

Shape reconstruction from medical images and quality mesh generation via implicit surfaces

J. Peiró^{1,*},[†], L. Formaggia², M. Gazzola^{1,2}, A. Radaelli^{1,3} and V. Rigamonti²

¹*Department of Aeronautics, Imperial College London, South Kensington Campus, London SW7 2AZ, U.K.*

²*MOX, Mathematics Department 'F. Brioschi', Politecnico di Milano, Piazza L. da Vinci 32, 20133 Milano, Italy*

³*Computational Imaging Lab, Department of Technology, Pompeu Fabra University, Passeig de Circumval.lació 8, 08003 Barcelona, Spain*

SUMMARY

The ability of automatically reconstructing physiological shapes, of generating computational meshes, and of calculating flow solutions from medical images is enabling the introduction of computational fluid dynamics (CFD) techniques as an additional tool to aid clinical practice.

This article presents a set of procedures for the shape reconstruction and triangulation of geometries derived from a set of medical images representing planar cross sections of the object. The reconstruction of the shape of the boundary is based on the interpolation of an implicit function through a set of points obtained from the segmentation of the images. This approach is favoured for its ability of smoothly interpolating between sections of different topology. The boundary of the object is an iso-surface of the implicit function that is approximated by a triangulation extracted by the method of marching cubes. The quality of this triangulation is often neither suitable for mesh generation nor for flow solution. We discuss the use of mesh enhancement techniques to maximize the quality of the triangulation together with curvature adaption to optimize mesh resolution.

The proposed methodology is applied to the reconstruction and discretization of two physiological geometries: a femoral by-pass graft and a nasal cavity. Copyright © 2006 John Wiley & Sons, Ltd.

Received 20 February 2006; Revised 17 July 2006; Accepted 6 August 2006

KEY WORDS: mesh generation; shape reconstruction; mesh enhancement; radial basis functions; implicit surfaces; medical image processing

*Correspondence to: J. Peiró, Department of Aeronautics, Imperial College London, South Kensington Campus, London SW7 2AZ, U.K.

[†]E-mail: j.peiro@imperial.ac.uk

Contract/grant sponsor: EU; contract/grant number: HPRN-CT-2002-00270

1. INTRODUCTION

Fluid dynamics plays an important role in the function of many parts of the human body such as, for instance, the cardiovascular and respiratory systems. The ability of reconstructing patient-specific geometries and performing computational flow simulations in these geometries provides a useful tool to aid clinical diagnosis, the planning of surgical interventions and, in the long term, the prediction of disease development.

Non-invasive image acquisition modalities currently employed in medicine, such as computerized tomography (CT) or magnetic resonance angiography (MRA), produce a stack of planar grey-scale images representing cross-sectional cuts of a specific region of the body. These images constitute a three-dimensional map of intensities where the boundary of the region of interest is generally characterized by a sharp gradient of intensity in the images.

The finite resolution of the image acquisition techniques requires some form of interpolation between the images of the stack to reconstruct the missing information. The major difficulty is to obtain a reconstructed shape of adequate smoothness with minimal user intervention. Several techniques have been proposed in the literature to achieve a smooth surface reconstruction from the map of intensities. We could cite, amongst others, the region growing technique [1], the deformable triangulation [2], the level set method [3] and the implicit function [4].

Here we adopt the implicit function approach previously described in Reference [5] where the ideas presented in References [4, 6] were adapted and applied to the reconstruction of a parametric spline representation of the surface. The main reason for the use of an implicit function is its ability of smoothly interpolating sections of complicated geometry, and even changing topology, without the need of user-defined parameters.

The implicit function is defined in terms of radial basis functions (RBF) [7] and interpolated through a set of points on the surface in a manner that will be briefly described in Section 2. The surface is then represented as an iso-contour of the interpolated implicit function. A triangulation of this boundary surface can be obtained in the form of a triangulation by using an efficient implementation [8] of the marching cubes algorithm [9].

This triangulation is suitable for the visualization of the geometry and therefore a useful tool in clinical diagnosis. However, the quality of this triangulation is often poor and thus requires further processing to produce a surface mesh of sufficient quality to be used first for mesh generation and then for flow simulation in complex physiological geometries. This is accomplished by means of standard mesh modification techniques that enhance the quality of the mesh and optimize its resolution through curvature adaptation.

If the triangulation is sufficiently fine to provide a high-fidelity representation of the surface, then it can be used as the reference surface where nodes of improved triangulations will be placed. This is the approach followed for instance in References [1, 10]. However, this high-fidelity triangulation is only a piecewise linear representation of the surface and thus requires the use of averaging procedures to recover surface properties such as point normals and curvatures.

Although the fidelity of the triangulation can be increased by reducing the size of the triangles, this might not always be practical or feasible. On the other hand, the implicit function approach provides a functional description of the surface that permits the geometric enquiries required by the mesh enhancement techniques applied to improve the quality of the triangulation to be carried out to any degree of accuracy.

The outline of the paper is as follows. Section 2 will present the techniques for the processing of medical images and the shape reconstruction via implicit functions leading to an initial triangulation

obtained by the method of marching cubes. Section 3 will discuss a mesh remodelling strategy to improve the mesh quality and optimize the resolution of the triangulation to achieve a surface mesh suitable for mesh generation and flow simulation. Finally, Section 4 will demonstrate the ability of the proposed strategy to produce high-quality meshes for complex physiological geometries and its performance using examples of patient-specific reconstructions of the geometries of a by-pass graft and of a nasal cavity.

2. SHAPE RECONSTRUCTION AND TRIANGULATION VIA IMPLICIT FUNCTIONS

The interpolation of an implicit surface requires the definition of a suitable set of points on the surface of the object and its interior. The implicit surface is then approximated by a triangulation obtained by sampling the function on a Cartesian grid by the marching cubes method [9]. These steps are described in more detail in the following.

2.1. Image segmentation and point selection

Medical images obtained with the different modalities of image acquisition usually come in standard DICOM format as a set of grey-scale images of planar cross sections of the volume of interest. We have developed a user-friendly graphical user interface using the MATLAB image processing toolbox [11] to read the images in DICOM format, to extract the boundary of the object within the cross sections, and to provide a graphical output that will help the user in deciding whether the reconstruction is appropriate.

Given the heterogeneous nature of signal intensity distribution and quality in CT and MRA images due to the presence of artifacts inherent to these imaging techniques, it is generally not feasible to apply methods of fully automatic segmentation to the three-dimensional intensity map. A certain degree of user intervention is always required during the segmentation. This includes, amongst many others, the identification of the vessel section or sections of interest, the reliable detection of small side vessels, the solution of uncertainties due to partial volume effects and the detection of in-plane saturation and signal drop due to the complexity of the flow. In our segmentation strategy, we exploit the two-dimensional nature of the medical data by first applying an automatic segmentation algorithm based on the Otsu method [12] to each image separately. The graphical user interface displays the outcome of the segmentation using visualization formats such as maximum intensity projection (MIP) to assist the user in the investigation of the morphology of the object and the assessment of its suitability. It also provides alternative segmentation methods, such as manual threshold selection, to apply to regions where the automatic segmentation fails.

The segmentation process generates a stack of planar pixelized contours for each section along the stack. We then proceed to interpolate a closed B-spline for each section using a least-squares approach. This is described in detail in Reference [5]. This enables us to accurately generate a finite set of points on the interpolated curve and on an interior curve offset along the normal. These points are used as the constraints for the implicit surface interpolation. An illustration of the various steps described above is shown in Figure 1 for a graft geometry.

2.2. Implicit surface interpolation using radial basis functions

The shape is defined as the zero iso-surface of an implicit function that interpolates a set of points $\{\mathbf{x}_i, i = 1, \dots, N\}$. Here \mathbf{x} denotes the Cartesian coordinates of a point of \mathbb{R}^3 . The implicit function

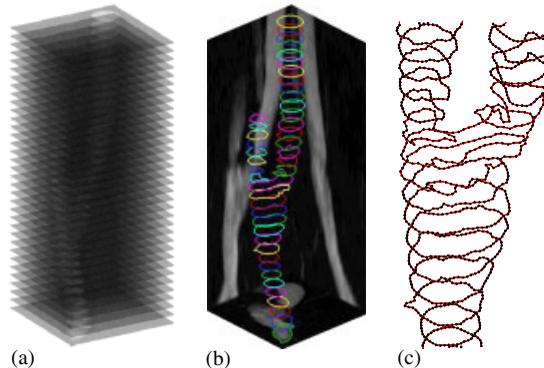


Figure 1. Image processing of a stack of planar sections of a graft: (a) the intensity map is cropped to show the geometry of interest; (b) the planar contours resulting from image segmentation are overlaid to a multiple MIP to help interpretation; and (c) the contours are evenly sampled to generate the interpolation points.

is written in terms of a family of RBF, see for instance References [7, 13], as

$$f(\mathbf{x}) = p(\mathbf{x}) + \sum_{i=1}^N c_i \phi(r_i), \quad r_i = \|\mathbf{x} - \mathbf{x}_i\| \quad (1)$$

where $\|\cdot\|$ denotes the Euclidean norm, $p(\mathbf{x})$ is a polynomial of low degree, $\phi(r_i)$ denotes the radial function centred at the point \mathbf{x}_i and c_i is a constant coefficient. The presence of the polynomial $p(\mathbf{x})$ is required to ensure the well-posedness of the interpolation problem for degenerate cases such as, for instance, a set of interpolation points located on a plane. According to References [4, 6], $p(\mathbf{x})$ can be safely omitted for the interpolation of general three-dimensional shapes as long as a sufficiently large set of interpolation points is used, say $N > 10$. In what follows, and without loss of generality, we will assume $p(\mathbf{x}) = 0$.

The coefficients c_i are calculated so that a set of interpolation conditions of the form $f(\mathbf{x}_j) = d_j$, $j = 1, \dots, N$ are satisfied. This leads to a linear system of equations of the form

$$A \mathbf{c} = \mathbf{d} \quad (2)$$

where A is a symmetric $N \times N$ matrix with components $A_{ij} = \phi(\|\mathbf{x}_i - \mathbf{x}_j\|)$, $\mathbf{c} = [c_1, \dots, c_N]^T$ is the unknown vector of coefficients and $\mathbf{d} = [d_1, \dots, d_N]^T$. The interpolated surface is arbitrarily taken to be the zero iso-contour of the implicit function, i.e. we set $f(\mathbf{x}_i) = 0$ if \mathbf{x}_i is on the surface.

This system (2) has a solution if A is non-singular and a non-trivial solution may be obtained only if $\mathbf{d} \neq \mathbf{0}$. This requires that, at least, one interpolation point must lay away from the iso-surface. These points are often referred to as *offset points*. In our implementation an offset point is generated, for each interpolation point, at a prescribed distance from the interpolation point along the normal to the contour on the plane of the image as depicted in Figure 2. A constant value of d_i , typically $d_i = -1$, is used at the offset points. The sign is taken to be negative so that the condition $f(\mathbf{x}) > 0$ identifies those points on the outside of the surface.

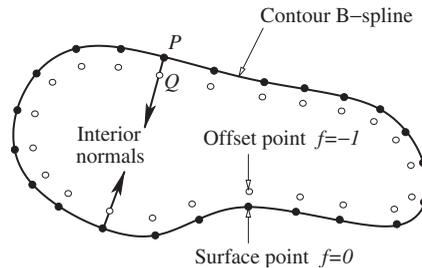


Figure 2. Generation of the offset points. An offset point Q is generated along the normal to the planar contour at the corresponding interpolation point P .

There is considerable freedom in the choice of the function $\phi(r)$ to be used in Equation (1) [7]. Possible choices of RBF are

$$\phi(r) = \begin{cases} r & \text{linear} \\ (r^2 + C^2)^{1/2} & C > 0 \text{ multiquadratic} \\ r^3 & \text{cubic} \\ \exp(-Cr^2) & C > 0 \text{ Gaussian} \\ (r^2 + C^2)^{-1/2} & C > 0 \text{ inverse multiquadric} \end{cases} \quad (3)$$

The matrix A is non-singular if $N \geq 2$ and $\mathbf{x}_i \neq \mathbf{x}_j$ if $i \neq j$, but it is full and badly conditioned in general [7, 13] with the condition number rapidly degrading as the size of the data set increases.

The solution of small data sets ($N < 1000$) can be performed using direct methods such as LU decomposition. The memory requirements for these methods is of order N^2 whilst the computational cost grows like N^3 . This means that to deal with larger data sets, and data sets with $5000 < N < 15000$ are common place in medical reconstructions, we have to resort to the use of iterative methods. Since, depending on the choice of RBF, the matrix is not always positive definite, we have used a matrix-free implementation of the GMRES algorithm [14, 15] which combines good convergence rates with memory requirements of the order of N .

The condition number of the matrix A determines the convergence rate of the iteration and depends on the distribution of the surface constraints and the choice of RBF. The numerical stability of the system is enhanced if a homogeneous constraints distribution is used [16]. It is therefore beneficial to generate an equispaced set of points along the contours whenever possible to improve the GMRES convergence.

The selection of the RBF is a trade-off within the, sometimes conflicting, requirements of achieving a fairly smooth shape reconstruction and good convergence rates. Amongst the possibilities given in Equation (3) we favour the use of either the cubic or linear RBF. The use of cubic RBF leads to a smoother surface, but, on the other hand, linear RBF produce better convergence rates. The choice between the two depends on the resolution and quality of the medical images.

We have also considered two other approaches to ameliorate the convergence properties. The first method is an implementation of the preconditioner based on the domain decomposition method (DDM) proposed in Reference [17]. It has been especially designed to be fully automatic. The

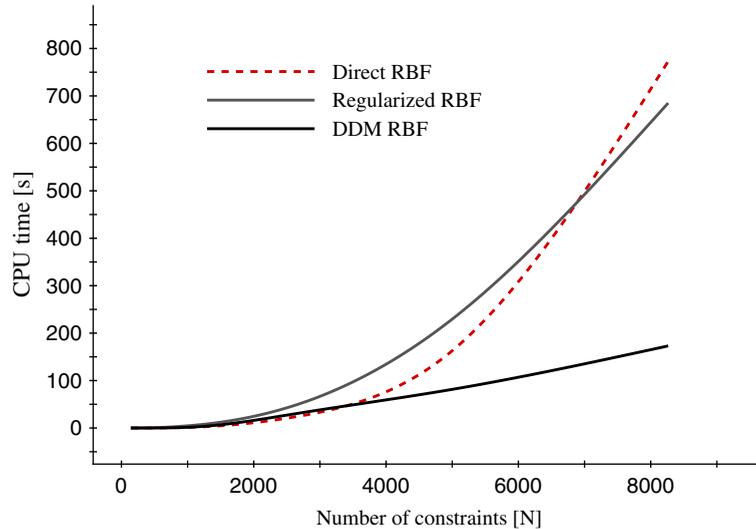


Figure 3. Comparison of the performance of the proposed techniques for improving the condition of the matrix of the implicit function interpolation.

second approach is a regularization technique. Here a parameter $\rho_i \neq 0$, $i = 1, \dots, N$ replaces the zero diagonal element of the matrix A . This acts as a noise reduction factor and shifts the balance between strict satisfaction of the interpolation constraints and the smoothness of the surface. Following analysis of the bounds of the RBF coefficients for several data sets, we found that taking the value of ρ_i to be smaller than the pixel accuracy for surface points and using $\rho_i \geq 2$ for the offset points improves the convergence rate without significantly degrading the accuracy of the surface interpolation. Figure 3 shows a comparison of observed convergence rates. The computations were performed on a 1.9 Gflops Pentium IV processor with 1 Gbyte of memory. Thus far regularization and DDM have been studied separately. The analysis of the stability of a solver implementing both the techniques is under investigation.

2.3. Surface triangulation via marching cubes

For rapid visualization, the surface can be extracted using an implicit ray-tracer or other fast methods of surface extraction. In our case, the triangulation that approximates the iso-surface is obtained using the method of marching cubes originally proposed in Reference [9]. The marching cubes technique subdivides a box containing the iso-surface into a regular lattice of small cubes or voxels. The size of the cubes used is typically of the order of the pixel size in the medical images. The implicit function is evaluated at the vertices of the cubes. If the value of f at the vertices changes sign, the implicit surface crosses the cube. The intersection points of the surface with the sides of the cubes are calculated and a polygonal approximation reconstructed locally for each cube. The union of all polygons thus found forms a polygonal surface that approximates the analytical one.

The evaluation of the implicit function on a pre-defined set of voxels is very expensive as it requires $O(N^3)$ calculations where N is the number of vertices in the grid. A more efficient

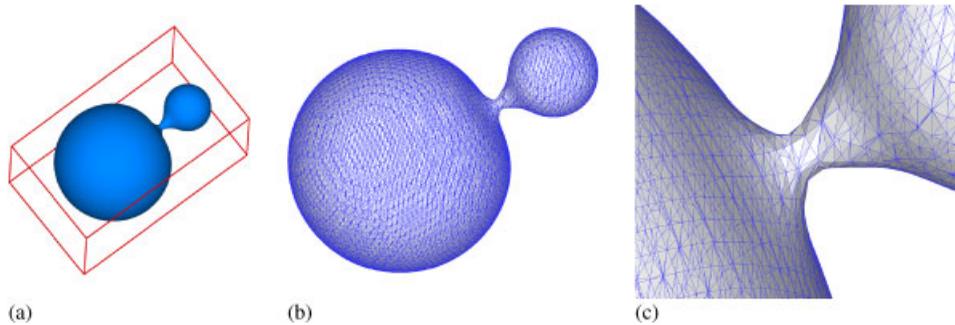


Figure 4. Surface reconstruction of a dumbbell: (a) representation as an iso-surface of the implicit function evaluated in a lattice of voxels (only the lattice's boundary is shown); (b) marching cubes triangulation; and (c) enlargement near the blending surface.

implementation of the algorithm has been proposed in References [6, 8]. This is a continuation method that, starting from a point on the surface, searches the neighbour voxels to the surface point and creates polygons within each cube intersected by the surface. The process is repeated for each new surface point found and, eventually, covers the entire iso-surface defined by the implicit function. An optimal efficiency of the algorithm is obtained by adopting a fast space search based on an octree data structure.

The surface triangulations produced by the various implementations of the marching cubes algorithm are appropriate for surface visualization but their quality is often unsuitable for computational fluid dynamics (CFD) simulations because they might not be watertight or contain very badly distorted elements.

Here we use the freeware C++ implementation [18] of the marching cubes algorithm proposed by Bloomenthal in Reference [8]. The processing speed of this algorithm is satisfactory and it produces watertight triangulations if a suitable resolution for the interpolation grid is chosen. This requires a grid size of the order of 4 pixels or smaller. This is verified through a simple consistency check based on the well-known Euler's formula for polyhedra [19] as follows. The number of vertices, V , and triangles, T , in a triangulation of a single closed compact surface which is topologically equivalent to a sphere with n handles ($n = 0, 1, 2, \dots$) should satisfy the relation $2V - T = 4(1 - n)$. The parameter n is sometimes referred to as the genus of the surface and it is a topological invariant. In the examples considered here we have always $n = 0$. This is, of course, only a necessary condition for the triangulation to be watertight and it is used to discard those triangulations that do not satisfy it. More pathological cases of invalid triangulations are identified by visual inspection, through a calculation of the volume as an integral over the triangulation as described in Reference [20] or, ultimately, by means of geometrical consistency checks during the generation of a tetrahedral volume mesh.

The presence of badly shaped elements is inherent to triangulations produced by the marching cubes method. Since distorted elements often degrade the accuracy of CFD simulations, the quality of the triangulation must be improved before it can be used to generate a volume mesh suitable for CFD.

Figure 4 shows an example of the surface reconstruction of a geometric dumbbell configuration formed by two spheres connected by a smooth blending surface of variable curvature. The implicit

surface is shown in Figure 4(a). Figures 4(b) and (c) depict the marching cubes triangulation and an enlargement near the blending region, respectively. The triangulation contains 47 148 triangles. This example will be used to illustrate the performance of the mesh optimization techniques to be discussed in the next section. The assessment of its quality will be postponed to a later section since we must define a suitable measure of mesh quality first. This will be done in Section 3.3.

It should be noted that the interpolated implicit surface is closed. This means that the mesh generation for open surfaces will require an additional step to trim the unwanted parts of the interpolated surface. This post-processing step is relatively simple to perform and requires minimal user intervention.

3. MESH OPTIMIZATION PROCEDURE

Starting from the triangulation produced by the method of marching cubes, we propose to apply a sequence of local operations that modify the topology and geometry of the triangulation to either improve its quality or to increase or decrease its resolution. The ultimate goal is to produce an optimal triangulation in terms of both mesh quality and surface representation.

To achieve this, we have employed well-established mesh modification techniques. Improvements in mesh quality are obtained by the use of mesh smoothing and side swapping. Control of mesh resolution is accomplished through mesh refinement via side or triangle splitting, or mesh derefinement via side collapsing. These techniques are used in combination with a curvature-based metric to define the desired distribution of mesh size on the surface.

The key to produce an efficient strategy for remodelling the initial mesh is to minimize the number of geometric interrogations to the implicit surface as the evaluation of the function and its derivatives is expensive and other operations often require the solution of a non-linear equation.

3.1. Geometrical properties of the implicit surface

The availability of an analytical definition of the surface as the zero iso-contour of an implicit function, i.e. $f(\mathbf{x}) = 0$ in Equation (1), permits the exact evaluation of geometrical parameters that are required at the various steps of the optimization procedure. The derivation of the expressions for the main geometrical properties of implicit surfaces can be found in Reference [21]. Here we just include these expressions relevant to the mesh improvement methods to be described in later sections.

The normal \mathbf{N} to the surface is given by

$$\mathbf{N} = \frac{\nabla f}{\|\nabla f\|} \quad (4)$$

The normal curvature κ at a point along a curve $\mathbf{X}(s)$ on the surface parametrized in terms of the arc-length s is given by

$$\kappa = \|\ddot{\mathbf{X}}\| = \frac{|\dot{\mathbf{X}}^T \mathbf{H} \dot{\mathbf{X}}|}{\|\nabla f\|} \quad (5)$$

where \mathbf{H} denotes the Hessian of the function f , i.e. the 3×3 matrix of second derivatives of f . The principal curvatures correspond to the maximum and minimum values of κ . Their calculation reduces to the problem of finding the eigenvalues and eigenvectors of the matrix

$$\mathbf{G} = (\mathbf{I} - \mathbf{N}\mathbf{N}^T)\mathbf{H} \quad (6)$$

The normal \mathbf{N} is an eigenvector of \mathbf{G} , the other two eigenvectors are the principal directions in the tangent plane and the principal curvatures are their corresponding eigenvalues.

The computation of the curvature in this manner is expensive. To reduce the number of evaluations required, an approximate expression of the curvature is obtained from the triangulation as proposed in Reference [22]. The approximate normal curvature, $\hat{\kappa}$, at a point \mathbf{x}_i of the triangulation is calculated as

$$\hat{\kappa}\mathbf{N} = \frac{1}{4A} \sum_{j=1}^{N_i} (\cot \alpha_j + \cot \beta_j)(\mathbf{x}_i - \mathbf{x}_j) \quad (7)$$

where the index j represents those vertices in the triangulation sharing a side with vertex i , N_i is the number of the triangles sharing the vertex i and the sum of their areas is denoted by A . The angles α_j and β_j are the two angles, opposite to the side ij , of the two triangles sharing it.

This expression is considerably cheaper to calculate than solving the eigenproblem for the matrix \mathbf{G} given by (6) but it is not very accurate, especially for coarse triangulations. Here it is used only to identify these points where the curvature might be large, then the exact expression is employed only for these points.

3.2. Projection of a point onto the implicit surface

In the process of modifying the mesh towards improving its quality we will introduce points which are close to but not necessarily on the implicit surface. To project the newly generated points onto the surface we have adopted an algorithm proposed in Reference [21] which only requires gradient evaluations.

This procedure iteratively determines the projection of the point on the surface by first moving along the gradient of the implicit function f towards the tangent plane and then by locally approximating the surface $f = 0$ by a tangent parabola in which the final point in the previous step is projected. These steps are sequentially applied until the projection along the normal is found. A more detailed description and a pseudo-code for the implementation of the algorithm is provided in Reference [21].

This algorithm will be employed to project points on implicit surfaces but it is also useful to assess how accurately the triangulation approximates the iso-contour $f = 0$. The algorithm is reasonably robust for as long as the point to be projected is not very far from the surface in the neighbourhood of concave regions of high curvature.

3.3. Assessment of mesh quality

Various indices can be defined to indicate how good the quality of a triangle is. An overview of these is given in Reference [23]. The majority of them are based on the assumption that the best quality triangle is the equilateral. If the triangulation is generated to comply with a mesh metric, then the quality index must be computed in a suitable normalized space as described in References [20, 24]. In essence, the metric represents a mapping from the physical space to a normalized

space where the characteristic length of the transformed sides is approximately one. Here we have adopted a quality index, denoted by Q , of the form

$$Q = \frac{2r}{R}, \quad 0 \leq Q \leq 1 \quad (8)$$

where R and r denote the radii of the circumscribed and inscribed circles to the triangle, respectively. The index has been normalized so that $Q = 1$ for an equilateral triangle. Values of Q close to zero indicate poor quality triangles with the limiting case $Q = 0$ corresponding to a zero-area triangle.

The quality index Q can be simply evaluated as

$$Q = \frac{8(p-a)(p-b)(p-c)}{abc}, \quad p = \frac{a+b+c}{2} \quad (9)$$

where a , b and c represent the lengths of the sides of the triangle.

3.4. Mesh smoothing

To improve the quality of the triangulation we use the non-shrinking low-pass Laplacian filter proposed in Reference [25]. This proceeds in two steps. In the first step the coordinates \mathbf{x}_i of a vertex i on the surface triangulation are moved to a new position, \mathbf{x}_i^* given by

$$\mathbf{x}_i^* = \mathbf{x}_i + \lambda \sum_{j=1}^{N_i} w_{ij} (\mathbf{x}_j - \mathbf{x}_i); \quad 0 < \lambda < 1 \quad (10)$$

where N_i is the number of vertices in the triangulation connected with the vertex i and the weights w_{ij} are positive numbers that satisfy $\sum_{j=1}^{N_i} w_{ij} = 1$. Different weights can be used without altering the smoothing qualities of the algorithm. Here we have used constant weights. The application of the smoothing step (10) improves the quality of the triangulation but it moves the point towards the centre of mass of the triangles surrounding the point. As a result, it shrinks the convex regions of the surface.

A second anti-smoothing step is applied to compensate for the shrinkage of the surface. This is given by

$$\mathbf{x}'_i = \mathbf{x}_i^* + \mu \sum_{j=1}^{N_i} w_{ij} (\mathbf{x}_j - \mathbf{x}_i); \quad \mu < -\lambda \quad (11)$$

The smoothing parameters μ and λ are related by $1/\mu = k - (1/\lambda)$ where $k > 0$ is a measure of the bandwidth of the un-smoothed low frequencies [25].

The smoothing of the triangulation is achieved by an alternating sequence of a few smoothing and anti-smoothing steps followed by projection of the new points onto the implicit surface. The anti-smoothing step is not strictly required in this context but it improves the efficiency of the projection step by providing an initial guess which is closer to the surface. The amount of smoothing applied depends on the values of k and λ and the number of iterations. Here we have used $k = 0.01$ and $\lambda = 0.6$ and applied up to 20 smoothing iterations. Since there is no guarantee that the application of this smoothing technique will not generate badly distorted triangles, we do not move points that substantially diminish the quality of the new triangulation.

Figure 5 illustrates the effect of smoothing on the marching cubes triangulation of the dumbbell geometry shown in Figure 4(a). Figures 4(b) and (c) show frequency graphs of mesh quality for

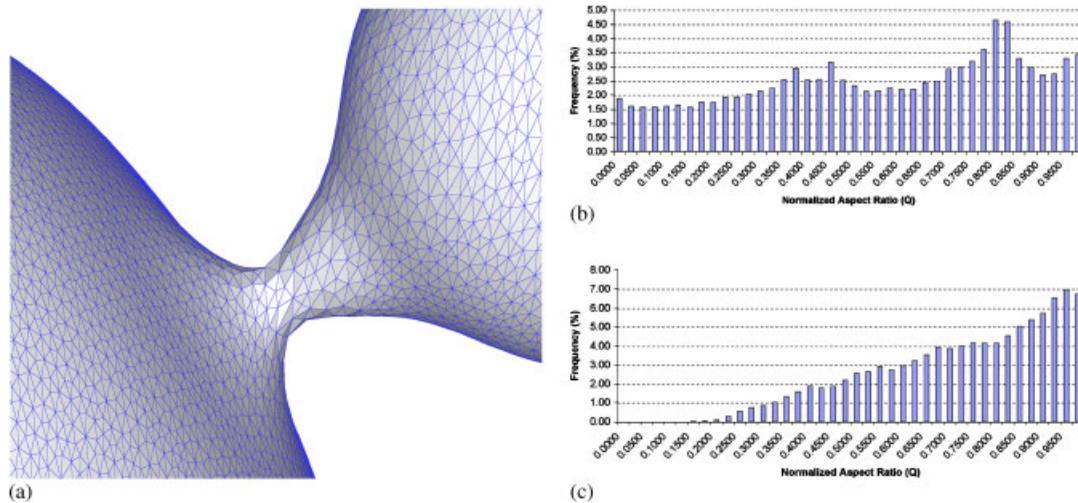


Figure 5. Smoothing applied to the marching cubes triangulation of the dumbbell geometry: (a) enlargement of the mesh near the neck; (b) mesh quality statistics for the marching cubes triangulation depicted in Figures 4(b) and (c); and (c) mesh quality statistics for the smoothed triangulation.

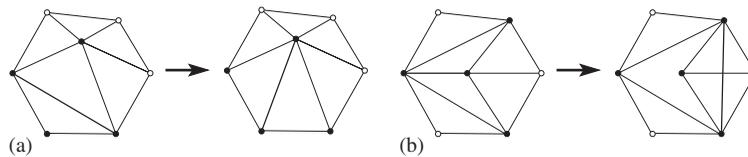


Figure 6. An illustration of the side swapping operation: (a) valid swap; and (b) invalid swap.

the marching cubes triangulation and the smoothed mesh respectively. It can be observed that the distribution of quality greatly improves after smoothing with most triangles having $Q > 0.2$. A triangle can be considered to be of reasonably good quality if $Q > 0.5$, therefore there is still room for improvement.

3.5. Side swapping

This operation considers the quadrilateral formed by two adjacent triangles and substitutes the shared side by the other diagonal of the quadrilateral if it improves the mesh. This is illustrated in Figure 6(a). This operation could lead to an invalid new configuration if any of the angles adjacent to the common side is obtuse as shown in Figure 6(b). There are several strategies that could be adopted here to improve the mesh.

One possibility is to swap sides to achieve a regular distribution of sides per vertex. The criterion uses the concept of *degree* of a vertex defined as the number of sides that share it. Side swapping is performed if the new configuration is valid and it leads to a more even distribution of vertex degrees in the new configuration. This approach is particularly useful after refinement by triangle splitting which will be discussed in Section 3.8.

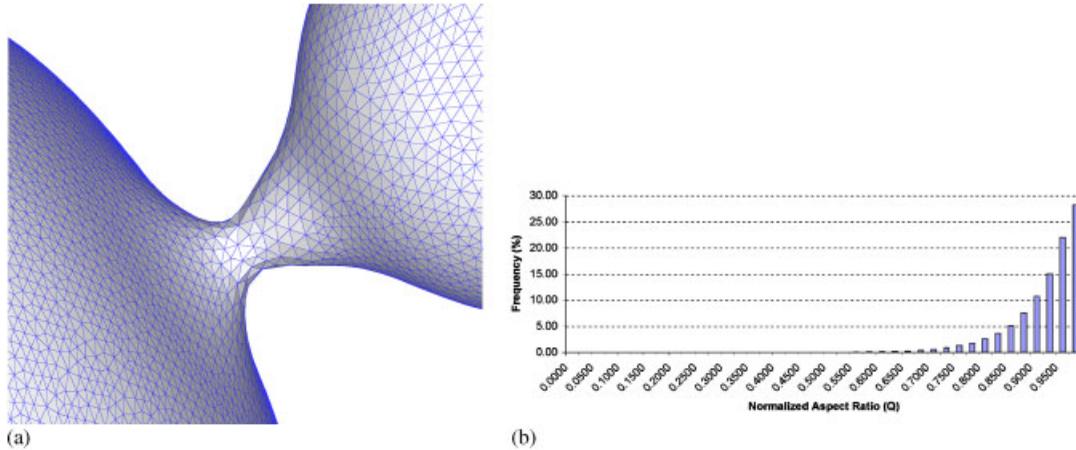


Figure 7. Swapping according to the mesh quality index Q applied to the smoothed triangulation of the dumbbell geometry: (a) enlargement near the neck; and (b) mesh quality statistics.

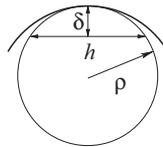


Figure 8. Approximation of the error δ arising from a piecewise linear representation of the geometry.

Alternatively, we simply swap sides if this leads to a higher value of mesh quality evaluated using the quality index (8) for the new configuration. Figure 7(a) illustrates the effect of this type of swapping on the smoothed triangulation of the dumbbell geometry shown in Figure 5. The improvement on the mesh quality is evident from the frequency graph of Q that shows a clear shift towards higher values of Q . Most of the triangles have a mesh quality index $Q > 0.6$, a considerable improvement with respect to the previous mesh.

3.6. Curvature based mesh optimization

An efficient mesh generation should improve the accuracy of the approximation of the implicit surface by ensuring that the mesh resolution matches the surface by using smaller elements in the regions of high curvature and larger elements where it is small.

Following Reference [24] we adopt a mesh metric to characterize the variation of triangle size and shape. The size h of an element in the mesh is selected to obtain a piecewise linear approximation that is sufficiently close to the curved surface according to some measure of the geometric error. Here we define this error as the ratio of the distance between the mesh and the surface, δ , over the curvature radius ρ . This can be approximately calculated in the following manner. Considering a curve on the surface and following the notation of Figure 8, the curve is locally approximated by a circle of radius ρ , the radius of curvature. We assume that the mesh spacing can be represented by

a chord of length h in the circle. This allows us to evaluate δ as

$$\delta = \rho \left(1 - \sqrt{1 - h^2/4\rho^2} \right) \quad (12)$$

The mesh size h is selected by requiring that the geometric error satisfies $\delta/\rho < \varepsilon$, where ε is a user-defined threshold. As a guide, a value of $\varepsilon = 0.01$ results in a division of a circle into 26 segments.

In the general case, we impose the same spacing restrictions along the principal directions \mathbf{d}_1 and \mathbf{d}_2 with radii of curvature ρ_1 and ρ_2 , respectively, by requiring that

$$\frac{\delta_1}{\rho_1} < \varepsilon, \quad \frac{\delta_2}{\rho_2} < \varepsilon \quad (13)$$

This means that the spacings h_1 and h_2 along the principal directions are given by

$$h_1 < 2\rho_1\sqrt{\varepsilon(2-\varepsilon)}, \quad h_2 < 2\rho_2\sqrt{\varepsilon(2-\varepsilon)} \quad (14)$$

The mesh metric can now be written as

$$\mathbf{M} = \mathbf{D} \begin{bmatrix} 1 & & \\ \frac{1}{h_1^2} & 0 & 0 \\ 0 & \frac{1}{h_2^2} & 0 \\ 0 & 0 & \nu \end{bmatrix} \mathbf{D}^T \quad (15)$$

where the matrix \mathbf{D} represents a rotation to the principal directions of the surface given by $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \mathbf{N}]$. The entry ν has no influence on the surface mesh but can be used to control the mesh spacing in the direction normal to the boundary. This permits to control the generation of three-dimensional meshes near the surface, an important aspect in modelling flow in boundary layers.

3.7. Mesh coarsening

The reduction of the size of the mesh is accomplished by eliminating triangles through side collapsing. This operation selects a side of the triangulation if its normalized length, calculated according to the metric (15), is smaller than a user-specified value taken here to be 0.3. The side is collapsed onto a point and the two adjacent elements are then eliminated from the mesh. This is illustrated in Figure 9(a).

Collapsing the side to its midpoint might lead to a better mesh quality but requires the additional cost of using the point projection algorithm. Therefore, we have chosen alternatively to collapse the side to the end point that leads to a better mesh quality of the two. This approach however will not work as well as collapsing to the mid point if the meshes are very coarse.

The collapse of a side could potentially result in the creation of a mesh of much inferior quality or, worse, topologically invalid as illustrated in Figure 9(b). To avoid this we check the quality of the triangles surrounding the point representing the collapsed side and if the new configuration is invalid or the deterioration in quality with respect to the previous configuration is large, say the

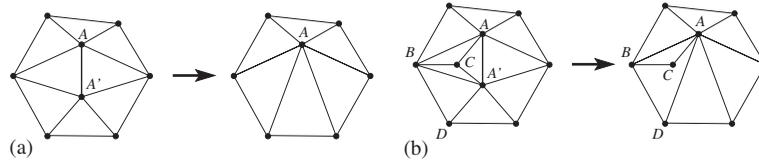


Figure 9. Side collapsing operation: (a) a valid side collapse; and (b) invalid side collapse where the triangle ABC appears twice (with opposite orientations) and also overlaps with triangle ABD in the transformed configuration.

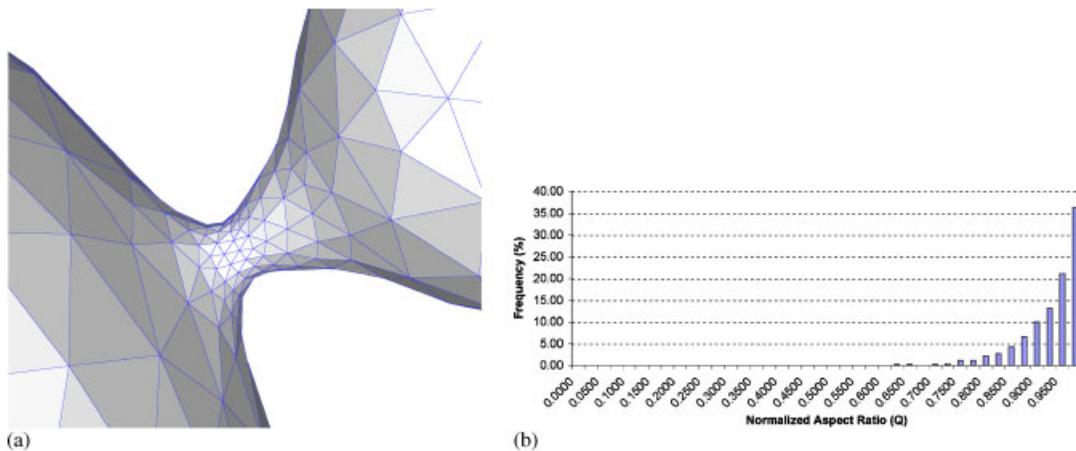


Figure 10. Mesh coarsening: (a) enlargement of the mesh near the neck; and (b) mesh quality statistics.

new value is less than half the previous value, the operation is not performed for this side at this stage.

Figure 10 shows the effect of coarsening the mesh according to the curvature-based metric given by Equation (15). The starting mesh is the enhanced mesh depicted in Figure 7. The number of triangles has been decimated to only 720. The quality of the mesh is very good as shown in Figure 10(b) that presents the mesh quality statistics. The majority of triangles have a quality index $Q > 0.6$.

3.8. Mesh refinement

We have considered two alternative methods to increase the resolution of the mesh: side splitting or triangle splitting. In the side-splitting method an additional vertex is added at the midpoint of the side. The triangle-splitting method adds a new vertex at the centre of mass of the triangle. Splitting is performed if the normalized length of the side or the maximum side length of the triangle are larger than a user-specified threshold taken to be 1.5. These procedures are illustrated in Figure 11. The new vertex is then projected into the surface using the procedure described in Section 3.2.

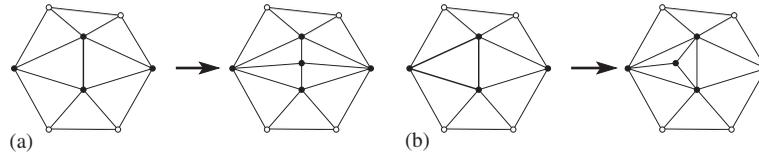


Figure 11. Mesh refinement: (a) side splitting; and (b) triangle splitting.

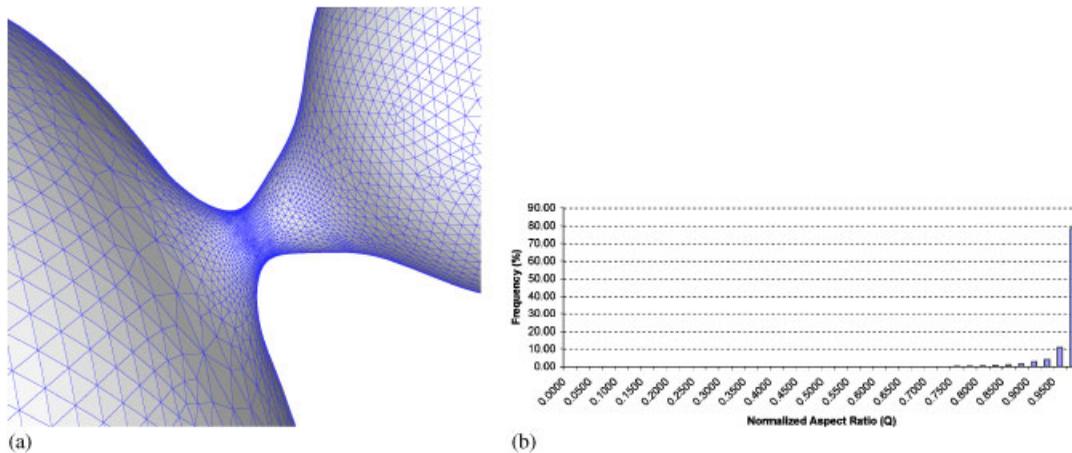


Figure 12. Mesh refinement: (a) enlargement of the mesh near the neck; and (b) mesh statistics.

The strategy of splitting triangles tends to generate fewer points but produces a better quality mesh after swapping and smoothing than the alternative side splitting. The only caveat is that, if coarsening is to be applied at a later stage, we must ensure that there are no vertices with degree equal to three to avoid the type of problems highlighted in Section 3.5. This problem can be ameliorated through swapping based on vertex degree. However, these techniques are not incompatible and both can be used at different stages of the mesh optimization cycle.

An illustration of the curvature-based refinement is given in Figure 12. Here the procedure has been applied to the coarse mesh depicted in Figure 10(a). The number of triangles was increased to 20 162 and the quality statistics are given in Figure 12(b). Again the quality is very good with most of the triangles having $Q > 0.7$. It is interesting to observe, by comparison with the coarsening case, that it is easier to achieve a better distribution of mesh quality index with fine meshes.

4. EXAMPLES OF APPLICATION

To demonstrate the ability of the techniques presented to handle complex geometries and to produce high-quality meshes, we consider the reconstruction of two different physiological geometries: a femoral by-pass graft acquired using MRA imaging and a nasal cavity acquired via a CT scan.

The strategy for generating the meshes for these examples is the following. The starting point is the marching cubes triangulation which is then processed through a pre-defined sequence of the previously described mesh modifications. This sequence starts with a step of mesh refinement or derefinement, as required, where all the points are created or deleted at once to produce a new

mesh with the sought resolution. The quality of this mesh is improved by mesh smoothing (and projection onto the implicit surface). The smoothed mesh is then enhanced by side swapping. These two steps are repeated three times. The mesh modification process finishes with an additional mesh smoothing and projection step. For simplicity, the curvature-based metric used in these examples is taken to be isotropic, i.e. the spacing is the same in all directions and it is calculated from the minimum value of the radius of curvature.

4.1. Femoral by-pass graft

Here we consider the reconstruction and mesh generation for the geometry of a by-pass graft applied to an occluded femoral artery. In this operation a vein of the patient is extracted and grafted to an occluded artery to allow blood to by-pass the occlusion. This is schematically represented in Figure 13(a). Figures 13(b)–(f) show some reconstructions of the distal anastomosis, in other

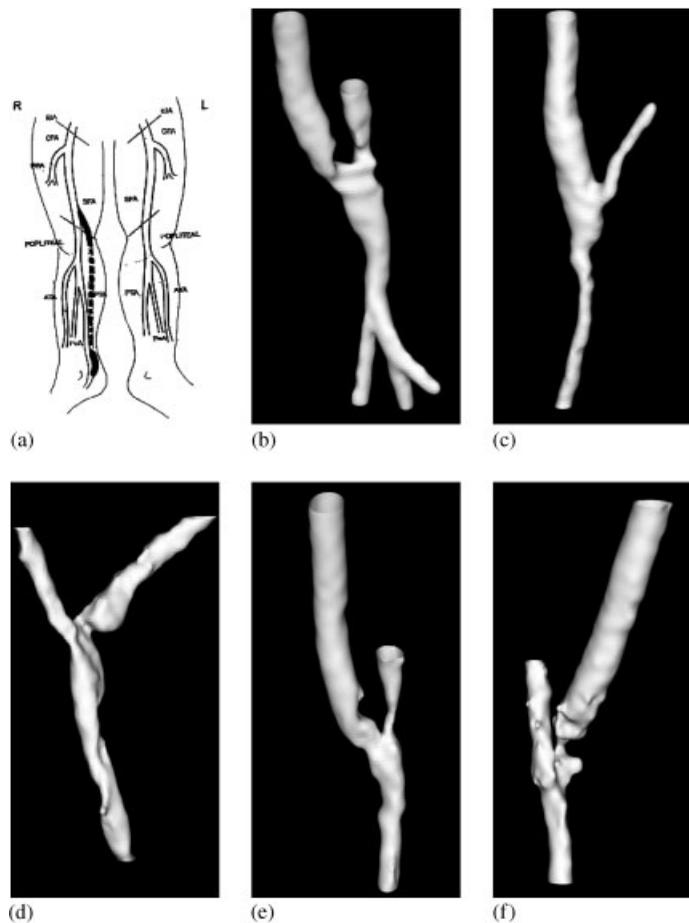


Figure 13. Femoral by-pass graft: (a) schematic representation of the by-pass operation and its location; and (b–f) examples of reconstruction of the distal anastomosis.

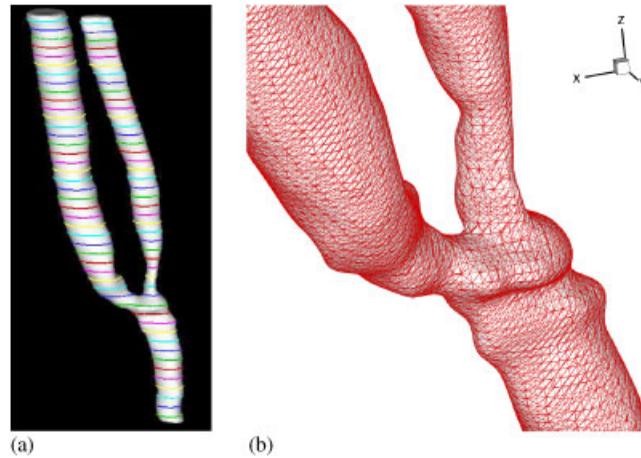


Figure 14. Femoral by-pass graft: (a) planar contours and interpolated implicit surface; and (b) an enlarged view, near the bifurcation, of the marching cubes triangulation.

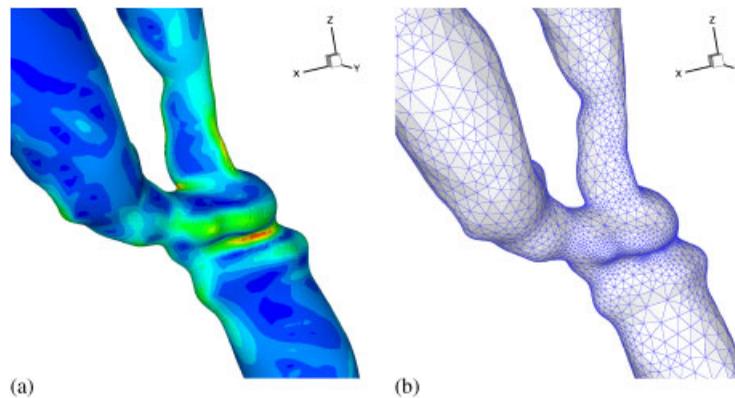


Figure 15. Femoral by-pass graft: (a) contours of normal curvature; and (b) curvature-based triangulation.

words, the graft bifurcation close to the foot. It can be observed that the geometries vary widely in complexity and some of them also show significant variations in curvature. This surgical procedure has a high rate of failure due to myo-intimal hyperplasia, i.e. the deposition of fatty deposits in the smooth muscle of the artery, which is linked to the local flow dynamics.

Figure 14(a) shows the implicit surface reconstruction together with the contours used for its interpolation. The marching cubes triangulation, obtained by evaluation on a lattice where the voxel size is that of a pixel in the input images, is depicted in Figure 14(b). This mesh contains 144 768 triangles.

Figure 15(a) depicts the surface distribution of the normal curvature. The mesh modification procedure has been applied to obtain a mesh with 16 238 triangles, shown in Figure 15(b), that is adapted to this curvature distribution.

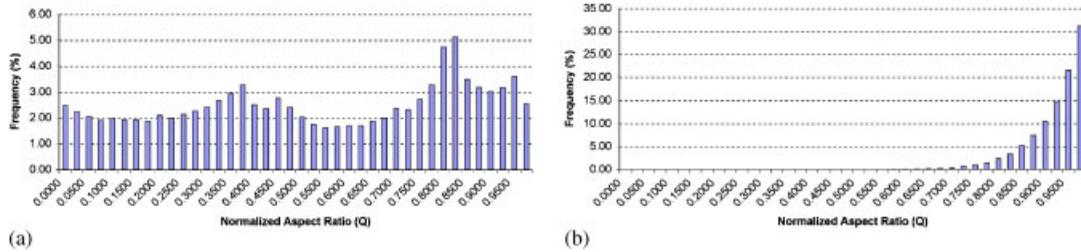


Figure 16. Comparison of mesh quality: (a) marching cubes mesh; and (b) optimal mesh.

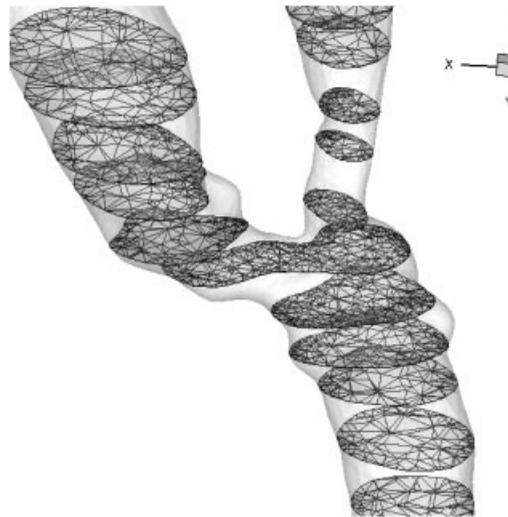


Figure 17. Femoral by-pass graft: volume mesh represented by means of a sequence of planar triangular meshes that result from the intersection of the tetrahedral mesh with a equispaced sequence of planes perpendicular to the z -axis.

The quality of the initial marching cubes triangulation and that of the modified by the proposed method are compared in Figure 16. It shows that the quality of the initial mesh is not very good with an even distribution of values of Q . On the other hand, the frequency graph for the modified triangulation clearly shows a shift towards values of $Q > 0.6$ which is an excellent distribution of mesh quality.

To further demonstrate the validity of the surface, a volume mesh containing 14 180 points and 59 091 tetrahedral elements was generated using the unstructured volume generator TGrid [26] and it is shown in Figure 17.

4.2. Nasal cavity

This example considers a model of the geometry of the nasal cavity as depicted in Figure 18(a). Here surface reconstruction and flow simulation could prove very useful in the planning of corrective

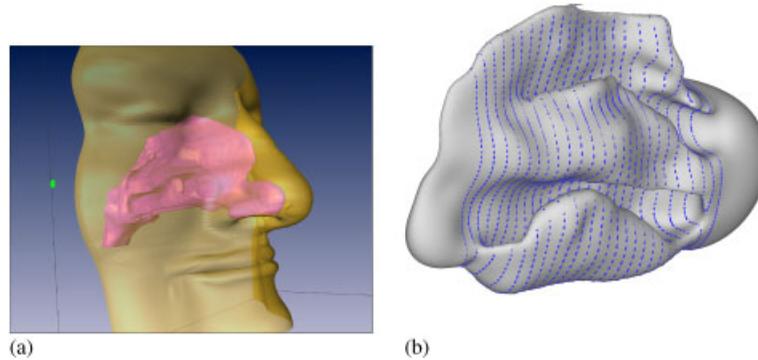


Figure 18. Nasal cavity: (a) schematic representation of the geometry of the nasal cavities and their location; and (b) planar contours and interpolated implicit surface for a left-side nasal cavity.

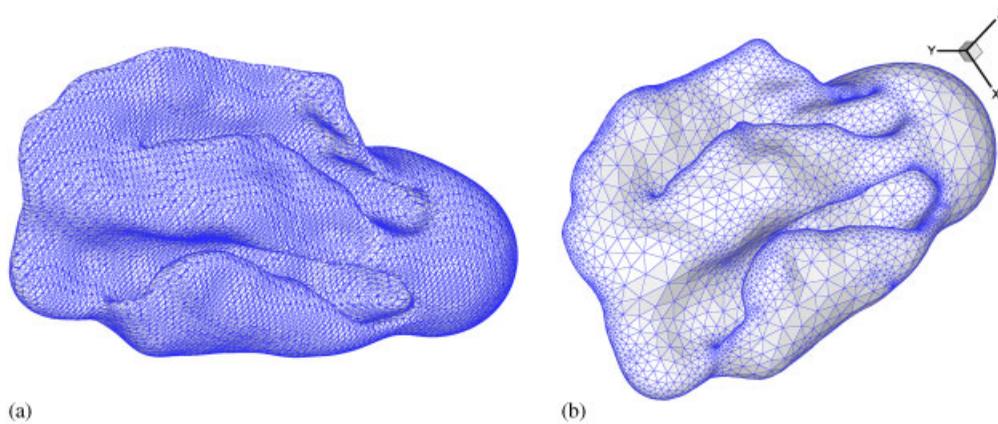


Figure 19. Nasal cavity: (a) marching cubes mesh obtained by evaluation of the implicit function on a uniform lattice; and (b) optimal mesh obtained using a curvature-based metric.

surgery for the nasal cavities. This is sometimes referred to as *virtual septoplasty*. The implicit surface representation and the contours used for interpolating it are depicted in Figure 18(b). It can be observed that the geometry of the cavity is very complex presenting significant variations in curvature throughout.

The initial triangulation was obtained using the method of marching cubes on a uniform lattice of voxels twice the size of the pixel in the input images. This triangulation contains 94 422 triangles and is shown in Figure 19(a). The triangulation obtained by mesh remodelling to adapt it to the curvature of the surface is depicted in Figure 19(b). This mesh has a total of 18 740 triangles and, as it can be noted even by a simple visual inspection, much better quality.

A quantitative assessment of the improvement on mesh quality is provided by the mesh statistics for both meshes shown in Figure 20. The result is consistent with that discussed in the previous example. The quality of the marching cubes triangulation is very bad, showing an almost uniform distribution of mesh quality index frequencies. The curvature adapted mesh shows a bias towards values of $Q > 0.6$ thus indicating a very significant improvement in mesh quality.

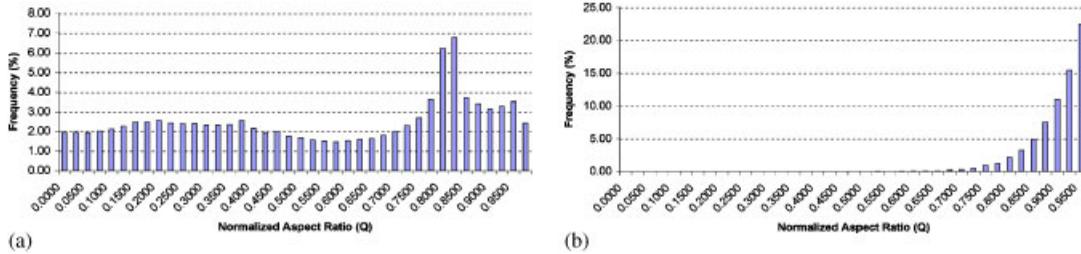


Figure 20. Comparison of mesh quality for the nasal cavity: (a) marching cubes triangulation; and (b) curvature-based triangulation.

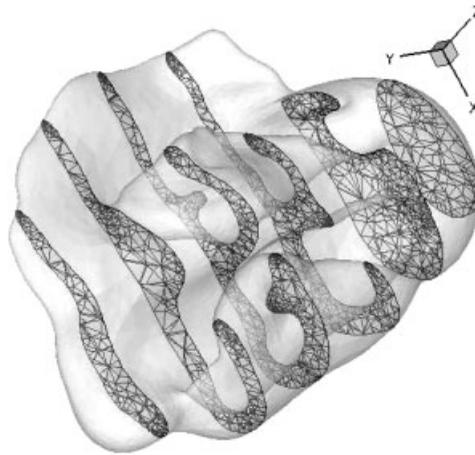


Figure 21. Nasal cavity: volume mesh represented by means of a sequence of planar triangular meshes that result from the intersection of the tetrahedral mesh with a equispaced sequence of planes perpendicular to the z -axis.

The validity of the surface is again shown by the generation using TGrid [26] of a volume mesh containing 15 100 points and 59 831 tetrahedral elements which is shown in Figure 21.

4.3. Assessment of the geometrical accuracy of the optimal triangulation

The two optimal triangulations generated for the geometries of the by-pass graft and the nasal cavity examples presented in the previous sections have been obtained by specifying a target error $\varepsilon = 0.01$. They show that the optimization procedure causes a decimation of the elements with respect to the marching cubes triangulation, with an improvement of the overall mesh quality. To show that this has been achieved with little loss in spatial resolution we have assessed the spatial distribution of the geometric error $\varepsilon = \delta/\rho$. Its value has been calculated for all the sides of the triangulation by projecting the midpoint of the side onto the implicit surface to obtain the foot point, then the radius of curvature ρ is evaluated at the foot point and, finally, the distance to the surface δ is taken to be the distance between the mid-side point and the foot point.

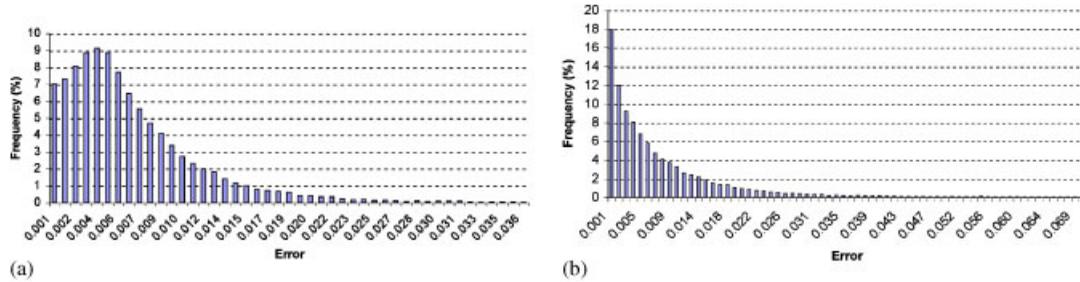


Figure 22. Frequency of geometric error $\varepsilon = \delta/\rho$ for a target error $\varepsilon = 0.01$:
 (a) femoral by-pass graft; and (b) nose cavity.

Frequency plots of the geometric error are shown in Figures 22(a) and (b) for the graft by-pass and the nose cavity, respectively. They show that an error smaller than the target $\varepsilon = 0.01$ has been achieved for the majority of triangles in the mesh, thus confirming our claims of improved quality with minimal loss of geometrical accuracy. The different shape of the frequency plots could be explained by the very different spatial distribution of curvature in these geometries. The nasal cavity presents a much larger surface area where the radius of curvature is small than the femoral by-pass graft.

5. CONCLUSIONS

We have shown that the use of an implicit surface defined in terms of RBF is a viable tool for shape reconstruction from a stack of medical images. An initial triangulation is extracted from the zero iso-contour of the implicit function by a marching cubes algorithm. The quality of this triangulation is poor in general but the proposed mesh enhancement techniques are able to transform it into a good quality triangulation suitable for mesh generation and subsequent flow solution. This has been demonstrated by comparing mesh quality statistics of the initial and modified triangulation. The sequence of operations required to obtain a surface mesh from a set of medical images is fully automated and requires minimal user intervention.

The refinement and derefinement strategies used here are based on a geometric mesh metric that accounts for surface curvature only. However, they can be easily modified to incorporate adaption procedures based on a more general mesh metric based on error estimators to produce flow adapted solutions.

ACKNOWLEDGEMENTS

Partial support for this work was provided by the EU through the Research Training Network 'HaeModel' grant HPRN-CT-2002-00270. The authors would like to thank Alberto Gambaruto for providing the mesh smoothing routine. We would also like to thank Denis Doorly, Kuan Loke 'Adrian' Lee and Donal Taylor for providing the CT images of the nasal cavities and Figure 18(a) as well as helping with the reconstruction and for fruitful discussions.

REFERENCES

1. Löhner R, Cebal J, Soto O, Yim P, Burgess JE. Applications of patient-specific CFD in medicine and life sciences. *International Journal for Numerical Methods in Fluids* 2003; **43**:637–650.

2. Ladak HM, Milner JS, Steinman DA. Rapid three-dimensional segmentation of the carotid bifurcation from serial MR images. *Journal of Biomechanical Engineering* 2000; **122**:96–99.
3. Sethian JA. *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science*. Cambridge University Press: Cambridge, MA, 1999.
4. Carr JC, Fright WR, Beatson RK. Surface interpolation with radial basis functions for medical imaging. *IEEE Transactions on Medical Imaging* 1997; **16**(1):96–107.
5. Peiró J, Giordana S, Griffith C, Sherwin S. High-order algorithms for vascular flow modelling. *International Journal for Numerical Methods in Fluids* 2002; **40**:137–151.
6. Turk G, O'Brien JF. Modelling with implicit surfaces that interpolate. *ACM Trans. Graph.* 2002; **21**(4):855–873.
7. Buhmann MD. *Radial Basis Functions: Theory and Implementations*. Cambridge University Press: Cambridge, MA, 2003.
8. Bloomenthal J. An implicit surface polygonizer. *Graphics Gems IV* 1994; 324–349.
9. Lorensen WE, Cline HE. Marching cubes: a high resolution 3D surface construction algorithm. *Computer Graphics; Proceedings of SIGGRAPH 87*, 1987; **21**(4):163–169.
10. Frey PJ. Generation and adaptation of computational surface meshes from discrete anatomical data. *International Journal for Numerical Methods in Engineering* 2004; **60**:1049–1074.
11. MATLAB—The language of technical computing. www.mathworks.com
12. Otsu N. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics* 1979; **9**(1):62–66.
13. Shaback R. Error estimates and condition numbers for radial basis function interpolation. *Advances in Computational Mathematics* 1995; **3**:251–264.
14. Beatson RK, Cherrie JB, Mouat CT. Fast fitting of radial basis function: methods based on preconditioned GMRES iteration. *Advances in Computational Mathematics* 1999; **11**:253–270.
15. Saad Y, Schultz MH. GMRES: a generalized minimal residual algorithm for solving non-symmetric linear equations. *SIAM Journal on Scientific and Statistical Computing* 1986; **7**:856–869.
16. Iske A. Optimal distribution of centres for radial basis function methods. *Report TUM-M0004*, Technische Universität München, 2000.
17. Beatson RK, Light WA, Billings S. Fast solution of the radial basis function interpolation equations: domain decomposition methods. *SIAM Journal on Scientific Computing* 2000; **5**(22):1717–1740.
18. <http://www.unchainedgeometry.com/jbloom/misc/polygonizer++.tgz>, last accessed 13 July 2006.
19. Hilbert D, Cohn-Vossen S. *Geometry and the Imagination* (2nd edn). Chelsea Publication Company: New York, 1990.
20. Peiró J. Surface grid generation. In *Handbook of Grid Generation* (Chapter 19), Thompson JF, Soni BK, Weatherill NP (eds). CRC Press: Boca Raton, FL, LLC, 1999.
21. Hartmann E. On the curvature of curves and surfaces defined by normalforms. *Computer Aided Geometric Design* 1999; **16**:355–376.
22. Desbrun M, Meyer M, Schroder P, Barr AH. Implicit fairing of irregular meshes using diffusion and curvature flow. *Computer Graphics, Annual Conference Series* 1999; **33**:317–324.
23. Shewchuck J. What is a good linear element? Interpolation, conditioning, and quality measures. *Eleventh International Meshing Roundtable*, Ithaca, New York, September 2002; 115–126.
24. Frey PJ, George PL. *Mesh Generation*. Hermes Science Publishing: Oxford, Paris, 2000.
25. Taubin G. A signal processing approach to fair surface design. *Computer Graphics, Annual Conference Series* 1995; **29**:351–358.
26. *TGrid: Unstructured Volume Meshing*. Fluent Inc.: Lebanon, U.S.A., www.fluent.com