An Improved Facial Orthopedic Surgery Planning System with Pre-processing FEM Modeling

Lei Pan, Lixu Gu, Zhennan Yan, Sizhe Lv, Bo Zhu
Image Guided Surgery and Therapy Laboratory, Shanghai Jiao Tong University
800 Dongchuan Road, Minhang District, Shanghai, 200240, China
lionpan@sjtu.edu.cn

Abstract

Finite Element Method(FEM), which is based upon continuum mechanics and the physics of materials, is one of the most widely used deformation method in digital biomedical area. Unlike other light-weighted deformation models (Mass-Spring, Free Form, Chainmail, BEM, etc)using in other areas, FEM gets the most accurate deforming results in the cost of large scale of computation. In this paper, we present an improved Facial Orthopedic Surgery Planning system using pre-processing FEM modeling. This planning system calculates and simulates the bone-reshaped surgical result of facial soft issue deformation more quickly. According to the specific demands of the facial surgery planning system, we designed and implemented FEM modeling in our own way and improved the computational efficiency. Finally we validate and represent the deformation results, compare our method and results with the traditional one.

1. Introduction

Bone-related facial orthopedic surgery became more and more acceptable and welcomed by people who care about their outlook. Since the human face plays an important role in interpersonal relationships and gives a first impression to others, people are very sensitive and prudent to changes to their outlook. As a result, a Facial Orthopedic Surgery Planning System which provides fast and accurate enough simulative results of the surgery is of great values[1].

1.1. Improved Finite Element Method

The Finite Element Method (FEM) is a widely used and common way for computation of deformation. It is employed in lots of areas including industrial design and construction, material modeling and reshaping, durative evaluation. The continuous constitutive equations that describe the physical behavior of a solid body can be analyzed by dividing the body into a set of discrete sub-domains called elements (e). Constitutive behavior can be solved analytically in each element, and combined to estimate the global solution. The accuracy of deformation result is based on self-contained physical and mathematical model and satisfies the needs of the patients who want to see the detail.

Conventional FEM costs extreme computation and large memory usage in solving a system containing more than a few hundred mesh nodes. Moreover, its implementation is not as straightforward as for some other light-weighted but less-accurate methods, such as mass-spring models, implicit surface models (Free-form), ChainMail models and Boundary Element Method. Hybrid models based on global parameterized deformations and local deformations based on FEM have been introduced to solve the problem[2], however the precision of the deformation can not be guaranteed.

1.2. Linear and nonlinear FEM

Linear FEM and most of the other methods mentioned above are only applicable to linear deformations and valid for small displacements. However, nonlinear FEM[3] is more computational complex therefore inappropriate for realtime simulation of human facial deformation. In this paper, we mainly discuss improved nonlinear FEM with preprocessing of matrix operation and boundary-interior vertices separation aimed at speed-up the computation and reduce the scale of the matrix. Facial orthopedic surgery planning system is our target project and no large displacements beyond the restriction of linear plastic deformation would be possible. All the demands of this project decide that an improved linear FEM would be suitable enough. A comparison between improved linear FEM and other deformation methods will be presented in this paper.

1.3. The facial orthopedic surgery planning system

Our facial orthopedic surgery planning system calculates and simulates the facial orthopedic surgical results through operating on patients' real skulls data and total facial data, which were obtained through CT or MRI and then recon-

ICAT 2008
Dec. 1-3, Yokohama, Japan
ISSN: 1345-1278

structed by segmentation and meshing(Data reconstruction and remeshing are also parts of our research but beyond the scope of this paper). System users can easily use UI interactive tools to select the region of interests and execute basic orthopedic surgical operation. All these operation can be canceled and redone until a satisfied surgical result on skull is obtained.

After manual operation on the virtual vertices on skull, vertices with displacement on skull are mapped to the closest soft issue and recorded[4]. Deformation on the whole face including the muscle and other soft issue will be quickly calculated and then displayed. The simulated results after the surgery can be sent to patients themselves for feedback without waiting for a long time. The total cost of resource and time for the whole planning process is reduced.

2. Related work

2.1. Previous surgical planning and prediction system

The idea of implementation of a facial orthopedic surgery planning system and simulation the result of the facial orthopedic surgery is not new in and of itself. Conventional or specific FEM applications in clinical soft issue deformation on human body were also through long period development. As early as 1996, [R.M.Koch et al. 1996]a prototype system for surgical planning and prediction of human facial shape after craniofacial and maxillofacial surgery for patients with facial deformities.

2.2. Data meshing

FEM needs clinical craniofacial and skull data with finite elements of patients while such clinical data is always extracted from MR or CT image[5]. Initial data preprocessing, reconstruction, and registration are performed and some meshing algorithms are employed. Finally the mesh data of the craniofacial and skull of the patients with finite tetrahedron elements is constructed. All of the mesh data used in our experiment were created by Sizhe Lv using Delaunley and improved Delaunley algorithms, [Sizhe Lv et al. 2006].

3. Methodology

Our facial orthopedic surgery planning system combines and adopts different methods in different parts of the system[6]. According to the requirements of doctors and patients, all the methods and algorithms we selected and developed are the most suitable for the demands of our system.

Our system is a bone-related orthopedic surgery planning system. Users execute filling or cutting operations on skull and the changes of the skull shape cause the total soft issue deformation. Since skull is rigid, vertices on soft issue (such as muscle) closed to the deformed skull can be seen as the boundary condition for FEM model.

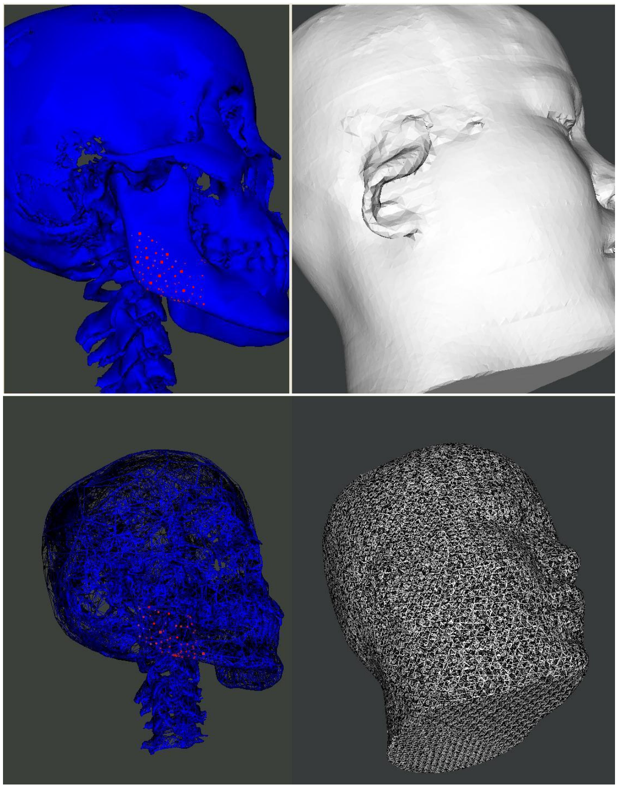

Figure 1. Selection of ROI(skull vertices marked red) on both rendering mode and meshing mode

3.1. Skull operation

Before Skull operation begins, it is necessary to specify parts of the skull data to be Region Of Interests(ROI)(Fig.1). The number of vertices of ROI determines the scale of later computation. Common clinical ROI of facial orthopedic surgery are cheekbone, mandible and frontal bone. The boundary between ROI and rest of the skull are the non-displacement vertices.

We employed Cubic Bézier curves as a basic tool in our skull operation (for both cutting and filling). We used at least two curves to create the final discretional surface for system user to decide how the skull would be operated. These two Cubic Bézier curves share one of those control points(Fig.2). For more detail operative surface, extension of each of these two Cubic Bézier curve is provided. An accurate surface is always determined by more than two extended curves respectively on two perpendicular planes.

The control points of the Cubic Bézier curves can be located and edited directly by system user interface(Fig.3). Every Cubic Bézier curve is created by at least four control

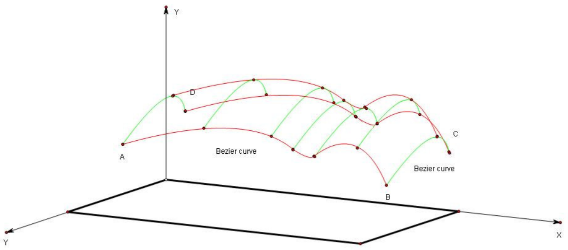

Figure 2. A surface created by two Cubic Bézier curves on two perpendicular planes to accurately determine the surgical operation

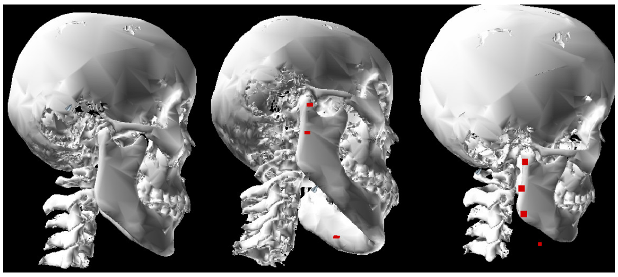

Figure 3. Adding control points of the Cubic Bézier curve inside or outside the skull vertices

points in the space. Users are able to adjust the final operative surface by moving, transforming, rotating any of the control points of each of the Cubic Bézier curves therefore each part of the surface is modifiable(Fig.4). All operated vertices on the skull are in a single 3-dimension ROI. After the final operative surface is fixed, vertices will be mapped onto the surface, filling or cutting operation will be determined(Fig.5), pre-processing, such as vertices separation, vertices reordering, will be executed in a background thread for the specific ROI.

The parametric form of the Cubic Bézier curve is:

$B(t) = (1-t)^3 P_0 + 3t(1-t)^2 P_1 + 3t^2(1-t) P_2 + t^3 P_3, t \in [0, 1]$

While four points P_0, P_1, P_2 and P_3 in the plane or in three-dimensional space. Two Cubic Bézier curve in two different such planes share P_0 or P_3 and form the final surface.

In order to stably use FEM and avoid complex remeshing operation, we need to preserve the topology of the original skull data after vertices operation. The number of the vertices can not be increased nor decreased. Triangle structures can not be destroyed nor created. Displacements on each dimension of each modified vertices are recorded after both cutting and filling operation.

While skull data is modified, vertices mapping from original position to new position is employed. One of the feasible method is directly interpolation(Fig.6). However, in some complex clinical situation, directly mapping can

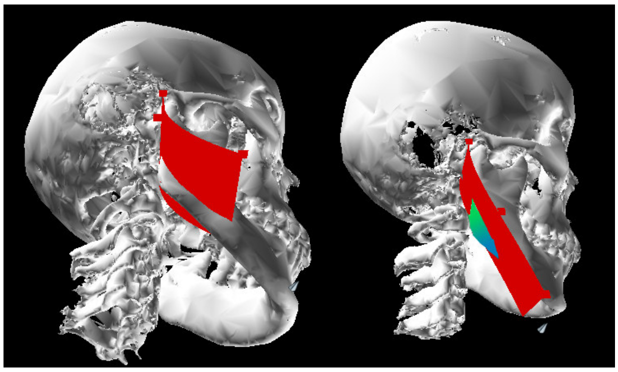

Figure 4. Adjust the control points to modify the operative surface. Some failed control points that formed an ill-defined surface need to be relocated

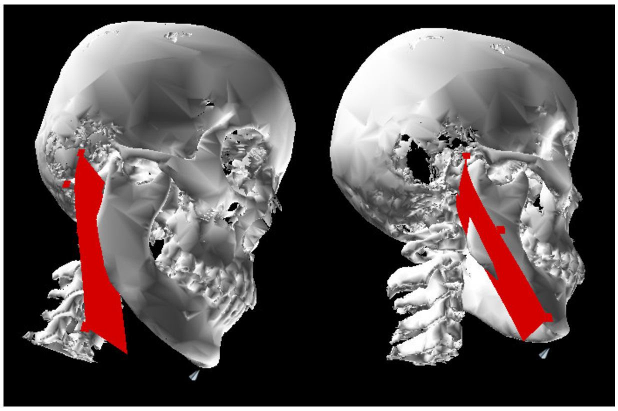

Figure 5. Calculation and simulation of the cutting operation after fix the surface

cause problems(Fig.7a). It is possible that the structure of some triangles is broken, which may cause the collapse of tetrahedron and FEM solving model[7].

One or more intermediary surfaces can solve this problem(Fig.7b). By inserting the intermediary surfaces, vertices mapping is simplified. The order of vertices interpolation to target surface can be preserved. The shape of the intermediary surface is calculated after each iteration and structure check. The algorithm iterates until no triangle edges intersect others.

3.2. Improved FEM application

This section discusses the finite element method we use to calculate the facial deformation. Finite element modeling is a fundamental engineering method with many different types of elements proposed in the past. We focus on the linear static FEM, which is based on the principle of minimum potential energy, with tetrahedron elements and propose a combination of several effective techniques to reduce the computation cost and improve the efficiency of deformation.

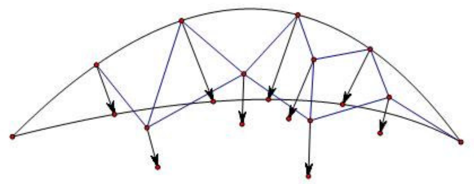

Figure 6. Example of direct vertices mapping using simple interpolation on smooth surgical surface

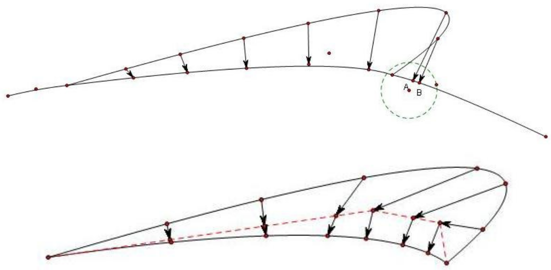

Figure 7. Examples of vertices mapping without(a) and with(b) intermediary surface on complex surgical surface

Since patients are more concerned about the final visual result of the orthopedic surgery than inner computation detail, our target is to reduce the amount of computation but provide a quick rending of result without losing any accuracy. However the extreme computational complexity is the bottleneck conventional FEM and prevents its application. We combined following methods to achieve this target:

(1)Separating the vertices of all tetrahedrons into surface group and internal group.

One effective method to speed-up the process is to reduce the size of the stiffness matrix K. In most mesh data, the number of internal vertices, which would not be displayed in the rendering result, is far more than the number of surface vertices. In the final linear matrix equation

$$\mathbf{K}x = \mathbf{P}$$

Internal vertices take up large proportion of the stiffness matrix and cost large computational complexity in the equation solving process, which is the most time-cost process of the whole FEM model.

We adjust the order of surface vertices and internal vertices and group them, then rewrite the above matrix equation to

$$\begin{pmatrix} \mathbf{K}_{bb} & \mathbf{K}_{bi} \\ \mathbf{K}_{ib} & \mathbf{K}_{ii} \end{pmatrix} \begin{pmatrix} x_b \\ x_i \end{pmatrix} = \begin{pmatrix} \mathbf{P}_b \\ \mathbf{P}_i \end{pmatrix}$$

While subscript b means boundary vertices, and subscript i means internal vertices. The stiffness matrix $\mathbf{K}$ is rewritten into $\mathbf{K_{bb}}$, $\mathbf{K_{ib}}$, $\mathbf{K_{bi}}$ and $\mathbf{K_{ii}}$. The displacement vector x and load vector $\mathbf{P}$ are also divided into two groups. We focus on surface part $\mathbf{K_{bb}}$, x_b and $\mathbf{P_b}$.

Since we separated the matrix, the algebra agglomeration is necessary to ensure the result of matrix equation remain the same. From above equation, we get

$$x_i = K_{ii}^{-1}(P_i - K_{ib}x_b)$$

Put this equation into the first line of matrix equation, we get

$$(K_{bb} - K_{bi}K_{ii}^{-1}K_{ib})x_b = P_b - K_{bi}K_{ii}^{-1}P_i$$

Let

$$K'_{bb} = K_{bb} - K_{bi}K_{ii}^{-1}K_{ib}$$
$$P'_b = P_b - K_{bi}K_{ii}^{-1}P_i$$

We get the reduced form of the matrix equation:

$$K'_{bb}x_b = P'_b$$

(2)Matrix triangular factorization

Although the size of the matrix equation is reduced, solving the linear matrix equation still takes up much of processing time. Kinds of methods were proposed to overcome this problem in computer science field. In order to sequentially reduced the complexity of the matrix equation, we employed advanced Cholesky matrix triangular factorization.

$$\begin{pmatrix} K_{11} & K_{12} & \cdots & K_{1n} \\ K_{21} & K_{22} & \cdots & K_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ K_{n1} & K_{n2} & \cdots & K_{nn} \end{pmatrix} = \begin{pmatrix} L_{11} & 0 & \cdots & 0 \\ L_{21} & L_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ L_{n1} & L_{n2} & \cdots & L_{nn} \end{pmatrix} \cdot \begin{pmatrix} D_{11} & 0 & \cdots & 0 \\ 0 & D_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & D_{nn} \end{pmatrix} \cdot \begin{pmatrix} U_{11} & U_{12} & \cdots & U_{1n} \\ 0 & U_{22} & \cdots & U_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & U_{nn} \end{pmatrix}$$

Then the matrix equation can be rewritten in

$$U_{bb}x_b = D_{bb}^{-1}L_{bb}^{-1}P'_b$$

In this form, U_{bb} is upper triangular matrix, L_{bb} is lower triangular matrix, D_{bb} is diagonal matrix. Solving the triangular matrix equation can be easily achieved by calculating the inverted matrix of U_{bb}, which is much more convenient than solving the original equation.

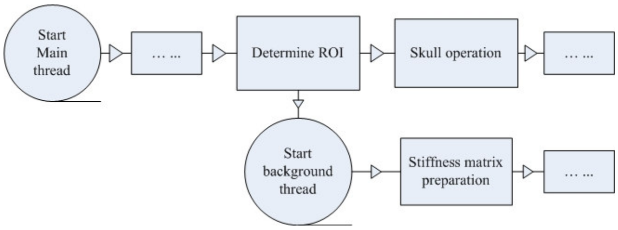

Figure 8. Processing flow of multi-threading. Background thread starts when ROI is determined.

(3)Execution of pre-process in background threads.

The above methods can greatly reduce the final equation solving time in the cost of matrix separation, agglomeration and factorization. However, all these operation can be done in a background thread when ROI is determined(Fig.8). In most condition, ROI only takes up parts of total mesh data and can be quickly determined in an early stage far before the surgical surface is fixed. (Usually it takes quite a long time to select the fianl surgical surface before both surgeon and patients satisfy when background thread has enough time to complete all the computation.) Since the rest of the region is considered to be static, all the variants and constants needed in previous operation are provided from the tetrahedron in ROI.

In final stage of the equation solving process, boundary condition is recorded by system after users fixed the surgical surface and complete the cutting or filling operation. The displacement of the vertices on the tetrahedron, which were mapped to the modified skull, is inserted to U_{bb}.Effective load can be imported to vector $\mathbf{P}$ according to the environment.

(4)CUDA, MKL and symmetric band storage

The implementation of conventional FEM is complicated when using the strain matrix ε, stress matrix σ, Poisson ratio and Youngs module to prepare of the element stiffness matrix $\mathbf{K_e}$ and assembling them to total stiffness $\mathbf{K}$(or directly assemble them to K_{bb}, K_{ii}, K_{ib} and K_{bi}). Although it is difficult to effectively accelerate these repetitive basic computation, CUDA can more or less help. CUDA framework supports general purpose GPU parallel computation which is suitable for large floating data. GPU executes the same computation on each tetrahedron element(Fig.9).

$$K^e = \int_{V^e} B^T DBdV = B^T DBV^e$$

Under the video memory limits, for large matrices, CUDA framework shows remarkable performance. However, the advantage is unconspicuous for small data.

Another factor that prevents FEM from light-weighted application is the memory. The storage of the stiffness matrix K in memory can be extremely huge for a mesh data

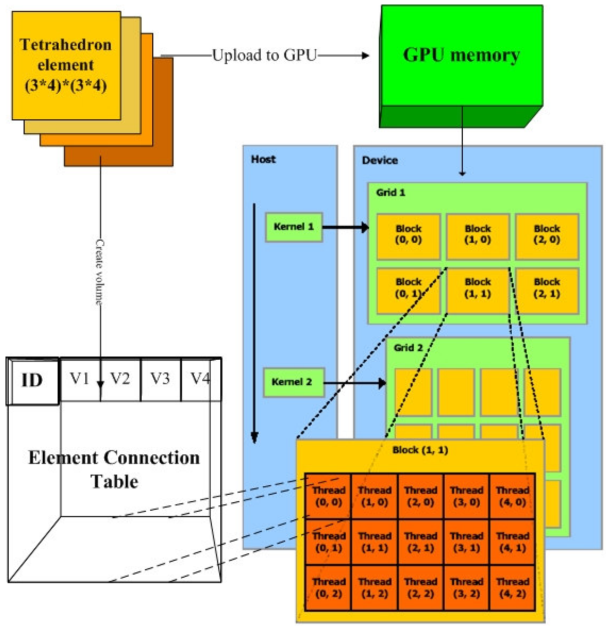

Figure 9. Data transfer and allocation on GPU and parallel K^e computation in CUDA

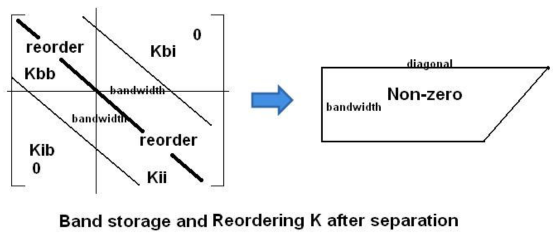

Figure 10. Reordering after the matrix separation reduces the bandwidth of each part. Symmetric band storage format saves the memory by storing the low triangle of the stiffness matrix within the bandwidth by column.

with thousands of tetrahedrons. Symmetric band storage format supported by Intel MKL can effectively save memory by omitting large amount of zero elements far away from diagonal(Fig.10).

In order to reduce the bandwidth of the stiffness matrix, both internal vertices and surface vertices need to be reordered. Tetrahedrons that have effects on each other should be placed together. Intel MKL Sparse BLAS Level 2 and Level 3 provide series of high performance functions that support basic sparse matrix computation.

4. Experimental results

Our facial orthopedic surgery planning system and all related experiments were developed and test on DELL Desktop Optipex 320 with 1.5GB of RAM, 1.80GHz Genuine In-

Tetra number	CUDA(GPU)	CPU
Kidney(1860tetras)	78ms	103ms
Heart(6630tetras)	92ms	598ms
Mandible(10637tetras)	110ms	1477ms

Table 1. Time cost of K^e computation on GPU and CPU

Operation	single thread	multi-thread
ROI and surface selection	1min(avg)	1min(avg)
Reordering	none	143ms
Ke,K	231ms	231ms
Separation	none	74ms
Agglomeration	none	101ms
Factorization	none	1496ms
Boundary Condition	161ms	77ms
Load calculation	11ms	4ms
Solving(MKL)	8389ms	653ms
Total	8792ms	2779ms

Table 2. Time cost in different stage on mandible surgery by running traditional FEM and improved FEM

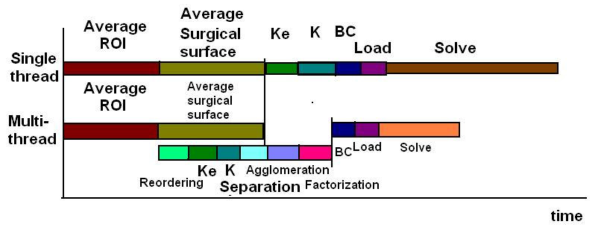

Figure 11. Stage time cost in FEM processing by using single-thread and multi-thread(Time lengths are not labeled by proportion)

tel(R) CPU 2160, NVIDIA(R) GeForce 8800 512MB video memory, Microsoft Windows XP Professional Service Pack 2 OS.

Besides facial orthopedic surgery simulation, we also tested our algorithms and methods in other different soft organs such as kidney and heart. By manually specifying some internal vertices on tetrahedrons, we simulated the cutting and filling operation on bone(Table.1).

By creating multi-threads for pre-process, we made full use of dual-core Intel(R) CPU 2160 and save time for final matrix equation solving(Fig.11). Without losing any computational accuracy, both surgeons and patients can receive a simulative rendered results in an acceptable time, which allow them to adjust the surgery times again.

In an example of facial orthopedic surgery simulation, ROI is selected in patient's right mandible. Raise edge of patient's mandible is cut(Fig.12) and the deformation of the muscle and skin is then calculated(Fig.13).

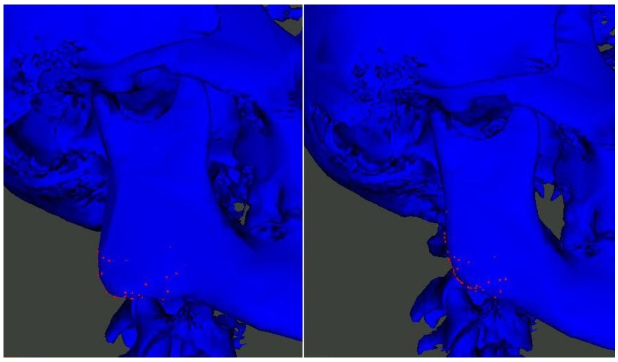

Figure 12. Mandible shapes before and after cutting operation using a suitable simulative surgical surface

Figure 13. Final simulative results before and after the cutting surgery

5. Conclusion

In this paper, we represented an improved facial orthopedic surgery planning system using multi-thread technique and several matrix operation for pre-process, using Bézier curves for surgical surface creating. Our method provides quick and accurate simulative results by reducing matrix size in advance and simplifying the equation solving complexity. Table.2 shows the time cost in each stage and indicates the advantage of our method over convenient FEM.

Although kinds of well-developed methods were proposed to improve orthopedic surgery planning system, linear static FEM still has its limitation. In other simulations which are more complex than static orthopedic surgery, linear FEM can not satisfy the reality needs. Future work includes non-linear FEM modeling and application in virtual surgery and reducing the complexity of non-linear FEM system.

Acknowledgement

This paper is partially supported by the Chinese National

Natural Science Foundation under Grant No.60571061. Kidney, heart and head data is provided by Shanghai Renji hospital. Tetrahedron mesh data from CT image is provided by Sizhe Lv.

Reference

[1]W. Mollemansa, F. Schutysera, N. Nadjmib, F. Maesa and P. Suetens: Predicting soft tissue deformations for a maxillofacial surgery planning system: From computational strategies to a complete clinical validation, Medical Image Analysis Volume 11, Issue 3, June 2007, Pages 282-301

[2]R. M. Koch, M. H. Gross, F. R. Carlsy: Simulating Facial Surgery Using Finite Element Models, ACM-0-89791-746-4/96/008, 1996

[3]Hualiang Zhong, Mark P.Wachowiak, Terry M. Peters: A real time finite element based tissue simulation method incorporating nonlinear elastic behavior, Computer Methods in Biomechanics and Biomedical Engineering, Vol. 8, No. 3, June 2005, 177C189

[4]M. Kauer, V. Vuskovic, J. Dual, G. Szekely, M. Bajka: Inverse Finite Element Characterization of Soft Tissues, MICCAI 2001, LNCS 2208, pp. 128C136, 2001

[5]Fulvia Taddei, Saulo Martelli, Barbara Reggiani, Luca Cristofolini, Marco Viceconti: Finite-Element Modeling of Bones From CT Data: Sensitivity to Geometry and Material Uncertainties, IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, VOL. 53, NO. 11, NOVEMBER 2006

[6]Zhennan Yan, Lixu Gu, Pengfei Huang, Sizhe Lv, Xiao Yu, Xianming Kong: Soft Tissue Deformation Simulation in Virtual Surgery using Nonlinear Finite Element Method, EMBC 2007

[7]Xucheng Wang, Finite Element Method, Qing Hua University Publishing Company, 2003

[8]Duysak, A. and Zhang, J. J. 2004. Fast Simulation of Deformable Objects. In Proceedings of the information Visualisation, Eighth international Conference on (Iv'04) - Volume 00 (July 14 - 16, 2004). IV. IEEE Computer Society, Washington, DC, pp.422-427

[9]Huang, P., Gu, L., and Zhang, S.: Real-Time Simulation for Global Deformation of Soft-Tissue Using Deformable Centerline and Medial Representation. ISBMS (2006)67-74

[10]Shaoting Z, Lixu G, Weiming L, Jingsi Z, Feng Q: Real-time virtual surgery simulation employing MM-Model and Adaptive spatial hashing, ICAT 2006