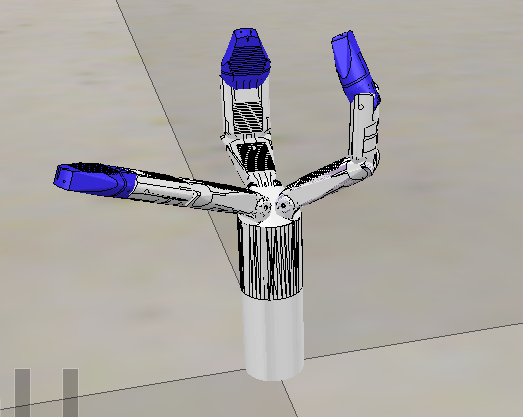
**MUSHUA HAND**

Presented by

Alex Gebinmachor, Neeha Chowdary & Nithin Reji Johnson

Under the guidance of Proff. Marilena Vendittelli

Sapienza Universita di Roma



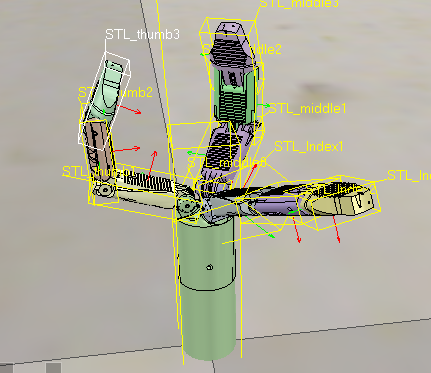
**INTRODUCTION**

Minimal Invasive surgery gaurantees certain advantages when compared with the traditional open surgery followed from times. Today,the use of robots help us to achieve the desired results overcoming many limitations like loss of dexterity, non intiutive skills of the surgical system and bad execution of the task. Our project aims to have an efficient grasping through a simulation, open and close mode.

**OUR WORK**

BUILDING THE SHAPE:

We started the kick off by importing the mesh from an external application in V-REP by using the Menu Bar> File> Import as a mesh. We considered a few factors while importing the mesh ,the file should not be heavy( meaning, it should not contain many triangles) because it would be an hindrance for further calculation modules like minimum distance calculations or dynamics.



1. The imported mesh

In order to avoid using heavy meshes there are list of options in play like

1) Decimate the mesh,

2) Automatic mesh division or manual mesh division(Triangle Edit mode),

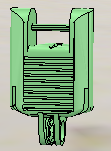
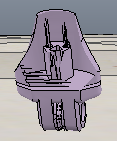
3) Extracting the convex hull,

4) Removing the inside of the mesh.

In our case, these options in iteration for various entities and sub entities worked better and may differ from case to case.

For an instance > the link 1 of the thumb, we first copied in a new scene then used one of the above options “Manual mesh division” which is done through triangle edit mode, and we manually select the triangles that logically belongs together and extract the shape, which generates new scene and then delete the selected triangle from the previous operation.

To simplify further ,we went through a more complicated way via the shape edit mode(Triangle edit mode), where we can extract individual elements that logically belong to the same convex sub-entity.

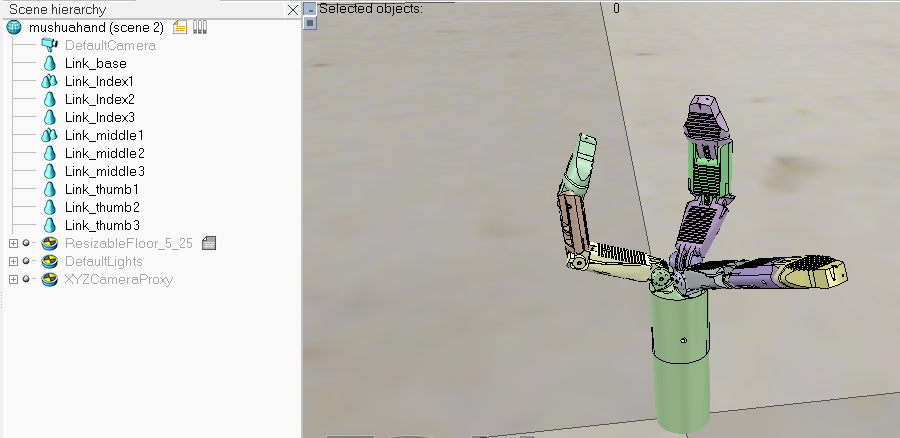
1. The three links of middle finger of the robot

We repeated the same process for all the links of the three fingers .

For better visualization and manipulation we worked by copying several entities or links in various open scenes then copying back to the original scene and saving the progress step by step.

So finally we have link1 to link3 for thumb, link 1 to link3 for the middle finger, and link 1 to link3 for the index finger.

Now we can group the shapes that are part of the same link with Menu bar > Edit > Grouping/merging > Group selected shapes. We end up with 10 shapes: the base of the robot (or base of the robot's hierarchy tree), and 9 mobile links



BUILDING THE JOINTS:

Now we observe that the visible shapes are determined perfectly we now dive deep in building the joints for all the respective links of the three fingers.

Most of the time, we know the exact position and orientation of each of the joints but in our case, we simply add the joints with [Menu bar > Add > Joints --> Revolute], then we can change their position and orientation with the position dialog and orientation dialog.

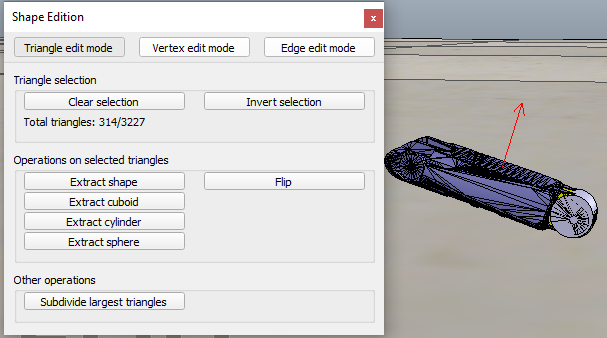
For further simplfication,we will look for revolute shapes, that could be used as reference to create joints at their locations, with the same orientation.

Following the above step we now click Extract cylinder (Extract shape would also work in that case), this just created a cylinder shape in the scene, based on the selected triangles.

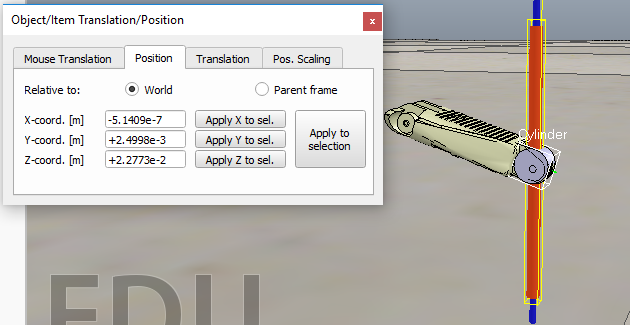
Now we add a revolute joint with [Menu bar > Add > Joint > Revolute], keep it selected, then control-select the extracted cylinder shape. In the position dialog and the orientation dailog, we click Apply to selection: this basically copies the x/y/z position and the orientation of the cylinder to the joint. Both positions and orientations are now identical with respect to the selected shape.

Now we copy the progress made to the original scene and repeat all the above procedures for all the remaining joints.

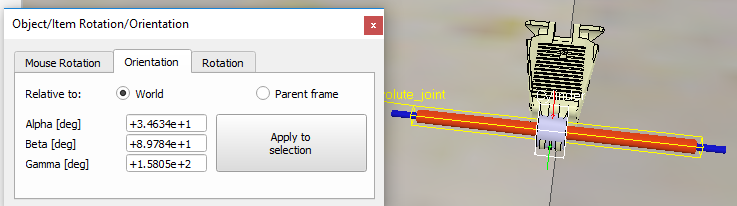
Here is the glimpse of the triangle edit mode and further steps shown diagramatically to place the joint in the desired position and orientation :



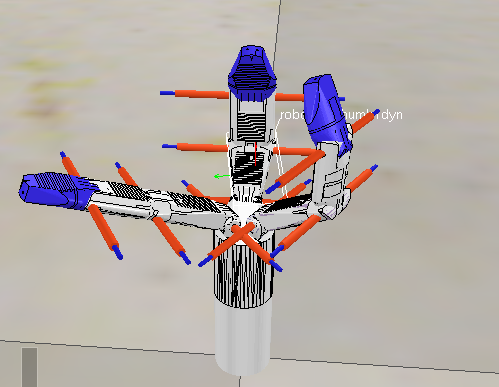
(C)Triangle Edit Mode: extracting shape



1. Assigning the position of the joint wrt shape



1. Assigning the orientation of the joint wrt shape



1. Assigning all the joints in correct configuration

BUILDING DYNAMIC SHAPES:

Now we have build the visible shapes and joints , further if we want our robot to respond to the collisions and fall we need to dynamically enable it.

In order to dynamically enable our robot the shape can be of two types:

1. Static or Non static

2)Respondable or Non respondable

Respondable shapes are further divided into Pure shapes, pure compound shapes, convex shapes, convex compound shapes were used in our work depending upon the need.

For instance,if the object robot was a compound shape, we would first have had to ungroup it by Menu bar > Edit > Grouping/Merging > Ungroup selected shapes then merge the individual shapes by Menu bar > Edit > Grouping/Merging > Merge selected shapes before being able to start the triangle edit mode.

Now we renamed the new shape as robot\_dyn, assign it to visibility layer 9, then copy it to the original scene combined up with the rest of the links modelled as convex shapes, or compound convex shapes renamed it to robot\_link\_dyn respectively.

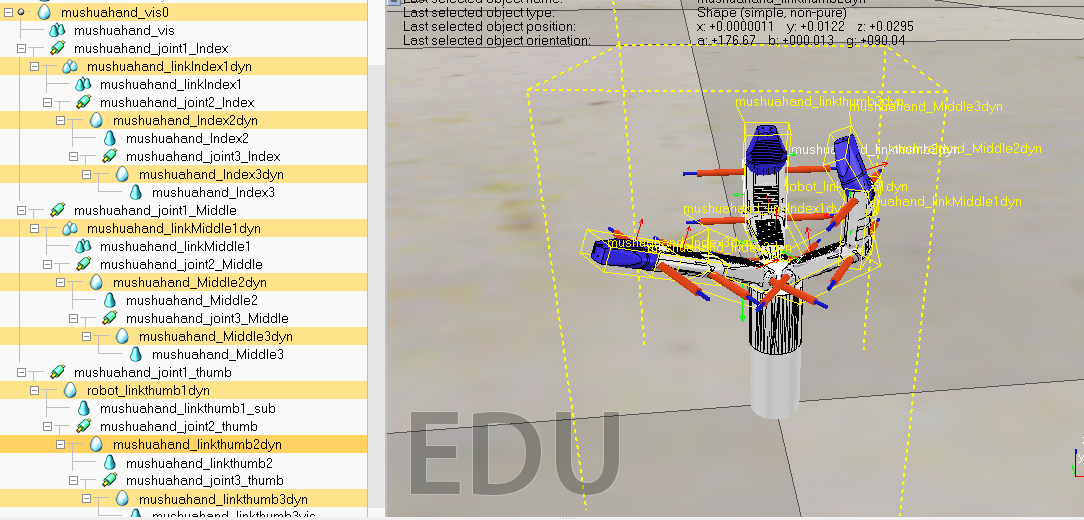
We repeat the same procedure for all the remaining links and once it is done we attached the each visible shape to its corresponding invisible dynamic pendant by making it the parent.

Now in order to configure the dynamic shapes as dynamic and respondable we enable the first 4 Local respondable mask flags, and disable the last 4 Local respondable mask flags.

For the first mobile dynamic link in our robot , we also enable the Body is respondable item, but this time we disable the first 4 Local respondable mask flags, and enable the last 4 Local respondable mask flags. Likewise we do in alteration for the remaining dynamic links of the robot from the base of the robot.

Most of the time we want the base of the robot to be static.

We do this by enabling the Set to dynamic if gets parent item, then disabling the Body is dynamic item.



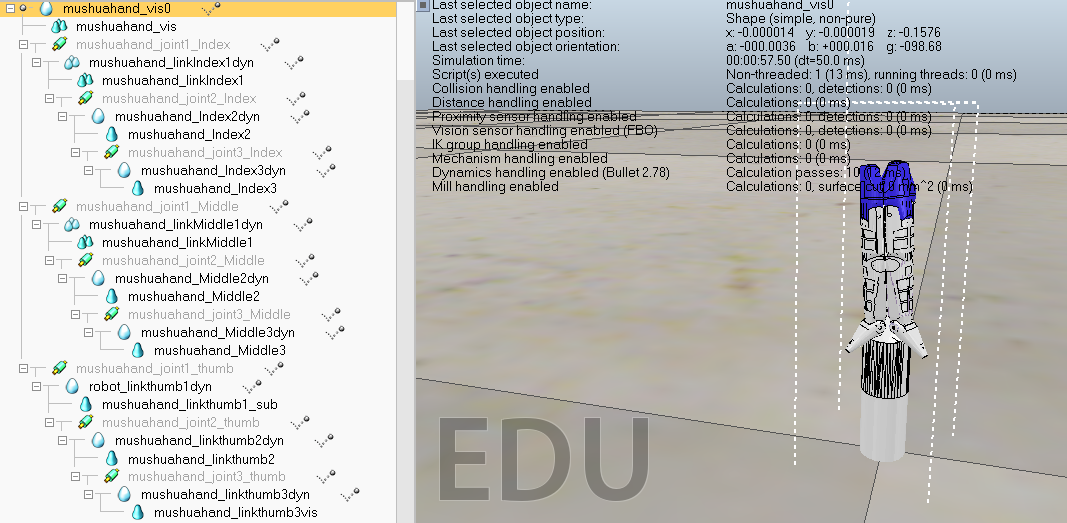
(F)Dynamically enabled shapes and model definition

MODEL DEFINITION:

Now as a final step we have to define our model, in order to do that we attach the last dynamic robot link to its corresponding joint so on with all the dynamic robot links to their respective joints until we reach the base of the robot.

Now we select the base and we enable Object is model base, thus potraying a bounding box to the robot .

Now as a final step to control the joints we open the joint dynamic properties then, we click Motor enabled and adjust the maximum torque. We then click Control loop enabled and select Position control (PID). We now run the simulation again: the mushua hand should hold its position.



(F)Model definition

This completes all the procedures for creating a hand from the imported mesh and thus is ready for simulation.