, "Interval Volume Tetrahedrization, Visualization," Phoenix, Arizona, 1997, pp. 221-228.
- [65] Ning, P. and J. Bloomenthal, "An Evaluation of Implicit Surface Tilers," *IEEE Computer Graphics and Applications*, 13 (1993), pp. 33-41.
- [66] Ottoson, D., "Analysis of the Electrical Activity of the Olfactory Epithelium," *Actaphysiol. Scand*, 1956. 35(suppl. 122): p. 1-83.
- [67] Ottoson, D, "The Electro-Olfactogram. Handbook of Sensory Physiology (Olfaction)," Springer-Verlag, Berlin, 1971 pp. 96-131.
- [68] Parthasarathy, V. N. "Parthasarathy and S. Kodiyalam, A Constrained Optimization Approach to Finite Element Mesh Smoothing," *Finite Elements in Analysis and Design*, 9 (1991), pp. 309-320.

- [69] Poggio, T., V. Torre, and C. Koch, "Computational Vision and Regularization Theory," *Nature*, 1985. 317(6035): p. 314-319.
- [70] Press, W.H.F., B.P., Teukolsky, S.A. Vetterling, W.T., "Numerical Recipes in C: The Art of Scientific Computing," 2nd ed. 1992, Cambridge, England: Cambridge Univ. Press.
- [71] Proctor, D.F. and I. Anderson, "The Upper Airway" the Nose," 1982, New York: Elsevier Biomedical Press. 23-43.
- [72] Proetz, A.W., "Applied Physiology of the Nose," 2nd ed. 1953, St. Louis, MO: Annals.
- [73] Proetz, A.W., "Air Currents in the Upper Respiratory Tract and their Clinical Importance," 2nd ed. 1953, St. Louis, MO: Annals.
- [74] Ravi, D., "A New Active Contour Model for Shape Extraction," *Mathematical Methods in the Applied Sciences*, 2000. 23(8): p. 709-722.
- [75] Rassineux, A., "3D mesh adaptation. Optimization of Tetrahedral Meshes by Advancing Front Technique," *Computer Methods in Applied Mechanics and Engineering*, 141 (1997), pp. 335-354.
- [76] Raya, S.P. and J.K. Udupa, "Shape-Based Interpolation of Multidimensional Objects," *IEEE Transactions on Medical Imaging*, 1990. 9(1): p. 32-42.
- [77] Axel, Richard, "The Molecular Logic of Smell," *Scientific American*, Oct. 1995. ,154-159.
- [78] Sarangapani, R. and A. Wexler, "Modeling Particle Deposition in Extrathoracic Airways," *Aerosol Science and Technology*, 2000. 32: p. 72-89.
- [79] Scherer, P.W., et al., "Correlations Between Flow Resistance and Geometry in a Model of the Human Nose," *Jap. J. Physiol*, vol. 75(4): pp. 1767-1775, 1993.
- [80] Shephard, M. S. and M. K. Georges, "Automatic 3-Dimensional Mesh Generation by the Finite Octree Technique," *International Journal for Numerical Methods in Engineering*, 32 (1991), pp. 709-749.
- [81] Shewchuk, J. R., "Delaunay Refinement Algorithms for Triangular Mesh Generation," *Computational Geometry-Theory and Applications*, 22 (2002), pp. 21-74.
- [82] Shibuya, T. and S. Shibuya, "Olfactory Epithelium: Unitary Responses in the Tortoise," *Science*, 1963. 140, 495-496.
- [83] Simmen, D., J. Scherrer, and B. Heinz, "Biophysics of Nasal Airflow: A Review," *Archives OF Otolaryngology-Head and Neck Surgery*, vol. 125(9), pp. 1015-1021, 1999.

- [84] Sinnarajah, S., C. W. Dessauer, D. Srikumar, J. Chen, J. Yuen, S. Yilma, J. C. Dennis, E. E. Morrison, V. Vodyanoy, and J. H. Kehrl, "RGS2 Regulates Signal Transduction in Olfactory Neurons by Attenuating Activation of Adenylyl Cyclase III," *Nature*, 2001. 409, 6823, 1051-5.
- [85] Sklar, P. B., R. R. H. Anholt, and S. H. Snyder, "The Odorant-Sensitive Adenylate-Cyclase of Olfactory Receptor-Cells - Differential Stimulation by Distinct Classes of Odorants," *J. Biol. Chem.*, 1986. 261, 33, 5538-5543.
- [86] Steiglitz, K., "Introduction to Discrete Systems," 1974 (John Wiley and sons).
- [87] Swift, D.L. and D.F. Proctor, "Access of Air to the Respiratory Tract, in Respiratory Defence Mechanisms," 1977, Dekker: New York. p. 63-93.
- [88] Szeliski, R., "Fast Surface Interpolation Using Hierarchical Basis Functions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1990. 12(6): p. 513-528.
- [89] Szeliski, R. and H.Y. Shum, "Motion Estimation with Quadtree Splines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1996. 18(12): p. 1199-1210.
- [90] Szeliski, R. and J. Coughlan, "Spline-Based Image Registration," *International Journal of Computer Vision*, 1997. 22(3): p. 199-218.
- [91] Szeliski, R. et al., "Fast Surface Interpolation Using Hierarchical Basis Functions", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12(6), pp. 513-528, 1990.
- [92] Takagi, S.F. and K. Omura, "Responses of the Olfactory Receptor Cells to Odors," *Proc. Jap. Acad.*, 1963. 39: p. 253-255.
- [93] Takagi, S.F. and T. Shibuya, "The 'on' and 'off' Responses Observed in the Lower Olfactory Pathway," *Jap. J. Physiol.*, 1960. 10: p. 99-105.
- [94] Takagi, S.F. and T. Shibuya, "The Electrical Activity of the Olfactory Epithelium Studied with Micro- and Macro-Electrodes," *Jap. J. Physiol.*, 1960. 10: p. 385-395.
- [95] Takagi, S.F., T. Shibuya, S. Higashino, and T. Arai, "The Stimulative and Anaesthetic Actions of Ether on the Olfactory Epithelium of the Frog and the Toad," *Jap. J. Physiol.*, 1960. 10: p. 571-584.
- [96] Tarabichi, M. and N. Fanous, "Finite Element Analysis of Airflow in the Nasal Valve", *Arch Otolaryngol Head Neck Surg*, vol. 119, pp. 169-172, 1993.
- [97] Wang, K., T.S. Denney, E. E. Morrison, V. J. Vodyanoy, "Numerical Simulation of Air Flow in the Human Nasal Cavity," *27th IEEE EMBS International Conference*, Shanghai, China, 2005.

- [98] Wang,K., T.S. Denney, E. E. Morrison, V. J. Vodyanoy, “Application of Hierarchical Spline-Based Image Registration to Human Nasal Cavity Reconstruction,” *the 2005 Annual Fall Meeting of the Biomedical Engineering Society (BMES 2005)*, Baltimore, Maryland, September 30 2005.
- [99] Wang,K., T.S. Denney, E. E. Morrison, V. J. Vodyanoy, “Construction of 3-D Finite Element Meshes from Computed Tomography Data,” *27th IEEE EMBS International Conference*, Shanghai, China, 2005.
- [100] Watson, D. F., “Computing the N-Dimensional Delaunay Tessellation with Application to Voronoi Polytopes,” *Computer Journal*, 24 (1981), pp. 167-172.
- [101] Yerry, M. A. and M. S. Shephard, “Automatic 3-Dimensional Mesh Generation by the Modified-Octree Technique,” *International Journal for Numerical Methods in Engineering*, 20 (1984), pp. 1965-1990.