Computers & Graphics 33 (2009) 424–432

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Chaos and Graphics

Gradient field based inhomogeneous volumetric mesh deformation for maxillofacial surgery simulation

Sheng-hui Liao [a,*], Ruo-feng Tong [b], Jin-xiang Dong [a], Fu-dong Zhu [b]

[a] State Key Laboratory of CAD and CG, Department of Computer Science and Engineering, Zhejiang University, Hangzhou, China
[b] The Affiliated of Stomatological Hospital, College of Medicine, Zhejiang University, Hangzhou, China

ARTICLE INFO

Article history:
Received 8 December 2008
Received in revised form
1 March 2009
Accepted 3 March 2009

Keywords:
Volumetric mesh deformation
Volumetric gradient field
Local transformation
Inhomogeneous material
Maxillofacial surgery
Soft tissue simulation

ABSTRACT

This paper presents a novel inhomogeneous volumetric mesh deformation approach by gradient field manipulation, and uses it for maxillofacial surgery simulation. The study is inspired by the state-of-the-art surface deformation techniques based on differential representations. Working in the volumetric domain instead of on only the surface can preserve the volumetric details much better, avoid local self-intersections, and achieve better deformation propagation because of the internal mesh connections. By integrating the mesh cell material stiffness parameter into our new discrete volumetric Laplacian operator, it is very convenient to incorporate inhomogeneous materials into the deformation framework. In addition, the system matrix for solving the volumetric harmonic field to handle the local transformation problem is the same used for Poisson reconstruction equation, thus it requires solving essentially only one global linear system. The system is easy to use, and can accept explicit rotational constraints, or only translational constraints to drive the deformation. One typical maxillofacial surgery case was simulated by the new methodology with inhomogeneous material estimated directly from CT data, and compared to the commonly used finite element method (FEM) approach. The results demonstrated that the deformation methodology achieved good accuracy, as well as interactive performance. Therefore, the usage of our volumetric mesh deformation approach is relevant and suitable for daily clinical practice.

1. Introduction

Maxillofacial surgery treats abnormalities of the skeleton of the head, including osteotomies, bone fragment repositioning, restoration of bone defects, and inserting implants. The human face plays a key role in interpersonal relationships, and even very subtle malformation of facial proportions can strongly affect the appearance of a face and determine aesthetic aspects such as individual beauty. Therefore, planning of the operation and reliable prediction of the facial changes are very important in view of a better preparation, shorter operation time, and improved surgical outcome.

Currently, bone related pre-operative planning and simulation is a relatively mature technology [1], but the prediction of the soft tissue deformation is still an open and very active research field. In particular, the deformation accuracy and computation time are the main constraints for the modeling of soft tissue in daily clinical practice.

The two most popular and widely used methods for modeling of deformable soft tissue are mass–spring model and finite element method (FEM). A mass–spring model consists of a set of point masses that are connected by springs, whose physics is straight forward. After the pioneering work of Terzopoulos and Walters [2,3], a large variety of these models have been implemented due to their simplicity of implementation and relatively low computational complex [4–8]. Until the work of Teschner et al. [7,8], mass–spring models were typically combined with a time-integration scheme of the Newtonian motion equation to find the deformation of the soft tissues. Teschner suggested that only the deformation of the tissues was important and not the animation to the new position. He proposed the use of an iterative scheme to directly find the new deformed position, and reduced the simulation time to less than 20 s. But the simplicity of mass–spring model comes at the price of a few drawbacks, e.g., the behavior of the system depends on the way the spring network is set up; it is also difficult to tune the spring constants to get the desired behavior; and it cannot capture volumetric effects directly such as volume conservation or prevention of volume inversions.

In contrast, FEM is a more biomechanical relevant model as derived directly from continuum mechanics. Although considered

* Corresponding author. Tel.: +86 057187933346.
E-mail addresses: shliao@zju.edu.cn, lshenghui@gmail.com (S.-h. Liao).

0097-8493/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cag.2009.03.018

to be less efficient than mass–spring model, there is a growing trend in using FEM to model soft tissue behavior for providing more accurate deformation result. Koch et al. [9] presented a nonlinear FEM model with C^1-continuous surface to simulate the new facial outlook. Keeve et al. [10] employed prismatic elements to construct the facial model. Sarti et al. [11] provided a voxel-based model constructed directly form CT data and simulated on a super computer. Zachow et al. [12] presented a tetrahedral model, which was better suited for simulation of biological objects, and was able to compute the new facial outlook in less than 5 min. Chabanas et al. [13] was one of the first to present a complete maxillofacial planning protocol. To overcome the difficult meshing step, he registered a generic model with muscles structures to patient specific data, which enabled simulation of the post-operative functional behavior. This functional simulation of the facial expressions after surgery was introduced by Gladilin at the same moment [14]. More recently, Westermark et al. [15] reported on a quantitative validation for one patient, and concluded that a mean prediction error of 1–1.5 mm could be achieved with their model.

It is worth noticing that Cotin et al. [16] recently introduced a new soft tissue deformation model, called the "mass-tensor model", which has the easy architecture of the mass–spring model, but keeps the bio-mechanical relevance of FEM. Mollemans et al. proposed the usage of a tetrahedral mass-tensor model with two extensions—a direct computation step and a local dynamic stop criterion, to simulate the maxillofacial surgery very fast with a limited accuracy [17]. They reported that the average simulation time of the mass-tensor model equals more or less 10 s, achieving an average time gain of more than 40% compared to FEM or mass–spring model [18].

Despite these good results, further studies are still very necessary to make the soft tissue deformation model more computationally efficient and physically plausible. It is increasingly required to develop real-time interactive simulation systems from the point of view of clinical practice. To address the problem, this paper presents a novel inhomogeneous volumetric mesh deformation approach by volumetric gradient field manipulation, and uses it to predict new facial outlook for maxillofacial surgery simulation.

The major contributions in this work include the following:

(1) Deduce a new *material weighted cotangent Laplacian operator for volumetric mesh*.
(2) Develop a novel inhomogeneous volumetric mesh deformation approach by gradient field manipulation, which requires solving essentially only one global linear system and is very efficient. In addition, the system can accept explicit rotational constraints, OR only translational constraints to drive the deformation, thus is very easy to use.
(3) Demonstrate an application of the inhomogeneous volumetric mesh deformation approach for the maxillofacial surgery simulation.

2. Design considerations

The study is inspired by the state-of-the-art surface deformation techniques based on differential representations, which have gained significant popularity recently for their robustness, speed, and ease of implementation [19–34]. The main idea behind this family of deformation techniques is to use a surface representation that puts the local differential properties in focus, and to preserve these differential properties under deformation, aspiring to obtain an intuitive, detail-preserving deformation result [32]. Generally speaking, this is achieved by constructing the differential representation of the input surface, then manipulating this representation according to the modeling constraints, and finally performing "integration" or reconstruction of the surface coordinates from the modified differential representation. Various techniques differ by the particular differential properties they use, e.g., gradient-based representation [19,22,23,28] and Laplacian-based representation [20,21,26,29], and the manipulation thereof, but the general framework remains the same.

Despite the ability to preserve surface geometric detail, it should be noted that almost those surface based deformation methods do not take the volume of the object into account. As a result, self-intersections might happen during deformation, and in general the volume of the shape cannot be preserved. In other words, surface based deformation methods are not well-suited for the modeling of soft tissue, because approximately 70 percent of the human body is based on water and the overall volume is well maintained even during a large deformation.

A natural and straightforward solution to address this problem is to extend the surface based deformation methods to volume based approaches, and enforce volume preservation through an explicit construction of a representation that models the interior of the shape, e.g., using volumetric mesh. Fig. 1(a) illustrates a 360° large twist deformation of a bar-like tetrahedral volumetric mesh by our method, and Fig. 1(b) shows the same deformation of its boundary triangle surface mesh. Under this extreme deformation, the surface based method leads to obvious shrinking effect and volume losing in the cross-sectional areas, while the volumetric mesh deformation exhibits much better volume preservation property. This is because the volumetric mesh deformation enables distance preservation not only on the surface, but also throughout the interior of model, which makes the deformation resistant to changes in volume and cross-sectional areas.

Another important motivation to use volumetric deformation it that there is some difference between the setting locations of boundary conditions in most surface shape editing tasks and bone-related planning boundary conditions in surgery. One usually selects the boundary conditions on some "tip", "end", or "border" features, which are convenient to control the behavior of the whole shape. But the bone repositioning boundary conditions in surgery usually affect the "center" regions on one side of soft tissue, if there is no internal mesh connection, the other side of soft tissue will exhibit unnatural deformation or even serious local self-intersections. Fig. 2(a)–(d) simulates a simplified soft tissue represented by a rectangle of three layers of tetrahedral elements, whose left and right boundaries are fixed, and the center location of top surface is set as the moving handle. Natural and intuitive deformation results are obtained for both the dragging up (Fig. 2(a) and (b)) and pushing down operations (Fig. 2(c) and (d)). In

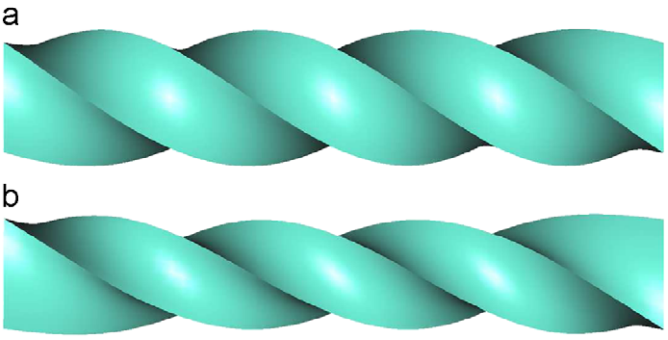

Fig. 1. Twist deformation (360°) for (a) tetrahedral volumetric mesh and (b) triangle surface mesh.

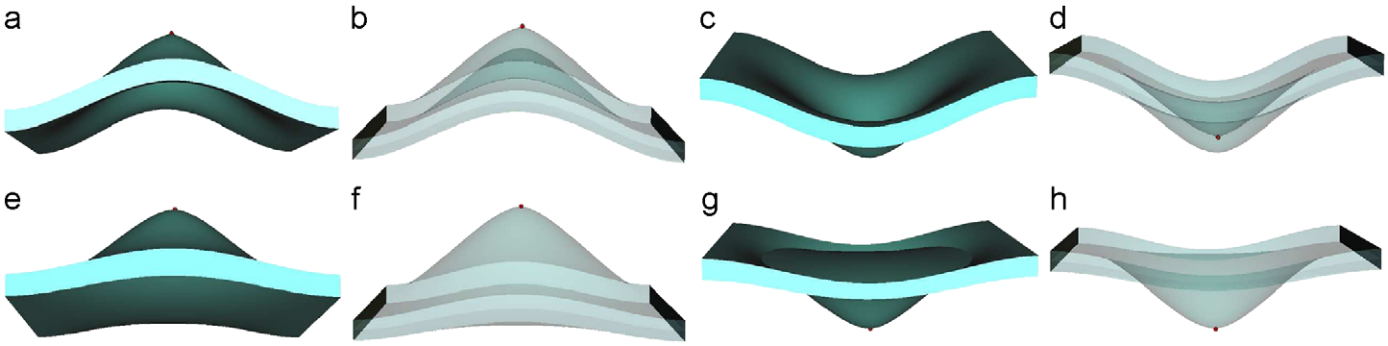

Fig. 2. Drag and push on the center region of a simplified rectangle soft tissue. (a)–(d) Tetrahedral volumetric mesh. (e)–(h) Surface mesh.

contrast, the deformation of its boundary surface mesh exhibits unnatural volume changes in the dragging operation (Fig. 2(e) and (f)), and serious local self-intersection in the pushing operation (Fig. 2(g) and (h)). In other words, in addition to impose volumetric constraints, working in the volumetric domain can further avoid local self-intersections, and achieve better deformation propagation. This also benefit from the internal mesh connections, which help to retain distances between diametrically opposed vertices on the model's surface, and prevent unintuitive shape transformations.

Besides, it is important to note that most of these geometric deformation techniques ignore the properties of the underlying materials, and thus make it difficult to generate physically plausible deformations, especially for inhomogeneous objects such as soft tissue. Popa et al. [28] proposed a two step approach to model surface deformation that incorporates non-uniform materials based on the framework of deformation gradients [22,27]. The material stiffness across the surface was specified by a paint-like interface or learned from sample deformations. In the first step, the deformation gradients defined at the anchor triangles were propagated to remaining triangles by a weighted blending. The interpolation weights were determined by solving an edge material stiffness weighted quadratic minimization. In the second step, optimal vertex positions were found such that the gradient transformation of each triangle remained as close as possible, in the least squares sense and weighted by the triangle material stiffness, to the previously calculated transformation. Thus, there were two different linear systems to be solved per selection of anchors.

In this paper, we extend the idea of surface editing with Poisson-based gradient field manipulation [19,23], edit the tetrahedral mesh by volumetric gradient field, and employ the volumetric harmonic field to handle the local transformation problem. We notice that Zhou et al. [26] also proposed to use volumetric deformation to achieve better volume preservation properties. While they augment the surface mesh with an inner regular grid, and employ a hybrid setting that using a triangle-based Laplacian for the surface and a graph-based quadratic programming formulation for computing interior weights. In contrast, our unified tetrahedral volumetric Laplacian setting is a lot less complex, and in addition to work with explicit rotational constraints, we investigate a scheme to accept only translational constraints to drive the deformation. Most importantly, to deal with the inhomogeneous objects such as soft tissue, we integrate the mesh cell material stiffness parameter into our new discrete volumetric Laplacian operator, and maintain the convenience of using harmonic field for local rotation interpolation.

The detailed method is described in Section 3. Then in Section 4, one typical maxillofacial surgery case is simulated by the new methodology with inhomogeneous material estimated directly from CT data. Finally, Section 5 summarizes our work.

3. Volumetric gradient based mesh deformation

Let $\mathscr{M} = (\mathscr{V}, \mathscr{K})$ be the tetrahedral mesh we want to deform, where $\mathscr{V} = \{\mathbf{v}_i \in R^3 | 1 \leqslant i \leqslant n\}$ is a set of point positions, and $\mathscr{K}$ is an abstract simplicial complex which contains all the vertex connectivity information. There are four types of elements in $\mathscr{K}$, vertices $\{i\}$, edges $\{i,j\}$, faces $\{i,j,k\}$, and tetrahedrons $\{i,j,p,q\}$. Take the tetrahedral mesh as a volumetric domain Ω, a discrete potential scalar field is defined as a piecewise linear function $u(x) = \sum_i \varphi_i(x) u_i$, with φ_i being the piecewise linear basis function valued 1 at vertex $\mathbf{v}_i$ and 0 at all other vertices, and u_i being the value of u at $\mathbf{v}_i$.

3.1. Basic deformation

Follow the basic setup of [19], we consider the three coordinates of a tetrahedral mesh as three scalar fields defined on the volumetric domain. Since tetrahedral meshes are piecewise linear models, such scalar fields are actually piecewise linear, and satisfy the definition of discrete potential fields. Then, the gradients of the coordinate functions $\boldsymbol{x}$, $\boldsymbol{y}$, $\boldsymbol{z}$ are encoded as the volumetric differential properties. When provided with new boundary conditions that stem from the user-defined modeling constraints (for the most part, Dirichlet boundary conditions that prescribe positions of some vertices), the deformed mesh can be reconstructed by a global optimization process that looks for a mesh whose gradients are as close as possible to the original gradients. In the continuous formulation, the deformed mesh $\mathscr{M}'$ is defined by the coordinate functions $\boldsymbol{x}'$, $\boldsymbol{y}'$, $\boldsymbol{z}'$ that minimize

$$\int\int\int_{\Omega} \|\nabla x' - \mathbf{g}_x\|^2 \, dV \tag{1}$$

(and the same for $\boldsymbol{y}'$, $\boldsymbol{z}'$), under some modeling constraints, where $\nabla = [\partial/\partial x, \partial/\partial y, \partial/\partial z]^T$ is the gradient operator, and $\mathbf{g}_x = \nabla x$ are the gradients of the original mesh coordinate functions. The Euler–Lagrange equation of this minimization is the Poisson equation

$$\Delta x' = \mathrm{Div}(\mathbf{g}_x), \tag{2}$$

where $\Delta = \partial^2/\partial x^2 + \partial^2/\partial y^2 + \partial^2/\partial z^2$ is the Laplacian operator, and $\mathrm{Div} = \partial/\partial x + \partial/\partial y + \partial/\partial z$ is the divergence operator.

Unfortunately, it is not straightforward for theoretical analysis and practical computing to discretize Poisson equation defined on an irregular 3D domain. Until recently Tong et al. [35] proposed well-defined discrete differential operators of scalar and vector fields on irregular tetrahedral mesh. For a piecewise linear scalar field $u(x) = \sum_i \varphi_i(x) u_i$, its discrete gradient is $\nabla u(x) = \sum_i u_i \nabla \varphi_i(x)$, which is a piecewise constant vector field, e.g., inside each tetrahedron the gradient is a constant vector. For such a piecewise constant vector field $\mathbf{w}$, the discrete divergence at a vertex $\mathbf{v}_i$ is defined as $(\mathrm{Div}\,\mathbf{w})(\mathbf{v}_i) = \sum_{T_k \in \mathscr{N}_{\mathscr{T}}(i)} \nabla \varphi_{ik} \cdot \mathbf{w}_k |T_k|$, where $\mathscr{N}_{\mathscr{T}}(i)$ is the set of all tetrahedra immediately adjacent to the vertex i, $\nabla \varphi_{ik}$

is the gradient of basis function at vertex i in tetrahedron T_k, $\mathbf{w}_k$ is the constant vector in T_k, and $|T_k|$ is the volume of tetrahedron. Finally, the discrete Laplacian operator is formulated as $\Delta u = \mathrm{Div}(\nabla u)$. But that needs a numerical process to form the global Laplacian operator matrix by computing the non-zero coefficients involved, because they did not reduce the definition to a simple vertex-based linear formula like the well-known cotangent weights Laplacian operator for triangular surface mesh, e.g., $\Delta u_i = \sum_{\mathbf{v}_j \in \mathcal{N}_\mathcal{V}(i)} k_{ij}(u_i - u_j)$, where $\mathcal{N}_\mathcal{V}(i)$ is the set of vertices immediately adjacent to vertex i, and $k_{ij} = \frac{1}{2}(\cot\alpha_{ij} + \cot\beta_{ij})$ is a scalar weight assigned to the edge (i,j) with α_{ij} and β_{ij} to be the two angles opposite to the edge [36]. To improve this problem, we deduced the vertex-based linear formula of volumetric Laplacian operator, with details in Appendix A, which turns out that the simple cotangent formula is still valid, by taking the dihedral angles of the tetrahedra:

$$\Delta u_i = \sum_{\mathbf{v}_j \in \mathcal{N}_\mathcal{V}(i)} \left(\frac{1}{6} \sum_{k=1}^{n} l_k \cot(\theta_k) \right) (u_i - u_j), \tag{3}$$

where $l_k = l_{pq}$ is the length of opposite edge (p,q) to which edge (i,j) is against, and $\theta_k = \theta_{pq}$ is the dihedral angle, such as shown in Fig. 9 in Appendix A. It should be noted that there are probably $n > 2$ tetrahedra incident to edge (i,j). Then one can set the volumetric mesh edge weight as $w_{ij} = \frac{1}{6}\sum_{k=1}^{n} l_k \cot(\theta_k)$, and get the same vertex-based linear formula form as the surface mesh case, which is considerably easier to use than the original definition above.

Now, we shall describe the practical implementation of the theoretical results presented above, which mainly involves the construction of the global volumetric gradient operator matrix, volumetric divergence operator matrix, and volumetric Laplacian operator matrix.

As the gradient inside each tetrahedron (i,j,p,q) is a constant vector $\mathbf{w}_k$, it is simple to verify that the follow equation holds:

$$\begin{bmatrix} (\mathbf{v}_i - \mathbf{v}_q)^T \\ (\mathbf{v}_j - \mathbf{v}_q)^T \\ (\mathbf{v}_p - \mathbf{v}_q)^T \end{bmatrix} \cdot \mathbf{w}_k = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} u_i \\ u_j \\ u_p \\ u_q \end{bmatrix},$$

where

$$\mathbf{w}_k = [\nabla\varphi_i, \nabla\varphi_j, \nabla\varphi_p, \nabla\varphi_q] \begin{bmatrix} u_i \\ u_j \\ u_p \\ u_q \end{bmatrix},$$

then the gradients of basis functions are

$$[\nabla\varphi_i, \nabla\varphi_j, \nabla\varphi_p, \nabla\varphi_q] = \begin{bmatrix} (\mathbf{v}_i - \mathbf{v}_q)^T \\ (\mathbf{v}_j - \mathbf{v}_q)^T \\ (\mathbf{v}_p - \mathbf{v}_q)^T \end{bmatrix}^{-1} \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}.$$

Take all m tetrahedra and n vertices of the mesh into consideration, one can formulate the volumetric gradient $\nabla u(x) = \sum_i u_i \nabla\varphi_i(x)$ using a global operator $\mathbf{G}$, expressed as a $3m \times n$ matrix that multiplies the n-vector of the discrete values u_i to obtain a vector of 3 m stacked gradients (each gradient having 3 spatial coordinates). Thus, one can write down the following formula for the coordinate scalar field $\boldsymbol{x}$ (same for $\boldsymbol{y}$ and $\boldsymbol{z}$) of input mesh:

$$\mathbf{G}\mathbf{x} = \mathbf{g}_x. \tag{4}$$

Assume $\mathbf{M}$ is a $3\,\mathrm{m} \times 3\,\mathrm{m}$ diagonal "mass" matrix of the volumes of the tetrahedra, it is simple to verify that matrix $\mathbf{G}^T\mathbf{M}$ corresponds to the volumetric divergence operator $(\mathrm{Div}\,\mathbf{w})(\mathbf{v}_i) = \sum_{T_k \in \mathcal{N}_\mathcal{T}(i)} \nabla\varphi_{ik} \cdot \mathbf{w}_k |T_k|$, and $\mathbf{G}^T\mathbf{M}\mathbf{G}$ is none other than the volumetric Laplacian matrix. Of course, there is no need to use this connective matrices operator, as we have deduced the vertex-based linear formula of volumetric Laplacian operator, e.g., Eq. (3), which can be easily represented as a sparse symmetric matrix $\mathbf{L}$ containing the edge weights w_{ij}:

$$\mathbf{L}_{ij} = \begin{cases} -\sum\limits_{v_k \in \mathcal{N}_\mathcal{V}(i)} w_{ik} & \text{if } i = j, \\ w_{ij} & \text{if } v_j \in \mathcal{N}_\mathcal{V}(i), \\ 0 & \text{if otherwise.} \end{cases} \tag{5}$$

Finally, we can get the discretized version of the Poisson equation (2) as

$$\mathbf{L}\mathbf{x}' = \mathbf{G}^T\mathbf{M}\mathbf{g}_x \tag{6}$$

and the same for the other two coordinate functions.

Note that the matrix $\mathbf{L}$ is singular whose rank is $n-1$ (assume the tetrahedral mesh is single connected) like the surface mesh case, in order to uniquely restore the global coordinates $\boldsymbol{x}'$, one needs to specify the coordinates of at least one vertex. This is of course not a problem, as the user will define certain boundary conditions to express their editing intention in the interactive application. As shown in Fig. 2, this typically means to fix certain mesh parts (usually on the surface) $\mathscr{F} \subset \mathscr{M}$, and to prescribe displacements for the so-called handle region(s) $\mathscr{H} \subset \mathscr{M}$. The boundary constraints are incorporated into system (6) by eliminating rows and columns corresponding to the constraint vertices $\mathbf{v}_i \in \mathscr{F} \cup \mathscr{H}$ by bringing them to the right-hand side. This yields a sparse, symmetric, and positive definite linear system

$$\mathbf{A}\mathbf{u} = \mathbf{b}. \tag{7}$$

As the same linear system is solved three times only for different right-hand side vector, and the coefficient matrix $\mathbf{A}$ is only dependent on the original tetrahedral mesh and the regions of boundary constraints $\mathscr{F} \cup \mathscr{H}$, the Cholesky factorization [37] is constructed only once when the regions of constraints are selected, while the user interaction with the handle region $\mathscr{H}$ requires solely updating the right-hand side vector and then solving by a quick back-substitution, which can achieve interactive rates.

However, the result of such editing approach is not satisfactory, as it ignores the fact that the gradient is dependent on the particular placement of the tetrahedral mesh in space, i.e., it is not rigid-invariant, and thus when the mesh deforms, the representation must be updated. In other words, the inherent lack of rotation and scale invariance of the linear operator requires local transformation of the volumetric differential properties.

3.2. Local transformation by volumetric harmonic field

To address the local transformation problem, Yu et al. [19] employed an explicit propagation scheme using geodesic field on the surface. The transformation matrix for the handle $\mathscr{H}$ is provided or deduced, e.g., from a transform UI attached to the handle where the user visually manipulates the rotation axes. Then the local transformations over the region of interest on the mesh are interpolated according to the geodesic distance from the handle. While, the discrete geodesic distances turned out to be a suboptimal parameter to propagate the transformations because they may be non-smooth and also attenuate the transformations of highly protruding features too much [32]. Instead, Zayer et al. [23] proposed to use harmonic fields on the surface for deformation guidance, which results in a smooth transformation propagation.

These ideas can be extended naturally to a volumetric domain as we have deduced the volumetric Laplacian operator.

A volumetric harmonic scalar field $\mathbf{s}$ defined over the tetrahedral mesh vertices is determined by the Laplace equation

$$\mathbf{Ls} = 0, \tag{8}$$

where the Dirichlet boundary conditions require $\mathbf{s}_i = 0$ for $\mathbf{v}_i \in \mathscr{F}$ and $\mathbf{s}_i = 1$ for $\mathbf{v}_i \in \mathscr{H}$. The resulting harmonic scalars smoothly blend between 0 and 1 with no local extrema other than constraint vertices. Note that there is additional advantage of economic computation, as the system matrix for harmonic field is the same matrix used for Poisson equation, the same factorization can be used.

At each vertex, the value $\mathbf{s}_i$ is used to blend between the transformation for the handle $\mathscr{H}$ (containing only rotational and scaling components) with the identity for the fixed part $\mathscr{F}$, which generates a smooth deformation field. By sampling the deformation field at the barycenters of the tetrahedra, we obtain a piecewise constant field, i.e., deformations per tetrahedron. Then each tetrahedron is deformed separately, which yields a deformed but fragmented, discontinuous mesh, from which the modified volumetric gradient field $\mathbf{g}'_x$ is computed. The deformed coordinates $\boldsymbol{x}'$ is solved by the new discretized Poisson equation

$$\mathbf{Lx}' = \mathbf{G}^T\mathbf{M}\mathbf{g}'_x \tag{9}$$

(and the same for the other two coordinate functions). It appears that the Poisson reconstruction stitches disconnected tetrahedra back together, resulting a satisfactory deformation.

However, this explicit propagation scheme is translation-insensitive [32], e.g., only works when an appropriate rotation transformation is specified at the handle in addition to translation. Fig. 3(a) illustrates a horizontal bar whose right end is translated with no rotational transformation, and the deformation with the explicit propagation scheme is of course unnatural and not satisfactory. We can see that the gradient vectors of $\boldsymbol{x}$ coordinates remain in original horizontal direction.

A possible solution to address this problem is to use a similar iterative differential update technique as Au et al. [29], who iteratively update the deformed surface Laplacian coordinate. That is, at each iteration, we can perform an analysis of previous deformation result to extract for each tetrahedron an estimate of its rotational transformation, then, modify the volumetric gradient field as discussed above and solve for new deformation mesh by Eq. (9). However, this direct and trivial update scheme does not work for the gradient field based deformation approach. Such as Fig. 3(b) shows, the gradient vectors of $\boldsymbol{x}$ coordinates have indeed changed their directions, but the Poisson reconstruction gives the same result as before.

Our key observation is that, the local rotations nearby the constraint boundary regions should have similar transformations as the boundary ($\mathscr{F} \cup \mathscr{H}$), e.g., identical transformations in current cases. While, all of these estimated rotational transformations from previous deformation are almost the same, which leads to all the volumetric gradient vectors almost have the same change, thus the Poisson reconstruction dose not work properly as before. To address this problem, we provide a modified update scheme: first, build the volumetric harmonic field as before and sample the scalar value $\boldsymbol{s}$ at the barycenters of each tetrahedron; then, do a hat-like map to the interval [0, 1], that is, map the old value 0 and 1 to the new value 0, while the old value 0.5 to the new value 1, e.g., a linear map $(1 - 2 \cdot ABS(s - 0.5))$, where ABS represent an absolute value function, or a non-linear map $COS(\pi \cdot ABS(s - 0.5))$, where COS represent the cosine function; finally, these new scalars are used as weights to interpolate the estimated rotational transformations and the identical transformation. Fig. 3(c) demonstrates the new updated gradient vectors, and the Poisson reconstruction works properly this time.

Fig. 4(a)–(d) illustrate a horizontal bar with translational constraints at the iteration step 1, step 6, step 10, and step 20, using the linear map weights and (e)–(h) using the cosine map weights. The non-linear map tends to produce more natural deformation results. In our experiments, these iterations usually converge in less than 10 steps.

We further point out that for the surface Laplacian coordinates, the iterations may lead to accumulating tangential drift if applied to the original geometry, making it necessary to work on the dual mesh to avoid these problems. While we readily avoid this for working on the tetrahedral configurations, thus stable even under extreme conditions.

3.3. Inhomogeneous deformation by material weighted laplace

As discussed before, to generate physically plausible deformations for inhomogeneous objects such as soft tissue, one must take the properties of the underlying materials into consideration. We follow the spirit of [28] and use a volumetric scalar field m to represent the material stiffness. Then, the basic minimization formula (1) can be modified as

$$\iiint_{\Omega} m \| \nabla x' - \mathbf{g}_x \|^2 \, dV. \tag{10}$$

The result is that local regions with large material stiffness will undergo less shape deformation, and local regions with small material stiffness will allow for more shape deformation to take place. This provides an intuitive mean of controlling the deformation of inhomogeneous objects, and is agree with the setup of those physics based methods, e.g., some material setting

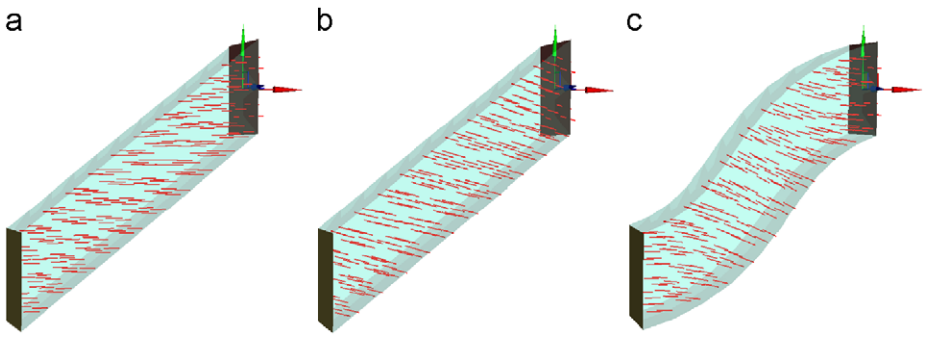

Fig. 3. A horizontal bar whose right end is translated with no rotational transformation. (a) The deformation with the explicit propagation scheme, whose gradient vectors of $\boldsymbol{x}$ coordinates remain in original horizontal direction. (b) Direct application of the estimated rotational transformations from previous deformation can update the volumetric gradient field, but the deformation remains as before. (c) Our new blending scheme properly updates the volumetric gradient field, and improves the deformation result.

methods in physics based methods can be used directly for our approach.

The discretized version of the material weighted Poisson equation turns out to $\mathbf{G}^T\mathbf{MDGx}' = \mathbf{G}^T\mathbf{MDg}_x$, where $\mathbf{D}$ is a $3\,\mathrm{m} \times 3\,\mathrm{m}$ diagonal weighted matrix containing the mesh cell material stiffness m_k in each tetrahedron T_k . As $\mathbf{M}$ is also a $3\,\mathrm{m} \times 3\,\mathrm{m}$ diagonal "mass" matrix of the volumes of the tetrahedra, it is convenient to take them together and define a new diagonal matrix $\mathbf{M}'$ containing the weights $m_i|T_i|$. In addition, a similar derivation to the Appendix A can be performed to define the discrete material weighted volumetric Laplacian operator:

$$\Delta' u_i = \sum_{\mathbf{v}_j \in \mathcal{N}_{\mathcal{T}}(i)} \left(\frac{1}{6} \sum_{k=1}^{n} m_k l_k \cot(\theta_k) \right) (u_i - u_j). \tag{11}$$

Thus, the final material weighted Poisson equation is

$$\mathbf{L}'\mathbf{x}' = \mathbf{G}^T\mathbf{M}'\mathbf{g}'_x, \tag{12}$$

where $\mathbf{L}'$ is the new sparse symmetric Laplacian matrix containing above material weighted coefficients. The solving procedure of the new system is precisely the same as the original system Eq. (6) or Eq. (9).

The additional advantage of the new material weighted volumetric Laplacian operator is that we can maintain the convenience of using harmonic field for deformation propagation guidance, by using $\mathbf{L}'\mathbf{s} = 0$. In other words, the same factorization can still be used, and our inhomogeneous meth deformation framework requires solving essentially only one linear system.

Note that the same derivation can be applied to the triangle surface mesh, and the corresponding discrete material weighted Laplacian operator turns out to

$$\Delta' u_i = \frac{1}{2}(m_\alpha \cot \alpha_{ij} + m_\beta \cot \beta_{ij})(u_i - u_j), \tag{13}$$

where m_α and m_β are the material stiffness of the two triangles adjacent to edge (i,j). We thus call both of them as *material weighted cotangent Laplacian operators*. Fig. 5 demonstrates an inhomogeneous 360° twist of a bar with two different regions of materials, the advantages of using volumetric deformation over surface deformation is the same as before.

We further point out that our integration of the mesh cell material stiffness parameter into the geometric aware cotangent Laplacian operator is not only convenient, but also is necessary. The material weighted quadratic minimization for blending anchor transformations in [28] is essentially a Laplace equation weighted by the stiffness parameters [32]. Like the uniform Laplacian operator, it takes neither edge lengths nor angles into account, and cannot provide a good approximation for irregular meshes. Fig. 6 illustrates a bar with different regions of mesh density under 360° twist. Fig. 6(a) shows the propagation harmonic field generated by the geometric aware cotangent Laplacian, while Fig. 6(b) by the non-geometric Laplacian. Fig. 6(c) and (d) illustrate the corresponding deformation results (Note that the Laplacian operator for Poisson reconstruction is the same). We can see that the second harmonic field value pattern is

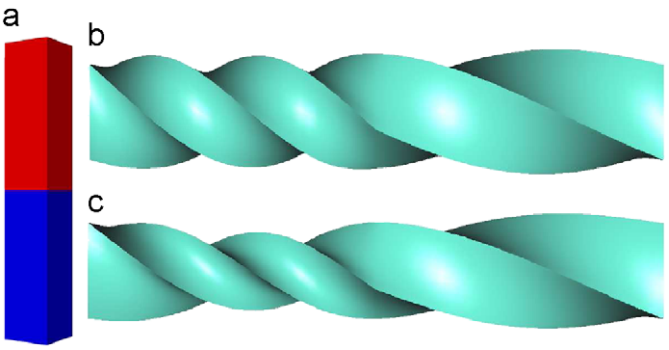

Fig. 5. (a) A bar with two different regions of materials, the red region is more soft than the blue region. Inhomogeneous 360° twist deformation for (b) tetrahedral volumetric mesh and (c) triangle surface mesh.

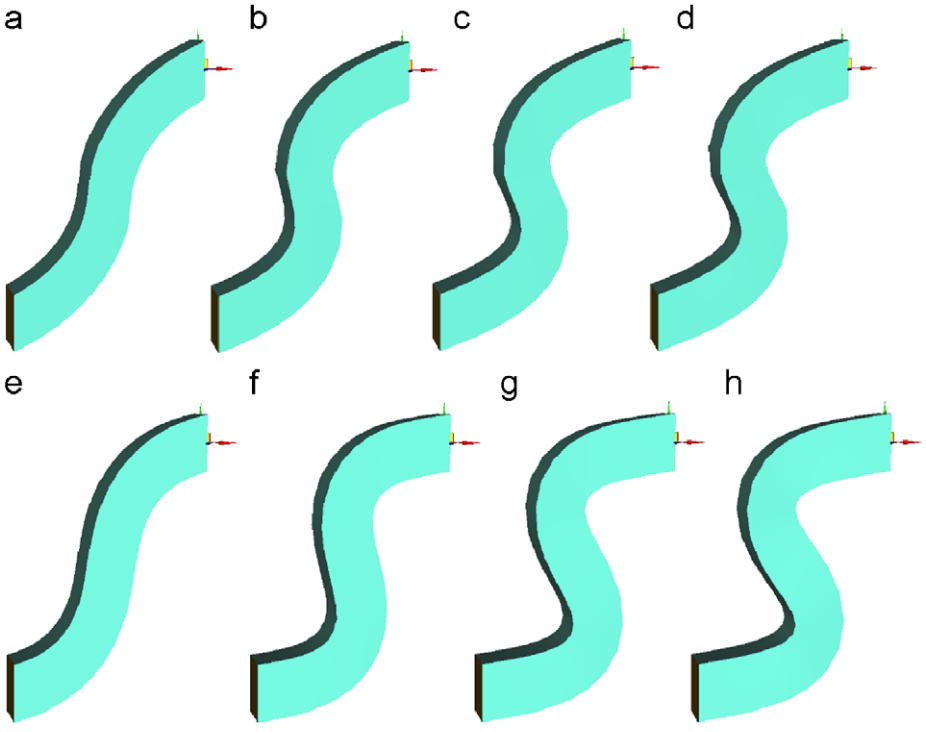

Fig. 4. (a)–(d) A horizontal bar with translational constraints at the iteration step 1, step 6, step 10, and step 20, using the linear map weights and (e)–(h) using the cosine map weights.

asymmetrical, which leads to unnatural shrinking effect in some area, such as the region pointed by the red arrow in Fig. 6(d), whereas the geometric aware cotangent weighting enables to eliminate the influence of the meshing bias, which leads to smooth and natural deformation result.

4. Maxillofacial surgery simulation

We shall now describe the procedure of a typical maxillofacial surgery simulation.

The results of lateral cephalometric analysis on the 2D X-ray image (Fig. 7(a)) and 3D anatomical measurements on the reconstructed model from CT data suggested that a mandibular advancement surgery should be performed. The stand treatment is a procedure called bilateral sagittal split ramus osteotomies (BSSRO) to pull forward the anterior inferior mandible on the horizontal plane of mandibular dentition, such as illustrated in Fig. 7(b). The problem is how much should the displacement be. In other words, one needs a tool to predict the facial soft tissue change or deformation resulting from bone repositioning boundary conditions, with interactive performance, to determine a best surgical planning.

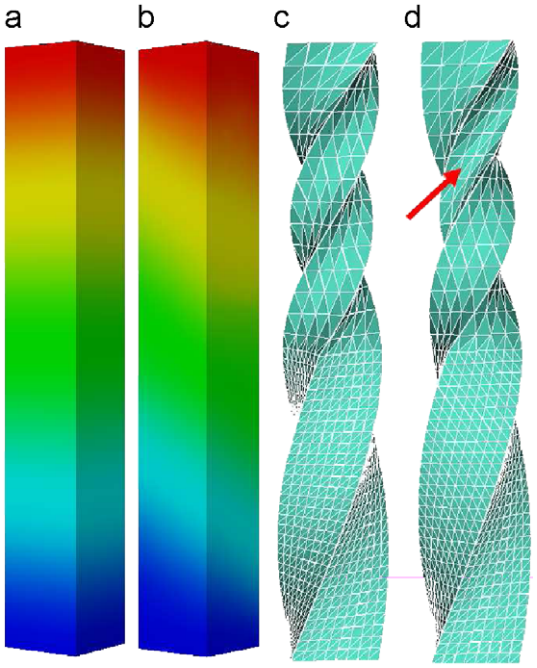

Fig. 6. Twist of a bar (360°) with different regions of mesh density: (a) shows the propagation harmonic field generated by the geometric aware cotangent Laplacian (red color represents value 1, blue color represents value 0), (b) by the non-geometric Laplacian. (c) and (d) show the corresponding deformation results.

To address this problem, the tetrahedral volumetric mesh of the facial soft tissue was constructed from CT data using a modified biomechanical semi-automatic mesh generator [38], such as demonstrated in Fig. 8(a). The interface region with the moving anterior inferior mandible was defined as the handle $\mathscr{H}$ with corresponding displacement, and the interface with other craniofacial bone was set as the fixed part $\mathscr{F}$. The deformation result is illustrated in Fig. 8(b) using the best bone repositioning plan by interactive comparing.

Because the post-operative CT data are currently unavailable, the FEM simulation result is considered to be the ground truth as its prediction achieves relatively high accuracy [18]. The soft tissue was assumed to behave as an inhomogeneous linear elastic continuum, and the element-by-element material property assignment was accomplished by finding the number of CT scan pixels lying within each element as well as the gray values [39]. These inhomogeneous materials were directly applicable for our deformation approach.

Fig. 8(c) shows the FEM deformation result, with color mapping of the difference with Fig. 8(b). The statistical results demonstrated that the peak vertex difference was below 0.8 mm, and the average vertex difference was only about 0.05 mm. The simulation time of FEM was about 30 min, while our deformation achieved interactive performance (Table 1).

5. Conclusions

This paper presented a novel inhomogeneous volumetric mesh deformation approach by gradient field manipulation. The volumetric deformation approach inherits the strengths of recent surface differential deformation techniques: it is very efficient, simple to implement, and robust. By working in the volumetric domain instead of on the mesh surface can effectively impose volumetric constraints to avoid unnatural volume changes and local self-intersections, and achieve better deformation propagation for most of bone repositioning boundary conditions in surgery simulation. In addition, the system can deal with both explicit rotational constraints and/or only translational constraints to drive the deformation, thus is very easy to use.

In particular, we proposed the material weighted cotangent Laplacian operator to generate physically plausible deformations for inhomogeneous objects such as soft tissue. The inhomogeneous deformation framework requires solving essentially only one linear system and is very efficient. The method can provide real-time interactive performance for surgery simulation, yet the quality of simulation is comparable to FEM based model.

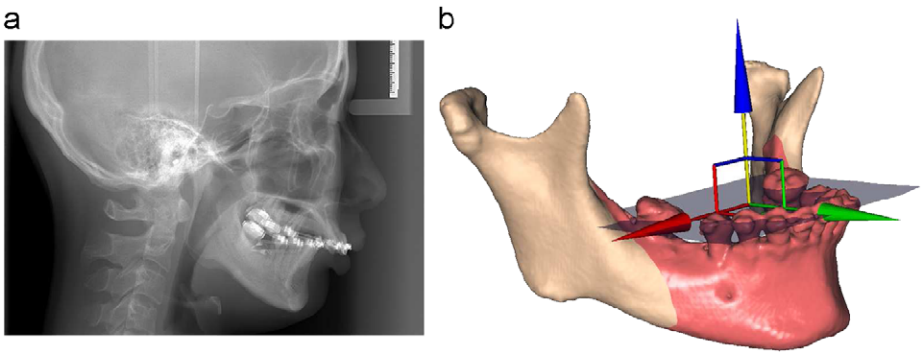

Fig. 7. (a) 2D lateral X-ray image. (b) Bilateral sagittal split ramus osteotomies.

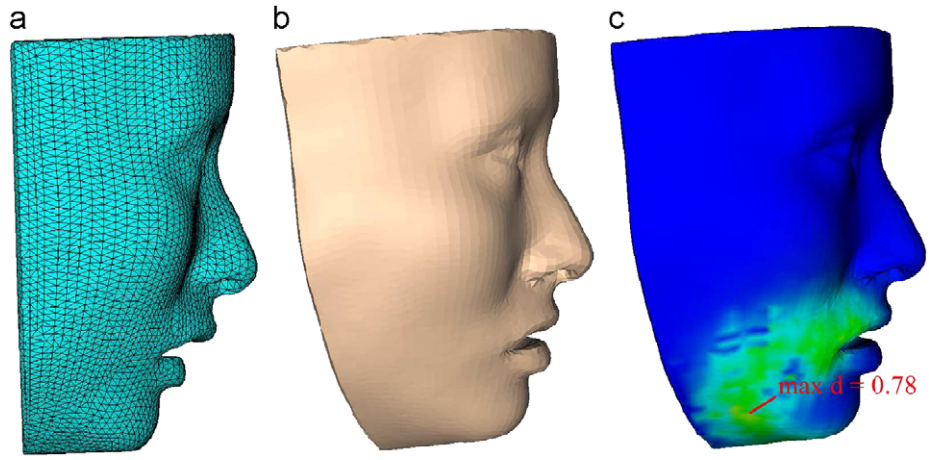

Fig. 8. Tetrahedral mesh of the facial soft tissue. (b) Our volumetric mesh deformation. (c) FEM deformation, with color mapping of the difference to (b).

Table 1
Statistics and timing on a 3 GHz Intel Pentium IV with 4 Gb of RAM.

Object	Vertices	Tetrahedra	Factorization time (s)	Three times back substitution (s)
Test 1	2541	12,000	0.094	0.016
Test 2	7381	36,000	0.438	0.047
Test 3	12,896	67,500	5.031	0.234
Test 4	22,491	120,000	7.375	0.406
Test 5	34,476	187,500	16.797	0.838
Surgical face	9232	28,596	0.094	0.031

Although designed for biomechanical tissues and computer surgery simulation, the inhomogeneous volumetric mesh deformation approach could be used directly in other virtual and simulation tasks in computer graphics.

In the near future we will validate this new model on a database of several patients who had a facial surgery, and compare the predicted facial contour with the real post-operative result.

Acknowledgments

The research was supported by the National Basic Research Program (No. 2006CB303106) of China, the National Natural Science Foundation (No. 60873126) of China, and Doctorial subject special scientific research fund of Education Ministry of China (No. 20070335074).

Appendix A. Discrete volumetric Laplacian operator

In this section, we simplify the discrete Laplacian formula in [35]: $\Delta u = \mathrm{Div}(\nabla u)$, and reduce it to the simple form of vertex-based Laplacian operator (Fig. 9).

From the definition of gradient $\nabla u(x) = \sum_i u_i \nabla \varphi_i(x)$ and divergence $(\mathrm{Div}\,\mathbf{w})(\mathbf{v}_i) = \sum_{T_K \in \mathcal{N}_{\mathcal{T}}(i)} \nabla \varphi_{ik} \cdot \mathbf{w}_k |T_k|$, we get $\Delta u_i = \sum_{T_K \in \mathcal{N}_{\mathcal{T}}(i)} \nabla \varphi_{ik} \cdot (\sum_j u_{jk} \nabla \varphi_{jk}) |T_k|$. First consider the case of one tetrahedron $T_k = (i,j,p,q)$, such as shown in Fig. 9. The gradient of hat basis function is $\nabla \varphi_i = -\mathbf{n}_i / H_i = -\mathbf{n}_i S_i / 3|T|$, and $S_i = -\sum_{j \neq i} \mathbf{n}_i \cdot \mathbf{n}_j S_j$, $\mathbf{n}_i \cdot \mathbf{n}_j = -\cos\theta_{pq}$, $H_i = h_{ipq} \sin\theta_{pq}$, where θ_{pq} is the dihedral angle of the opposite edge (p,q) to which edge (i,j) is against. Therefore, the following equations hold:

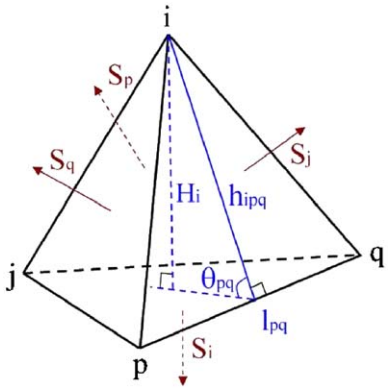

Fig. 9. Illustration for discrete volumetric Laplacian operator.

$$
\begin{aligned}
&\nabla \varphi_i \cdot \left(\sum_j u_j \nabla \varphi_j \right) |T| \\
&= \frac{-\mathbf{n}_i S_i}{3|T|} \left(\frac{-\mathbf{n}_i S_i}{3|T|} u_i + \frac{-\mathbf{n}_j S_j}{3|T|} u_j + \frac{-\mathbf{n}_p S_p}{3|T|} u_p + \frac{-\mathbf{n}_q S_q}{3|T|} u_q \right) |T| \\
&= \frac{1}{9|T|} \left(-\sum_{j \neq i} \mathbf{n}_i \cdot \mathbf{n}_j S_i S_j (u_i - u_j) \right) \\
&= \frac{1}{3 S_i H_i} \left(\sum_{j \neq i} \cos\theta_{pq} S_i \left(\frac{1}{2} l_{pq} h_{ipq} \right) (u_i - u_j) \right) \\
&= \sum_{j \neq i} \frac{1}{6} l_{pq} \cot\theta_{pq} (u_i - u_j),
\end{aligned}
$$

where l_{pq} is the length of opposite edge (p,q).

Take all tetrahedra incident to vertex $\mathbf{v}_i$ into consideration, one can get the simple vertex-based form of discrete volumetric Laplacian operator:

$$
\Delta u_i = \sum_{\mathbf{v}_j \in \mathcal{N}_{\mathcal{V}}(i)} \left(\frac{1}{6} \sum_{k=1}^{n} l_k \cot(\theta_k) \right) (u_i - u_j),
$$

where $\mathcal{N}_{\mathcal{V}}(i)$ is the set of vertices adjacent to vertex $\mathbf{v}_i$, $l_k = l_{pq}$ is the length of opposite edge (p,q) to which edge (i,j) is against, and $\theta_k = \theta_{pq}$ is the dihedral angle. It should be noted that there are probably $n > 2$ tetrahedra incident to edge (i,j).

References

[1] Zachow S, Gladilin E, Zeilhofer HF, Sader R. Improved 3D osteotomy planning in cranio-maxillofacial surgery. In: Lecture notes in computer science (MICCAI), vol. 2208; 2001. p. 473–81.

[2] Terzopoulos D, Walters K. Physically-based facial modelling, analysis and animation. Visualization and Computer Animation 1990;1(4):73–80.

[3] Waters K. A physical model of facial tissue and muscle articulation derived from computer tomography data. In: Proceedings of the visualization in biomedical computing. Bellingham: SPIE; 1992. p. 574–84.

[4] Keeve E, Girod S, Girod B. Computer-aided craniofacial surgery. In: Proceedings of the computer assisted radiology CAR'96. Paris, France; 1996. p. 757–62.

[5] Delingette H. Toward realistic soft-tissue modeling in medical simulation. IEEE Special Issue on Virtual and Augmented Reality in Medicine 1998; 86(3):512–23.

[6] Teschner M, Girod S, Girod B. Optimization approaches for soft-tissue prediction in craniofacial surgery simulation. In: Proceedings of the second international conference on medical image computing and computer-assisted intervention. Cambridge, England; 1999, p. 1183–90.

[7] Teschner M, Girod S, Girod B. Direct computation of nonlinear soft-tissue deformation. In: Proceedings of the vision, modeling, visualization'00. Saarbrucken, Germany; 2000. p. 383–90.

[8] Meehan M, Teschner M, Girod S. Three-dimensional simulation and prediction of craniofacial surgery. Orthodontics and Craniofacial Research 2003;6(Suppl.1):102–7.

[9] Koch RM, Gross MH, Carls FR, von Buren DF, Frankhauser G, Parish YIH. Simulating facial surgery using finite element models. In: Proceedings of the SIGGRAPH. New York: ACM; 1996. p. 421–8.

[10] Keeve E, Girod S, Kikinis R, Girod B. Deformable modelling of facial tissue for craniofacial surgery simulation. Computed Aided Surgery 1998;3(5): 228–38.

[11] Sarti A, Gori R, Lamberti C. A physically based model to simulate maxillofacial surgery from 3D CT images. Future Generation Computer Systems 1999;15(2):217–21.

[12] Zachow S, Gladiline E, Hege HC, Deuflhard P. Finite element simulation of soft tissue deformation. In: Proceedings of CARS; 2000. p. 23–8.

[13] Chabanas M, Luboz V, Payan Y. Patient specific finite element model of the face soft tissues for computer-assisted maxillofacial surgery. Medical Image Analysis 2003;7(2):131–51.

[14] Gladilin E. Biomechanical modeling of soft tissue and facial expressions for craniofacial surgery planning. PhD thesis, FU Berlin, Germany; 2003.

[15] Westermark A, Zachow S, Eppley B. Three-dimensional osteotomy planning in maxillofacial surgery including soft tissue prediction. Journal of Craniofacial Surgery 2005;16(1):100–4.

[16] Cotin S, Delingette H, Ayache N. A hybrid elastic model allowing real-time cutting, deformations and force-feedback for surgery training and simulation. The Visual Computer 2000;16(8):437–52.

[17] Mollemans W, Schutyser F, Nadjmi N, Suetens P. Very fast soft tissue predictions with mass tensor model for maxillofacial surgery planning systems. In: Proceedings of CARS; 2005. p. 491–6.

[18] Mollemans W, Schutyser F, Nadjmi N, Maes F, Suetens P. Predicting soft tissue deformations for a maxillofacial surgery planning system: from computational strategies to a complete clinical validation. Medical Image Analysis 2007;11:282–301.

[19] Yu Y, Zhou K, Xu D, Shi X, Bao H, Guo B, et al. Mesh editing with poisson-based gradient field manipulation. ACM Transactions on Graphics 2004;23(3): 644–51.

[20] Lipman Y, Sorkine O, Cohen-Or D, Levin D, Rossl C, Seidel H-P. Differential coordinates for interactive mesh editing. In: Proceedings of the shape modeling international. Silver Spring, MD: IEEE Computer Soc. Press; 2004: p. 181–90.

[21] Sorkine O, Cohen-Or D, Lipman Y, Alexa M, Rossl C, Seidel H-P. Laplacian surface editing. In: Proceedings of the Eurographics/ACM SIGGRAPH symposium on geometry processing. New York, NY, USA: ACM; 2004. p. 175–84.

[22] Sumner RW, Popovic J. Deformation transfer for triangle meshes. ACM Transactions on Graphics 2004;23(3):399–405.

[23] Zayer R, Rossl C, Karni Z, Seidel H-P. Harmonic guidance for surface deformation. In: Computer graphics forum, Proceedings of Eurographics 2005. Dublin, Ireland: Eurographics, Blackwell; 2005. p. 601–9.

[24] Igarashi T, Moscovich T, Hughes JF. As-rigid-as-possible shape manipulation. ACM Transactions on Graphics 2005;24(3):1134–41.

[25] Lipman Y, Sorkine O, Levin D, Cohen-Or D. Linear rotation-invariant coordinates for meshes. ACM Transactions on Graphics 2005;24(3):479–87.

[26] Zhou K, Huang J, Snyder J, Liu X, Bao H, Guo B, et al. Large mesh deformation using the volumetric graph Laplacian. ACM Transactions on Graphics 2005;24(3):496–503.

[27] Botsch M, Sumner R, Pauly M, Gross M. Deformation transfer for detail-preserving surface editing. In: Proceedings of the vision, modeling, and visualization. 2006; p. 357–64.

[28] Popa T, Julius D, Sheffer A. Material-aware mesh deformations. In: Proceedings of the shape modelling international. Silver Spring, MD: IEEE Computer Soc. Press; 2006. p. 141–52.

[29] Au OK-C, Tai C-L, Liu L-G, Fu H-B. Dual Laplacian editing for meshes. IEEE Transactions on Visualization and Computer Graphics 2006;12(3):386–95.

[30] Huang J, Shi X-H, Liu X-G, Zhou K, Wei L-Y, Teng S-H, et al. Subspace gradient domain mesh deformation. ACM Transactions on Graphics 2006;25(3): 1126–34.

[31] Lipman Y, Cohen-Or D, Gal R, Levin D. Volume and shape preservation via moving frame manipulation. ACM Transactions on Graphics 2007;26(1):5.

[32] Botsch M, Sorkine O. On linear variational surface deformation methods. IEEE Transactions on Visualization and Computer Graphics 2008;14(1):213–30.

[33] Yan H-B, Hu S-M, Martin RR, Yang Y-L. Shape deformation using a skeleton to drive simplex transformations. IEEE Transactions on Visualization and Computer Graphics 2008;14(3):693–706.

[34] Hu S-M, Li C-F, Zhang H. Actual morphing: a physical-based approach for blending two 2d/3d shapes. In: ACM symposium on solid modeling and applications. Genova, Italy; 2004. p. 309–14.

[35] Tong Y, Lombeyad S, Hirani A, Desbrun M. Discrete multiscale vector field decomposition. ACM Transactions on Graphics 2003;22(3):445–52.

[36] Pinkall U, Polthier K. Computing discrete minimal surfaces and their conjugates. Experimental Mathematics 1993;2:15–36.

[37] Toledo S. TAUCS: a library of sparse linear solvers. Tel Aviv University; 2003.

[38] Liao S-H, Tong R-F, Dong J-X. Anisotropic finite element modeling for patient specific mandible. Computer Methods and Programs in Biomedicine 2007; 88(3):197–209.

[39] Taddei F, Pancanti A, Viceconti M. An improved method for the automatic mapping of computed tomography numbers onto finite element models. Medical Engineering & Physics 2004;26:61–9.