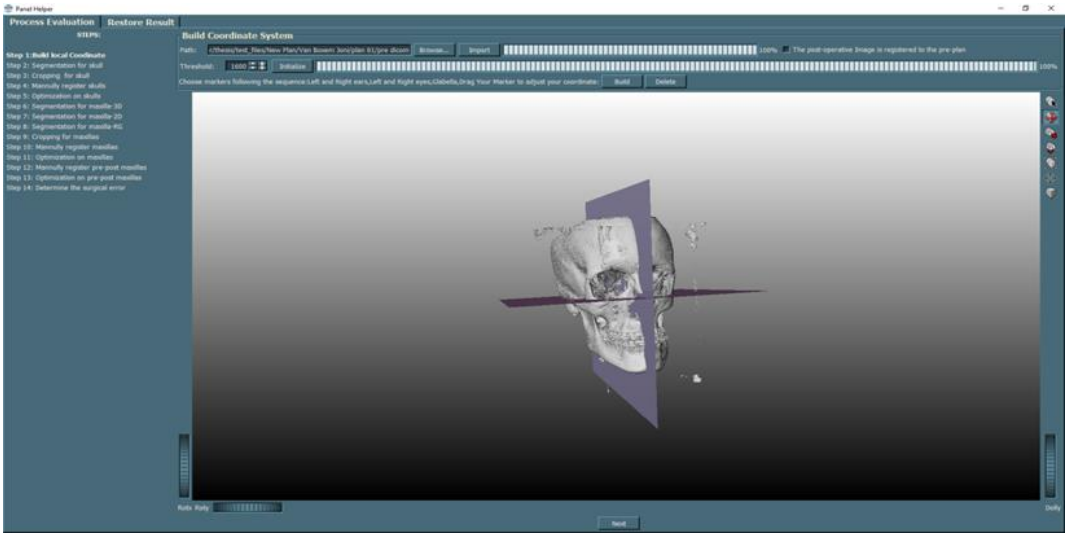
The guide of OrthoCalc

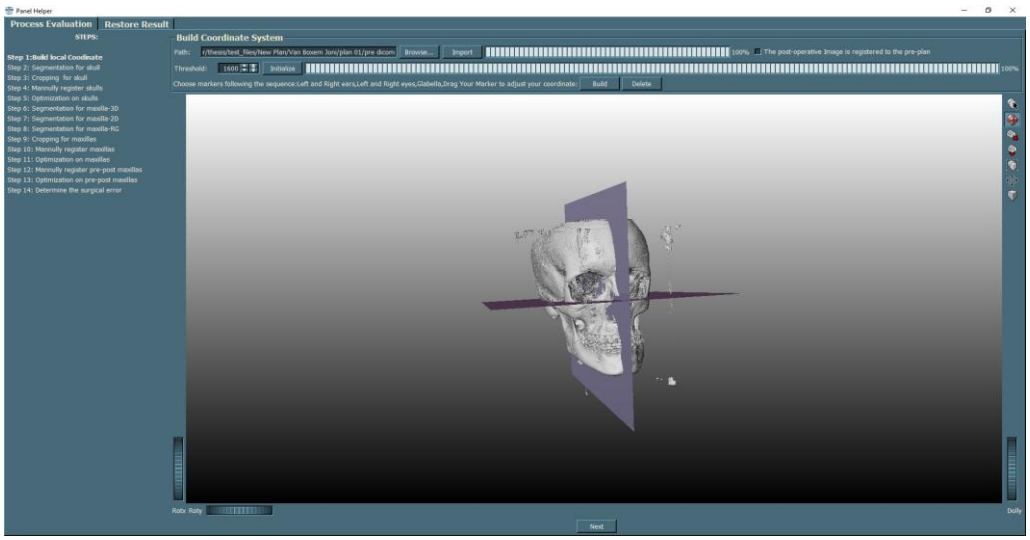
yao.gao@kuleuven.be

This document will introduce how our application named OrthoCalc is implemented. Users do not have to be very familiar with the work flow, all you need to do is follow the wizard steps shown in the left “Navigation Window”. Since most of the operations are about modify, manipulate and observe the images, the control panels, which are located above the views, only occupy a very limited area of the whole GUI. Giving most of the area to the views on which users can modify, manipulate and check the images easily.



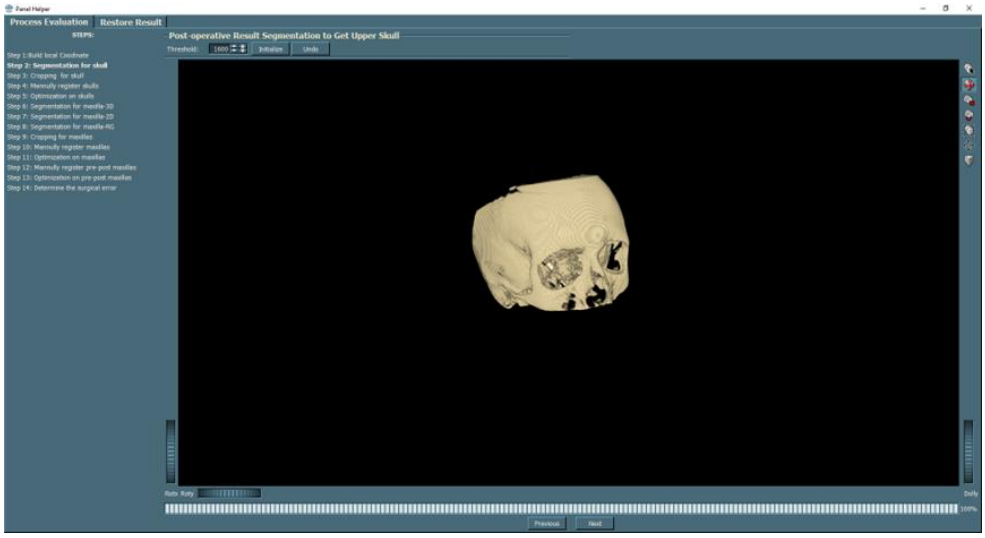
- **Step 1: Load post-surgical (CB)CT scan and Build Coordinate System**

This step is the beginning of the whole work flow, in this step users can complicate the “Coordinate Construction” part and some loading and pro-processing parts on the work flow figure. As shown in Figure below, in the first step of “Process Evaluation”, users can load the DICOM image which represent the (CB)CT scan of a post-surgical patient. In our network, the Reample3D module is used to down sample the (CB)CT images. By determine the voxel size to 0.4-0.5 and set image size to constant, the down sampling is performed.



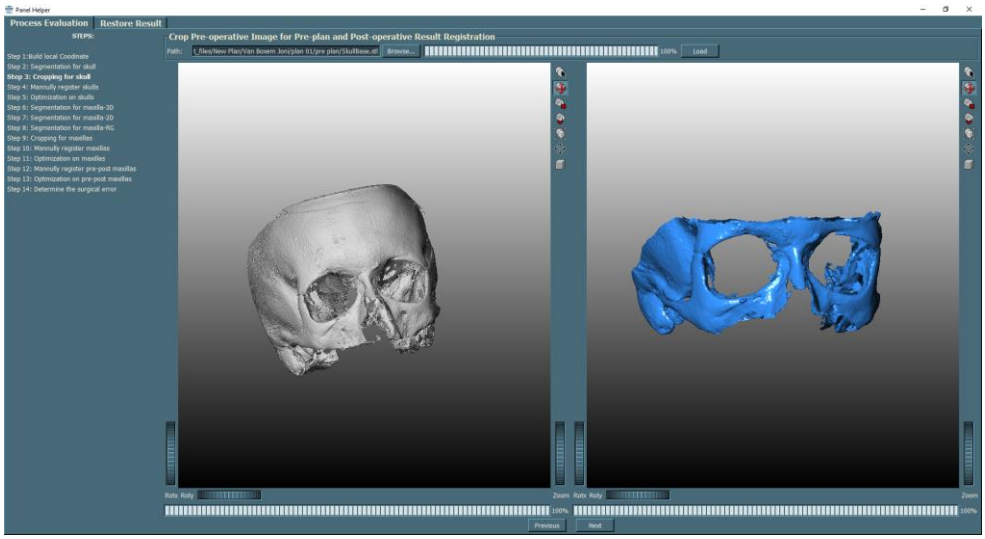
- **Step 2: Cropping (CB)CT DICOM to get 3D model of post-operative upper skull**

The input of the step is a down sampled and thresholder DICOM image of (CB)CT scan of post-surgical patient. As shown below, users can adjust the threshold if they are not satisfied the threshold from step 1. Then they can simply crop out the maxilla and mandible part of the image. The given figure shows a cropped skull.



- **Step 3: Crop post- and pre-surgical skulls**

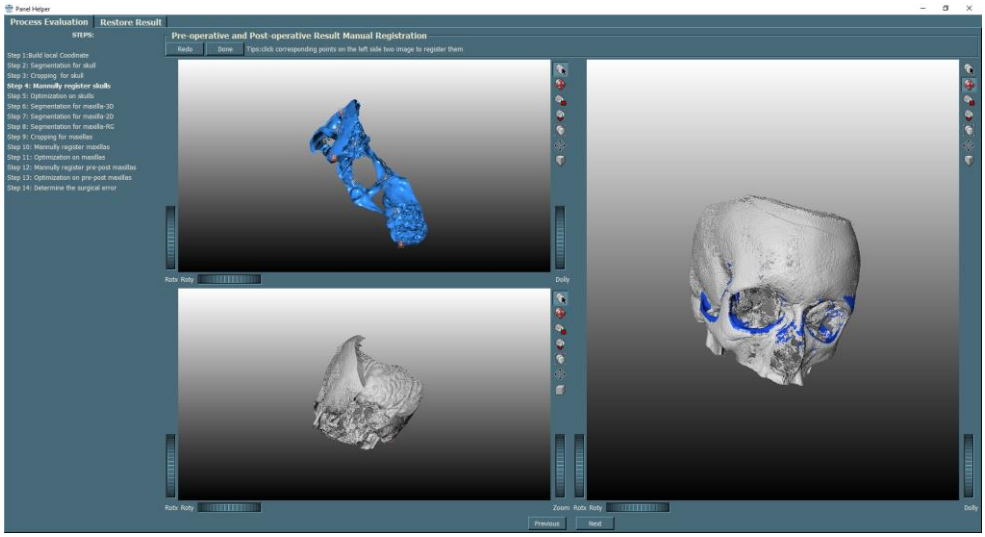
As shown below, users should load the patient's skull image from pre-surgical plan (Blue), crop it for registration. To ensure the accuracy of registration, users should crop all the parts on the pre-surgical skull to make sure that the remained part can be completely overlapped by the post-surgical skull from step 2(Grey part in Figure 3.11). If the user is not satisfied with the result of the cropping in step 2.They can still crop the skull in this step.



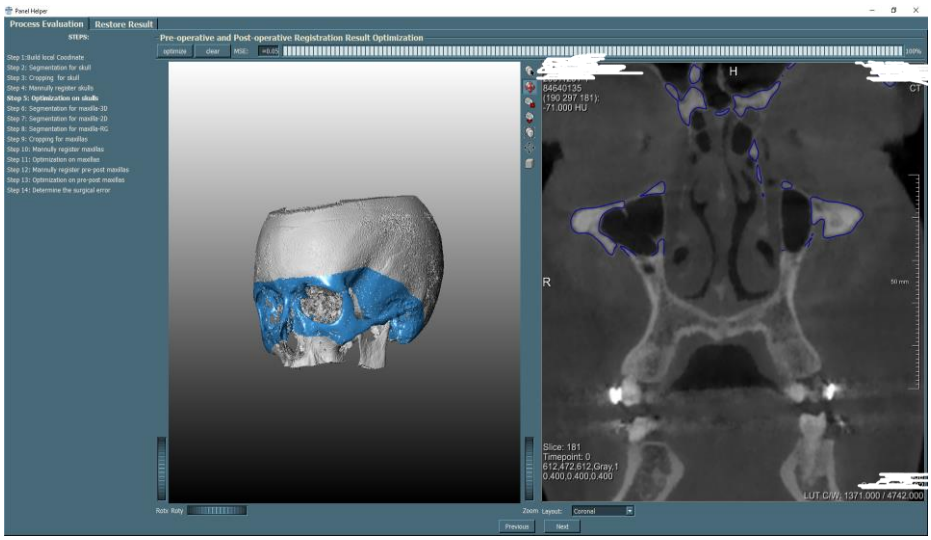
- **Step 4 & Step 5: Register pre-surgical skull to post-surgical skull**

After all the pre-processing in previous steps, users can finally perform the first registration in the whole evaluation procedure. As mentioned in section 2.3 and 2.5. Before performing ICP registration. It is necessary to roughly register the input image pair to improve the accuracy and speed of ICP registration. Therefore, as shown below, in step 4, users should manually register

the pre-surgical skull and post-surgical skull by choosing corresponding point pairs on each image.



After manual registration, the registered pre-surgical skull and the post-surgical skull will be send to step 5 for ICP registration.

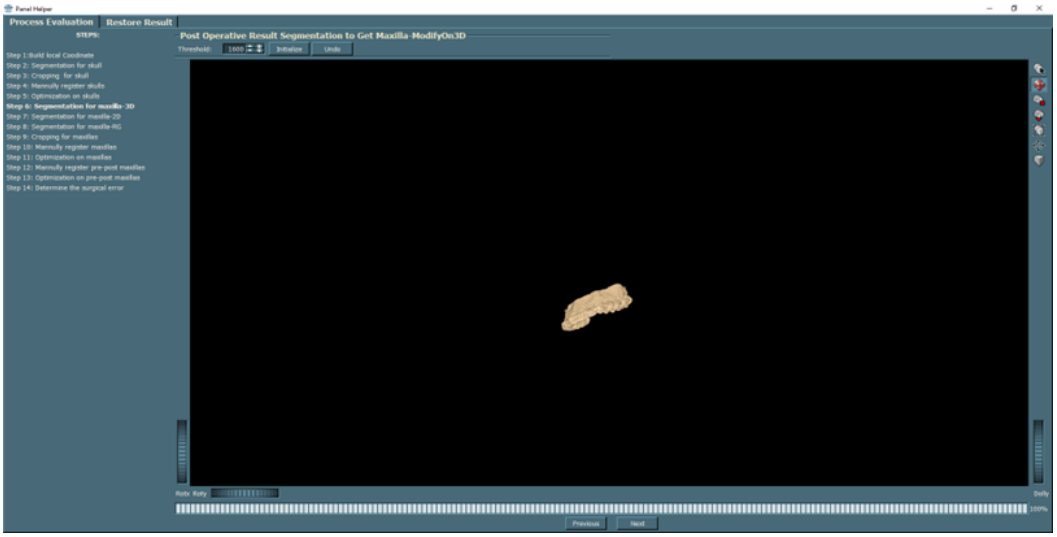


- **Step 6 & Step 7 & Step 8: Segmentation of post operative maxilla on- 3D,/2D**

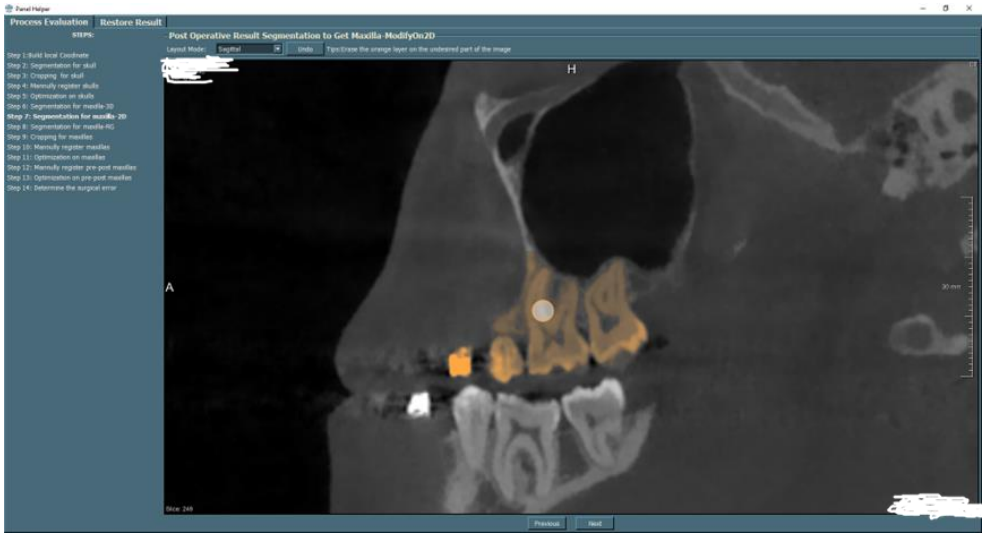
These 3 steps are the implementation of the "Segmentation on the post-surgical CBCT Image for maxilla" block in the work flow. The segmentation operations in the steps are still on (CB)CT DICOM image of post-surgical patient.

The input image is exactly the same as the input of step 2.But to get the maxilla instead of the upper skull of the image. Users need to perform more complicated operations because patients may close their teeth during (CB)CT scanning.

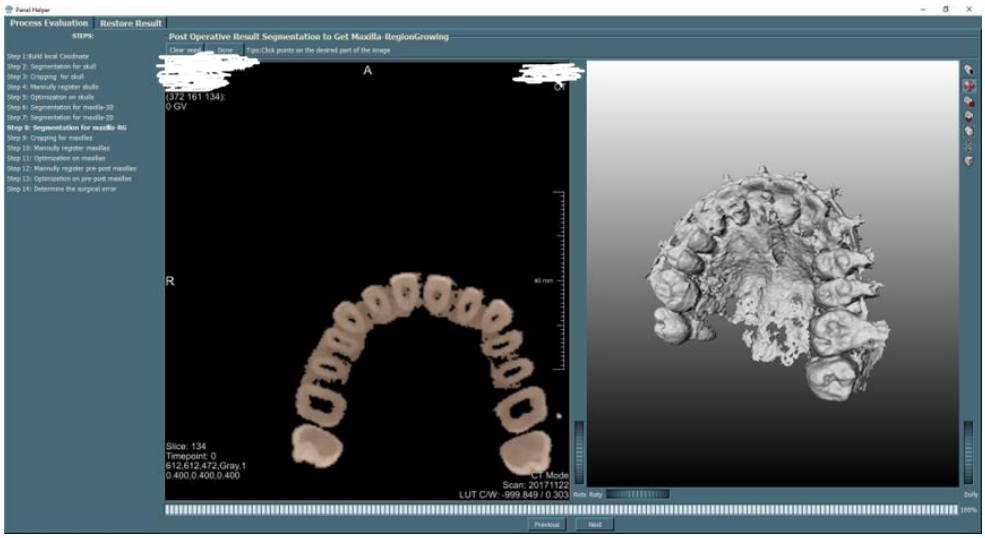
Firstly, in step 6, user should crop out as many nodes as they can on the image except the maxilla. The GUI and back stage network and operation is exactly the same as step 2.



Therefore, in step 7, users can erase the pixels of mandible on every (CB)CT slice.

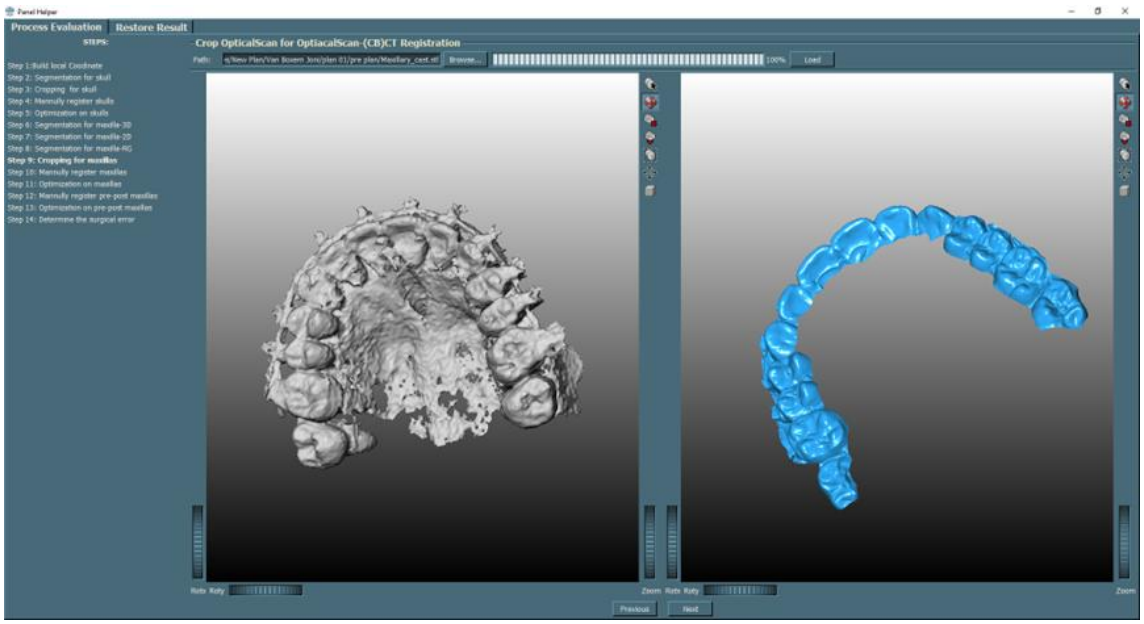


After erasing all the connected parts between maxilla and mandible. Users can segment the remained mandible on step 8 using “Region Growing” method. User can simply choose a point from the maxilla part of the image on a 2D view as seed. Then the module “Region Growing” will remove all the unconnected points.



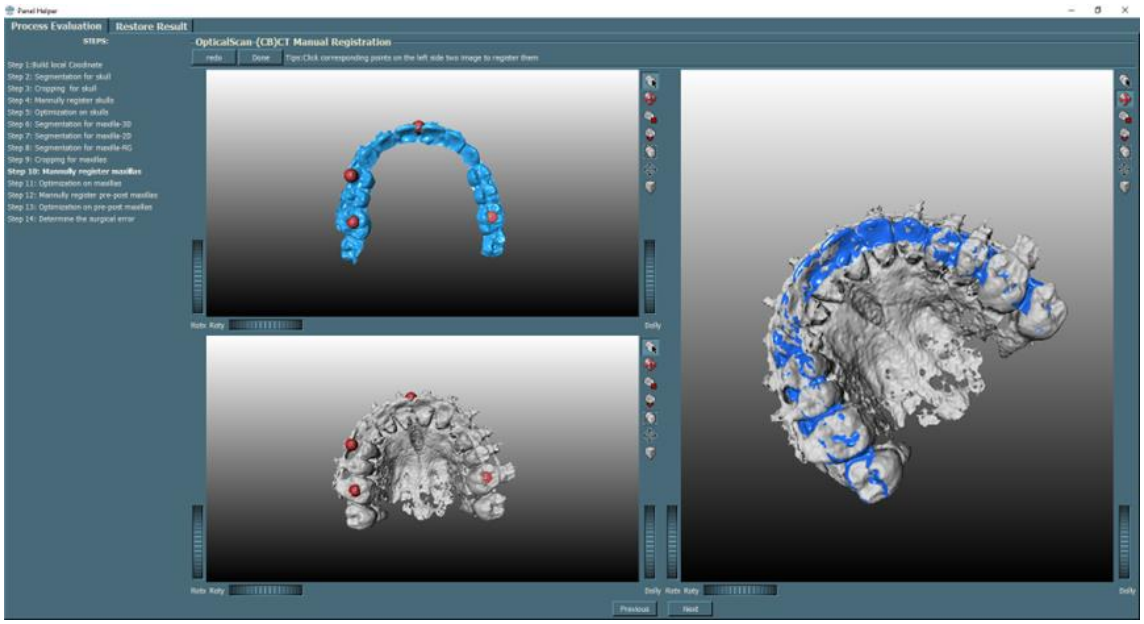
- **Step 9: Crop (CB)CT scan of post-surgical maxilla and optical can of maxilla**

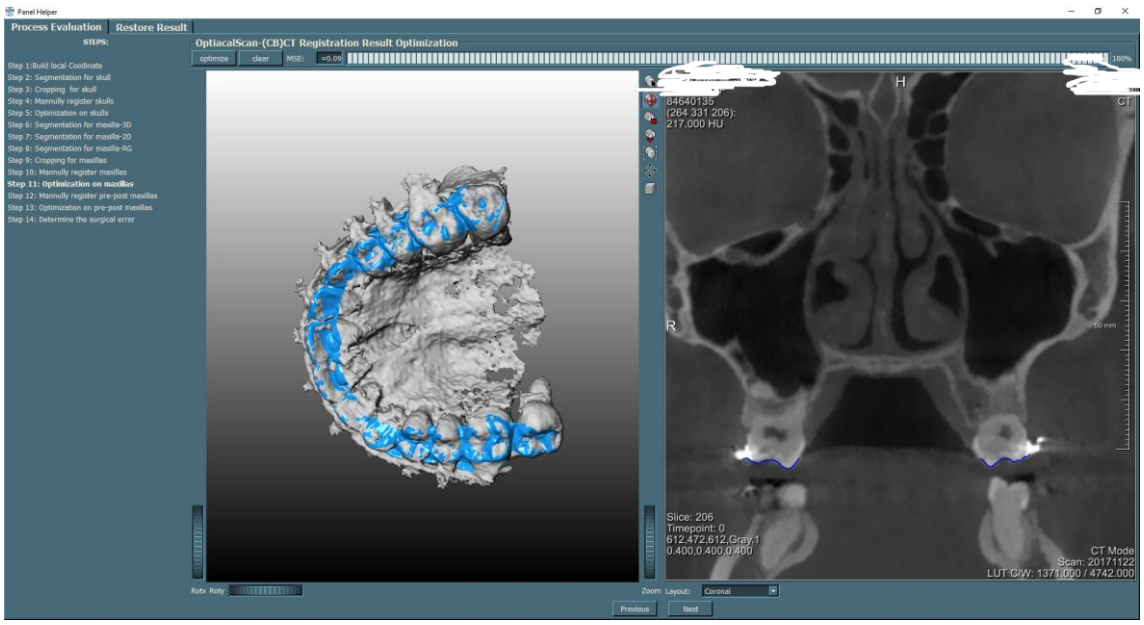
This step is the implementation of the "further segmentation and pre-processing" block before the "register the optical-scan of the post-surgical maxilla to the post-surgical maxilla" block of the work flow. The GUI and back stage modules of the step are exactly the same as step 3. The only difference is the input.



- **Step 10 & Step 11: Registration from optical-scan of maxilla to the (CB)CT scan of post-surgical maxilla**

These two steps are the implementation of the "register the optical-scan of the maxilla to the post-surgical maxilla" block of the work flow. The GUI and back stage modules of the steps are exactly the same as step 4 and step 5. The only difference is the input. After the registration, the registered optical scan of post-surgical maxilla will be used as the input of further registration.

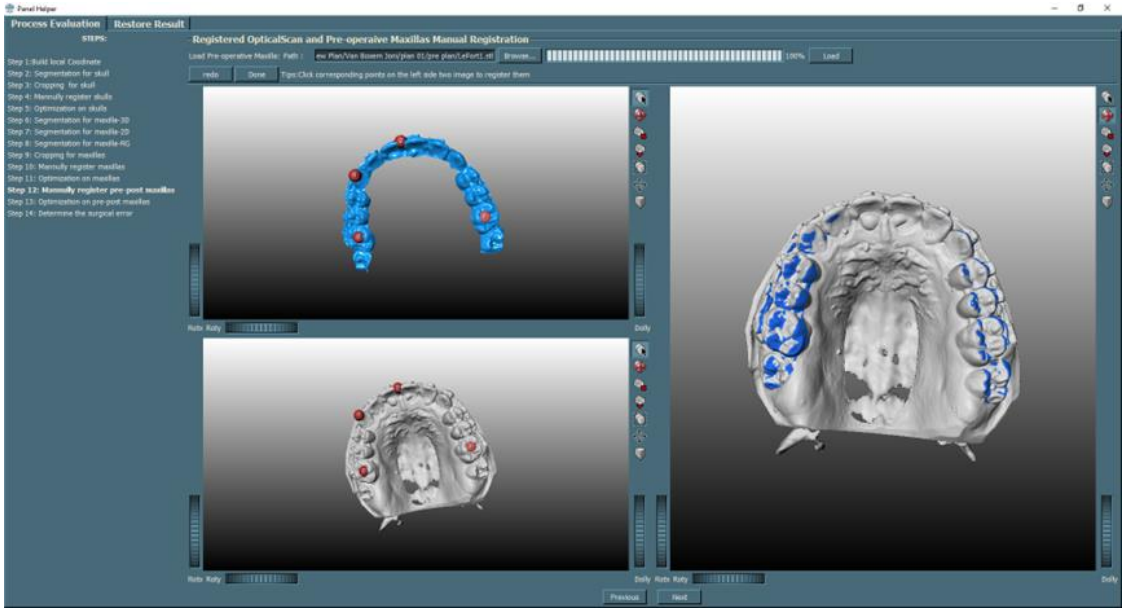


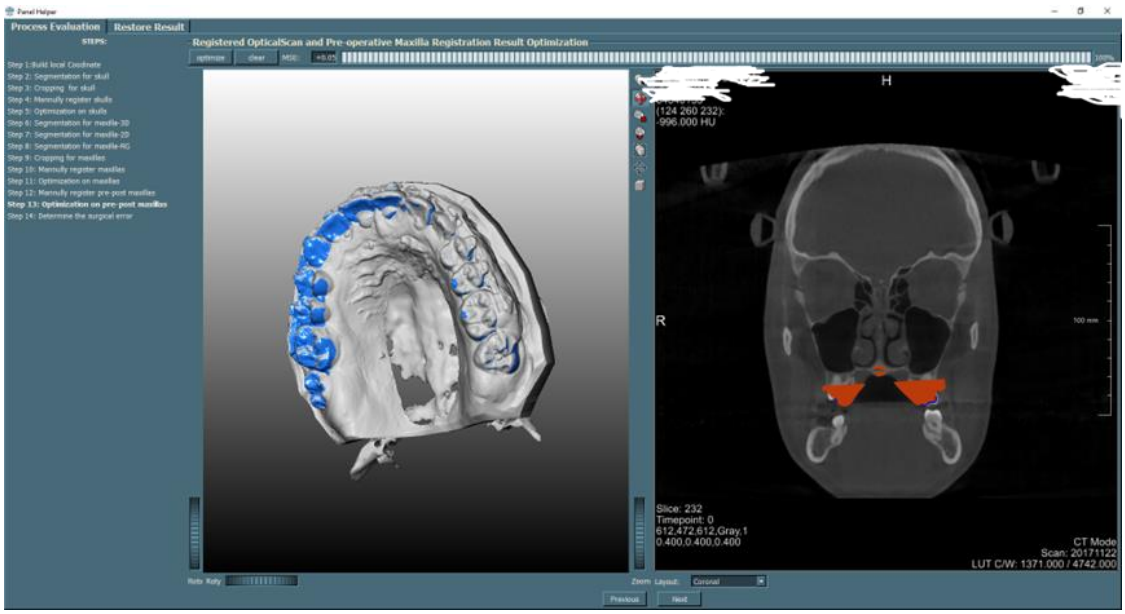


- **Step 12 & Step 13: Registration from matched optical scan of post-surgical maxilla to the pre-surgical maxilla**

These two steps are the implementation of "transform maxilla" and "register the matched optical scan of post-surgical maxilla to the maxilla of pre-surgical plan" blocks.

As shown below. In step 12, users should load the maxilla from pre-surgical plan first. After the loading, the application will automatically perform the transformation matrix from step 5 to the maxilla. Therefore, the only difference between the transformed maxilla and the registered optical scan of post-surgical maxilla is the surgical error that we want to find out using the application. The rest operations and the related GUI pages and back stage modules are exact the same as step 10 and step 11.





- **Step 14: Surgical Error Determination**

This step is the implementation of the "Difference Determination" block of the work flow. As shown in Figure below, the surgical error can be divided into translation difference and rotation difference, therefore, the GUI is also divided into two parts.