


May 2019

# CAD-Based Porous Scaffold Design of Intervertebral Discs in Tissue Engineering

Ye Guo

*University of Wisconsin-Milwaukee*

Follow this and additional works at: <https://dc.uwm.edu/etd>

 Part of the [Computer Sciences Commons](#), and the [Electrical and Electronics Commons](#)

---

## Recommended Citation

Guo, Ye, "CAD-Based Porous Scaffold Design of Intervertebral Discs in Tissue Engineering" (2019). *Theses and Dissertations*. 2072.  
<https://dc.uwm.edu/etd/2072>

This Dissertation is brought to you for free and open access by UWM Digital Commons. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of UWM Digital Commons. For more information, please contact [open-access@uwm.edu](mailto:open-access@uwm.edu).

**CAD-BASED POROUS SCAFFOLD DESIGN OF  
INTERVERTEBRAL DISCS IN TISSUE  
ENGINEERING**

by

Ye Guo

A Dissertation Submitted in  
Partial Fulfillment of the  
Requirements for the Degree of

Doctor of Philosophy

in Engineering

at

The University of Wisconsin–Milwaukee

May 2019

# ABSTRACT

## CAD-BASED POROUS SCAFFOLD DESIGN OF INTERVERTEBRAL DISCS IN TISSUE ENGINEERING

by

Ye Guo

The University of Wisconsin–Milwaukee, 2019

Under the Supervision of Professor Zeyun Yu

With the development and maturity of three-dimensional (3D) printing technology over the past decade, 3D printing has been widely investigated and applied in the field of tissue engineering to repair damaged tissues or organs, such as muscles, skin, and bones. Although a number of automated fabrication methods have been developed to create superior bio-scaffolds with specific surface properties and porosity, the major challenges still focus on how to fabricate 3D natural biodegradable scaffolds that have tailor properties such as intricate architecture, porosity, and interconnectivity in order to provide the needed structural integrity, strength, transport, and ideal microenvironment for cell- and tissue-growth. In this dissertation, a robust pipeline of fabricating bio-functional porous scaffolds of intervertebral discs based on different innovative porous design methodologies is illustrated. Firstly, a triply periodic minimal surface (TPMS) based parameterization method, which has overcome the integrity problem of traditional TPMS method, is presented in Chapter 3. Then, an implicit surface modeling (ISM) approach using tetrahedral

implicit surface (TIS) is demonstrated and compared with the TPMS method in Chapter 4. In Chapter 5, we present an advanced porous design method with higher flexibility using anisotropic radial basis function (ARBF) and volumetric meshes. Based on all these advanced porous design methods, the 3D model of a bio-functional porous intervertebral disc scaffold can be easily designed and its physical model can also be manufactured through 3D printing. However, due to the unique shape of each intervertebral disc and the intricate topological relationship between the intervertebral discs and the spine, the accurate localization and segmentation of dysfunctional discs are regarded as another obstacle to fabricating porous 3D disc models. To that end, we discuss in Chapter 6 a segmentation technique of intervertebral discs from CT-scanned medical images by using deep convolutional neural networks. Additionally, some examples of applying different porous designs on the segmented intervertebral disc models are demonstrated in Chapter 6.

# TABLE OF CONTENTS

<b>Abstract.....</b>	<b>ii</b>
<b>1 Introduction.....</b>	<b>1</b>
<b>2 Summary of Existing Porous Design Methods.....</b>	<b>7</b>
<b>2.1 Regular Porous Structure Design.....</b>	<b>8</b>
2.1.1 CAD-Based Methods.....	8
2.1.2 Image-Based Methods.....	9
2.1.3 Implicit Surface Modeling Methods.....	10
2.1.4 Space-filling Curves.....	10
<b>2.2 Irregular Porous Structure Design.....</b>	<b>11</b>
<b>2.3 Triply Periodic Minimal Surface.....</b>	<b>12</b>
<b>3 Porous Structure Design Using Parameterized Hexahedral Meshes and Triply Minimal Surfaces.....</b>	<b>18</b>
<b>3.1 Related Work.....</b>	<b>19</b>
3.1.1 Field Function.....	20
3.1.2 Solving Geodesic Lines Using Heat Conduction Model.....	22
<b>3.2 Method Description.....</b>	<b>23</b>
3.2.1 Construction of Hexahedral Edges.....	24
3.2.2 Construction of Hexahedral Faces.....	26
3.2.3 Surface Meshing.....	29
3.2.4 Volume Meshing.....	30
3.2.5 Reconstruction of Hexahedral Parameterized TPMS Surface.....	31
<b>3.3 Experimental Results.....</b>	<b>35</b>
<b>3.4 Summary.....</b>	<b>42</b>
<b>4 Porous Structure Design Using Tetrahedral Implicit Surfaces.....</b>	<b>44</b>
<b>4.1 Tetrahedral Meshing and Smoothing.....</b>	<b>45</b>
<b>4.2 Method Description.....</b>	<b>48</b>
4.2.1 Mathematical Description of Tetrahedral Implicit Surface.....	48

4.2.2	Continuity Analysis of Tetrahedral Implicit Surface.....	51
4.2.3	Porous Structure Generation from Surface Models.....	53
<b>4.3</b>	<b>Experimental results.....</b>	<b>54</b>
<b>4.4</b>	<b>Summary.....</b>	<b>66</b>
<b>5</b>	<b>Porous Structure Design Using Anisotropic Radial Basis Function.....</b>	<b>68</b>
<b>5.1</b>	<b>Traditional Radial Basis Function (RBF) Interpolation.....</b>	<b>69</b>
<b>5.2</b>	<b>Anisotropic Radial Basis Function (ARBF) Interpolation.....</b>	<b>71</b>
<b>5.3</b>	<b>Algorithm Description.....</b>	<b>73</b>
<b>5.4</b>	<b>Experimental Results.....</b>	<b>77</b>
<b>5.5</b>	<b>Summary.....</b>	<b>82</b>
<b>6</b>	<b>Segmentation of Intervertebral Discs and Application of Porous Design for Intervertebral Disc Scaffolds.....</b>	<b>84</b>
<b>6.1</b>	<b>Overview of Approaches for Segmentation of Intervertebral Disc.....</b>	<b>85</b>
6.1.1	Traditional Methods.....	85
6.1.2	Machine Learning Methods.....	86
<b>6.2</b>	<b>Segmentation of Intervertebral Disc Using U-Net.....</b>	<b>88</b>
6.2.1	Multimodal Dataset.....	89
6.2.2	Data Augmentation.....	93
6.2.3	2D U-Net.....	94
6.2.4	3D U-Net.....	96
6.2.5	Training.....	100
6.2.6	Evaluation Metrics.....	101
6.2.7	Experimental Results.....	102
<b>6.3</b>	<b>Porous Intervertebral Disc.....</b>	<b>109</b>
<b>7</b>	<b>Conclusion.....</b>	<b>116</b>
	<b>References.....</b>	<b>117</b>
	<b>Appendix.....</b>	<b>128</b>
	<b>Curriculum Vitae.....</b>	<b>129</b>

# LIST OF FIGURES

1.1	The structure of partial spine. <a href="https://www.spineuniverse.com/anatomy/intervertebral-discs">https://www.spineuniverse.com/anatomy/intervertebral-discs</a> . . . . .	1
2.1	Schematic diagram illustrating a TPMS that divides the unit cell into two phases [24]. . . . .	15
2.2	Common TPMS-based unit cell libraries. (a) TPMS P surface. (b) TPMS D surface. (c) TPMS G surface. . . . .	16
2.3	Porous femur bone scaffold designed by TPMS. From the close-up view, one can see broken and deformed structures of the surface layer. . . . .	17
3.1	The framework of the proposed approach. . . . .	23
3.2	The cube of labeled vertices and edges. . . . .	25
3.3	The in-plane based seed searching algorithm. . . . .	27
3.4	Parametric mapping. . . . .	32
3.5	Tri-linear interpolation. . . . .	34
3.6	Results of extractions of Geodesic lines and in-plane points. (a) Original mesh. (b) 12 Geodesic lines. (c) Extraction of in-plane points. . . . .	36
3.7	Process of parameterization of femur bone. (a) W-axis-oriented iso-parametric lines ( $5 \times 7 \times 20$ ). (b) W-axis-oriented iso-parametric lines ( $5 \times 10 \times 60$ ). (c) Cross planes on the W-axis ( $5 \times 7 \times 20$ ). (d) Cross planes on the W-axis ( $5 \times 10 \times 60$ ). (e) Hexahedral surface mesh ( $5 \times 7 \times 20$ ). (f) Hexahedral surface mesh ( $5 \times 10 \times 60$ ). (g) Larger views of (e) and (h) respectively. . . . .	38
3.8	Parameterized hexahedral surface imbedded with TPMS P porous inner structure. (a) Parameterized TPMS P surface mesh ( $5 \times 7 \times 20$ ). (b) Parameterized TPMS P surface mesh ( $5 \times 10 \times 60$ ). (c) Larger views of (a) and (b) respectively. . . . .	39
3.9	Effects of generation of parameterized molecule model. (a) Original mesh. (b) 12 Geodesic lines. (c) Extraction of in-plane points. . . . .	40
3.10	Effects of generation of parameterized dog bone. (a) Original mesh. (b) 12 Geodesic lines. (c) Extraction of in-plane points. . . . .	41
4.1	The framework of the proposed TIS method. . . . .	45
4.2	An octree topology. . . . .	46
4.3	Tetrahedral meshing on 3D femur model. (a) Tetrahedral meshing on the femur model. (b) Cross-section of the tetrahedralized femur model. . . . .	48
4.4	The new coordinates system using scaled point-to-plane distances. . . . .	50
4.5	Proof of continuity property between adjacent tetrahedrons. . . . .	51
4.6	Flowchart showing the procedures for designing a 3D heterogeneous porous scaffold using the TIS method. . . . .	54
4.7	TPMS Schwarz P surface (left two columns) vs. TIS P surface (right two columns). . . . .	56

4.8	TPMS Schwarz D surface (left two columns) vs. TIS D surface (right two columns). . . . .	56
4.9	TPMS Schwarz G surface (left two columns) vs. TIS G surface (right two columns). . . . .	57
4.10	Two views of the TIS P surface generated from two connecting tetrahedrons. . . . .	57
4.11	Two views of the TIS D surface generated from two connecting tetrahedrons. . . . .	57
4.12	Two views of the TIS G surface generated from two connecting tetrahedrons. . . . .	58
4.13	TIS surfaces generated from an icosahedron (P surface – top, D surface - bottom). . . . .	58
4.14	Distribution maps of the mean curvature and corresponding histograms for TPMSs and TISs. (a) TPMS P-surface. (b) TPMS D-surface. (c) TPMS G-surface. (d) TIS P-surface. (e) TIS D-surface. (f) TIS G-surface. . . . .	60
4.15	Deformed TPMS and TIS surfaces with a threshold $C$ that exceeds the reasonable range. (a) TPMS P-surface ( $C=1.10$ ). (b) TPMS D-surface ( $C=1.10$ ). (c) TPMS G-surface ( $C=1.10$ ). (d) TIS P-surface ( $C=0.33$ ). (e) TIS D-surface ( $C=0.80$ ). (f) TIS G-surface ( $C=0.67$ ). . . . .	63
4.16	Porous femur generated by TIS P-surface. . . . .	64
4.17	Sectional view of porous femur generated by TIS P-surface. . . . .	65
5.1	2D interpolation schemes. $x$ is the pixel to be interpolated. Dashed circle (for RBF) or ellipses (for anisotropic RBF) describe the support domains of underlying basis functions. (a) RBF interpolation. (b) Anisotropic RBF interpolation. . . . .	71
5.2	Distance definitions used by ARBF. (a) Point $x$ is on line segment $(a, b)$ . (b) Point $x$ and line segment $(a, b)$ form an acute triangle. (c) Point $x$ and line segment $(a, b)$ form an obtuse triangle. (d) Distance between line segments $(a, b)$ and $(c, d)$ . . . . .	73
5.3	Values assigned to mesh nodes. Red dots indicate value 1 is assigned while blue dots indicate value -1 is assigned. Brown lines indicate value -1 is assigned as well. In 3D meshes, green dots represent interior nodes. (a) Sample distribution of node values in the 2D triangular mesh. (b) Sample distribution of node values in the 3D tetrahedral mesh. (c) Sample distribution of node values in the 3D hexahedral mesh. . . . .	74
5.4	ARFB global interpolation. . . . .	75
5.5	ARFB local interpolation. . . . .	76
5.6	Generated porous surfaces in regular tetrahedrons. (a) Surface in single regular tetrahedron. (b) Surface in two connecting tetrahedrons. (c) Surface in an icosahedron. . . . .	78
5.7	Generated porous surface for regular hexahedrons. (a) Surface in single hexahedron. (b) Surface in 2x2 stack of hexahedrons. (c) Surface in 4 hexahedrons arranged to form a rod. . . . .	78
5.8	Results obtained from icosahedron domain using different basis functions. (a) multiquadrics (MQ). (b) inverse multiquadrics (IMQ). (c) Gaussian. (d) thin plate spline (TPS). . . . .	79
5.9	Interpolated porous inner structure of femur scaffold. . . . .	80
5.10	Resulted meshes from hexahedron domain with disturbance. . . . .	81
5.11	Internal pore becomes larger with iso-value increased. . . . .	81
5.12	Relationships between porosity and different factors. . . . .	82



6.1	MR spinal images collected from different modalities. ....	92
6.2	2D U-Net architecture. ....	95
6.3	Framework of the proposed 3D neural network method. ....	97
6.4	Training 3D U-Net based on: (1) full-modality (left) (2) opp, wat, fat modalities (middle) (3) ground truth (right). ....	104
6.5	A comparison between the 2D and 3D methods. The images from leftmost column to rightmost column are respectively source images, ground truths, 2D prediction results, 3D prediction results, and merged contours of all results. ....	106
6.6	Effective analysis of data augmentation. ....	109
6.7	Segmented IVDs from spine column. ....	110
6.8	Different views of porous IVD generated by using the parameterization method. (a) Front view. (b) Side view. (c) Oblique view. ....	112
6.9	Different views of porous IVD generated by using the TIS method. (a) Front view. (b) Side view. (c) Oblique view. ....	113
6.10	Different views of porous IVD generated by using the ARBF method. (a) Front view. (b) Side view. (c) Oblique view. ....	115

# LIST OF TABLES

2.1	Categorization of porous scaffolds design techniques used in tissue engineering. ....	7
2.2	Various types of TPMS approximated with periodic surfaces composed of simple trigonometric functions. ....	14
3.1	The relationships between heat sources and their neighboring vertices along with directions of geodesic lines. ....	26
3.2	Numbering rules for hexahedral faces. ....	27
3.3	Run-time for each experimental node. ....	35
4.1	Statistical analysis to the curvatures of TPMSs and TISs. ....	61
4.2	Effect of diverse thresholds $C$ on different TPMS and TIS surfaces. ....	62
4.3	Surface areas of TPMSs and TISs for a unit volume. ....	63
4.4	Running time comparison between TIS and TPMS. ....	66
5.1	Common radial basis functions. ....	70
6.1	Statistical analysis to different image modalities. ....	92
6.2	3D segmentation network architecture. ....	99
6.3	Performance analysis of 3D segmentation using different combinations of image modalities. ....	103
6.4	Segmentation performance of 2D U-Net trained with images sliced from different directions (X-, Y-, Z-axis). ....	105
6.5	Segmentation result evaluation of the conventional 3D U-Net, UNICLK, UNIJLU, and our methods. ....	108

## ACKNOWLEDGEMENTS

Here I would like to take this opportunity to thank everyone who has witnessed my completion of the doctorate philosophy in Computer Science, especially to those who have helped me without expecting any paying back.

Firstly, I want to express my great gratitude to my advisor professor Zeyun Yu for his aspiring guidance and continuous support. I was so honored being a member of his biomedical modeling and visualization laboratory. He always tries every way to encourage and motivate me to accomplish better work and gives me the most care and understanding even on some trivial things in life.

My sincere thanks also go to my dissertation committee: Prof. Ichiro Suzuki, Prof. Rohit Kate, Prof. Dexuan Xie, and Prof. Zhi Zheng. Their insightful comments, unbiased guidance, and also the challenging questions have inspired me to widen my research from various perspectives.

Last but not the least, I would like to sincerely appreciate the support and love from my parents and my girlfriend. I have to admit that whenever I felt frustrated and even thought of giving up, their encouragements and cares always re-lighted the candle of hope in my journey of life.

By the end of my doctoral career, I cherish all the knowledge and guidance I have obtained at UW-Milwaukee and I am excitedly going to start a new chapter in my life.

# Chapter 1

## Introduction

From the perspective of ergonomics, an intervertebral disc (IVD) lies between adjacent vertebrae in the vertebral column. Each disc forms a fibrocartilaginous joint (a symphysis) to allow slight movement of the vertebrae and acts as a ligament to hold the vertebrae together [1]. The structure of the human spine is extremely complicated. There are normally 24 vertebrae in the adult spine. Due to the absence of IVD between the atlas and the sacral vertebrae, there are only 23 discs in the body. The intervertebral discs are located between every two of the vertebrae and the total thickness of the disc is ranged from 20% to 25% of the total length of the spine. Among them, the lumbar intervertebral disc in the thickest is about 9 millimeters [2]. Figure 1.1 gives an example of the intervertebral disc lying between two vertebrae.

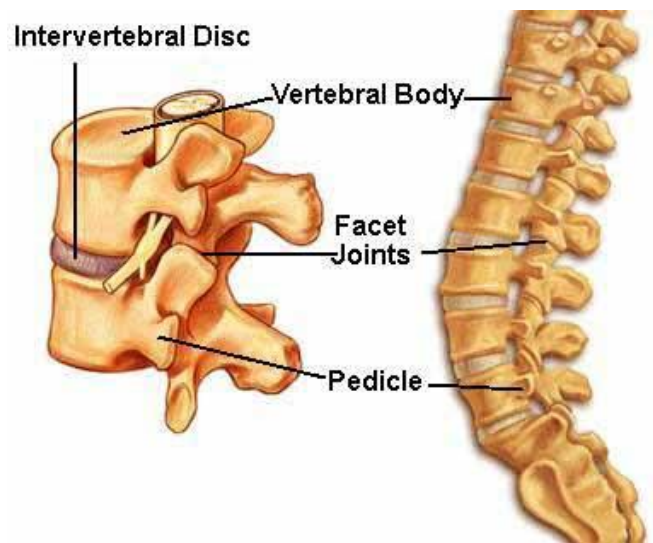


Figure 1.1 The structure of partial spine (by courtesy of [3]).

In recent years, the incidence of the prolapse of intervertebral discs has been increasing year by year and some complications caused by intervertebral disc problems are also becoming more and more common, such as sciatica caused by posterior protrusion of the intervertebral disc and disc atrophy. Millions of people suffer from intervertebral disc disease (IVDD) and its complications every year. People working in certain industries, such as drivers, construction workers, software programmers etc., are high-risk crowds with disc diseases. Moreover, the intervertebral disc may cause muscle tightness and inflammation of tendon due to aging, trauma, poor posture, and overwork. Most pain can be alleviated after proper posture correction, recuperation, and medication. Nonetheless, some patients still have persistent lower back pain (LBP) and the pain can extend to thighs and shanks. Additionally, some severe disc herniations can cause spinal canal, spinal stenosis, and nerve compression. For these cases, the surgery of disc replacement becomes inevitable and also the best option for the treatment. Consequently, finding a new alternative with same bio-features comes into a serious challenge.

With the birth of a new paradigm, 3D printing, the opportunity of building superior scaffolds with excellent bio-features has greatly increased. 3D printing (3DP), also known as additive manufacturing (AM), has provided robust technical supports for manufacturing high-quality bio-compatible intervertebral disc scaffolds. The latest American Society for Testing and Materials (ASTM) standards define 3D printing as “a process of joining materials to make objects from three-dimensional model data, usually layer upon layer, as opposed to subtractive manufacturing methodologies” [5]. The earliest 3D printing technology originated in Japan in the 1980s [4]. Later in 1993, Massachusetts Institute of Technology (MIT) obtained a patent for 3D printing technology. Ordinary printers used in everyday life can print flat items designed by

computers. The so-called 3D printers basically have the same working principle, but the printing materials are changed into metal, plastic, and ceramics to replace paper and ink. The working principles of 3D printing are firstly modeling by computer modeling software, and then “partitioning” the built three-dimensional model into a layer-by-layer cross-section, i.e. slicing, so as to guide the printer to print layer by layer. Some most common AM techniques include selective laser sintering (SLS), stereolithography (SLA), fused deposition modeling (FDM), precision extrusion deposition (PED), etc. [6 – 9].

With the help of these advanced 3D printing technologies, the fabrication of tailor scaffolds such as intervertebral disc is no longer an impossible task. Actually, 3D printing holds remarkable promise for tissue engineering (TE) as it can potentially provide a rapid and robust approach to assemble functional tissue in vitro. Tissue engineering is the use of a combination of cells, engineering and materials methods, and suitable biochemical and physiochemical factors to improve or replace biological tissues [10]. Tissue engineering researches mainly focus on four aspects: seed cells, biological materials, methods, and techniques for constructing tissues and organs, and clinical application of tissue engineering. At present, there are roughly three kinds of tissue repair methods commonly used in clinical practice: autologous tissue transplantation, allogeneic tissue transplantation, or application of artificial scaffolds. The creation of physiological tissue scaffolds involves the development of viable substitutes that restore, maintain and improve the function of human tissue. In the success of tissue engineering, 3D printing scaffolds have played critical roles as extracellular matrices onto which cells can attach, grow, and form new tissues. A few most common artificial scaffold materials include high polymers (carbon fiber, polyester, Teflon), metal materials (stainless steel, cobalt-based alloys, titanium alloys), bio-inert ceramics (Alumina, zinc oxide, silicon carbide), bioactive

ceramics (bioglass, hydroxyapatite, calcium phosphate), etc. These materials are characterized by high mechanical strength (abrasion resistance, fatigue resistance, non-deformation, etc.), biological inertness (acid, alkali, anti-aging, non-degradation). But there is a problem with reoperation, so then people began to focus on biodegradable and biologically active materials such as fibrin gel, collagen gel, polylactic acid, polyalcohol acid and its copolymer, polylactic acid and polyhydroxy acid, agarose, chitosan and hyaluronic acid, etc. Currently, most of the studied and used tissue scaffold materials are degradable materials or combinations of degraded and non-degraded materials [11].

In addition to the most basic material issues, the design of the disc scaffold also has played a critical role to determine the quality of the final product. In the natural world, most of the human tissues are anisotropic, complex structures. Heterogeneous inner pore surface is the creature of natural selection and evolution. In fact, the authentic tibia, femur, and intervertebral disc are all inhomogeneous structures with complicated fracture. Their micro-architectures significantly affect the development and some bio-functions. Ideally, a superior bio-functional porous scaffold should have at least following characteristics [25, 138-139]:

1. Biocompatible and bioresorbable. Cells must adhere and migrate onto the inner and outer surfaces to proliferate through the scaffold before laying down new matrix. The implanted scaffold must elicit a negligible immune reaction to prevent any inflammatory responses or rejections by the body.
2. Biodegradable with a controllable degradation and resorption rate. The scaffold must be biodegradable to allow cells to produce their own extracellular matrix. And the by-products of this degradation have to be non-toxic and able to exit the

body without interference with other organs. Moreover, the biodegradation is generally required to match the rate of new tissue formation;

3. Suitable surface microtopography and chemistry that promote cell attachment, proliferation and differentiation. Surface chemistry as well as surface microtopography determine whether protein molecules can adsorb and how cells attach and align themselves;
4. Three-dimensional porous structure with an appropriate porosity and an interconnected pore network for cell growth, nutrient transport and metabolite discharge. Usually, low porosity simulates osteogenesis by suppressing cell proliferation and force cell aggregation while high porosity results in greater cell ingrowth;
5. Mechanical properties to match those of surrounding tissue. The mechanical properties of the scaffold should be consistent with the anatomical site into which it is to be implanted. Some materials with good mechanical properties have the detriment of retaining high porosity and insufficient capacity for vascularization.

From these well-defined characteristics, one of the most important parameters in the manufacture of tissue engineering scaffolds is the creation of porous structure inside the scaffold because these pores directly affect the growth state of cells. For superior biomedical scaffold, the micro-architecture should be highly porous with interconnected pores of defined diameters, e.g. 200 – 900 micrometers as the diameter for bones, and also exhibit high surface area-to-volume ratio to allow high rates of mass transfer, cell in-growth, and vascularization [12]. And the internal communication pore allows the transport of nutrients and metabolic wastes from outside to the



inside of the system. Diverse pore structures also can act as a physiological mechanism to affect the movement and metabolism of cells.

On the other hand, even if we can perfectly master the porous design techniques, acquiring the 3D contour model of the IVDs becomes another obstacle of fabricating tailored disc scaffold for specific patient. The vertebral column is a complex anatomical construct, composed of vertebrae and intervertebral discs supported by ligaments and muscles. Due to the special relationship between vertebrae column and IVDs, the automated recognition of IVDs becomes hard for implementation. And the difficulties mainly reflected in three aspects [103]:

- 1) Multiple modalities. The image resolution, contrast, and appearance for the same spine structure could be very different when it is exposed to MR/CT, or T1/T2 weighted MR images;
- 2) High repetition. The appearances of vertebrae and intervertebral discs are highly repetitive that mismatching could happen easily;
- 3) Various poses. The vertebrae sizes and orientations are highly diverse in pathological data that regular detectors such as appearance detectors are insufficient to match all vertebrae;

Furthermore, for some unusual conditions of pathological cases, e.g. the abnormal spine curvature and bright visual imaging artifacts caused by metal implants, the difficulties of accurate localization and segmentation of IVDs are greatly increased correspondingly.

In this article, a variety of porous design methods will be discussed in separate chapters. Additionally, a few art-of-the-state segmentation approaches based on machine learning will also be analyzed to form a complete pipeline of creating superior functional porous IVD scaffolds.

# Chapter 2

## Summary of Existing Porous Design Methods

Various conventional methodologies are available for scaffold fabrications and they are generally classified into two major categories: Regular and Irregular Porous Designs [13] (see Table 2.1). In this chapter, you will have a more comprehensive understanding of different advanced porous design techniques.

Scaffold Type	Methods
<b>Regular porous scaffolds</b>	CAD-based methods [14-16] Image-based methods [17-19] Implicit surface modeling [20-25] Space-filling curves [26]
<b>Irregular porous scaffolds</b>	An optimization method proposed by [27] Stochastic methods using Voronoi models [28-29] A hybrid Voronoi-spline method [30] Methods using volumetric meshes [31]

Table 2.1: Categorization of porous scaffolds design techniques used in tissue engineering.

## **2.1 Regular Porous Structure Design**

The physiological structure of native tissues is inherently heterogeneous and complex. Instead of trying to exactly reproduce their internal microarchitecture, literature is mainly focused on the creation of simplified models that are functionally equivalent to the tissue to be repaired in terms of porosity and mechanical properties. Different elemental units with well-customized mechanical and transport properties have been discussed in design of functional porous scaffolds. The libraries of unit pore structure are designed based on pixel, voxel, and wireframe mesh. And the 3D model is used to be created either using the computer-aided software or image-based programs. By aggregating these repetitive base units, various scaffolds with tunable properties and architectures could be manufactured. In this section, the details of a group of methodologies for regular porous structure design are presented.

### **2.1.1 CAD-Based Methods**

Most of commercial CAD tools, such as NX (Siemens PLM Software), CATIA (Dassault Systèmes), Pro/Engineer (PTC), SolidWorks (Dassault Systèmes) and MIMICS (Materialize Gmbh), are supporting complicated model design based on solid or surface modeling systems. Constructive solid geometry (CSG) and boundary representation (B-Rep) are two most prevalent geometric modeling tools. CSG-based software attempt to produce complex structure by combining standard solid primitives (cylinders, spheres or cubes) through regular Boolean operations (e.g. intersection). However, B-Rep methods describe the solid based on the geometric boundary information like vertices, edges, and loops and without specifying their explicit relations. Additionally, a preliminary check is usually required to verify that neither gaps

or overlaps exist among the boundaries of targets [14]. Conventionally, B-Rep models require more storage space but less computation time than the CSG ones. As the inner architectures of objects come to larger and more complicated, their dramatically increasing size also makes it much harder for visualization and manipulation. To overcome this limitation, additional solid unit cells with more bio-inspired features were introduced by [15-16].

### **2.1.2 Image-Based Methods**

Image-based methods prefer to combine image processing and free-form fabrication techniques for modeling 3D scaffold geometries. Due to their compatibility with real patient data, the image-based methods can quickly create porous architectures by intersecting two 3D binary images, among which one depicts the outline of the defect and the other one consisting of a stack of binary unit cells. Empirically derived geometries are created in a unit cell with basic geometric shapes (cylinders, spheres) to represent regular porous structures. While randomly arranged pores can be obtained by the usage of a random number generator to set voxel states. Among various image-based methods, the integration of topological optimization algorithms has been proved pivotal to obtain smooth superior porous scaffolds [17-18]. Hollister et al. defined the craniofacial scaffold's topology by specifically setting the voxels' densities within an image design cube [18]. In addition, Smith et al. used both image-based design and CAD tools to generate a precisely sized and shaped scaffold for osseous tissue via selective laser sintering (SLS) [19].

### **2.1.3 Implicit Surface Modeling Methods**

Implicit surface modeling (ISM) is a highly flexible approach, recently proposed as a valid tool for the generation of cellular structures providing a compact representation of potentially complex surfaces. ISM methods allow scaffold architectures to be simply described using a single mathematical equation, with freedom to introduce different pore shapes and architectural features, including pore size and porosity etc. In general, the ISM methods not only inherited the advantages of both traditional CSG and image-based methods but also have shown more efficient computational capacity for modeling and fabrication of biomedical scaffolds.

Due to the distinguishing properties of minimal surface in nature, a majority of ISM methods that are based on and derived from triply periodic minimal surfaces (TPMS) have attracted people's attention. An early attempt using the TPMS-based method to control tissue fabrication was first presented by [20]. Then, a majority of TPMSs, such as Schwartz's Primitive Surface (P-surface), Schwartz's Diamond Surface (D-surface), and Schoen's Gyroid Surface (G-surface), had been proved their efficacy in the high-precision fabrication of biomorphic scaffolds [21-23]. Nonetheless, their final products were all limited to simple cubic or cylindrical outer shape. Recently, an improved process for constructing a porous inner architecture within an arbitrary complex anatomical model was developed and successively optimized by Yoo [24-25], which we will spend more time to introduce in the later section.

### **2.1.4 Space-filling Curves**

Space-filling curves methodology is an improvement of extrusion-based techniques, which consist of the extrusion of a micro-diameter polymeric filament terminating with a nozzle having an orifice diameter in the hundreds of microns range. The fabrication process involves the

deposition of polymeric layers that adhere to each other by heating temperature while retaining their shape. This process leads to regular repetition of identical pores. Thus, these geometries have been named honeycomb-like patterns [26]. A few complex patterns can be obtained by changing the deposition angle between adjacent layers. Moreover, fractal space-filling curves, an enhanced version of space-filling curves, can be mathematically generated by starting with a simple pattern that grows through recursive rules.

## **2.2 Irregular Porous Structure Design**

Although regular periodic porous structures have enormous advantages in biomedical scaffold modeling such as controllable pore size and porosity etc., however, the main disadvantage lies in the difficulty of controlling the shape and distribution of the pores since slight modification of the unit cells will have an unexpected impact on the entire structure. Furthermore, current CAD tools are not suitable to design the complex scaffolds whose inner pore networks are randomized and non-repetitive. In real human tissue with variational porous architectures, discontinuities of deposition path planning are often discovered at the interface of two adjacent regions [32]. In order to increase the authenticity of generated pores, a few mechanical approaches have been proposed and analyzed. Mikos et al. demonstrated a native scaffold design technique using salt leaching [33]. Mooney et al. proposed to fabricate porous sponges of poly using gas-foaming [34]. Additionally, some other methods, such as thermal induced phase separation [35-36], porous ceramics [37], electrospinning [38-39], biomineralization [40-41], and phase-separation followed by freeze-drying [42], were also suggested for the manufacture of functionally graded scaffolds (FGSs). Recently, Khoda et al. proposed a CAD-method method using a micronozzle biomaterial deposition system to connect

differently spaced tool paths, which has improved continuity and connectivity between functionally graded regions [27]. Besides aforementioned methods, Stochastic and Voronoi models also have been used to mimics the randomness of porous structure in natural scaffold. Heterogeneous pores distributed according to a given porosity level are generated by stochastic methods in scaffold design [28-29]. To overcome the limitation that only simple spheres can be used to represent pores, Schaefer et al. introduced a hybrid Voronoi-spline representation combined with a random colloid-aggregation model [30]. The proposed method has been extended to implement graded porosities and porous distributions [31]. Volumetric mesh generators derived from finite element tools are used to create heterogeneous porosity within a solid model as well [32].

### **2.3 Triply Periodic Minimal Surface**

Among all of the state-of-the-art methodologies mentioned above, triply periodic minimal surface (TPMS) has attracted extensive considerations over the past decade due to its inherent advantages of geometrical and biological arrangement of pore structure. TPMS has inherited several attributes from nature: multi-functionality, mathematical precision, optimized topology for mechanical applications, and geometrical, thermal and electrical extremals [43-45]. It is noted that some mechanical and transport (electrical, thermal, and fluid) properties are competing in minimal surfaces. Not only applied in tissue scaffold manufacturing, TPMS also became a natural choice among many scientists working in the areas of mechanical, chemistry, biology etc.

A minimal surface is a surface that is locally area-minimizing, that is, a small piece has the smallest possible area for a surface spanning the boundary of that piece. Soap films are

minimal surfaces. The first example of TPMS were the surfaces described by Schwarz in 1865, followed by a surface described by his student E. R. Neovius in 1883 [46-47]. Later in 1970s, Alan Schoen came up with 12 new TPMSs based on skeleton graphs spanning crystallographic cells [48-49]. While Schoen's surfaces became popular in natural science the construction did not lend itself to a mathematical existence proof and remained largely unknown in mathematics, until H. Karcher proved their existence in 1989 [50].

The periodic implicit surface uses a single-valued function of three variables. The TPMS is the locus of points for which the function has some constant value. A zero-valued surface, known as zero set or level set, represents the interface regions of the space lying on or inside and outside the space. The zero level sets corresponding to the P, G, D, I-WP, F-RD, L, Tubular P, Tubular G, and I<sub>2</sub>-Y\*\* minimal surfaces can be described, to the first order of approximation, by the equations list in Table 2.2 [24].

TPMS	Periodic surface model
P	$\varphi(r) = \cos(X) + \cos(Y) + \cos(Z) = 0$
D	$\varphi(r) = \cos(X)\cos(Y)\cos(Z) - \sin(X)\sin(Y)\sin(Z) = 0$
G	$\varphi(r) = \sin(X)\cos(Y) + \sin(Z)\cos(X) + \sin(Y)\cos(Z) = 0$
I-WP	$\varphi(r) = 2[\cos(X)\cos(Y) + \cos(Y)\cos(Z) + \cos(Z)\cos(X)] - [\cos(2X) + \cos(2Y) + \cos(2Z)] = 0$
F-RD	$\varphi(r) = 4\cos(X)\cos(Y)\cos(Z) - [\cos(2X)\cos(2Y) + \cos(2X)\cos(2Z) + \cos(2Y)\cos(2Z)] = 0$



L	$\varphi(\mathbf{r}) = 0.5[\sin(2X)\cos(Y)\sin(Z) + \sin(2Y)\cos(Z)\sin(X) + \sin(2Z)\cos(X)\cos(Y)] - 0.5[\cos(2X)\cos(2Y) + \cos(2Y)\cos(2Z) + \cos(2Z)\cos(2X)] + 0.15 = 0$
Tubular P	$\varphi(\mathbf{r}) = 10[\cos(X) + \cos(Y) + \cos(Z)] - 5.1[\cos(X)\cos(Y) + \cos(Y)\cos(Z) + \cos(Z)\cos(X)] - 14.6 = 0$
Tubular G	$\varphi(\mathbf{r}) = 10[\cos(X)\sin(Y) + \cos(Y)\sin(Z) + \cos(Z)\sin(X)] - 0.5[\cos(2X)\cos(2Y) + \cos(2Y)\cos(2Z) + \cos(2Z)\cos(2X)] - 14 = 0$
I <sub>2</sub> -Y**	$\varphi(\mathbf{r}) = -2[\sin(2X)\cos(Y)\sin(Z) + \sin(X)\sin(2Y)\cos(Z) + \cos(X)\sin(Y)\sin(2Z)] + \cos(2X)\cos(2Y) + \cos(2Y)\cos(2Z) + \cos(2X)\cos(2Z) = 0$

Table 2.2 [24]: Various types of TPMS approximated with periodic surfaces composed of simple trigonometric functions.

TPMSs are built of fundamental units spanning the asymmetric domain in a given crystallographic symmetry group. For the TPMS, the domain is taken to be a unit cell of the periodically repeating structure. As shown in Figure 2.1 [24], let the surface of interest,  $\Gamma$ , be represented by the zero level set of  $\varphi$ . Then  $\Gamma$  divides the unit cell into the two distinct phases. We define the region where  $\varphi(\mathbf{r}) \leq 0$  to be phase 1, and the region where  $\varphi(\mathbf{r}) \geq 0$  to be phase 2, respectively. The surface generation will be accomplished by using the well-known marching cube algorithm. Figure 2.2 gives an overview of some common designed architectures in a unit cell composed of multiple triangular elements generated from marching cube algorithm.

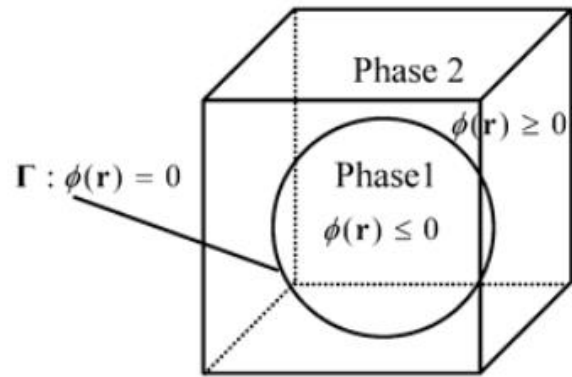
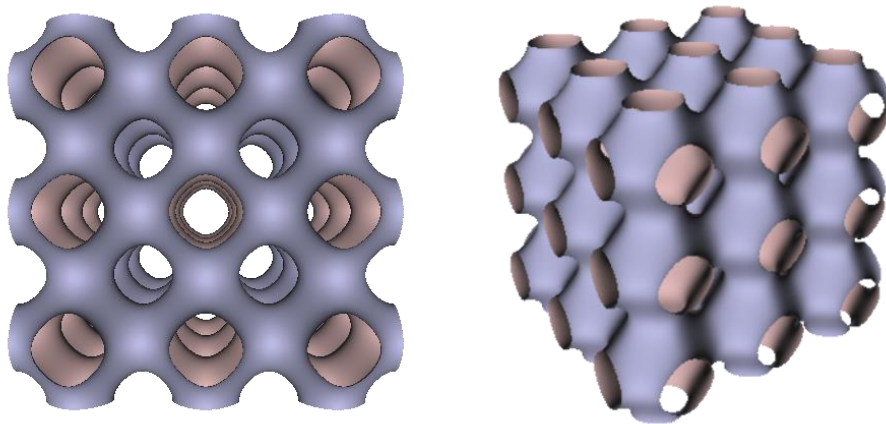
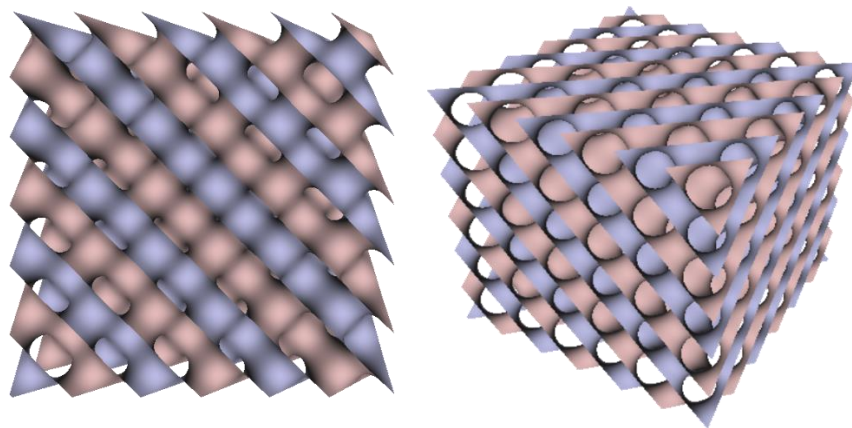


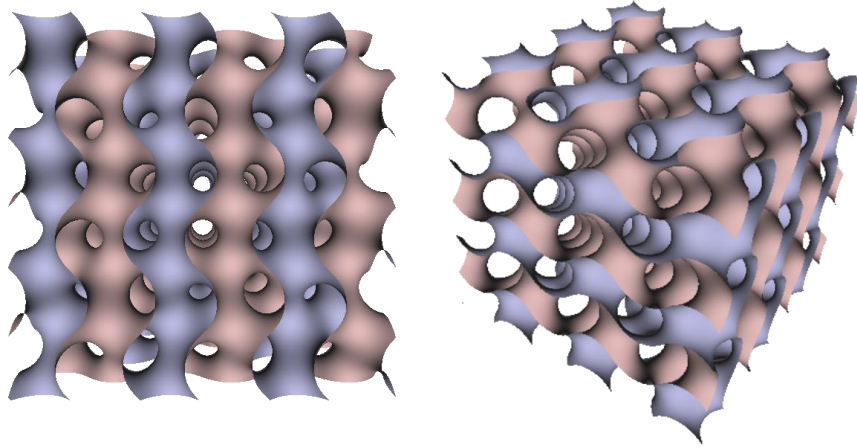
Figure 2.1 [24]: Schematic diagram illustrating a TPMS that divides the unit cell into two phases.



a) TPMS P surface



b) TPMS D surface



c) TPMS G surface

Figure 2.2: Common TPMS-based unit cell libraries.

Although, scaffold modeling using the TPMS is relatively mature, a few obvious challenges still remain unsolved regarding this method. For instance, incomplete surfaces may occur when resolutions of surfaces are seriously insufficient. As demonstrated in Figure 2.3, some structures from the superficial layer are broken and deformed. This situation becomes even worse when the resolution drops further. Additionally, the surface mesh between two adjacent distorted hexahedrons may not be smoothly continuous as in regular hexahedrons. Broken and deformed porous surfaces mainly exist in superficial part of a model when the continuity of the wrapping surface mesh is not guaranteed. To overcome such problems, a few mapping techniques, such as solid T-spline [51] and distance field [25] etc., are alternative solutions.

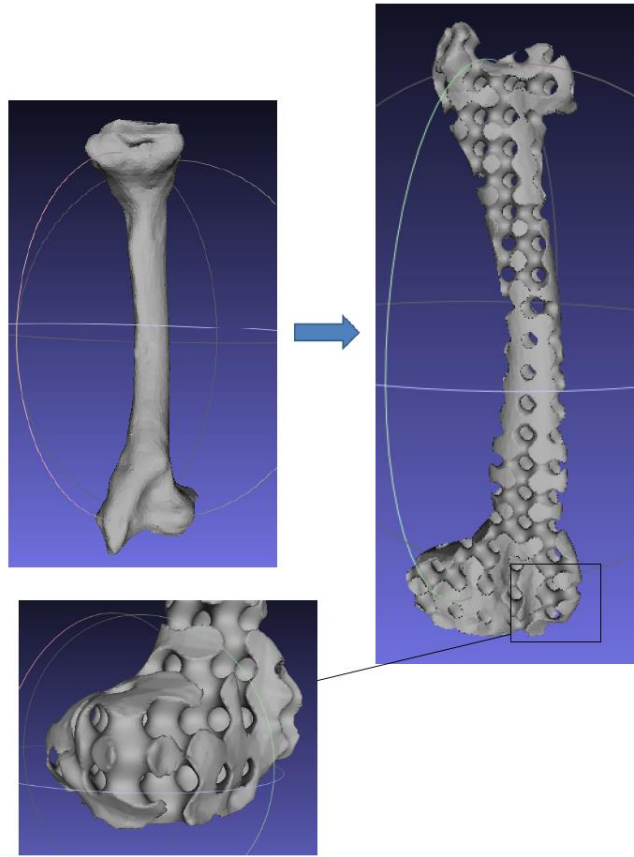


Figure 2.3: Porous femur bone scaffold designed by TPMS. From the close-up view, one can see broken and deformed structures of the surface layer.

# Chapter 3

## **Porous Structure Design Using Parameterized Hexahedral Meshes and Triply Periodic Minimal Surfaces**

In this chapter, a new heterogeneous methodology of modeling scaffolds with internal pore structure using parameterized hexahedral mesh and triply periodic minimal surfaces (TPMSs) is proposed to address the problems mentioned in Chapter 2. The algorithm not only ensures the consistency and continuity of the generated mesh, but also achieves highly homogeneous hexahedral meshes with approximate real pore structure. Moreover, the suggested parameterized approach will solve the problems mentioned in Chapter 2 from the following aspects:

- 1) Generating hexahedral surface meshes for a particular model by calculating 12 geodesic lines via heat conduction functions;
- 2) Mapping between regular hexahedral and transformed hexahedral meshes via tri-linear interpolation, by which the resulting meshes are guaranteed to be continuous on the boundary;
- 3) Solving the problem of deformation and protruding ridges by using parameterization.

The proposed method consist of four primary steps:

- 1) Re-meshing an input model into a hexahedral mesh;
- 2) Solving parametric expressions of 12 geodesic lines using heat conduction models;
- 3) Constructing quadrilateral surface meshes, and hexahedral volume meshes with harmonic-map;
- 4) Generating standard TPMS surfaces within a regular hexahedron using the Marching Cube (MC) algorithm and then mapping the generated surface onto the parameterized hexahedral mesh using tri-linear interpolation.

### 3.1 Related Work

Parameterization, which is intended to build parametric equations for space curves or fluid, and to implement the mathematical expression of geometric modeling, is an ideal tool for representing physical models. This approach has been widely used in numerous domains including geometric analysis [52], texture mapping [53], medical image processing [54-55], etc. Satisfying different requirements of the parameterization for complex structure modeling, has recently motivated researchers to introduce volumetric parameterization methods based on the harmonic mapping theory:

- 1) Construction of parameterized volume meshes within a constraint set of boundary spline surfaces [52];
- 2) Construction of parameterized hexahedral meshes using identical morphology from PolyCube-Map to optimize unit volumes and reduce deformation of parameterized volumetric meshes [55-56];
- 3) Using biharmonic volumetric mapping on complex structure to ensure  $C_1$

- continuity of the surface on boundaries [57];
- 4) Proposing the method of fundamental solutions (MFS) and a strategy for characteristically alignment of diverse manifolds such as points, lines and planes [58];
  - 5) Generalizing conformal maps to volumetric meshes [59];
  - 6) Construction of layered hexahedral meshes for shell objects using distance field and parameterization [60];
  - 7) Creating higher order parametric trivariate representations from closed triangle meshes with higher genus or bifurcations [61].

Some researchers have implemented cross parameterization through standard models:

- 1) Generating patient-specific hexahedral meshes of facial soft tissue models based on a volumetric cross-parameterization mapping from a standard hexahedral mesh to the individual model [62];
- 2) Parameterization of the cortical surface defined via a harmonic mapping of each hemi-sphere surface to a rectangular planar domain that integrates a representation of the model [54];
- 3) Customizing 3D garment models via transferring garment models initially based on a human reference model onto a target model [63].

### 3.1.1 Field Function

With a given piecewise linear manifold  $M = (V, F)$  where  $V$  and  $F$  are sets of vertices and faces of  $M$ , respectively. And each vertex  $i$  is assigned a position  $x_i \in R^3$  in a 3D Euclidean space. If we can construct a scalar field  $u: V \rightarrow R^3$  over the vertices of  $M$ , then the orthogonal

vector field  $G_{\perp}: F \rightarrow R^3$  can be derived in consequence. [65]

Here we are interested in constructing a harmonic function which satisfies the Laplace equation  $\Delta u = 0$  subject to Dirichlet boundary conditions that vertices in the set  $C \subset V$  of constrained vertices take on the prescribed values. That can be described as

$$\begin{cases} \Delta u = 0 \\ \text{subject: } u|_C = \varphi \end{cases} \quad (3.1.1)$$

where  $\varphi$  is the constraint function defined on  $C$ .

On a triangulated manifold, the discrete Laplace-Beltrami operator [140, 141] can be described as

$$\Delta u_i = \sum_{j \in N_i} \frac{\omega_{ij}(u_j - u_i)}{A_i} \quad (3.1.2)$$

where  $N_i$  is the set of adjacent vertices to vertex  $i$  and  $\omega_{ij}$  is a scalar weight assigned to the directed edge  $(i, j)$  such that  $\sum_{j \in N_i} \omega_{ij} = 1$ . More specifically,  $\omega_{ij} = -0.5 \times (\cot(\alpha_{ij}) + \cot(\beta_{ij}))$  is the weight assigned to the directed edge  $e_{ij}$  where  $\alpha_{ij}$  and  $\beta_{ij}$  are two diagonals connected by edge  $e_{ij}$ . Additionally,  $A_i$  indicates  $\frac{1}{3}$  of the total areas of the triangles meeting at vertex  $i$ .

Therefore, by solving the Equation (3.1.2), the values of  $u$  can be obtained for all the vertices on the manifold. Then, the gradient vector for each triangle on the manifold is computed through Equation (3.1.3).

$$\Delta u = \sum_{i=0}^2 \frac{u_i(n \times e_i)}{2A_f} \quad (3.1.3)$$

where  $e_i$  is the  $i^{\text{th}}$  edge of triangle  $f$ ,  $u_i$  indicates the parameter of opposite vertex of edge  $e_i$ , and  $A_f$  represents the area of triangle  $f$ . Finally, the orthogonal vector  $G_{\perp}$  could be derived as  $n \times G$ .



### 3.1.2 Solving Geodesic Lines Using Heat Conduction Model

Heat conduction equations are used to describe the temperature changes over time in a certain region. In Euclidean space, temperature  $u = u(t, x, y, z)$  is a function of time variable  $t$  and space variables  $(x, y, z)$ . Suppose that there is a heat source  $S$  in the region  $M$  ( $S \subset M$ ), constrained by the first boundary condition, the problem of solving field function with  $S$  can be expressed as:

$$\begin{cases} u' = \Delta u \\ \text{subject: } u|_S = \varphi \end{cases} \quad (3.1.4)$$

where  $\varphi = u_0$  is the initial state when  $t = 0$ .

For a given time  $t$ , there is a corresponding function  $u_t$  such that the discrete time domain function can be written as:

$$(E - t\Delta) \cdot u_t = u_0 \quad (3.1.5)$$

where  $E$  is an identity matrix, and  $\Delta$  represents either the discrete Laplace or the Laplace-Beltrami operators.

If a space discrete operator such as the operator from Equation (3.1.2) is applied to a triangulated mesh, then by plugging Equation (3.1.2) into Equation (3.1.5) we will have:

$$E - t \sum_{j \in N_i} \frac{\omega_{ij}(u_j - u_i)}{A_i} = A_i - t \sum_{j \in N_i} \omega_{ij}(u_j - u_i) = \begin{cases} 0, & i \in M \setminus S \\ 1, & i \in S \end{cases} \quad (3.1.6)$$

where the value of equation equals 0 when  $i$  is not on the boundary  $S$ ; Otherwise, the equation equals 1. When  $t$  is given, the corresponding node parameter  $u$  will be solved by Equation (3.1.4).

### 3.2 Method Description

Based on the idea of combining parameterized hexahedral mesh and TPMS, the proposed method firstly takes a set of triangulated meshes as input to construct initial hexahedron. Secondly, it parameterizes surface and volumetric meshes, and eventually projects the regular TPMS porous surface onto the parameterized hexahedral mesh. The whole process is defined in Figure 3.1.

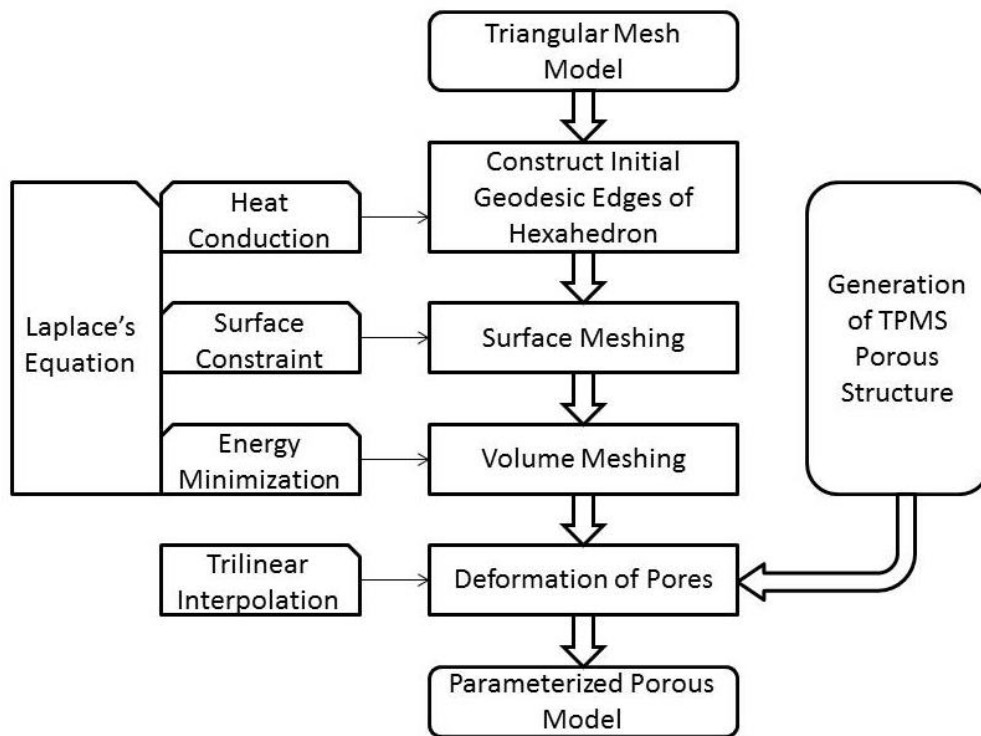


Figure 3.1: The framework of the proposed approach.

Four essential procedures of the suggested approach in addition to their specific operations are illustrated in Figure 3.1 as follows:

- 1) Construction of initial hexahedron:

Eight vertices of the input triangulated meshes are selected to construct 12 Geodesic lines. The 12 edges are solved using single-source heat conduction model.

2) Surface meshing:

Assume that different pairs of opposite faces are chosen as poles of the double-pore magnetic field system, then the iso-parametric lines are constructed. The parameterized surface can be obtained by solving the Laplace's equation of the vertices on the constraint surfaces.

3) Volume meshing:

Constrained by the parameterized surface meshes from Step 2), the inner vertices of the volume can be solved via the method of energy minimization.

4) Reconstruction of hexahedral porous surface:

Regular TPMS surface is generated based on the well-known MC algorithm. Then, the generated surface is projected onto the parameterized volumetric mesh using the method of tri-linear interpolation.

### **3.2.1 Construction of Hexahedral Edges**

It is known that the shortest distance between any two points on a triangular mesh, is the straight path through the two points along the surface, however, the efficiency of finding the shortest path is restricted by the complexity of surface shape. The time complexity of finding the shortest path on surface of a complicated model can be as large as  $O(n^n)$ . This method aims to

solve this problem by a practical mathematical method such as the single-source heat conduction model.

Given a heat source in a thermal field, the geodesic lines between any point in the field and the heat source could be measured [64]. In a cube, each vertex is connected with its three neighboring vertices and each pair of vertices constitutes an edge of the cube. Therefore, we are able to construct three edges for each heat source point. The twelve geodesic curves will be drawn out by choosing four different heat sources, and the corresponding thermal field could be constructed as well. Table 3.1 lists relationships between the heat sources and the targets which are labeled in Figure 3.2.

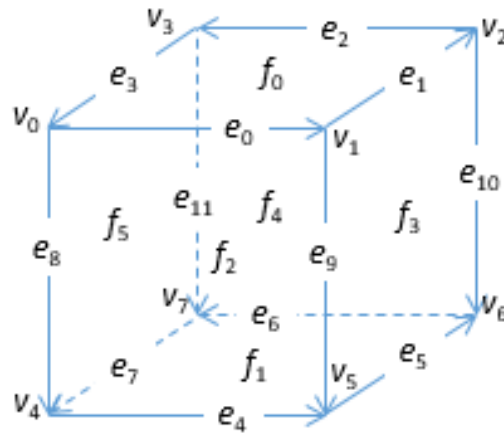


Figure 3.2: The cube of labeled vertices and edges.

Src.	Dest.	Reversed	Edge
0	1	Y	0
	3	N	3
	4	Y	8

2	1	N	1
	3	Y	2
	4	Y	10
5	1	N	9
	4	N	4
	6	Y	5
7	3	N	11
	4	Y	7
	6	N	6

Table 3.1: The relationships between heat sources and their neighboring vertices along with directions of geodesic lines.

### 3.2.2 Construction of Hexahedral Faces

A set of numbering rules which is demonstrated in Table 3.2 specifies how four surrounding edges construct a hexahedral face.

Face No.	Edge No. (R represents an edge with reversed direction)
0	0-1-2-3
1	4-5-6-7
2	0-9-4(R)-8(R)

3	1-10-5(R)-9(R)
4	2-11-6(R)-10(R)
5	3-8-7(R)-11(R)

Table 3.2: Numbering rules for hexahedral faces.

To collect the information regarding a plane, we can choose a seed point on the plane and increase searching area by adding nearby points to the seed point. Two alternative searching algorithms which could be used in this algorithm are listed the two following subsections.

### 3.2.2.1 In-plane Based Seed Searching Algorithm

To search the local area near a vertex, all in-plane points are restricted to the area constrained by the three edges that are connected to the vertex. For example, in Figure 3.3 (a)  $v_0$  is a vertex on plane  $f_0$ , and all points inside  $f_0$  are residing in the area constrained by edges  $e_0$  and  $e_3$ . As a result, all in-plane points from neighbor triangles could be found by using the approach of in-plane based seed searching.

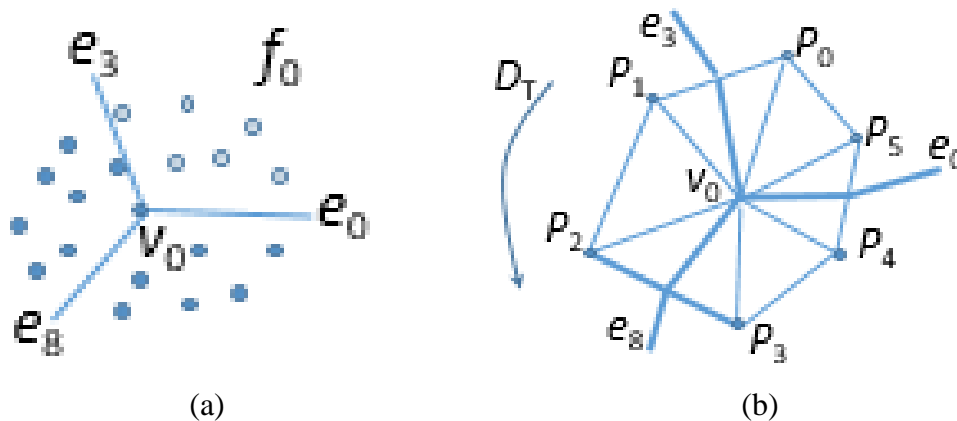


Figure 3.3: The in-plane based seed searching algorithm.

In Figure 3.3 (b), as an example of searching in-plane points of plane  $f_0(v_0, v_1, v_2, v_3)$  at vertex  $v_0$ , suppose  $e_i \cdot v_j$  and  $e_i \cdot t_j$  represent the  $i^{\text{th}}$  vertex forming edge  $e^i$  and the triangle where the point locates on, respectively. The searching algorithm is explained as follows:

- 1) Search  $v_0$ 's first-order neighbor vertices  $P_i$  and its first-order neighbor triangles  $T_i$  where  $i = 0, 1, \dots, m$  where  $m$  is the number of neighboring triangles. Also, sort  $P_i$  and  $T_i$  along direction  $D_T$ ;
- 2) If  $e_3 \cdot t_0$  is the same as  $T_i$ , then continue to Step 3). Otherwise, move back to Step 1) to search the next vertex;
- 3) Search seed point among the first-order neighboring points along direction  $D_T$  starting from point  $P_i$ . Firstly, connect points  $e_0 \cdot v_0$  and  $e_0 \cdot v_1$ , also, connect points  $e_3 \cdot v_0$  and  $e_3 \cdot v_1$ . Secondly, find the intersection point between line segment  $P_i P_{i+1}$  and the two segments above, separately. If only one intersection point is found from the three line segments, then the intersection point must lie on segment  $P_i P_{i+1}$ , which means it is located between points  $P_i$  and  $P_{i+1}$ . Therefore, report  $P_{i+1}$  as the seed point; If not, move back to Step 1) to search the next vertex.

As it is shown in Figure 3.3 (b), we could find  $\Delta v_0 P_2 P_3$  in Step 2, and next start from  $P_3$  to search  $P_4, P_5, \dots$  counterclockwise. Because segment  $P_4 P_5$  is intersected with edge  $e_0$  but not edge  $e_3$ , the intersection point must lie on segment  $P_4 P_5$  and  $P_5$  is an in-plane point of the area which is constrained by edges  $e_0$  and  $e_3$ .

### 3.2.2.2 In-plane Growing Searching Algorithm

If the boundary of a surface and a point on or inside the surface are given simultaneously, the other points on that surface could be traversed through the growing algorithm. Suppose  $e$  is the boundary of the surface, and the seed and current points are shown by  $ptSeed$  and  $ptCur$ , respectively. Also, suppose the set  $C$  indicates an empty set used to accommodate all points to be visited, then the growing algorithm can be demonstrated using following steps:

- 1) Push  $ptSeed$  into the stack and coerce it into the set  $C$  at the same time;
- 2) Pop out the top point from the stack and refer  $ptCur$  to it. Next, search the adjoining points to  $ptCur$ , and for every adjacent point  $P_i$  to  $ptCur$  check if it resides inside the surface. Then, connect  $ptCur$  and  $P_i$ , and find the intersection point for each edge of  $e$ . If no intersection point is found, then push  $P_i$  into stack and include it into the set  $C$  simultaneously;
- 3) Move back to Step 2) and continue until the stack is empty.

### 3.2.3 Surface Meshing

To reconstruct the iso-parametric quadrilaterals on the surface, iso-parametric curves between each pair of opposite faces must be solved at first. Constrained by the pair of opposite planes, the projection would be implemented by using Laplace's equation to solve the gradient field  $G$  and its orthogonal gradient field  $G^\perp$ . We can assume that the iso-parametric lines between left and right, front and back, and up and down planes lie on  $U$ -axis,  $V$ -axis and  $W$ -axis, respectively; and the density of the mesh is  $n_u \times n_v \times n_w$ . As it is illustrated in Figure 3.2, suppose the up face  $f_0$  indicates a set of minimal points ( $w = 0$ ) and the down face  $f_1$  indicates a set of



maximal points ( $w = 1$ ). Solving function  $\Delta w = 0$  can provide  $G$  and  $G\perp$ , and the iso-parametric lines along  $W$ -axis would be drawn by following the gradient lines along the direction of  $G\perp$  [65]. Similarly, the iso-parametric lines along  $U$  and  $V$  axes could be acquired by solving  $\Delta u = 0$  and  $\Delta v = 0$ , separately.

The intersection points of lines along  $U$ ,  $V$ ,  $W$  axes are the vertices of quadrilateral meshes, however, the superabundant lines along  $U$ ,  $V$ ,  $W$  would lead to a low efficiency in calculation. To avoid this problem, the proposed approach has adopted a simplified method of evenly offset interception to get homogeneous mesh nodes. Using the iso-parametric lines along  $W$ -axis as an example:

- 1) Divide an arbitrary iso-parametric line  $W_i$  into four segments, which are labeled with  $W_1^i$ ,  $W_2^i$ ,  $W_3^i$  and  $W_4^i$  accordingly, by intersecting with the four edges  $e_8$ ,  $e_9$ ,  $e_{10}$  and  $e_{11}$ , separately;
- 2) Divide  $W_1^i$  and  $W_3^i$  evenly into  $n_u$  slices and divide  $W_2^i$  and  $W_4^i$  evenly into  $n_v$  slices, simultaneously. The equalization points would be the mesh nodes as demanded.

It is necessary to say that the output mesh has to be consistent with the directions of  $U$ ,  $V$  and  $W$  axes.

### 3.2.4 Volume Meshing

To generate the parameterized mesh with minimal energy, Equation (3.2.1) is used to construct Laplace's equation [66] as below:

$$E = \sum_{i \in V} (|\Delta u_i - u_i| + |\Delta v_i - v_i| + |\Delta w_i - w_i|) \quad (3.2.1)$$

where  $V$  is the set of points of the volumetric mesh. Then Equation (3.2.1) can be resolved into:

$$E(i, j, k) = \sum_{h \in N_{ijk}} \omega_h \left| |v_{ijk} - v_{ijk}^h| \right| \quad (3.2.2)$$

where  $v_{ijk}^h$  is a neighbor point of  $v_{ijk}$ , and  $N_{ijk}$  is the neighboring area of  $h$ , and  $\omega_h$  is the weight of point  $v_{ijk}^h$ . By setting  $\omega_h$  to  $1/6$  as a constant in Equation (3.2.2), the equation can be further decomposed into [67]:

$$v_{i,j,k} = \frac{(v_{i-1,j,k} + v_{i+1,j,k} + v_{i,j-1,k} + v_{i,j+1,k} + v_{i,j,k-1} + v_{i,j,k+1})}{6} \quad (3.2.3)$$

Constrained by the point set of quadrilateral meshes on the surface, all points  $v_{i,j,k} \in V \rightarrow R^3$  can be solved through Equation (3.2.3).

### 3.2.5 Reconstruction of Hexahedral Parameterized TPMS Surface

Simple patterns can be projected onto surfaces of volumes using the parametric mapping technique. Similarly, a regular quadrilateral or a hexahedral mesh can be transformed into a distorted quadrilateral or a hexahedron in the same way. As it is shown in Figure 3.4, mapping relationship between two meshes can be described as:

$$f(x, y) \xrightarrow{L} g(u, v) \quad (3.2.4)$$

where  $f(x, y)$  and  $g(u, v)$  are functions for regular and distorted meshes, respectively, and  $L$  is the topological mapping function in term of discrete Laplace's equation.

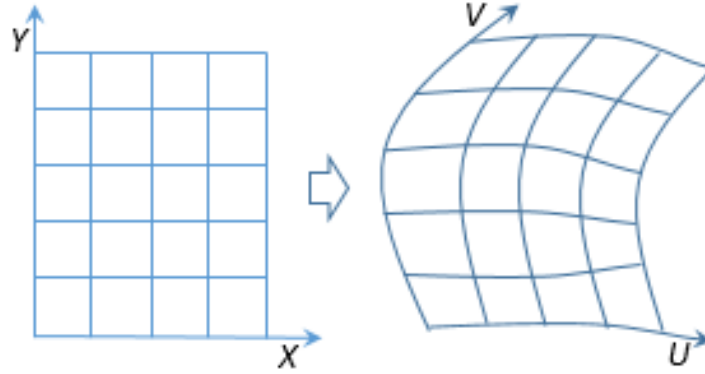


Figure 3.4: Parametric mapping.

### 3.2.5.1 Generation of Standard Periodic TPMSs

Based on the TPMS algorithm described in Chapter 2, we could take any 3D model as input and partition its corresponding bounding box into a matrix of small cubes. The resolution for the matrix is defined as  $n_u \times n_v \times n_w$ . And the surface type can be chosen from P, D, G, and I-WP. Accordingly, for each of the small cubes, a TPMS iso-surface is generated using the marching cube algorithm. And then all of these small surface patches consist a pore network which is used as the base model to construct hexahedral porous surface. More details of the TPMS surface generation could refer to Section 2.3.

### 3.2.5.2 Pores' Mapping Relationship Between Regular TPMS and Parameterized Meshes

Parameterized hexahedral meshes are distorted meshes. Although, not all of space quadrilaterals can be guaranteed to be a planar quadrilateral, we do know that the two adjacent hexahedrons share a common quadrilateral so that the surface on the two hexahedrons should be seamless and continuous. To construct a distorted hexahedron which consists of a certain number

of nonplanar space quadrilaterals, the method of tri-linear interpolation [68] may be employed to map vertices from the regular to the distorted hexahedron. Since both tri-linear interpolation function and the point-by-point mapping function are continuous, the transformed pores' structure can be guaranteed to be continuous.

In mathematical language, this problem can be translated into that for an arbitrary point  $v_{ijk}^0$  on the regular hexahedron  $V^0$ , suppose the transformed hexahedron is  $V^l$  and the mapping function  $\Phi$  is a function of trilinear interpolation, we need to find the projected point  $v_{ijk}^1$  of  $v_{ijk}^0$  on  $V^l$ . Combining with Figure 3.2 and Figure 3.5, we can find the solution through following steps:

- 1) Apply bilinear interpolation to  $f_0$  on the  $U$ -axis,

$$_v_0 = v_0 + (v_1 - v_0) \times v_{ijk}^0 \cdot x \quad (3.2.5)$$

$$_v_1 = v_3 + (v_2 - v_3) \times v_{ijk}^0 \cdot x \quad (3.2.6)$$

Then interpolate points  $_v_0$  and  $_v_1$  along  $V$ -axis,

$$_v_2 = _v_0 + (_v_1 - _v_0) \times v_{ijk}^0 \cdot y \quad (3.2.7)$$

- 2) Similarly, apply the bilinear interpolation to  $f_l$  to obtain points  $_v_3$ ,  $_v_4$  and  $_v_5$ .

- 3) Interpolate points  $_v_2$  and  $_v_5$  along  $W$ -axis,

$$_v_6 = _v_2 + (_v_5 - _v_2) \times v_{ijk}^0 \cdot z \quad (3.2.8)$$

eventually point  $v_6$  is the desired point  $v_{ijk}^1$ .

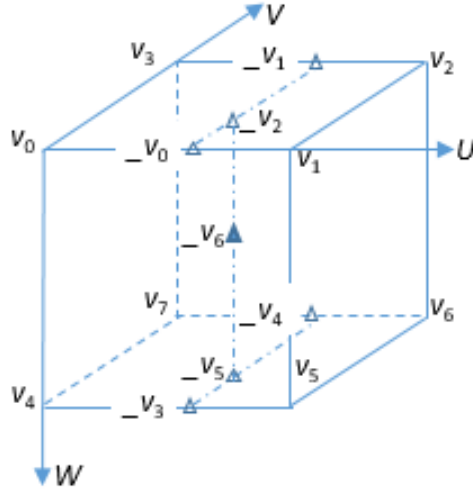


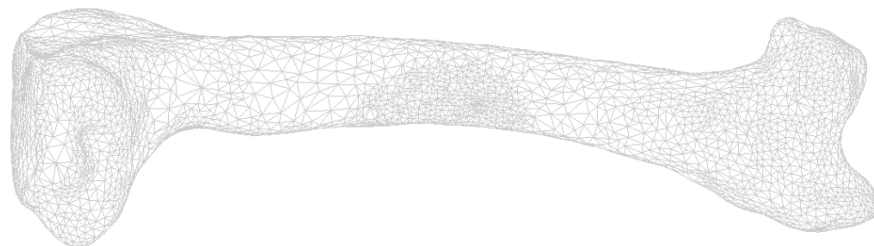
Figure 3.5: Tri-linear interpolation.

### 3.3 Experimental Results

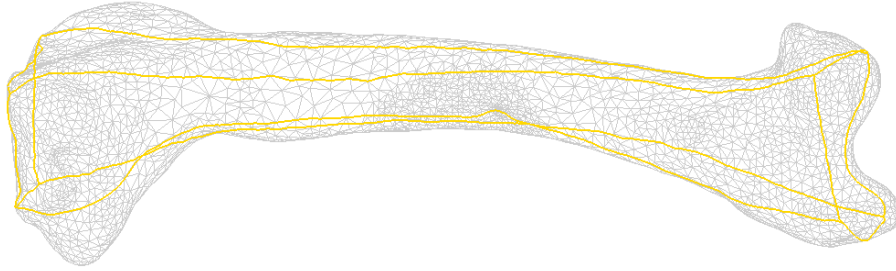
All experiments are processed on an Intel Core i3 3.4 GHz desktop with 8GB of memory and Windows 10 OS. And the proposed method was implemented on a VS2008 + OpenGL platform, and the displayed models were enhanced using MeshLab. The 3D femur model (the same one as used in Chapter 2) is reused to demonstrate some experimental results. TPMS P surface is selected as the base type to construct porous surface. And two different resolutions ( $5 \times 7 \times 20$  and  $5 \times 10 \times 60$ ) are chosen to render the outputs. Figure 3.6 demonstrates the hexahedral boundaries and the distribution of extracted in-plane points. Figure 3.7 shows the process of generating parameterized hexahedral mesh in steps (a) to (g). Figure 3.8 gives the generate porous TPMS femur scaffolds using hexahedral parameterized surface. Moreover, Figures 3.9 and 3.10 display different stages of simulating porous molecule model and dog bone with inner porous structure, respectively. Table 3.3 summarizes the statistics of running time for different experiments.

Surface Type	Resolution			
	$5 \times 7 \times 20$		$5 \times 10 \times 60$	
	MC / IO	$n_V / n_T$	MC / IO	$n_V / n_T$
P Surface	0.25s /	1047800 /	1.20s /	4492400 /
	3.00s	2077000	13.10s	8925200
D Surface	0.39s /	1674500 /	1.70s /	7181000 /
	4.90s	3327720	21.55s	14306750
G Surface	0.38s /	1321700 /	1.61s /	5669000 /
	3.83s	2621984	16.76s	11272900
I-WP Surface	0.40s /	1383800 /	1.72s /	5932400 /
	4.05s	2751528	17.56s	11823100
Geodesic Line ( $\times 12$ )	0.285s			
U, V, W Iso-parametric Lines	0.03s $\times$ 0.04s $\times$ 0.06s			
Volumetric Mesh	0.03s		0.11s	

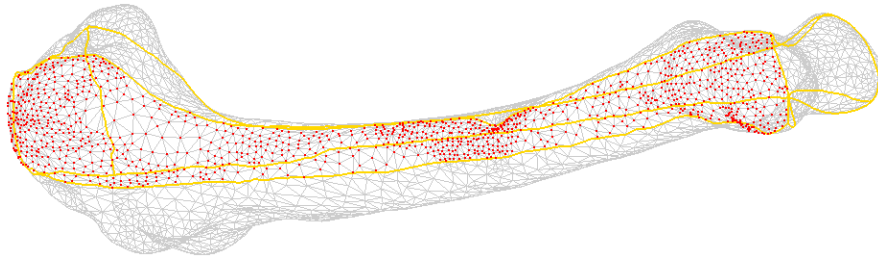
Table 3.3: Run-time for each experimental node.



(a) Original mesh

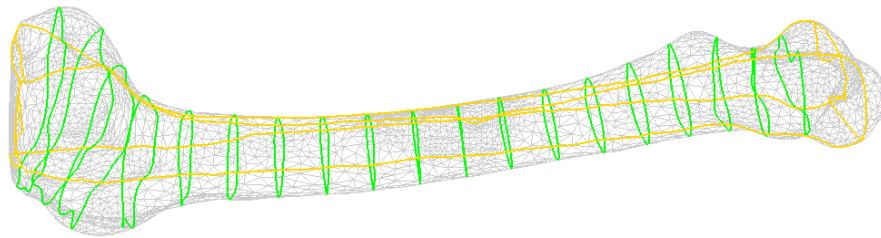


(b) 12 Geodesic lines

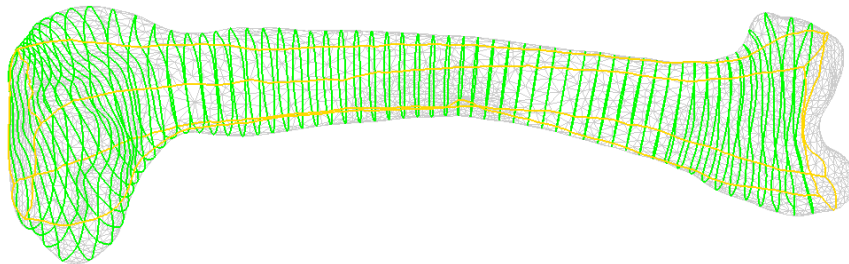


(c) Extraction of in-plane points

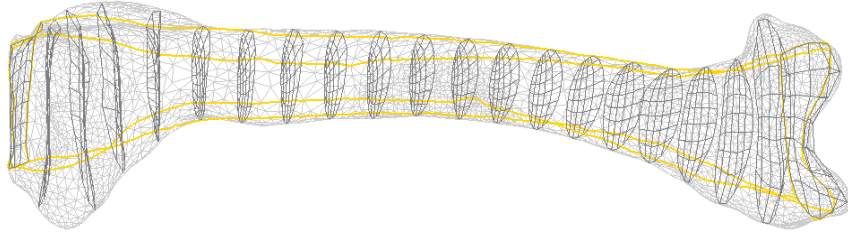
Figure 3.6: Results of extractions of Geodesic lines and in-plane points.



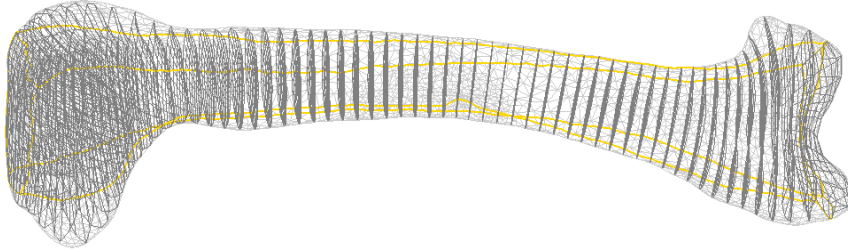
(a) W-axis-oriented iso-parametric lines ( $5 \times 7 \times 20$ )



(b) W-axis-oriented iso-parametric lines ( $5 \times 10 \times 60$ )



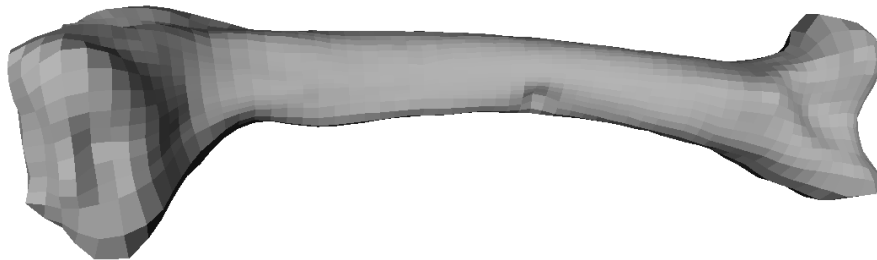
(c) Cross planes on the W-axis ( $5 \times 7 \times 20$ )



(d) Cross planes on the W-axis ( $5 \times 10 \times 60$ )

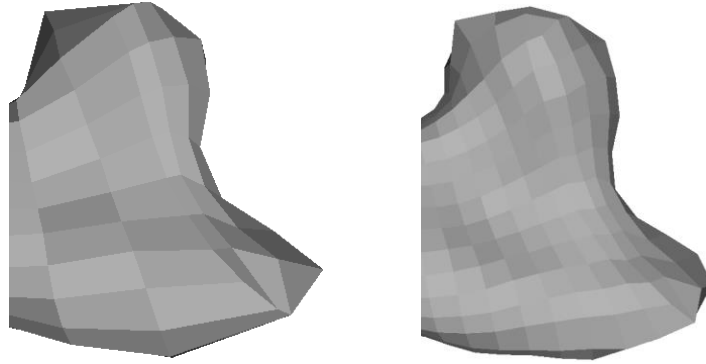


(e) Hexahedral surface mesh ( $5 \times 7 \times 20$ )



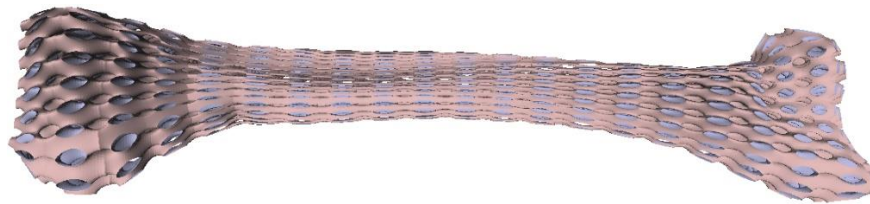
(f) Hexahedral surface mesh ( $5 \times 10 \times 60$ )



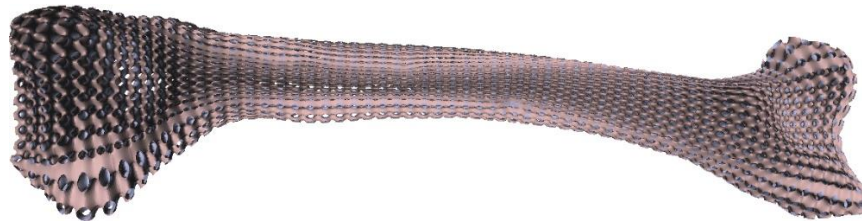


(g) Larger views of (e) and (h) respectively

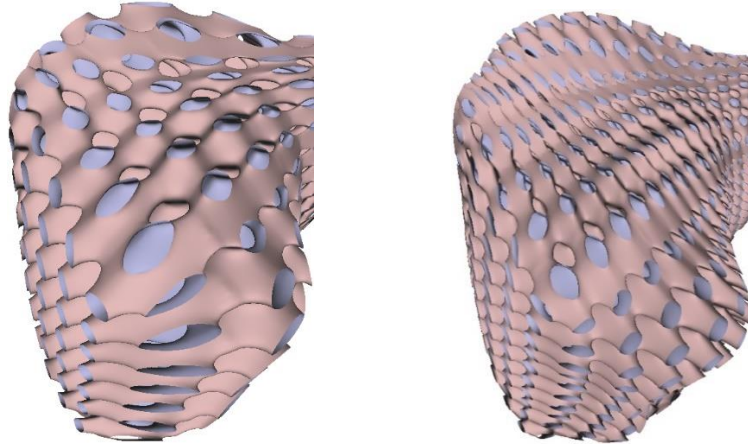
Figure 3.7: Process of parameterization of femur bone.



(a) Parameterized TPMS P surface mesh ( $5 \times 7 \times 20$ )

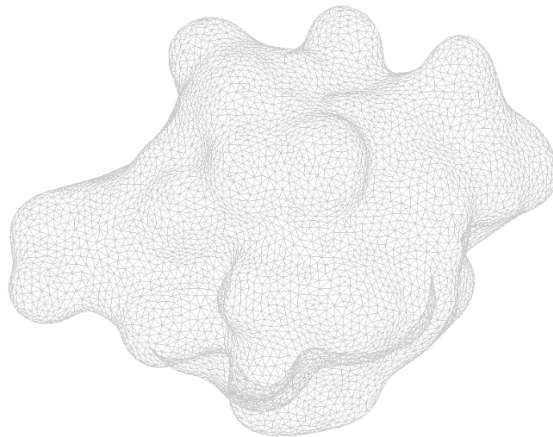


(b) Parameterized TPMS P surface mesh ( $5 \times 10 \times 60$ )

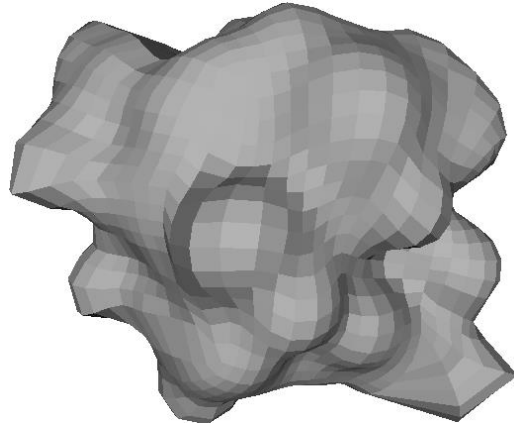


(c) Larger views of (a) and (b) respectively

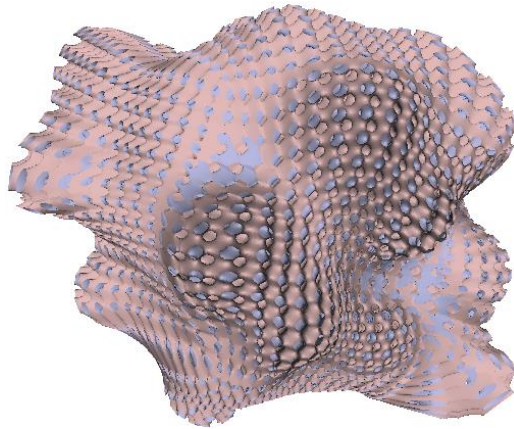
Figure 3.8: Parameterized hexahedral surface imbedded with TPMS P porous inner structure.



(a) Original mesh

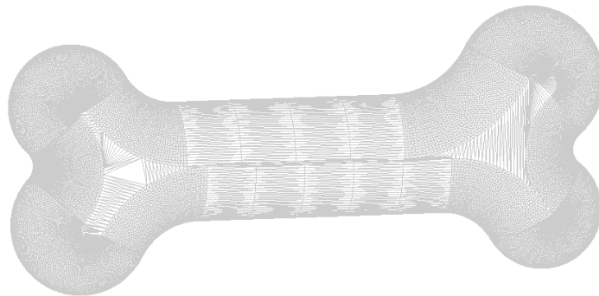


(b) 12 Geodesic lines

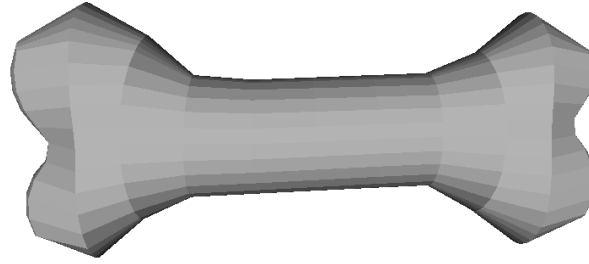


(c) Extraction of in-plane points

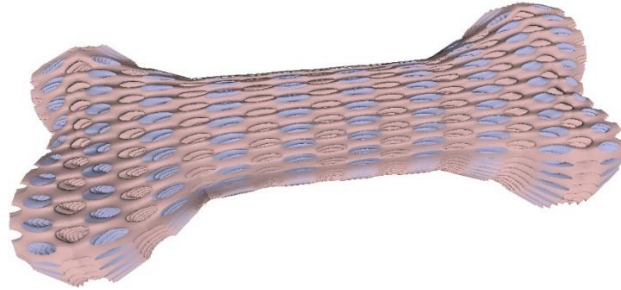
Figure 3.9: Effects of generation of parameterized molecule model.



(a) Original mesh



(b) 12 Geodesic lines



(c) Extraction of in-plane points

Figure 3.10: Effects of generation of parameterized dog bone.

From above-mentioned experiments, the 12 Geodesic lines are obtained by solving the point heat source function for four times; the iso-parametric lines along  $U$ ,  $V$  and  $W$  axes are obtained via executing three times of local facial heat source function; and volumetric mesh is obtained by solving all points from the hexahedron once. Among which, the time complexity for solving Geodesic lines and iso-parametric lines are related to the number of input triangular mesh and the time complexity for solving the volumetric mesh is based on the resolution of hexahedral mesh ( $n_u \times n_v \times n_w$ ). On the other hand, the memory consumption is related to the grid's scale and the major consumption belongs to the following three aspects:

- 1) Number of triangles in the pore network

The memory of storing points and triangles of the porous network in scale of  $n_V$  /  $n_T$  are  $n_V \times 3 \times 8$  bytes and  $n_T \times 3 \times 4$  respectively. As it is shown in Table 3.3, the generated D Surface with  $5 \times 10 \times 60$  resolution would consume 344 MB of memory ( $7181000 \times 3 \times 8 + 14306750 \times 3 \times 4$  bytes).

## 2) Size of volumetric computing matrix

For a volumetric mesh with  $n_u \times n_v \times n_w$  scale, the size of Laplace matrix would be  $n_u \cdot n_v \cdot n_w \times n_u \cdot n_v \cdot n_w$ . However, if we express the matrix in the format of sparse matrices and suppose there are six contiguous triangles connected to each triangle, then the size will decrease to  $n_u \cdot n_v \cdot n_w \times 6$ . Apparently, the larger matrices consume more memory space. For a hexahedral mesh with  $5 \times 7 \times 20$  resolution, the required memory to store the density matrix is 3.92 MB while the memory needed for the sparse matrix is only 9.6 KB.

## 3) Data capacity of volumetric mesh

For a volumetric mesh with  $n_u \times n_v \times n_w$  resolution, the storage needed for all the points in the mesh is  $(n_u + 1) \times (n_v + 1) \times (n_w + 1)$  bytes. Consequently, the desired memory for the meshes with  $5 \times 7 \times 20$  and  $5 \times 10 \times 60$  resolutions are 24.192 KB and 96.624 KB, respectively.

## 3.4 Summary

In the present chapter, we proposed a heterogeneous methodology of modeling scaffolds with internal pore structure using parameterized hexahedral mesh and TPMS. It solved the integrity and continuity problem of regular TPMS surface in two steps: 1) Firstly, generating

hexahedral surface meshes for a particular model by calculating 12 geodesic lines via heat conduction functions; 2) Secondly, mapping between regular hexahedral and transformed hexahedral meshes via tri-linear interpolation, by which the resulting meshes are guaranteed to be continuous on the boundary. The algorithm not only ensures the consistency and continuity of the generated mesh, but also achieves highly homogeneous hexahedral meshes with approximate real pore structure. Moreover, it's relatively simple for implementation. Even if this method has so many advantages, nonetheless, it still has a deficiency that can't be ignored, that is, it can only be used for hexahedral modeling.

# Chapter 4

## **Porous Structure Design Using Tetrahedral Implicit Surfaces**

In Chapter 3, the parameterized method deformed the triangular mesh into distorted hexahedral mesh and then mapped the regular TPMS surface onto the parameterized mesh. Although it is a viable solution to solve the discontinuity and integrality problems for some complex shapes. Nonetheless, it remains the limitation of only applying to hexahedral models. That TPMS must be constructed based on hexahedral meshes doesn't mean hexahedron is the only and optimal basic unit of modeling porous architectures. Actually, it would save us a lot of time and unnecessary work if we can skip the hexahedral mesh generation and model it directly on the tetrahedron. Since most of 3D modeling are using triangular mesh and the fact that it's much simpler to re-mesh a triangular mesh into a tetrahedral mesh rather than a hexahedral mesh, it would be more efficient to design porous scaffold based on tetrahedrons. Meanwhile, a variety of open-source tetrahedralization tools, such as TetGen [69], are easy to use and free to download from the internet.

In this chapter, we introduce a new approach using tetrahedral implicit surface (TIS) to construct porous structures. The framework of our proposed porous scaffold design method is shown in Figure 4.1. Firstly, we reconstruct the 3D model (spatial triangular mesh) of the damaged tissue from the scanned biomedical images and the model is partitioned into a number of small tetrahedrons, followed by a mesh smoothing process to improve the equality of the generated

tetrahedral mesh. Then, a specific surface type is selected from the TIS library to fabricate pore structure for each tetrahedron individually. In the meantime, numerous small patches of TIS surfaces consist of an inner-connected pore network. Lastly, the final product is produced by the intersection of the original tissue surface and the smoothed inner-connected porous mesh.

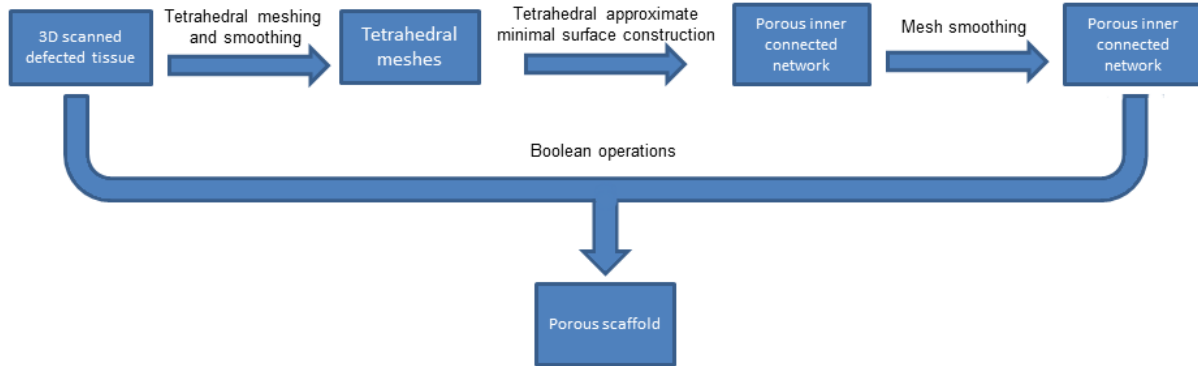


Figure 4.1: The framework of the proposed TIS method.

## 4.1 Tetrahedral Meshing and Smoothing

Tetrahedral mesh generation is a very critical step in this algorithm because the quality of the tetrahedral meshing directly affects the surface quality of the generated biomedical scaffold. Some efforts have been contributed by researchers to figure out the most efficient tetrahedral mesh generation algorithms. Overall, the related work can be classified by Delaunay-based, advancing-front-based, and octree-based methods [70]. The Delaunay-based methods usually distribute a set of vertices in the domain and then triangulated by Delaunay triangulation [71]. Miller et al. presented a sphere-packing method for Delaunay-based mesh generation [72]. Chew proposed an algorithm that produces constant density meshes by adding points in a randomized manner to eliminate slivers [73]. Alliez et al. attempted to generate isotropic tetrahedral meshes mixed with



a constrained relaxation on the boundary [74]. Advancing front-based methods insert new Steiner points inside a given domain to generate tetrahedrons with acceptable shapes and sizes [75]. The disadvantage of advancing front-based techniques lies at front-merging which used to occur near the high curvature regions of the boundary [76]. Octree-based approaches subdivide the domain enclosing the given mesh recursively until certain stopping criterion is reached. Fuchs proposed a meshing method based on the body-centered cubic lattice and obtained high quality tetrahedrons [77]. Ito et al. created an octree-based mesh generator to fabricate volumetric mesh from triangulated surface without sharp features [78]. An example of octree topology is given in Figure 4.2.

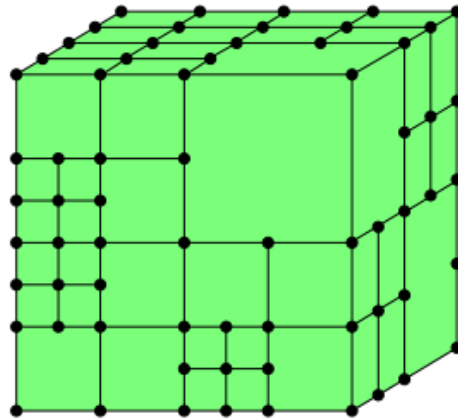


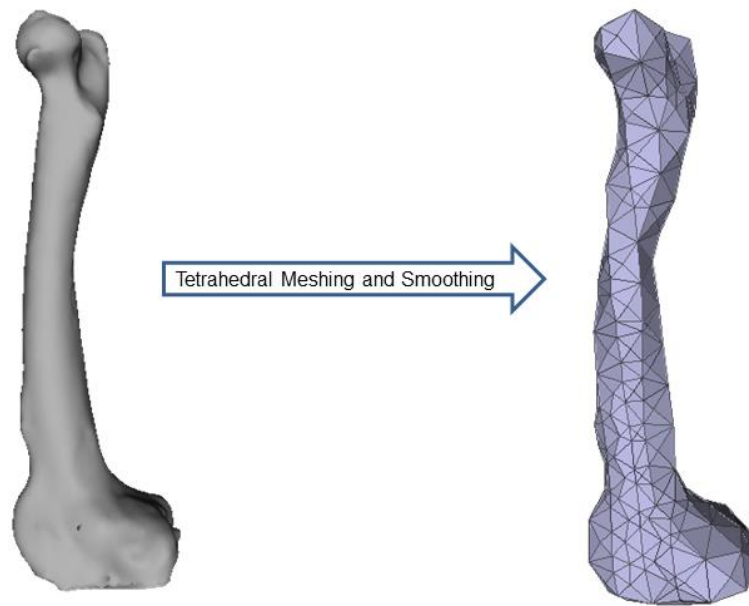
Figure 4.2: An octree topology.

Too small and too large dihedral angles both will lead to sharp edges and surface deformation. To reduce the computational time, the number of tetrahedral elements should be as small as possible while important geometric features should be faithfully retained. While taking these factors into account, we adopted a tetrahedral mesh generation algorithm based on the body centered cubic (BCC) tetrahedral lattice [76], followed by a quality tetrahedral mesh smoothing

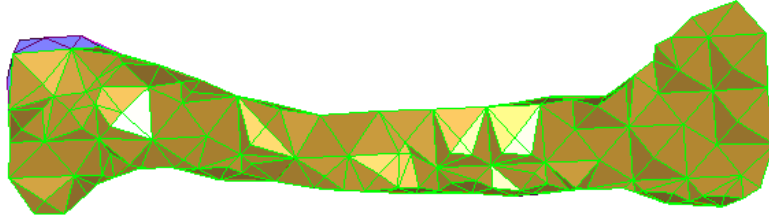
via boundary-optimized Delaunay triangulation (B-ODT) [79-80]. The algorithm is illustrated in following steps:

- 1) Construct and subdivide the octree of the input 3D mesh based on Euclidean distance transformation;
- 2) Compute the sign of each node in the BCC lattice, and then calculate and snap the intersecting points where the edge crosses the input surface mesh;
- 3) Decompose the boundary polyhedra into tetrahedra;
- 4) The tetrahedral meshes are smoothed using the B-ODT approach.

Fig. 4.3 (a) shows the tetrahedral mesh of the 3D femur model obtained through the tetrahedral meshing and smoothing. Fig. 4.3 (b) shows the cross-section of the tetrahedralized femur model.



a) Tetrahedral meshing on the femur model



b) Cross-section of the tetrahedralized femur model

Figure 4.3: Tetrahedral meshing on 3D femur model.

## 4.2 Method Description

### 4.2.1 Mathematical Description of Tetrahedral Implicit Surface

In mathematics, a minimal surface is a surface with the mean curvature of zero that locally minimizes its area overall. The well-known Weierstrass's method describes a global parametrization of minimal surface by [81]

$$x = \operatorname{Re} \int_{\omega} (1 - \omega'^2) R(\omega') d\omega'$$

$$y = \operatorname{Re} \int_{\omega} i(1 - \omega'^2) R(\omega') d\omega' \quad (4.2.1)$$

$$z = \operatorname{Re} \int_{\omega} 2R(\omega') d\omega'$$

By evaluating the integrals of the real part ( $\operatorname{Re}$ ) of the Weierstrass's function, the Equation (4.2.1) can be simplified as:

$$\Phi(\mathbf{r}) = \sum_{k=1}^K A_k \cos[2\pi k \cdot \mathbf{r} - \alpha(k)] = C \quad (4.2.2)$$

where  $k$  are the reciprocal lattice vectors,  $\alpha(k)$  is a phase shift, and  $A_k$  is an amplitude to a given  $k$ -vector. The equations of TPMS (see Table 2.1) are actually derived from Equation (4.2.2). For more details of TPMSs, you can refer to Section 2.3.

TPMS is characterized as a minimal surface which is periodic in three perpendicular directions. Obviously, the same manner does not apply to the generation of periodic surfaces in a tetrahedron. Our goal is to make the surface periodic in certain particular directions, such as the vertical direction of each of the four faces of a tetrahedron. Here, we introduce a new coordinates system using point-to-plane distances. In this system, there are four axes, corresponding to the vertical directions of four faces of the tetrahedron. The coordinate of an arbitrary point inside the tetrahedron on a certain axis is determined by the ratio of the distance from the point to the corresponding face divided by the corresponding height of the tetrahedron. As shown in Figure 4.4, the coordinates of point  $P$ ,  $(a, b, g, s)$ , can be calculated by:

$$\begin{aligned}
 P_\alpha &= \frac{d_\alpha}{h_\alpha} \times 2\pi \times \lambda_\alpha \\
 P_\beta &= \frac{d_\beta}{h_\beta} \times 2\pi \times \lambda_\beta \\
 P_\gamma &= \frac{d_\gamma}{h_\gamma} \times 2\pi \times \lambda_\gamma \\
 P_\delta &= \frac{d_\delta}{h_\delta} \times 2\pi \times \lambda_\delta
 \end{aligned} \tag{4.2.3}$$

in which  $d_\alpha, d_\beta, d_\gamma,$  and  $d_\delta$  denote the projection distances from point  $P$  to plane  $BCD, ACD, ABD,$  and  $ABC,$  respectively, and  $h_\alpha, h_\beta, h_\gamma, h_\delta$  are the height of the tetrahedron corresponding to each plane. The parameter  $\lambda_i (i = \alpha, \beta, \gamma, \delta)$  determines the periodicity of its corresponding direction.

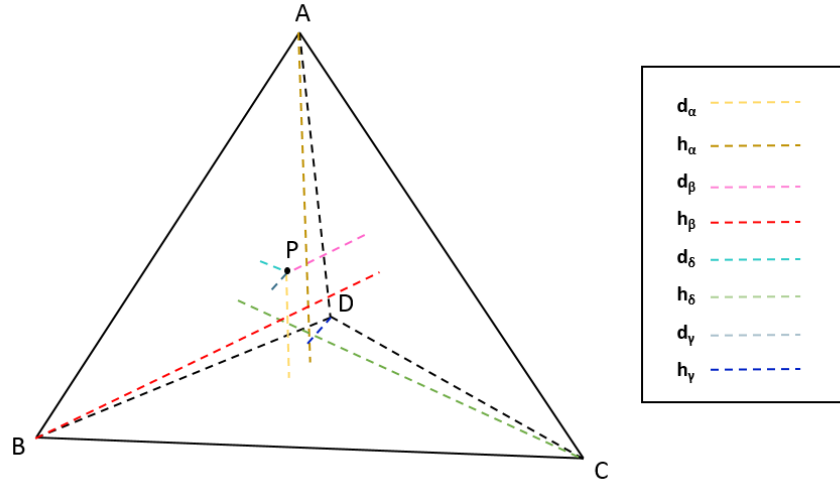


Figure 4.4: The new coordinates system using scaled point-to-plane distances.

To mimic similar properties of TPMS, we adopt the mathematical expressions of the TPMSs on each periodic component and derive the following TIS equations:

$$\Phi_P(r) = \cos(\alpha) + \cos(\beta) + \cos(\gamma) + \cos(\delta) + \cos(\alpha) \cos(\beta) \cos(\gamma) \cos(\delta) = C$$

$$\begin{aligned} \Phi_D(r) = & \sin(\alpha) \cos(\beta) \cos(\gamma) \cos(\delta) + \cos(\alpha) \sin(\beta) \cos(\gamma) \cos(\delta) \\ & + \cos(\alpha) \cos(\beta) \sin(\gamma) \cos(\delta) + \cos(\alpha) \cos(\beta) \cos(\gamma) \sin(\delta) \\ & + \cos(\alpha) \cos(\beta) \cos(\gamma) \cos(\delta) = C \end{aligned}$$

$$\begin{aligned} \Phi_G(r) = & \cos(\alpha) \sin(\beta) + \cos(\beta) \sin(\gamma) + \cos(\gamma) \sin(\delta) + \cos(\delta) \sin(\alpha) + \\ & \cos(\alpha) \cos(\beta) \cos(\gamma) \cos(\delta) = C \end{aligned} \quad (4.2.4)$$

It is worth noting that these TIS surfaces are expressed by implicit functions with a constant threshold  $C$ . With a proper threshold given, the TIS surface mesh can be generated as an iso-surface by using the Marching Cubes algorithm, which is one of the most famous approaches to extracting iso-surfaces from a three-dimension data field.

## 4.2.2 Continuity Analysis of Tetrahedral Implicit Surface

As we discussed in Section 2.3, TPMS couldn't produce a porous structure fitting any irregular complex shape unless some mapping techniques, such as the distance field [25], T-spline [51], or hexahedral parameterization [82], is integrated. Unlike TPMS, a significant advantage of the proposed tetrahedral implicit surface method is that it can generate surfaces with same morphological features inside the tetrahedron of any shape. When multiple tetrahedra are considered, which is almost always true in modeling real-world shapes, the surfaces between neighboring tetrahedrons must be seamlessly connected at the joint. Otherwise, it may cause holes or gaps in the resulting surfaces. In this section, a mathematical geometry proof is provided to validate the  $C_0$  continuity of TIS between two adjacent tetrahedrons.

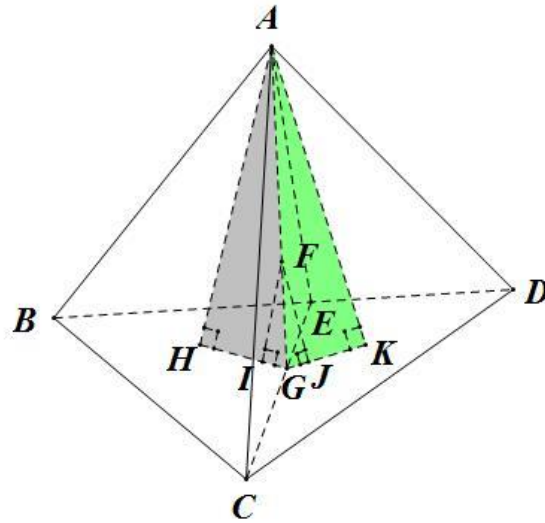


Figure 4.5: Proof of continuity property between adjacent tetrahedrons.

As shown in Figure 4.5,  $ABCE$  and  $ADCE$  are two adjacent tetrahedrons of arbitrary shapes that share the triangular face  $ACE$ .  $F$  is an arbitrary point on the face  $ACE$ .  $I$  and  $J$  are the

perpendicular projections of the point  $F$  onto the two faces  $BCE$  and  $DCE$ , respectively. Similarly,  $H$  and  $K$  are the projections of the point  $A$  onto the two faces  $BCE$  and  $DCE$ , respectively.

$$\therefore FI \perp BCE \text{ and } AH \perp BCE$$

$$\therefore FI \parallel AH$$

$$\therefore \Delta GFI \sim \Delta GAH$$

$$\text{so we get: } r1 = \frac{GF}{GA} = \frac{GI}{GH} = \frac{FI}{AH} \quad (4.3.1)$$

The same holds for tetrahedron  $ACDE$  and we have:

$$r2 = \frac{GF}{GA} = \frac{GJ}{GK} = \frac{FJ}{AK} \quad (4.3.2)$$

Combine Equation (4.2.5) and Equation (4.2.6), then:

$$r = r1 = r2 = \frac{GF}{GA} = \frac{FI}{AH} = \frac{FJ}{AK} \quad (4.3.3)$$

In tetrahedron  $ABCE$ ,  $\frac{FI}{AH} \cdot 2\pi$  and  $\frac{FJ}{AK} \cdot 2\pi$  is the coordinate of point  $F$  with respect to the faces  $BCE$  and  $CDE$ , respectively. Similarly, we can conclude that the other two coordinates of point  $F$  in tetrahedron  $ABCE$  with respect to faces  $ABC$  and  $ABE$  are the same as those of point  $F$  to faces  $ADC$  and  $ADE$  in tetrahedron  $ADCE$ . Meanwhile, the coordinate of  $F$  to face  $ACE$  equals to 0 in both tetrahedrons  $ABCE$  and  $ADCE$ , which means the four coordinates of point  $F$  are exactly the same in both of the two tetrahedrons. As a result, we can conclude that the TIS function values of all points lying on the shared face of two neighboring tetrahedrons will be the same, regardless of their shape.

### **4.2.3 Porous Structure Generation from Surface Models**

Based on the proposed TIS method, we can achieve porous scaffolds by tetrahedralizing the tissue with a customized configuration. Biological tissues are inherently heterogeneous architectures. At the macrostructure level, the tissue exhibits great differences in both morphology and bio-functionability. Pore-intensive parts have higher structural strength and can support most of the weight. Conversely, the sparsely populated parts have better permeability and transport of nutrients and metabolic waste. Figure 4.6 shows a flowchart of the details for designing a 3D heterogeneous porous scaffold from the 3D anatomical shape.



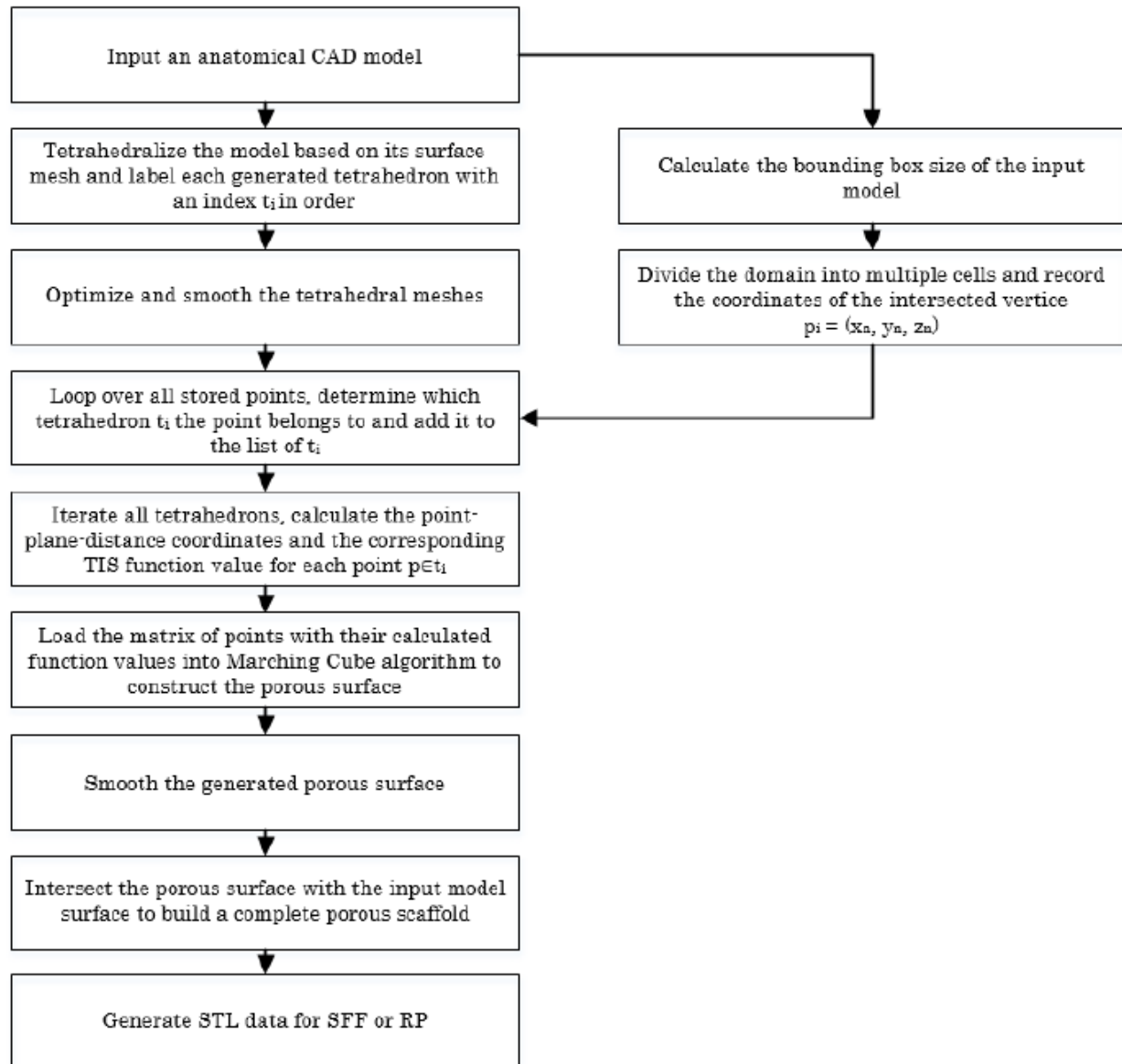


Figure 4.6. Flowchart showing the procedures for designing a 3D heterogeneous porous scaffold using the TIS method.

### 4.3 Experimental Results

In this section, various types of TIS surfaces are compared with the TPMS surfaces from different perspectives. The TIS algorithm is implemented in C programming language running on an 8×2.20-GHz Intel(R) Core(TM) i7 computer with 16 GB of memory. A volume of

200×200×200 voxels was used as the resolution for both TPMS and TIS to generate various types of surfaces. While different selections of the threshold  $C$  can significantly change the porosity and pore size, we chose the constant 0 as the threshold of the generation of P, D, and G surfaces for all TPMS models in this paper. For TIS surface generation, however, we used a concept of relative threshold, which is related to the actual threshold as follows:

$$\text{actual\_threshold} = (\text{max} - \text{min}) * \text{relative\_threshold} + \text{min} \quad (4.4.1)$$

in which  $max$  and  $min$  are the minimum and maximum of the calculated TIS function values from Equation (4.2.4). The range of the relative threshold was rescaled into 0 to 1 according to the actual threshold, which is the real threshold used in the experiments. The periodicity parameter  $\lambda$ , as used in Equation (4.2.3), was chosen as follows. To stereoscopically illustrate the fabricated TIS surfaces with a moderate pore size, 0.16 and 0.44 were used as the thresholds for P surface ( $C = 0.16$  when  $\lambda = 1$ ,  $C = 0.44$  when  $\lambda = 2$  for the generation of TIS surfaces). Also, we used the threshold of 0.67 for the TIS D surface and a threshold of 0.5 for the TIS G surface both for  $\lambda = 1$  and  $\lambda = 2$ .

In Figure 4.7, 4.8, and 4.9, the surfaces in the two left columns are TPMS Schwarz P, D, and G surfaces, while the right two columns contain the TIS surfaces of the same type, respectively. The meshes in the first row were constructed using a single period ( $\lambda = 1$ ), while the meshes in the second row were constructed using a double period ( $\lambda = 2$ ) on each axis instead. From these figures, it is not hard to figure out that the shapes of the surfaces and the locations of the holes are very similar between the same types of surfaces for different groups. For example, the P surfaces in both TPMS and TIS have an opening on each face of the unit domain (either tetrahedron or hexahedron). The opening on the tetrahedral face is an ellipse of the triangle outline, while the opening on the hexahedral face is a regular circle. The interiors are

also connected, which allows for cell growth through the holes on the surface. In addition, Figures 4.10, 4.11, and 4.12 also give closer views of the surfaces between two neighboring tetrahedrons. And Figure 4.13 shows the generated porous surface obtained from an icosahedron.

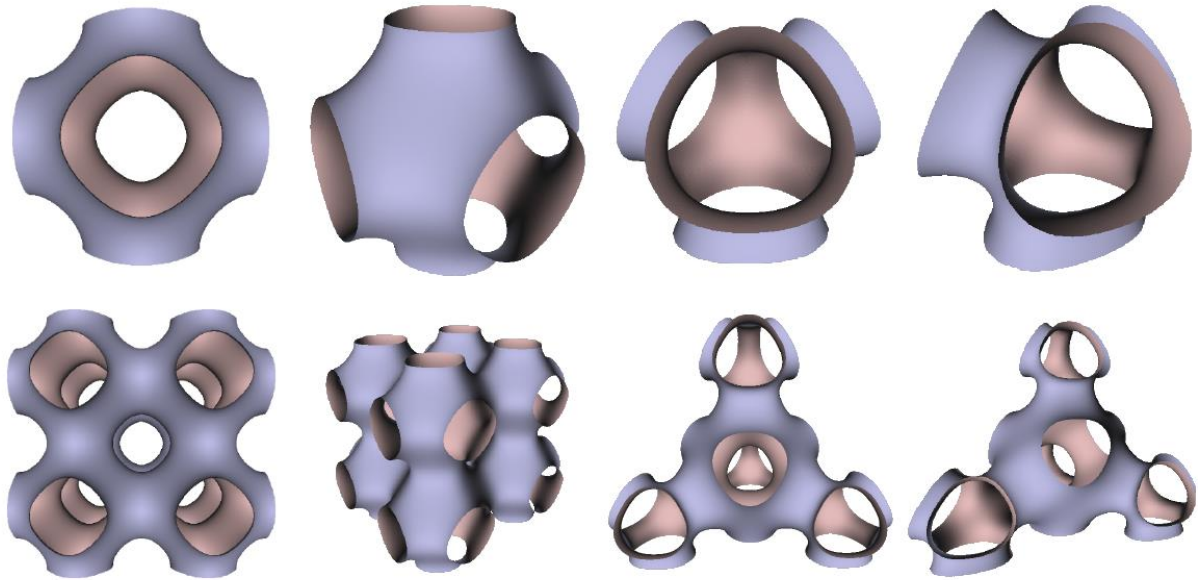


Figure 4.7: TPMS Schwarz P surface (left two columns) vs. TIS P surface (right two columns).

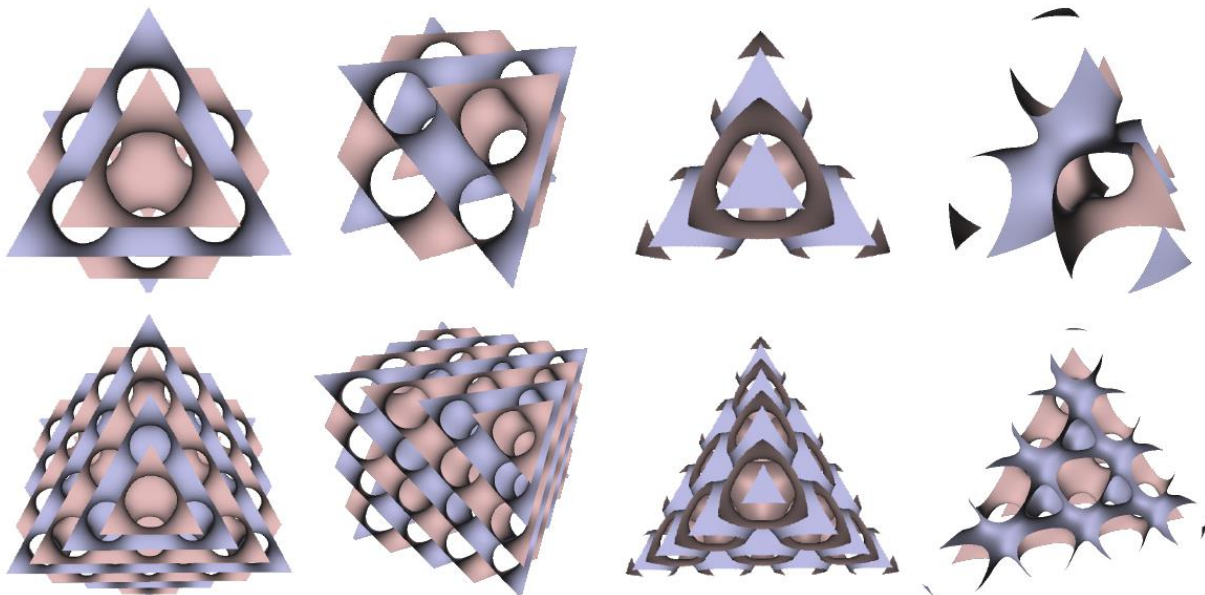


Figure 4.8: TPMS Schwarz D surface (left two columns) vs. TIS D surface (right two columns).

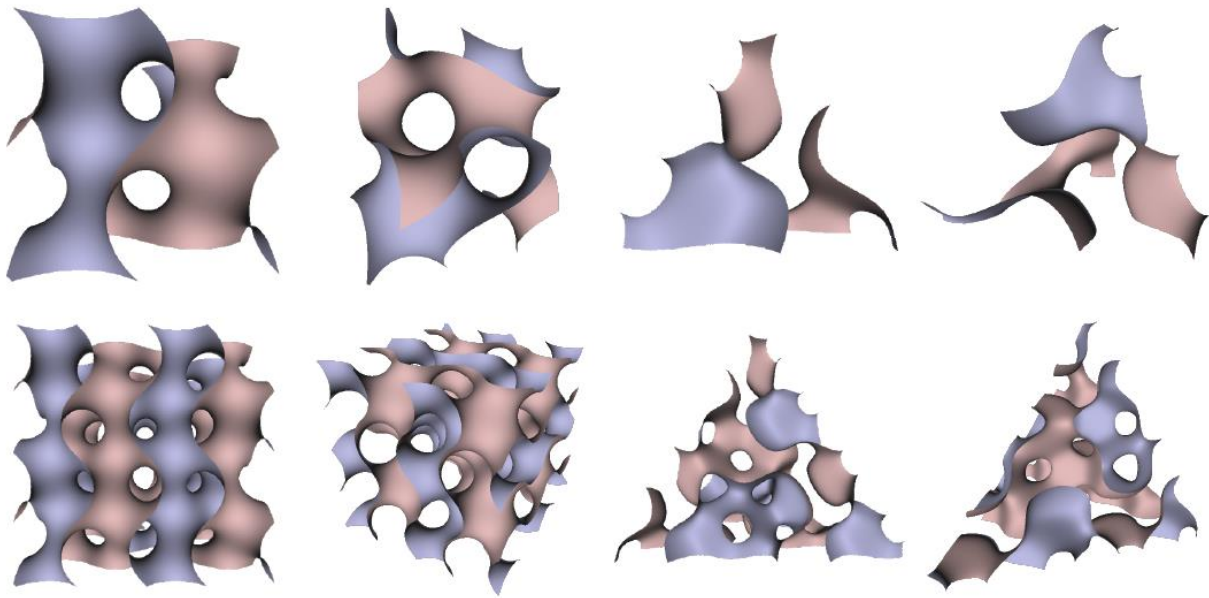


Figure 4.9: TPMS Schwarz G surface (left two columns) vs. TIS G surface (right two columns).

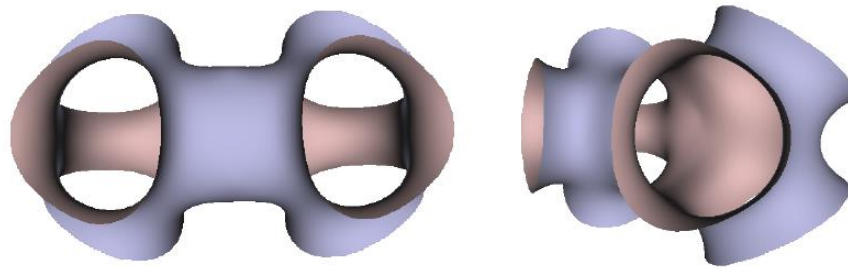


Figure 4.10: Two views of the TIS P surface generated from two connecting tetrahedrons.

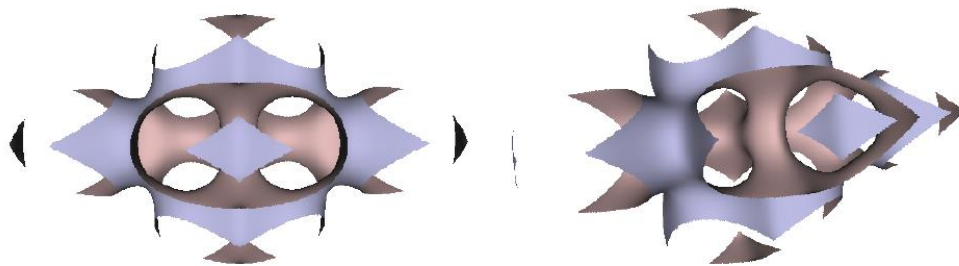


Figure 4.11: Two views of the TIS D surface generated from two connecting tetrahedrons.

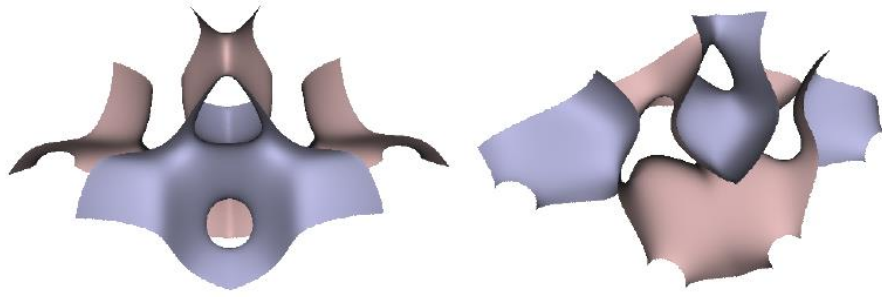


Figure 4.12: Two views of the TIS G surface generated from two connecting tetrahedrons.

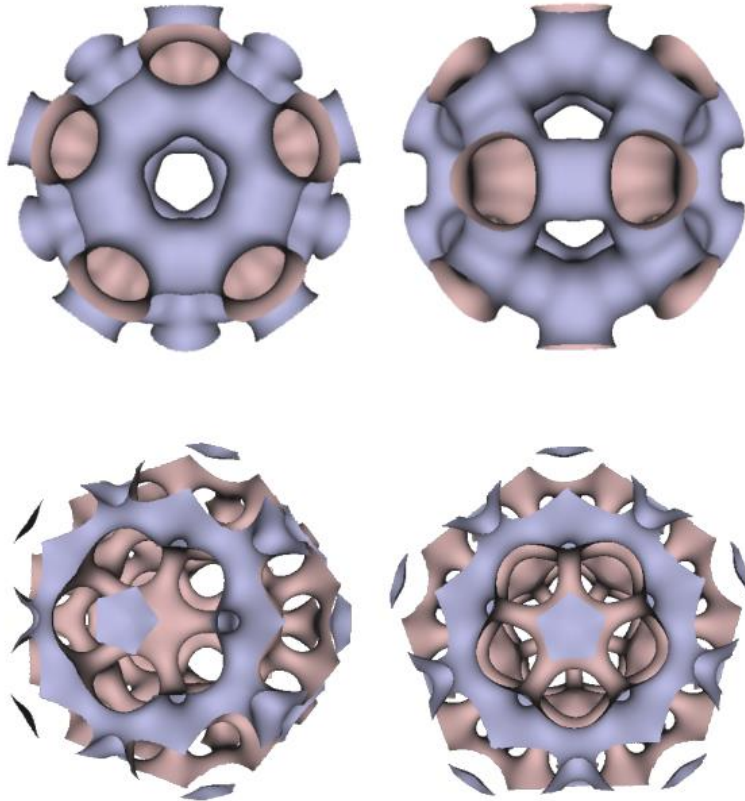


Figure 4.13: TIS surfaces generated from an icosahedron (P surface – top, D surface - bottom).

To further analyze the characteristics of these tetrahedral implicit surfaces, we first used the well-known open source CAD software, Meshlab [137], to compute the average curvatures

over the whole surface. Figure 4.14 demonstrates the distributions of curvatures of different surfaces, as well as their corresponding detailed histograms. The color map ranges from red to blue indicating the curvature values from the minimum of -30 to the maximum of 30 (the values on X-axis are curvatures in ascending order while the values on Y-axis are the number of points that whose curvatures share the same value). Additionally, more information including the minimum/maximum curvature and the mean curvature and also the standard deviation of the curvatures is also provided for each of the TPMS and TIS surfaces in Table 4.1. In the table, the minimum curvatures of TIS P-surface and G-surface are almost double of the values of TPMS P-surface and G-surface. And only the maximum curvature of TIS P-surface is significantly larger than the same type TPMS surface. Moreover, the standard deviation of the curvatures of TIS surfaces is relatively larger than that of TPMS surface of the same type. From Figure 4.14 and Table 4.1 it's not hard to conclude that the distributions of curvatures of TPMSs are all symmetrical while this symmetry doesn't apply to TISs. Nonetheless, one of the most important common attributes for TPMS and TIS is that the overall distribution of the curvatures tends to converge to the mean curvature which approximates to 0.

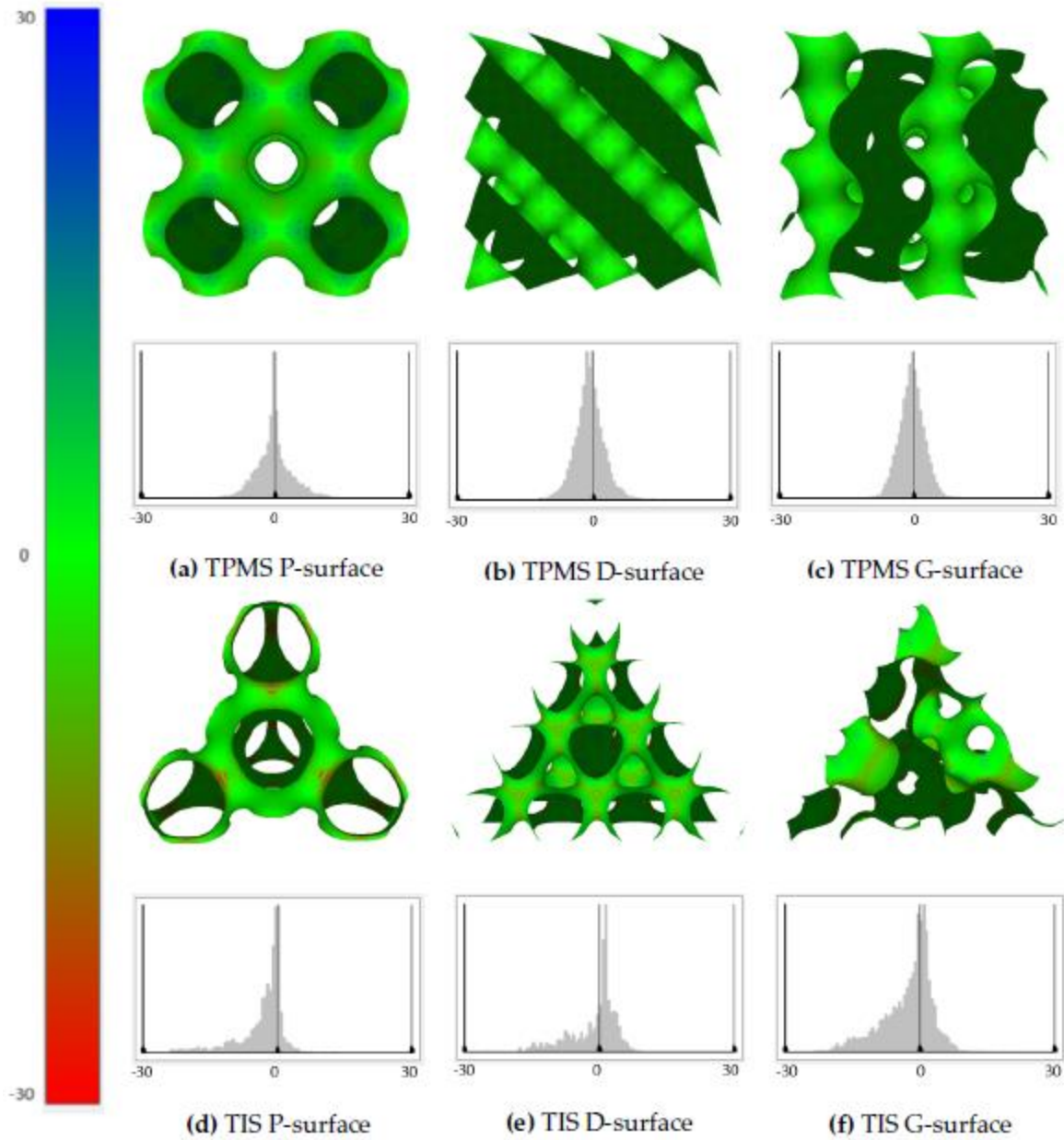


Figure 4.14: Distribution maps of the mean curvature and corresponding histograms for TPMSs and TISs.

Curvatures	TPMS Surfaces			TIS Surfaces		
	P-Surface	D-Surface	G-Surface	P-Surface	D-Surface	G-Surface
Minimum	-12.66	-14.99	-14.34	-33.19	-14.88	-26.48

Maximum	12.59	13.83	18.32	21.09	12.16	15.17
Medium	-0.57	-1.18	-0.66	-1.68	0.45	-1.68
Average	-0.49	-1.29	-0.79	-2.88	-0.94	-2.79
Standard Deviation	3.78	3.32	2.85	4.11	4.32	4.51
Variance	14.28	11.04	8.11	16.88	18.63	20.36

Table 4.1 Statistical analysis to the curvatures of TPMSs and TISs.

Besides, we also realized that the porosity and pore size of the surfaces are quite sensitive to the selection of threshold. Through experiments we found that not all thresholds in the range from 0 to 1 can generate a completely connected surface. To create the TIS surfaces of a single period from a regular tetrahedron, the ideal threshold range for P-surface is 0.118 - 0.275, for D-surface is 0.569 - 0.765 and for G-surface is 0.431 - 0.627. A value out of these ranges will cause the generated corresponding type of surface deformed, which means the pore will disappear. On the other hand, TPMS surfaces also have the same issues of surface discontinuity and deformation as TIS surfaces. Nonetheless, the difference is that the selective areas of the reasonable threshold for TPMSs all uniformly concentrate on the range from -1 to 1. Table 4.2 gives more information of the relationship between measured porosities and different thresholds  $C$ . Additionally, some examples of the deformed TPMSs and TISs with a threshold that exceeds the proper range are demonstrated in Figure 4.15. It is noted that the hole size gets bigger and the porosity behaves more saturated when  $C$  gets increased in the given range for both TPMS and TIS. Meanwhile, the surface areas of different TPMS and TIS surfaces for a unit volume are measured by the sum area of the generated mesh triangles, and the results are presented in Table



4.3 as a reference. All related parameters, including the resolution and threshold, are the same as mentioned at the beginning of Section 4.4.

TPMS P-Surface		TPMS D-Surface		TPMS G-Surface	
Threshold $C$	Porosity (%)	Threshold $C$	Porosity (%)	Threshold $C$	Porosity (%)
-0.95	22.79	-0.95	10.28	-0.95	18.77
-0.5	35.71	-0.5	29.42	-0.5	33.81
0	49.98	0	49.94	0	50.00
0.5	64.28	0.5	70.57	0.5	66.19
0.98	77.21	0.95	89.71	0.95	81.22

TIS P-Surface		TIS D-Surface		TIS G-Surface	
Threshold $C$	Porosity (%)	Threshold $C$	Porosity (%)	Threshold $C$	Porosity (%)
0.118	18.36	0.569	57.35	0.431	44.03
0.157	35.80	0.618	66.88	0.480	57.29
0.196	49.36	0.667	75.73	0.529	67.76
0.235	61.17	0.716	83.04	0.578	76.09
0.275	71.86	0.765	89.90	0.627	84.04

Table 4.2 Effect of diverse thresholds  $C$  on different TPMS and TIS surfaces.

Surface Type		Surface Area for Unit Volume
TPMS Surfaces	P-Surface	2.353180
	D-Surface	3.828864
	G-Surface	3.083422

TIS Surfaces	P-Surface	3.798055
	D-Surface	2.367160
	G-Surface	2.654556

Table 4.3 Surface areas of TPMSs and TISs for a unit volume.

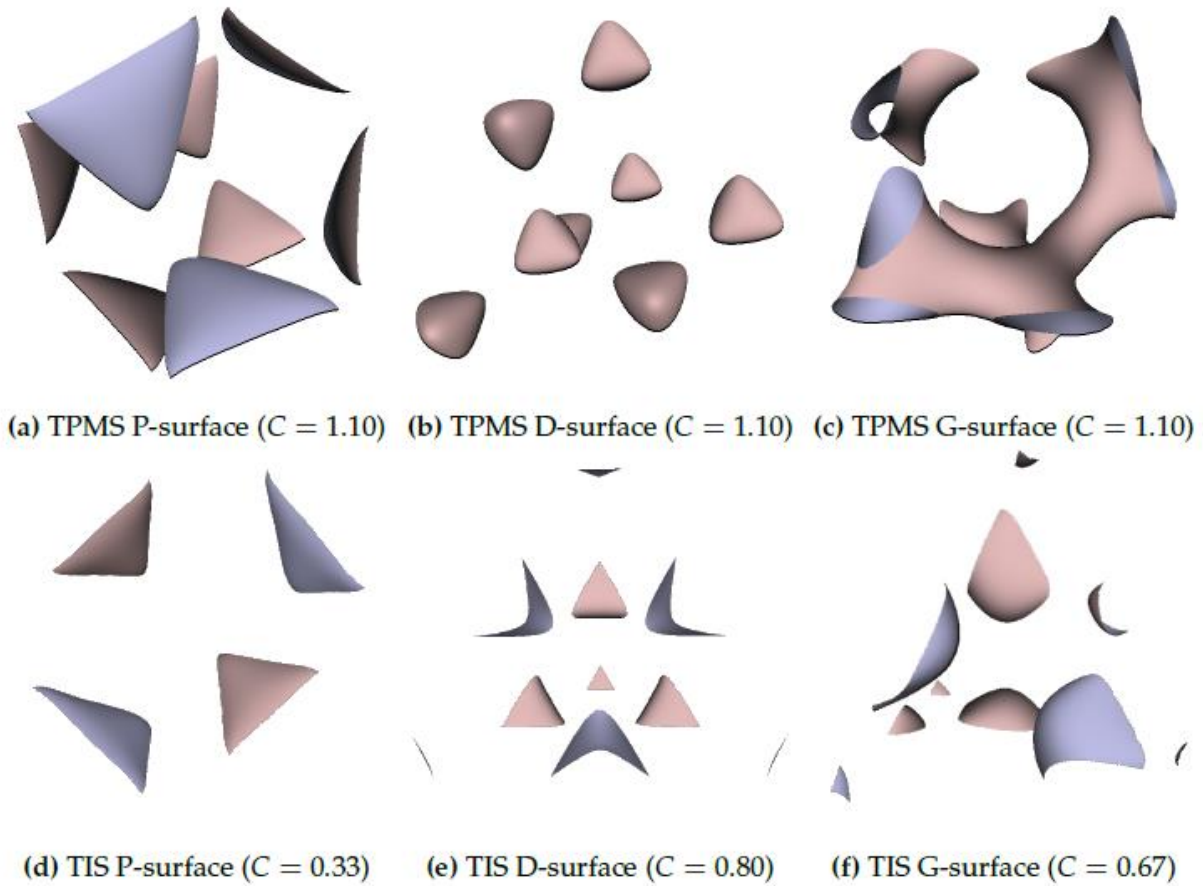


Figure 4.15: Deformed TPMS and TIS surfaces with a threshold  $C$  that exceeds the reasonable range.

Moreover, by analyzing the properties of the P-, D-, G-type surfaces generated by TIS, we found that the TIS P surfaces offer better quality in constructing porous architectures from large-scale models. An example showing the comprehensive process of constructing 3D porous scaffold based on TIS P surface is given as follows. Firstly, as shown in Figure 4.3 (a), the 3D femur model

is tiled into 989 tetrahedra of different sizes and shapes. Secondly, we apply TIS algorithm to each of the tetrahedrons, resulting in 989 TIS P surfaces. As we have proved, the continuity between adjacent TIS surfaces are guaranteed. Thirdly, we smooth the porous network using Laplacian smoothing and curvature-based smoothing algorithms. Lastly, as shown in Figure 4.16, the smoothed porous surface is intersected with the original input femur model to generate a porous femur scaffold. Figure 4.17 gives the cross-section of the generated porous scaffold.

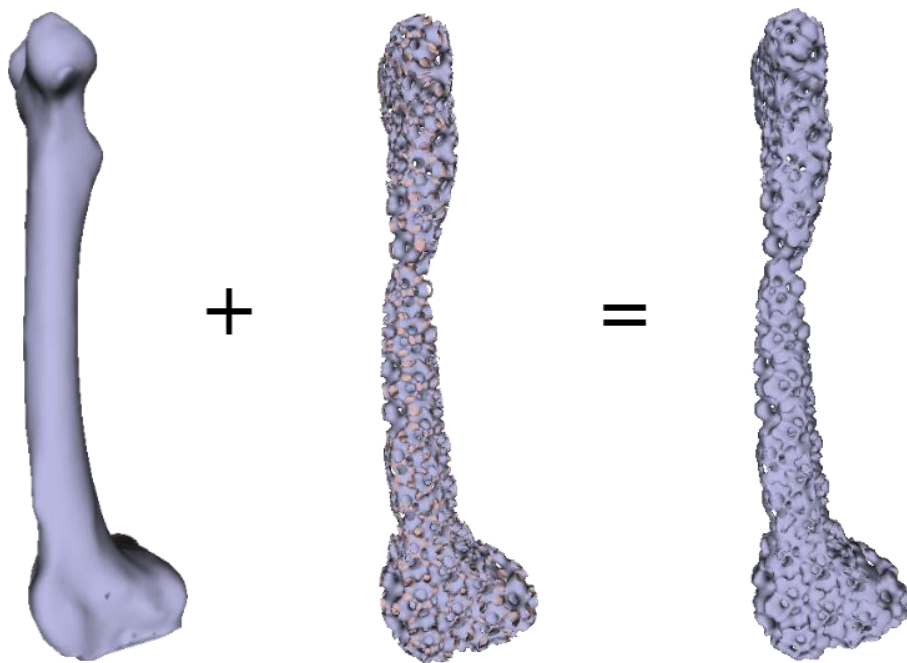


Figure 4.16: Porous femur generated by TIS P-surface.

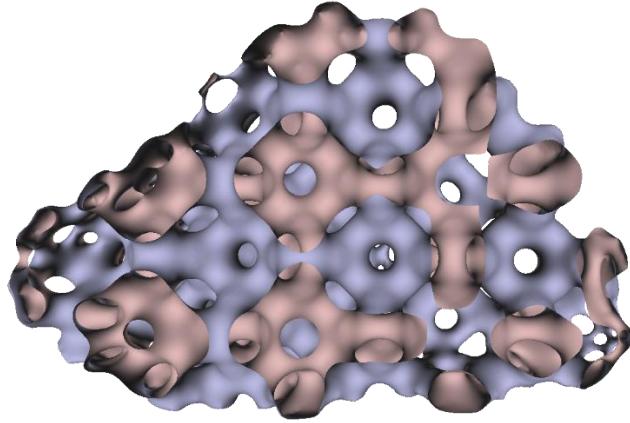


Figure 4.17: Sectional view of porous femur generated by TIS P-surface.

In the process of generating TIS surfaces, with varying resolutions chosen for each single tetrahedron domain, the computational time and the consumed memory change correspondingly. To construct a high-quality porous surface, higher resolution is demanded. Meanwhile, the heavy computation for the computer is also required. Under the assumption that a single period of P type surface with the unit volume is generated, a comparison of the performance between the TPMS algorithm and TIS algorithm is offered in Table 4.4. From the results, you can see that our TIS method generates fewer grids in less time than TPMS with the same resolution.

Resolution	TIS		TPMS	
	Computational Time (s)	Number of Triangles	Computational Time (s)	Number of Triangles
100	0.195	15329	0.355	116492
200	1.451	63293	2.756	471692
300	4.829	143766	9.382	1067468

400	11.575	256662	23.78	1900676
500	22.615	402198	43.276	2975372

Table 4.4: Running time comparison between TIS and TPMS.

## 4.4 Summary

In this chapter, an implicit 3D surface method for designing bio-scaffolds based on the tetrahedral implicit surface is proposed. To the best of our knowledge, this is the first attempt to construct implicit surface based on tetrahedral elements. The TIS method has demonstrated its great potential in the field of designing optimized porous structure. Comparing to the well-established TPMS approach, this concise method brings several advantages to the tissue engineering scaffold design:

- 1) The generation of TIS surface could avoid the restriction of modeling based on regular unit domain such as regular hexahedron in TPMS. Even in the deformed tetrahedron, there still can create characteristic tetrahedral surface and guarantee  $C^0$  continuity between adjacent TIS surfaces;
- 2) Different from the TPMS method described in Chapter 3, which needs to parametrize the tissue surface into hexahedral meshes first, it is more convenient to tetrahedralize the triangular surface mesh as in the proposed TIS method. Moreover, without integrating the special mapping procedure makes the entire process simplified;
- 3) The strong interconnectivity and tetrahedron-base modeling grant the TIS method more flexibility and creativity for complex shape modeling.

Notwithstanding, there are also several limitations to the TIS method. Though the TIS surface of multiple period obtained from a single unit has shown the characteristics of approximate zero mean curvature and  $C^0$  continuity, the  $C^1$  continuity of the surface at the junction of two neighboring tetrahedrons cannot be guaranteed. Moreover, some isolated closed surfaces could be rendered using the TIS D-surface when the input is a complicated shape model.

# Chapter 5

## **Porous Structure Design Using Anisotropic Radial Basis Function**

In the last two chapters, we have discussed porous scaffold designs based on hexahedron and tetrahedron. It's true that using subdivided volumetric mesh could reproduce periodic porous elements. The resulting pore network has some outstanding characteristics such as the consistent shape of pores, strong connectivity, and controlled porosity over the entire network. Nonetheless, the uniform distribution of identical pores also exposes the problem of lacking flexibility and randomness. That means changing the unique iso-value  $C$  would adjust the porosity and the pore size through the entire surface. And a small modification to the equation of implicit surface can definitely change the architecture of the whole pore network. As a result, these features make an inhomogeneous design of porous structure with different pore sizes almost an impossible task.

In this chapter, we try to turn a new route to design and construct porous architecture using anisotropic radial basis function (ARBF). This proposed method not only can produce porous models from hexahedral or tetrahedral units but also can use spatial point cloud to generate pores. It has three major advantages over the implicit surface modeling methods that were introduced in the last three chapters.

- 1) Firstly, adaptive regional modifications to the pore shape, pore size or pores' distribution can be easily achieved by adjusting the local configuration of the sub-volumes or the point cloud accordingly.
- 2) Secondly, this approach is flexible to simulate heterogeneities and discontinuities in natural tissue structures by using purposely-designed point cluster as the input. For example, if we want a specific region of the tissue to be sparse, then we just need to reduce the distributed points in that region. On the contrary, the operation of assigning more points could generate the denser pores in that region.
- 3) Thirdly, unlike other implicit methods that need post-actions such as Boolean operations to get the final pieces built, the only post-action in this approach is to retrieve iso-surfaces using marching cube algorithm, which makes it much easier, faster and more flexible to implementation.

In order to assign points equably inside the space of a scaffold, we would use the vertices of the tetrahedralized volumetric mesh as the input to build spatial pore network. The adopted tetrahedral meshing and smoothing algorithm is the BCC-based one introduced in Section 4.1.

## 5.1 Traditional Radial Basis Function (RBF) Interpolation

The conventional radial basis function interpolation is given by

$$f(\mathbf{x}) = \sum_{i=1}^N w_i \phi(\|\mathbf{x} - \mathbf{x}_i\|) \quad (5.1.1)$$



where the interpolated function  $f(x)$  is represented as a weighted sum of  $N$  radial basis functions  $\phi(\cdot)$  and each function is centered differently at  $x_i$  and weighted by  $w_i$ . Once the values of  $f_i$  where  $i = 1 \dots N$  are given at  $x_i$ , the weights  $w_i$  can be solved through

$$\begin{bmatrix} \phi_{11} & \cdots & \phi_{1N} \\ \vdots & \ddots & \vdots \\ \phi_{N1} & \cdots & \phi_{NN} \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_N \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_N \end{bmatrix} \quad (5.1.2)$$

where  $\phi_{ji} = \phi(\|x_j - x_i\|)$ . Accordingly, the value at an arbitrary point can be evaluated through Equation (5.1.1).

In conventional RBF method, the distance between point  $x \in R^d$  and center of basis function  $x_i \in R^d$  is measured by Euclidean distance which is denoted as  $r = \|x - x_i\|$ . Some commonly used radial basis functions are listed in Table 5.1.

Types of basis function	Definition
Gaussian	$\phi(r) = e^{-(cr)^2}$
Multiquadrics (MQ)	$\phi(r) = \sqrt{r^2 + c^2}$
Inverse multiquadrics (IMQ)	$\phi(r) = \frac{1}{\sqrt{r^2 + c^2}}$
Thin plate spline (TPS)	$\phi(r) = r^2 \ln(r)$

Table 5.1: Common radial basis functions.

In radial basis functions, the shape parameter  $c$  plays an important role in improving the accuracy of numerical solutions. In general, the selection of optimal shape parameter depends on densities, distributions and function values at all mesh nodes. The selection of adaptive shape

parameter has been an active topic in approximation theory [83]. Interested readers can refer to [84-88] for more details.

## 5.2 Anisotropic Radial Basis Function (ARBF) Interpolation

Conventional RBF interpolation is isotropic to all directions such that the target domain of any basis function tends to be circular in 2D or spherical in 3D, which makes it difficult to create comprehensive porous topological structures. Anisotropic RBF, on the other hand, could perfectly overcome the limitation of constructing a tailored directional porous network. The major feature of anisotropic RBF interpolation is to assign the basis functions consistent directionality with the generated pores by redefining the distance functions. Figure 5.1 (a) and (b) depict the support domains by using isotropic RBF and anisotropic RBF in a 2D triangular mesh, respectively.

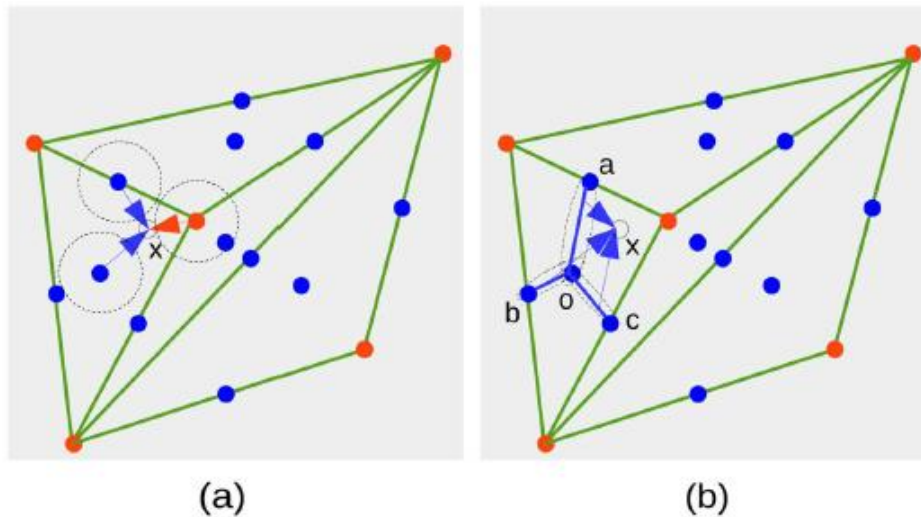


Figure 5.1: 2D interpolation schemes.  $x$  is the pixel to be interpolated. Dashed circle (for RBF) or ellipses (for anisotropic RBF) describe the support domains of underlying basis functions. (a) RBF interpolation. (b) Anisotropic RBF interpolation.

In anisotropic RBF interpolation, a new basis function centered at a line segment rather than a point is introduced such that the resulting surface is connected along the line's direction. Given a combination of line segments and single points  $L = \{l_j\}_{j=1\dots N}$  as the centers of basis functions, the anisotropic RBF interpolation is described as

$$\Phi(u) = \sum_{j=1}^N \omega_j \phi(\|u - l_j\|_L) \quad (5.2.1)$$

where  $\|u - l_j\|_L$  denotes the distance between two points or two line segments or a point and a line segment. The distance between two points are still measured using Euclidean distance in ARBF. However, the distance between two line segments or between a point and a line segment is redefined as follows. Assume we are calculating the distance between point  $x$  and line segment  $(a, b)$ , there are three possible cases.

- 1) For case 1, if point  $x$  is on the line segment  $(a, b)$ , then the distance between  $x$  and  $(a, b)$  directly equals to 0.
- 2) For case 2, if point  $x$  and the endpoints of line segment  $(a, b)$  form an acute triangle, the distance is evaluated as the perpendicular distance from  $x$  to line segment  $(a, b)$ .
- 3) For case 3, if  $x$  and the endpoints of line segment  $(a, b)$  form an obtuse triangle, the distance is defined as  $\min\{\|x - a\|, \|x - b\|\}$ .

Figure 5.2 (a-c) illustrates the three distances between point  $x$  and line segment  $(a, b)$  under the above circumstances. Besides, the distance between two line segments  $(a, b)$  and  $(c, d)$  is defined as  $\min\{\|a - c\|, \|a - d\|, \|b - c\|, \|b - d\|\}$ . An example is provided in Figure 5.2 (d).

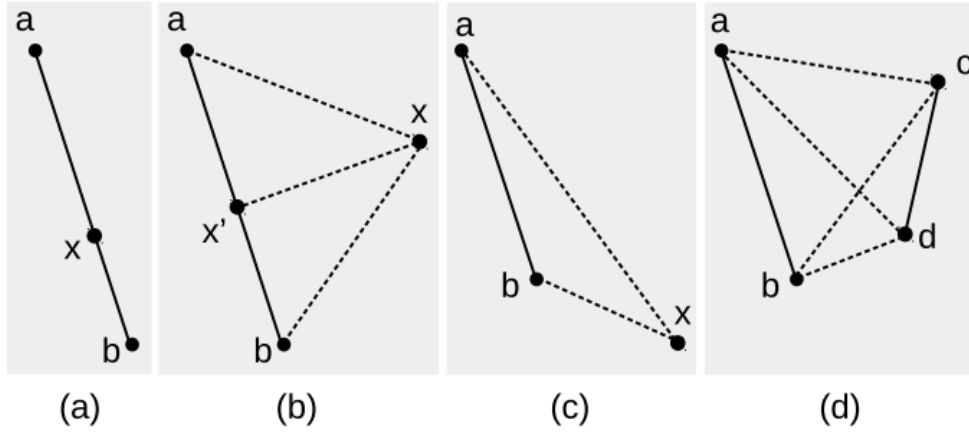


Figure 5.2: Distance definitions used by ARBF. (a) Point  $x$  is on line segment  $(a, b)$ . (b) Point  $x$  and line segment  $(a, b)$  form an acute triangle. (c) Point  $x$  and line segment  $(a, b)$  form an obtuse triangle. (d) Distance between line segments  $(a, b)$  and  $(c, d)$ .

The new distance functions of ARBF take both magnitude and direction into account to change the support domain of the basis function from a regular circle or sphere to an ellipse or ellipsoid, which enables the conventional RBF method with some new properties to control the shapes of internal pore structures.

### 5.3 Algorithm Description

Anisotropic RBF interpolation mainly involves three stages. The first is function-fitting by solving a linear system corresponding to the interpolation conditions. To figure out the weight coefficients of the basis functions in Equation (5.1.2), some specified values are assigned to the given mesh. In the current method, the values of all mesh vertices are assigned 1. For 2D meshes, the middle point of each edge, the face (triangle or quadrangle) center and also the line segments from the center to the middle point are all set to -1. For 3D meshes, the centers of surfaces and the centers of sub-volume (tetrahedron or hexahedron) are assigned -1. Moreover, the line segments from the volume center to the face centers are assigned -1 as well. Figure 5.3

(a-c) demonstrates the distribution of values at the centers of all radial basis functions in a 2D triangular mesh, a 3D tetrahedral mesh, and a 3D hexahedral mesh, respectively. Secondly, once the weight coefficients of the basis functions are solved using the revised distance definitions, the function values of the interpolated points could be easily calculated through Equation (5.2.1). In our algorithm, the bounding box of the target object is partitioned into a matrix of small cubes of the same size, and the vertices of the small cubes are chosen as the interpolation points. Lastly, an iso-surface is extracted using the well-known Marching Cube algorithm.

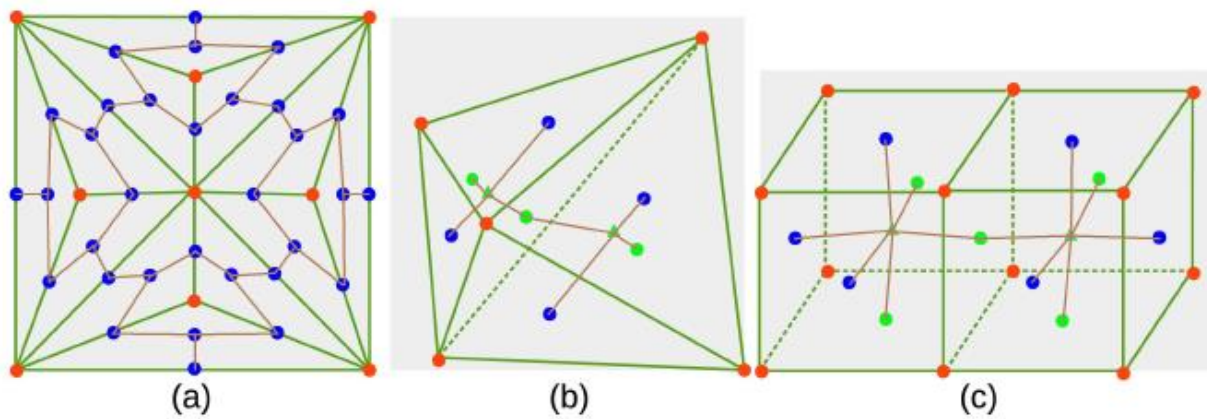


Figure 5.3: Values assigned to mesh nodes. Red dots indicate value 1 is assigned while blue dots indicate value -1 is assigned. Brown lines indicate value -1 is assigned as well. In 3D meshes, green dots represent interior nodes. (a) Sample distribution of node values in the 2D triangular mesh. (b) Sample distribution of node values in the 3D tetrahedral mesh. (c) Sample distribution of node values in the 3D hexahedral mesh.

In general, a finite number of mesh nodes can be used to initialize the matrix in Equation (5.1.2) with a reasonable size for simple scaffolds. Nonetheless, the matrix will become singular or extremely time-consuming to solve when a complex model is given as the input. To this end, the global interpolation method (see Figure 5.4) can be substituted with a local ARBF interpolation method (see Figure 5.5). Instead of solving the coefficients for all nodes at once, a set of adaptive coefficients are figured out for each unit domain. At the same time, the nodes from neighboring domains will be taken into consideration to ensure the connectivity and

smoothness between the target domain with surrounding domains. To identify the neighboring domain of each point instantly and accurately, we integrate the well-known k-d tree algorithm [89] into our local interpolation scheme.

```
ARBFIInterpolation()
{
    // load mesh points into memory
    loadMesh();

    // assign values to mesh nodes and build matrix
    buildMatrixs();

    // solve unknown weight coefficients
    solveCoefficients();

    // construct a 3D array to contain
    // interpolated values
    construct3DDataset();

    // calculate the value for each point
    // interpolate it into the output array
    for each point  $x$  with same interval do
        sum  $\leftarrow$  0;
        for each center of basis function do
            distance  $\leftarrow$  computeDistance();
            sum  $\leftarrow$  sum + distance * coefficients[i];
        end for
        dataset[x]  $\leftarrow$  sum;
    end for

    // extract iso-surface using MC algorithm
    marchingCube();

    // print/export resulted mesh surface
    exportResult();
}
```

Figure 5.4: ARFB global interpolation.

```

ARBFLocalInterpolation()
{
    // load mesh points to construct k-d tree
    loadMeshAndConstructKDTree();

    // construct a 3D array of points with same
    // length interval
    construct3DDataset();

    for each point  $x$  in the dataset do
        // include all mesh nodes with a given
        // range of  $\theta$  centered at point
        //  $x$  into an empty list
        includeNearNodes();

        // assign values to each node in the
        // list and build matrix
        buildMatrixs();

        // solve unknown weight coefficients
        solveCoefficients();

        sum  $\leftarrow$  0;
        for each nearby mesh node do
            distance  $\leftarrow$  computeDistance();
            sum  $\leftarrow$  sum + distance *
                coefficients[i];
        end for

        dataset[x]  $\leftarrow$  sum;
    end for

    // extract iso-surface using MC algorithm
    marchingCube();

    // print/export resulted mesh surface
    exportResult();
}

```

Figure 5.5: ARFB local interpolation.

## 5.4 Experimental Results

Given an iso-value and a specified shape parameter, a variety of porous architectures can be obtained by determining the locations of various centers of the radial basis functions within a given boundary constraint. In this section, we choose the inverse multiquadric (IMQ) function as basis functions and use a constant  $C = 0.1$  as its shape parameter. A proper iso-value is applied to each independent porous structure. Figure 5.6 (a) shows a pore structure obtained from single regular tetrahedron whose opening grows from the center of the tetrahedron toward the centers of its four triangle surfaces. Figure 5.6 (b) presents a surface inserted into two adjacent regular tetrahedrons. Figure 5.6 (c) describes a pore network comprised of the ARBF surface pieces from twenty tetrahedrons. Figure 5.7 (a) shows the surface interpolated from a single hexahedron. Figure 5.7 (b) presents the structure obtained from eight hexahedrons in a cube shape. Figure 5.7 (c) describes the surface obtained from a rod comprised of four hexahedrons in a row. Furthermore, diverse basis functions lead to distinctive patterns of morphology. Interpolated results using different basis functions, including multi-quadric (MQ) function, inverse multiquadric (IMQ) function, Gaussian function and thin plate spline (TPS) function, are illustrated in Figure 5.8. To further study the practicality of ARBF surfaces in biomedical engineering, we manufactured a realistic porous femur scaffold in 3D vision. At first, a scanned femur model was tiled into 989 pieces of arbitrary tetrahedrons. Here we use a body-centered cubic (BCC) lattice [76] based tetrahedral mesh generation algorithm followed by a quality tetrahedral mesh smoothing via boundary-optimized Delaunay triangulation (B-ODT) [79-80]. For each tetrahedron, we use the four vertices, the centers of four triangle faces, the tetrahedron center, and the line segments from the triangle centers to the tetrahedron center as the centers of basis functions. Moreover, the basis functions from the first layer of surrounding tetrahedrons are



included in current tetrahedron's linear system to ensure the pore surface smoothly connected with the surfaces from adjacent tetrahedrons. Consequently, a large number of connected porous structures are formed tetrahedron by tetrahedron as illustrated in Figure 5.9.

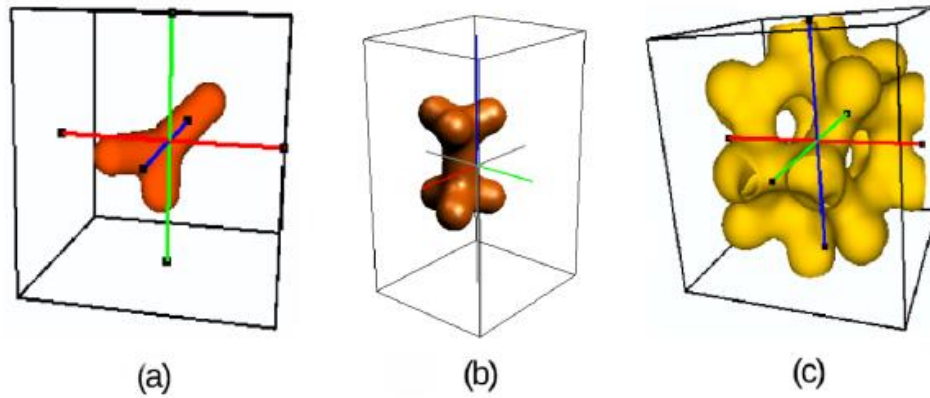


Figure 5.6: Generated porous surfaces in regular tetrahedrons. (a) Surface in single regular tetrahedron. (b) Surface in two connecting tetrahedrons. (c) Surface in an icosahedron.

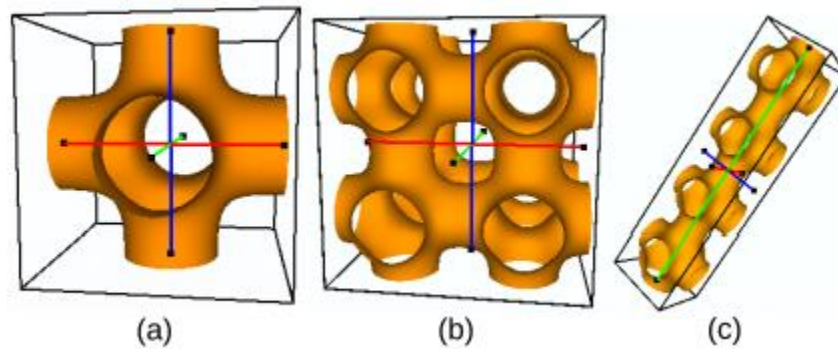


Figure 5.7: Generated porous surface for regular hexahedrons. (a) Surface in single hexahedron. (b) Surface in 2x2 stack of hexahedrons. (c) Surface in 4 hexahedrons arranged to form a rod.

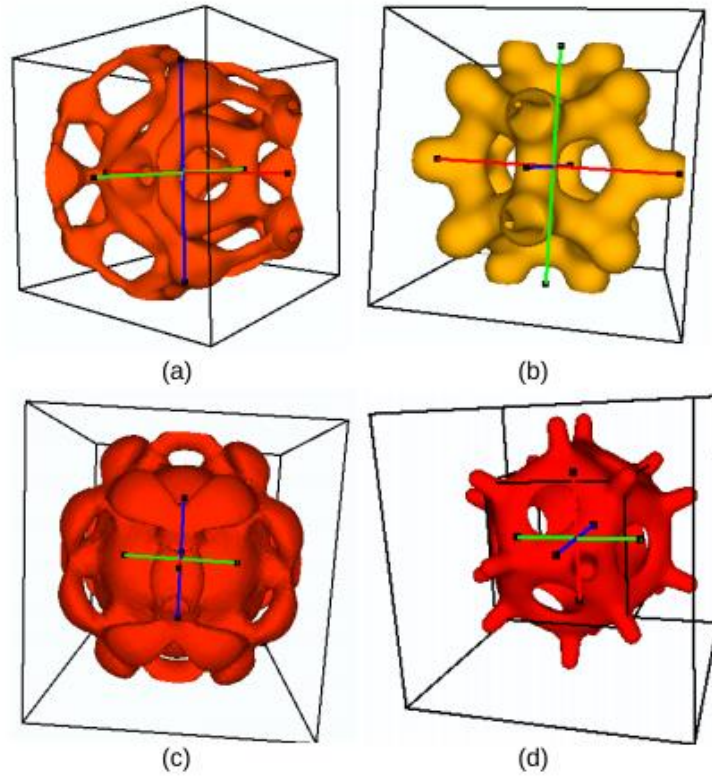


Figure 5.8: Results obtained from icosahedron domain using different basis functions. (a) multiquadrics (MQ). (b) inverse multiquadrics (IMQ). (c) Gaussian. (d) thin plate spline (TPS).

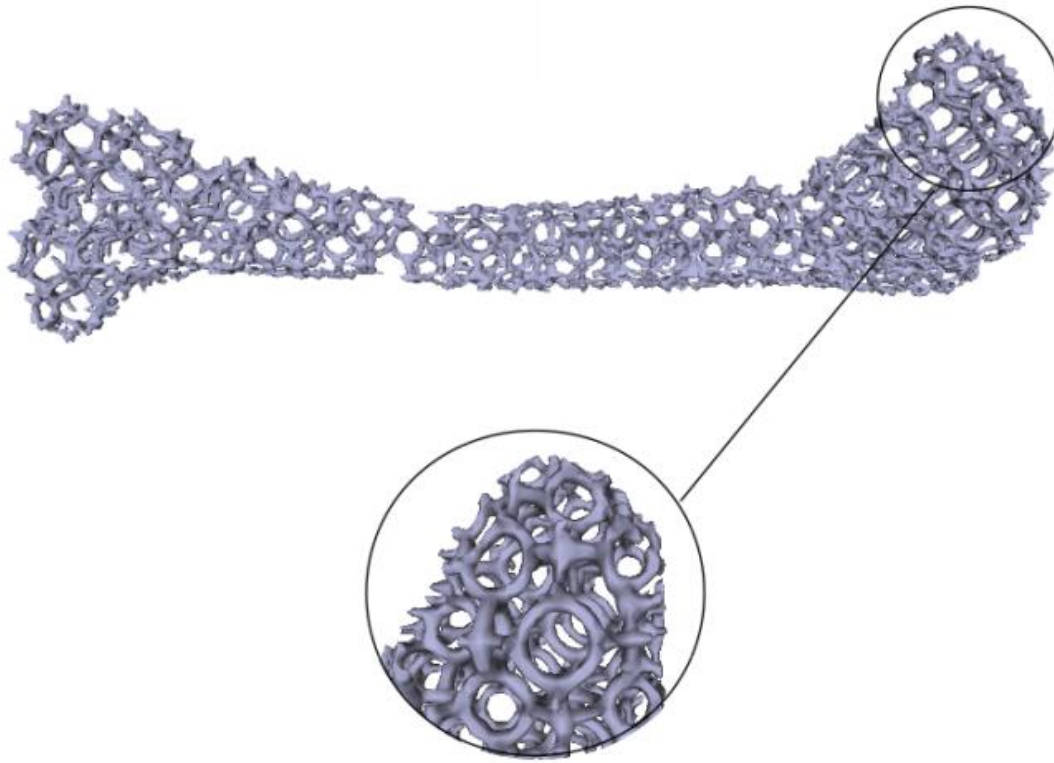


Figure 5.9: Interpolated porous inner structure of femur scaffold.

The randomness of porous distribution can be enhanced by moving the centers of basis functions toward arbitrary directions, which is a viable method to manufacture more realistic tissue scaffold with complex physiological architectures. Figure 5.10 shows some randomized mesh surfaces generated from the hexahedral domain. Moreover, the surface appearance and the porosity would change hugely when the iso-value or the shape parameter varies. Figure 5.11 (a-d) reveal that the porosity increases as the iso-value augments.

respectively.

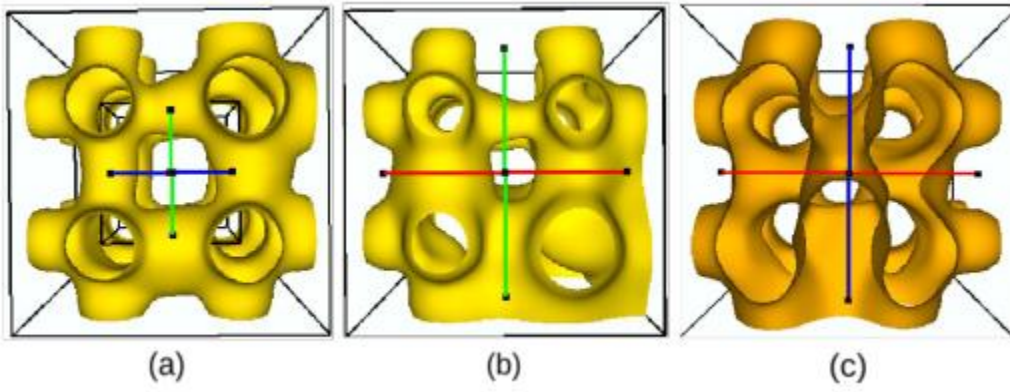


Figure 5.10: Resulted meshes from hexahedron domain with disturbance.

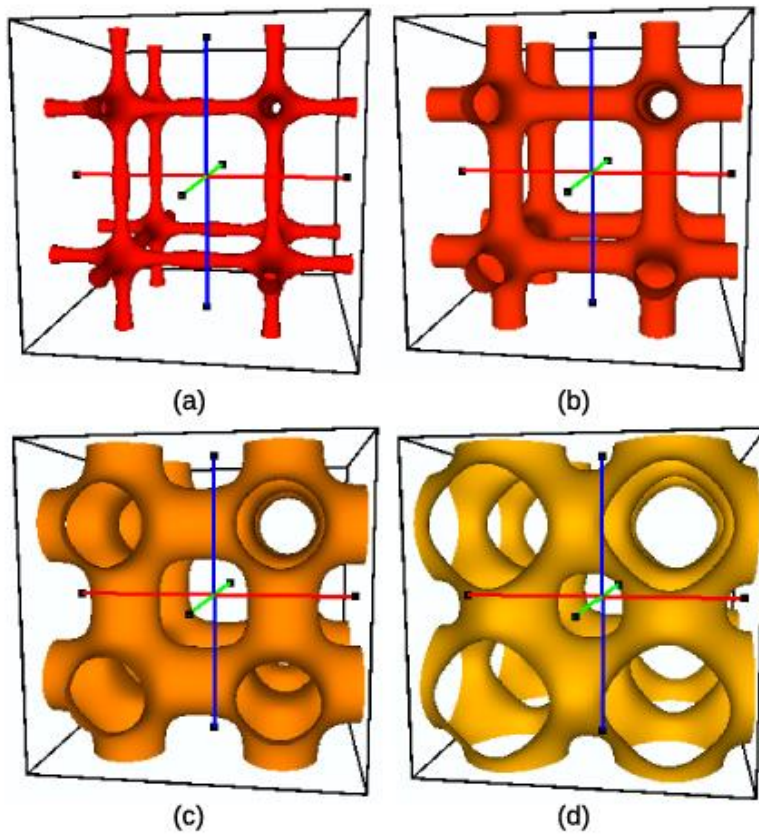


Figure 5.11: Internal pore becomes larger with iso-value increased.

In addition to the shape configuration, we also did some research on the porosity of the generated porous structures. There are two crucial factors determining the porosity of ARBF surface. First is the iso-value, which is the threshold for separating the internal and external spaces. Secondly, the unified shape parameter  $c$  of all radial basis functions listed in Table 5.1 also could change the surface appearance greatly. Figure 5.12 reveals that the porosity increases as one of the mentioned factors are augmented.

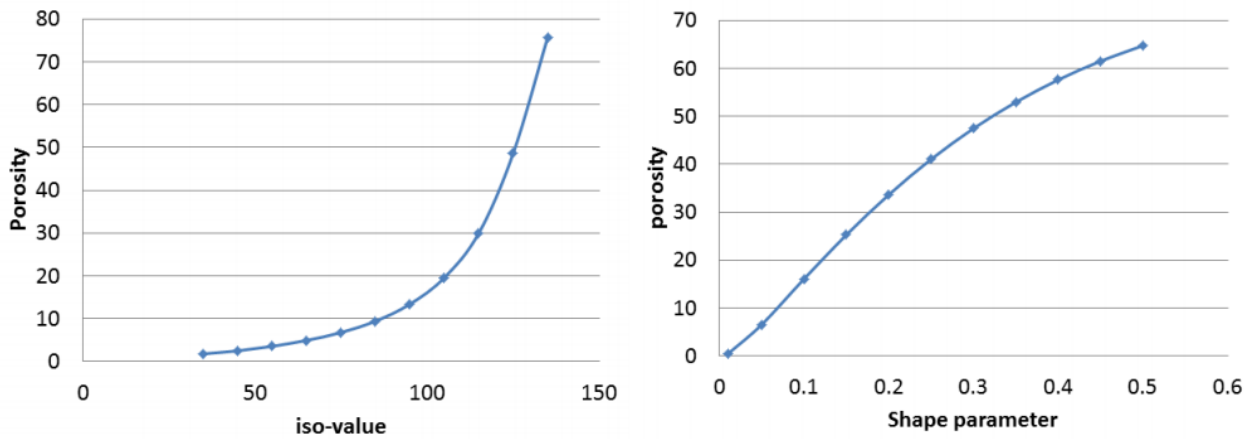


Figure 5.12: Relationships between porosity and different factors.

## 5.5 Summary

In this chapter, we presented an ARBF-based implicit surface modeling approach to generating inhomogeneous porous tissue scaffolds. The discussed method has several advantages that can be mainly described in four aspects:

- 1) Firstly, for most of periodic surface methods, a small modification to the base shape would greatly change the appearance of the entire porous architecture. Nevertheless, in the ARBF method, the local modifications of pore shape, size or pore distribution can be easily achieved by accordingly adjusting the local

configuration of sub-volumes.

- 2) Secondly, the ARBF approach is the most flexible one which can handle different types of mesh as inputs (tetrahedral mesh or hexahedral mesh).
- 3) Thirdly, unlike other implicit methods, the ARBF algorithm doesn't require any post-action (e.g. intersection) to produce the final product;
- 4) Compared with the aforementioned parameterization method and the TIS method, the proposed ARBF method not only can construct porous surface based on tetrahedral or hexahedral mesh surface, but also could generate connected hollow surfaces using designated point clouds.

Although our ARBF porous design method has such many advantages, the performance of the experimental result is still sensitive to the quality of the input meshes. For some complicated shapes, a poor tetrahedral/hexahedral input mesh could seriously affect the quality of the generated porous surface.

# Chapter 6

## **Segmentation of Intervertebral Discs and Application of Porous**

### **Design for Intervertebral Disc Scaffolds**

In previous chapters, we have sufficiently discussed the porous design of artificial tissue scaffold using several innovative methodologies. Both the tetrahedral implicit surface (TIS) method and the parameterization method based on triply periodic minimal surface (TPMS) and hexahedral mesh use volumetric mesh as the input to generate unit-based repetitive implicit porous surfaces. However, the anisotropic radial basis function (ARBF) method utilizes a more flexible approach based on points and line segments to construct connected hollow paths. Various porous structure design methods give the fabricated tissue scaffolds more reliable mechanical and transport properties and increase their values for clinic purposes.

One of the main purposes of this study is to apply these porous designs to the extremely high incidence of intervertebral disc (IVD) scaffolds. In clinical practice, the identification of intervertebral discs is usually manually labeled out by experienced experts from scanned medical images that are collected by traditional medical imaging techniques such as computed tomography (CT) and magnetic resonance imaging (MRI). However, this process is extremely tedious and time consuming. Moreover, most of the labeled out intervertebral disc models are somewhat unsmooth and bumpy such that they can't be used for 3D printing directly. The fully-automatic localization

and segmentation of intervertebral discs have been analyzed in the field for years. In this section, we are discussing a few prevalent techniques for the segmentation of intervertebral discs by using machine learning (ML) algorithms. In addition, a few generated porous IVD models will be presented in the Section 6.3.

## **6.1 Overview of Approaches for Segmentation of Intervertebral Disc**

Despite tons of studies have been focused on brain, heart and bone images, relatively less researches can be found for spinal image with regard to vertebra or intervertebral disc localization and segmentation. Research interests in automatic segmentation of intervertebral discs mainly can be classified into two groups: traditional methods and machine learning based methods.

### **6.1.1 Traditional Methods**

Most of successful traditional semantic segmentation techniques developed in the past decade rely on hand-crafted features and shape information. Carballido-Gamio et al. proposed the normalized cut to segment vertebral bodies from MR images [90]. And Huang et al. improved this method by proposing an interactive algorithm and evaluated their method on 2D sagittal MR slices [91]. Chevretils et al. presented 3D segmentation of intervertebral scoliotic discs from three specific MR image acquisition protocols with average DICE coefficient of 0.73 [92]. They implemented the watershed method across the sagittal and coronal volume planes resulting in over-segmentation, which was corrected through pixel classification utilizing image textual features. A semi-automatic approach based on probabilistic atlas was deployed by Michopoulou et al. to segment degenerated lumbar IVDs from 2D MR images of the spine [93]. Their method requires an interactive selection of the leftmost and rightmost disc points. Ghosh et



al. presented a herniated disc diagnostic method that classifies pathology limited to 2D MR scans using a Bayesian classifier with a coupled active shape model and a gradient vector flow snake [94]. Seifert et al. proposed an automatic cervical intervertebral disc segmentation technique that identified discs using the Hough transform based on segmented vertebrae and refined segmentation using statistical shape-aware deformable models [95]. Neubert et al. segmented the IVDs and vertebral bodies from high-resolution spine MR images using a statistical shape model based method [96]. Also, this method requires an interactive placement of a set of initial rectangles among spinal curves. Glocker et al. presented a method based on regression forests and probabilistic graphic models. However, their method is likely to suffer from the narrow field-of-view (FOV) because the broad contextual information is not always available [97]. To overcome this limitation, Glocker et al. again proposed a randomized classification forest based approach which achieved reasonable localization and identification performances on pathological cases and those with limited FOV [98].

### **6.1.2 Machine Learning Methods**

Great successes of machine learning based methods have gained increasing interests in the domain of computer vision, especially for medical image analysis. At first, Zheng et al. proposed a marginal space learning (MSL) to localize the heart chamber in 3D CT data [99]. In the study of Kelm et al. [100], the MSL strategy has been used again to detect and segment IVDs based on Haar-like features. Then Chen et al. [101] and Wang et al. [102] introduced two regression-based methods for the segmentation of IVDs. More specifically, Chen et al. proposed a unified data-driven estimation framework to localize IVDs by predicting foreground and background probability of each pixel [101]. And Wang et al. designed a sparse kernel machine

based regression approach to address the segmentation of multiple anatomic structures in multiple anatomical planes from multiple imaging modalities [102]. More recently, some literatures have been focused on segmentation of vertebrae or IVDs from 3D volumetric data or 2D images using deep learning based methods. For example, Cai et al. used a 3D deformable hierarchical model from multi-modality images to extract multi-modality features [103]. And Chen et al. employed a deep learning method based on convolutional neural network (CNN) for vertebrae identification in spine images [104]. One year later, Chen et al. proposed a 3D fully convolutional network (FCN), which is a derivative framework of CNN, had achieved the state-of-the-art localization performance of IVDs [105]. Since then FCNs have become the back-bone of state of the art medical image segmentation systems and a lot of variant FCNs have been introduced to advance this stream. For example, Kamnitsas et al. presented a multi-scale 3D FCN, which includes the low resolution and the normal resolution input, with two convolutional pathways for brain lesion segmentation [106]. A DSMS-FCN method proposed by Guodong Zheng et al. archived a mean Dice overlap coefficient of 92% and a mean average symmetric surface distance of 0.41 mm in the MICCAI 2015 IVD localization and segmentation challenge [107]. Moreover, Li et al. introduced an advanced 3D multi-scale FCN with random modality voxel dropout learning for IVD localization and segmentation from multi-modality MR images [108]. The multi-scale FCN method expands the typical single-path FCN to three pathways where each pathway takes volumetric regions in a different scale. And the features extracted from the three pathways are then concatenated to generate a probability map, from which the final 3D segmentation mask is generated with a single threshold.

## 6.2 Segmentation of Intervertebral Disc Using U-Net

As one of the most famous fully convolutional networks (FCNs), U-Net has been published in 2015 international conference on medical image computing & computer assisted intervention (MICCAI) with more than 3000 citations. U-Net and its variants have outperformed the state-of-art in many biomedical image segmentation tasks, including brain tumor detection [109], liver image segmentation [110], pixel-wise regression [111], etc. The pertinacious architecture and affluent data augmentation allow U-Net to quickly converge to the optimal model from a limited number of annotated samples. Comparing to the other two popular convolutional neural networks, CNN and FCN, U-Net uses skip connections between contraction and expansion and a concatenation operator instead of a sum, which could provide more local information to global information while expansion. Moreover, U-Net is symmetric such that a large number of feature maps in expansive path facilitate to transfer more information. In fact, U-Net has been widely applied to the IVD segmentation problem. Chen et al. [112] applied a conventional 3D U-Net [113] on the IVD dataset provided by the 3rd MICCAI Challenge of Intervertebral Discs Localization and Segmentation. And Kim et al. designed a new network architecture, boundary specific U-Net (BSU), to segment IVDs and their complex boundaries from MR spine images [114]. Their architecture contained a modified convolutional and pooling layer scheme and applied a cascaded learning method to overcome these structural limitations of the max-pooling layer of a conventional U-Net. Dolz et al. extended the U-Net by adding three identical pathways in the contracting path to process the multi-modality channels of the input [115]. These pathways were interconnected with hyper-dense connections to better model relationships between different modalities in the multi-modal input images. More recently, Lu et al. proposed an IVD segmentation pipeline which first localize the IVD centers using the 2D U-

Net and then cropped transverse 2D images and sagittal 3D patches around the centers to train a ResNeXt-50 [116] which mixed both 2D and 3D convolutions [117]. However, the effectiveness of data augmentation and multi-modality input images are not fully explored in these works.

The performance of 2D neural networks versus 3D neural networks has remained as an open discussion. Some researchers believe the accuracy of 2D deep learning exceeds that of 3D mainly because the manual segmentations of the original IVD dataset is performed in a slice-by-slice manner [115]. These annotation labels usually contain sharp contours, which makes the segmentation task difficult for 3D CNNs that generally predict smooth contours. However, only considering the pixels in each image from the stack will lose the spatial information between different slices. To clarify this consideration, next we will tackle the IVD segmentation problem from the perspectives of both 2D and 3D. For the 2D framework, we train a modified 2D U-Net that takes images slices from the volumetric dataset as input and predict the IVD segmentation mask in 2D space. For the 3D framework, we apply 3D convolutional kernels directly on the volume and generate a volumetric mask using a modified multimodal 3D U-Net. Additionally, powerful augmentation techniques, including homogeneous and non-homogeneous deformations, are also utilized to increase the number of training samples. In the following sections we will introduce the multi-modality dataset and the augmentation methods. Also, the architectures of the 2D network and 3D U-Nets will be demonstrated in detail.

### **6.2.1 Multimodal Dataset**

Due to the excellent soft tissue contrast and no ionizing radiation, magnetic resonance (MR) imaging (MRI) has been widely applied in the diagnose and assessment of lumbar IVD

diseases. Multi-modality MR images can be obtained using different scanning configurations for the same object. The spine dataset was provided by Prof. Guoyan Zheng from University of Bern [118]. It consists of 8 sets of 3D multi-modality MR spine images collected from 8 patients in 2 different stages of prolonged bed test. Each subject was scanned with 1.5 Tesla MR scanner of Siemens Magnetom Sonata (Siemens Health-care, Erlangen, Germany). The repetition time of 5240 ms and echo time of 101 ms were used to generate T2-weighted sagittal images. The resolution of all images was  $2\text{ mm} \times 1.25\text{ mm} \times 1.25\text{ mm}$ . Each spine image contains at least 7 IVDs of the lower spine (T1-L5) and four modalities following Dixon protocol: in-phase (inn), opposed-phase (opp), fat and water (wat) images. In total, there are 32 3D single-modality volumes and 66 IVDs. For each IVD, the segmentation ground truth is a binary mask manually labeled by three trained raters under the guidance of clinicians.

The traditional multi-modality deep learning methods conventionally fed all modalities images into different channels as input to the neural network. Zhang et. al. use multi-modality images to segment infant brains. Li et. al. trained the 3D FCN separately on each single modality (fat, in-phase, opposed-phase, and water) and then on a merged full-modality of the spine dataset to validate the superiority of training with multi-modality data. Both works show that training on the full-modality images could yield a more accurate IVD segmentation result than using any of the single-modality [108]. Moreover, Li's research indicates that the image modalities of opposed-phase and water could enhance the performance than the image modalities of fat and in-phase. Nonetheless, the full-modality-fused images will not guarantee the best outcomes in all experiments. The dependency between different modalities could lead to data co-adaptation, which means the same features are detected repeatedly. In addition, the significant difference between

modalities at the same locations could trigger data corruption problem to misdirect the corresponding neurons.

To address these concerns, we made a further exploration on the intensity distribution of various modalities and the mean gradient on the boundary of the intervertebral discs. As shown in Figure 6.1, the image modality of fat has relatively low intensity contrast at the IVD boundaries while compared with the other three image modalities. More specifically, assume the target IVDs are foreground and the other areas are background and Table 6.1 lists the average intensities and the standard deviations for the foreground and background samples using the provided label maps as a reference. Moreover, it also shows the absolute Weber contrast coefficients of in-phase, opposed-phase, fat and water images. The absolute Weber contrast coefficient is defined by [119]

$$C = \frac{|I - I_b|}{I_b} \quad (6.2.1)$$

where  $I$  is the mean intensity value of the foreground voxels and  $I_b$  is the mean intensity value of the voxels in the background. The Weber contrast indicates the relationship between the luminance or color of an interest area and that of adjacent areas. The significant contrast yields a big value of absolute Weber contrast. The results from Table 6.1 show that the image modalities of fat and in-phase have relatively low Weber contrast values than the image modalities of water and opposed-phase. To sufficiently take advantage of the multiple modalities of the dataset, we train our networks on different combinations of multi-modality images to examine their effectiveness in segmentation accuracy.

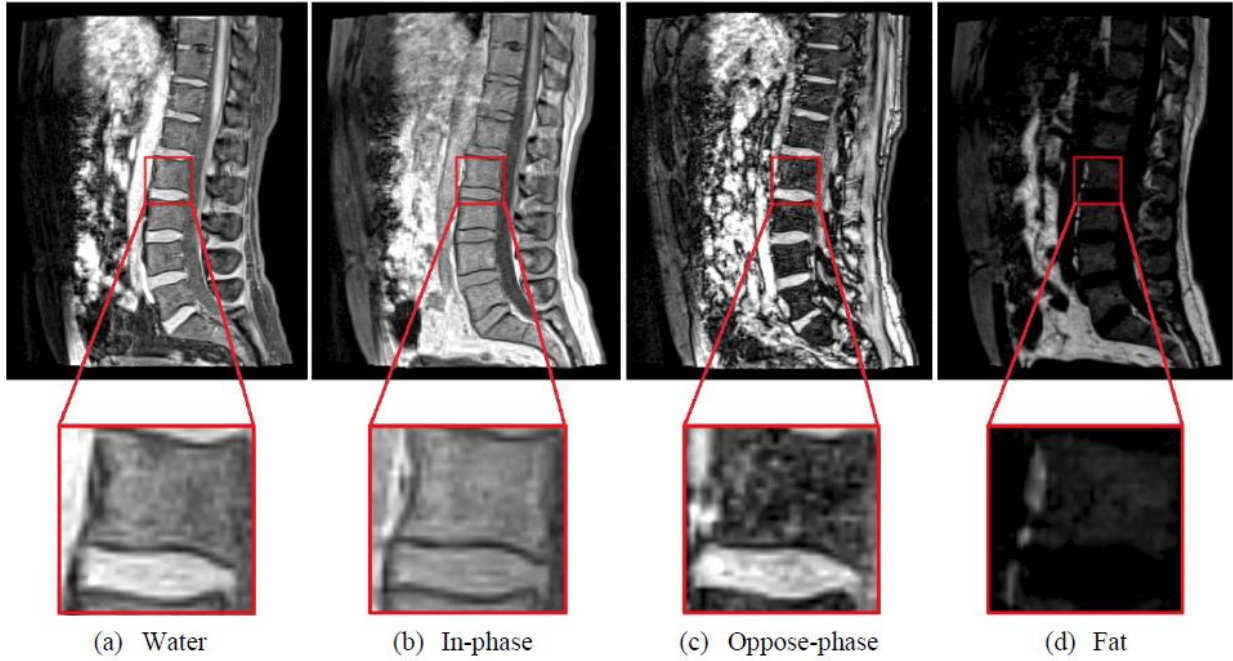


Figure 6.1. MR spinal images collected from different modalities.

Modality	Mean $\pm$ STD (foreground)	Mean $\pm$ STD (background)	Absolute Webber Contrast
Fat	15.9 $\pm$ 12.7	35.2 $\pm$ 47.5	0.57
In-phase	172.1 $\pm$ 39.9	97.9 $\pm$ 89.5	0.75
Opposed- phase	155.4 $\pm$ 47.5	63.2 $\pm$ 65.1	1.46
Water	163.4 $\pm$ 41.1	67.9 $\pm$ 67.5	1.43

Table 6.1. Statistical analysis to different image modalities.

## 6.2.2 Data Augmentation

When limited training samples are available, data augmentation is essential to increase the size of the desired dataset and increase the network's robustness to data variances. Also, data augmentation is a regularization technique which is used to eliminate overfitting when training machine learning models. Existing image augmentation methods are generally grouped into two categories: traditional (white-box) methods and black-box methods [120]. The most popular practice for data augmentation is to perform a combination of affine transformations and color modification to create new samples. The common affine transformation techniques include translation, rotation, reflection, scaling (zoom in/out), shearing and etc. Affine transformation is a linear mapping method that preserves points, lines and planes and parallel lines remain parallel after an affine transformation. The affine transformations have been proved as fast, reproducible and reliable and the according codes are easy to implement. Additionally, some color modification approaches are also frequently used combined with affine transformations. The most popular techniques include histogram equalization, enhancing contrast or brightness, white-balancing, sharpening and blurring. On the other side, black-box augmentation methods are all adaptive and distinct. For example, as a new and powerful tool to perform unsupervised generation of new images using min-max strategy [121], generative adversarial networks (GANs) [122] are found to be extremely useful in the image manipulation and generation problems such as super-resolution [123], image blending [124], and text-to-image synthesis [125], etc. Moreover, texture transfer [126] is another prevalent tool which synthesize a texture-source image while constraining the semantic content of a content-source image. Additionally, Gaussian noise and stochastic noise [127] are also black-box tools that reduce the risk of overfitting and makes the model more robust.



With only 8 sets of spine data provided, our networks demand more data for training and validation. Especially for the 2D U-net, augmenting the training samples could significantly enhance the performance. A variety of conventional 3D image processing techniques are adopted in our method. In addition to these affine transformations, as a particularly important augmentation technique in biomedical segmentation tasks, elastic deformation is also used in combination with other transform functions due to the fact that the most common variation in tissue is deformation [118]. We fabricate smoothly deformed surfaces by generating random displacement fields and the fields are convolved with a Gaussian of standard deviation  $\delta$  (in pixels). And the displacement fields are then multiplied by a scaling factor  $\alpha$  that controls the intensity of the deformation [118]. The per-pixel displacements are computed using bicubic interpolation. Based on the augmentation functions mentioned above, a random combination of operations is selected in an arbitrary order and applied to the original spine dataset. Consequently, the size of the training dataset is increased from 6 to 24 (2 of the 8 sets of original spine data are not used for augmentation but for validation of the prediction). To sufficiently take advantage of the information between adjacent layers, besides using the augmented spine data in 3D framework, we also use the stack of images sliced along each axis of the 3D augmented data as the training set of the 2D network rather than directly applying 2D augmentation techniques on the sliced images from the original dataset.

### **6.2.3 2D U-Net**

The architecture of our 2D U-Net is shown in Figure 6.2. It consists of a contraction path and an expansion path. The contracting path is composed of 4 convolutional blocks. Each block contains the repeated application of two unpadded  $3 \times 3$  convolutions that are followed by a

rectified linear unit (ReLU). Consequently, the number of feature channels is doubled for each down-sampling step and the number is increased from 1 to 1024 through the whole four blocks. Moreover, for each down-sampling step, a  $2 \times 2$  max pooling with stride 2 is applied to the end of every blocks except the last block, which makes the size of feature maps decreases from  $512 \times 512$  to  $16 \times 16$ . In the expansive path, we operate the opposite that every block starts with a deconvolutional layer with the filter size of  $3 \times 3$  and the stride of 2 and follows with a ReLU layer, which doubles the size of feature maps in each block but halves the number of feature channels such that the size of feature maps increases from  $16 \times 16$  to  $512 \times 512$  and the number of channels decreases to 64. In the end, a  $1 \times 1$  convolution with the sigmoid function as the post-operation is used to restore the original size of the input images. The last but the most important is that a concatenation with correspondingly cropped feature map from the contacting path is placed after each up-convolutional layer.

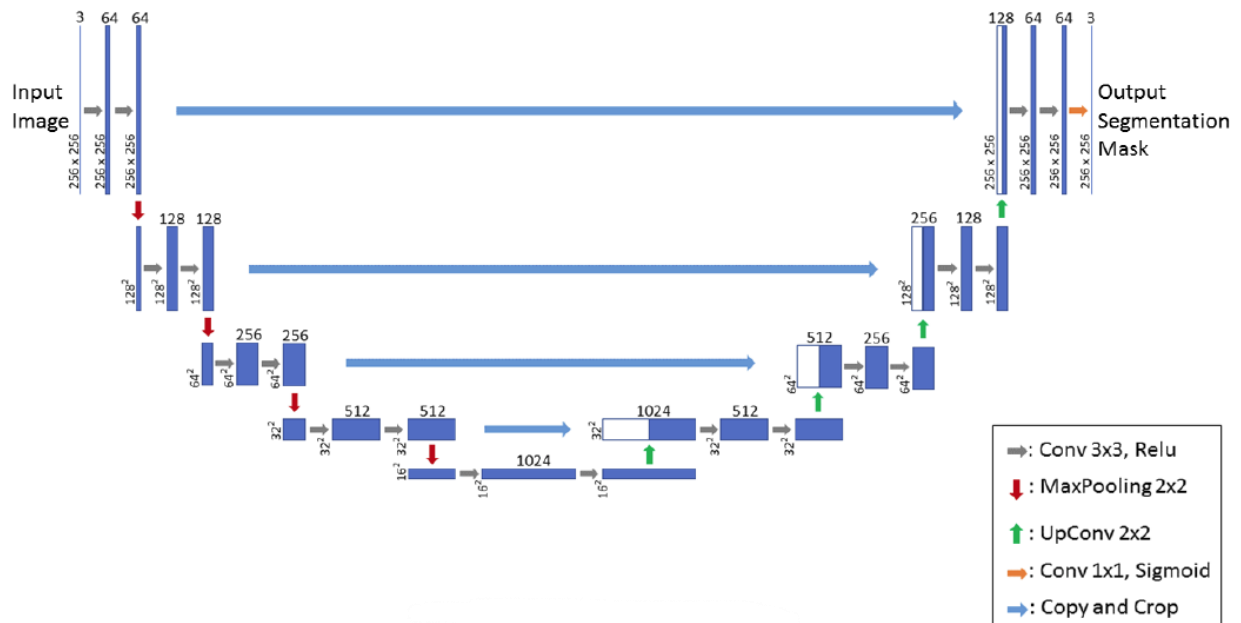


Figure 6.2. 2D U-Net architecture.

The soft tube models of the lumbar spine column are all 3D volumetric data such that partitioning the same 3D model from different angles could create different 2D image datasets. To analyze the efficiency of trainings along different directions, we train the 2D U-Net on image slices extracted from the augmented spine dataset along varying directions (X-axis, Y-axis and Z-axis) to predict segmentation masks. The stack of images sliced along X-axis are of size  $256 \times 256$  pixels while the images retrieved from the other two axes are both size of  $256 \times 36$  pixels. To ensure the trained network can handle images of different sizes, we preprocess all images to the same size of  $256 \times 256$  pixels using zero-padding. It's noted that the number of images retrieved from a single 3D sample along Y-axis or Z-axis is 256 while the number of images sliced from Y-axis direction decreases to 36, which is roughly seventh of 256. However, the number of samples will be increased to the same for all directions after the data augmentation. There will be 1200 images generated in total, among which 1000 of them consist of the training set and the rest make up the validation set.

#### **6.2.4 3D U-Net**

Since the provided spinal dataset is in NIfTI (Neuroimaging Informatics Technology Initiative) format [128], which is a volumetric image, and the volumetric contexture information between different layers could facilitate to enhance the performance of the IVD prediction, we decided to try to apply U-Net directly on the 3D spinal images rather than the sliced 2D images. Moreover, we also noticed that because the colors of some of the target IVDs and the background are too similar such that unexpected segmented IVDs could be detected at some positions in the background. Here we adopt a two-stage coarse-to-fine strategy to tackle the problem using 3D U-Net. The general framework is demonstrated in Figure 6.3. In the first

stage, the centers of IVDs will be localized based on a conventional 3D U-Net. Next, the small patches of IVD samples, whose centers are the ones detected from the first step, are cropped out with a fix-size window. In the second stage, those cropped samples are used to train a multimodal model for accurate segmentation of each IVD. This could simplify the problem by removing most deceptive information from the background. In the end, a post-operation will be placed to finalize the segmentation results.

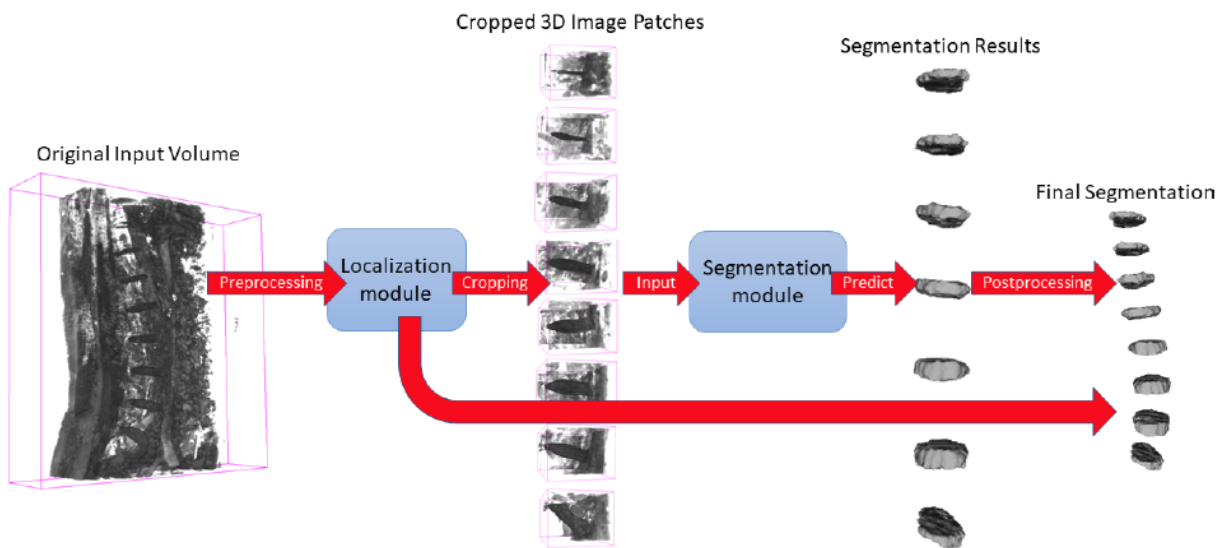


Figure 6.3. Framework of the proposed 3D neural network method.

#### 6.2.4.1 Localization of IVD Centers

It has been shown that 3D U-Net achieves the best result for localization but not for segmentation [129]. Consequently, a conventional 3D U-Net is deployed to find out the centers of the intervertebral discs from the spine column in our method. From the observation, the spines are sharing a common morphological profile that the IVDs are sparsely located in the spine with a roughly distance of 25 voxels. Thus, we use a fix-size bounding box, whose resolution is  $35 \times 35 \times 25$  voxels, around each detected center to crop a 3D patch. Then we resize these patches

to resolution of  $36 \times 36 \times 28$  using zero-padding so they can be nicely fed into the segmentation network in the second stage.

#### 6.2.4.2 Segmentation of IVDs

For IVD segmentation from the 3D patches, we employ a modified 3D U-Net architecture shown in Figure 6.4 that essentially looks at IVD segmentation as a regression problem. This network takes 3D patches as input and predicts 3D patches where the intensity value on each voxel stands for how confident is the network in the voxel belonging to an IVD. Figure 6.4 presents an overview of the architecture of our 3D segmentation network. Each step in the contracting path consists of repeated application of two  $3 \times 3 \times 3$  unpadded 3D convolutions followed by a ReLU. A dropout operation is inserted between the two convolutions to reduce the dependence on the training dataset. Experiments show that, compared with a network trained without dropout strategy, the network with dropout strategy can generate more discriminative features and achieve more accurate segmentation results. A dropout rate of 0.2 is used following the analysis on the dropout effect in CNN [130]. We also apply batch normalization to speed up and stabilize the training process and a  $2 \times 2 \times 2$  max pooling layer with stride 2 for down-sampling after every two convolutional layers. At each down-sampling step we double the number of feature channels. Every step in the expansive path consists of an up-sampling of the feature map followed by a  $2 \times 2 \times 2$  up convolution that halves the number of feature channels, a concatenation with the corresponding feature map from the contracting path, and two  $3 \times 3 \times 3$  convolutions, each followed by a ReLU. The output layer is a  $1 \times 1 \times 1$  convolution layer with

sigmoid activation used to generate the segmentation mask for each modality. In total the network has 12 convolutional layers and 1.4 million parameters.

	Layer name	Input shape	Output shape
	Input	$36 \times 36 \times 28 \times 3$	$36 \times 36 \times 28 \times 3$
	Conv3D	$36 \times 36 \times 28 \times 3$	$36 \times 36 \times 28 \times 32$
	Dropout	$36 \times 36 \times 28 \times 32$	$36 \times 36 \times 28 \times 32$
	Conv3D	$36 \times 36 \times 28 \times 32$	$36 \times 36 \times 28 \times 32$
	Max Pooling	$36 \times 36 \times 28 \times 32$	$18 \times 18 \times 14 \times 64$
Contracting Path	Conv3D	$18 \times 18 \times 14 \times 64$	$18 \times 18 \times 14 \times 64$
	Dropout	$18 \times 18 \times 14 \times 64$	$18 \times 18 \times 14 \times 64$
	Conv3D	$18 \times 18 \times 14 \times 64$	$18 \times 18 \times 14 \times 64$
	Max Pooling	$18 \times 18 \times 14 \times 64$	$9 \times 9 \times 7 \times 128$
	Conv3D	$9 \times 9 \times 7 \times 128$	$9 \times 9 \times 7 \times 128$
	Dropout	$9 \times 9 \times 7 \times 128$	$9 \times 9 \times 7 \times 128$
	Conv3D	$9 \times 9 \times 7 \times 128$	$9 \times 9 \times 7 \times 128$
	Up Sampling	$9 \times 9 \times 7 \times 128$	$18 \times 18 \times 14 \times 64$
	Concatenate	$18 \times 18 \times 14 \times 64$	$18 \times 18 \times 14 \times 64$
	Conv3D	$18 \times 18 \times 14 \times 64$	$18 \times 18 \times 14 \times 64$
	Dropout	$18 \times 18 \times 14 \times 64$	$18 \times 18 \times 14 \times 64$
	Conv3D	$18 \times 18 \times 14 \times 64$	$18 \times 18 \times 14 \times 64$
Expensive Path	Up Sampling	$18 \times 18 \times 14 \times 64$	$36 \times 36 \times 28 \times 32$
	Concatenate	$36 \times 36 \times 28 \times 32$	$36 \times 36 \times 28 \times 32$
	Conv3D	$36 \times 36 \times 28 \times 32$	$36 \times 36 \times 28 \times 32$
	Dropout	$36 \times 36 \times 28 \times 32$	$36 \times 36 \times 28 \times 32$
	Conv3D	$36 \times 36 \times 28 \times 32$	$36 \times 36 \times 28 \times 32$
	Conv3D	$36 \times 36 \times 28 \times 32$	$36 \times 36 \times 28 \times 2$
	Batch Norm	$36 \times 36 \times 28 \times 2$	$36 \times 36 \times 28 \times 2$
	Output	$36 \times 36 \times 28 \times 2$	$36 \times 36 \times 28 \times 3$

Table 6.2. 3D segmentation network architecture.

### 6.2.4.3 Post-operation

The prediction results of the IVD segmentation usually contain continuous intensity values rather than a static value set (0 and 255). However, those continuous values cannot represent either an IVD voxel or the background. As a result, the final segmentation mask for each IVD would be obtained through a binary thresholding with a threshold of 0.5. Then the small IVD models are assembled back to the 3D volumes of the lower spine based on the location information of the IVD centers obtained from 6.2.4.1.

### 6.2.5 Training

The input images and their corresponding segmentation maps are used to train the deep learning networks. Both the 2D and the 3D networks in the present work are implemented in python with Keras [131] and Tensorflow [132] as the backend. To minimize the overhead and make maximum use of the GPU memory, the batch size is set to 1 for the 3D network and 16 for the 2D network. The models are trained on a PC with an 8-core 3.4GHz CPU and a single NVIDIA GTX Titan Xp GPU. The training time of a single epoch takes about 3 seconds in the 3D model and 45 seconds in the 2D model. Eventually, the validation loss will stop increasing at around 10000 epochs for the 3D model and 500 epochs for the 2D model before overfitting. In deep networks with many convolutional layers and different paths through the network, a good initialization of the weights is extremely important. Otherwise, parts of the network might give excessive activations, while other parts never contribute. The convolutional kernels of our networks are initialized with the He initialization [133] in order to speed up the training process.

For updating the parameters in the networks, we employ the Adam optimization algorithm [135], which has been popularized in the field of stochastic optimization due to its fast convergence compared to other optimization functions. The learning rate is set to 1e-5 for accurate predictions and reasonable training time. The energy function is computed by the Dice-coefficient loss function defined as:

$$E = \frac{2 \times I + S}{\sum T + \sum P + S} \quad (6.2.2)$$

where  $T$  stands for the ground truth label and  $P$  for the prediction.  $I$  is the number of true positives calculated by the number of intersection pixels between  $T$  and  $P$ .  $S$  is a smoothing factor with a value of 1.

### 6.2.6 Evaluation Metrics

To evaluate the segmentation performance, two metrics from the 2015 MICCAI Challenge are adopted:

- Dice overlap coefficient (DOC)

DOC reveals the similarity between the segmentation result and the ground truth annotation. More specifically, DOC measures the rate of correctly segmented voxels by

$$DOC = \frac{2 \times |X \cap Y|}{|X| + |Y|} \quad (6.2.3)$$

where  $X$  denotes the set of voxels in the ground truth segmentation mask and  $Y$  denotes the set of voxels predicted by the neural network. A larger DOC score indicates more accurate segmentation result.



- Hausdorff distance (HD)

HD measures how far two metric spaces are from each other. Usually, it can be used to judge the similarity between two surface meshes. We compute HD for the surfaces reconstructed from the ground true segmentation mask and the segmentation result. The surfaces are generated using Iso2Mesh [136] from binary segmentation masks. The HD is computed as the closest distance from each vertex on the source surface mesh to the target surface mesh. In contrast to DOC, a smaller HD indicates better segmentation performance.

### 6.2.7 Experimental Results

From Section 6.2.1, we concluded that the fat and in-phase image modalities have lower absolute Weber contrast than the opposed-phase and water image modalities. In order to analyze the influence of the image modalities to the performance of IVD segmentation, we mean to train our 3D neural network on 4 different configurations of modality combination:

- 1) Full-modality (4 modalities)
- 2) Water, opposed-phase, and fat modalities (in-phase modality excluded)
- 3) Water, opposed-phase, and in-phase modalities (fat modality excluded)
- 4) Opposed-phase and water modalities (in-phase and fat modalities excluded)

The generated performance for different modality combination is listed in Table 6.2 as below.

The experimental results reveal that the network trained on the full-modality dataset has the lowest DOC score, which indicates the worst performance in the segmentation task. On the other

hand, the 3D image dataset which consists of the water, opposed-phase, and fat modalities produces the best outcome. Moreover, from the reconstructed 3D models with different modality configuration (Figure 6.5), the combination of opposed-phase, water, and fat modalities obviously can get rid of a few extra bad estimated pieces when compared with the ground truth. Based on what we have found, the rest of section will focus on discussing the results generated using the second combination of modalities that are water, opposed-phase, and fat modalities.

Modality Combination	DOC $\pm$ STD
opp, wat, fat and inn	87.9 $\pm$ 1.7
opp, wat and fat	89.0 $\pm$ 1.4
opp, wat and inn	88.0 $\pm$ 1.6
opp and wat	88.5 $\pm$ 1.6

Table 6.3. Performance analysis of 3D segmentation using different combinations of image modalities.

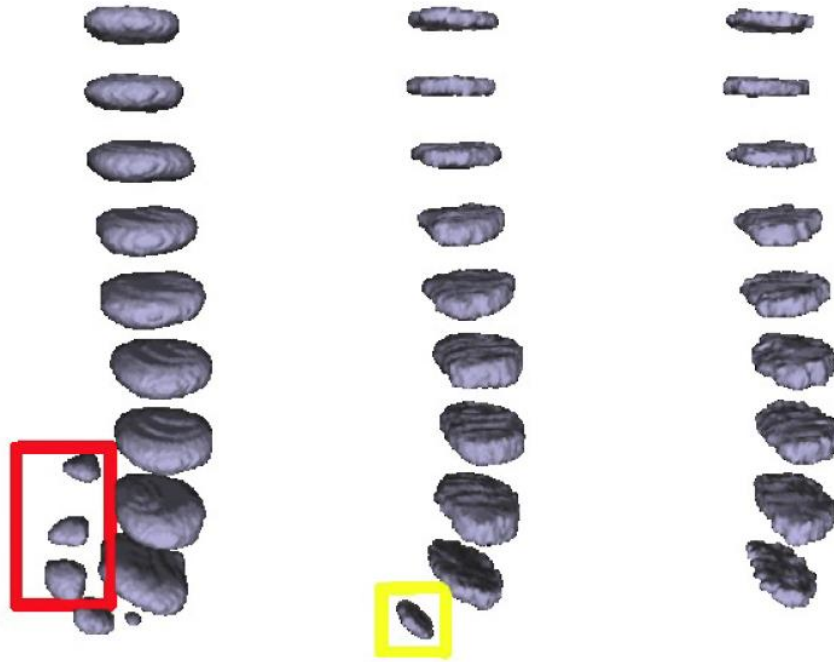


Figure 6.4. Training 3D U-Net based on: 1) full-modality (left) 2) opp, wat, fat modalities (middle) 3) ground truth (right).

For the 2D-Unet, using the sliced images of the spine volume from three different directions could yield various accuracies. Table 6.3 lists the corresponding prediction performances based on the datasets obtained by partitioning the augmented spine columns from X-axis, Y-axis, and Z-axis, respectively. Additionally, we overlap the models generated from the three directions and take half of the maximum value, which is 127.5, as the threshold to distinguish the IVDs and the background. From Table 6.3, it's noted that the results using the dataset sliced from Y-axis have a higher mean accuracy than the results using the data from the other two directions. Also, the standard deviation (STD) of the intensity on X-axis is the smallest. Furthermore, using the average intensity from all directions will not improve the outcomes.

Direction	Mean DOC $\pm$ STD
X-axis	79.4 $\pm$ 1.0
Y-axis	81.8 $\pm$ 1.3
Z-axis	77.3 $\pm$ 6.7
Merged	80.0 $\pm$ 1.5

Table 6.4. Segmentation performance of 2D U-Net trained with images sliced from different directions (X-, Y-, Z-axis).

Combining Table 6.2 and Table 6.3, we observe that the 3D network achieves a much better result compared with the 2D version. Figure 6.6 gives a more comprehensive visualization of the comparison between the results generated by the 2D network and the 3D network. It also has verified that the conjunction relationship information between adjacent slices can contribute to enhance the efficiency of neural network and improve the accuracy of the prediction result.

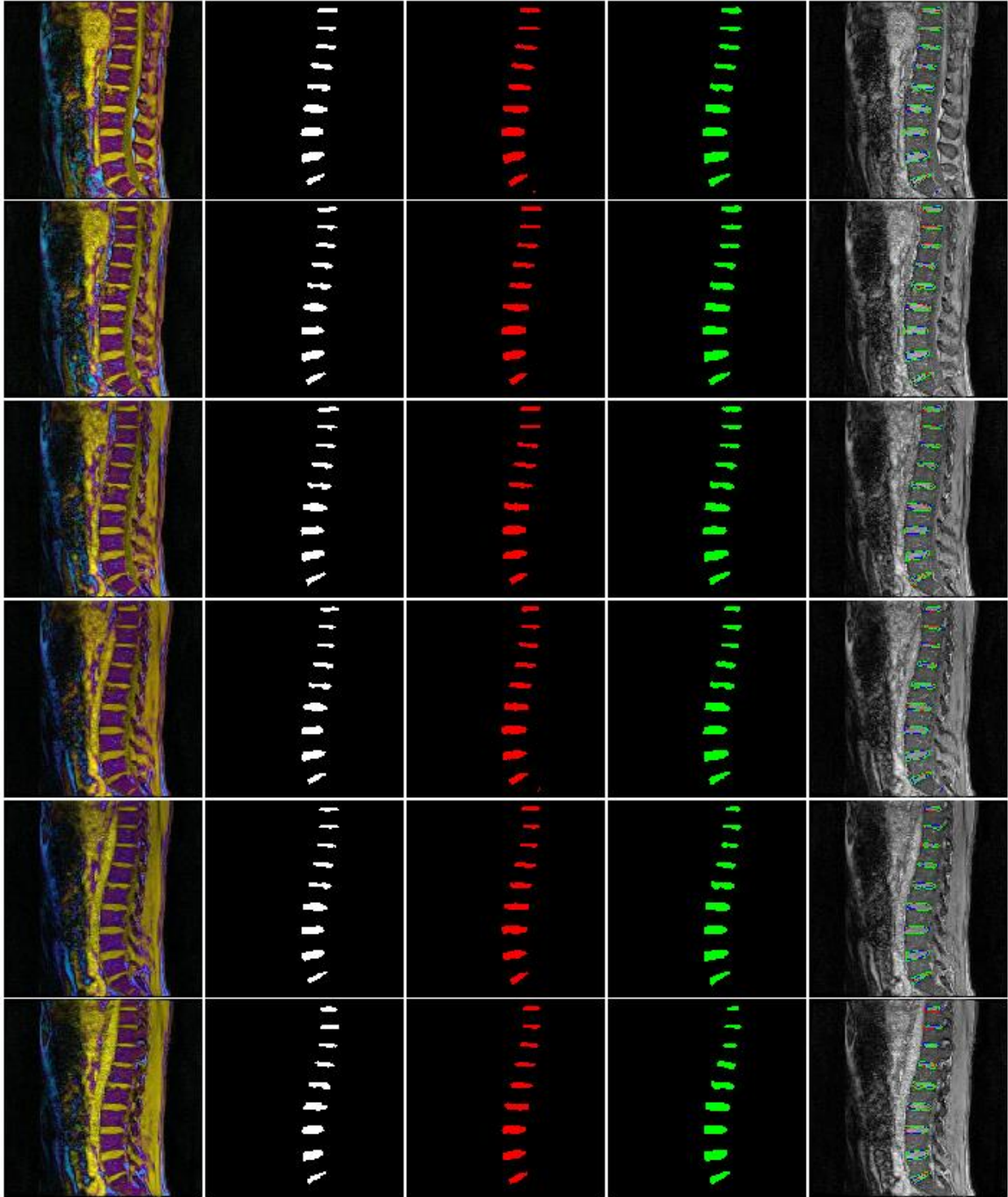


Figure 6.5. A comparison between the 2D and 3D methods. The images from leftmost column to rightmost column are respectively source images, ground truths, 2D prediction results, 3D prediction results, and merged contours of all results.

Moreover, to evaluate the performance of the proposed methods, we also compare the segmentation results achieved by our methods with those by 3D U-Net [113], the CNN-based team UNICHK [112] and the winning team UNIJLU [136] of the 2015 MICCAI Challenge. Quantitative results evaluated with the different methods are presented in Table 6.4. The mean DOC score obtained by our 3D method is 89.0% with a standard deviation (STD) of 1.4%. We bring a 1.5% boost comparing to the conventional 3D U-Net by training our network on 3D image patches using image modalities of opposed-phase, water and fat. But it is still 2.5% behind the state-of-the-art performance achieved by team UNIJLU. The mean HD of our 3D method achieves 0.8 mm with a standard deviation of 0.3 mm, which indicates that our method is slightly better when the segmentation results are reconstructed to 3D models. The strength of deep learning methods is the computation time. The Theano-based implementation of 3D U-Net from UNICHK takes 3.1 s to process one  $40 \times 512 \times 512$  volume. Our network is implemented based on Tensorflow and it takes about 0.5 s to segment all the IVDs in a  $36 \times 256 \times 256$  input volume. Overall, the computation time of our end-to-end segmentation is about 10 seconds, including localization, preprocessing, segmentation and postprocessing, whereas it takes 5 min on average to segment all IVDs for a patient by UNIJLU. It is also worth mentioning that the training dataset used in our study only contains data from 6 patients while UNICHK and UNIJLU have access to a training dataset from 16 patients i.e. our network is able to learn the 3D geometric morphometrics of IVDs with much less data.

To explore the effectiveness of data augmentation in training a neural network, we conducted experiments by training our 3D network with and without the proposed data augmentation method. The results do not suggest obvious differences in terms of DOC and HD. However, we found that data augmentation enables the network to learn more details in the

boundary area. More specifically, segmentation results are improved in the regions between an IVD and the adjacent vertebral body. Detailed examples are given in Figure 6.7, the regions well segmented by our 3D network trained with augmented dataset are marked by the red boxes. Segmentation on the regions with sharp boundaries is a difficult task for convolutional kernels since most of convolutional neural networks tend to predict smooth contour. The core of an IVD is composed of a jelly-shape material and the smooth elastic deformation technique from the data augmentation techniques mimics the deformation of IVDs from the real world and enriches the training dataset with more morphological features. This makes U-Net more capacity to deal with unseen IVD data and make a better prediction.

Method	Mean DOC $\pm$ STD	Mean HD $\pm$ STD
Conventional 3D U-Net	87.5 $\pm$ 0.9	1.1 $\pm$ 0.2
UNICLK	88.4 $\pm$ 3.7	1.3 $\pm$ 0.2
UNIJLU	91.5 $\pm$ 2.3	1.1 $\pm$ 0.2
Our 3D Network	89.0 $\pm$ 1.4	0.8 $\pm$ 0.3
Our 2D Network	81.8 $\pm$ 1.3	2.4 $\pm$ 1.0

Table 6.5. Segmentation result evaluation of the conventional 3D U-Net, UNICLK, UNIJLU, and our methods.

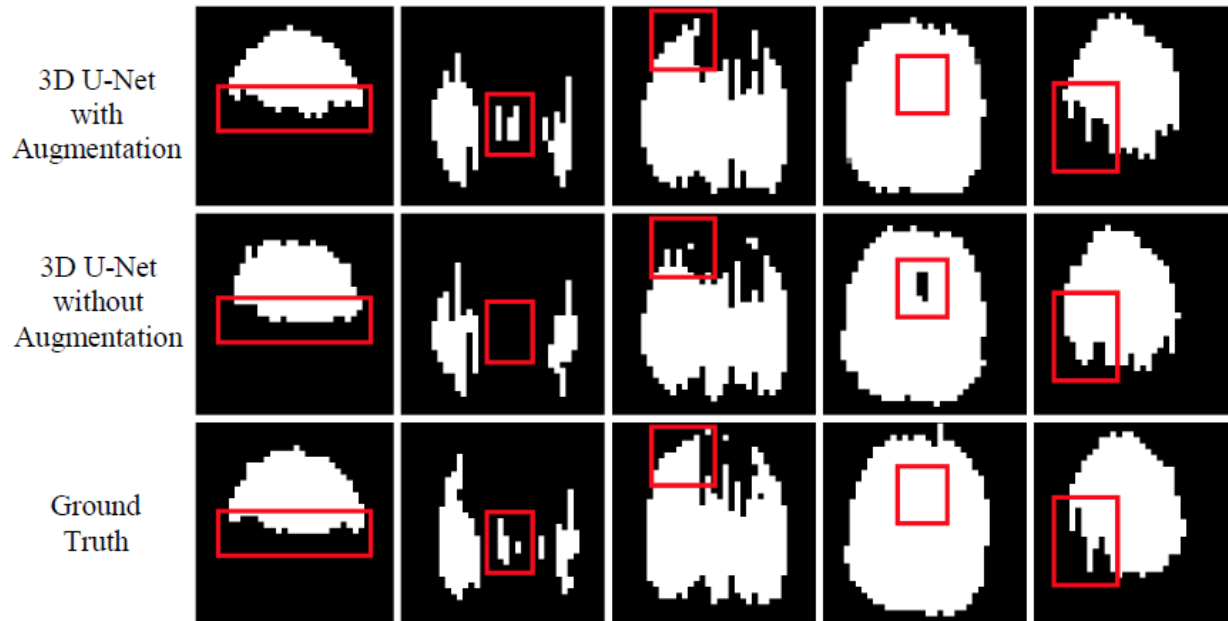


Figure 6.6. Effective analysis of data augmentation.

### 6.3 Porous Intervertebral Disc

In Section 6.2, we have successfully segmented the IVDs out from a spine column. Here we are trying to fabricate porous structures based on each of these segmented IVDs using different porous design methods. As an example, a complete set of IVDs retrieved from a spine, which is randomly selected from the given spine dataset (refer to Section 6.2.1), is shown in Figure 6.8. And the 7<sup>th</sup> and the 8<sup>th</sup> IVDs are displayed in a zoom-in view.



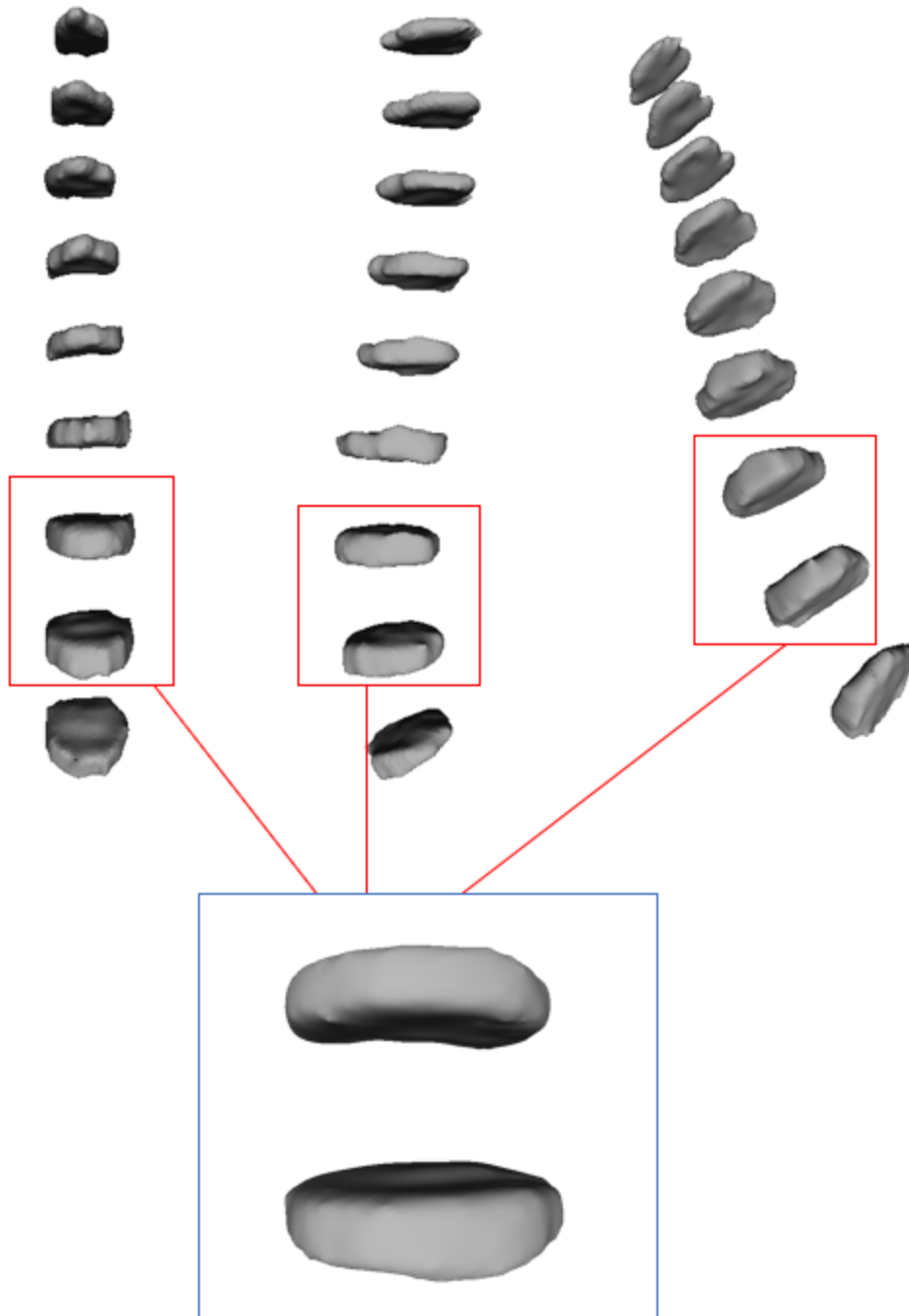
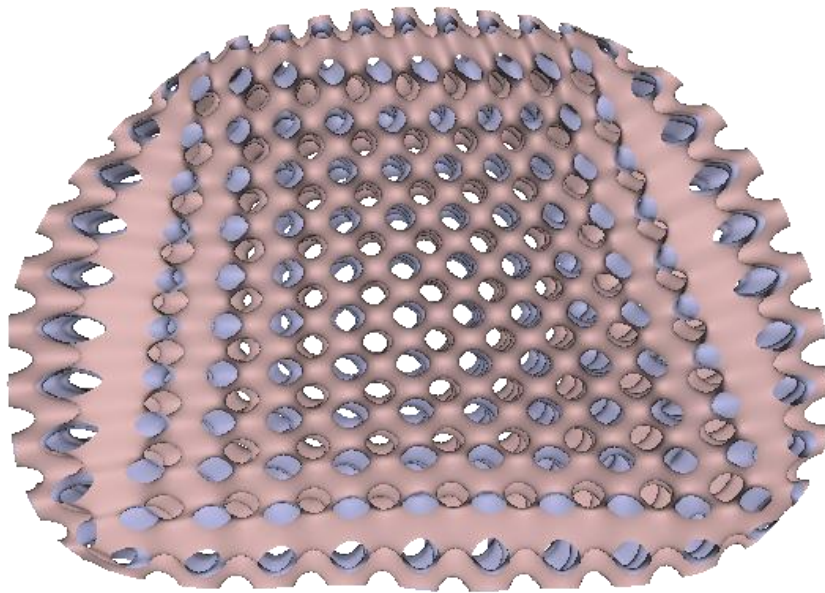


Figure 6.7. Segmented IVDs from spine column.

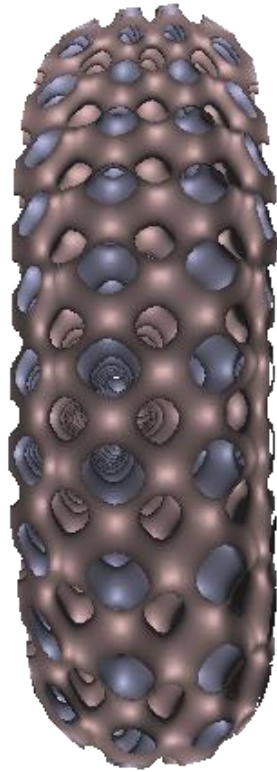
Now the three porous design approaches that are proposed in Section 3-5 respectively are independently applied to of the 7<sup>th</sup> IVD of the reconstructed 3D IVD set (Figure 6.8). The

surface type of P surface is selected for all these methods for better presentation. And for different approaches, the optimal parameters, including threshold and shape parameter etc., are automatically deployed to demonstrate the best results. For more details of how to configure the parameters, you can refer to each of previous chapters.

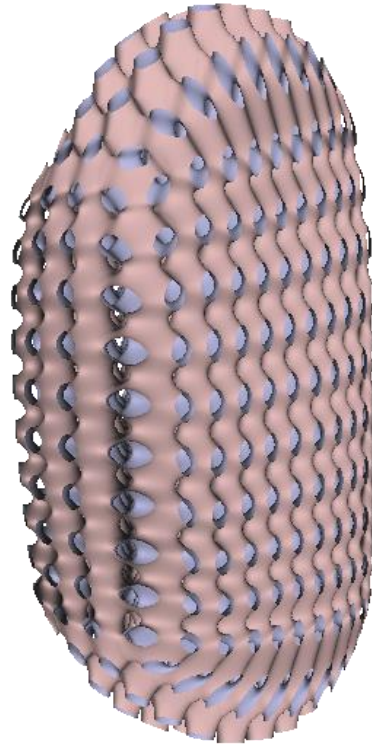
1) Parameterized hexahedral mesh and TPMS (Section 3)



(a) Front view



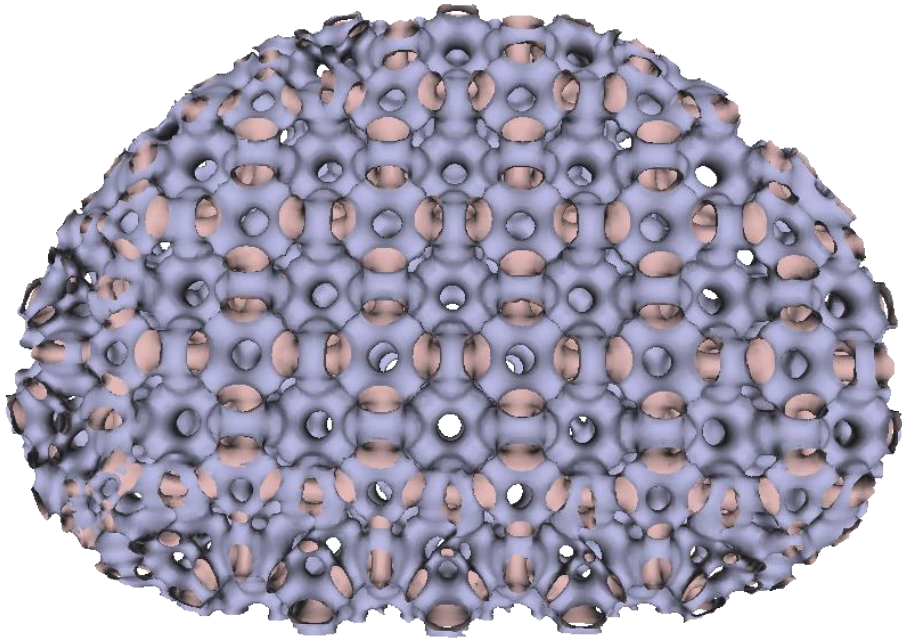
(b) Side view



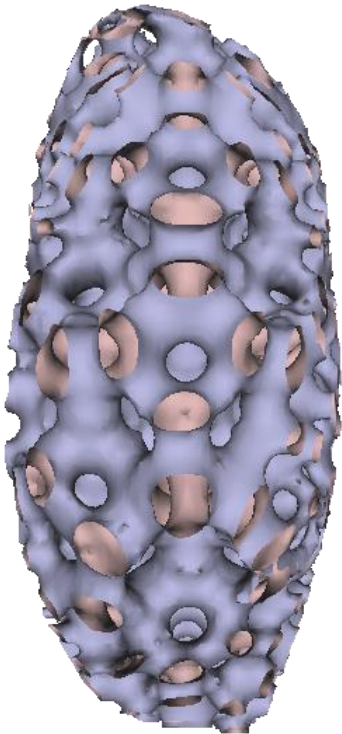
(c) Oblique view

Figure 6.8. Different views of porous IVD generated by using the parameterization method.

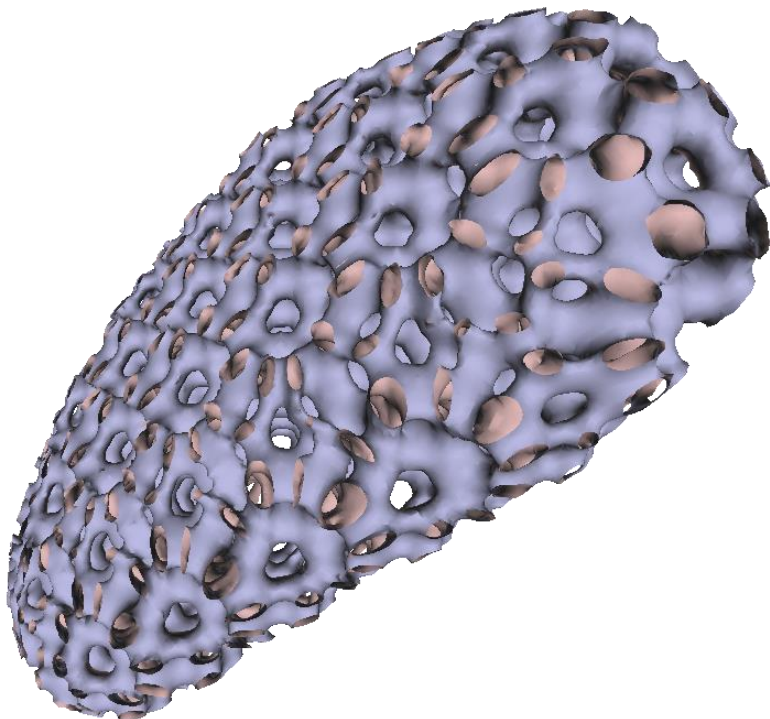
2) Tetrahedral implicit surface (Section 4)



(a) Front view



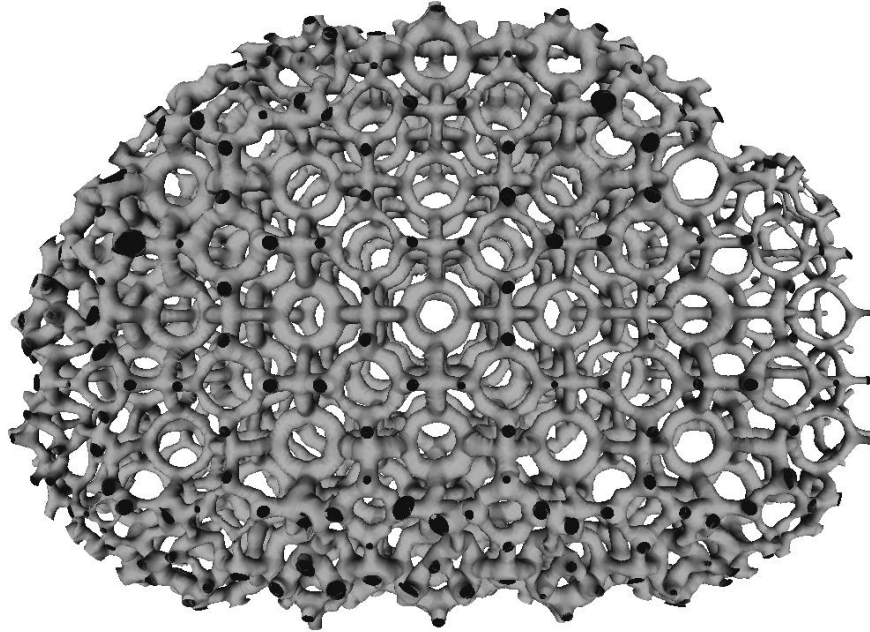
(b) Side view



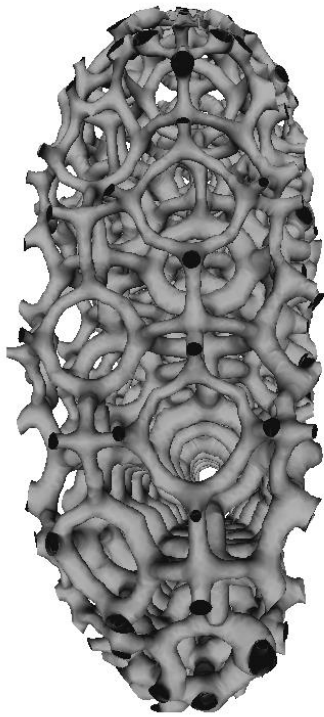
(c) Oblique view

Figure 6.9. Different views of porous IVD generated by using the TIS method.

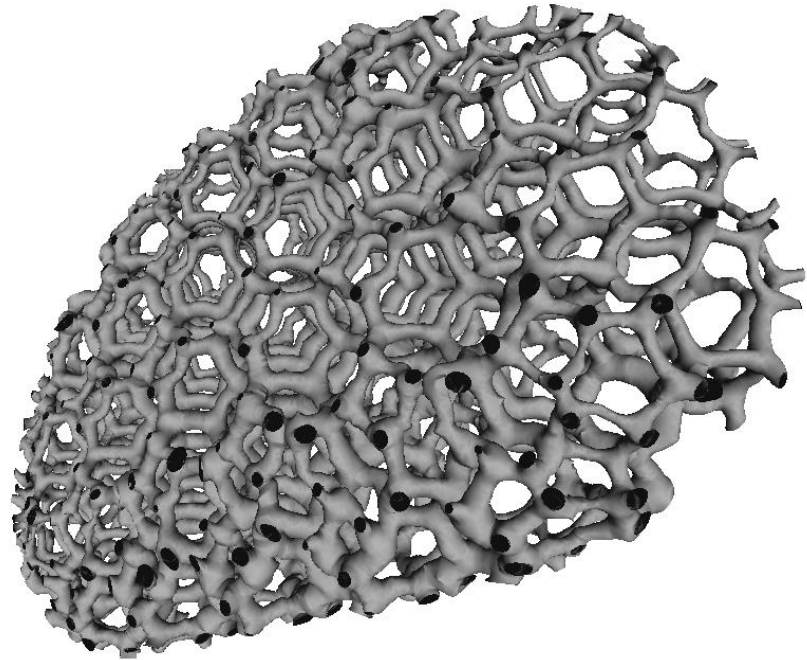
3) Anisotropic radial basis function (Section 5)



(a) Front view



(b) Side view



(c) Oblique view

Figure 6.10. Different views of porous IVD generated by using the ARBF method.

# Chapter 7

## Conclusion

As one of the most prevalent health problems amongst the world's population, low back pain (LBP) seriously affects people's work performance and well-being. Intervertebral disc degeneration has been seen as the major cause of LBP by various clinical reports. In addition to severe pain in the region of lumbar area, disk disease can cause pain in necks which may lead to splitting headache. Disc replacement has become a significant tool to help patients relieve from long-term pain in the lumbar and neck. To manufacture high-quality disc scaffold, the interconnected porous structure plays a crucial role in nutrients transportation and cell migration. In this article, three innovative porous design methods are introduced to fabricate bio-functional artificial disc scaffold through 3D printing. The first approach applies TPMS on the parameterized volumetric mesh and the second one creates implicit surface based on tetrahedral elements. The two methods both take advantages of implicit surface modeling to build repetitive minimal surface from basic units. The last method sufficiently considers the connectivity along a directional path in the usage of radial basis functions. Meanwhile, the 3D reconstruction of intervertebral discs from MR images is also incorporated to the pipeline of contrasting porous disc scaffolds. Finally, an artificial disc scaffold that can be used in the clinic could be manufactured by reading through the entire article.

## References

- [1]. Wikipedia contributors. "Intervertebral disc", Wikipedia. 6<sup>th</sup> April 2018.  
< [https://en.wikipedia.org/wiki/Intervertebral\\_disc](https://en.wikipedia.org/wiki/Intervertebral_disc)>
- [2]. Baidu Baike. "Intervertebral disc". 11<sup>th</sup> March 2014.  
< <https://baike.baidu.com/item/%E6%A4%8E%E9%97%B4%E7%9B%98>>
- [3]. Keith Bridwell, MD. "Intervertebral discs", SpineUniverse. < <https://www.spineuniverse.com/anatomy/intervertebral-discs>>
- [4]. Jane Bird. "Exploring the 3D printing opportunity", The Financial Times, 9<sup>th</sup> August 2012.  
< <https://www.ft.com/content/6dc11070-d763-11e1-a378-00144feabdc0>>
- [5]. ASTM F2792-10. Standard terminology for additive manufacturing technologies. ASTM International, West Conshohocken, PA (2010).
- [6]. Ferry P.W. Melchels, Marco A.N. Domingos, Travis J. Klein, Jos Malda, Paulo J. Bartolo, Dietmar W. Hutmacher. Additive manufacturing of tissues and organs. Progress in Polymer Science, 37(8), 1079–1104 (2012).
- [7]. P.J.S. Bartolo, H. Almeida, T. Laoui. Rapid prototyping and manufacturing for tissue engineering scaffolds. International Journal of Computer Applications in Technology, 36(1), 1–9 (2009).
- [8]. S.M. Peltola, F.P.W. Melchels, D.W. Grijpma, M. Kellomäki. A review of rapid prototyping techniques for tissue engineering purposes. Annals of Medicine, 40(4), 268–280 (2008).
- [9]. S. Mellor, L. Hao, D. Zhang. Additive manufacturing: A framework for implementation. International Journal of Production Economics, 149, 194-201 (2014).
- [10]. Wikipedia contributors. "Tissue engineering", Wikipedia, 8<sup>th</sup> April 2018.  
< [https://en.wikipedia.org/wiki/Tissue\\_engineering](https://en.wikipedia.org/wiki/Tissue_engineering)>
- [11]. T.J. Klein, J. Malda, R.L. Sah, D.W. Hutmacher. Tissue Engineering of Articular Cartilage with Biomimetic Zones. Tissue Engineering Part B, 15(2), 143-157 (2009).
- [12]. Alvaro Mata, Eun Jung Kim, Cynthia A. Boehm, Aaron J. Fleischman, George F. Muschler, Shuvo Roy. A three dimensional scaffold with precise micro-architecture and surface micro-textures. Biomaterials, 30(27), 4610-4617 (2009).
- [13]. S.M. Giannitelli, D. Accoto, M. Trombetta, A. Rainer. Current trends in the design of scaffolds for computer-aided tissue engineering. Acta Biomaterialia, 10(2), 580-594 (2014).
- [14]. W.K. Chiu, Y.C. Yeung, K.M. Yu. Toolpath generation for layer manufacturing of fractal objects. Rapid Prototyping Journal, 12(4), 214–221 (2006).



- [15]. W. Sun, B. Starly, J. Nam, A. Darling. Bio-CAD modeling and its applications in computer-aided tissue engineering. *Computer-Aided Design*, 37(11), 1097–1114 (2005).
- [16]. B.S. Bucklen, W.A. Wettergreen, E. Yuksel, M.A.K. Liebschner. Bone-derived CAD library for assembly of scaffolds in computer-aided tissue engineering. *Virtual and Physical Prototyping*, 3(1), 13–23 (2008).
- [17]. S.J. Hollister, R.D. Maddox, J.M. Taboas. Optimal design and fabrication of scaffolds to mimic tissue properties and satisfy biological constraints. *Biomaterials*, 23(20), 4095–4103 (2002).
- [18]. S.J. Hollister. Porous scaffold design for tissue engineering. *Nature Materials*, 4(7), 518–524 (2005).
- [19]. M.H. Smith, C.L. Flanagan, J.M. Kemppainen, J.A. Sack, H. Chung, S. Das, S.J. Hollister, S.E. Feinberg. Computed tomography-based tissue-engineered scaffolds in craniomaxillofacial surgery. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 3(3), 207–216 (2007).
- [20]. S. Rajagopalan, R.A. Robb. Schwarz meets schwann: design fabrication of biomorphic and durataxic tissue engineering scaffolds. *Medical Image Analysis*, 10(5), 693–712 (2005).
- [21]. T.M. Seck, F.P.W. Melchels, J. Feijen, D.W. Grijpma. Designed biodegradable hydrogel structures prepared by stereolithography using poly(ethylene glycol)/poly(d,l-lactide)-based resins. *Journal of Controlled Release*, 148(1), 34–41 (2010).
- [22]. F.P.W. Melchels, J. Feijen, J. D.W. Grijpma. A poly(d, l-lactide) resin for the preparation of tissue engineering scaffolds by stereolithography. *Biomaterials*, 30(23–24), 3801–3809 (2009).
- [23]. L. Elomaa, S. Teixeira, R. Hakala, H. Korhonen, D.W. Grijpma, J.V. Seppala. Preparation of poly(e-caprolactone)-based tissue engineering scaffolds by stereolithography. *Acta Biomaterialia*, 7(11), 3850–3856 (2011).
- [24]. D.J. Yoo. Computer-aided porous scaffold design for tissue engineering using triply periodic minimal surfaces. *International Journal of Precision Engineering and Manufacturing*, 12(1), 61–67 (2011).
- [25]. D.J. Yoo. Porous scaffold design using the distance field and triply periodic minimal surface models. *Biomaterials*, 32(31), 7741–7754 (2011).
- [26]. I. Zein, D.W. Hutmacher, K.C. Tan, S.H. Teoh. Fused deposition modeling of novel scaffold architectures for tissue engineering applications. *Biomaterials*, 23(4), 1169–1185 (2002).
- [27]. A.K. Khoda, I.T. Ozbolat, B. Koc. Engineered tissue scaffolds with variational porous architecture. *Journal of Biomechanical Engineering*, 133(1), 011001 (2011).

- [28]. C. Schroeder, W.C. Regli, A. Shokoufandeh, W. Sun. Computer-aided design of porous artifacts. *Computer-Aided Design*, 37(3), 339–353 (2005).
- [29]. S. Sogutlu, B. Koc. Stochastic modeling of tissue engineering scaffolds with varying porosity levels. *Computer-Aided Design and Applications*, 4(5), 661–670 (2007).
- [30]. D.W. Schaefer, K.D. Keefer. Structure of random porous materials: silica aerogel. *Physical Review Letters* 56, 2199–2202 (1986).
- [31]. F.P.W. Melchels, P.S. Wigganhauser, D. Warne, M. Barry, F.R. Ong, W.S. Chong, D.W. Huttmacher, J.T. Schantz. CAD/CAM-assisted breast reconstruction. *Biofabrication*, 3(3), 034114 (2011).
- [32]. S.J. Kalita. Development of controlled porosity polymer-ceramic composite scaffolds via fused deposition modeling. *Materials Science and Engineering*, 23(5), 611–620 (2003).
- [33]. A.G. Mikos, G. Sarakinos, S.M. Leite, J.P. Vacanti, R. Langer. Laminated three dimensional biodegradable foams for use in tissue engineering. *Biomaterials*, 14(5), 323–330 (1993).
- [34]. D.J. Mooney, D.F. Baldwin, N.P. Suh, J.P. Vacanti, R. Langer. Novel approach to fabricate porous sponges of poly(d,l-lactic-coglycolic acid) without the use of organic solvents. *Biomaterials*, 17(14), 1417–1422 (1996).
- [35]. J. Guan, K.L. Fujimoto, M.S. Sacks, W.R. Wagner. Preparation and characterization of highly porous, biodegradable polyurethane scaffolds for soft tissue applications. *Biomaterials*, 26(18), 3961–3971 (2005).
- [36]. P. van de Witte, P.J. Dijkstra, J.W.A. van den Berg, J. Feijen. Phase behavior of polylactides in solventnonsolvent mixtures. *Journal of Polymer Science*, 34(34), 2553–2568 (1996).
- [37]. H. Yoshikawa, N. Tamai, T. Murase, A. Myoui. Interconnected porous hydroxyapatite ceramics for bone tissue engineering. *Journal of The Royal Society Interface*, 6(Suppl 3), S3418 (2009).
- [38]. T.J. Sill, H.A. Recum. Electrospinning: Applications in drug delivery and tissue engineering. *Journal of The Royal Society Interface*, 29(13), 1989–2006 (2008).
- [39]. A. Martins, A.R.C. Duarte, S. Faria, A.P. Marques, R.L. Reis, N.M. Neves. Osteogenic induction of hbmscs by electrospun scaffolds with dexamethasone release functionality. *Journal of The Royal Society Interface*, 31(22), 5875–5885 (2010).
- [40]. M.A. Knackstedt, C.H. Arns, T.J. Senden, K. Gross. Structure and properties of clinical coralline implants measured via 3d imaging and analysis. *Biomaterials*, 27(13), 2776–2786 (2006).

- [41]. A. Tampieri, S. Sprio, A. Ruffini, G. Celotti, I. G. Lesci, N. Roveri. From wood to bone: multi-step process to convert wood hierarchical structures into biomimetic hydroxyapatite scaffolds for bone tissue engineering. *Journal of Materials Chemistry*, 19(28), 4973-4980 (2009).
- [42]. H. Lo, M.S. Ponticciello, K.W. Leong. Fabrication of controlled release biodegradable foams by phase separation. *Tissue Engineering*, 1(1), 15–28 (1995).
- [43]. S.C. Kapfer, S.T. Hyde, K. Mecke, C.H. Arns, G.E. Schröder-Turk. Minimal surface scaffold designs for tissue engineering. *Biomaterials* 32(29), 6875–6882 (2011).
- [44]. S. Torquato, S. Hyun, A. Donev. Multifunctional composites: optimizing microstructures for simultaneous transport of heat and electricity. *Physical Review Letters*, 89(26), 266601 (2002).
- [45]. S. Torquato, A. Donev. Minimal surfaces and multifunctionality. *Proceedings of The Royal Society A*, 460(2047), 1849–1856 (2004).
- [46]. H.A. Schwarz. *Gesammelte Mathematische Abhandlungen*. Springer, Berlin, 1933.
- [47]. E. R. Neovius. Bestimmung zweier spezieller periodischer Minimal Flächen. *Akad. Abhandlungen*, Helsingfors, 1883.
- [48]. Alan H. Schoen. Infinite periodic minimal surfaces without self-intersections. *NASA Technical Note*, TN D-5541 (1970).
- [49]. Hermann Karcher, Konrad Polthier. Construction of Triply Periodic Minimal Surfaces. *Philosophical Transactions of The Royal Society A*, 354(1715), 2077–2104 (1996).
- [50]. H. Karcher. The triply periodic minimal surfaces of Alan Schoen and their constant mean curvature companions. *Manuscripta Mathematica*, 64(3), 291-357 (1989).
- [51]. J. Feng, J. Fu, C. Shang, Z. Lin, B. Li. Porous scaffold design by solid T-splines and triply periodic minimal surfaces. *Computer Methods in Applied Mechanics and Engineering*, 336(1), 333-352 (2018).
- [52]. G. Xu, B. Mourrain, X. Wu, L. Chen, K. Hui. Efficient construction of multi-block volumetric spline parameterization by discrete mask method. *Journal of Computational and Applied Mathematics*, 290, 589-597 (2015).
- [53]. M. Tarini. Volume-encoded UV-maps. *ACM Transactions on Graphics (TOP)*, 35(4), Article No. 107 (2016).
- [54]. G. Auzias, J. Lefe`evre, A. Troter, C. Fischer, M. Perrot, J. R´egis, O. Coulon. Model-driven harmonic parameterization of the cortical surface: HIP-HOP. *IEEE Transactions on Medical Imaging*, 32(5), 873-887 (2013).
- [55]. W. Yu, K. Zhang, S. Wan, X. Li. Optimizing polycube domain construction for hexahedral remeshing. *Computer-Aided Design*, 46, 58-68 (2014).

- [56]. B. Li, X. Li, K. Wang, H. Qin. Surface mesh to volumetric spline conversion with generalized polycubes. *IEEE Transactions on Visualization and Computer Graphics*, 19(9), 1539-1551 (2013).
- [57]. H. Xu, W. Yu, S. Gu, X. Li. Biharmonic volumetric mapping using fundamental solutions. *IEEE Transactions on Visualization and Computer Graphics*, 19(5), 787-798 (2013).
- [58]. X. Li, H. Xu, S. Wan, Z. Yin, W. Yu. Feature-aligned harmonic volumetric mapping using MFS. *Computer and Graphics*, 34(3), 242-251 (2010).
- [59]. G.P. Paill'e, P. Poulin. As-conformal-as-possible discrete volumetric mapping. *Computer and Graphics*, 35(5), 427-433 (2012).
- [60]. S. Han, J. Xia, Y. He. Constructing hexahedral shell meshes via volumetric polycube maps. *Computer-Aided Design*, 43(10), 1222-1233 (2011).
- [61]. T. Martin, E. Cohen. Volumetric parameterization of complex objects by respecting multiple materials. *Computers and Graphics*, 34(3), 187-197 (2010).
- [62]. M. Li, S. Liao, R. Tong. Facial hexahedral mesh transferring by volumetric mapping based on harmonic fields. *Computers and Graphics*, 35(1), 92-98 (2011).
- [63]. J. Li, G. Lu. Customizing 3D garments based on volumetric deformation. *Computers in Industry*, 62(7), 693-707 (2011).
- [64]. K. Crane, C. Weischedel, M. Wardetzky. A new approach to computing distance based on heat flow. *ACM Transactions on Graphics (TOG)*, 32(5) (2013).
- [65]. S. Dong, S. Kircher, M. Garland. Harmonic functions for quadrilateral remeshing of arbitrary manifolds. *Computer Aided Geometric Design*, 22(5), 392-423 (2005).
- [66]. B. Li, H. Qin. Feature-Aware Reconstruction of Volume Data via Trivariate Splines. *Pacific Graphics Short Papers*, 049-054 (2011).
- [67]. X. Wang, X. Qian. An optimization approach for constructing trivariate B-spline solids. *Computer-Aided Design*, 46, 179-191 (2014).
- [68]. Y. Bai, D. Wang. On the Comparison of Trilinear, Cubic Spline, and Fuzzy Interpolation Methods in the High Accuracy Measurements. *IEEE Transactions on Fuzzy Systems*, 18(5), 1016-1022 (2010).
- [69]. TetGen. Research Group: Numerical Mathematics and Scientific Computing Weierstrass Institute for Applied Analysis and Stochastics (WIAS). Berlin, Germany. <<http://wias-berlin.de/software/index.jsp?id=TetGen&lang=1>>
- [70]. P. George, P. Frey. *Mesh Generation: Application to Finite Elements*, 2<sup>nd</sup> edition. Wiley, 2013.

- [71]. Q. Du, V. Faber, M. Gunzburger. Centroidal Voronoi Tessellations: Applications and Algorithms. *SIAM Review*, 41(4), 637-676 (1999).
- [72]. G. Miller, D. Talmor, S. Teng, N. Walkington, H. Wang. Control Volume Meshes using Sphere Packing: Generation, Refinement and Coarsening. 5th International Meshing Roundtable, Pittsburgh PA, 1996.
- [73]. P. Chew. Guaranteed-Quality Delaunay Meshing in 3D. SCG '97 Proceedings of the thirteenth annual symposium on Computational geometry, 274-280 (1997).
- [74]. P. Alliez, D. Cohen-Steiner, M. Yvinec, M. Desbrun. Variational Tetrahedral Meshing. *ACM Transactions on Graphics*, 24(3), 617-625 (2005).
- [75]. W. Choi, D. Kwak, I. Son, Y. Im. Tetrahedral Mesh Generation based on Advancing Front Technique and Optimization Scheme. *International Journal for Numerical Methods in Engineering*, 58, 1857-1872 (2003).
- [76]. N. Molino, R. Bridson, J. Teran, R. Fedkiw. A Crystalline, Red Green Strategy for Meshing Highly Deformable Objects with Tetrahedra. 12<sup>th</sup> International Meshing Roundtable, 103-114 (2003).
- [77]. A. Fuchs. Automatic Grid Generation with Almost Regular Delaunay Tetrahedra. 7<sup>th</sup> International Meshing Roundtable, 133-148 (1998).
- [78]. Y. Ito, A.M. Shih, B.K. Soni. Octree-based Reasonable-quality Hexahedral Mesh Generation using a New Set of Refinement Templates. *International Journal for Numerical Methods in Engineering*, 77, 1809–1833 (2009).
- [79]. J. Wang, Z. Yu, Feature-sensitive tetrahedral mesh generation with guaranteed quality. *Computer-Aided Design*, 44(5), 400–412 (2012).
- [80]. Z. Gao, Z. Yu, M. Holst. Quality tetrahedral mesh smoothing via boundary-optimized delaunay triangulation. *Computer Aided Geometric Design*, 29(9), 707-721 (2012).
- [81]. A. Fogden, S.T. Hyde. Parametrization of triply periodic minimal surfaces. I. mathematical basis of the construction algorithm for the regular class. *Acta Crystallographica*, 48(4), 11–12 (1992).
- [82]. H. Chen, Y. Guo, R. Rostami, S. Fan, K. Tang, Z. Yu. Porous Structure Design Using Parameterized Hexahedral Meshes and Triply Periodic Minimal Surfaces. *Proceedings of Computer Graphics International 2018 (CGI 2018)*, 117-128 (2018).
- [83]. J. Wang, G. Liu. On the optimal shape parameters of radial basis functions used for 2-D meshless methods. *Computer Methods in Applied Mechanics and Engineering*, 191(23-24), 2611–2630 (2002).
- [84]. E. Divo, A. Kassab. An efficient localized RBF meshless method for fluid flow and conjugate heat transfer. *ASME Journal of Heat Transfer*, 129(2), 124–136 (2006).
- [85]. G. Kosec, B. Sarler. Local RBF collocation method for darcy flow. *Computer Modeling in Engineering and Sciences*, 25, 197–208 (2008).

- [86]. B. Sarler, R. Vertnik. Meshfree explicit local radial basis function collocation method for diffusion problems. *Computers and Mathematics with Applications*, 51(8), 1269–1282 (2006).
- [87]. R. Vertnik, B. Sarler. Meshless local radial basis function collocation method for convective-diffusive solid-liquid phase change problems. *International Journal of Numerical Methods for Heat and Fluid Flow*, 16(5), 617–640 (2006).
- [88]. R. Vertnik, B. Sarler. Solution of incompressible turbulent flow by a mesh-free method. *Computer Modeling in Engineering and Sciences*, 44(1), 65–95 (2009).
- [89]. Nikolett Bereczky, Amalia Duch, Krisztián Németh, Salvador Roura. Quad-K-d Trees. *Latin American Symposium on Theoretical Informatics*, 743-754 (2014).
- [90]. J. Carballido-Gamio, S. Belongie, S. Majumdar. Normalized cuts in 3-d for spinal MRI segmentation. *IEEE Transactions on Medical Imaging*, 23(1), 36–44 (2004).
- [91]. S. Huang, Y. Chu, S. Lai, C.L. Novak. Learning-based vertebra detection and iterative normalizedcut segmentation for spinal MRI. *IEEE Transactions on Medical Imaging*, 28(10), 1595–1605 (2009).
- [92]. C. Chevretil, F. Cheriet, C. Aubin, G. Grimard. Texture analysis for automatic segmentation of intervertebral disks of scoliotic spines from MR images. *IEEE Transactions on Information Technology in Biomedicine*, 13(4), 608–620 (2009) .
- [93]. S.K. Michopoulou, L. Costaridou, E. Panagiotopoulos, R. Speller, G. Panayiotakis, A. Todd-Pokropek. Atlas-based segmentation of degenerated lumbar intervertebral discs from MR images of the spine. *IEEE Transactions Biomedical Engineering*, 56(9), 2225–2231 (2009).
- [94]. S. Ghosh, R.S. Alomari, V. Chaudhary, G. Dhillon. Computer-aided diagnosis for lumbar MRI using heterogeneous classifiers. *2011 IEEE International Symposium on Biomedical Imaging: From Nano to Macro, Chicago IL* (2011).
- [95]. S. Seifert, I. Wächter, Rüdiger Dillmann. Segmentation of Intervertebral Discs Trachea and Spinal Cord from MRI Images. *International Journal of Computer Assisted Radiology and Surgery* (2006).
- [96]. A. Neubert, J. Frupp, S. Chandra, S. Engstrom, S. Crozier, S. Automated intervertebral disc segmentation using probabilistic shape estimation and active shape models. *Computational Methods and Clinical Applications for Spine Imaging (CSI 2015)*, 150-158 (2016).
- [97]. B. Glocker, J. Feulner, A. Criminisi, D.R. Haynor, E. Konukoglu. Automatic localization and identification of vertebrae in arbitrary field-of-view CT scans. *Medical Image Computing and Computer-Assisted Intervention (MICCAI 2012)*, 590–598 (2012).

- [98]. B. Glocker, E. Konukoglu, D. Haynor. Random forests for localization of spinal anatomy. *Medical Image Recognition, Segmentation and Parsing: Methods, Theories and Applications*, 94–110 (2016).
- [99]. Y. Zheng, A. Barbu, B. Georgescu, M. Scheuering, D. Comaniciu. Four-chamber heart modeling and automatic segmentation for 3-D cardiac CT volumes using marginal space learning and steerable features. *IEEE Transactions on Medical Imaging*, 27(11), 1668–1681 (2008).
- [100]. M.B. Kelm, M. Wels, K.S. Zhou, S. Seifert, M. Suehling, Y. Zheng, D. Comaniciu. Spine detection in CT and MR using iterated marginal space learning. *Medical Image Analysis*, 17(8), 1283–1292 (2013).
- [101]. C. Chen, D. Belavy, W. Yu, C. Chu, G. Armbrecht, M. Bansmann, D. Felsenberg, G. Zheng. Localization and segmentation of 3d intervertebral discs in MR images by data driven estimation. *IEEE Transactions on Medical Imaging*, 34(8), 1719–1729 (2015).
- [102]. Z. Wang, X. Zhen, K. Tay, S. Osman, W.M. Romano, S. Li. Regression segmentation for M(3) spinal images. *IEEE Transactions on Medical Imaging*, 34(8), 1640–1648 (2014).
- [103]. Y. Cai, S. Osman, M. Sharma, M. Landis, S. Li. Multi-modality vertebra recognition in arbitrary view using 3d deformable hierarchical model. *IEEE Transactions on Medical Imaging*, 34(8), 1676–1693 (2015).
- [104]. H. Chen, C. Shen, J. Qin, D. Ni, L. Shi, J.C. Cheng, P.A. Heng. Automatic localization and identification of vertebrae in spine CT via a joint learning model with deep neural networks. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 515–522 (2015).
- [105]. H. Chen, Q. Dou, X. Wang, J. Qin, J.C. Cheng, P.A. Heng. 3D fully convolutional networks for intervertebral disc localization and segmentation. *International Conference on Medical Imaging and Virtual Reality*, 375–382 (2016).
- [106]. K. Kamnitsas, C. Ledig, V.F. Newcombe, J.P. Simpson, A.D. Kane, D.K. Menon, D. Rueckert, B. Glocker. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Medical Image Analysis*, 36, 61–78 (2017).
- [107]. G. Zeng, G. Zheng. DSMS-FCN: A deeply supervised multi-scale fully convolutional network for automatic segmentation of intervertebral disc in 3D MR images. *Computational Methods and Clinical Applications in Musculoskeletal Imaging (MSKI 2017)*, 148-159 (2018).
- [108]. X. Li, Q. Dou, H. Chen, C.W. Fu, X. Qi, D.L. Belavý, G. Armbrecht, D. Felsenberg, G. Zheng, P.A. Heng. 3D multi-scale FCN with random modality voxel dropout learning for Intervertebral Disc Localization and Segmentation from Multi-modality MR Images. *Medical Image Analysis*, 45, 41-54 (2018).

- [109]. H. Dong, G. Yang, F. Liu, Y. Mo, and Y. Guo. Automatic Brain Tumor Detection and Segmentation Using U-Net Based Fully Convolutional Networks. *Annual Conference on Medical Image Understanding and Analysis*, 506-517 (2017).
- [110]. X. Li, H. Chen, X. Qi, Q. Dou, C. Fu, and P. A. Heng. H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation from CT Volumes. *IEEE Transactions on Medical Imaging*, 37(12), 2663-2674 (2018).
- [111]. W. Yao, Z. Zeng, C. Lian, and H. Tang. Pixel-wise regression using U-Net and its application on pansharpening. *Neurocomputing*, 312, 364-371 (2018).
- [112]. H. Chen, Q. Dou, X. Wang, P. A. Heng. Deepseg: Deep segmentation network for intervertebral disc localization and segmentation. *Proc. 3rd MICCAI Workshop & Challenge on Computational Methods and Clinical Applications for Spine Imaging* (2015).
- [113]. Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. 3D U-Net: learning dense volumetric segmentation from sparse annotation. *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 424-432 (2016).
- [114]. S. Kim, W. Bae, K. Masuda, C. Chung, and D. Hwang. Fine-grain segmentation of the intervertebral discs from MR spine images using deep convolutional neural networks: BSU-Net. *Applied Sciences*, 8(9), 1656 (2018).
- [115]. J. Dolz, C. Desrosiers, and I. B. Ayed. IVD-Net: Intervertebral disc localization and segmentation in MRI with a multi-modal Unet. *arXiv preprint arXiv:1811.08305* (2018).
- [116]. Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *CVPR 2017*.
- [117]. J.T. Lu, S. Pedemonte, B. Bizzo, S. Doyle, K.P. Andriole, M.H. Michalski, R.G. Gonzalez, S.R. Pomerantz. DeepSPINE: Automated Lumbar Vertebral Segmentation, Disc-level Designation, and Spinal Stenosis Grading Using Deep Learning. *arXiv preprint arXiv:1807.10215* (2018).
- [118]. C. Chen, D. Belavy, and G. Zheng. 3D intervertebral disc localization and segmentation from MR images by data-driven regression and classification. *International Workshop on Machine Learning in Medical Imaging*, 50-58 (2014).
- [119]. D.G. Pelli, P. Bex. Measuring contrast sensitivity. *Vision Research*, 90, 10-14 (2013).
- [120]. Agnieszka Mikołajczyk, Michał Grochowski. Data augmentation for improving deep learning in image classification problem. *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, Swinoujście, Poland (2018).
- [121]. L. Engstrom, D. Tsipras, L. Schmidt, A. Madry. A Rotation and a Translation Suffice: Fooling CNNs with Simple Transformations. *2017 International Conference on Learning Representations* (2017).



- [122]. Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio. Generative Adversarial Networks. arXiv preprint arXiv:1406.2661 (2014).
- [123]. C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. arXiv preprint arXiv:1609.04802 (2016).
- [124]. H. Wu, S. Zheng, J. Zhang, K. Huang. Gp-gan: Towards realistic high-resolution image blending. arXiv preprint arXiv:1703.07195 (2017).
- [125]. S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele, H. Lee. Generative adversarial text to image synthesis. arXiv preprint arXiv:1605.05396 (2016).
- [126]. A. A. Efros, W. T. Freeman. Image quilting for texture synthesis and transfer. Proceedings of the 28th annual conference on Computer graphics and interactive techniques, 341-346 (2001).
- [127]. J. Jin, A. Dundar, E. Culurciello. Robust convolutional neural networks under adversarial noise. arXiv preprint arXiv:1511.06306 (2015).
- [128]. Brandon J. Whitcher, Volker J. Schmid, Andrew J. Thornton. Working with the DICOM and NIfTI Data Standards in R. Journal of Statistical Software, 44(6), 11173 (2011).
- [129]. G. Zheng, C. Chu, D. L. Belavý, B. Ibragimov, R. Korez, T. Vrtovec, and H. Hutt et al. Evaluation and comparison of 3D intervertebral disc localization and segmentation methods for 3D T2 MR data: A grand challenge. Medical image analysis, 35, 327-344 (2017).
- [130]. S. Park, and N. Kwak. Analysis on the dropout effect in convolutional neural networks. Asian Conference on Computer Vision, 189-204 (2016).
- [131]. F. Chollet et al. Keras. chollet2015keras, <<https://keras.io>>.
- [132]. S. S. Girija. Tensorflow: Large-scale machine learning on heterogeneous distributed systems, 2016.
- [133]. K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. Proceedings of the IEEE international conference on computer vision, 1026-1034 (2015).
- [134]. D. P. Kingma, and J. Ba. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014).
- [135]. Q. Fang, D. A. Boas. Tetrahedral mesh generation from volumetric binary and grayscale images. Biomedical Imaging: From Nano to Macro, ISBI'09, IEEE International Symposium, 1142-1145 (2009).
- [136]. R. Korez, B. Ibragimov, B. Likar, F. Pernuš, T. Vrtovec. Deformable model-based segmentation of intervertebral discs from MR spine images by using the SSC descriptor.

International Workshop on Computational Methods and Clinical Applications for Spine Imaging, 117-124 (2015).

- [137]. P. Cignoni, M. Callieri, M. Corsini, M. Dellepiane, F. Ganovelli, G. Ranzuglia. MeshLab: An Open-Source Mesh Processing Tool. Sixth Eurographics Italian Chapter Conference, 129-136 (2008).
- [138]. D. Yoo. Heterogenous minimal surface porous scaffold design using the distance field and radial basis functions. *Medical Engineering and Physics*, 34, 625-639 (2012).
- [139]. F.J. O'Brien. Biomaterials & scaffolds for tissue engineering. *Materials Today*, 14(3), 88-95 (2011).
- [140]. A.I. Bokenko, B.A. Springborn. A discrete Laplace-Beltrami operator for simplicial surfaces. *Discrete and Computational Geometry*, 38(4), 740-756 (2007).
- [141]. T. Caissard, D. Coeurjolly, J.O. Lachaud, T. Roussillon. Heat kernel Laplace-Beltrami operator on digital surfaces. *International Conference on Discrete Geometry for Computer Imagery*, 241-253 (2017).

## Appendix

- [1] Huawei Chen\*, Ye Guo\*, Reihaneh Rostami, Shuqian Fan, Kanglai Tang, and Zeyun Yu, "Porous Structure Design Using Parameterized Hexahedral Meshes and Triply Periodic Minimal Surfaces", *Proceedings of Computer Graphics International 2018 (CGI 2018)*, Bintan, Island, Indonesia, ACM, pages 117-128, 2018.
- [2] Ye Guo, Ke Liu, and Zeyun Yu, "Porous Structure Design in Tissue Engineering Using Anisotropic Radial Basis Functions", *International Symposium on Visual Computing (ISVC 2018)*, Las Vegas, LNCS 11241, pages 79-90, 2018.
- [3] Ye Guo, Ke Liu, and Zeyun Yu, "Tetrahedron-Based Porous Scaffold Design for 3D Printing", *Designs* 3(1), page 16, 2019.
- [4] Chuanbo Wang, Ye Guo, and Zeyun Yu, "Fully Automatic Intervertebral Disc Segmentation Using Multimodal 3D U-Net", *2019 IEEE Computer Society Signature Conference on Computers (COMPSAC 2019)*. (Accepted)

## **Curriculum Vitae**

Ye Guo

Place of birth: Huanggang, Hubei, China PRC

Education:

B.S., Wuhan University, June 2012

Major: Software Engineering

B.S., University of Wisconsin – La Crosse, May 2012

Major: Software Engineering

M.S., University of Wisconsin – La Crosse, Aug 2013

Major: Software Engineering

Dissertation Title: CAD-based Porous Intervertebral Disc Scaffold Design in Tissue Engineering