UNIVERSIDAD
NACIONAL
DE COLOMBIA

3D reconstruction of anatomical structures using interpolation techniques and local approaches.

Luis Felipe Arteaga Daza

Universidad Nacional de Colombia
Faculta de Ingeniería y Arquitectura
Departamento de Ingeniería Eléctrica y Eléctronica
Manizales, Colombia
2019

3D reconstruction of anatomical structures using interpolation techniques and local approaches.

Luis Felipe Arteaga Daza

Tesis o trabajo de grado presentado como requisito parcial para optar al título de:
Magíster en Ingeniería - Automatización Industrial

Director(a):
Ph.D. Cesár Germán Castellanos Domínguez
Co-Director(a):
Ph.D. David Cárdenas Peña

Línea de Investigación:
Procesamiento de Imagines y Señales
Grupo de Investigación:
Grupo de Control y Procesamiento Digital de Señales

Universidad Nacional de Colombia
Faculta de Ingeniería y Arquitectura
Departamento de Ingeniería Eléctrica y Eléctronica
Manizales, Colombia
2019

Hay una fuerza motriz más poderosa que el vapor, la electricidad y la energía atómica: la voluntad

Albert Einstein

Agradecimientos

Quiero agradecer muy especialmente a mi director de tesis Ph.D. Cesár Germán Castellanos Domínguez; y co-director Ph.D. David Cárdenas Peña, por todas y cada unas de las acertadas orientaciones que me brindaron y que permitiron que esta tesis llegara a buen termino, y especialmente a David por el tiempo brindado y su ayuda. A mis padres y hermana, por apoyarme y animarme para comenzar la realización de la tesis y la culminación de la misma, por demostrarme siempre lo orgullosos que están de mí, por hacer mis problemas también suyos y acompañarme en cada uno de los logros que deseo alcanzar. Quiero agradecer a mis amigos del grupo de investigación, por todo el tiempo que hemos compartido y agradecerles por todos los buenos momentos a lo largo des estos años. Esto no habría sido posible sin ustedes.

Esta investigación se llevó a cabo en el marco del proyecto, Desarrollo de un sistema de identificación de estructuras nerviosas en imágenes de ultrasonido para la asistencia del bloqueo de nervios periféricos. Aplicación al tratamiento de dolor agudo traumático y revención del dolor neuropático crónico con código COLCIENCIAS 1110-744-55958

Resumen

La reconstrucción de superficie es el proceso mediante el cual un objeto 3D se reproduce de una colección de valores discretos que muestrean la forma. Estos valores generalmente son llamados nubes de puntos. Comúnmente, los métodos de reconstrucción se basan en las propiedades básicas de las nubes de puntos, que son la densidad de muestras, ruido, datos faltantes y los valores atípicos. Nuestro objetivo es reconstruir la superficie de estructuras anatómicas a partir de imágenes médicas. Consideraremos dos problemas principales que son los datos faltantes y la presencia de ruido.

Resolvemos la falta de datos generando nuevos muestras a partir de un conjunto de contornos, basándonos en técnicas de *Shape Morphing* (forma cambiante). Si adicionamos ruido al problema anterior, debemos cambiar de enfoque por lo tanto utilizamos un método de reconstrucción implícita que se ha demostrado que es robusto a los problemas anteriormente mencionados. Por último combinamos parte de las propuestas anteriores para resolver un problema especifico, que se presenta cuando reconstruimos imágenes médicas, que se trata, cuando un contorno se bifurca en otros contornos. Los métodos se evaluarán sobre bases de datos públicas de imágenes médicas y se compararon con dos algoritmos estándar del estado del arte y la medición de rendimiento de reconstrucción sera la distancia de Hausdorff.

Palabras claves: 3D reconstrucción, nube de puntos, forma cambiante, bifurcacion, fusión local

Abstract

The reconstruction of the surface is the process by which a 3D object is reproduced from a collection of discrete values that sample the shape. These values are generally called point cloud. Commonly, the reconstruction methods are based on the fundamental properties of the point clouds, which are the density samples, noise, missing data, and outliers. We aim to reconstruct the surface of anatomical structures from medical images; Consider two main problems that are missing data and the presence of noise.

We resolve the missing data by generating new samples from a set of contours based on Shape Morphing techniques. If we add noise to the previous problem, we must change the focus and therefore use an implicit rebuild method that is solid for the problems presented above. Finally, we combine part of the previous proposals to solve a specific problem, that occurs when we reconstruct medical images, when a contour is bifurcated into another. The methods are evaluated with the public database from medical images and compared with the standardized algorithms of state of the art and the Hausdorff distance is used to measure the performance.

Keywords: 3D reconstruction, point cloud, shape morphing, bifurcation, local merging

Contents

List of Figures

1 Introduction

Surface reconstruction is increasingly essential in geometric modeling for generating surfaces from data points captured from real objects, often by laser range scanners but also by hand-held digitizers, computer vision techniques, or among other technologies. Industrial applications include reverse engineering, product design, and the construction of personalized medical appliances [3]. Reconstruction methods have different types of entry requirements generally associated with a point cloud. The minimum requirement of all reconstruction algorithm is a set of 3D points lying on the surface [5]. The properties of the point cloud impact on the reconstruction algorithms and methods. The most common properties are sample density, noise, missing data, and atypical values [6]. Some of these properties depend on how point clouds are acquired or processed. All methods of point cloud reconstruction in the state of the art encompass one or more of these problems, as can be read in [18, 16, 25].

This thesis work is devoted to reconstruct the surface of anatomical structures, due the analysis of medical images is an essential tool in the diagnosis of diseases or injuries inside the body [34]. To accurately study organs, medical personnel usually acquire images with a variety of devices and techniques that can create models of structures and their physiology. The most common types of models are magnetic resonance imaging (MRI), X-rays, ultrasound (US), computerized tomography, and nuclear medicine studies. The most visible and relevant information for medical personnel is the boundary of the anatomical structure that is shown in the scene [24, 9]. When working on the reconstruction of medical images, we find two significant problems that are related to the missing data and noise.

The manipulation of missing data is a common situation, where only parts of the data can be used [36]. This problem is originated in the medical image segmentation, despite advances in techniques and methods for the automatic segmentation of anatomical structures, due to the ambiguity and noise of the images. The medical staff is nowadays in charge of manually delimiting the structures in the volume segments [14]. Therefore, an algorithm that reconstructs the surface of these structures must have the least possible amount to create a satisfactory result. We propose a methodology that generates new data based on segmented base contours. This allows us to have a small number of contours and still obtain an excellent approximate reconstruction of the anatomical structure. To generate new contours, we use Shape Morphing techniques, which is a field of research in computer science, dealing with algorithms and methods to build a gradual and continuous transformation between two

flat or stable shapes [17]. We assume that the surface of specific anatomical structures has smooth changes. Therefore, by interpolating shapes between the contours, we would not be alternating the form of the surface. We compare our approach with the standard algorithms of the state of the art [3] and [22].

In addition to missing data, clouds generally have a presence of noise, that forces the reconstruction methods to produce surfaces as close to the point cloud and as smooth as possible to overcome noisy points. Some robust noise algorithms employ robust statistics and impose softness on their reconstructions [15, 29]. In the reconstruction of surfaces, implicit representation schemes are the most used, since they are independent on the topology, robust in the absence of data and noisy data [11]. The implicit reconstruction methods use a signed-distance function suitably fitting the point cloud, and can be classified into two categories: global approaches where the value to interpolate a point depends on the data from the whole point cloud, and local ones that insert values depending on the neighboring points [11]. One of the significant advantages of local methods is the improved detail on the reconstruction in comparison to global methods [20]. To take advantage of local methods, this work proposes a reconstruction by parts that allows a more precise reconstruction in certain regions.

When reconstructing medical images, cross-sections are often used that are sensitive to noise, and number of sections [19, 10]. But the most relevant issue for reconstructing some anatomical structures is the contour bifurcation due to the lack of data [12, 26]. A bifurcation is understood as a contour in a 2D slide that is divided into several contours in the next slide. We will only focus on a specific case that is known as *one to many*, in which part of a single base contour that is connected to multiple contours [12].

1.1. Objetives

1.1.1. Geneal Objective

Develop a methodology for 3D reconstruction of anatomical structures from medical imaging based on data interpolation and local approaches to enhance the 3D model quality.

1.1.2. Specefic Objectives

- Develop a methodology for data interpolation aiming to improve 3D reconstruction of anatomical structures based on morphing.

- Develop a 3d reconstruction methodology using support vector and local merging approach for enhancing the data fitting on spatially varying shapes.

- Develop a rule-based approach for improving the data fitting of 3D reconstruction models in the presence of bifurcations.

2 3D Reconstruction by deformation of contours

2.1. Materials and Methods

2.1.1. Point Cloud

Given a three-dimensional point cloud with N samples $\mathcal{P}=\{\boldsymbol{p}_i \in \mathbb{R}^3, i = \{1, ..., N\}\}$, where each $\boldsymbol{p}_i$ samples the surface $\mathcal{S}$ of a three-dimensional real object by scanning and registration procedures, the reconstruction task aim at estimate the most similar surface to $\mathcal{S}$ [20, 27]. The estimated surface, defined as $\hat{\mathcal{S}}=\{\mathcal{V}, \mathcal{T}\}$, is composed by a set of vertices $\mathcal{V}=\{\boldsymbol{v}_l \in \mathbb{R}^3, l = \{1, ..., V\}\}$ and a set of triangles $\mathcal{T}=\{\boldsymbol{t}_n \in \{1, ..., V\}^3, n = \{1, ..., T\}\}$ where $\boldsymbol{v}_l$ is the l-th vertex, $\boldsymbol{t}_n$ indexes the coordinates of the n-th triangle in the vertex set, V is the number of vertices, and T is the number of triangles of the 3D reconstruction.

2.1.2. Curve Representation

We assume a contour as a parameterized closed curve, ζ, given by $\zeta = D \to \mathbb{R}^d$, where D is a domain that determines the parameterization, $D = [0, 1]$, and $d \in \{2, 3\}$ corresponds to the space dimension [33]. The curve ζ can be extracted from a medical image by segmenting the region of interest and extracting the points over the boundary. The shape analysis must be carried out on a shape space representation endowed with a metric that quantifies the differences between contours. The metric must be invariant to the shape preserving transformations, translation, scaling, rotation, and parameterization [17]. When comparing two contours, it is necessary to know which point on each shape corresponds to another point on the other, that is the correspondence problem. The correspondence between two curves ζ_1 and ζ_2 is equal to the optimal rotation $O \in S_R O(d)$ from the Special Orthogonal Group and re-parametrization $\gamma \in \Gamma$, such that the distance between the curves is ζ_1 and $O(\zeta_2)$ is minimal [17].

To analyze a closed curve ζ, a mapping is defined as $H : \mathbb{R}^d \to \mathbb{R}^d$, where $H(v) = v/\sqrt{\|v\|}$, if $\|v\| \neq 0$ and 0 otherwise. Here, $\|\bullet\|$ is the euclidean 2-norm in $\mathbb{R}^d$, noting that H is a continuos map. To study the form of ζ, it is represented using the Square Root Velocity Function (SRVF) defined as $q : D \to \mathbb{R}$ [33], where

$$q(t) \equiv H(\dot{\zeta}(t)) = \frac{\dot{\zeta}(t)}{\sqrt{\left\|\dot{\zeta}(t)\right\|}} \tag{2-1}$$

$\dot{()}$ is the derivative, for every $q \in \mathbb{L}^2(D, \mathbb{R}^d)$, there exists a curve ζ such that the given q is the SRVF of that ζ. The curve can be re-obtained $\zeta(t) = \int_0^t q(\ell)\,\|q(\ell)\|\,d\ell$ [33].

Let U be the set of all the representations of SRVFs of the closed curves. U can be called pre-shape space, therefore, it is a hypersphere in the Hilbert manifold [33]. When we compare two closed curves in the pre-shape space, it is necessary to calculate the shortest path length, called *geodesic*, which connects its corresponding points in U. This measure of dissimilarity is not invariant to the rotations and depends on the parameterization of the curves[17].

The rotation of a contour curve ζ with $O \in S_R O(d)$ given another curve $O\zeta$ is equivalent to ζ from the point of view of the shape analysis. Then, the reparameterization of ζ is also applied with a diffeomorphism $\gamma \in \Gamma$[17]. Therefore, all rotations and diffeomorphisms in the ζ curve will generate a set of curves that have the same shape. This set is equal to class under the product $\gamma \times S_R O(d)$. The SVRF of $O(\zeta \circ \gamma)$, denoted by $O(q, \gamma)$ is given by $\sqrt{\dot{\gamma}}O(q \circ \gamma)(t)$ [33]. $[q]$ is defined as the orbit of q, which is the set of all curves that are similar to q in the rotation and parameterization [17].

$$[q] = \{O(q, \gamma) = \sqrt{\dot{\gamma}}O(q \circ \gamma)(t) | O \in S_R O(d), \gamma \in \Gamma\} \tag{2-2}$$

$\mathcal{O}$ is the set of all the orbits, therefore, it is our shape space. $\mathcal{O}$ would form a quotient space of U modulo $\Gamma \times S_R O(2)$. Therefore, $\mathcal{O}$ inherits the Riemannian metric from U[17]. Taking into account the previous representation, the analysis of the curves is equal to the analysis of their orbits, which in this case are the elements of the shape space U. Then, the dissimilarity between two curves of the contours ζ_1 andζ_2 is defined as the length of the shortest geodesic that join the elements of their orbits, $\rho \in \mathbb{R}^+$ [17].

$$\rho(\zeta_1, \zeta_2) = \rho_s([q_1, q_2]) = \min_{O \in S_R O(d), \gamma \in \Gamma} \rho_c(q_1, O(q_2, \gamma)) \tag{2-3}$$

where $\rho_c(q_1, q_2)$ defines the length of the geodesic between ζ_1 and ζ_2.It is necessary to optimize in O and γ, to find the optimal correspondence between ζ_1 and ζ_2. ρ addition to measuring the length of the geodesic, also the rotation and the parameterization between ζ_1 and ζ_2, both are represented with their SRVFs. A geodesic is a continuous deformation, which flexes and stretches a curve towards the other. Its length quantifies the proportion of deformation necessary to deform a curve into other [17]. The optimization problem 2-3 is solved following the procedure described in [33]. A geodesic in two curves in the form space this definition by:

$$q_\tau = \frac{1}{sin(\Theta)}(sin(\Theta(1-\tau)))q_1 - sin(\Theta\tau)q_2, \tag{2-4}$$

where $\Theta = cos^{-1}(\langle q_1, q_2 \rangle)$ and $\tau \in \{0, 1\}$ quantifies the amount of deformation.

2.1.3. Power Crust algorithm

The power crust (PC) algorithm approximates the surface $\mathcal{S}$ using the cloud of points $\mathcal{P}$ based on the inverse of the medial axis transformation (MAT) [3]. MAT represents an object by a set of maximum containing balls. Despite PC exactly represents $\mathcal{S}$, computations of MAT and its inverse are extremely complex and computationally expensive [3].

Assuming that a bounded open region $\mathcal{Q} \in \mathbb{R}^3$ completely contains $\mathcal{S}$, $\mathcal{S}$ divides $\mathcal{Q}$ into interior and exterior solids. Then, a ball $\mathcal{B}=\{\boldsymbol{r}, \|\boldsymbol{r}-\boldsymbol{c}_b\| \leq \rho_b\}$ centered at $\boldsymbol{c}_b \in \mathbb{R}^3$ with radius $\beta_b \in \mathbb{R}^+$ is said to be empty if its domain does not contain any points in $\mathcal{P}$, that is, $\mathcal{P} \cap \mathcal{B}=\emptyset$. The MAT of $\mathcal{S}$ is the set of all B maximal empty balls (*medial balls*) and the set $\{\boldsymbol{c}_b, b=\{1,...,B\}]\}$ is the surface medial axis. Therefore, MAT attains an exact representation of $\mathcal{S}$ at the cost of a high computational complexity [3].

Aiming to cope the above issue, MAT is approximated by a subset $\mathcal{J}$ of the Voronoi vertices, known as *poles*. Voronoi vertices are extracted from the Voronoi diagram of the point set $\mathcal{P}$ that splits $\mathcal{P}$ into a set of cells. The i-th cell holds all the points $p \in \mathbb{R}^3$ closer to the i-th entry point p_i than to any other point in $\mathcal{P}$, that is, $\mathcal{Q}_i=\{p_i \in \mathbb{R}^3, d_{pc}(p,p_i) < d_{pc}(p,p_{i'}), \forall i \neq i'\}$, where $d_{pc}(\cdot,\cdot)$ is a predefined distance function. Such a definition produces cells as convex polyhedrons, and their vertices are known as the Voronoi vertices $\boldsymbol{o} \in \mathbb{R}^3$ ($\mathcal{J}=\{\boldsymbol{o}_v, v=\{1,...,V\}\}$). The Voronoi diagram used to approximate the MAT is the Power diagram that is built from the power distance between an ordinary unweighted point p_i and a ball $\mathcal{B}$:

$$d_{pow}(p_i, \mathcal{B}) = \|\boldsymbol{c}-p_i\|_2^2 - \beta^2. \tag{2-5}$$

As a result, the subset of the two-dimensional polygonal faces in the power diagram separating these internal cells from the outer cells forming our output surface.

2.1.4. A fast surface reconstruction method from Hermite points (HRBFQI)

A closed form formulation is proposed to calculate the approximate surface reconstruction solution based on Hermite Radial Basis Functions from scattered data points [22].

The method constructs a signed scalar function by directly combining the normal positions and points without any global operation. The calculation is based on compact support locally and numerically. An automatic scheme is applied to adjust the support sizes of the basic functions; in this way, the errors between the exact solution and the quasi-solution are limited. The formulation provides a surface reconstruction solution on approximate HRBF-based. In addition to being robust to noise.

The HRBF implicits are built upon the theory of Hermite–Birkhoff interpolation with radial basis functions [22]. Given a point cloud $\mathcal{P}$, with their respective set of normals $\mathcal{N} = \{\boldsymbol{n}_i, i = \{1, ..., N\}\}$. The goal is to find an interpolation function h, given the points and normal vectors.

$$h(\boldsymbol{x}) = \sum_{i=1}^{N} \{\alpha_i \vartheta(\boldsymbol{x} - p) - \langle \boldsymbol{l}_i . \nabla(\boldsymbol{x} - p)\rangle\} \tag{2-6}$$

where $\vartheta : \mathbb{R}^3 \to \mathbb{R}$, is determined by a radial basis function $\vartheta(\boldsymbol{x}) = \varphi(\|\boldsymbol{x}\|)$, where $\varphi : [0, \infty) \to \mathbb{R}$, as a univariate function , $\langle \cdot, \cdot \rangle$ denotes the dot-product of two vectors, and ∇ is the gradient operator [22]. Given an approximation of 2-6, the surface of our point cloud can be reconstructed from the set p and its normal $\mathcal{N}$. In the reconstruction of the mesh a polynomial mesh is used, the zero isosurfaces are extracted as a result of the reconstruction of the surface.

2.1.5. Hausdorff Distance

One way to compare the geometrical differences between the two models is to use Hausdorff distance. Given a compact metric space W, we consider the space X of nonempty closed subsets of W:

$$X = \{\mathcal{A} \subset W | \mathcal{A}, \textit{is nonempty, closed, bounded}\} \tag{2-7}$$

Then the Hausdorff metric is defined on pairs of elements in X as follows [13].

$$d_H(\mathcal{A}, \mathcal{E}) = max\{\sup_{g \in \mathcal{A}} m_H(g, \mathcal{E}), \sup_{g \in \mathcal{E}} m_H(g, \mathcal{E})\} \tag{2-8}$$

where $m_H(g, K) : J \times X \to \mathbb{R}$ is given by

$$m_H(g, K) = \inf_{g \in H} d_h(g, V) \tag{2-9}$$

The function m_H represents the "minimum"distance from a point g in J a to punto in K in X, as a subset of J.

2.2. Experimetal Set up

2.2.1. Dataset

Spleen : The basis of the data we use is "Decathlon of medical segmentation", which is publicly available [31]. The database has ten semantic segmentation tasks, which are: Liver Tumors, Brain Tumors, Hippocampus, Lung Tumors, Prostate, Cardiac, Pancreas Tumor,

Colon Cancer, Hepatic Vessels, and Spleen. In our case, we only use images of Spleen, which come from the Memorial Sloan Kettering Cancer Center, where the objective is to segment this structure. The modality of image acquisition is computed tomography. The size of the database is 61 images, 41 for training and 20 for testing. We are only interested in training images since they are segmented. The size of each training image is $512 \times 512 \times 168$ voxeles.

2.2.2. Point Cloud Construction

We get the point cloud as the point lying on the limit of the segmented region extracted from each image. With the coordinates of the boundary voxels of each region, we generate the point cloud. We scale the points in the range $[-1, 1]$, as shown in Figure **2-1**. We can see that the data is scattered since there are only 14 segmented images for the reconstruction.

(a) Slice CT (b) Point cloud Spleen

Figure 2-1: A CT slice for the spleen structure and its point cloud.

2.2.3. Method description

Given the limited number of contours that the point cloud has, shown in the Figure **2-1**b. We need to generate new data that allows us to create a surface that recreates the shape of the spleen. One way to generate new data is to create new contours from existing ones. To deform and analyze the contours, we use the SRVF as described in Section 2.1.2 taking into account that each of them is resized from the resulting one unit length to its original scale. It should be noted that the proposed analysis has a significant advantage over the methods based on the reference points present in state of the art, which generally require a curve registration, either manually or automatically. The Figure **2-2** shows the deformation

correspondences calculated between contours. Since the contours are parallel to an axis, we can work with them as closed flat curves, which decreases the calculations.

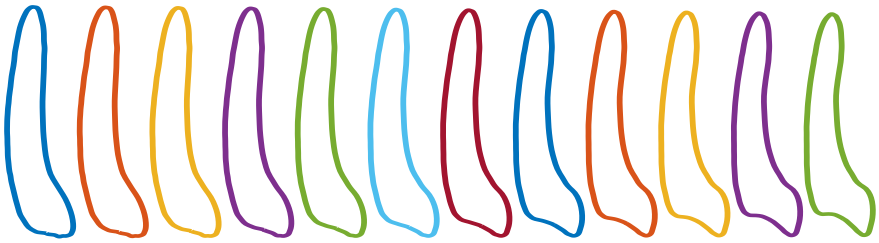


Figure 2-2: Deformation between contours.

The contours that are used as a basis for the deformation, in addition to being rescaled, are aligned and rotated. Therefore, the interpolated curves must be rescaled and realigned. We use Spline Interpolation to define the alignment path, as well as the indexes that allows scaling the contours. In the case of rotation, we use Spherical Spline Quaternion interpolation (Squad) that is described in [1] to rotate the new shapes. We select the contour number that we want to add to the base contours, and we distribute them uniformly. A new point cloud is generated as shown in Figure **2-3**. To generate the mesh, we use a triangulation algorithm available in Python library **PyMesh**[1]. Hence, the resulting surface is a smooth approximation that suitably fits the input contours.

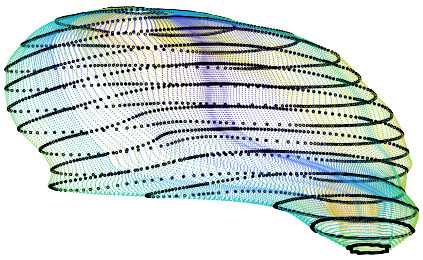
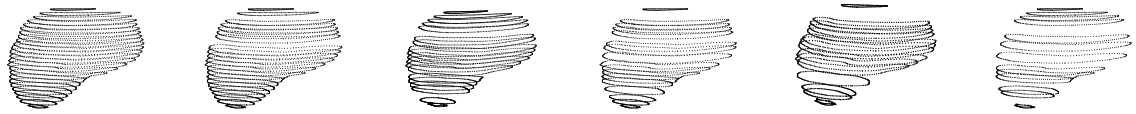


Figure 2-3: The black contours are the initials.

[1]https://pymesh.readthedocs.io/en/latest/index.html

2.2.4. Results and Discussion

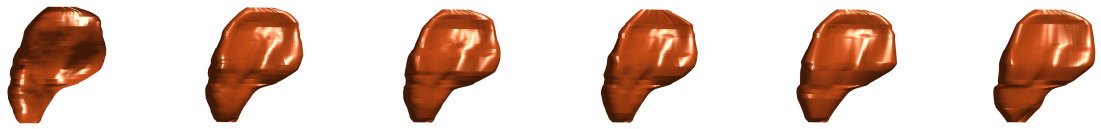
We evaluated our reconstruction scheme in three clouds of different points extracted from the computed tomography of the spleen. Each point cloud is generated with is generated with different numbers of contours from 16 images in the database, the selected images have a number greater than 15 segmented contours. We want to know how our scheme behaves by reducing the number of contours and if it is robust to the lack of data. To this end, we extract the contours as shown in the Figure **2-4**, always maintaining the upper and lower contour. Below is the reconstructed surface for an image of our image set.



(a) All contours (b) 10 % contours (c) 20 % contours (d) 30 % contours (e) 40 % contours (f) 50 % contours

Figure 2-4: Examples of point clouds resulting from randomly removing contours.

Figure **2-5** shows the surface generated by our approach, where we can see that a smooth surface is obtained in all cases.



(a) All contours (b) 10 % contours (c) 20 % contours (d) 30 % contours (e) 40 % contours (f) 50 % contours

Figure 2-5: 3D Reconstruction surface approach.

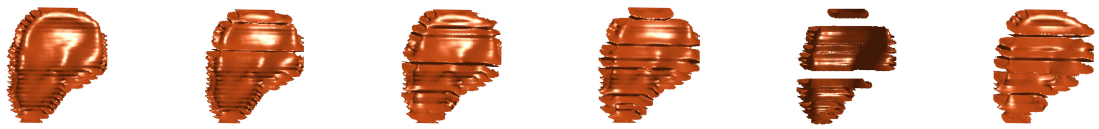
Figure **2-6** shows the reconstruction of the surface of the PC, where we observe that, in certain regions, the surface is less smooth due to the lack of data.

(a) All contours (b) 10 % contours (c) 20 % contours (d) 30 % contours (e) 40 % contours (f) 50 % contours

Figure 2-6: 3D Reconstruction surface PC.

Finally, Figure **2-7** illustrates the HRBFQI 3D reconstruction, evidencing that the algorithm

is not robust to the absence of data.



(a) All contours (b) 10 % contours (c) 20 % contours (d) 30 % contours (e) 40 % contours (f) 50 % contours

Figure 2-7: 3D Reconstruction surface HRBFQI.

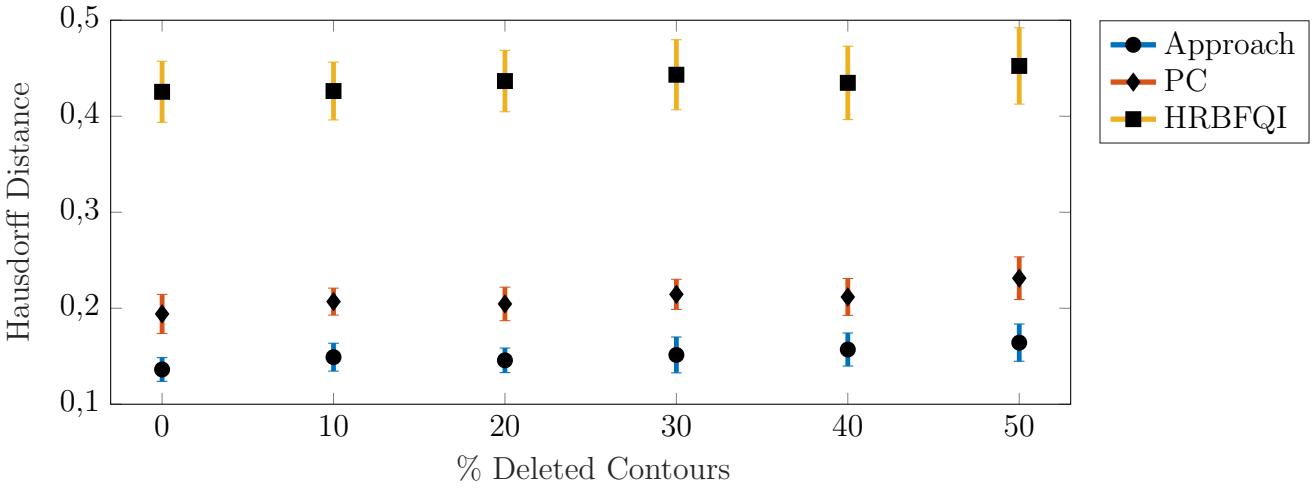


Figure 2-8: Hausdorff distance comparing our approach with two algorithms of the state of the art. The marks indicate the average reconstructed surface of the 16 images and the bars the standard deviation.

Figure **2-8** shows that the average Hausdorff distance for the reconstructed surfaces is shorter for our method, a smaller distance indicates that our reconstruction methodology has a better performance than the methods we compare. We can also observe that the standard deviation in our method is smaller, which shows that the surfaces generated are consistent in most cases. This is because the surface of the spleen is smooth and there are no sudden changes in its shape. Therefore, the new contours generated replace the deleted contours, as well as correspond to the shape of said deleted contours. We can say that our scheme allows reconstructing surfaces from a few contours. The generated surface can serve as a guide to give a general shape of an anatomical structure, but they would not be useful in the diagnosis since we are generating new data that may not be real.

3 Surface reconstruction of point clouds by parts using implicit reconstruction methods

3.1. Materials and Methods

3.1.1. Implicit Surface

The objective of the implicit surface reconstruction is to find a function $f : \mathbb{R}^3 \to \mathbb{R}$ which defines an approximate implicit surface and satisfies $f(\boldsymbol{p}) = 0$, for each p_1 lying on the surface. The implicit surface $\mathcal{S}' = \{\boldsymbol{p}_i \in \mathbb{R}^3 | f(\boldsymbol{p}_i) = 0\}$ can be declared as a zero-level set of the *signed distance function* $f(\boldsymbol{p})$ [37]. The value that the function takes in a point can be defined as the Euclidean Distance to the point closest to the surface. The domain can be seen as a signed distance field. The points on the outside of the surface take positive values, and the points inside the surface take negative values. The values of the implicit function and the implicit surface can be expressed as:

$$\begin{cases} f(\boldsymbol{p}) = 0, & \textit{on-surface} \\ f(\boldsymbol{p}) = +d(\boldsymbol{p}, \boldsymbol{p}'), & \textit{exterior-surface} \\ f(\boldsymbol{p}) = -d(\boldsymbol{p}, \boldsymbol{p}'), & \textit{interior-surface} \end{cases} \tag{3-1}$$

Such that $\boldsymbol{p} \in \mathbb{R}^3$, and $d(\boldsymbol{p}, \boldsymbol{p}')$ is the distance from a point $\boldsymbol{p}$ to a next point $\boldsymbol{p}'$ on the surface $\mathcal{S}$.

3.1.2. Training Point Cloud Generation

As stated in the previous section, the objective is to estimate a function f that reproduces the signed distance in the proximity of the surface. To avoid the trivial solution, points outside the surface are added to the input data, and they are given values other than zero [8]. One way to obtain points off the surface and extend $\mathcal{P}$ is by considering the set of normalized normal directions $\mathbf{n}_i$ of the surface to $\mathcal{S}$ in the points p_i [11]. The calculation of normals is

done with [38]. For each p of $\mathcal{P}$, additional points defined as.

$$\boldsymbol{p}_i^+ = \quad \boldsymbol{p}_i + \lambda \mathbf{n}_i \tag{3-2}$$
$$\boldsymbol{p}_i^- = \quad \boldsymbol{p}_i - \lambda \mathbf{n}_i \quad i = 1, ..., n \tag{3-3}$$

where λ which is given by $\lambda = 0{,}4\varsigma$, $\varsigma = min(\underset{i \neq j}{d}(p_i, p_j))$. The new set of extended training data is given by the union of.

$$P^{ext} = \mathcal{P} \cup \mathcal{P}^+ \cup \mathcal{P}^-, \quad with \quad \mathcal{P}^+ = \{\boldsymbol{p}_i^+ \,|\boldsymbol{p}_i \in \mathcal{P}\}, \quad \mathcal{P}^- = \{\boldsymbol{p}_i^- \,|\boldsymbol{p}_i \in \mathcal{P}\} \tag{3-4}$$

To reduce the computational complexity and the conditioning of the interpolation problem, many suggest that the interpolation fits only a subset $Y = \{y_i \in \mathbb{R}, i = \{1, .., N\}\}$ of $\mathcal{P}^{ext}$[11]. It is assumed that the signed distance function f is 0 in $\mathcal{P}$, 1 in $\mathcal{P}^+$ and -1 on $\mathcal{P}^-$. Therefore, f could only take the following values for the extended data. That you assign to each pair of elements of the input space $\mathcal{P}$,

$$f(\boldsymbol{p}) = 0, \quad if \quad \boldsymbol{p} \in \mathcal{P} \tag{3-5}$$
$$f(\boldsymbol{p}) = +1, \quad if \quad \boldsymbol{p} \in \mathcal{P}^+ \tag{3-6}$$
$$f(\boldsymbol{p}) = -1, \quad if \quad \boldsymbol{p} \in \mathcal{P}^- \tag{3-7}$$

For the interpolation, support vector regression (SVR) is used

3.1.3. Linear SVR

Given a training set $G = \{(\boldsymbol{p}1, y1), ..., (\boldsymbol{p}_m, y_m)\}$, where $\boldsymbol{p}_i \in \mathbb{R}^3$, $y_i \in \mathbb{R}$ and $G \in \mathcal{P}^{ext}$, in which it is assumed that the values and of all the samples of G can be adjusted (or quasi-adjusted) by a linear functional, the objective of task regression is to find the parameters $\omega = \{\omega_1, ..., \omega_m\}$ that allows to define the function:

$$f(p) =< \omega^T, \boldsymbol{p} > +b \tag{3-8}$$

To allow some noise in the training examples, the error condition can be relaxed between the value predicted by the function and the actual cost. To do this, we use the call *loss function ϵ-insensitive*. Characterized by being a linear function with a sensitive area, of width 2ϵ, in which the error is null, and is defined by:

$$L_\epsilon(y, f(x)) = \begin{cases} 0 & if\, |y - f(\boldsymbol{p})| \leq \epsilon \\ |y - f(\boldsymbol{p})| - \epsilon & in\ other\ cases \end{cases} \tag{3-9}$$

The main reason for choosing this function is to allow some dispersion of the solution, so that all the examples that remain confined in the defined tabular region$\pm\epsilon$,will not be considered a support vector,as show in the Figure **3-1**. Given that in practice it is challenging for training

data to be adjusted to the a model with a prediction error equal to zero, the concept of soft margin is used. For this purpose, the slack variables ξ_i^+ and ξ_i^-, allows quantifying the magnitude of the error. Thus, the variable $\xi_i^+ > 0$ when the prediction of the sample $f(\boldsymbol{p}_i)$ is greater than its real value, y_i, in an amount greater than ϵ,i.e. $f(\boldsymbol{p}_i) - yi > \epsilon$. Otherwise, its value will be zero. In the opposite case, similarly $\xi_i^+ < 0$ when the real value of the example is higher than its prediction in an amount greater than ϵ, i.e. $f(\boldsymbol{p}_i) - yi < \epsilon$. Otherwise, its value will be zero. Therefore, the function to be optimized will be:

$$\begin{aligned} \text{mín} \quad & \frac{1}{2}\langle\omega,\omega\rangle + C\sum_{i=1}^{n}(\xi_i^+ + \xi_i^-) \\ s.t \quad & (\langle\omega,\boldsymbol{p}\rangle + b) - y_i - \epsilon - \xi_i^+ \quad \leq 0 \\ & y_j - (\langle\omega,\boldsymbol{p}\rangle + b) - \epsilon - \xi_i^- \quad \leq 0 \\ & \xi_i^+, \xi_i^- \quad \geq \quad 0, \qquad i = \{1,...,m\} \end{aligned} \tag{3-10}$$

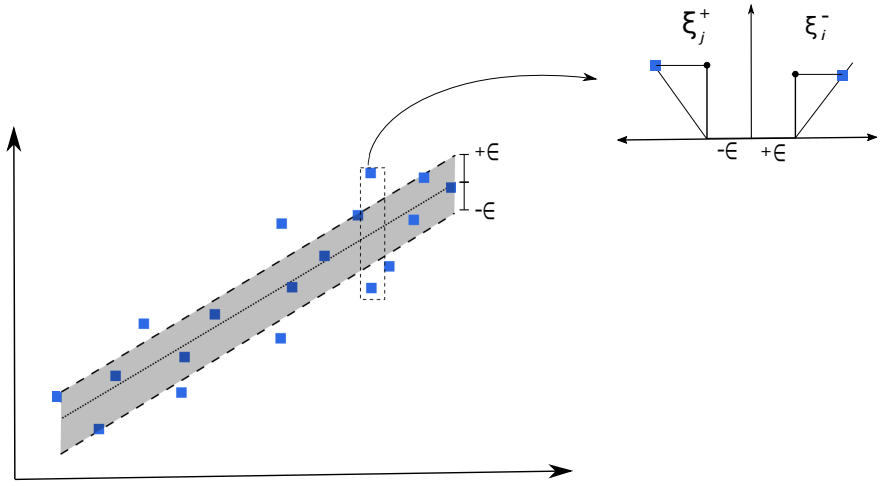


Figure 3-1: SVR, shows the relationship between slack variables, $\xi_i^+ + \xi_i^-$, and the tabular zone ϵ-*insensitive*

In 3-10 the parameter $C > 0$ adjusts the trade-off between the closeness of the solution to the data points and its smoothness [5]. Also determines the amount up to which we tolerate deviations higher than ϵ. So only an immense value of the constant C, in the limit case ($C \to \infty$). We would be considering that the set is correctly represented by our hyperplane predictor ($\xi_i \to 0$).On the contrary, a number too small for C would allow high ξ_j values, that is, we would be admitting a very high number of poorly represented data

The problem of optimization with restrictions belongs to a problem of quadratic programming and is approached through the theory of optimization. The theorem states that an optimization problem, called primal, has a dual form if the function to be optimized and the constraints are strictly convex functions. Under these circumstances, the dual problem allows

us to obtain the solution of the primal problem [32]. The key idea is to construct a Lagrange function from the objective function 3-10 (primal) and the corresponding constraints, by introducing a dual set of variables. It can be shown that this function has a saddle point concerning the primal and dual variables at the solution [32]. As follows,

$$\begin{aligned} L(\omega, b, \xi^+, \xi^-, \alpha^-, \alpha^+, \eta^-, \eta^+) = & \frac{1}{2} < \omega, \omega > + C \sum_{i=1}^{N} (\xi_i^+ + \xi_i^-) - \\ & \sum_{i=1}^{N} \alpha_i^+ (\epsilon + \xi^+{}_i - y_i + < \omega, \boldsymbol{p}_i > + b) - \\ & \sum_{i=1}^{N} \alpha_i^- (\epsilon + \xi_i^- + y_i - < \omega, \boldsymbol{p}_i > - b) - \sum_{i=1}^{N} (\eta^+ \xi_i^+ + \eta^- \xi_i^-) \end{aligned} \tag{3-11}$$

L is the Lagrangian. ω, b, ξ^+, ξ^- are original variable of the problem and $\alpha^-, \alpha^+, \eta^-, \eta^+$ are the dual variables associated with the restrictions. What we are looking for is:

$$\max_{\alpha^-, \alpha^+, \eta^-, \eta^+} \{ \min_{\omega, b, \xi^-, \xi^+} L(\omega, b, \xi^-, \xi^+, \alpha^-, \alpha^+, \eta^-, \eta^+) \} \tag{3-12}$$

We understand that the variables of the dual are positive. We also deduced from the saddle point condition that the partial derivatives concerning the primary variables should be canceled for the optimal [32].

$$\frac{\partial L}{\partial \omega} = \omega + \sum_{i=1}^{N} (\alpha_i^+ - \alpha_i^-) p_1 = 0 \tag{3-13}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N} \alpha_i^- - \sum_{i=1}^{N} \alpha_i^+ = 0 \tag{3-14}$$

$$\frac{\partial L}{\partial \xi_i^+} = C - \alpha_i^+ - \eta_i^+ = 0 \tag{3-15}$$

$$\frac{\partial L}{\partial \xi_i^-} = C - \alpha_i^- - \eta_i^- = 0 \tag{3-16}$$

with the only restriction of $\{\alpha_i^-, \eta_i^-\} \geq 0$. Substituting the previous equations in function 3-11 we obtain:

$$\begin{aligned} \max_{\alpha^-, \alpha^+} \{ & \frac{1}{2} \sum_{i,j=1}^{N} (\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-) < \boldsymbol{p}_i, \boldsymbol{p}_j > + C \sum_{i=1}^{N} (\xi_i^+ + \xi_i^-) - \\ & \sum_{i=1}^{N} \alpha_i^+ (\epsilon + \xi^+{}_i - y_i + < \left(\sum_{i=1}^{N} (\alpha_i^- - \alpha_i^+) \boldsymbol{p}_j \right), \boldsymbol{p}_i > + b) - \\ & \sum_{i=1}^{N} \alpha_i^- (\epsilon + \xi_i^- + y_i - < \left(\sum_{i=1}^{N} (\alpha_i^- - \alpha_i^+) \boldsymbol{p}_j \right), \boldsymbol{p}_i > - b) - \\ & \sum_{i=1}^{N} ((C - \alpha_i^{-+}) \xi_i^+ + (C - \alpha_i^-) \xi_i^-) \} \end{aligned} \tag{3-17}$$

simplifying 3-17, we get the following dual optimization problem:

$$\begin{aligned} \text{máx} \quad & \frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-) < \boldsymbol{p}_i, \boldsymbol{p}_j > -\epsilon\sum_{i=1}^{N}(\alpha_i^+ + \alpha_i^-) + \sum_{i=1}^{N} y_i(\alpha_i^+ - \alpha_i^-) \\ & s.a \quad \sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-) = 0 \\ & \alpha_i^+, \alpha_i^- \in [0, C] \end{aligned} \tag{3-18}$$

The regressor associated with the linear function sought turns out to be

$$f(x) = \sum_{i=1}^{N}(\alpha_i^- - \alpha_i^+) < \boldsymbol{p}_i, \boldsymbol{p}_j > +b \tag{3-19}$$

In this way, the desired function is obtained without depending of the resolution of the problem of the dimension in which the input examples are located and would rely solely on the support vectors.

To complete our regression function we should calculate for this we use the Karush-Kuhn-Tucke(KKT) [32] slack complementarity conditions. These that in the optimal solution the product between the slack variables and the dual constraints must be annulled. That is:

$$\begin{aligned} & \alpha_i^+(\epsilon + \xi_i^+ + y - y_i + < \omega, \boldsymbol{p}_i > +b) = 0 \\ & \alpha_i^-(\epsilon + \xi_i^- + y_i - y + < \omega, \boldsymbol{p}_i > -b) = 0 \end{aligned} \tag{3-20}$$

$$\begin{aligned} & (C - \alpha_i^+)\xi_i^+ = 0 \\ & (C - \alpha_i^-)\xi_i^- = 0 \end{aligned} \tag{3-21}$$

Expression 3-21 allows us to deduct the samples p, y such that $\alpha_i^+ = 0$ or $\alpha_i^- = 0$. They would be out of the $\epsilon - tube$. Being $\alpha_i^+\alpha_i^- = 0$, you can not activate two dual variables associated with the same data at the same time. In the case that $\alpha_i^+\alpha_i^- \in (0, C)$, we would have that the corresponding ξ_i^+, ξ_i^- variables must cancel, so we could clear the value of b^* of the two restrictions:

$$\begin{aligned} & b = y_i - < \omega, \boldsymbol{p}_i > -\epsilon, \quad if \quad \alpha_i^+ \in (0, C) \\ & b = y_i - < \omega, \boldsymbol{p}_i > +\epsilon, \quad if \quad \alpha_i^- \in (0, C) \end{aligned} \tag{3-22}$$

3.1.4. non-linear SVR and Kernelization

We consider the same $G \in \mathcal{P}^{ext}$ data set. Let $\Phi : \mathcal{P} \to \mathcal{F}$ is the transformation function that maps each point of P into a dimensional space, which is called feature space. Therefore, each point of $\mathcal{P}$ corresponds to a point in $\mathcal{F}$, where $\Phi(p) = \{\phi_1(p), ..., \phi_m(p)\}$ and $\forall\phi_i(\boldsymbol{p}), i = \{1, ..., m\}$, such that $\phi_j(\boldsymbol{p})$ it is a non-linear function. $\mathcal{F}$ is a Hilbert space.

Our objective will be to transfer our examples to this new space and to find here the function that best approximates the images of our group. The function we are looking for will be as follows:

$$f(p) = < \omega^T, \phi(\boldsymbol{p}) > +b \tag{3-23}$$

The primal problem is raised as in section 3.1.4. But in this case it does not depend directly on the examples of the set but on its images for a certain function ϕ.

$$\begin{aligned} &\text{mín}\, \frac{1}{2}\langle\omega,\omega\rangle + C\sum_{i=1}^{n}(\xi_i^+ + \xi_i^-) \\ &s.t \quad (\langle\omega,\phi(\boldsymbol{p})\rangle + b) - y_i - \epsilon - \xi_i^+ \quad \leq 0 \\ &\qquad y_i - (\langle\omega,\phi(\boldsymbol{p})\rangle + b) - \epsilon - \xi_i^- \quad \leq 0 \\ &\xi_i^+, \xi_i^- \geq 0, \qquad i = \{1,...,m\} \end{aligned} \tag{3-24}$$

This case, after being transformed by the function ϕ, these examples could have an extremely high dimension, which would complicate in excess the resolution of this primal problem. By analogy we could deduce the dual problem, taking into account the linear case.

$$\begin{aligned} &max \frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-) < \phi(\boldsymbol{p}_i), \phi(\boldsymbol{p}_j) > -\epsilon\sum_{i=1}^{N}(\alpha_i^+ + \alpha_i^-) + \sum_{i=1}^{N} y_i(\alpha_i^+ - \alpha_i^-) \\ &s.a \quad \sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-) = 0 \\ &where \quad \alpha_i^+, \alpha_i^- \in [0, C] \end{aligned} \tag{3-25}$$

We can solve the previous problem without getting to know the function ϕ explicitly. After proposing the dual problem, the objective function depends on the dot product of the images of the data. Therefore we can use a Kernel, instead of calculating the dot products in $\mathcal{F}$. Then we define a function kernel $\mathcal{K} : \mathcal{P} \times \mathcal{P} \to \mathbb{R}$, a real value corresponding to the scalar product of the images of those elements in the new space $\mathcal{F}$:

$$\mathcal{K}(\boldsymbol{p}, \boldsymbol{p}') =< \Phi(\boldsymbol{p}), \Phi(\boldsymbol{p}') > \tag{3-26}$$

where $\Phi : \mathcal{P} \to \mathcal{F}$. By transforming the input space to the feature space can be different from an infinite set of base functions, the question arises as to how to transform the input data, in another space of infinite dimension. The following theorem answers the question:

Aronszajn's theorem. *Compute $\mathcal{K} : \mathcal{P} \times \mathcal{P} \to \mathcal{R}$ which is symmetric and positive semi-definite, there is a Hilbert space and a function $\Phi : \mathcal{P} \to \mathcal{F}$ such that*

$$\mathcal{K}(p, p') =< \Phi(\boldsymbol{p}), \Phi(\boldsymbol{p}') > \forall \boldsymbol{p}, \boldsymbol{p}' \in \mathcal{P} \tag{3-27}$$

An essential consequence of this theorem is that to build a kernel function it is not necessary to do it from a set of functions, $\Phi(\boldsymbol{p}) = \{\phi_1(\boldsymbol{p}), ..., \phi_m(\boldsymbol{p})\}$. Simply define a function that meets the above conditions.

Once you have set the kernel to use, we will solve the problem in the same way:

$$\begin{aligned} \text{máx} \quad & \frac{1}{2}\sum_{i,j=1}^{N}(\alpha_i^+ - \alpha_i^-)(\alpha_j^+ - \alpha_j^-)\mathcal{K}(\boldsymbol{p}_i, \boldsymbol{p}_j) - \epsilon\sum_{i=1}^{N}(\alpha_i^+ + \alpha_i^-) + \sum_{i=1}^{N} y_i(\alpha_i^+ - \alpha_i^-) \\ & s.a \quad \sum_{i=1}^{N}(\alpha_i^+ - \alpha_i^-) = 0 \\ & where \quad \alpha_i^+, \alpha_i^- \in [0, C] \end{aligned} \tag{3-28}$$

and the prediction function:

$$f(p) = \sum_{i=1}^{N}(\alpha_i^- - \alpha_i^+)\mathcal{K}(\boldsymbol{p}_i, \boldsymbol{p}) + b \tag{3-29}$$

The most used kernel is the Gaussian kernel:

$$\mathcal{K}(\boldsymbol{p}_i, \boldsymbol{p}) = G(\|\boldsymbol{p} - \boldsymbol{p}_i\|; \sigma) = exp\left(\frac{\|\boldsymbol{p} - \boldsymbol{p}_i\|^2}{\sigma^2}\right) \tag{3-30}$$

3.2. Experimental Set-Up

3.2.1. Dataset

The proposed approach is evaluated in the MM-WHS: Multi-Modality Whole Heart Segmentation with magnetic resonance imaging (MRI); the whole dataset is freely available online [1]. The database has 120 multimodality whole-heart images from multiple sites, cardiac CT/CTA, and cardiac MRI covering all substructures of the heart. All these clinical data have obtained the approval of institutional ethics and have been anonymized. In this case, we are only interested in the images of cardiac MRI. The data were collected according to the clinical setting in vivo, and the data were used in clinics. So the data had several image qualities, some were of relatively low quality. However, it is necessary to include these data sets to validate the robustness of the algorithms developed when it comes to real clinical use.

MRI data were acquired at St Thomas Hospital and Royal Brompton Hospital, London, United Kingdom, using 3D balanced free precession sequences (b-SSFP), with an acquisition resolution of approximately 2 mm in each direction and they were reconstructed (resampled) in about 1 mm. For training data sets, manual segmentation of the seven entire heart structures is provided [39, 2] left ventricle blood cavity, right ventricle blood cavity, left atrium blood cavity, right atrium blood cavity, the myocardium of the left ventricle, ascending aorta and pulmonary artery.

3.2.2. Point Cloud Construction

We want to reconstruct the myocardium of the left ventricle (MLV). Therefore, we must obtain the myocardial limit of the MRI of the database that has a size of $288 \times 288 \times 133$ voxels. The MLV

[1] http://www.sdspeople.fudan.edu.cn/zhuangxiahai/0/mmwhs/

point cloud is constructed from the edge of the manual segmentations provided by the database, as shown in Figure **3-2**. The coordinates of each point correspond to the location of the edge pixels in the magnetic resonance; the values of the point cloud are scaled between $[-1, 1]$.

(a) Slice MRI (b) Point cloud MLV

Figure 3-2: Point cloud extracted from the myocardium of the left ventricle.

3.2.3. Estimation of the SVR parameters

The most straightforward and most used procedure is to define a mesh over the space of the parameters and select the values to the parameters that provide better estimators of predicted prediction error. The parameter grid this defines by the following values.

$$
\begin{aligned}
C &= 2^{T_1}, \quad T_1 = \{0, 0{,}5, .., 5\} \\
\epsilon &= 10^{T_1}, \quad T_2 = -\{1, 1{,}5, .., 5\} \\
\sigma^2 &= median(d(p_i, p_j)) \times T_3, \quad T_3 = \{0{,}1, 0{,}2, ..., 1\}
\end{aligned}
$$

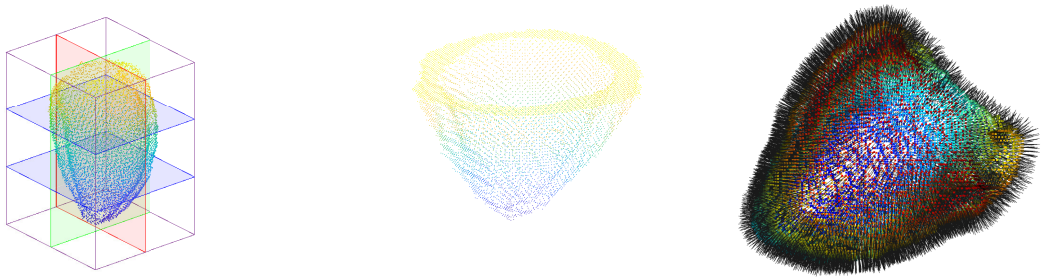
The most used method for estimating the prediction error is cross-validation. This method calculates the error directly when the method, or the prediction function $f(p)$, is applied to a test sample. Denoted by $Err = \mathbb{E}[L(y, f(p)]$,$L(y, f(x))$ was defined, as in the root of the mean square error (RMSE) [5].

If we had a sufficiently large set, we could select a validation set to evaluate our prediction model. Since the collection is often not large enough, we could say that this will not always be possible. In cases where there is not a broad set, we can use cross-validation to evaluate our predictor. When we apply cross-validation, we subdivide our set of k approximate parts of the same size. We use $k-1$ parts to adjust the model and the remaining ones to validate. We perform this same procedure K times, in our case, $K = 10$, selecting each time a different validation set. In each case, RMSE is calculated for each set of parameters, and the settings with the smallest error are chosen to reconstruct the point cloud.

3.2.4. Method description

We are based on the premise that dividing the point cloud into several sections gives us a more accurate reconstruction of the surface. To demonstrate this, the MLV is rebuilt first with all the points and then divides the data. We use planes parallel to the axes of three-dimensional Euclidean space to separate the point clouds, as shown in Figure **3-3**.

We generate the extended point cloud as shown in the section 3.1.2, the normal for each point in the cloud can be seen in Figure **3-3**c. Then, we divide the point cloud, calculate the parameters for the SVR as described in the section. A signed distance function is obtained for each section of the point cloud or the entire cloud, the function is evaluated in a 3D grid of $200 \times 200 \times 200$ and cubes are used Marching [23], to extract the surface, the sections of the surfaces are aligned, if necessary they are scaled according to the original points, to align the points that we use ICP [7]. Finally, we join the surface sections of MLV with boolean mesh operations defined in **PyMesh**, in this case, we use a union.



(a) Planes that cut the point cloud

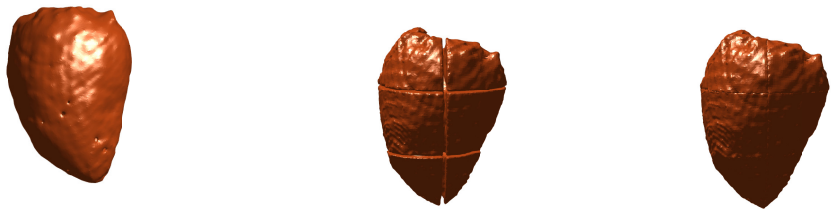
(b) Point cloud section

(c) Normals point cloud

Figure 3-3: Point cloud process schema.

3.2.5. Results and Discussion

By dividing the point cloud into several sections, as shown in Figure **3-4**b, we obtain several advantages. One is that SVRs fit better with each data set. This can be seen since the surface of Figure **3-4**a is softer and the sections of Figure **3-4**b have a more detailed surface. One of the parameters that affect the detail is σ^2 because it affects the extent of the influence of each support vector p in its vicinity. Using Gaussian on a very small scale, it would allow reconstructing the finer details, while a large-scale Kernel would provide a coarser approximation [5].
As seen in the graph **3-5**, if we divide our initial point cloud into several sections, we have a better reconstruction. With more divisions, the distance from Hausdorff decreases as the number of sections increases. Therefore, it can be inferred that the reconstructed surfaces in the points cloud sections are more similar to the original data.



(a) SVR reconstruction (b) Split point cloud (c) Mesh union

Figure 3-4: Surface reconstruction myocardium of the left ventricle.

We can conclude that the use of a single-scale core may not be the best option when the data set is sampled from a non-stationary source that generates data across a uniform variety in some regions and data with rapid variations in others.

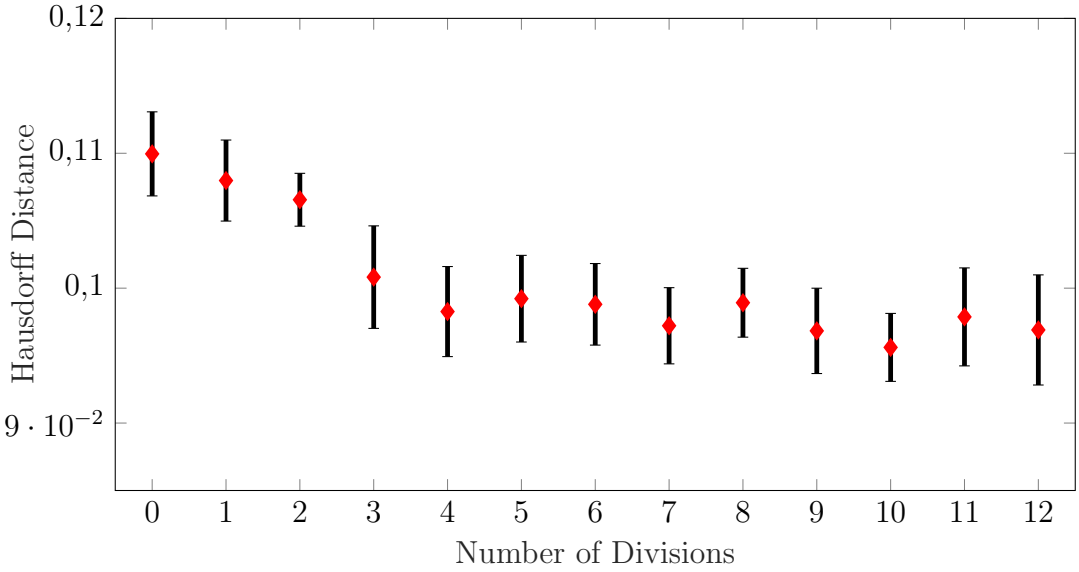


Figure 3-5: Performance of our approach, in 26 reconstructed 3D surface, the red diamond indicates the average of the surface when dividing the initial point cloud, the bar indicates the standard deviation.

4 Bifurcations on cross-sections.

4.1. Materials and Methods

4.1.1. Convex Hull

Convex hull from a set of points with M samples $\mathcal{Z} = \{z_k \in \mathbb{R}^2, k = \{1, 2, ..., M\}\}$, as the intersection of all convex sets containing z [4, 30], its convex $\mathcal{T}$ is given by the expression:

$$\mathcal{T}(z) = \left\{ \sum_{i=1}^{k} \varrho_i z_i | z_i \in Z, \varrho_i \in \mathbb{R}, \varrho_i \geq 0, \sum_{i=1}^{k} \varrho_i = 1 \right\} \tag{4-1}$$

In the case of 2-dimensions, if not all the points are aligned, then its convex envelope corresponds to a convex polygon whose vertices are some of the points of the initial set of points.

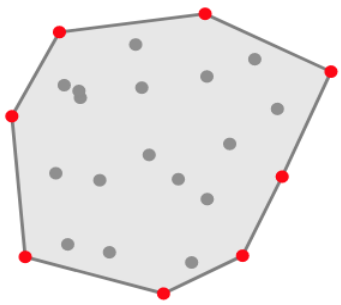


Figure 4-1: The convex hull of a set of points

4.1.2. Delaunay triangulation

A triangulation of a set of points $\mathcal{Z}$, is the decomposition of its convex envelope into a set of triangles, so they can only intersect at their edges, and all $z \in \mathcal{Z}$ is the vertex of at least one of them.

There is no single triangulation for the same set of points. One of the best-known forms of triangles sets of points is through a triangulation of Delaunay. It has the property that each trio of points belongs to a triangle, the circumcircle circumscribed to it has no other point in its interior.

The property is called the Delaunay condition and allows to ensure that the angles the interiors of the triangles should be as large as possible, maximizing the narrowest angle of the triangulation and thus avoiding - when possible - the formation of triangles thin.

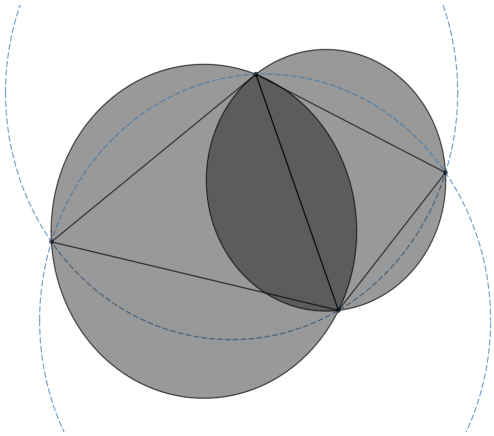


Figure 4-2: The delaunay triangulation of a set of points.

4.1.3. Voronoi Diagram

It is an interpolation method, based on Euclidean distance. It is created by joining points to each other, tracing the bisection of the union segments. The intersections of the bisection determine a set of polynomials in a 2D space, around a control point, in such a way that the perimeter of the generated polygons is equidistant from the neighboring points and designates an area of influence.

Given the Euclidean distance between two points in $\mathbb{R}^2$. like $d(z_k, z_o)$. Let $\mathcal{Z}$, the set of k different points in the plane that are called "sites". The Voronoi diagram de $\mathcal{Z}$ is defined, as the subdivision of a plane in k regions, for each $z \in \mathcal{Z}$. Fulfilling the property of proximity in which a point z_o belongs to the region of a z_i site. If and only if $d(z_i, z_k) < d(z_i, z_k)$ for each $z_k \in \mathcal{Z}$, $k \neq l$. We denote the Voronoi diagram of $\mathcal{Z}$ by $Vor(\mathcal{Z})$. Each region that corresponds to a z_k site will be denoted as $Vr(z_k)$ and it will be called Voronoi region of z_k[30].

The Voronoi region for a site z_k, is constructed from the intersections of the half-planes formed by plotting the bisectors of z_k towards the sites z_l, $l \neq i$. Taking the case where there are only two sites z and z_{or}, the line segment $\overline{zz_o}$ is plotted and the $\overline{z_o z}$. This bisector divides the plane into two half-planes, where the half-plane containing z, represented as $\hbar(z, z_o)$ represents the region of all points closest to z that to z_o, and the half-plane that contains $a(\hbar(z, z_o))$, houses all points closest to z and z_0. According to this, then you can establish in a general way how the Voronoi region is defined for a site p_i[30].

$$Vr(z_k) = \bigcap_{k=1, l \neq k}^{n} \hbar(z_k, z_l) \tag{4-2}$$

$Vr(z_k)$ is composed by the intersection of $n-1$ semiplanes that make up a convex polygonal region that can be open or closed.

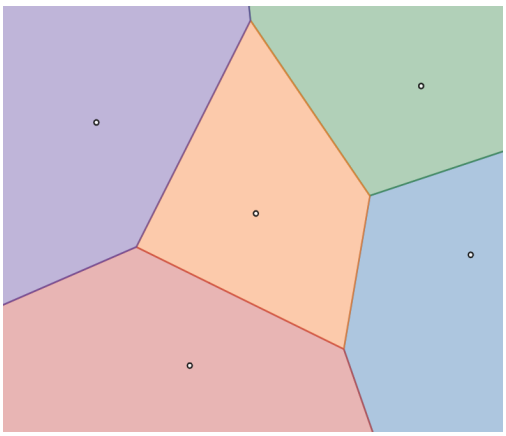


Figure 4-3: The Voronoi Diagram of a set of points

4.1.4. Surface reconstruction from curve networks (SRCN).

Given a set of planes $\mathcal{D} \in \mathbb{R}^3$ possibly non-parallel cross-sections partitioned $\mathcal{D}$ by curve networks labelled 2D regions [21], We desire a surface network with the following properties. The surface network divides the space into labeled 3D regions, which are two regions that share conventional boundaries, have different labels, and the surface interpolates the networks of curves, and the labels of the partitioned 3D regions are consistent with those of the 2D areas of each cross section.

The algorithm of surface reconstruction from curve networks. Follow the divide-and-conquer strategy and consider the partitioning of space into cells by all cross-section planes. The core of the algorithm is the construction of a closed surface net-work within each partitioned cell, from between two parallel planes to within an arbitrary convex space [21].

4.1.5. Topology-constrained surface reconstruction (TCSR)

The input of the algorith is a set of planar cross-secctions, each containig one or multiple curves along the intersection between the plane with the object [40]. These curves divide each cross-section plane into internal and external regions. The cross sections may not be parallel in all cases and the section curves may have an arbitrary shape. The output of the algorithm is a closed surface interpolated between the section curves. The method combines divide and conquer with a global optimization. The arrangement of the cross-sections, which partitions the space into convex cell, is considered.Like each cross-section, the cell boundary is divided by section curves into internal and external regions. The part of the final surface inside each cell is calculated, which we call a tile,each delimited by one or more section curves at the cell boundary.A tile must divide the cell

into internal and external volumes that match the labeling at the cell boundary [40].

The algorithm proceeds in three steps,i)calculate a family of different mosaic topologies within each cell and mark the probability of each topology. ii) Select a tile topology by tge cell so that the topology of the combined surface of all the cells has the gender specified by the user, while the cumulative score is maximized,ii) output a smooth surface that realizes the chosen tile topology within each cell [40].

4.2. Method description

Let be a set of closed of closed and smooth curves on a 2D plane. A base curve, in the plane-1, and multiple curves on the plane-2 being the plane-1 parallel to the plane-2, as shown in Figure **4-4**.

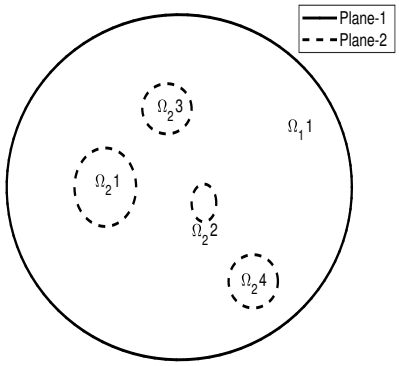
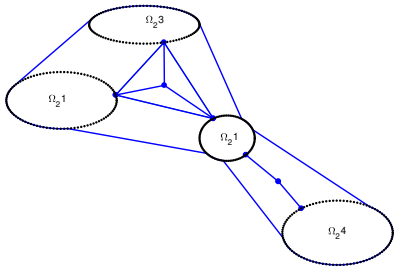


Figure 4-4: We have a curve or contour of origin $\Omega_1 1$, and several objective contours $\Omega_2 m$

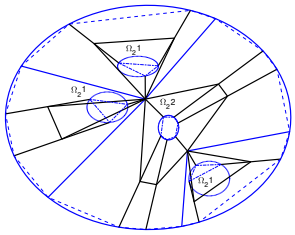
To interconnect Ω_1 with $\Omega_2 m$, we need to generate some control points. It is necessary to establish several points or vertices between planes. We define that the contour of origin must be connected to all the objective contours. For the control points are obtained from the following procedure (extracted from [12]):

As a first step,a set of points called $\mathbf{\Omega_2}$ is created with all the contours of plane 2, for this set we calculate the convex hull (CVH). As well as Delaunay triangulation, then we identify the Voronoi triangles, which are defined as the triangles where their vertices correspond to points in three different contours. We seek to replace the limits of VCH with the sides of the Voronoi triangle that are parallel. We will establish a criterion for this purpose. First, we determine that the limits of VCH are parallel to the sides of the triangles, when we find the corresponding sides, we calculate the opposite internal angle; if this angle is higher than that of a threshold ($\theta > \tau$), the side is removed. The remaining sides of the Voronoi triangle are estabished as new limits of VCH. The remaining sides of the Voronoi triangle. In each Voronoi triangle, the Voronoi vertex is calculated. A line segment is generated between the vertices of the Voronoi triangles and the Voronoi vertices. If two

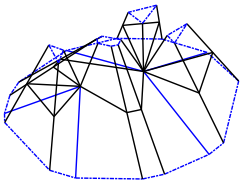
line segments share the same vertex, this vertex is replaced by adjacent vertices, at an appropriately set distance. We generate a polynomial line with the vertices; this polynomial line is projected on the curve of the plane-1 with the help of a algorithm registration points [28]. Subsequently, we create a line segment between the vertices and their projection on the line. Every Vertice Voronoi is projected, taking into account the projection of the vertices of the triangles that are connected to it.



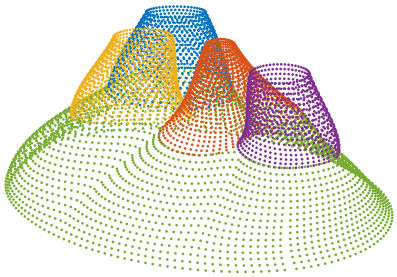
(a) Connection contours plane 2



(b) Interconnection of planes



(c) Control points between planes



(d) Shape Morphing by sessions

Figure 4-5: Basic form of connections between planes.

4.3. Results and Discussion

Based on the control points we divide the region between the planes into several sections to delimit these regions, we generate intermediate closed curves with a B-Spline [35] taking into account the control points as shown in the Figure **4-5**. In Figure **4-5**d, we split the region into five sections, each of these regions starts from the initial contours until the intermediate curves, and in the central zone, the sections are connected. Finally we join the surfaces of each section.

To evaluate the performance of our methodology, evaluate the available Bronchi 3D models is freely available online [1], which are anatomical structures that have bifurcations. We extract a region of

[1] https://www.3dcadbrowser.com/

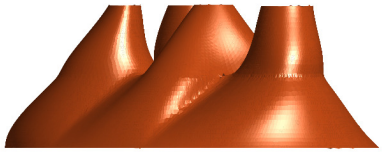
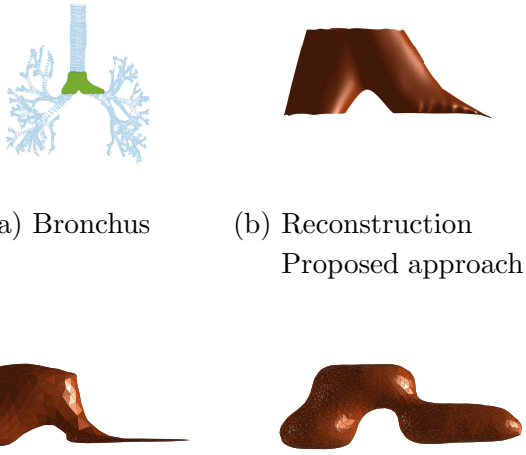


Figure 4-6: Surface reconstruction between cross-section with synthetic data.

the bronchi where we will evaluate our proposal as in Figure **4-7**a.

As it was demonstrated previously when dividing the region into several sections, it is much more comfortable the reconstruction, in this case, it facilitates the interpolation of forms. In Figure **4-6** we observe that our methodology generates a better surface thanks to the added control points, this allows us to create a topology that connects the different contours. The surface generated between the contours is smooth, since assuming another type of surface would cause additional reconstruction problems. Again we can see in the Figure **4-7**c, **4-7**d and **4-7**b that our methodology generates a satisfactory reconstruction of the surface between contours. The upper contour is connected to the lower contours, which shows that our method is capable of generating bifurcations, taking into account only the segmentation of two cross-sections.



(a) Bronchus

(b) Reconstruction Proposed approach

(c) SRNC reconstruction

(d) TCSR reconstruction

Figure 4-7: Comparison surface reconstruction of the bronchus section.

In the Figure **4-8** we can see that our performance is superior to SRNC and TCSR, so we can conclude that our methodology generates a robust surface reconstruction, which takes into account the bifurcations between the cross-sections.

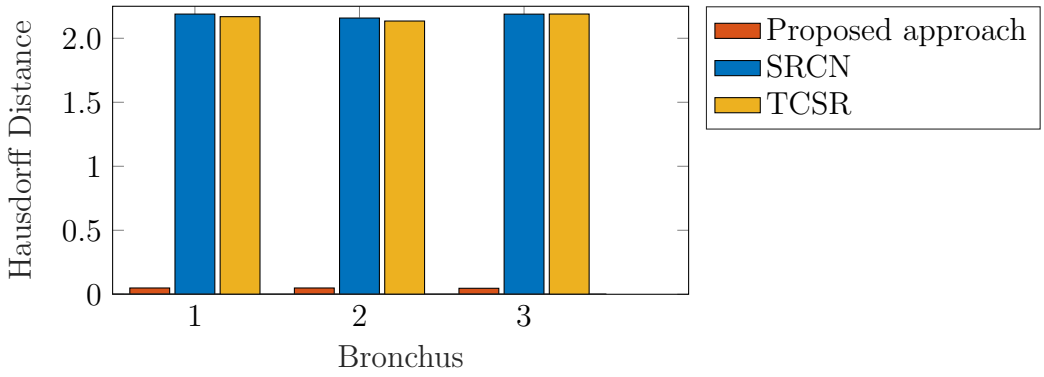


Figure 4-8: Performance of our approach, surface reconstruction in different bronchus section.

5 Conclusion and Future Work

5.1. Conclusion

In this work, we introduce three methodologies that allow us to reconstruct the surfaces of anatomical structures based on medical images. We must bear in mind that the task we want to do. If we are only interested in the shape of the structure, we recommend the first proposal, because when we generate new contours with shape morphing, we introduce information on the surface that may not be reliable, which affects the measurement of the characteristics on the surface. But it gives us the advantage of not investing too much time and effort in stages such as segmentation and having a reconstruction that shows the main characteristics of the surface and form.

In the second methodology proposed, when dividing our data. We calculate the signed distance function better since we can better adjust the SVR parameters as well as decrease the calculation time. When we split, we take into account the local characteristics of the data, which allows us to better adjust the reconstructed surface in certain regions, which contributes to improving the total reconstruction of our data.

Reconstruction between cross-sections is a difficult problem to address, due to the limited information available, in this case. We demonstrate that our methodology that combines base shape, control points and Shape morphing generates good results.

5.2. Future Work

The interpolar forms between contours only take into account two contours, and if we add information from the other contours, we could add information about the structure form in general, which would allow us to have an approximate improvement of the surface.

Method two, can be improved if we take into account local characteristics, such as the curvature of a region or the presence of noise when dividing the data, taking into account that we could modify the mesh size of the parameters, decrease its extension, for example, where the data is soft and increase the size of the grid in very noisy regions.

In the reconstructions of bifurcations, we assume that we start with a single contour, although it is possible that this condition is fulfilled in practice. We have to cover more situations if we want to generalize our methodology. Therefore we must take into account in the foreground we have more than one contour, to achieve this it would be convenient to have information apriori of the form of

the structure, to calculate the connections between contours.

Bibliografía

[1] Quaternions , Interpolation and Animation. In: *Reports* (1998), S. Technical Report DIKU–TR–98/5,40–50. http://dx.doi.org/TechnicalReportDIKU-TR-98/5. – DOI Technical Report DIKU–TR–98/5. ISBN 9781450305839

[2] Challenges and Methodologies of Fully Automatic Whole Heart Segmentation: A Review. In: *Journal of Healthcare Engineering* 4 (2013), Nr. 3, S. 371–408. http://dx.doi.org/10.1260/2040-2295.4.3.371. – DOI 10.1260/2040–2295.4.3.371. – ISSN 2040–2295

[3] Amenta, Nina ; Choi, Sunghee ; Kolluri, Ravi K.: The power crust, unions of balls, and the medial axis transform. In: *Computational Geometry: Theory and Applications* 19 (2001), Nr. 2-3, S. 127–153. http://dx.doi.org/10.1016/S0925-7721(01)00017-7. – DOI 10.1016/S0925–7721(01)00017–7. – ISBN 1581133669

[4] Barber, C. B. ; Dobkin, David P. ; Huhdanpaa, Hannu: The quickhull algorithm for convex hulls. In: *ACM Transactions on Mathematical Software* 22 (1996), Nr. 4, S. 469–483. http://dx.doi.org/10.1145/235815.235821. – DOI 10.1145/235815.235821. – ISSN 00983500

[5] Bellocchio, F ; Ferrari, S ; Piuri, V ; Borghese, N A.: Hierarchical Approach for Multiscale Support Vector Regression. In: *IEEE Transactions on Neural Networks* 23 (2012), Nr. 9, S. 1448–1460. – ISSN 2162–237X

[6] Berger, Matthew ; Tagliasacchi, Andrea ; Seversky, Lee M. ; Alliez, Pierre ; Guennebaud, Gaël ; Levine, Joshua A. ; Sharf, Andrei ; Silva, Claudio T.: A Survey of Surface Reconstruction from Point Clouds. In: *Computer Graphics Forum* 36 (2017), Nr. 1, S. 301–329. http://dx.doi.org/10.1111/cgf.12802. – DOI 10.1111/cgf.12802. – ISBN 1467–8659

[7] Besl, Paul J. ; McKay, Neil D.: Method for registration of 3-D shapes. In: *Sensor Fusion IV: Control Paradigms and Data Structures* Bd. 1611 International Society for Optics and Photonics, 1992, S. 586–607

[8] Carr, J. C. ; Beatson, R. K. ; Cherrie, J. B. ; Mitchell, T. J. ; Fright, W. R. ; McCallum, B. C. ; Evans, T. R.: Reconstruction and Representation of 3D Objects with Radial Basis Functions. In: *Proceedings of the 28th annual conference on Computer graphics and interactive techniques* (2001), 67–76. http://dx.doi.org/10.1145/383259.383266. – DOI 10.1145/383259.383266. – ISBN 158113374X

[9] Chen, Hao ; Dou, Qi ; Yu, Lequan ; Chen, Hao ; Jin, Yueming ; Yang, Xin ; Qin, Jing ; Heng, Pheng-ann: Medical Image Computing and Computer-Assisted Intervention – MICCAI

2013. 8150 (2013), Nr. October. http://dx.doi.org/10.1007/978-3-642-40763-5. – DOI 10.1007/978–3–642–40763–5. ISBN 978–3–642–40762–8

[10] Chen, S. .. ; Lin, W. .. ; Liang, C. .. ; Chen, C. ..: Improvement on dynamic elastic interpolation technique for reconstructing 3-D objects from serial cross sections (biomedical application). In: *IEEE Transactions on Medical Imaging* 9 (1990), March, Nr. 1, S. 71–83. http://dx.doi.org/10.1109/42.52984. – DOI 10.1109/42.52984. – ISSN 0278–0062

[11] Cuomo, Salvatore ; Galletti, Ardelio ; Giunta, Giulio ; Marcellino, Livia: Reconstruction of implicit curves and surfaces via RBF interpolation. In: *Applied Numerical Mathematics* 116 (2017), 157–171. http://dx.doi.org/10.1016/j.apnum.2016.10.016. – DOI 10.1016/j.apnum.2016.10.016. – ISSN 01689274

[12] Ginnis, A. I. ; Kostas, K. V. ; Kaklis, P. D.: Construction of smooth branching surfaces using T-splines. In: *CAD Computer Aided Design* 92 (2017), 22–32. http://dx.doi.org/10.1016/j.cad.2017.06.001. – DOI 10.1016/j.cad.2017.06.001. – ISSN 00104485

[13] Henrikson, Jeff: Completeness and total boundedness of the Hausdorff metric. In: *MIT Undergraduate Journal of Math.* 1 (1999), 69–79. http://dx.doi.org/10.1016/j.biocon.2006.08.005. – DOI 10.1016/j.biocon.2006.08.005. – ISSN 00063207

[14] Holloway, Michelle ; Grimm, Cindy ; Ju, Tao: Template-based surface reconstruction from cross-sections. In: *Computers & Graphics* 58 (2016), S. 84–91

[15] Kazhdan, Michael ; Bolitho, Matthew ; Hoppe, Hugues: Poisson surface reconstruction. In: *Proceedings of the fourth Eurographics symposium on Geometry processing* Bd. 7, 2006

[16] Kolluri, Ravikrishna ; Shewchuk, Jonathan R. ; O'Brien, James F.: Spectral surface reconstruction from noisy point clouds. In: *Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing* ACM, 2004, S. 11–21

[17] Laga, Hamid ; Kurtek, Sebastian ; Srivastava, Anuj ; Miklavcic, Stanley J.: Landmark-free statistical analysis of the shape of plant leaves. In: *Journal of Theoretical Biology* 363 (2014), 41–52. http://dx.doi.org/10.1016/j.jtbi.2014.07.036. – DOI 10.1016/j.jtbi.2014.07.036. – ISSN 10958541

[18] Li, Yangyan ; Wu, Xiaokun ; Chrysathou, Yiorgos ; Sharf, Andrei ; Cohen-Or, Daniel ; Mitra, Niloy J.: Globfit: Consistently fitting primitives by discovering global relations. In: *ACM transactions on graphics (TOG)* 30 (2011), Nr. 4, S. 52

[19] Liang, C. .. ; Lin, W. .. ; Chen, C. ..: Intensity interpolation for reconstructing 3-D medical images from serial cross-sections. In: *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 1988, S. 1389–1390 vol.3

[20] Lin, Chen-Hsuan ; Kong, Chen ; Lucey, Simon: Learning efficient point cloud generation for dense 3D object reconstruction. In: *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018

[21] Liu, Lu ; Bajaj, Chandrajit ; Deasy, Joseph O. ; Low, Daniel A. ; Ju, Tao: Surface reconstruction from non-parallel curve networks. In: *Computer Graphics Forum* Bd. 27 Wiley Online Library, 2008, S. 155–163

[22] Liu, Shengjun ; Wang, Charlie C L. ; Brunnett, Guido ; Wang, Jun: A closed-form formulation of HRBF-based surface reconstruction by approximate solution. In: *Computer-Aided Design* 78 (2016), 147–157. http://dx.doi.org/https://doi.org/10.1016/j.cad.2016.05.001. – DOI https://doi.org/10.1016/j.cad.2016.05.001. – ISSN 0010–4485

[23] Lorensen, William E. ; Cline, Harvey E.: Marching cubes: A high resolution 3D surface construction algorithm. In: *ACM siggraph computer graphics* Bd. 21 ACM, 1987, S. 163–169

[24] Lu, Guoyu ; Ren, Li ; Kolagunda, Abhishek ; Wang, Xiaolong ; Turkbey, Ismail B. ; Choyke, Peter L. ; Kambhamettu, Chandra: Representing 3D shapes based on implicit surface functions learned from RBF neural networks. In: *Journal of Visual Communication and Image Representation* 40 (2016), 852–860. http://dx.doi.org/10.1016/j.jvcir.2016.08.014. – DOI 10.1016/j.jvcir.2016.08.014. – ISSN 10959076

[25] Macedo, Ives ; Gois, Joao P. ; Velho, Luiz: Hermite radial basis functions implicits. In: *Computer Graphics Forum* Bd. 30 Wiley Online Library, 2011, S. 27–42

[26] Majka, Piotr ; Wójcik, Daniel K.: Possum—A Framework for Three-Dimensional Reconstruction of Brain Images from Serial Sections. In: *Neuroinformatics* 14 (2016), Nr. 3, S. 265–278. http://dx.doi.org/10.1007/s12021-015-9286-1. – DOI 10.1007/s12021–015–9286–1. – ISSN 15392791

[27] Marton, Z C. ; Rusu, R B. ; Beetz, M: On fast surface reconstruction methods for large and noisy point clouds. In: *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on* (2009), S. 3218–3223. http://dx.doi.org/10.1109/ROBOT.2009.5152628. – DOI 10.1109/ROBOT.2009.5152628. – ISBN 978–1–4244–2788–8

[28] Myronenko, Andriy ; Song, Xubo: Point set registration: Coherent point drift. In: *IEEE transactions on pattern analysis and machine intelligence* 32 (2010), Nr. 12, S. 2262–2275

[29] Öztireli, A C. ; Guennebaud, Gael ; Gross, Markus: Feature preserving point set surfaces based on non-linear kernel regression. In: *Computer Graphics Forum* Bd. 28 Wiley Online Library, 2009, S. 493–501

[30] Preparata, Franco P. ; Shamos, Michael I.: *Computational geometry: an introduction*. Springer Science & Business Media, 2012

[31] Simpson, Amber L. ; Antonelli, Michela ; Bakas, Spyridon ; Bilello, Michel ; Farahani, Keyvan ; Ginneken, Bram van ; Kopp-Schneider, Annette ; Landman, Bennett A. ; Litjens, Geert ; Menze, Bjoern u. a.: A large annotated medical image dataset for the development and evaluation of segmentation algorithms. In: *arXiv preprint arXiv:1902.09063* (2019)

[32] Smola, Alex J. ; Sc Olkopf, Bernhard: A tutorial on support vector regression *. In: *Statistics and Computing* 14 (2004), S. 199–222. http://dx.doi.org/10.1023/B:STCO.0000035301.49549.88. – DOI 10.1023/B:STCO.0000035301.49549.88. – ISBN 0960–3174

[33] Srivastava, A ; Klassen, E ; Joshi, S H. ; Jermyn, I H.: Shape analysis of elastic curves in euclidean spaces. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (2011), Nr. 7, 1415–1428. http://dx.doi.org/10.1109/TPAMI.2010.184. – DOI 10.1109/TPAMI.2010.184

[34] Ukwatta, Eranga ; Arevalo, Hermenegild ; Rajchl, Martin ; White, James ; Pashakhanloo, Farhad ; Prakosa, Adityo ; Herzka, Daniel A. ; McVeigh, Elliot ; Lardo, Albert C. ; Trayanova, Natalia A. u. a.: Image-based reconstruction of three-dimensional myocardial infarct geometry for patient-specific modeling of cardiac electrophysiology. In: *Medical physics* 42 (2015), Nr. 8, S. 4579–4590

[35] Unser, Michael ; Aldroubi, Akram ; Eden, Murray: B-spline signal processing. I. Theory. In: *IEEE transactions on signal processing* 41 (1993), Nr. 2, S. 821833

[36] You, Haoxuan ; Feng, Yifan ; Ji, Rongrong ; Gao, Yue: Pvnet: A joint convolutional network of point cloud and multi-view for 3d shape recognition. In: *2018 ACM Multimedia Conference on Multimedia Conference* ACM, 2018, S. 1310–1318

[37] Zhong, De Y. ; Wang, Li G. ; Bi, Lin: Implicit surface reconstruction based on generalized radial basis functions interpolant with distinct constraints. In: *Applied Mathematical Modelling* 71 (2019), 408–420. http://dx.doi.org/10.1016/j.apm.2019.02.026. – DOI 10.1016/j.apm.2019.02.026. – ISSN 0307904X

[38] Zhou, Qian-Yi ; Park, Jaesik ; Koltun, Vladlen: Open3D: A Modern Library for 3D Data Processing. In: *arXiv:1801.09847* (2018)

[39] Zhuang, Xiahai ; Shen, Juan: Multi-scale patch and multi-modality atlases for whole heart segmentation of MRI. In: *Medical Image Analysis* 31 (2016), 77–87. http://dx.doi.org/https://doi.org/10.1016/j.media.2016.02.006. – DOI https://doi.org/10.1016/j.media.2016.02.006. – ISSN 1361–8415

[40] Zou, Ming ; Holloway, Michelle ; Carr, Nathan ; Ju, Tao: Topology-constrained surface reconstruction from cross-sections. In: *ACM Transactions on Graphics (TOG)* 34 (2015), Nr. 4, S. 128