OBJECT RECONSTRUCTION ON 3D ULTRASOUND IMAGES USING SIMPLEX MESHES

E.V. Snezhko and A.V. Tuzikov

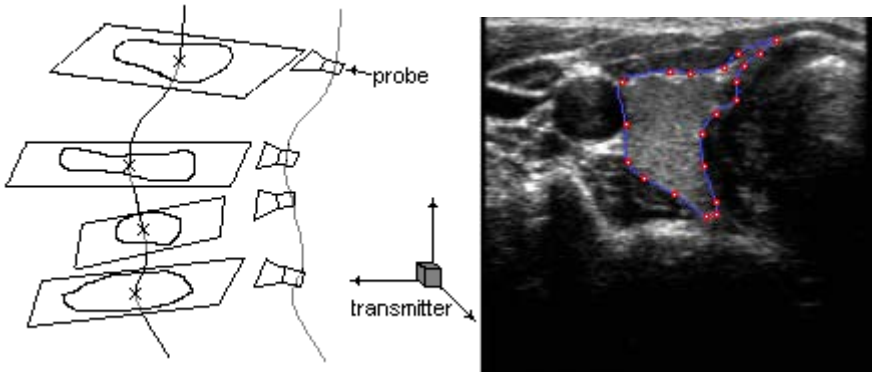
United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Surganova 6, 220012 Minsk, Belarus, e-mails: *{snezhko,tuzikov}@mpen.bas-net.by*

Abstract. We present an algorithm for 3D object reconstruction on ultrasound images based on simplex meshes. The algorithm uses manually traced object boundaries on several representative non-parallel cross-sections of 3D US image. Simplex meshes are attracted to these contours under the influence of external forces, derived from the initial data. Several algorithms for external force generation are investigated.

1. Introduction.

Visualization devices used in modern medicine allow acquisition of volumetric images of anatomical structures. They have necessitated the development of computer-aided medical image analysis techniques. Ultrasound imaging unlike other medical imaging technologies is preferable for some tasks because of its relative equipment simplicity and harmlessness during examination. 3D object reconstruction usually is performed using cross-section editing. Due to shortcomings typical for ultrasound data, such as artifacts, spatial aliasing, and noise, ultrasound image segmentation is usually done manually and becomes extremely labor intensive and time-consuming. Three-dimensional deformable models are very useful in this case for object segmentation because they allow to trace regions of interest manually only on several representative cross-sections. Deformable models adapt to these contours while intuitively shaping between them.

Ultrasound images are generated while measuring an acoustic response to an ultrasound impulse from an organ under investigation. 3D ultrasound unlike conventional ultrasound imaging allows acquiring the impulse response in three dimensions rather than in one plain. There are several techniques used for generating 3D ultrasound data. They are distinguished by positioning principles and scanning patterns [TrGe03]. In one of these techniques an operator passes the ultrasound probe near an organ being investigated. One of the approaches uses a magnetic field position sensor. The transmitter generates a magnetic field and the receiver mounted on the ultrasound probe allows determining the probe position and orientation in this field. After scanning one gets a sequence of 2D ultrasound images with their spatial characteristics. The Stradx software [PrGe99] allows accurate acquisition of ultrasound data, immediate visualization of arbitrary cross-sections and measurements of outlined objects.

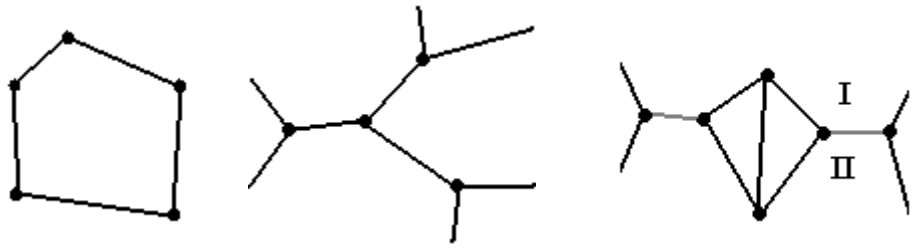


a) *b)*

Fig. 1. Non-parallel cross-sections with contours in space a); object boundary extraction on ultrasound image b).

Simplex meshes.

We use 1 and 2-simplex meshes for representation of contours and surfaces respectively. 1-simplex mesh is a closed simply connected polygonal line, and 2-simplex mesh is a mesh with three neighbors for each vertex and zero or one common edge for any two faces. An example of the mesh which is not 2-simplex mesh is shown in Fig. 2c) (faces I and II have two common edges).



a) *b)* *c)*

Fig.2. Examples of meshes: a) 1-simplex mesh; b) 2-simplex mesh; c) not 2 simplex mesh.

Detailed description of simplex meshes can be found in [Del97]. The definition of simplex meshes and triangulations are closely related. Their underlying graphs are dual, but this is only in terms of topology, since the dual relation between them does not depend on the positions of their vertices. There can be defined several notions for simplex meshes such as local curvature and normal vectors that allow easy shape control of simplex meshes. Furthermore, topological operators are defined to locally refine simplex meshes and to control their topology in efficient manner. Thus, simplex meshes can represent shapes of any topology and complexity.

The law of simplex mesh deformation contains an internal and external force:

$$P_i^{t+1} = P_i^t + (1-\gamma)(P_i^t - P_i^{t-1}) + F_{ext,i} + F_{\text{int},i} \quad (1)$$

Here t is a current time moment, P_i is a current simplex mesh vertex and γ is a coefficient which defines the kind of dependence between vertex positions at different moments. $F_{ext,i}$ and $F_{\text{int},i}$ are external and internal forces respectively which influence at the vertex P_i at time t.

The internal force is derived from the minimization of a local energy that is proportional to the deflection from some “rest shape”. It smoothes the deformable model and several smoothness constraints can be defined for simplex meshes. The external force attracts vertices of simplex mesh towards the dataset, contours in our case. Here we propose three algorithms for external force generation.

2. External force generation.

According to the first algorithm, the dataset is placed into a circumscribed parallelepiped. This parallelepiped is divided into a set of small parallelepipeds (boxels). Their number depends on the number of vertices in the simplex mesh and points in the initial dataset.

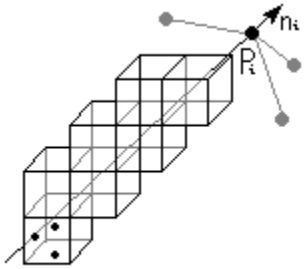


Fig. 3. Searching corresponding points from the dataset for the simplex mesh vertex P_i along the normal vector n_i at this vertex.

The simplex mesh is also placed into this parallelepiped. For each simplex mesh vertex the algorithm searches a boxel along the normal vector at this vertex, which contains points from the initial dataset. The search is carried out in two directions within a sphere of radius depending on the diameter of the initial dataset (for a well initialized simplex mesh we pick the value of the radius to be about 15% of the initial dataset diameter). The sphere is centered in the current simplex mesh vertex. As soon as the appropriate boxel is found we determine the closest point from the dataset points within the boxel and assume that the simplex vertex is attracted to this point. The number of boxels should not be too small since it may dramatically increase search time within a boxel containing points from the dataset. However, there should not be a lot of them since the algorithm may fail finding corresponding points almost for all simplex mesh vertices. Thus, it

is rather important to determine an appropriate number of boxels. We suggest taking a circumscribed cube with about $(\sqrt{m/8}+\sqrt[3]{n/4})/2$ boxels per side (where n is the number of points in the initial dataset and m is the number of simplex mesh vertices). The external force is applied along the direction of the normal vector at the vertex and is proportional to the distance between dataset points and simplex mesh vertices.

The second algorithm developed here builds a contour on the simplex mesh for each cross-section where the object boundary is defined. This contour consists of new simplex mesh vertices that are added while cutting up mesh faces by the selected cross-sections (see Fig. 4 and 5).

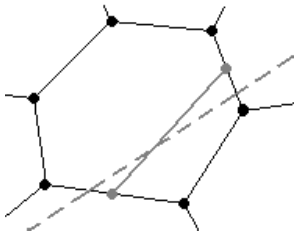


Fig. 4 Mesh face is cut up by a cross-section (shown in dotted line) and two new vertices are added.

The mesh vertices of this contour are attracted by the corresponding points of the outlined object boundary on the cross-section. The external forces for these vertices are computed the same way as in the first algorithm.

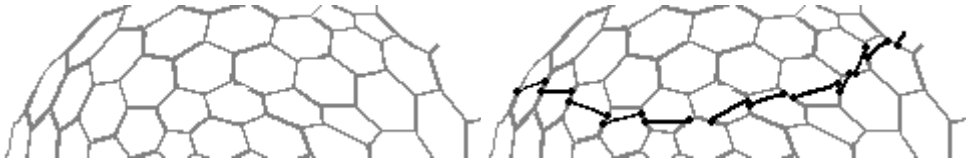


Fig. 5. Simplex mesh is cut up by a cross-section and a contour is added to the mesh.

The third algorithm generates a vector field near the initial dataset and any mesh vertex placed into this field will be moved according to the field value at the given point. The field is specified on a three-dimensional cubic lattice by calculating the force value at each lattice node. For the three-dimensional case a lattice with $150\times150\times150$ nodes was used in our tests (150 nodes for each dimension).

We describe here this algorithm for two-dimensional data but it is easily extended to the three-dimensional case.

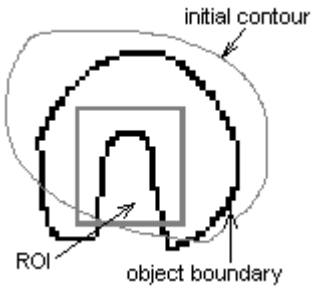


Fig. 6.

The initial contour from one cross-section with a region of interest selected is shown in Fig. 6. This region is outlined due to problems for a deformable contour to progress into this concavity. The distance function is computed for the image containing the contour (see Fig. 7). A 3×3 chamfer mask is used for computing this distance function in 2D [Bor94, LeLe92]. This algorithm is implemented for 3D case for $3\times3\times3$ mask.

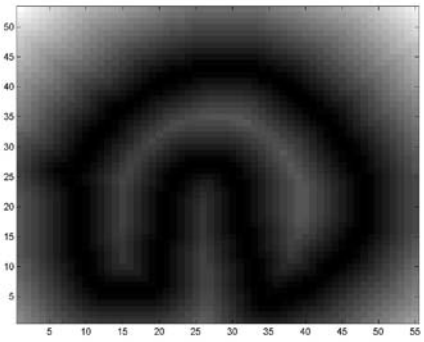


Fig. 7. Distance function.

One can use a vector field of negative gradient computed for the distance function as an initial approximation of the external force field. Vertices of a deformable contour placed into this field will be drawn towards the contour by the force. But within the ROI (see Fig. 8) it will be difficult for the deformable contour to progress into the boundary concavity. In this case external force field values within the concavity are directed towards the contour tabs in the opposite directions and not into the boundary concavity. In order to address this problem an additional transformation is carried out for the gradient vector field.

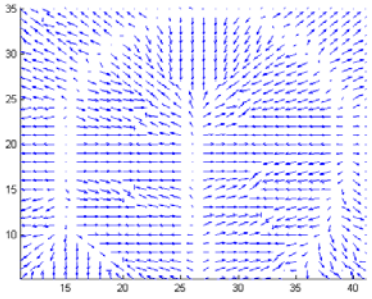
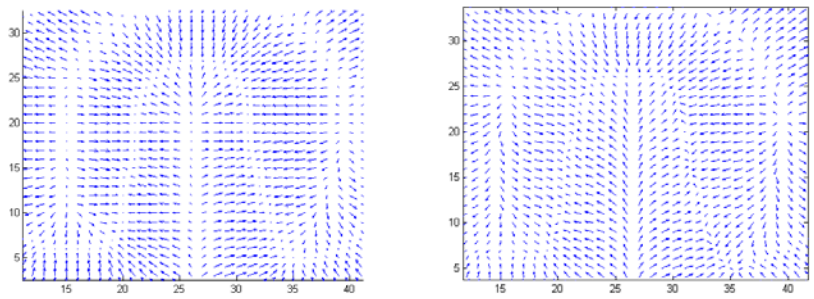


Fig. 8. Gradient vector field within the ROI.

Gradient vector flow field (GVF) is proposed in [XuPr98]. GVF in 2D is a vector field $v(x, y) = [u(x, y), v(x, y)]$ that minimizes the energy functional

$$E = \iint_{\Omega} \mu(u_x^2 + u_y^2 + v_x^2 + v_y^2) + |\nabla f|^2 |v - \nabla f|^2 \, d\Omega \qquad (2)$$

Here $\nabla f(x, y)$ is the gradient of the distance function. Parameter μ regularizes the balance between two parts of the integral (we apply $\mu = 2.5$). The result of this transformation is shown in Fig. 9(a). Unfortunately, the problem is not solved completely. Therefore, we suggest performing one more step. For each vector from $v(x, y)$ with relatively small length in some neighborhood its length is increased to become the same as the longest one in this neighborhood. The vector field obtained $v^*(x, y)$ then can be used as ∇f in (2) for one more transformation. The result is shown in Fig. 9(b). Now the deformable contour has no problem to progress into boundary concavities.



a) *b)*

Fig.9. Gradient vector flow fields: a) after the first transformation; b) after the second transformation.

For 3D case this algorithm is easily extended. It consists in obtaining GVF $\nu(x, y, z) = [u(x, y, z), v(x, y, z), w(x, y, z)]$ from the gradient vector field of three-dimensional distance function followed by the suggested heuristic for obtaining $\nu^*(x, y, z)$.

3. Implementation.

All three algorithms were implemented and tested (see Fig. 10 for the illustration). First two of them have shown a good speed and accuracy. The time of work for the first one is linear respecting to $m^{3/2}n^{1/3}$, and for the second one is linear respecting to m, where n is the number of points in the initial dataset and m is the number of simplex mesh vertices. The third algorithm utilizes up to 400Mb of memory and 330 seconds on Pentium 4 1.9Ghz processor to generate a vector field for $150 \times 150 \times 150$ lattice.

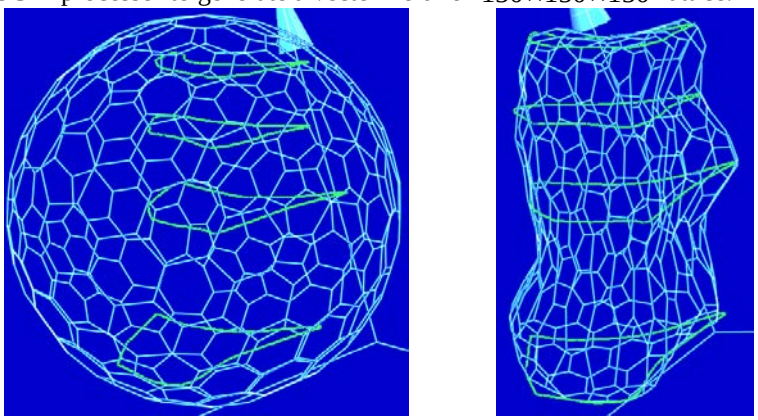


Fig.10. Object segmentation by the first algorithm: a) initial simplex mesh; b) final result.

4. Conclusion.

We have applied 2-simplex meshes for semi-automatic segmentation of objects on 3D ultrasound images and proposed three algorithms for simplex mesh external force generation. The resulting external force can be also chosen as a weighted sum of forces obtained by algorithms described above (with different weighting coefficients):

$$F_{ext,i} = \alpha_1 F^{(1)}_{ext,i} + \alpha_2 F^{(2)}_{ext,i} + \alpha_3 F^{(3)}_{ext,i} \quad (3)$$

These algorithms are applicable to various types of images. Simplex meshes with external force generators proposed here are expected to be utilized not only for object shape reconstruction on ultrasound images but also on 3D images of other modalities.

The work was done in framework of the ISTC B-517 project.

References

[1]. G.M. Treece, A.H. Gee, R.W. Prager, C.J.C. Cash, and L.H. Berman, "High-Definition Freehand 3-D Ultrasound", *Ultrasound in Med. & Biol.*, **29**, 4, (2003), 529-546.

[2]. G. Borgefors, "Applications using distance transformation. - Aspects of Visual Form Processing", *World Scientific*, (1994), 83-108.

[3]. H. Delingette, "General Object Reconstruction based on Simplex Meshes", *Technical Report 3111, Institute National de Recherche en Informatique et en Automatique, France*, (February 1997).

[4]. F. Leymarie, M. D. Levine, "Fast Raster Scan Distance Propagation on the Discrete Rectangular Lattice", *CVGIP: Image Understanding*, **55**, 1, (January 1992), 84-94.

[5]. R. Prager, A. Gee, and L. Berman, "Stradx: real-time acquisition and visualization of freehand 3D ultrasound", *Medical Image Analysis*, **3**, 2, (1999), 129-140.

[6]. Ch. Xu, J.L. Prince, "Snakes, Shapes, and Gradient Vector Flow", *IEEE Transactions on Image Processing*, **7**, 3, (March 1998), 359-369.