DEVELOPMENT OF A NOVEL COMPUTER-AIDED SURGICAL SIMULATION (CASS) SYSTEM FOR ORTHOGNATHIC SURGERY

A Thesis

by

PENG YUAN

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

Chair of Committee,	Zixiang Xiong
Committee Members,	Anxiao Jiang
	Jim Ji
	Serap Savari
Head of Department,	Chanan Singh

August 2014

Major Subject: Computer Engineering

ABSTRACT

A successful craniomaxillofacial (CMF) surgery depends not only on surgical techniques, but also on an accurate surgical plan. However, the traditional planning methods have proven problematic. Towards this end, we have designed a clinical protocol of using 3D Computer-Aided Surgical Simulation (CASS) method as a new planning method. A computerized composite skull model of a patient is generated to accurately represent the skeleton, the dentition and facial soft tissues. In addition, patient's neutral head posture (NHP) is recorded and the composite skull model is transferred into NHP coordinate system. Furthermore, the virtual osteotomies can be performed to simulate an orthognathic surgery. The surgical splints and templates are generated in the computer, fabricated by a rapid prototyping machine, and used during surgery to precisely position the bony segments.

The ultimate goal of this project was to design, develop and implement a computer CASS software system that allows surgeons to plan an orthognathic surgery following our clinical protocol. The system includes six modules: Segmentation, Registration/NHP, Cephalometric Analysis, Virtual Osteotomy Surgical Simulation and Surgical Splint/Template. Microsoft Visual Studio 2008, VTK and ITK were used to develop our CASS system.

DEDICATION

To My Beloved Wife Dr. Yan Zhang.

ACKNOWLEDGEMENTS

I came to realize that a master program is an extremely long process of hard work, which requires considerable dedication to complete the thesis. Encouragement and support through this lonely process is essential. I am very fortunate to have many colleagues and friends who have helped me along the journey. It would have been impossible for me to bring this thesis to the final stage without them.

I would like to express my deepest gratitude to my thesis advisor, Dr. Zixiang Xiong PhD, for providing me the opportunity to engage in this master program, for giving me research training and for his excellent guidance, caring, patience and inspiring confidence in me. He has spent so much of his time reading the manuscript and given me advice to improve the content. Discussions with him, whether formal or informal, academic or non-academic, always bring me fun and learning opportunities.

My highest gratitude goes to my supervisor, Professor Dr. James Xia MD PhD MS. He made it possible for me to undertake this challenging project, and provided his kind guidance. Despite his busy administrative, and research schedules, he was always available when I needed help or advice. He spent lots of his time to discuss with me about the schedule and procedures of this project, and revised my thesis many times. My sincere appreciation is to his initiation, encouragement and supervision of my thesis.

This computer-assisted system would not be complete without Dr. Jianfu Li PhD, never hesitated with his kind advice to my question. He saved me many hours for seeking the solution. He as a good friend and colleague was always willing to help and give his best suggestions. It would have been a lonely lab without him.

I cordially express my thanks to Dr. Ken-Chung Chen DDS MS, Dr. Zhen Tang DDS MS PhD, and Dr. Xiaoyan Zhang PhD in Dr. Xia's laboratory for their invaluable support, valuable suggestion and help.

My sincere thanks specially extend to Dr. Xiaobo Zhou PhD for his encouragement, insightful discussion, and invaluable assistance in time of difficulties.

I would like to thank my committee members, Dr. Serap Savari, Dr. Jim Ji and Dr. Anxiao Jiang for their guidance and support throughout the course of this research.

My whole-hearted appreciation goes to my parents and parents-in-law for their encouragement.

Last but certainly not least, my love goes to my wife Yan and daughter Amber. They are always there cheering me up and stood by me all the time. They surely are my best accomplishment in my life.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

Craniomaxillofacial (CMF) Surgery is surgery to correct facial deformities, which includes dentofacial deformities, congenital deformities, combat injuries, post-traumatic defects, defects after tumor ablation, and deformities of the temporomandibular joint (TMJ). Because of the complex anatomy and involvement of soft facial tissue, muscles and bones, it is a challenge to surgeons. A successful CMF surgery depends not only on the technical aspects of the operation, but also to larger extent, on the formulation of a precise surgical plan [1-8]. During the past 50 years, the technical aspects of surgery have achieved significant improvements, such as rigid fixation, resorbable material, distraction osteogenesis and so on. However, the planning method, an important step before operation, hasn't been updated for years even there are many well-known problems [2-3], [8-9].

1.1 Problems of Traditional Planning Methods in CMF Surgery

Generally, the traditional surgical planning process involves four stages. The 1st stage is to gather data from different sources, which include physical examination, medical photographs, plain x-rays (e.g. cephalogram, orthopantomogram), CT, mounted plaster dental models, and other studies. It provides basic but very important information for a successful planning.

The 2nd stage is cephalometric analysis [19] and quantification of the deformity. Surgeon examines the patient in natural head position (NHP) and performs cephalometric analysis, study of the dental and skeletal relationships in the head with the cephalogram oriented to the Frankfort Horizontal plane. Surgeon then visualizes the bony and soft tissue anatomy on a CT that has been obtained in supine position. Finally, surgeon evaluates the occlusion on plaster dental modes that have been mounted to the Axis-Orbital plane. There are three problems for this approach. The first problem is that there are three different reference coordinate systems involved in each step. They are around 7 degrees apart from each other. The second problem is that the cephalogram is two-dimensional. The right and left anatomical structures are compressed onto the midsagittal plane. However, 75% of human faces have some degrees of asymmetry. The compressed 2D information greatly decreases the surgeon's ability to analyze the deformity. The third problem is that the cephalometric analysis is also two-dimensional because it is based on the 2D cephalogram. As described above, asymmetry may significantly distort the measurement results. (ref Gateno et al 2011 JOMS)

The 3rd stage of traditional planning methods is to simulate the surgery. In craniomaxillofacial surgery, this stage may begin with prediction tracings and be completed by moving the bones to desired position. A drawback of prediction tracing is its two-dimensional method. Therefore, it is impossible to simulate surgery in three-dimensional, which is much more accurate compared with 2D. In addition, 3D CT images have been useful in allowing the surgeons to visual the patient's condition, but they do not render the teeth with accuracy which is necessary for surgical simulation. Dental model is made to establish the occlusion of upper and lower teeth. A drawback of it is that it does not depict the surrounding bony structures. Therefore, it is impossible for surgeons to visualize the skeletal effects of a surgery on the facial skeleton, which is critical in the treatment of complex CMF deformities. Another way of simulating surgery is to use CT-based physical models produced by rapid prototyping techniques (e.g., stereolithography [STL]). Although these models are useful, they have three disadvantages: (1) Artificial teeth such as metal implant of digital CT dental model will affect the accuracy for surgical simulation of cases involving the occlusion; (2) It is impossible to simulate different surgeries with a single model. Once the model is cut, it is impossible to undo it. (3) These models can be fairly expensive. Furthermore, prediction tracing, dental model and CT model surgeries are separated processes which are time-consuming.

The final stage of the traditional surgical planning process is transferring surgical plan to operation room. Surgical splints are used when dentition jaw is involved in the surgery. These splints help the surgeon place the osteotomized jaw bone segments to a desired position. In case the dentition is not involved, surgeons currently do not have an accurate method of transferring the plan to the operating room. Certain point-to-point measurements are taken during the planning process. During the surgery, the same measurements are made intraoperatively. The placement of the bone segments to a desired position is more of an art than a science.

In summary, there are many problems associated with the traditional planning methods for orthognathic surgery. Each of them may be just minor but all together can really make surgical outcome far away from ideal. Figures 1 and 2 showed two examples of the surgical outcomes achieved with the traditional planning method.

Figure 1. Pre- and post-operative comparison of a successful orthognathic surgery (Diagnosis: relative symmetrical dentofacial deformity). The patient was treated with double-jaw surgery that was planned using the traditional planning methods.

Figure 2. Pre- and Post-operative comparison of orthognathic surgery with insignificant outcome (Diagnosis: hemifacial microsomia - asymmetrical dentofacial deformity). The patient was also treated double-jaw surgery that was also planned using traditional planning methods.

1.2 Development of CASS Clinical Protocol for CMF Surgery

Problems associated with the traditional planning methods have led us to develop a new CASS protocol for CMF surgery [10-15]. In this protocol, a computerized composite skull model of a patient is generated to accurately represent skeleton, dentition, and facial soft tissue. In addition, the patient's NHP is recorded and transferred to composite skull model. Furthermore, our model can also help user perform virtual osteotomies and simulate orthognathic surgery. The surgical splints and templates are generated within the computer, fabricated by a rapid prototyping machine. They can be used at the time of the surgery to accurately position the bony segments.

1.3 Purpose of the Study

The ultimate goal of this project is to design, develop and implement a computer CASS software system that allows surgeons to plan an orthognathic surgery following our clinical protocol. The system includes six modules: Segmentation, Registration/NHP, Cephalometric Analysis, Virtual Osteotomy Surgical Simulation and Surgical Splint/Template. Microsoft Visual Studio 2008, VTK and ITK were used to develop our CASS system.

2. DATA COLLECTION FOR CASS SYSTEM

Based on our clinical protocol, our CASS system contains two major sequential stages: data collection and surgical planning. The data collection is usually completed by a surgeon or a clinician. It is then processed by the CASS system. Once the data is prepared, surgeon uses the same CASS system to virtually plan an orthognathic surgery. Since the clinical data has to be collected in a specific way, it is described in details in this chapter. The detailed description on how the CASS software system is designed and implemented will be described in Chapter Three.

Preoperative record collection includes dental impressions, stone dental models, clinical examination and photographs, bite-jig fabrication and NHP recording. The main idea behind this stage is that surgeon can use bite-jig fabrication to combine all the preoperative records that are collected separately. Data collection is done by the following four subsections.

2.1 Dental Impression and Pour Dental Models

Stone dental models poured from the dental impressions are the replica of the teeth. The stone dental models will be digitalized and combined with the CT data in the surgical planning.

2.2 Clinical Examination

Routine clinical examination (Figure 3) provides surgeon with extremely valuable dynamic information that cannot be obtained from any other imaging source. It is important to note that the decision of correcting maxillary midline deviation and vertical position should be based on clinical examination, rather than the study images. Clinical photographs should be taken when the patent is in NHP and with a plumb line in the background, together with the clinical measurements of the occlusal cant, degree of dystopia, and ear position, they are also used to verify whether the 3D models are correctly oriented to the NHP during the surgical planning.

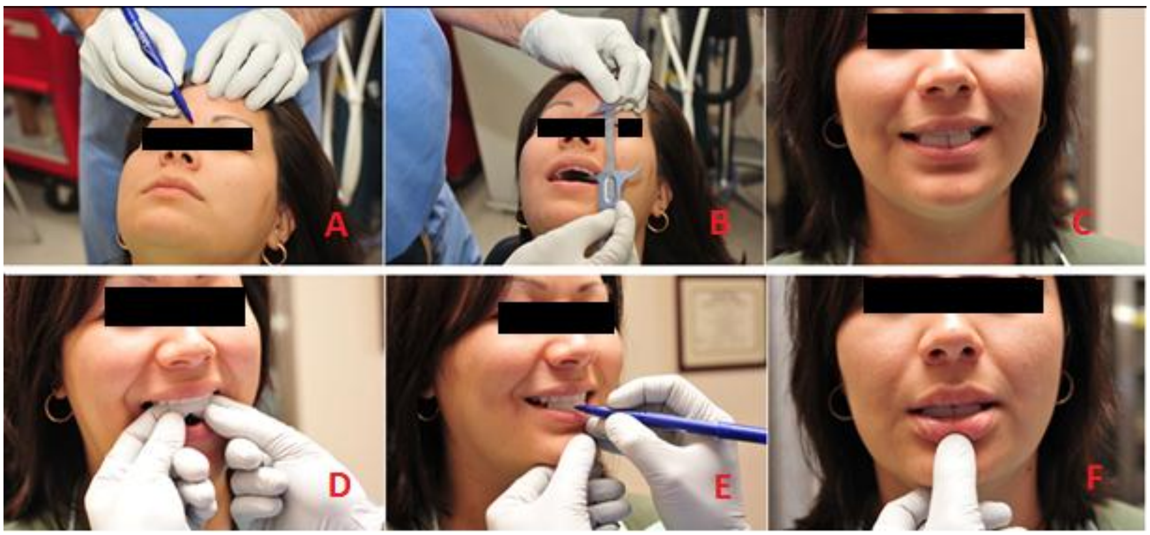

Figure 3. Clinical examination processes is illustrated from (A) to (F).

2.3 Bite-Jag Fabrication

This step is to fabricate a patient-specific bite-jig. This patient-specific bite-jig is used to maintain the maxillomandibular relationship constant during the patient's CT/CBCT scan, recoding NHP and incorporate digital dental model scan into the CT models. A face-bow with a set of fiducial markers is attached to the bite-jig. These markers are served as fiducial markers to register the digital dental models to the 3D CT/CBCT models (Figure 4).

Figure 4. (A) Stone dental model with patient-specific bite-jig. (B) Patient uses patient-specific bite-jig for CT scan.

2.4 NHP Recording

NHP is critical during surgical planning, as it defines the frame of reference. This step is to record a patient's NHP in pitch, roll, and yaw using a digital orientation sensor (3DM, MicroStrain Inc, Williston, VA) which is attached to the face-bow (Figure 5). We use either self-balanced orientation or physician-manipulated orientation to attain the NHP.

Figure 5. NHP recording of a patient using patient-specific bite-jig.

3. DESIGN, DEVELOPMENT AND IMPLEMENT OF CASS SYSTEM

Once the data is collected, orthognathic surgery can be planned and simulated by six interface modules (Figure 6) in our CASS system. CASS system is designed as a three-tier infrastructure, including interface, control and core library (Figure 7). Interface is used to provide human-computer interaction using six interface modules. Control is to connect core library and interface with message management. Core library is designed to store image data sets, declaring global massages and variables, providing methods of transformation, segmentation, and simulation, etc. The design principle, data organization and algorithm implementation process of our CASS system will be described below in details.

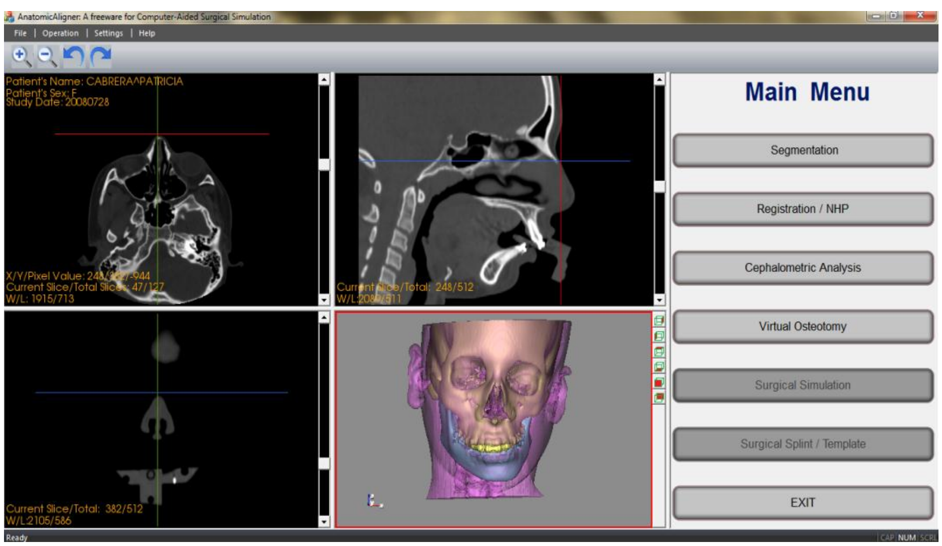

Figure 6. Main interface of CASS system. It includes six modules for surgeon to plan a surgery. The buttons with light gray means functions of modules have been started.

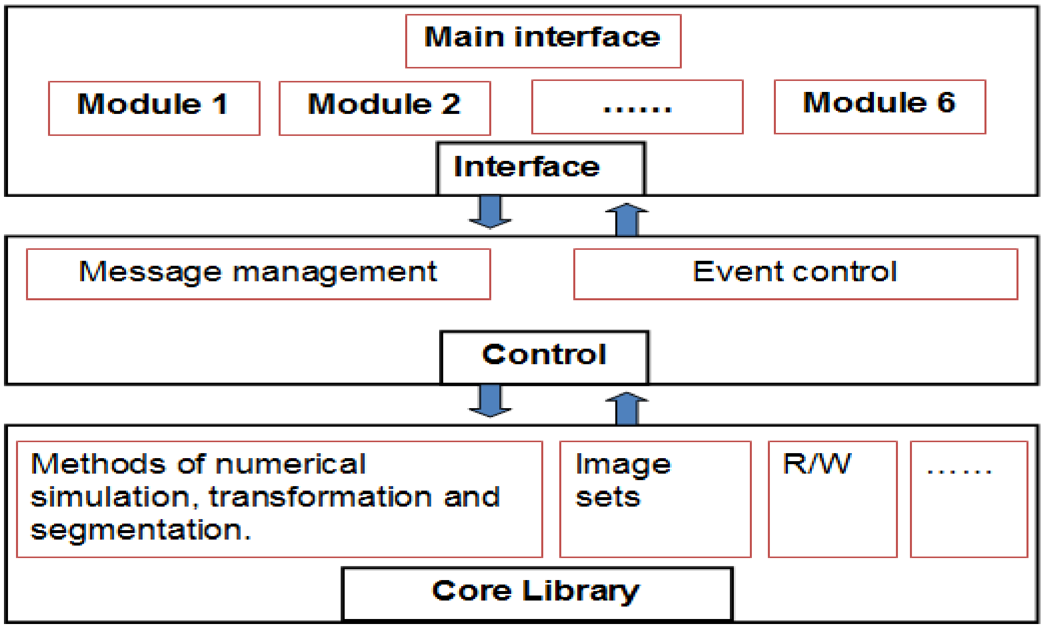

Figure 7. The logical structure of our CASS system.

3.1 Module 1: Creation of Composite Skull Model

Although CT image is excellent for generating bone models, it is incapable of accurately representing the teeth to the accuracy that is necessary for surgical planning. Furthermore, orthodontic metal brackets and dental restorations may cause severe streak artifacts and affect several slices that contain images of teeth in reconstructed 3D images. Therefore, the main function of this module is to generate 3D CT model and replace the CT teeth with the high-resolution digital dental model. This is called composite skull model [16-17]. The following models are used to generate a composite skull model: CT midface model, CT mandible model, CT soft tissues model, CT markers model, upper dental model, lower dental model, and teeth markers model. The orthognathic surgery then can be simulated and planned only once this composite skull model is generated. The process of creating composite skull model includes following four steps:

Step 1: CT Data Input. Original CT image sequence is imported in CASS system, and then reconstructed and displayed in three orthogonal views: axial, coronal and sagittal. CT image sequence usually includes two types of files: DICOM file and DICOMDIR file. DICOM file is used to store patient's name, the type of scan, image data, etc. DICOMDIR file is defined as a directory object, whose objective is to serve as an index for organizing and finding DICOM files inside a physical storage media. Since DICOM image files may be stored in different folders, a simple, efficient and interactive operation method is implemented in the system (Figure 8). The reconstructed sagittal and coronal views along with original axial view are shown in Figure 6.

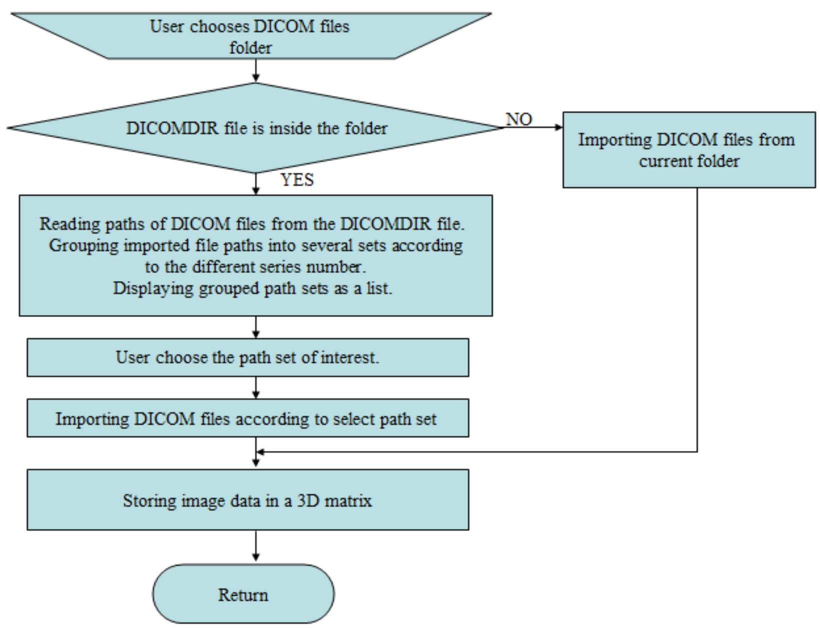

Figure 8. The process of loading DICOM images into the CASS system.

Step 2: Segmentation and 3D Reconstruction. Segmentation methods are implemented in the system to segmenting different anatomical structures from the CT scan. Threshold segmentation method (Figure 9) is implemented for initially generating bitmap masks (skull or soft tissues image sets) of a given anatomical structure. Different skulls and soft tissue can be marked with different colors automatically. Editing is performed by the surgeon using manual editing panel (Figure 10) and regional growing panel to separate a bitmap mask into several bitmap masks (such as midface, mandible, fiducial markers). Once the masks are generated and finalized, they are used to create 3D mesh datasets by Marching Cube Algorithm [18] (Figure 11).

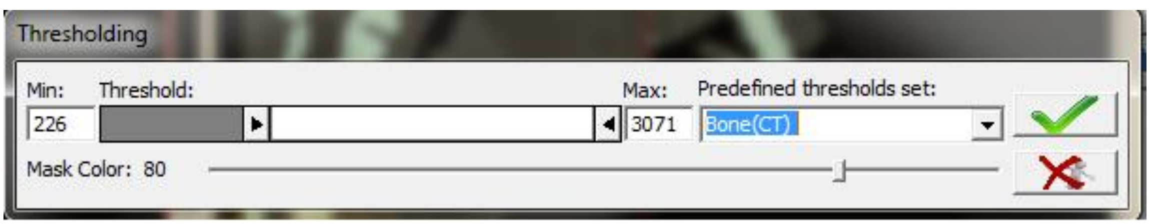

Figure 9. Threshold segmentation panel.

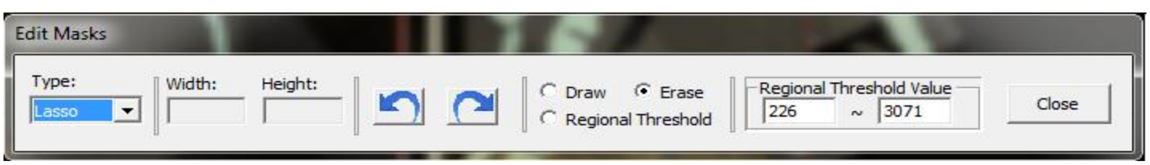

Figure 10. Edit panel.

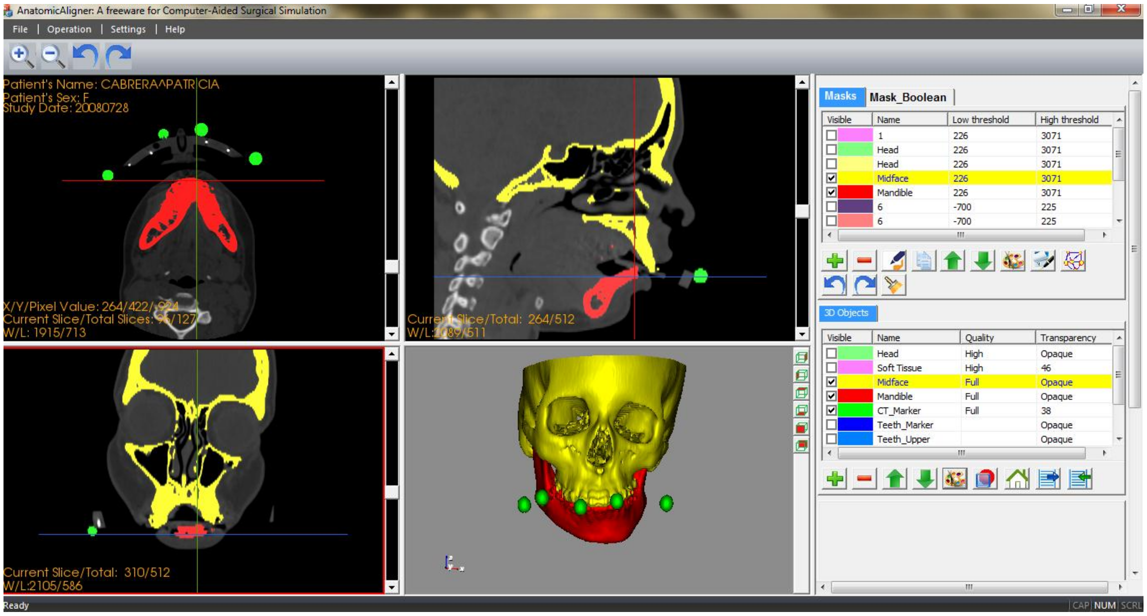

Figure 11. The main interface of module 1. The data set marked with yellow color is midface, red color is mandible, and green color is fiducial markers.

In order to improve the computational performance of this module, a multi-thread method is implemented to improve computational speed. A novel method is further implemented to save the physical memory space during the computation. In this method, two dynamic matrixes are used: one for storing original CT image set (called dynamic 3D image matrix), and the other for storing mask image sets (called 3D mask matrix). All the defined marks, i.e., midface, mandible and soft tissue, share a same 3D mask matrix. While the dimensions of the 3D mask matrix and the 3D image matrix are the same, the occupied memory spaces are different. The 3D image matrix uses 16 bits (the original CT image is 16-bit) and the 3D mask matrix uses 8 bits. Since the value of each element of 3D mask matrix is used to represent the index number of mask image set, 3D mask matrix can represent a total of 128 different kinds of mask images sets simultaneously, including all the masks of midface, mandible, soft tissue, etc. In addition, we use a specially-designed data structure to store the 3D mask image set. Each 3D mask image set is assigned to a data structure, including index number, display color, threshold value, the pointers of start and end addresses in the 3D mask matrix (Figure 12). There are two significant advantages using this approach: to significantly reduce the computational complexity and save occupied physical memory space. For example, when several masks are displayed at the same time, instead of repetitive computation, the system only needs to change the start and end pointers within the 3D mask matrix according to the index number in different data structure.

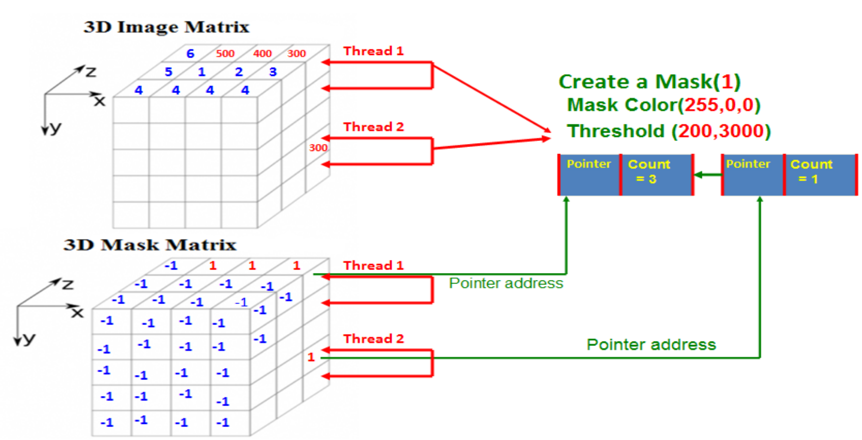

Figure 12. The data store structure.

In addition, user-friendly functions, including undo/redo, the color and transparency adjustment of masks and 3D models, are also implemented (Figures 13 and 14).

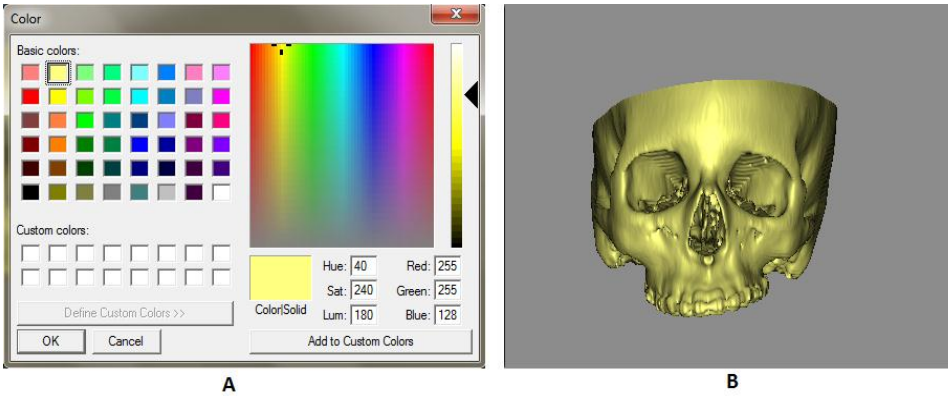

Figure 13. (A) Color control panel. (B) 3D model with a given color.

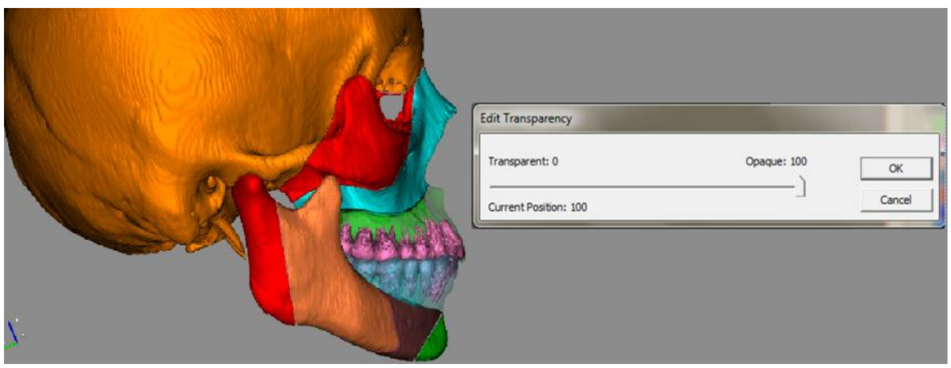

Figure 14. Transparency control panel.

Step 3: Import and register the digital dental models to the CT models. A set of digital dental models with bite-jig (fiducial markers) are imported into the system. Digital dental models are generated by a laser-scanner or high-resolution CBCT scanner. They include three individual models but with constant relationship as their physical counterparts: upper dental model, lower dental model and teeth marker model (Figure 17 A).

Since it requires a number of 3D models, from different imaging modalities, to plan a surgery, an important feature that we designed in this system is a customizable hierarchical structure for 3D models. Once 3D models are generated by Marching Cubes, and the digital dental models are imported, they are automatically classified to form a hierarchical list (Figure 15). Each node in the hierarchical tree represents a 3D model. The hierarchical tree is also customizable. Surgeon can simply use mouse to change the relations among the nodes in the hierarchical tree. When a 3D model is moved or rotated, all models under its hierarchical tree will also be moved or rotated accordingly. For instance, the "teeth marker" model is defined as parental node of upper and lower dental models in the hierarchical tree. Both the upper and lower dental models can be rotated it simultaneously. Surgeon can choose to move upper dental model only as well.

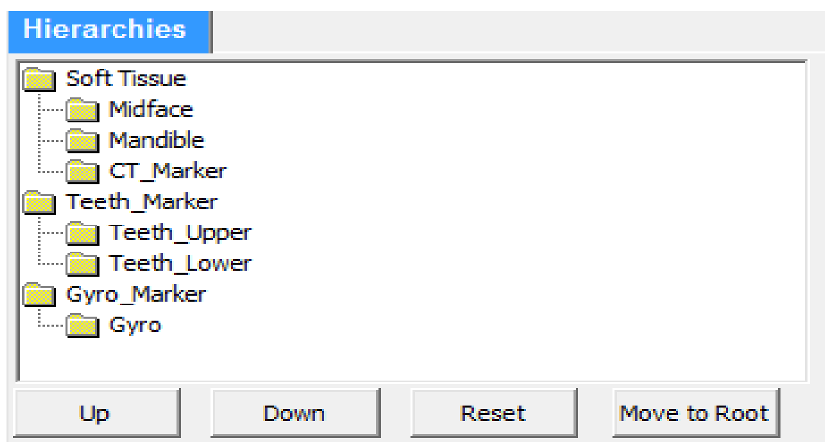

Figure 15. Hierarchical control panel.

During the incorporation process of digital dental models to the CT models, the upper and lower teeth models are assigned to be the children of the "Teeth Marker" model in the hierarchical tree. Therefore, when the "Teeth Marker" model is moved, rotated and registered to the "CT Marker" model, it also brings both upper and lower digital dental models along with it (Figure 17). During the registration, both manual and automatic registration methods are implemented.

Manual registration is achieved by a translational/rotational control panel (Figure16) to manually move and rotate the teeth marker model. Each 3D model has an origin, the center of the bounding box of the model. The model is rotated around the origin. However, during the registration or surgical planning, it is not feasible for a precise rotation when a pivot point is at the center of the model. We therefore implement a function to freely move the pivot to a new position, i.e., the tip of the central incisor or a specific anatomic landmark.

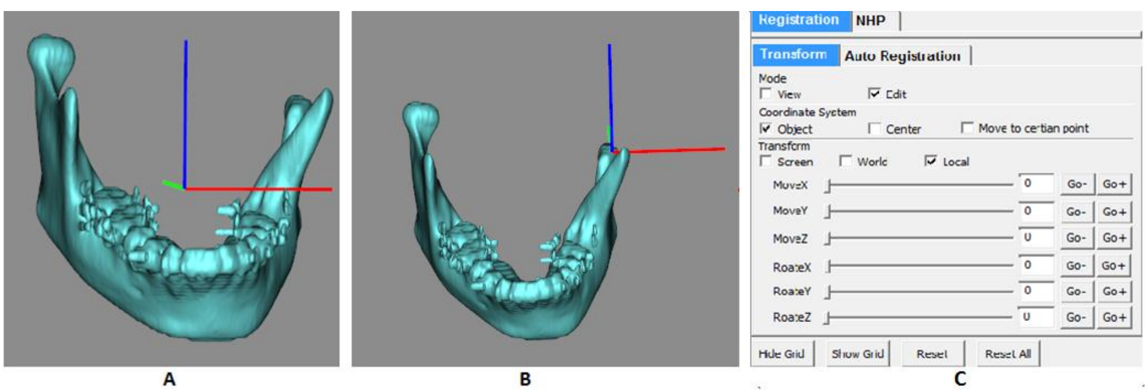

Figure 16. (A) Initial origin of the mandibular model. (B) The pivot is moved the center of right condylar head. (C) Transformation panel.

Auto-registration is based on iterative closest point (ICP) algorithm [23] that registers the teeth marker model of the digital dental models to the CT marker model automatically (Figure 17).

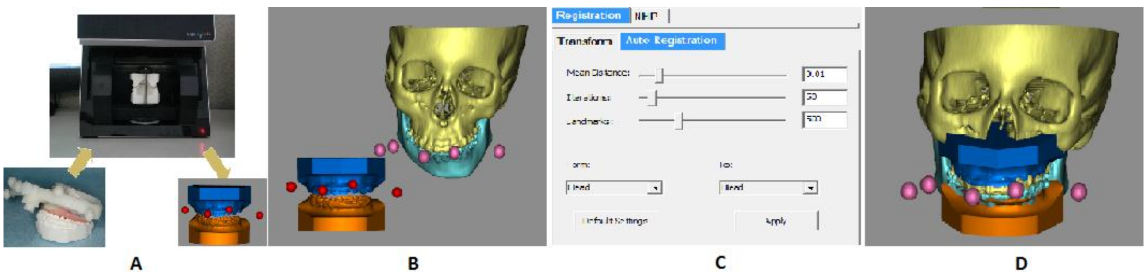

Figure 17. (A) A laser-scanner is used to create digital dental models with bite-jig (red: “Teeth Marker”; dark blue: upper teeth model; dark yellow: lower teeth model). (B) Before the digital dental models are registered to the 3D CT models (for CT models: yellow: midface; blue: mandible; pink: CT markers. (C) Auto-registration panel. (D) After the digital dental models are incorporated into the 3D CT model by registering the “Teeth Marker” model to the “CT Marker” model.

<u>Step 4: Merge digital dental models to the 3D CT models.</u> Multi-connect hexahedron method will be used to cut the “bad teeth” off the CT models, and bases off the digital dental models, thus replacing the upper and lower “bad teeth” in 3D CT model with the “good teeth” in digital dental model respectively. The method on how to use multi-connect hexahedron to cut a 3D model in to pieces will be described in details in Section 3.4. The new formed composite skull model has an accurate rendition of both bones and teeth. Figure 18 is the result after the digital dental models are incorporated into the 3D CT model.

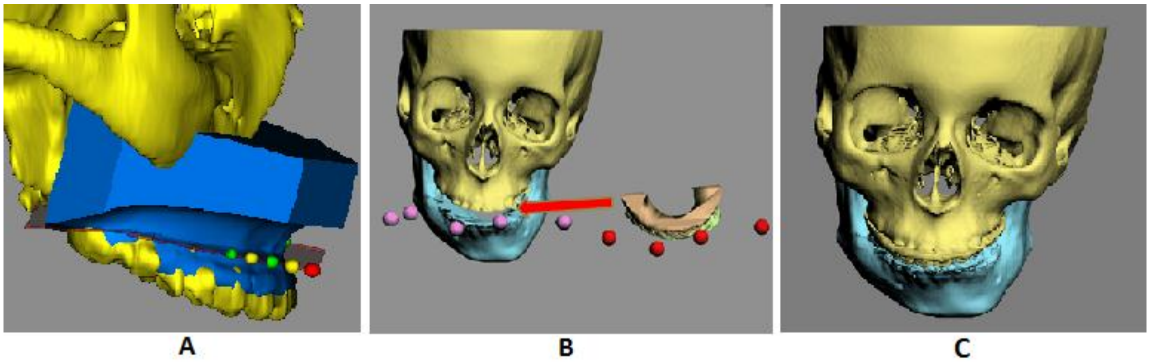

Figure 18. Merge digital dental models to the 3D CT models. (A) Using multi-connect hexahedron to cut teeth. (B) Result after cutting. (C) Result after replacement.

3.2 Module 2: Reorientation of Composite Skull Model to NHP

The step after creation of composite skull model is to re-orientate the composite model to recorded NHP. As described above, NHP is a unique head orientation that a human normally holds his/her head. Since human head can be rotated and tilted, we use NHP as the frame of reference to position the 3D models into the 3D world coordinate system in our CASS system. A new clinical protocol has been developed to achieve this purpose [10]. In this technique, a digital gyroscope is attached to the same facebow and bite-jig that is used to create the composite skull model. When the patient's head is positioned in NHP (looking infinite horizon; the head is neither rotated nor tilted), the pitch, roll, and yaw of the head are recorded. The recorded pitch, roll, and yaw are then used to reorient the composite skull model. In the computer, a computer-aided designing (CAD) model of the gyroscope is registered to the composite skull model by registering the "Gyroscope Marker" model to the "CT Marker". Finally, the recorded pitch, roll, and yaw are applied to the gyroscope CAD model, reorienting the composite skull model to the NHP (Figure 19).

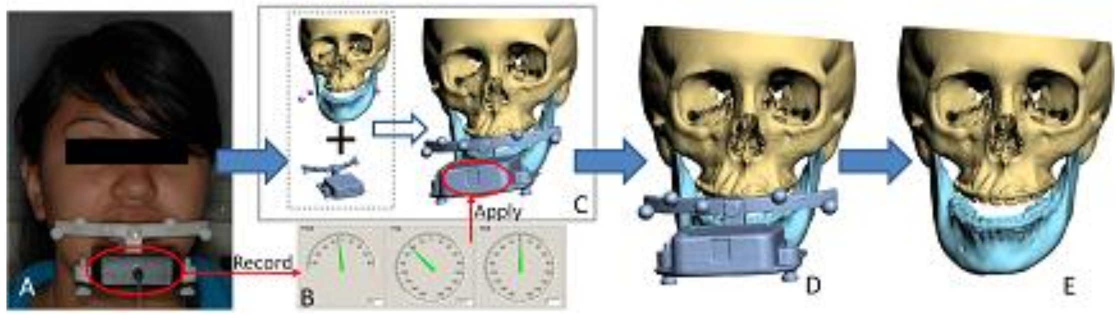

Figure 19. (A) Digital gyroscope is attached to bite-jig and facebow; (B) Pitch, roll, and yaw of gyroscope is recorded; (C) In the computer, digital replica (CAD model) of gyroscope is registered to composite skull model using the fiducial markers; (D) Recorded pitch, roll, and yaw are applied to gyroscope CAD model, reorienting composite skull model to NHP. (E) After composite skull is oriented to NHP, gyroscope CAD model and fiducial markers are marked hidden.

In our CASS system, the following two steps are implemented to orient the composite skull model to the recorded NHP:

<u>Step 1: To register the gyroscope to the composite skull model.</u> The predetermined gyroscope CAD model assembly includes a gyroscope CAD model and its associated fiducial marker model (called “Gyro Model”, shown in the left lower figure in Figure 19 C). The original rotation of the assembly is (0, 0, 0) in the world coordinate system. The gyroscope CAD model is registered to the composite skull models by registering the “Gyro Marker” model to the “CT Marker” model using either ICP algorithm or manual transformation (Figure 20). The algorithm of this function is

the same as ones described in section 3.1. During the registration, the “Gyro Marker” is the parental model to the gyroscope CAD model.

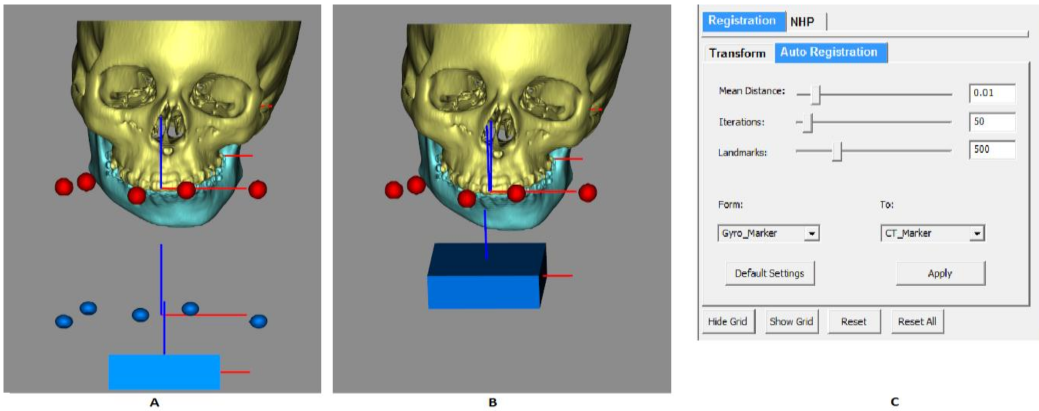

Figure 20. (A) Digital composite skull model and Gyroscope CAD model assembly (including the gyroscope and fiducial marker models). (B) The gyroscope CAD model assembly is registered to the CT model. (C) Auto registration panel.

<u>Step 2: To reorient the composite skull model to the recorded NHP.</u> In this step, the digital gyroscope CAD model becomes the root for all the 3D models in the hierarchical tree. The recorded the NHP parameter (pitch, roll and yaw) is applied to the gyroscope CAD model, thus reorienting the entire composite skull model to NHP, the exact head orientation as recorded in the clinic. A NHP transform panel is designed for this process (Figure 21). Finally, a midsagittal plane is defined for composite model. The midsagittal plane is the Y-O-Z plane in the world coordinate system. The midsagittal plane is defined in conjunction with the result of clinical examination.

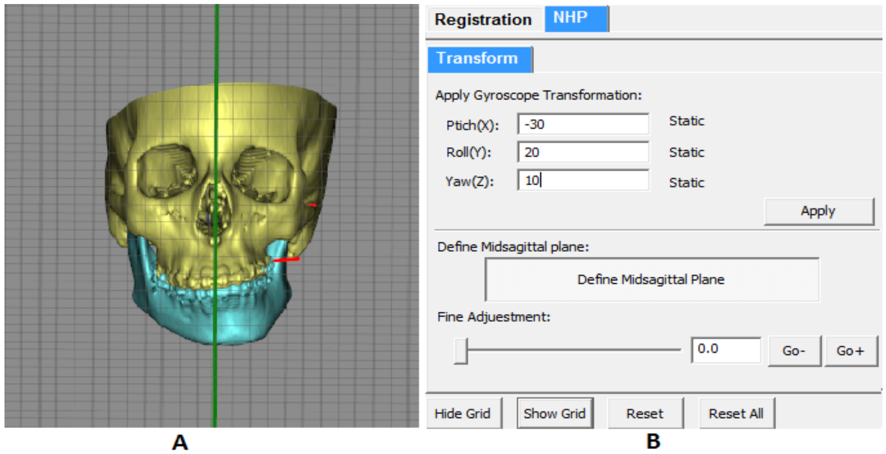

Figure 21. (A) Digital model after NHP transform displayed with grids and midsagittal plane. (B) NHP transform panel. Green line in (A) is midsagittal plane.

3.3 Module 3: Cephalometric Analysis

After the 3D composite skull model is oriented to NHP, 3D cephalometric analysis is performed. In orthodontics and CMF surgery, we use cephalometric analysis to quantify the deformity and plan the treatment. Cephalometric analysis is a set of linear, angular and orientational measurements of the composite skull model using a group of manually digitized anatomical landmarks.

In order to complete this process, it is necessary to create two types of relationships for each landmark. The first relationship is a forward mapping from the cross-sectional images to the 3D models. The second relationship is a reversed mapping from the 3D models to the cross-sectional images. For example, if a landmark is digitized on any of the cross-sectional image, either axial, sagittal or coronal, the landmark will be automatically projected onto the 3D model, as well as the two other cross-sectional images. The opposite is the true. If a landmark is landmark is digitized on a 3D model, either generated by *Marching Cubes*, or imported from a .stl file, the landmark will be automatically projected onto the three cross-sectional images (Figure 22).

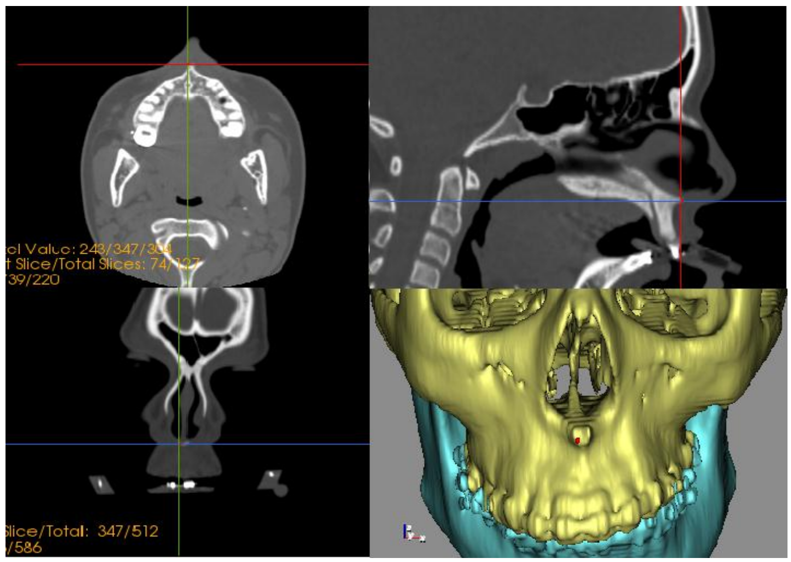

Figure 22. The landmark (red sphere) is the digitized landmark.

In order to efficiently perform cephalometry, only the landmarks that are used in the analysis are digitized. We implemented cephalometric analysis in the following five steps:

Step 1: To establish an initial landmark library. The initial landmark library includes all the routine anatomical landmarks that are commonly used in cephalometry. The library is customizable that a user can freely add, delete or edit a landmark. Cephalometric landmarks are defined by an input frame (Figure 23). The name and the description of these landmarks will be restored in a database file automatically.

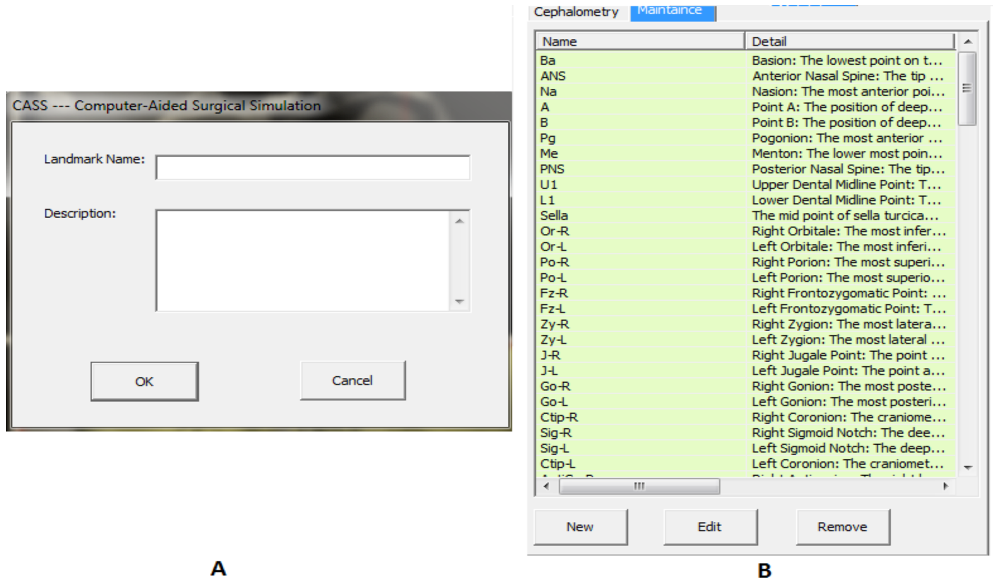

Figure 23. (A) Input window. (B) Landmark library.

<u>Step 2: To define cephalometric analysis.</u> Each cephalometric analysis is a group of linear, angular and orientational measurements. Depending on doctor's training and habit, different doctor, in different hospitals or regions, may use different group of the measurements. This system allows unlimited number of cephalometric analyses. The information of each cephalometric analysis, i.e., name of the cephalometric analysis, categories, measurement items, landmarks used and descriptions of analysis, is stored in a database file.

<u>Step 3: To define each measurement and its equation in the cephalometric analysis.</u> The type of the measurements includes angle, distance, orientation, ratio and other computation using the landmarks (Figure 24). The measurement equation can be either defined based on 3D coordinates, or on projected (on sagittal, axial or coronal plan)

2D coordinates. The measurements based on the projected landmarks are sometimes very useful during the decision making process.

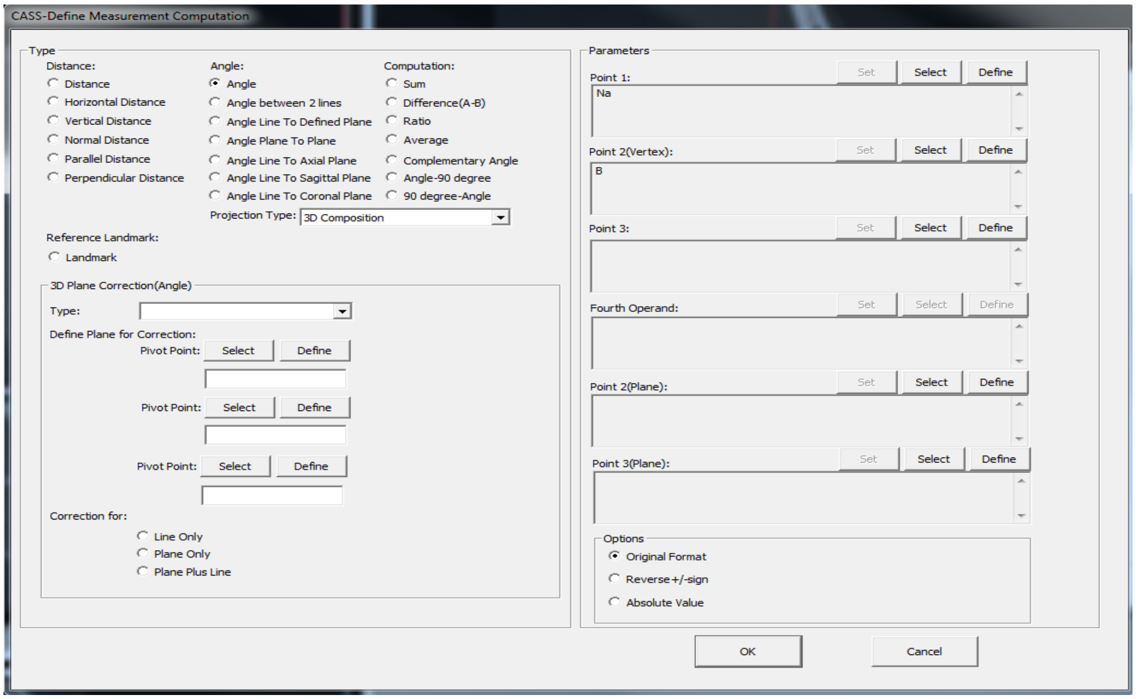

Figure 24. Defining measurement computation panel.

<u>Step 4: To record the initial positional coordinates of each landmark.</u> When a landmark is digitized, it will be "glued" to its corresponding 3D model and moved/rotated along with that model. This feature is especially important for the surgical planning, where an osteotomized bony segment is moved and orientated. The digitized landmark must also be rotated and translated along with the bony segment to a desired the position.

Step 5: To calculate and report the results. Angle, distance, and other computation will be performed and a measurement report will be created. A cephalometric analysis report includes five different categories of the measurements: orientation, symmetry, position, size and shape for the clinical purpose.

Orientation analysis will be measured as pitch, roll and yaw for each unit (i.e., maxilla, mandible and chin), by comparing the orientations of local and world coordinate systems. Because a 3D composite angle does not have any clinical meaning, we will have to separately measure the orientation of each facial unit in a special order: first yaw, then roll, and finally pitch. The yaw and roll will be used as a part of symmetry analysis (as described in the next paragraph). Pitch will be used for the size, shape and positional measurements. Therefore, it will be measured at last after the yaw and roll are removed.

Symmetry analysis will include both intrinsic symmetry and symmetric alignment measurements. For measuring the intrinsic symmetry, we will use a triangular technique. The degree of intrinsic asymmetry will be quantified by calculating the difference between the 2 base angles of the triangle. For the symmetric alignment, each facial unit will be measured with respect to the midsagittal plane, a plane that we used clinically to divide the head into right and left halves. The degree of symmetric alignment of a facial unit will be quantified by first measuring the transverse (right-left) deviation to the midsagittal plane. Then, the degree of yaw and roll of the facial unit will be measured using above 3D orientation measurement.

<u>Position</u> will be measured differently depending upon the direction of the measurements. Transverse position will be measured as a linear distance between the involved landmark and the midsagittal plane of *world* coordinate system, while anteroposterior and vertical positions will be measured after all the involved landmarks are projected onto the *world* midsagittal plane.

<u>Size</u> will be measured as a linear distance in 3D space. Finally, **<u>shape</u>** will be measured by projecting the involved landmarks onto the midsagittal plane of *local* coordinate system for each facial unit.

3.4 Module 4: Virtual Osteotomy

Once the composite skull model is created and reoriented to the NHP, virtual osteotomies (i.e., Le Fort I, sagittal split, or genioplasty) can be simulated in the CASS system. During simulation of a virtual osteotomy, a user first creates a virtual knife defined by a series of multi-connected hexahedrons, followed by cutting and separating the single 3D bone model into 2 segments (Figure25). Once the simulation is completed, the user can create a surgical plan by moving and rotating bony segments to their desired positions. He may also simulate various osteotomies to determine which will be the best to the patient. The process of virtual osteotomy includes two stages. The first stage is to form a cutting plane, while the second stage is to cut and separate a 3D model into two segments.

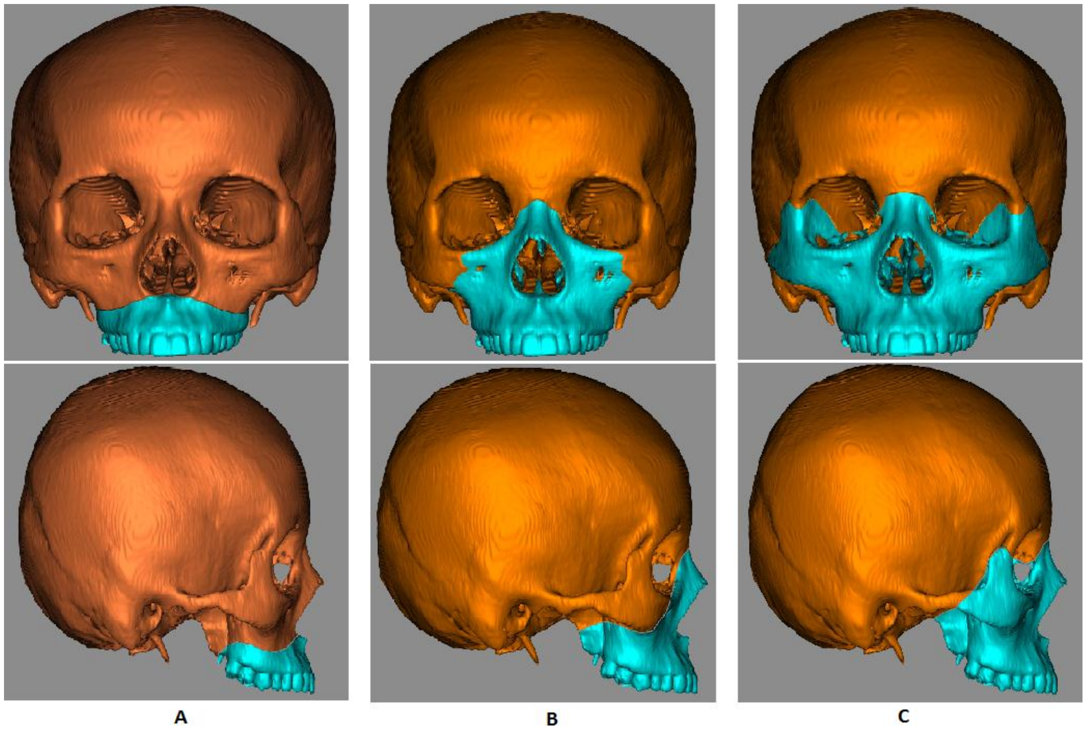

Figure 25. Three types of osteotomies. (A) Le Fort I osteotomy. (B) Le Fort II osteotomy. (C) Le Fort III osteotomy.

3.4.1 Stage 1: To form a "curved" cutting plane - the virtual knife

In the 1st stage, a series of multi-connected hexahedrons is used as a "curved" cutting plane for virtual osteotomy. The cutting plane is created and adjusted in the following two operations: 1) to form a cutting plane; and 2) to adjust the cutting plane. They are described below in details.

3.4.1.1 Operation 1: To form a "curved" cutting plane

As mentioned above, the cutting plane consists of a series of multi-connected hexahedrons. A surgeon needs to first manually define and digitize a serial of landmarks, indicating where the virtual osteotomy should take place. Each landmark represents one vertex of the cutting line (Figure 29 (A)). The multi-connected hexahedrons are created by first forming individual hexahedron using the two adjacent digitized landmarks, then connecting them together. The process is completed in the following two steps:

Step 1: Creating individual hexahedrons. Each hexahedron is created by six mutually perpendicular faces (top, bottom, front, back, and two sides), which are defined by three mutually perpendicular vectors. The top and bottom faces are defined by the height and length vectors; the front and back faces are defined by the length and depth vectors, while the two side faces are defined by the depth and height vectors. The depth vector is perpendicular to the screen. After all the landmarks are digitized, the screen depth of each landmark is adjusted to the landmark that has the shortest screen depth in screen coordinate system (the point closest to the operator) (Figure 26). Once all the landmarks have the same shortest screen depth, all the initial hexahedrons depth is assigned to 70 mm. Next, the length vector connects the two adjacent landmarks. Finally, the height vector is perpendicular to the length and depth. It represents the thickness of the virtual knife with an initial thickness of 0.5 mm (Figure 27). During landmark digitization and hexahedron creation, the camera is fixed so the viewing angle of the 3D model is static.

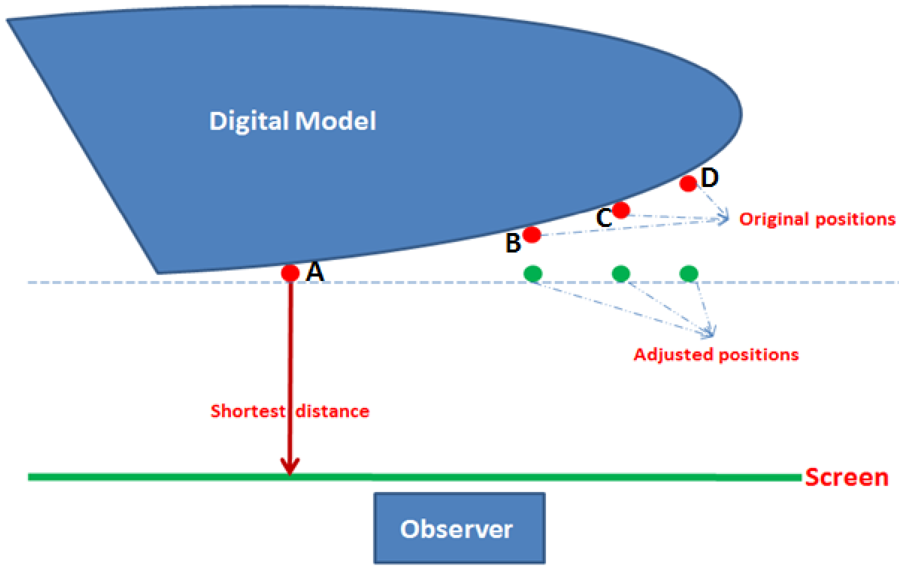

Figure 26. Landmark A has the shortest distance to screen. The following landmarks B, C, and D are further away from the screen respectively. Landmark B, C, and D are adjusted to a position that has equivalent distance with the landmark A to the screen. The green points represent the adjusted positions.

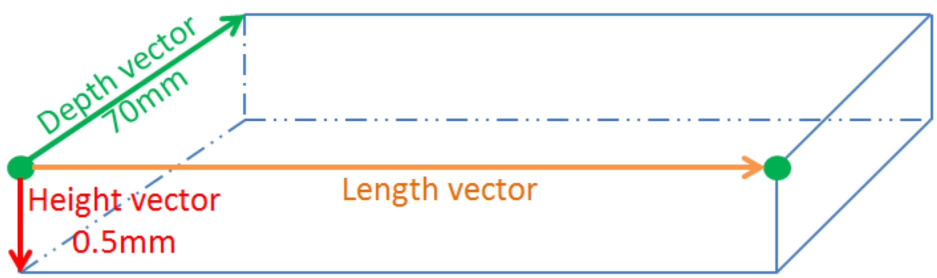

Figure 27. Individual hexahedron. The green spheres are planned landmarks.

Step 2: Connecting multiple hexahedrons. If two adjacent faces of the hexahedrons are parallel, they can be connected directly. However, if they are not parallel to each other, the top or bottom faces of the two adjacent hexahedrons will be extended accordingly depending on the direction of the angle (Figure 28). Therefore, after two adjacent rectangular hexahedrons are connected, they may be changed to non-rectangular hexahedrons which the lengths of the top and bottom faces are unequal. Since a series of hexahedrons are connected to construct to simulate a "curved" cutting plane (Figure 29 (B)), the depth and thickness of all the hexahedrons are the same and have to be adjusted simultaneously, while the orientation of each hexahedron can be adjusted individually along the hinge in the next operation (Operation 2). These connected hexahedrons are used to simulate a virtual surgical saw for the osteotomy.

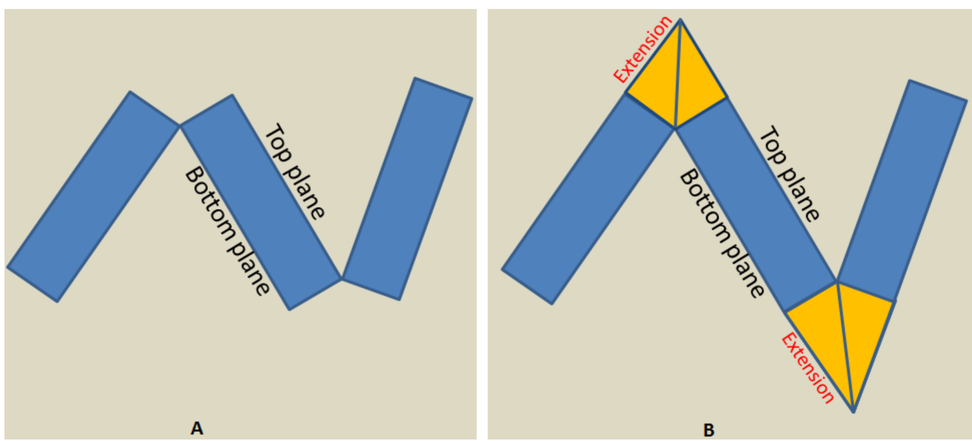

Figure 28. Connecting hexahedrons. (A) Original hexahedrons. (B) Extended hexahedrons.

3.4.1.2 Operation 2: Adjusting the cutting plane

Six control spheres are served to adjust the connected hexahedrons. Four green ones are used to control the size of each hexahedron while the two yellow spheres are used to control the angle between two adjacent hexahedrons (hinge rotation only). When a specific sphere is selected, it turns in red, indicating the current selected operation

(Figure29 (B)). Finally, a control panel is designed to adjust the translation, rotation, and thickness of entire cutting plane – the multi-connected hexahedrons (Figure 29 (C)).

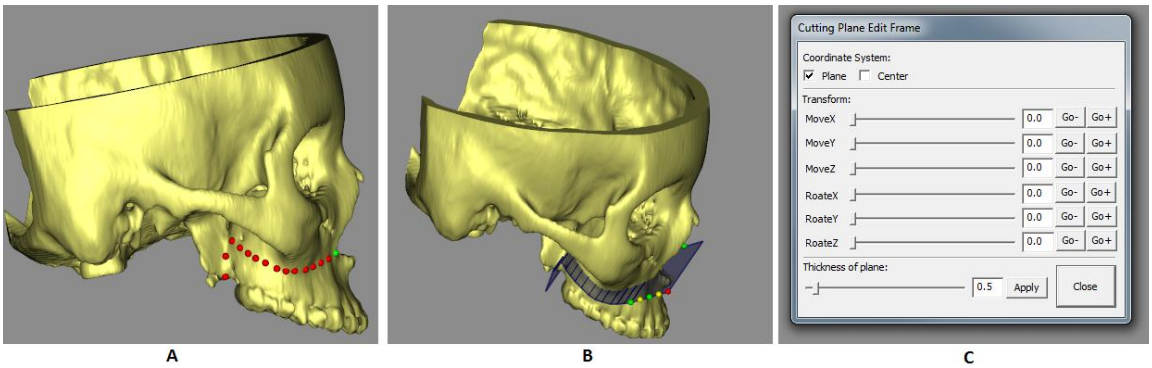

Figure 29. (A) The red spheres are determined landmarks. The green sphere is active landmark. (B) The green spheres and yellow spheres are used to control the size of each hexahedron and the angle between two adjacent hexahedrons respectively. The red sphere represents the current selected operation. (C) Control panel.

3.4.2 Stage 2: To cut and separate a 3D model into two segments

In the 2nd stage, multi-connected hexahedrons (virtual knife) are used to cut (osteotomize) a target 3D model into two models (medically we call them “segments”). In addition, it is important to form a closed surface along the cutting plane in the osteotomized segments by the re-triangulating broken triangles. This process includes four sequential operations: 1) To classify triangles; 2) To recreate new triangles for the “broken” intersection triangles; 3) To recreate a closed surface; and 4) To create the final two 3D segments (models).

<u>3.4.2.1 Operation 1: To classify triangles</u>

The triangles of the target 3D model can be classified into four sets based on where they are located in relation to the hexahedrons: completely outside (outside set), intersection at the upper face (upper intersection set), intersection at the lower faces (lower intersection set) and completely inside (inside set) as shown in Figure 30. There are two implicit rules in this operation:

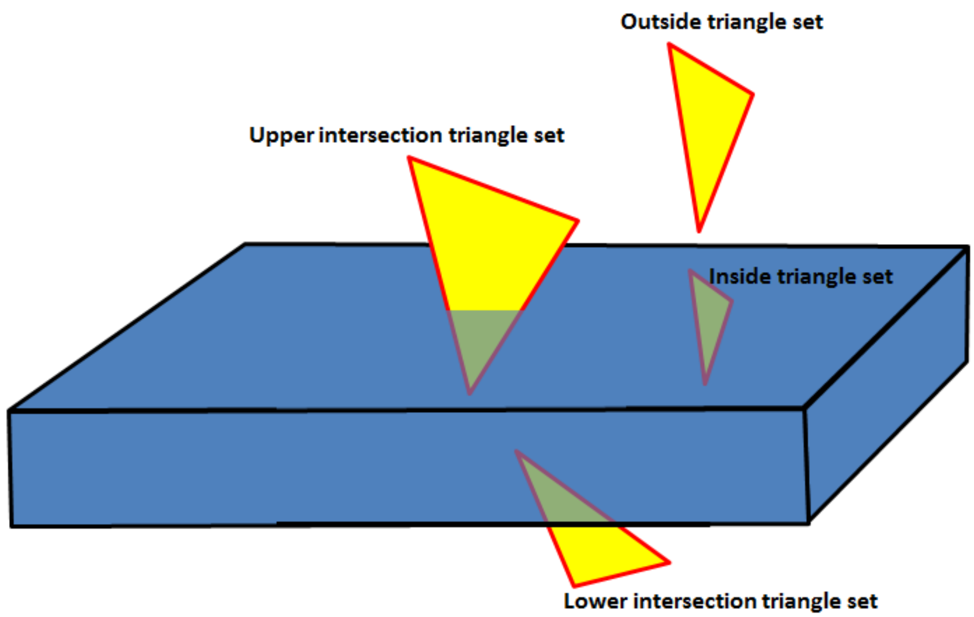

Figure 30. Relationship between triangles and a hexahedron.

(1) As defined above, each hexahedron is created by six faces (also called planes in the following equations) with eight vertices (Figure 31). Each plane has its own equation. For example, the bottom plane equation is created by ooo, xoo, and xoz; the top plane equation is created by oyo, xyo, and xyz, etc. These six equations are then grouped into three grouped equation sets: top-bottom, front-back, and left-right.

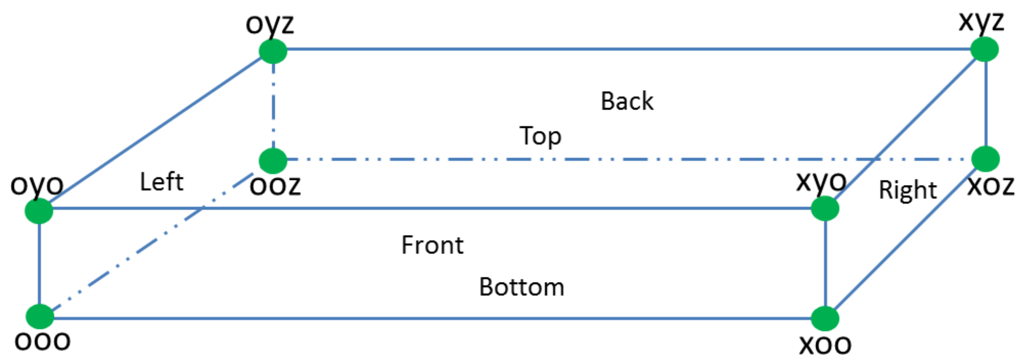

Figure 31. Eight vertices of hexahedron.

(2) The next step is to determine the relationship between a hexahedron and a triangle. Each triangle contains three sides and three vertices. Each vertex is plugged into three grouped equation sets (top-bottom, front-back, and left-right) of the hexahedron, respectively. The solution of each equation can be either positive, negative, or zero. If both two solutions of any grouped equation set are either positive or negative, this vertex is outside of the hexahedron. If one of two solutions is equal to zero, this vertex is laying on the hexahedron. If one solution is positive and the other is negative, this vertex is inside of the hexahedron (Figure 32). Therefore, if one vertex of a side is outside of the hexahedron while the other is inside, then this side has an intersection point with the hexahedron.

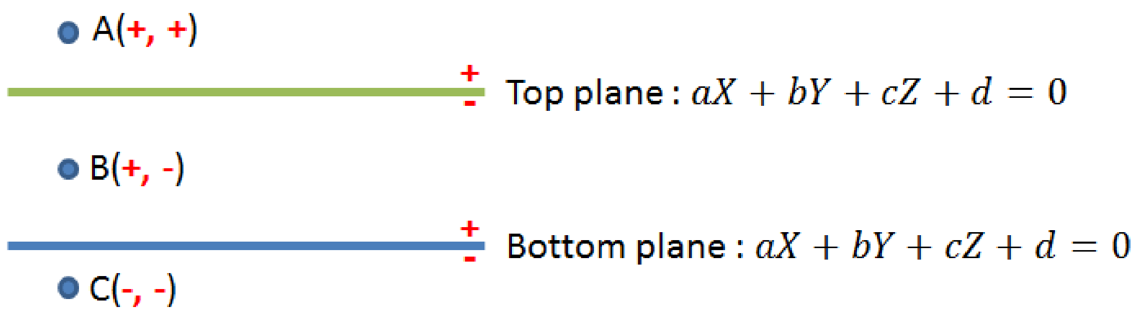

Figure 32. The green line represents top plane. The blue line represents bottom plane. "+" indicates the vertex is located above the plane. "-" indicates the vertex is located below the plane. Both "+" are assigned to Vertex A because it is above both top and bottom plane; by the same token, both "-" are assigned to Vertex C because it is below both top and bottom plane; a "+ and a "-" are assigned to Vertex B because it is below the top plane and above the bottom plane.

The above two rules are used to classify all the triangles. If all vertices of a triangle are completely outside of any hexahedron, this triangle is classified into the outside triangle set. Then if one vertex is inside of a hexahedron while the other two vertices are above the top plane of the hexahedron, this triangle is classified into the upper intersection set. And if one vertex of a triangle is inside of a hexahedron while either of the other two is below the bottom plane of the hexahedron, this triangle is classified into the lower intersection set. Finally, if all vertices of a triangle are completely inside of any hexahedron, it is classified into the inside triangle set (Figure 30). We only use the first three types of triangle sets (outside set, upper intersection set, and lower intersection set) because the inside triangle set will be cut away because they are located within the cutting plane. Figure 33 shows the result after classification.

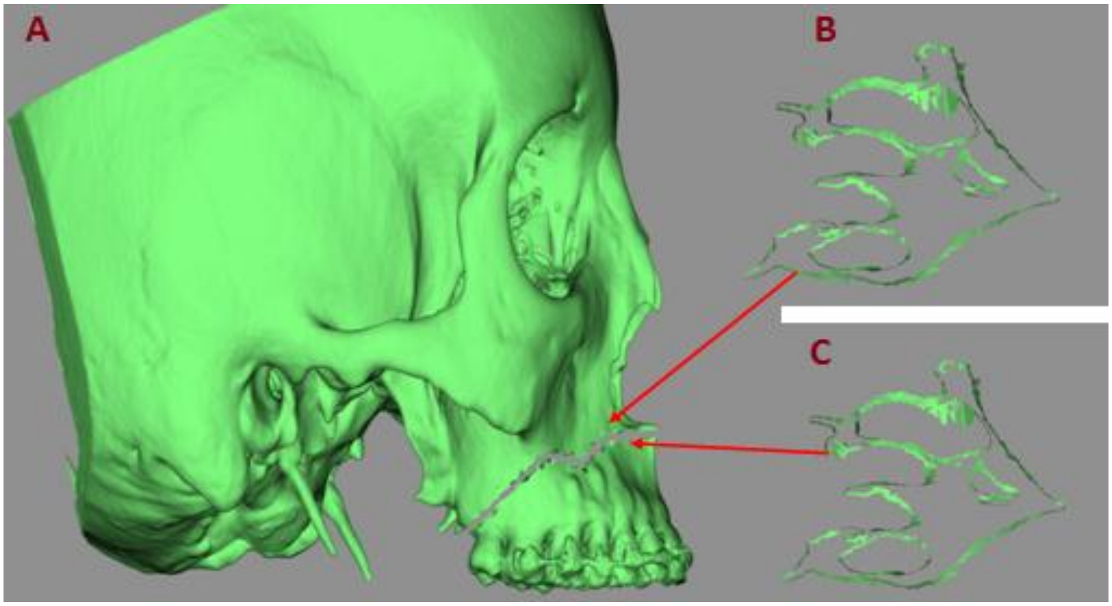

Figure 33. (A) Outside triangle set. (B) Upper intersection set. (C) Lower intersection triangle set.

3.4.2.2 Operation 2: To recreate new triangles for the "broken" triangles

After a triangle is classified into the upper or lower intersection set, the cutting plane will cut through the two sides of the triangle, resulting in a "broken" triangle with two intersection points, one on each side. Therefore, it is necessary to recreate two new triangle sets for the "broken" triangles along the intersection plane (Figure 34), the upper and lower triangle sets.

In the either intersection triangle set, when a triangle is intersected with a hexahedron, either one vertex or two vertices will be outside of the hexahedron. The "broken" triangles will be "fixed" using following algorithm. If only one vertex is outside of the hexahedron, it will be used to build a new triangle with the two intersection points on the two sides (Figure 35A). If two vertices of the triangle are outside of the hexahedron, they will be used to build two new triangles with the two intersection points on the two sides (Figure 35B). This algorithm is used to create two new triangle sets for the upper and lower intersection triangle sets. Finally, the new side constructed by the two intersection points will be used to create closed surfaces in the next operation (Operation 3).

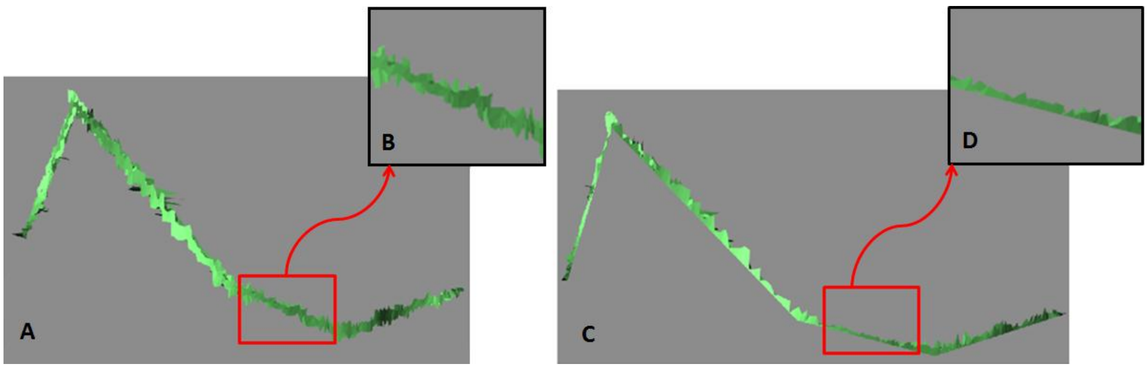

Figure 34. (A) Upper intersection triangle set before cut. (C) Upper intersection triangle set after cut. (B) (D) Close-up view of the region in the red box.

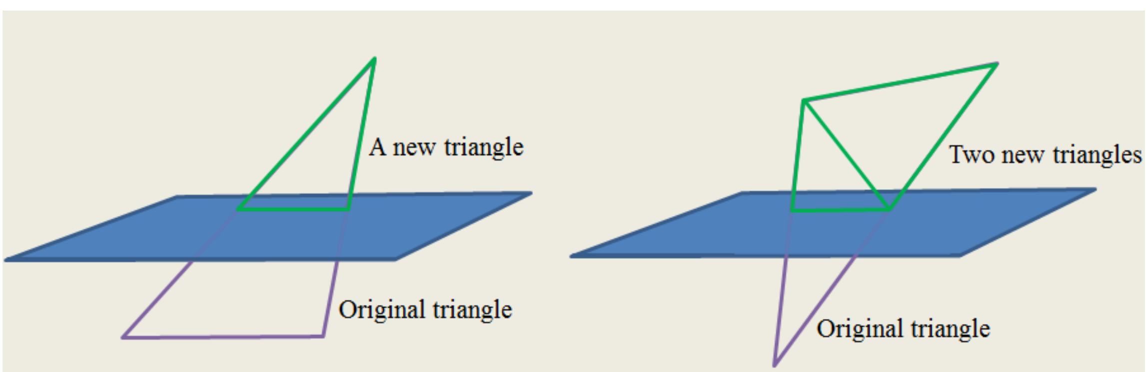

Figure 35. (A) Recreating a new triangle from a "broken" original triangle. (B) Recreating two new triangles from a "broken" original triangle.

3.4.2.3 Operation 3: To create a closed surface using the line segments on the cutting plane

The purpose of this operation is to create a closed surface along the cutting (intersection) plane for each osteotomized segment. In Operation 2, all the "broken" triangles are "healed" by connecting the two intersection points. The new side is called line segment in the following computation. The closed new surface is reconstructed by triangulating the line segments. The advantage of using line segments instead of intersection points is that the line segments can maintain the structural boundaries on new cutting surface of the osteotomized model. If the new surface is re-triangulated directly from intersection points, the structural boundaries will be altered. Another advantage is that the line segments can be used to maintain the anatomical structures on the new surface, i.e., maxillary sinus (a big hole) should be left open on the new surface.

As we described above, the "curved" cutting plane is constructed by a set of multi-connected hexahedrons. Each connected hexahedron is angulated to each other. Therefore, there are two major problems in forming a closed surface. The first is that the line segments will not be on the same plane, i.e., they may belong to different hexahedrons. The second problem is that the line segments may be partially or completely overlapped to each other (Figure 37). In order to solve the above problems, we developed the following novel method. The method includes the following six steps:

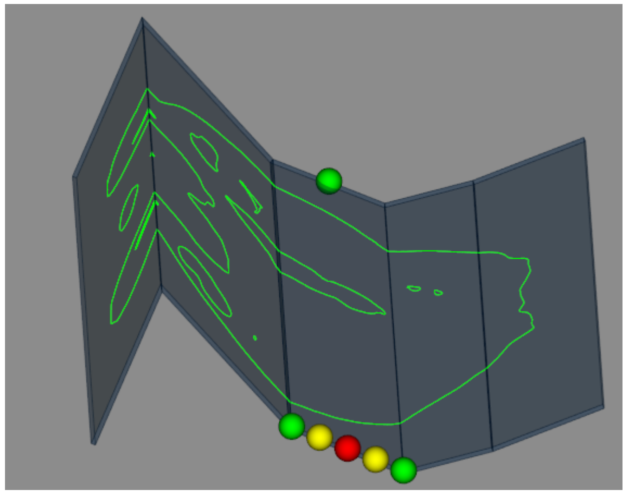

Figure 36. Green lines are original intersection lines.

Step 1: Re-indexing all the line segments in descending order by the length.

Step 2: Eliminating overlaps along the adjacent line segments. Three functions are developed to achieve this goal. **The first function** is to find shortest distance between two line segments. **The second function** is to determine whether two line segments are parallel to each other. **The last function** is to eliminate the overlaps along the adjacent line segments. For instance, two line segments (L_1, L_2) will be merged together if both following conditions are met: the distance between L_1 and L_2 is less than a given threshold, and they are parallel to each other. The more detail of the algorithm of the last function is described in Appendix A in pseudo code.

Step 3: Connecting intersection line segments together. After reconnection, closed line segment chains will be created based on the distance between two line segments. For instance, a line segment (L) will be connected to the last line segment (G) in a given line segment chain if the distance between L and G is less than a given threshold. In addition, if an opened line segment chain is found during the re-connection process, which indicates the cutting plane is not thoroughly cutting through the 3D model, an error message will pop up and inform user to redefine cutting plane. The pseudo code of the algorithm in the step is provided in Appendix B.

Step 4: Recombining closed line segment chains based on different planes. **Six algorithms** are developed to accomplish the purpose of the step. After combination, line segments of each closed line segment chain belong to one plane (Figure 37).

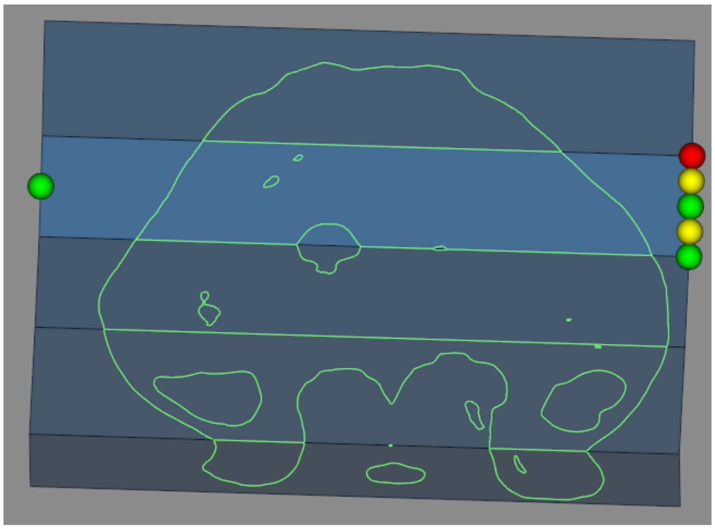

Figure 37. After recombination, each plane do not share any new closed lines set with others.

The first algorithm is to reconnect line segments in the closed line segment chains on a given plane (Figure 38). In this operation, the line segment chains are classified into two chain sets: closed or opened line segment chain set on a given plane. If the distance between the first line segment to the last one in the chain is zero, the line segment chain is classified into the closed line segment chain set. Otherwise, the chain is classified into the opened line segment chain set. The goal of the following algorithms is to create the closed line segment chain set from the open ones.

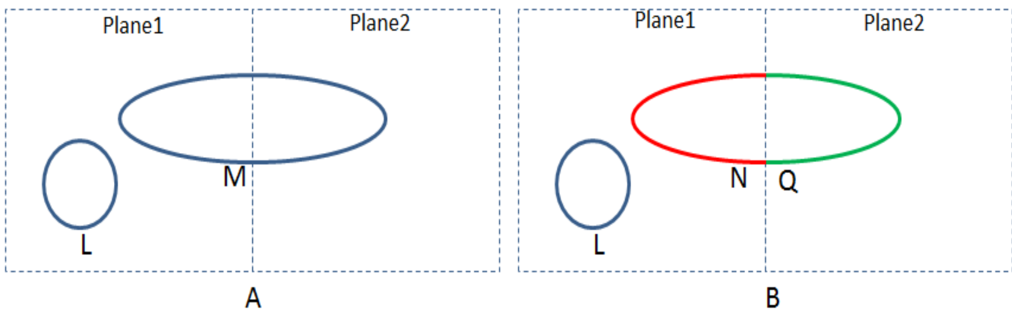

Figure 38. (A) Before the re-connection. (B) After the re-connection. After the re-connection, on Plane 1, the line segment chain (L) is classified into closed line segment chain set. The opened line segment chain set only includes N. On Plane 2, there is no closed chain set. The opened line segment chain set only includes Q.

The second algorithm is to classify the line segment chains in the opened line segment chain set into left and right set. This is done based on the distances from the first line segment, and last one, to the left side of the plane. Assuming D_1 and D_2 are distances from the head (first line segment) and tail (last line segment) of the opened line segment chain to the left side of the plane respectively. If D_1 is equal to D_2 and D_1, D_2 are larger than a specific threshold, the opened line segment chain is classified into right set. Otherwise, the opened line segment chain is classified into left set (Figure 39).

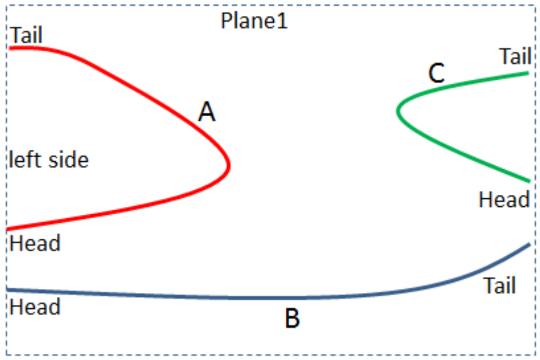

Figure 39. Line segment chains (A, B) are classified into left set. Line segment chain C is classified into right set.

The third algorithm is to re-connect the line segments of each line segment chain in the left and right set based on the distances from the first line segment, as well as the last one, to the front and left sides of the plane. The distances are sorted in ascending order. If the head (first line segment) and tail (last line segment) in the chain are located in the same left or right side of the plane, the head line segment will be re-ordered to make sure it has the shortest distance to the front side of the plane. If the head and tail line segments are not located in the same left or right side of the plane, the head line segment will be re-ordered to make sure it has the shortest distance to the left side of the plane (Figure 40).

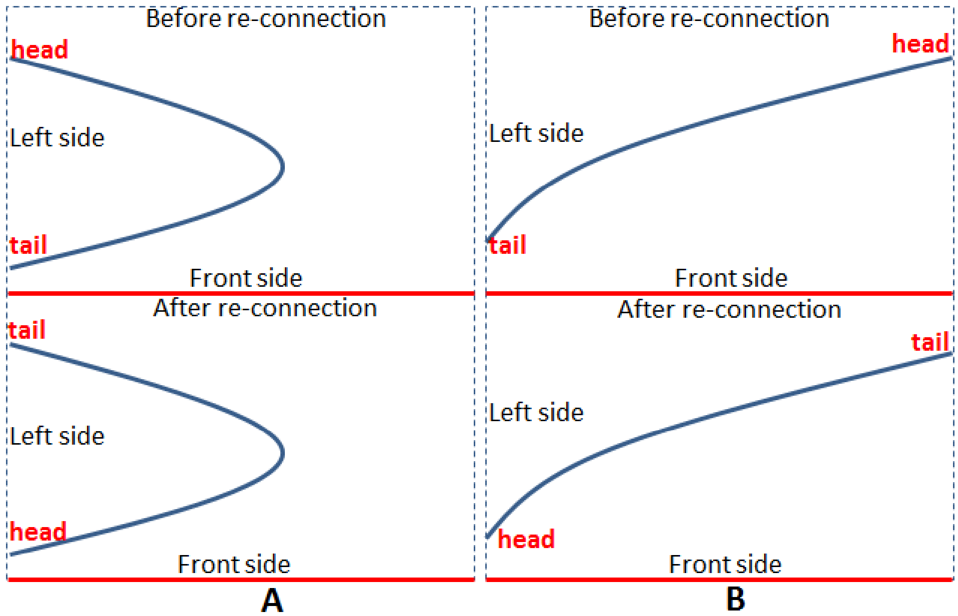

Figure 40. Re-connecting the line segments of each line segment chain in the left set. (A), (B) are two different connection methods

The fourth algorithm is to re-order the line segment chains in the left and right set in ascending order by the distance from the head (the first line segment) of each chain to the front side of the plane (Figure 41). After re- order, the first line segment chain of left or right set has the shortest distance to the front side of the plane.

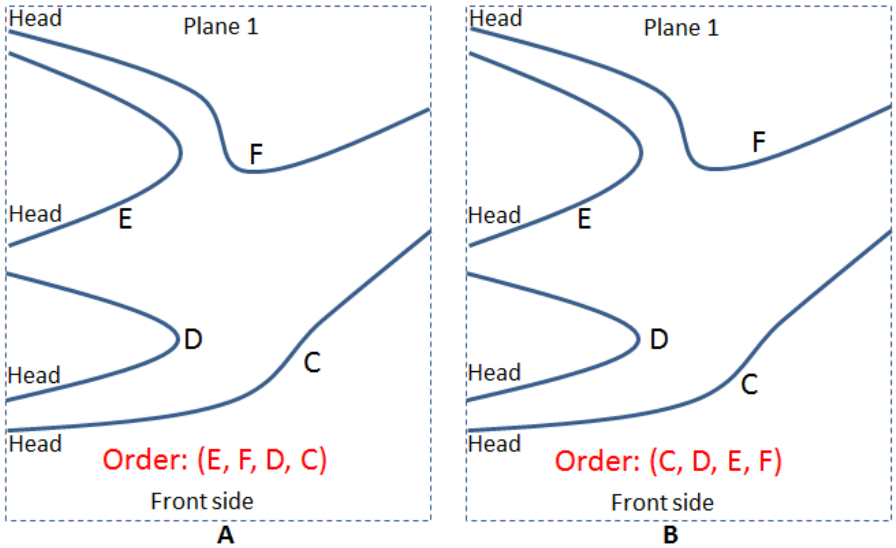

Figure 41. Re-ordering the line segment chains is shown in the left set. (A) Before re-ordering. (B) After re-ordering. The head of chain C has the shortest distance to the front side of plane 1. The following heads of B, C, and D are further away from the front side of plane 1 respectively. After re-ordering, the order is rearranged as it is shown in (B).

The fifth algorithm is to establish the relationships among the line segment chains in the left and right sets, based on the distances from the first and last line segments of the chain to the front side of the plane (Figure 42). For instance, assuming H_1, T_1 are the distances from the first line segment, and last one as well, of the chain (C_1) to the front side of the plane. H_2, T_2 are the distances from the first and last line segment of the chain (C_2) to the front side of the plane. If [H_2, T_2] inside [H_1, T_1], the C_2 is the children of the C_1.

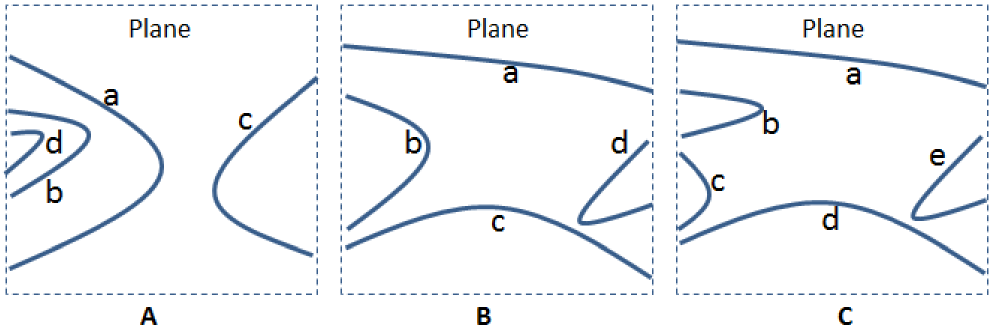

Figure 42. (A) Line segment chain (b) is the children of the line segment chain (a). (B) Line segment chains (b, d) are the children of the line segment chains (a, c). (C) Line segment chains (b, c, e) are the children of the line segment chains (a, d).

The last algorithm is to create new closed line segment chain set by connecting above classified opened line segment chains together. Three connection methods are shown in figure 43. The re-connection starts at the most lower left corner in a counterclockwise direction. At the end of this step, all the open line segment chains are closed.

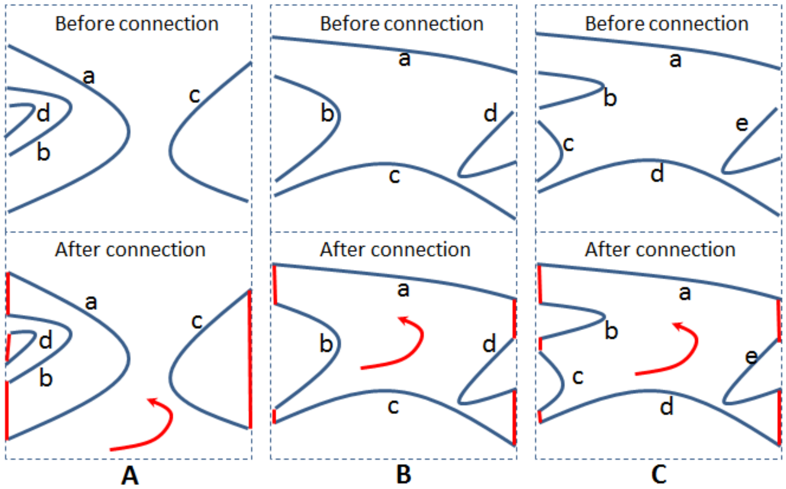

Figure 43. (A) Line segment chains (a, b, d) are classified into the left set. Line segment chain (c) is classified into the right set. Because only b is directly inside a, (a, b) are connected together. Because no line segment chain is inside c and d, the first line segment and the last line segment of the c and d are connected together respectively. (B) Line segment chains (a, b, c) are classified into the left set. Line segment chain (d) is classified into the right set. Because b and d are inside a and c, line segment chains (a, b, c, d) are connected together. (C) Line segment chains (a, b, c, d) are classified into the left set. Line segment chain (e) is classified into the right set. Because b, c and e are inside a and d, line segment chains (a, b, c, d, e) are connected together. The counterclockwise red arrows are the direction of re-connection.

<u>Step 5: Grouping and recombining all the closed line segment chains based on different planes.</u> The algorithms used in this step are similar with Step 4 and the difference is the algorithm here is to separate a given plane into two sub-planes based on the relationships among closed line segment chains on the same plane. It includes four functions.

The first function is to re-order the closed line segment chains which are lying on the same plane in ascending order by the distance to the left side of the given plane (Figure 44).

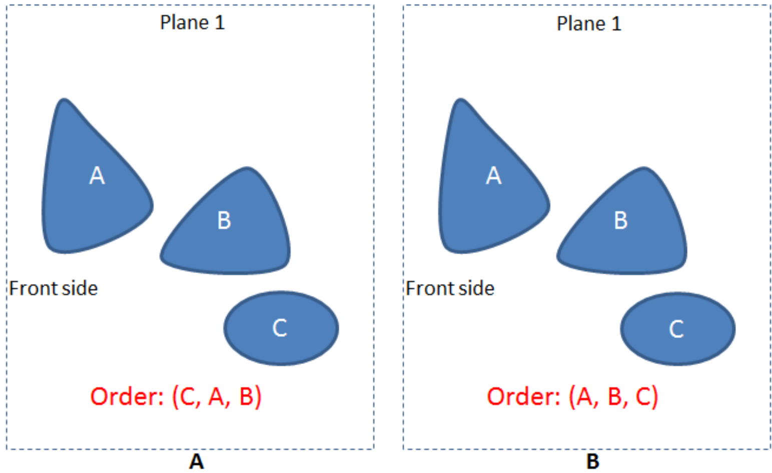

Figure 44. (A) Before re-ordering. (B) After re-ordering. Chain A has the shortest distance to the left side of plane 1. Chain B and C are further away from the left side of plane 1 respectively. After re-ordering, the order is rearranged as it is shown in (B).

The Second function is to project closed line segment chains into 2D coordinate systems based on the given plane. For instance, assuming V(x, y, z) is the normal vector of the plane. If MAX (x, y, z) is y, the closed line segment chains which on the plane will be projected into x-z coordinate systems (Figure 45).

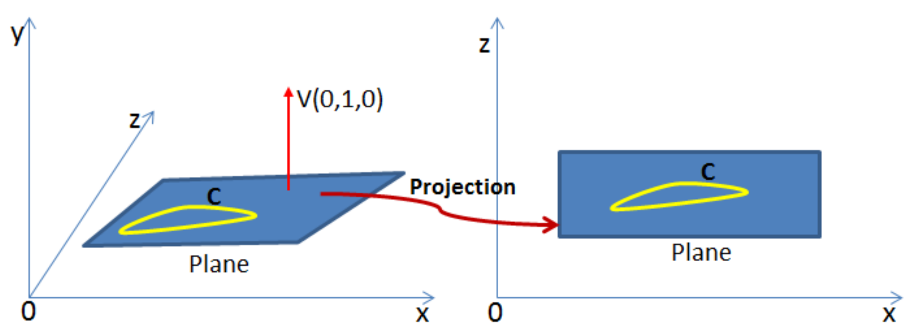

Figure 45. V is normal vector of the plane. Because y is the largest value in V, the line segment chain C is projected into the x-z coordinate systems.

The third function is to determine whether a closed line segment chain is inside of a closed line segment chain. Assuming R is a ray emitted by a selected point in the closed segment chain B. If the number of intersection points between R and another closed segment chain A is odd, B is inside A. Otherwise, B is outside A (Figure 46).

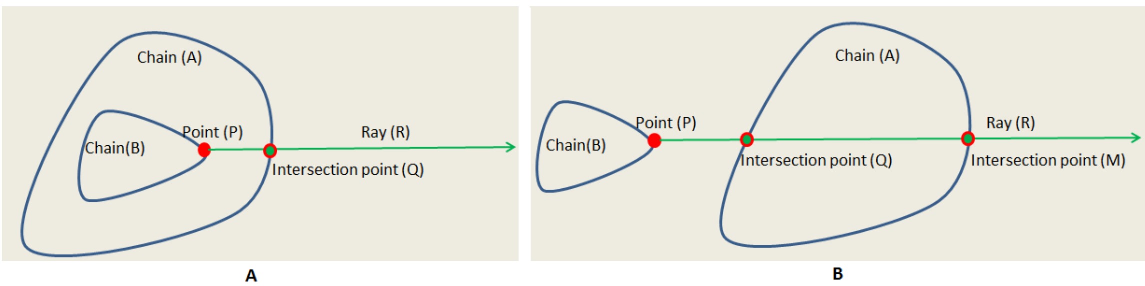

Figure 46. (A) P is a selected point in chain B. R is a ray emitted by P. Q is intersection point between R and the chain A. Because the number the intersection point is 1, B is inside A. (B) Q and M are intersection points between R and A. Because the number of intersection points are 2, B outside A.

The last function is to separate the plane into two sub-planes based on the relationships among closed line segment chains on the same plane (Figure 47).

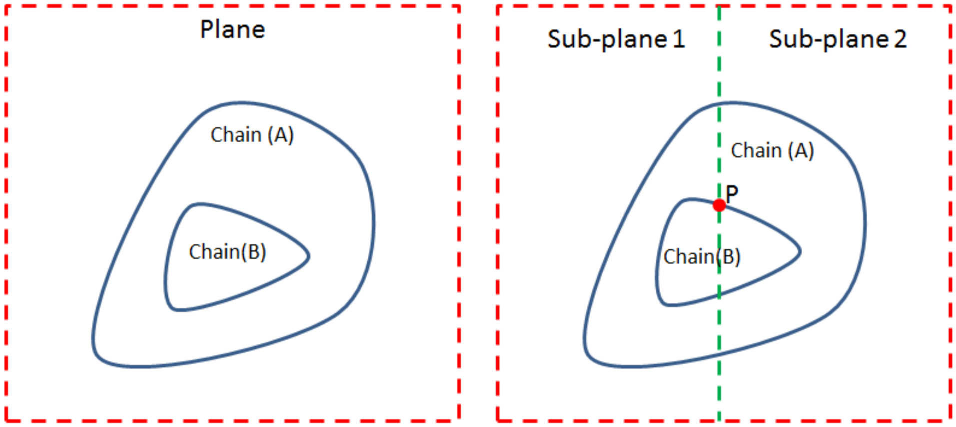

Figure 47. Because chain B is inside of chain A, the plane is separated into two sub-planes by the green line which passes through the selected point P on chain B.

Since the give plane is separated into two sub-planes, we can use similar algorithms used in step 4 to process closed line segment chains. After this step, each closed line segment chain will be no longer inside any other closed line segment chain on the same plane (Figure 48).

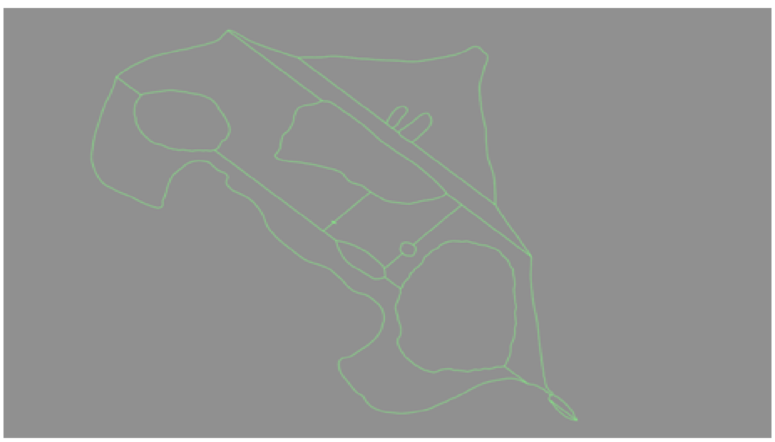

Figure 48. Recombination result. After recombination, each closed line segments set do not share any line segment with others.

Step 6: Triangulating each closed line set and combining triangulated data sets together. A closed surface is thus created using the line segments on the intersection plane (Figure 49).

Figure 49. Triangulation result.

3.4.2.4 Operation 4: Creating the final digital models. There are three steps in the operation. The complete process of operation 4 is shown in Figure 50.

Step 1: Choosing seed points. The seed points are chose from new upper intersection, new lower intersection, and new closed surfaces triangle sets respectively.

Step 2: Combing outside, new upper intersection, new lower intersection, and new closed surfaces triangle sets together. A new temporary digital model is created in the step.

Step 3: Separating the temporary digital model into two pieces by using regional growing method. The seed points are determined in step 1.

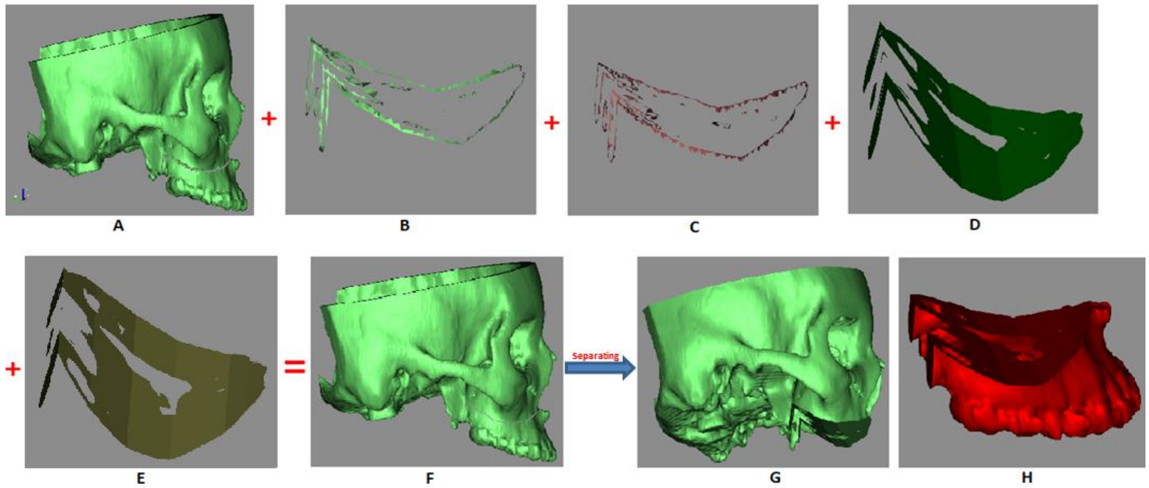

Figure 50. (A) Outside triangle set. (B) New upper intersection triangle set. (C) New lower intersection triangle set. (D) New upper closed surface. (E) New lower closed surface. (F) Temporary digital model. (G) (F) Two final digital models.

3.5 Module 5: Surgical simulation

After the bones are osteotomized, the user can move and rotate the bone segments to the desired position. Surgeon can perform several surgical simulations to compare the outcomes. The basic functions are the same as the Module 2. Figure 51 shows the bones before and after surgical simulation.

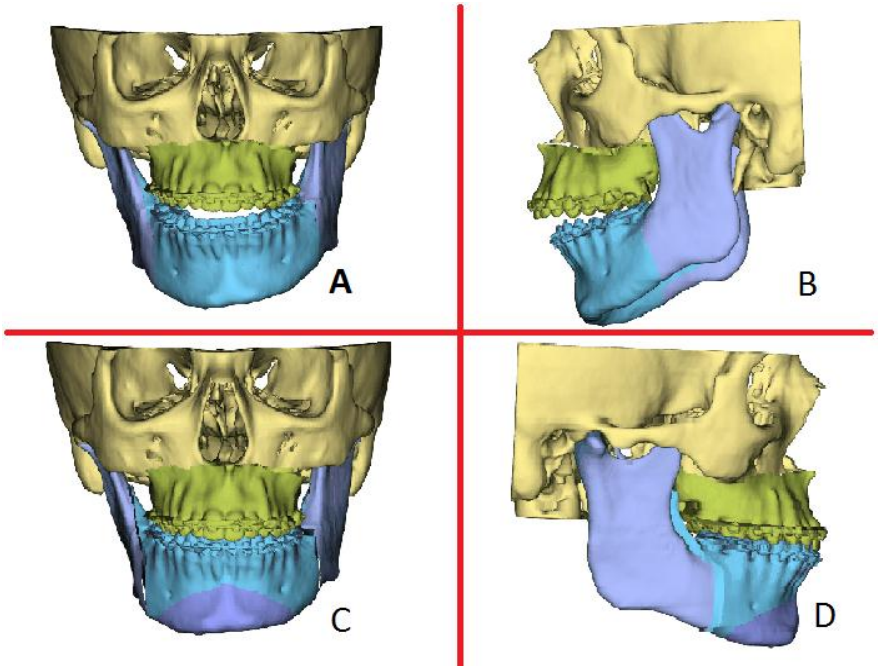

Figure 51. (A) and (B) are bones of skull before performing surgical simulation. (C) and (D) illustrate the surgical simulation in the computer by successively moving and rotating bone segments.

3.6 Module 6: Surgical splint/template fabrication

After the surgical simulation is completed, the final stage in our system is to transfer the computerized surgical plan to the patient at the time of the surgery. Surgical splints and surgical templates are created in this module. In procedures that involve the teeth, the surgical dental splints are created by inserting a digital wafer between the maxillary and mandibular dental arches. A Boolean operation is then performed, resulting in a digital surgical splint (Figure 52 A).In procedures that do not involve the teeth, the surgical template is created to record the 3D surface geometry of the area of interest to fit the bony segment (e.g., chin segment) onto the recipient bone (e.g., mandibular distal segment) in a unique position (Figure 52 B). In procedures that require bone grafts to achieve symmetry, the computerized mirror-imaging technique can be used to provide a mirror image of the geometry from the healthy side to the affected side. The difference between the 2 sides can then be computed, resulting in a digital template that is used at surgery to harvest and sculpt the graft. Finally, the system exports the digital splints and templates in ".stl" format. They are then fabricated using a rapid prototyping machine and used at the time of the surgery (Figure 52 C-F)

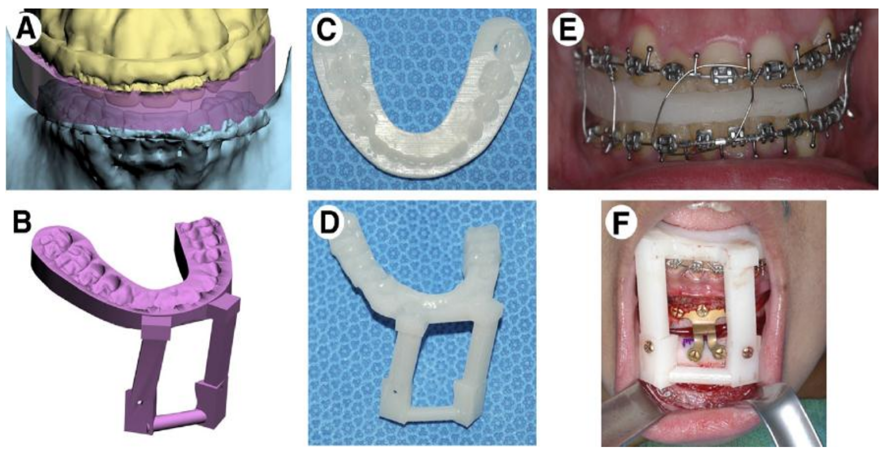

Figure 52. Surgical dental splints and templates created using our computer-aided designing/computer-aided manufacturing technique. (A) Digital surgical splint. (B) Digital chin template. (C) Physical surgical splint. (D) Physical chin template. (E) Use of physical surgical splint at surgery. (F) Use of physical chin template at surgery.

4. CONCLUSIONS

The ideal surgical plan is one that can be accurately reproduced in the operating room. Because of the problems discussed in Chapter one, it is evident that the traditional methods used to plan CMF surgery are often inadequate. In addition, it is also known that in many cases these methods produce unwanted outcomes. Moreover, the whole planning process is time consuming. An experienced surgeon frequently spends 4-6 hours to complete the surgical plan and to fabricate the splints. Finally, the cost of planning a complex case, both in time and in resources can be fairly high. The need to improve the traditional surgical planning methods has led us to develop a 3D CASS. With our CASS system, surgeon can perform "virtual surgery" and create a 3D prediction of the patient's surgical outcomes as if they are performing surgery in the operation room. There are following advantages in our CASS system:

1. The first advantage is that surgeons can not only plan the surgery in the computer, but also accurately transfer the computerized surgical plan to the patient in the operation room by using computer-generated surgical splints and the templates [8] [10]. (ref. 2009 Xia, 2005 Xia 2008 Gateno)

2. The second advantage is to significant save surgeon's time. Instead of spending 6 hours in the laboratory to develop a surgical plan and making surgical splints, the data preparation can be completed using a computer. The surgeon's time is reduced from 6 hours to 30 minutes. It not only save the surgeon's time, but also indirectly

reduce the surgeon's cost because surgeons will have more time seeing more patients [20].

3. The third advantage is to improve the surgical outcome and accuracy. A Study showed that the surgical outcomes achieved with CASS are significantly better than the outcomes achieved with the traditional method [21]. In addition, the accuracy of the surgery is that the surgeon can precisely reproduce the surgical plan that is developed in the lab. Another study showed the exceptional agreement between the planned and the actual outcomes [15] [22].

REFERENCES

[1] Santler G, “3-D cosmos: A new 3D model based computerized operation simulation and navigation system.” *J Craniomaxillofac Surg.*, vol. 28, no. 5, pp. 287-93, Oct. 2000.

[2] Gateno J, Xia JJ, Teichgraeber JF, Christensen AM, Lemoine JJ, Liebschner MA, Gliddon MJ, Briggs ME, “Clinical feasibility of computer-aided surgical simulation (CASS) in the treatment of complex craniomaxillofacial deformities.” *J Oral Maxillofac Surg.*, vol. 65, no. 2, pp. 728-34, Apr. 2007.

[3] Swennen GR, Barth EL, Eulzer C, Schutyser F, “The use of a new 3D splint and double CT scan procedure to obtain an accurate anatomic virtual augmented model of the skull. ” *Int J Oral Maxillofac Surg.*, vol. 36, no. 2, pp. 146–52, Feb. 2007.

[4] Swennen GR, Mommaerts MY, Abeloos J, De Clercq C, Lamoral P, Neyt N, Casselman J, Schutyser F, “The use of a wax bite wafer and a double computed tomography scan procedure to obtain a three-dimensional augmented virtual skull model.” *J Craniofac Surg.*, vol. 18, no. 3, pp. 533–9, Feb. 2007.

[5] Troulis MJ, Everett P, Seldin EB, Kikinis R, Kaban LB, “Development of a threedimensional treatment planning system based on computed tomographic data.” *Int J Oral Maxillofac Surg.*, vol. 31, no. 4, pp. 349–57, Aug. 2002.

[6] Xia J, Ip HH, Samman N, Wang D, Kot CS, Yeung RW, Tideman H, “Computer-assisted three-dimensional surgical planning and simulation: 3D virtual osteotomy. ” *Int J Oral Maxillofac Surg.*, vol. 29, no. 1, pp. 11–7, Feb. 2000.

[7] Xia J, Samman N, Yeung RW, Shen SG, Wang D, Ip HH, Tideman H, “Three-dimensional virtual reality surgical planning and simulation workbench for orthognathic surgery. ” *Int J Adult Orthodon Orthognath Surg.,* vol. 15, no. 4, pp. 265–82, 2000.

[8] Xia JJ, Gateno J, and Teichgraeber JF, “Three-dimensional computeraided surgical simulation for maxillofacial surgery,” *Atlas Oral Maxillofac Surg Clin North Am.*, vol. 13, no. 1, pp. 25–39, Mar. 2005.

[9] Bell WH and Guerrero CA, *Distraction Osteogenesis of the Facial Skeleton*. BC Decker. Hamilton, Ontario, 2007.

[10] Xia JJ, Gateno J, and Teichgraeber JF, “New Clinical Protocol to Evaluate Craniomaxillofacial Deformity and Plan Surgical Correction,” *Journal of Oral and Maxillofacial Surgery*, vol. 67, no. 10, pp. 2093–106, Oct. 2009.

[11] Xia J, Ip HH, Samman N, Wong HT, Gateno J, Wang D, Yeung RW, Kot CS, Tideman H, “Three-dimensional virtual-reality surgical planning and soft-tissue prediction for orthognathic surgery,” *IEEE Trans Inf Technol Biomed*, vol. 5, no. 2, pp. 97–107, Jun. 2001.

[12] Gateno J, Teichgraeber JF, Xia JJ, “Three-dimensional surgical planning for maxillary and midface distraction osteogenesis,” *J Craniofac Surg.*, vol. 14, no. 6, pp. 883–9, Nov. 2003.

[13] Gateno J, Teichgraeber JF, Aguilar E, “Computer planning for distraction osteogenesis,” *Plast Reconstr Surg.*, vol. 105, no. 3, pp. 873–82, Mar. 2000.

[14] Gateno J, Allen ME, Teichgraeber JF, Messersmith ML, "An in vitro study of the accuracy of a new protocol for planning distraction osteogenesis of the mandible," *J Oral Maxillofac Surg.*, vol. 58, no. 9, pp. 985–90, Sep. 2000.

[15] Malis DD, Xia JJ, Gateno J, Donovan DT, Teichgraeber JF, "New protocol for 1-stage treatment of temporomandibular joint ankylosis using surgical navigation," *J Oral Maxillofac Surg.*, vol. 65, no. 9, pp. 1843–8, Sep. 2007.

[16] Gateno J, Xia J, Teichgraeber JF, Rosen A, "A new technique for the creation of a computerized composite skull model," *J Oral Maxillofac Surg.*, vol. 61, no. 2, pp. 222–7, Feb. 2003.

[17] Gateno J, Xia J, Teichgraeber JF, Rosen A, Hultgren B, Vadnais T, "The precision of computer-generated surgical splints," *J Oral Maxillofac Surg.*, vol. 61, no. 7, pp. 814–7, Jul. 2003.

[18] Lorensen WE and Cline HE, "Marching cubes: A high resolution 3D surface construction algorithm," *Computer Graphics*, vol. 21, no. 4, pp. 163–169, 1987.

[19] Bell WH, *Surgical correction of dentofacial deformities.* Philadelphia: WB Saunders, Jun. 1980.

[20] Xia JJ, Phillips CV, Gateno J, Teichgraeber JF, Christensen AM, Gliddon MJ, Lemoine JJ, Liebschner MA, "Cost-effectiveness analysis for computer-aided surgical simulation in complex cranio-maxillofacial surgery," *J Oral Maxillofac Surg.*, vol. 64, no. 12, pp. 1870–4, Dec. 2006.

[21] Xia JJ, Shevchenko L, Gateno J, Teichgraeber JF, Taylor TD, Lasky RE, English JD, Kau CH, McGrory KR, "Outcome study of computer-aided surgical simulation

in the treatment of patients with craniomaxillofacial deformities," *J Oral Maxillofac Surg.*, vol. 69, no. 7, pp. 2014–24, Jul. 2011.

[22] Hsu SS, Gateno J, Bell RB, Hirsch DL, Markiewicz MR, Teichgraeber JF, Zhou X, Xia JJ, "Accuracy of a computer-aided surgical simulation protocol for orthognathic surgery: a prospective multicenter study," *J Oral Maxillofac Surg.*, vol. 71, no. 1, pp. 128–42, Jan. 2013.

[23] Besl, P.J., McKay, Neil D., "A method for registration of 3-D shapes," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 14, no. 2, pp. 239–256, Feb. 1992.

APPENDIX A

THE ALGORITHM OF ELIMINATING OVERLAPS ALONG THE ADJACENT LINE SEGMENTS

Input: S_1 (*S_1 is original line segment set)*

Output: S_2 (*S_2 is new line segment set. Each line in S_2 does not share any region with others.)*

```
1. S2 = NULL      (Initial S2 with empty.)
2. while S1 is not empty do
       L = the first line segment in S1
       S3 = NULL          (Initial S3 (line segment set) with empty.)
       if  S2 is empty  then
           Push L into S2
       else
           i=0
           while  i < the size of  S2 do
               L1 = the i-th line segment in S2
               D = the distance from  L1 to L
               if  (D <  0.0001 ) and (L1 ,L are parallel to each other)
                   Push  L1 into S3
                   Delete the i-th line segment in S2
                   Continue
               end if
               i = i+1
           end while
           if  S3 is not empty then
               L2 = merging all line segments in S3 and L together.
               Push  L2 into S2
           else
               Push  L into S2
           end if
       end if
       delete the first line segments in S1
   end while
```

APPENDIX B

THE ALGORITHM OF CREATING CLOSED LINE SEGMENT CHAINS

Input: S_1 (*S_1 is original line segment set.)*

Output: S_2 (*S_2 is line segment chain set.)*

while *S_1 is not empty* **do**

    C = NULL (*Initial a line segment chain(C) with empty*.)

    L = the first line segment in S_1

    Push L into *C*

    Delete *first line segment in S_1*

    T=False

    while *T* == False **do**

    L_1 <q_1, q_2> = the last line segment in C

    L_2 <p_1, p_2> = the i-th line segment with shortest distance to q_2 in S_1

    D_1 = the distance from the point (q_2) to the point (p_1)

    D_2 = the distance from the point (q_2) to the point (p_2)

    if ($D_1 < D_2$ **and** $D_1 < 0.001$)

        L_3 <q_2, p_2>

        Push L_3 into C (Connecting L_1 and L_2 together*)*

        Delete *the i-th line segment in* S_1

    else if ($D_2 < D_1$ **and** $D_2 < 0.001$)

        L_3 <q_2, p_1>

        Push L_3 into C (Connecting L_1 and L_2 together*)*

        Delete *the i-th line segment in* S_1

    else

        T =True

    end if

    end while

    if *C is a closed line segment chain* **then**

        Push C into S_2 (*Add* a *line segment chain* into *line segment chain set)*

    else

        Informing user to re-define cutting line.

    end if

end while
