**Mass modelling SOP for Blender**

I presume for this SOP that you have basic familiarity with Blender – how to tumble and pan the view, object and edit mode, selecting things, working with different types of work area, etc. If not, go through the basics tutorial here: <https://cgcookie.com/course/blender-basics/>

**1: importing skeletal geometry**

*Format*

Blender works best with .obj files, which most CAD or CT segmentation software will output.

*Individual .obj files*

For single files, you can just use the built –in import function. Go to **File->Import->Wavefront(.obj)**, then use the browser to navigate to where you target obj is, select it and click the **Import OBJ** button. (NB, it is worth checking that the ‘forward’ and ‘up’ settings in the bottom left corner of the import window match the settings in the CAD or CT software you exported the OBJ files from).

*Multiple .obj files*

For multiple files, you can just repeat the procedure for individual files multiple times. If you are loading an entire skeleton as individual files however, this gets boring very quickly. Instead, you can use the following script to automatically load all .obj files in a given folder. First, open a **text editor** area by **clicking on the small dark grey box in either the top or bottom left of one of blenders windows**, which will open a pop-up menu with **editor type** written on it. Select **text editor**, and then create a new text block by going to **Text->Create New Text Block**. Then, copy and paste the text script text below in the editor window. Go to the 7th line, starting with **path\_to\_obj\_dir**, and input the filepath to your obj folder as instructed in the comment on the 6th line (nb comments are anything that starts with **#**).

Once you have path input, run the script by going to **Text->Run Script**. It might take a while to complete, depending on the complexity of your obj files, but once it has finished you should see all your obj files in the 3D view window.

Script below:

############################# SCRIPT STARTS HERE ############################

import os

import bpy

# SET THE PATH TO YOUR OBJ FOLDER HERE. THIS IS IN TWO PARTS. THE FIRST (e.g. ‘C:\\’ as below) SETS THE DRIVE LETTER. THE DOUBLE \\ IS IMPORTANT. THE SECOND PART SETS THE FOLDER PATH (e.g. 'Documents\\myObjFolder'') FOR YOUR OBJ FOLDER. NB USING DOUBLE \\ RATHER THAN SINGLE \ IS IMPORTANT.

path\_to\_obj\_dir = os.path.join('C:\\', 'Documents\\myObjFolder')

# get list of all files in directory

file\_list = sorted(os.listdir(path\_to\_obj\_dir))

# get a list of files ending in 'obj'

obj\_list = [item for item in file\_list if item.endswith('.obj')]

# loop through the strings in obj\_list and add the files to the scene

for item in obj\_list:

path\_to\_file = os.path.join(path\_to\_obj\_dir, item)

bpy.ops.import\_scene.obj(filepath = path\_to\_file)

############################# SCRIPT ENDS HERE ##############################

**2: Posing**

If you do not already have the skeleton in the pose in which you wish to model the flesh, you need to move and rotate the bones until it is in the pose you want. Firstly, depending on the scale of your imported skeleton, you might notice when you tumble the view (**middle mouse button**), bits of your skeleton disappear. This is because the view clipping plane is set too low. To change it, open the **properties pane** of the view area you are working in by going to **View->Properties** (bottom left of the area), go to the **clip** part of the extra pane that pops up, and make either the **start** value smaller or the **end** value larger, until you can see your entire skeleton.

To translate or rotate a bone or group of bones once you have selected it/them (see here if you don’t know how to select objects: <https://wiki.blender.org/index.php/Doc:2.4/Manual/Modeling/Objects/Selecting>), you use the **grab/move** and **rotate** manipulators. These buttons are found at the bottom of the viewport area, next to a little red-green-blue set of axes – the **arrow** is **grab/move**, the **arc** is **rotate**, and the **line with a square end** is **scale**. Selecting any of these tools will display a set of three color coded axes (or circles, for rotate) at the object or group of objects **centre** (an orange dot that appears when things are selected). This is often set at zero for scanned/CT geometry, which might not be a convenient place to do transforms around.

A better option is to use the **3D cursor** as the basis for transformations instead. The **3D cursor** is the white and red cross hairs that appears whenever you **left click** on a the 3D view. To use this as the basis for transformations, click on the button on the bottom of the viewport area that looks like two circles joined by a blue dot. This opens the **pivot point** menu, which lets you set the transformation center (aka pivot point) to several options, one of which is the **3D cursor**. Select that, and any translation, rotation or scaling will be done relative to the position of the **3D cursor**.

Manually clicking on points you wish to rotate around can be useful, but I find that using simple geometry to estimate rotation centres is more useful. Say that you want to pose the femur in your skeleton by rotating it around approximately where the hip joint should be.

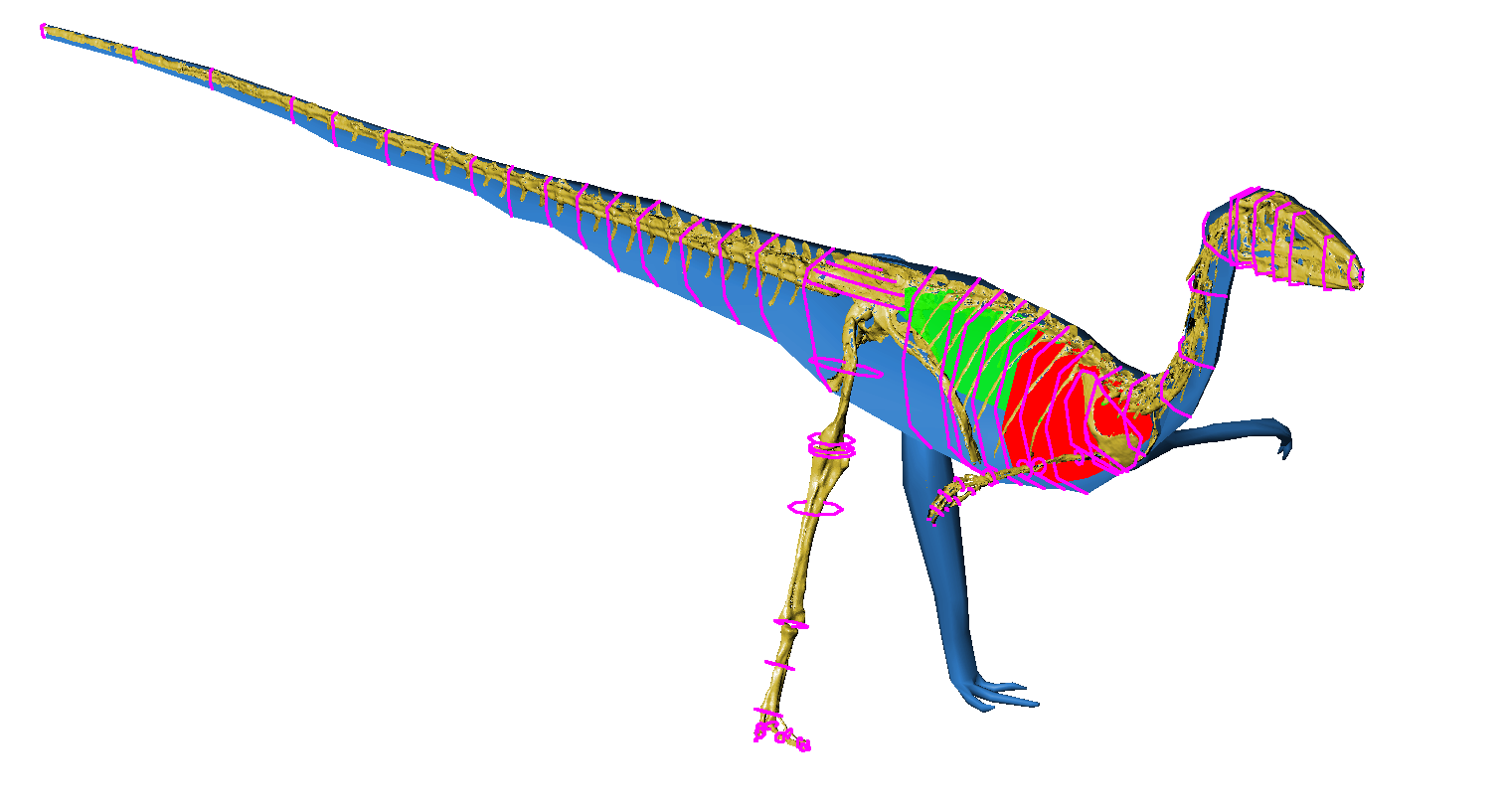
A simple way to estimate the rotation centre would be to **create a sphere** using the **add->mesh** menu at the bottom of the viewport. The sphere will be created at wherever the 3D cursor currently is, but it might be very small when you first create it, which might make it hard to see. You can use the **keypad ‘.’** key to zoom in to whatever you have selected, which should be the sphere you created (NB blender is quite shortcut focused, and needs a full keyboard. This might make it hard to use with smaller laptops). Now, you can move and scale the sphere to fit the femoral head using the manipulators (NB, don’t forget to change the pivot point back to the object centre (‘**active element’**) using the pivot point menu). You might find the shortcuts **g** (for translate), **r** (for rotate) and **s** (for scale) useful. NB, you might also find that your sphere will not scale – there is a button **next to the pivot point menu** that looks like a double ended arrow with three dots above it. If this is on (dark), you cannot scale anything.

Once you have your sphere (or whatever geometry you are using to estimate the rotation centre – cylinder, torus etc.) positioned, you can then move the 3D cursor to the centre of it using the **snap** menu, which you bring up by pushing **shift + s**. To move the 3D cursor to your sphere, make sure you have the sphere selected, bring up the snap menu, and click on **cursor to selected**. Now, when you select everything you want to pose (say, the whole limb), if you set the **pivot menu** to the 3D cursor again, any rotations or scaling will be about the centre of the sphere.   
  
NB – a note on rotations. By default, when you push the **r** key to rotate (rather than using the rotate manipulator), it will rotate about an axis parallel to the current view. To free rotate, push **r** twice in quick succession.

*Use of a reference pose*

For our work on theropods, we posed all of our subjects as follows – tail and back straight, hindlimbs straight down and then rotated laterally by 15 degrees (NB you can type in quantified rotations, using syntax like ‘**r x 15**’, which would rotate by 15 degrees in global x. **r x x 15** would do the same using the local axes), forelimbs pointing straight laterally from the glenoid, and the neck arranged into an s-curve so that its length along the cranio-caudal axis was half that it would be if fully stretched out (see figure below). We also move the entire skeleton so that the **global zero (0,0,0)** lies exactly between the two hip joints.

This pose is entirely arbitrary, and the idea behind it was to remove confounding factors when comparing between taxa. You may want to use a different pose, or a more ‘natural pose’, but think about the tradeoffs of those choices for what you do with your models.



*Zero position*

Finally, before we get to building the model, it is very important to have some idea of where the global zero (0,0,0) is relative to some landmark on your skeleton, otherwise the mass properties you estimate will be impossible to interpret. We usually move our posed skeleton to that global zero is exactly between the two spheres we have fitted to our hip joints. This point can be identified by first selecting both spheres, then using snap-to-selection to move the 3D cursor (it will move the cursor to the combined centre of the two spheres).

To move your skeleton so that this point is at zero, you will first need to create an object at this point (add a sphere or cube or whatever then snap it to the position of the 3D cursor), and you will then need to **parent** everything to your zero object. Do this by selecting everything (use **a** to select/unselect everything), then using **shift + right click** to select your zero object again (you probably need to do this twice as the first click will deselect it). You can then use **ctrl + p** to parent all selected objects to the last selected object. This means that all transformations done on the zero object (like moving it to global zero) will also happen to anything parented to it. You can now move the 3D cursor to global zero using the ‘**cursor to center**’ option in the snap menu. To zero your whole skeleton, select your zero object and snap it to the 3D cursor. You can then remove the parent relationship with the zero object by selecting everything and doing an ‘unparent’ with **alt + p**.

**3: Creating mass hull**

***General method***

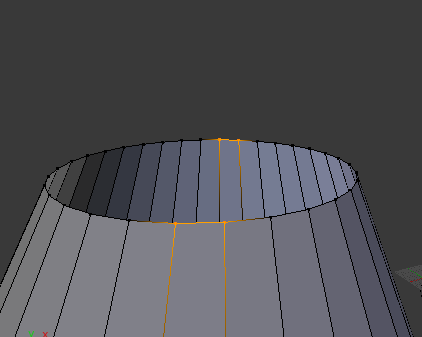
*Lofting*

We generally build models by fitting cross-section outlines to the skeleton and where we think the flesh should go, and joining them together to make a seamless hull using a tool called ‘lofting’. This is found in a Blender add-on called ‘LoopTools’ that has to be activated before it can be used. To do this, go to **File->User Preferences**, and then to the **Add-ons** tab, and type ‘LoopTools’ into the search box. It should pop up. To load the add-on, just check the box next to its name, then close the user preferences window.

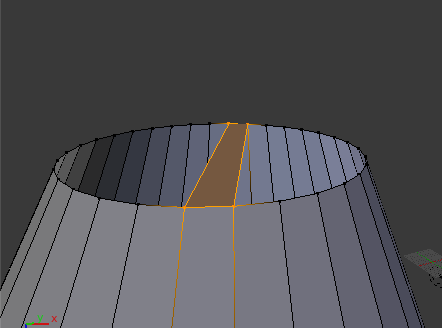
To see what it does, first create a couple of circles using the **Add->Mesh->Circle** command. Make a couple and move them so they’re stacked one above the other. Now, **join** the circles using **ctrl+j** so that they are part of a single mesh. Select the new mesh and go into **edit mode** (**tab key**), and make sure that all vertices on your new mesh are selected (they should all be orange). To loft the mesh, open the ‘tool shelf’ subwindow (if it’s not open already) by going to **View->Tool Shelf**. In the ‘tools’ part of the Tool Shelf, scroll down to the bottom and you should see a menu for ‘**loop tools**’ at the bottom. Expand that, and there should be a button for ‘**loft**’. Push that and a tube will automatically be created from your hoops!

*Manually creating polygons*

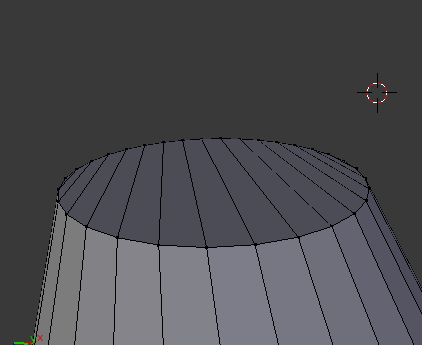
To make the end of your tube solid, you will have to manually create polygons. To do this, first unselect all vertices using **a**. Then, select three or four vertices from opposite sides of one of the ends of your tube, like so:



Then, simply press **f** to make a polygon between these vertices, like so:



And keep going, selecting vertices and making new faces, until you have created a complete cap:



This is basically the method we have used to model extinct animals. The circles are fitted to the posed skeleton and/or where we think the flesh was, they are then lofted to a create a hull, and finally have any holes on the ends or complicated bits filled in by manually creating polygons.

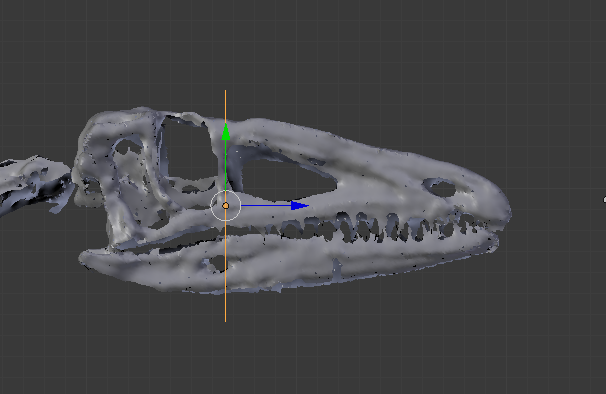
***Modelling a theropod***

*Head*

Assumptions: Flesh tightly approximates the skull. Buccal (mouth) cavity assumed to fill the space between mandibles and the roof of the mouth. Nasal cavities and the space between the orbits and the nasal openings are left empty, too (so the head is lower density).

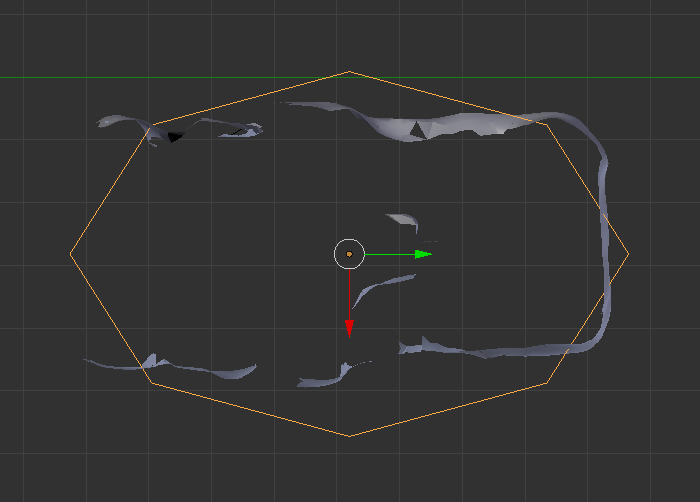
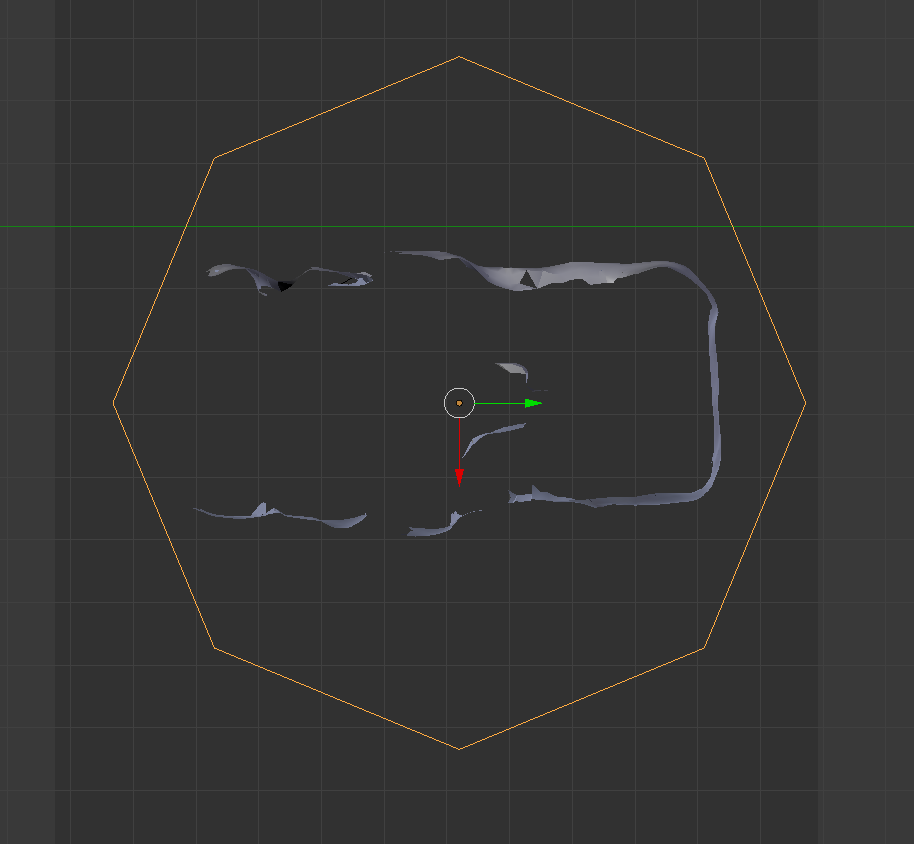
The head is the easy bit. We’re going to be working in cross sections, so we will need to switch between lateral and frontal views (how to set viewport views is detailed here: <https://wiki.blender.org/index.php/Doc:2.4/Manual/3D_interaction/Navigating>). First of all, lets fit a hoop around somewhere in the middle of the skull. To do this, make sure you’re in lateral view, then first **left click** somewhere in the view port to move the 3D cursor somewhere visible. We are going to use the **create** tool shelf to make a circle, rather than just **add** this time, because we want to change how many vertices our circle is made of (more on that later).

Making sure the **tool shelf** is visible (**view->tool shelf**), go to the **create** tab (second one down, and click on **circle** in the **mesh** area of the tab. This should a) create a circle at the location of the 3D cursor, and b) open a submenu at the bottom of the create tab called **add circle**. One of the options in **add circle** is the number of vertices the circle has. Change this to 8 to make the circle and octagon (we’ll be working a lot with octagons, I’ll explain why shortly).   
  
Now that you have made your circle into an octagon, it’s time to fit it to the skull. To do this, we first want to rotate it by 90 about whatever the dorsal axis for your skeleton is, so that the octagon lies in the frontal plane (if global y is dorsal, then the command would be **r y 90**). Now, move your octagon so that it’s centre (the orange dot) is in the middle of the skull somewhere, and then scale it so that it’s about as wide as the skull is tall, like so:

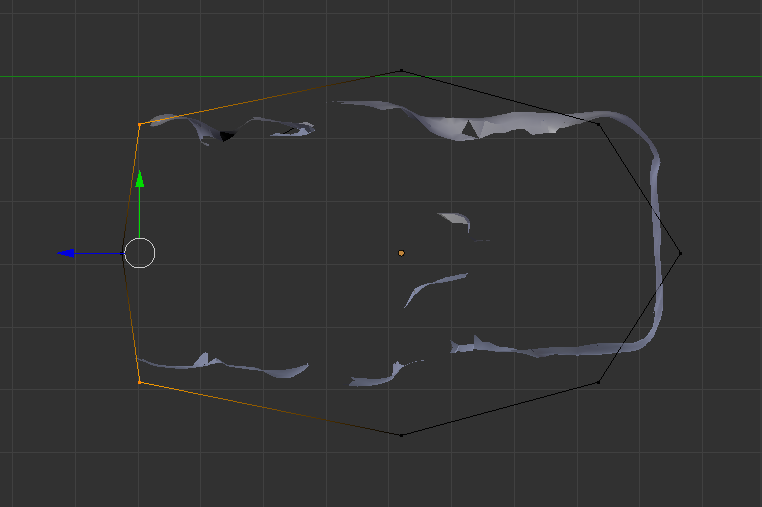


Now, to fit this hoop to the skull, we are going to want to look at it from the frontal view. But if we do that, there is often too much other stuff visible to see if we are fitting the hoop correctly. To **only view a cross section of the skull**, we are going to use a viewport clipping box – in the lateral view, press **alt+b**, and drag a box around the thin section of the skull where you have placed the hoop. Everything except that thing section of the view should disappear, and when you switch to the frontal view, you should only see that section of the skull displayed:

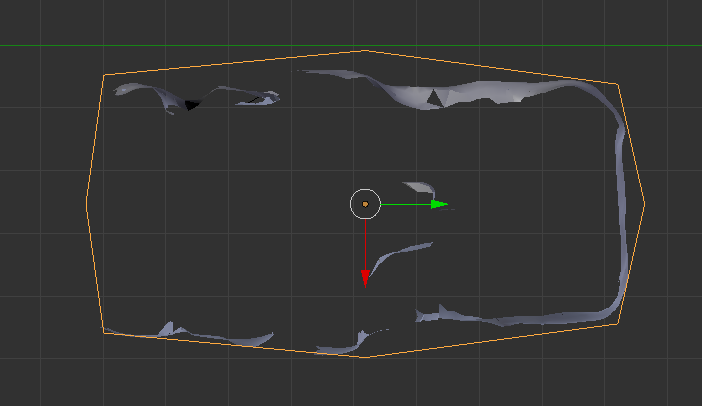
You can use the move and scale tool to now shrink the octagon until it’s about the right size (remember you can scale in both local and global axes).



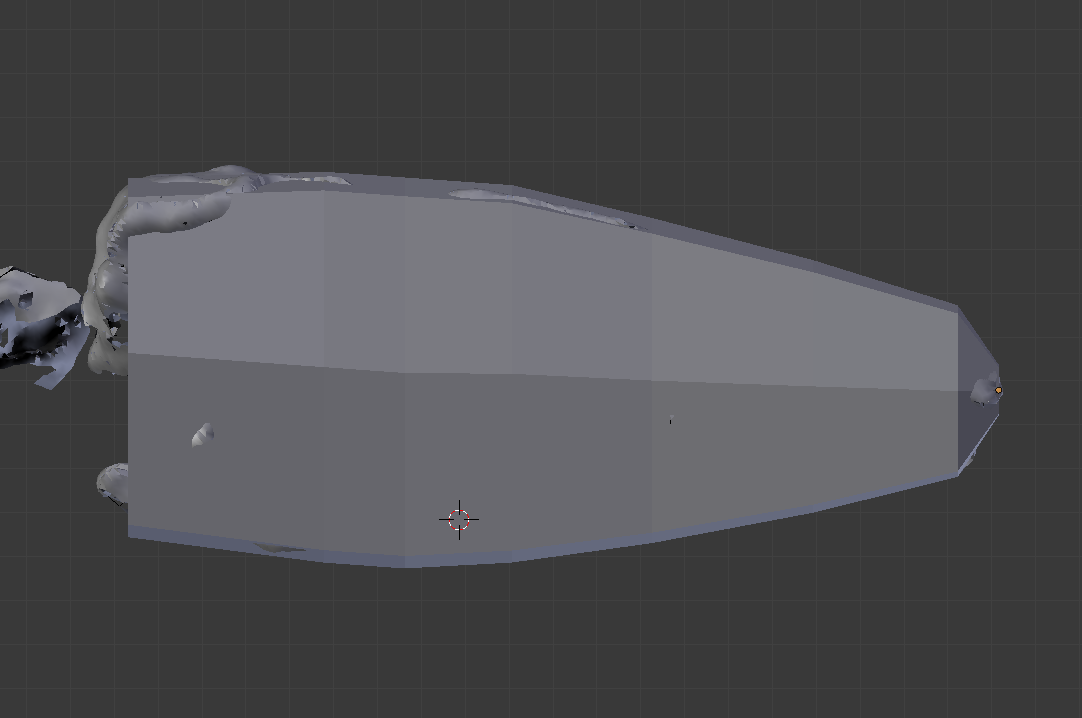
It’s still not a great fit though! To do better, we will need to go into **edit mode** and move some vertices around manually. When doing this, it’s best to work in **bilateral pairs** of vertices, in order to keep everything as symmetrical as possible, like so:



Using the scale and move tools on pairs of vertices, you should be able to get your outline hoop to fit the skull pretty well:



Now, repeat this process for multiple different cross sections of the skull, either making a new hoop each time or simply copying the hoop you have just fitted. Then once you are happy with it, loft the hoops into a single hull, and seal the ends using manually created polygons. To aid with selecting hoops, it might be useful to have the skeleton and the hoops in separate **layers**. To move something to a different layer, select it and then push **m**, thenselect a layer from the row of grey boxes on the popup. To display different layers, click on one of the grey boxes shown in two rows at the bottom of the viewport (boxes with a dot in represent layers contain something, empty boxes represent empy layers). To view multiple layers, **shift + left click** multiple boxes. You can also use hide (**h**) and inverse hide (**shift+h**, hides everything NOT selected) to hide/show different objects to make the view easier to work with. To unhide everything use **alt + h**.



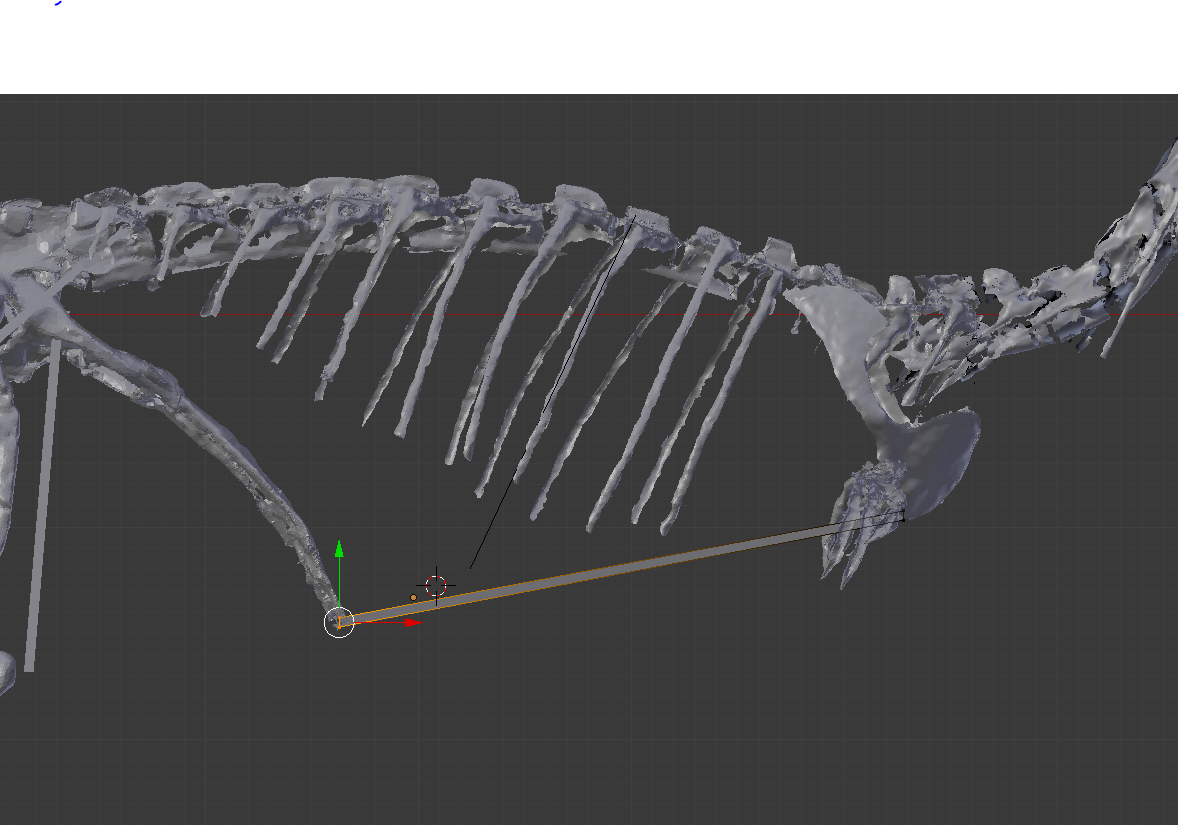
For the airspaces, repeat this procedure (make and fit hoops, loft, seal the ends) for the inside of the mouth and the inside of the nasal cavity.

*Chest*

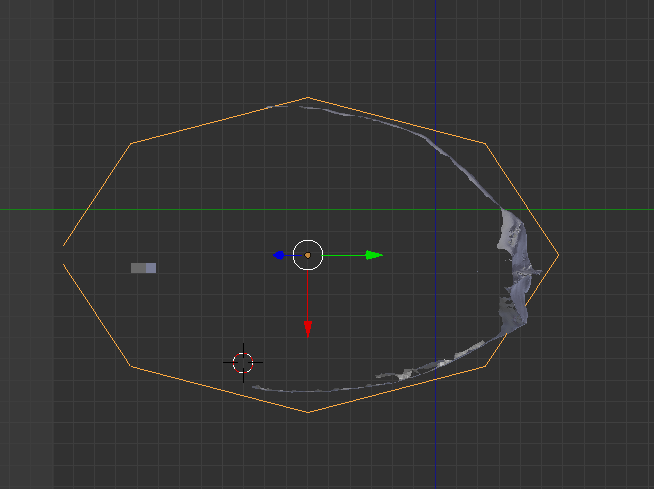
Assumptions: Chest cannot be smaller than the ribcage. Lungs fill a set proportion of the dorsal chest cavity (again, to reduce final density).

Skipping the neck (we’ll come back to it later, for reasons that should make sense eventually), the next section is the chest. This follows the same basic procedure as the head (make and fit hoops, loft, seal the ends). We’ll model the thorax (chest) first.

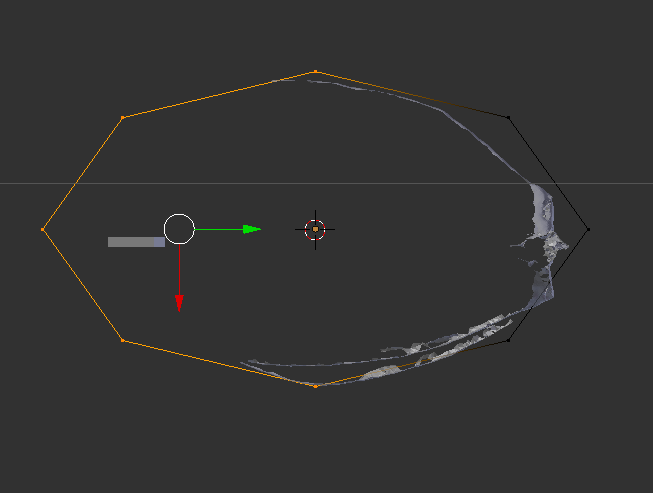
First of all, if you do not have any gastralia preserved, how do you know how deep the chest should be below the ribs? Short answer, you don’t. I have used a **line drawn from the caudoventral tip of the coracoids to the cranioventral tip of the distal pubes (along the midline)** as a rough estimate, but if you can think of anything better, go ahead! To create the line in Blender, create a small cube, go into edit mode, and then grab the vertices of one side and stretch out the cube into a long rod:



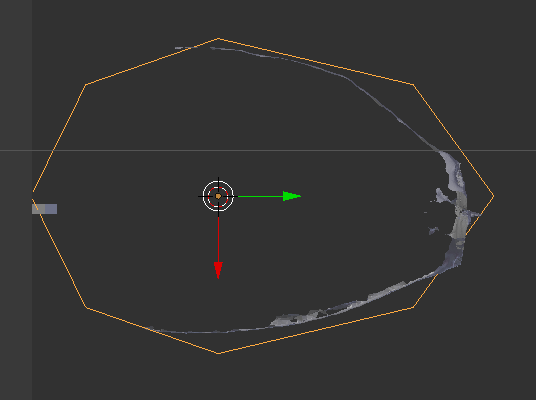
This will give us some idea of how deep our chest volume should be. To start with, create hoops and fit them to the ribs. Use **alt+b**to section out a single rib (this will be easier if you rotate the view so that the ribs are on the vertical axis, the 8,2,4 and 6 keys on the numpad will do this), then switch to a cross-section view (this may also need rotating to be perpendicular to the rib and outline hoop). You should see both the ribs and the depth-gauge line:



The chest is typical more of a squashed oval outline than a symmetrical oval or circle. This means that we have to scale the top and bottom parts of the hoop differently, **but**, we also want both parts to remain as geometric half ellipses (this is a big part of the max/min model iterating process). To do this, we will have to make use of the 3d cursor to control our scaling centre. First, fit the top half of the hoop to the ribs, as show in the image above. Then, use the snap menu (**shift + s**) to move the 3D cursor to the centre of the hoop. Then, in edit mode select only the verts of the bottom half of the ellipse as shown:



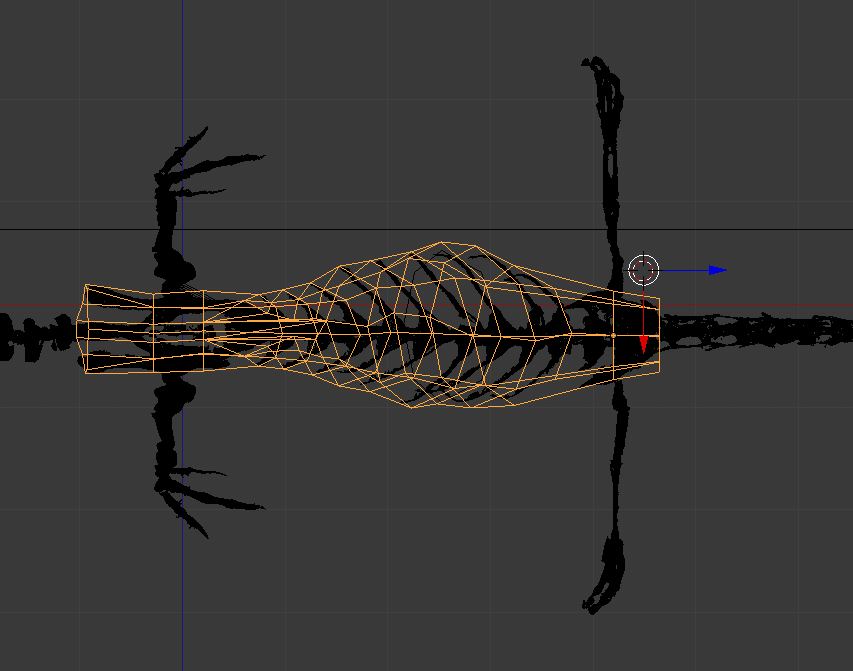
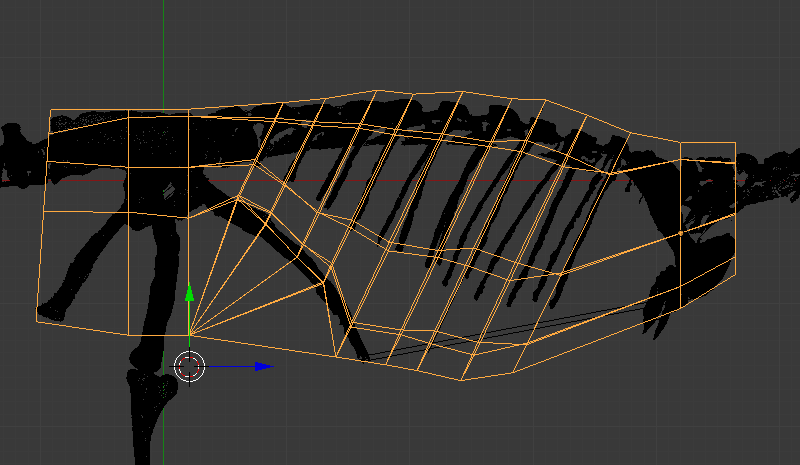
If we scaled the bottom half using the transform centre as show, it would distort the hoop so that it wasn’t two half-ellipses any more. To stop that, **set the pivot point to the 3D cursor**, and scale from there. We want the bottom half-ellipse to be shorter, so just scale in the vertical axis (Y, in the case above). You should get a teardrop shape like this, with the tip just over the depth gauge line:



Repeat this for all the ribs, using the 3D cursor to maintain the pivot point at the centre of the hoop, fitting the top half ellipse to the curve of the ribcage and scaling the bottom half ellipse to the depth-gauge line (or gastralia, if they are present).

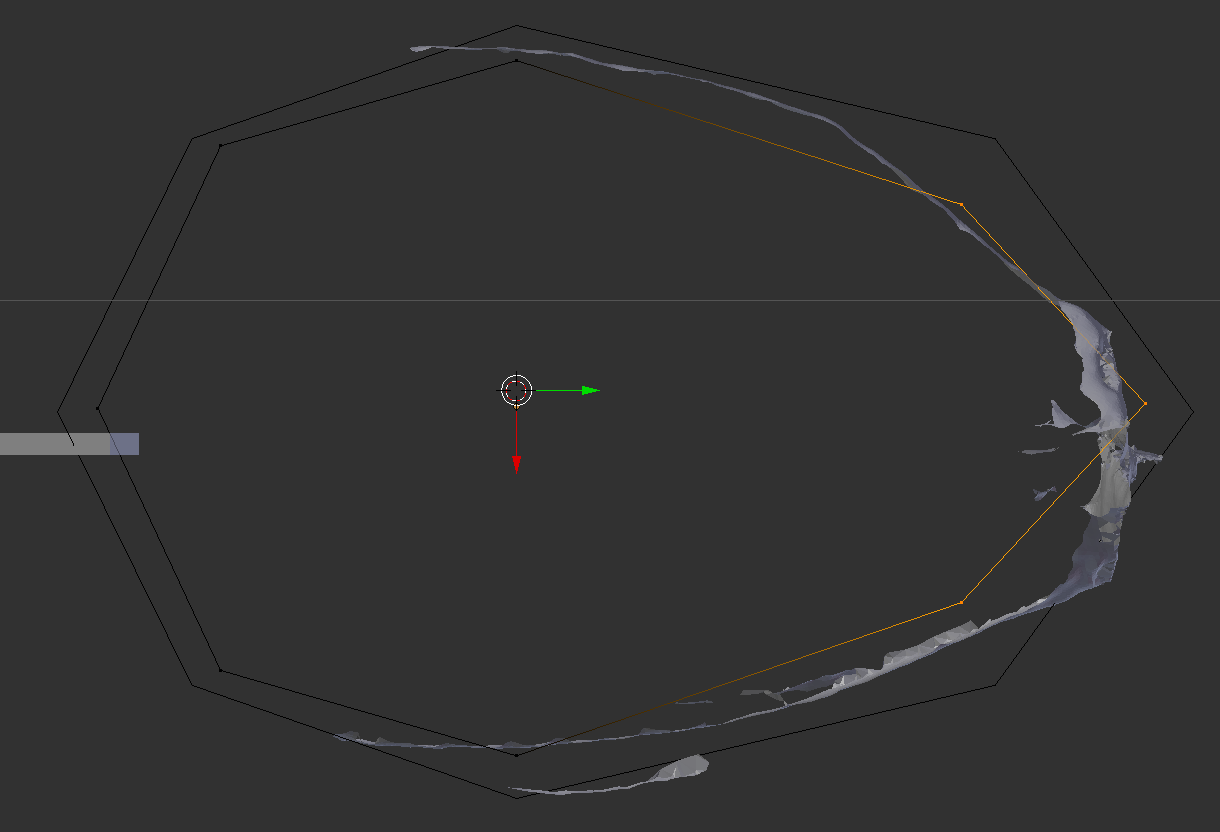
When you get to the pelvis, the pubis forms the ventral border of the chest cavity, so fit the bottom half of the ellipse to the pubis. For the pelvis itself, create a similar line between the ventral tips of the pubis and ischia as a guide, and fit the hoops tightly to the bone.

We consider the chest section to start with pectoral girdle and end at the caudal end of the pelvis. As the ischia usually protrude caudally beyond the ilium, this means the caudal end of the pelvic segment is often at least slightly slanted, as show below (nb this is taken in **wireframe mode** to show the underlying skeleton. You switch between wireframe and normal mode using the **z** key).

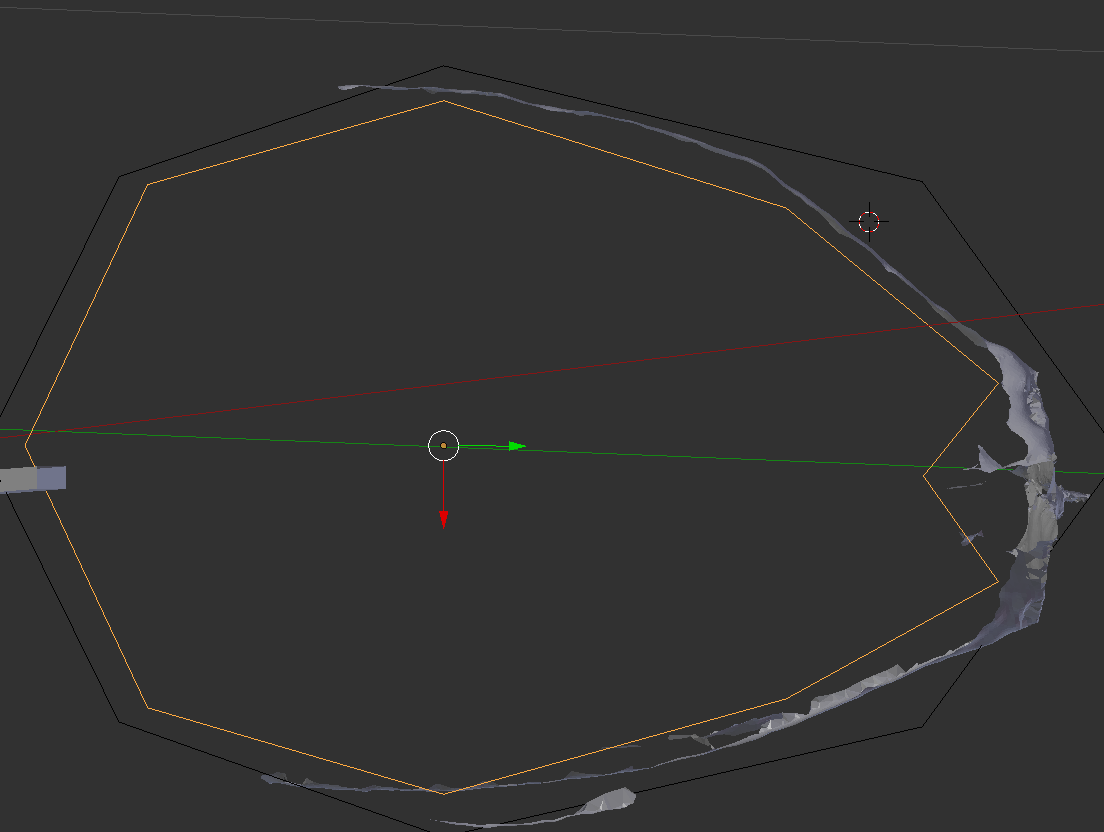


*Lungs*

The procedure for modelling the lungs is similar to modelling the chest, except that the hoops are fitted to the inside of the ribs and vertebrae rather than that outside. For the lungs it’s not important to maintain an elliptical shape, so you can deform the hoop as you see fit. To bend the hoop around the underside of the vertebrae, you will probably need to **add vertices to the hoop**. This is done via the **subdivide** menu (accessed with the **w** key). Select the top three vertices of your hoop, near the vertebral column (the verts in the example below are super scrappy, sorry).



Then hit **w** and select **subdivide**. This will add two more vertices, half way along the edges connecting the three vertices you have selected. You can then use these extra vertices to fit the hoop around the ventral surface of the vertebral column, as shown below (this is a fairly crude version done for illustrating this SOP, you might want to add more verts to get more detail:

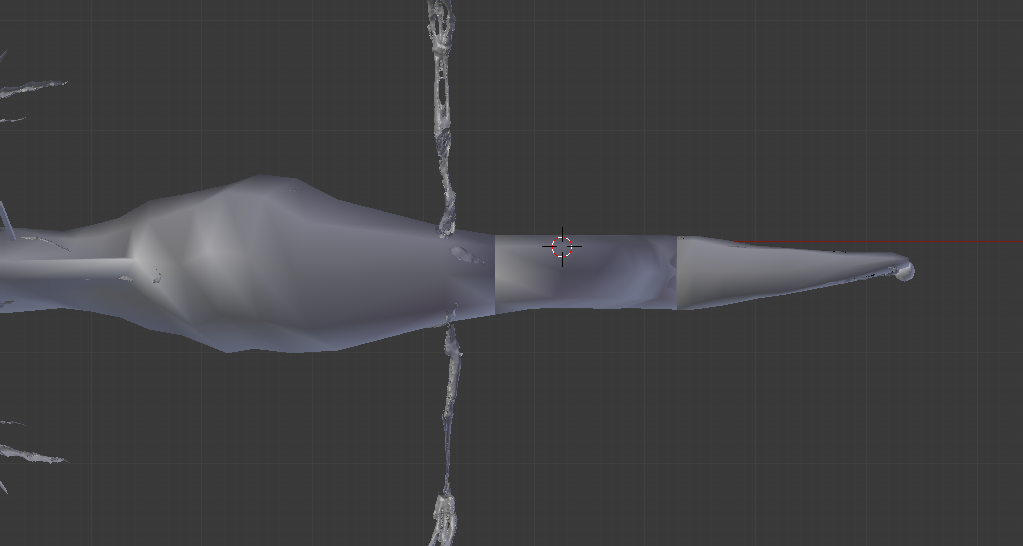
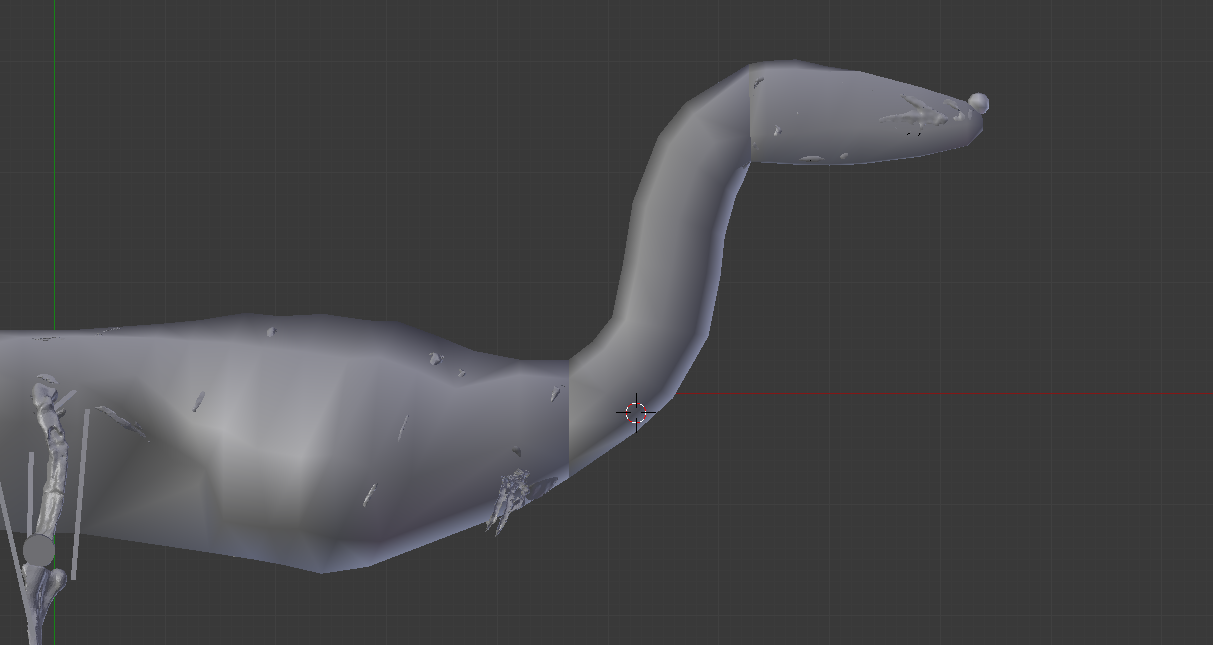


Doing that for all ribs, lofting and patching the ends will give you a model of the inside of the chest cavity. But the lungs do not fill the entire chest cavity! So, we will eventually need to subdivide the cavity to produce lungs, but we’ll leave that bit for the section on **mass/min model iteration**.

*Neck*

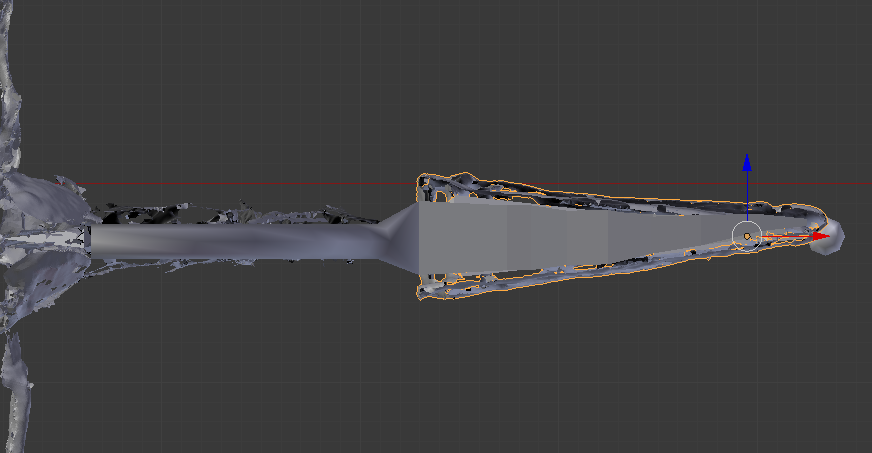
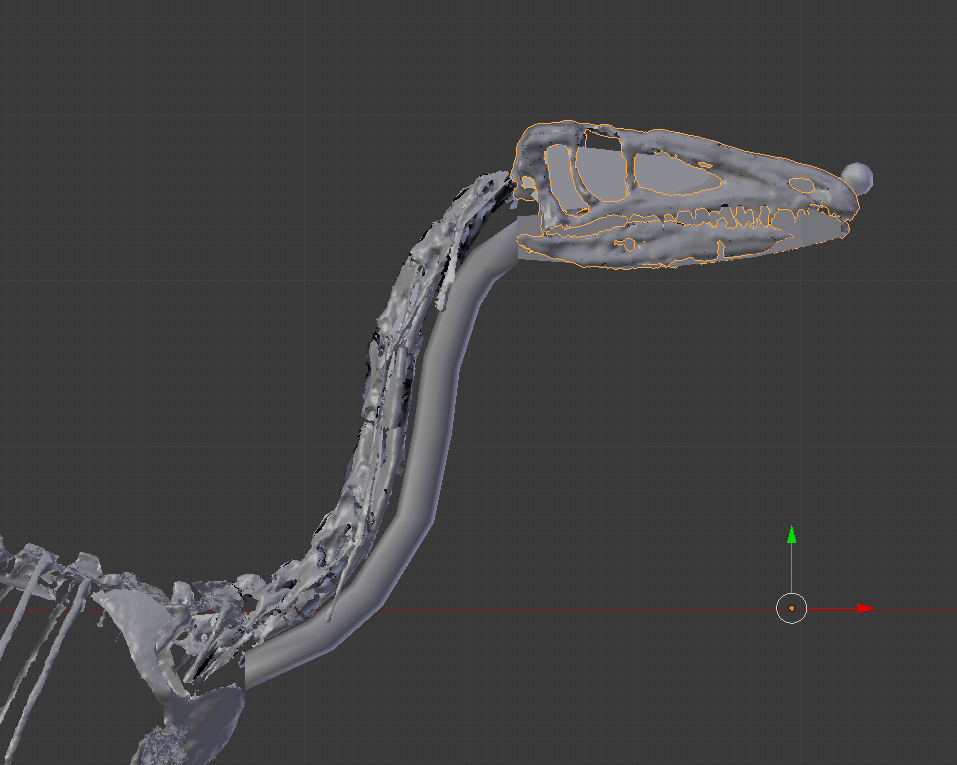
Assumptions: Necks are approximately as wide and deep as the back of the skull.

We model the neck based on the observation that necks in many things are roughly the width of the back of the skull (this is only sort-of true, but it’s a reasonable place to start – we’ll come back to this in the section on model iteration). We model the neck as a series of circular outline octagonal hoops that are roughly the same size as the back of the skull, arranged so that the midline of the hoops contacts the tips of the transverse processes on the cervical vertebrae. To a good fit, the first section and last sections are a copy of the caudal end of the skull and cranial end of the chest volumes. Results should look like this:



*Pharyngeal cavity*

As with the neck, there aren’t really any good skeletal landmarks to base the size of the pharyngeal cavity on. We have used a circular hoop fitted to the height of the buccal cavity as a guide, with more hoops of the same dimensions aligned to each cervical vertebra in a similar position all the way down the neck. While this is probably inaccurate, it’s hard to imagine how to *be*accurate in this case, so we rely on model iteration to hedge our bets (see model iteration section below). Here is an example of one of our pharyngeal cavities:

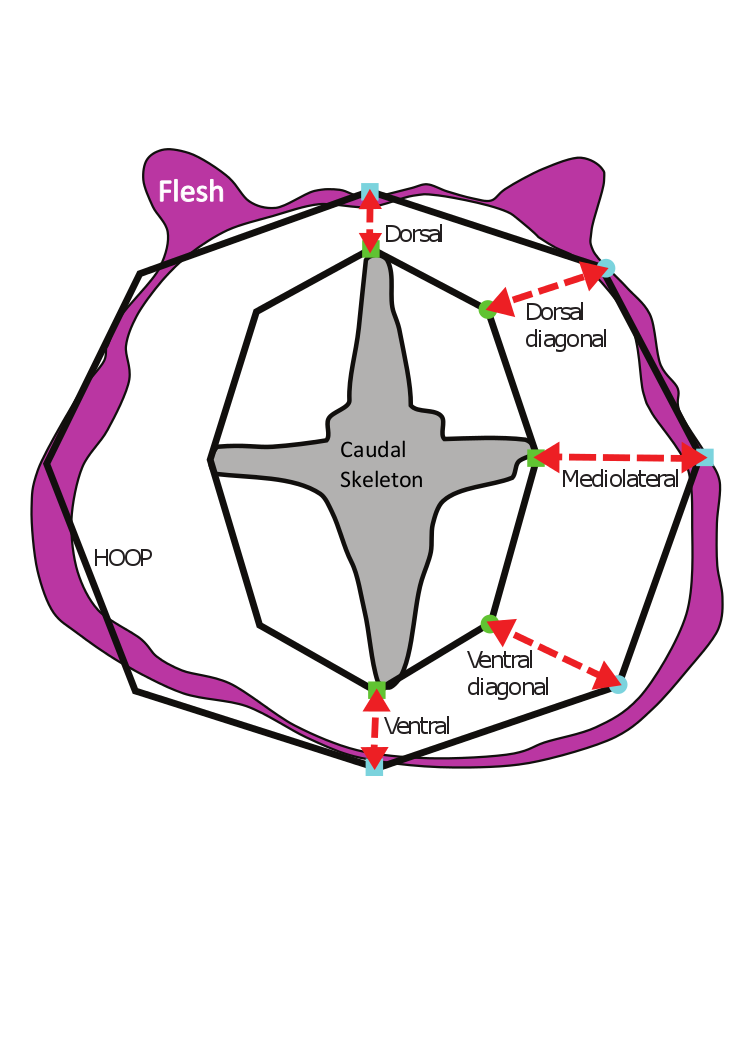


*Tail*

Assumptions: Saurian tails have a consistent relationship between the dimensions of an ellipse fitted to the ‘cardinal points’ of a caudal vertebrae (neural spine, transverse processes, chevron) and the dimensions of an ellipse fitted to the flesh.

The tail has similar problems to the neck, in that without ribs or equivalent structures it is hard to precisely reconstruct the fleshy volume surrounding the bones. To try and get around this problem, we CT scanned and took cross sections of lizard and crocodile tails to quantify the ‘extra-skeletal’ (i.e. fleshy) dimensions of saurian tails (Allen et al., 2009, Anatomical Record). We found the scaling values that transform a hoop fitted tightly to the skeleton into one fitted to the flesh to be reasonably consistent, and so we use these values to reconstruct extinct animals.

For each caudal vertebra, fit an octagonal hoop to the tips of the caudal skeleton as shown below (neural process, transverse processes, chevron bone where present or ventral tip of centra where not). You will need to scale the top and bottom half of the hoop separately, as we did for the chest section. We will then scale the outline hoop in five different ways to fit what we determined to be an ‘average’ saurian tail profile.



Firstly, we scale the dorsal half of the hoop dorsoventrally. All scaling needs to be done from the 3D cursor, positioned at the centre of the hoop (the middle of the two mediolateral vertices [see above]).

In edit mode, select the top five vertices of the hoop (the two mediolateral and two dorsal diagonal vertices, and the single dorsal vertex). In the **dorsoventral axis only**, scale these verts by **1.33**.

Next, select the bottom five vertices (two mediolateral, two ventral diagonal, and single ventral) and scale in the dorsoventral axis only by **1.86**.   
  
Next, select **all** vertices and scale in the **mediolateral axis only** by **1.58**.

Next, select the two **dorsal diagonal** vertices and scale in all axes (free scale) by **0.91.**

Finally, select the two **ventral diagonal** vertices and scale in all axes by **1.12**.   
  
Repeat this for every tail vertebra and you will have a complete tail! For the cranial face of the tail, just copy the caudal end of the chest section.

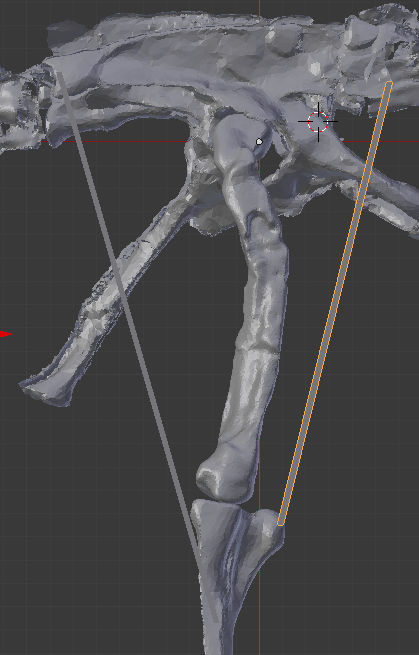
*Legs (pelvic limbs)*

Assumptions: Upper part of thigh (dorsal to acetabulum) can be approximated with a hemisphere. Cross-sectional profile of limb is elliptical. Extra-skeletal extent of limb can be roughly constrained by the paths of major muscles.

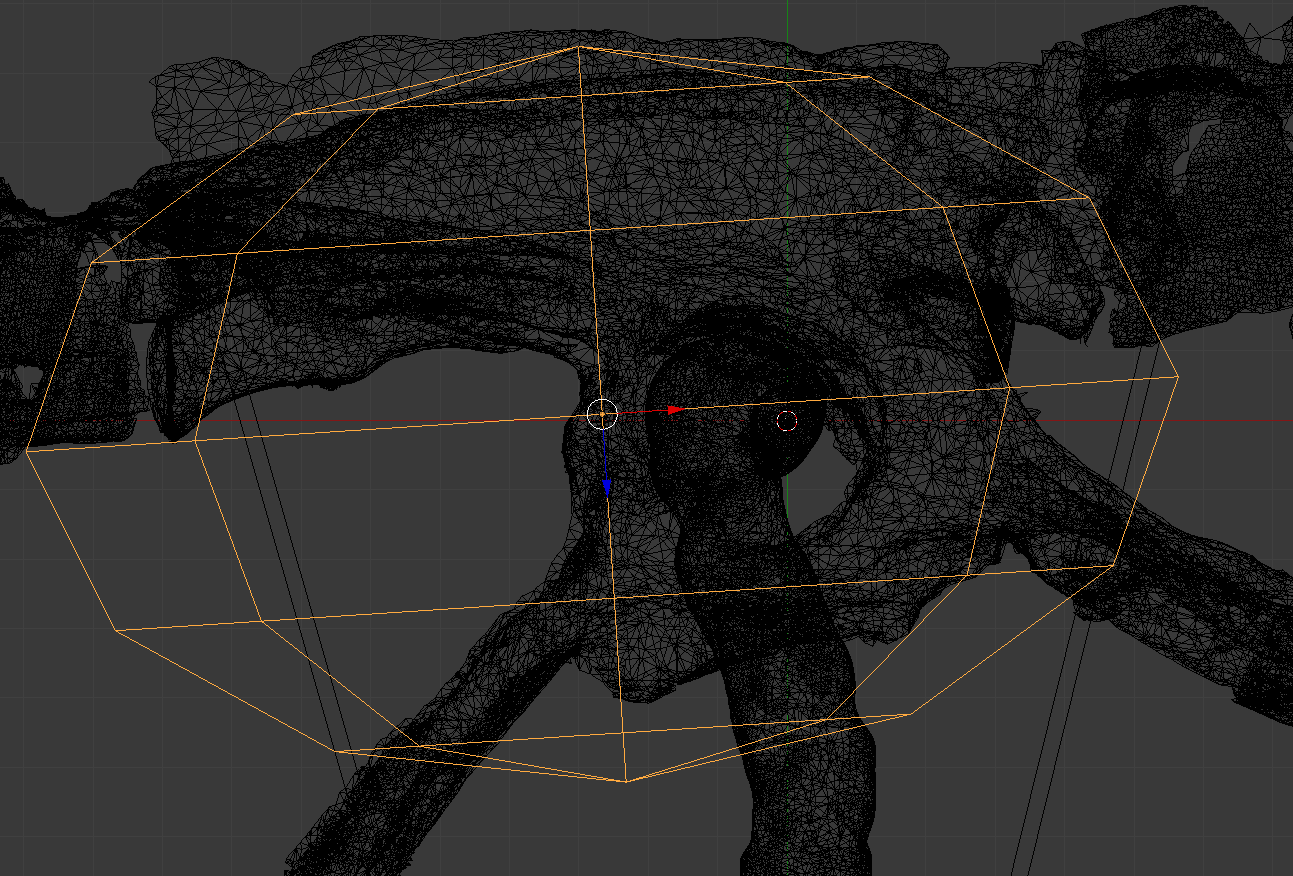
First of all, we will add some very crude approximations of major limb muscles to our limb skeleton to act as a guide. Aswe did with the depth-gauge line for the chest, create a cube and stretch it into a long thin oblong. The first ‘muscle’ we will make will approximate the caudal boundary of the ‘hamstring’ muscles, particularly the iliofibularis. Move the top of the ‘muscle’ oblong to the caudal tip of the ilium, and the other end to the iliofibularis tubercle on the proximal fibula:



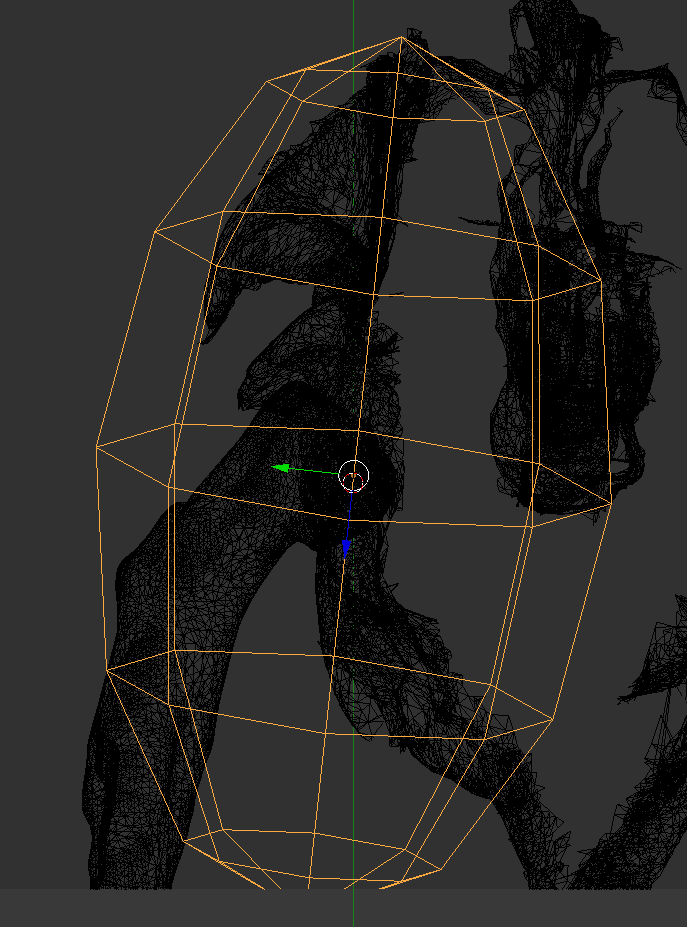
Next, we will make a ‘muscle’ approximating the cranial border of the iliotibialis, running from the cranial tip of the ilium to the cranial tip of the cnemial crest on the tibia:



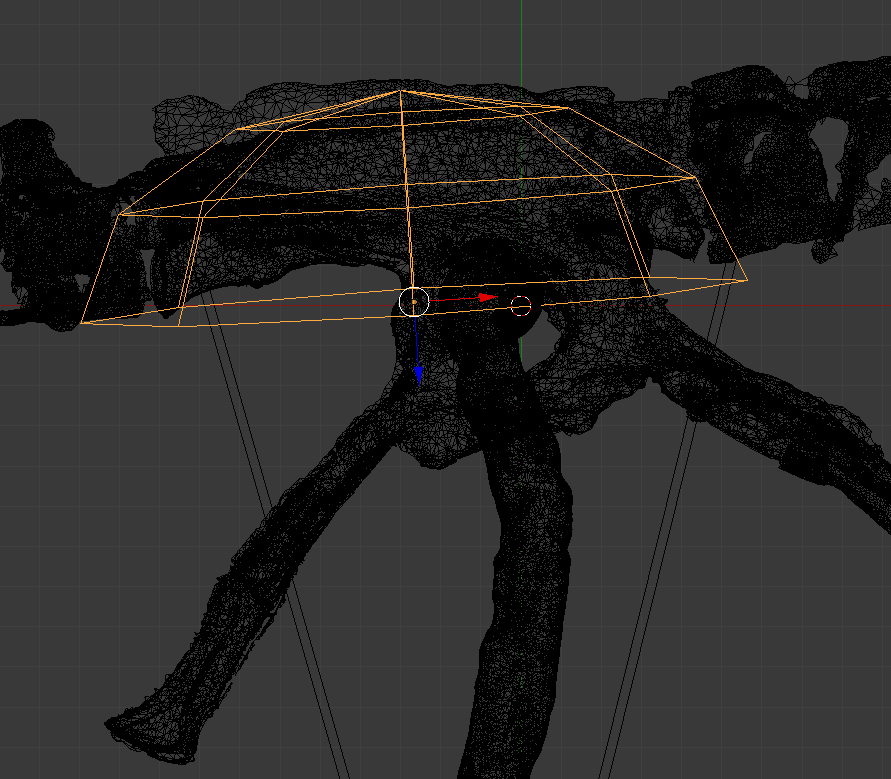
Now, we will approximate the dorsal-most part of the thigh with a hemisphere. First, **add a UV sphere** using the ‘create’ submenu in the toolbox. Set the sphere parameters (which appear at the bottom of the toolbox, same as for adding a circle) to have **8 segments and 6 rings**. Use snap to move the sphere to the centre of the hip joint (i.e. the centre of the sphere you fitted around the femoral head in the ‘posing’ section). Now, rotate, scale and move the sphere so that the top of it roughly fits the profile of the ilium in lateral view:



Next, in cranial view, rotate and scale the sphere so that its apex fits along the iliac blade, and the lateral extent is just larger than the lateral margin of the femur (the **alt+b** cutaway view might be useful here):

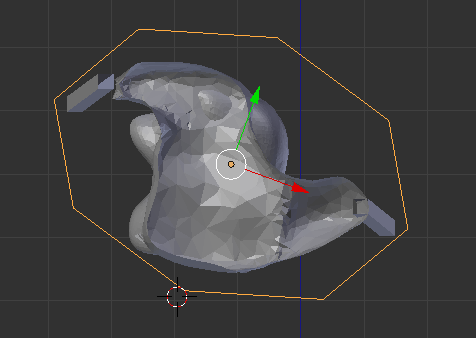


Then go into edit mode and delete the lower half of the sphere to make a hemisphere:

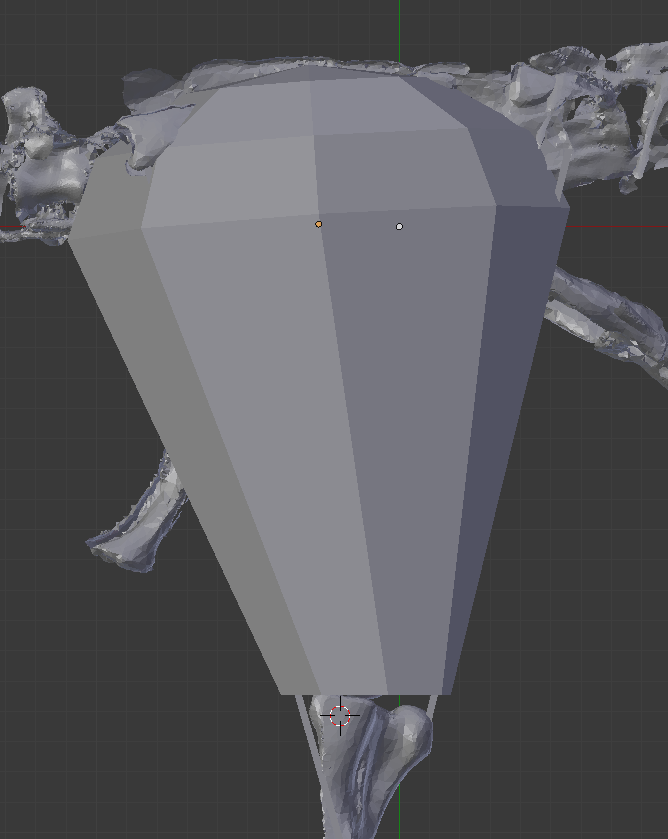


Next, fit an octagonal hoop to the knee, placed at the distal tip of the femur and using the cnemial crest and iliotibialis ‘muscle’ as a guide:

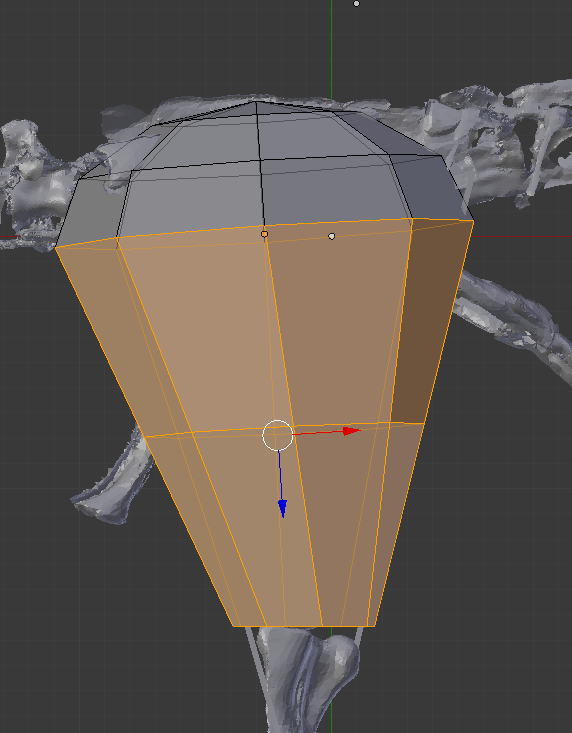




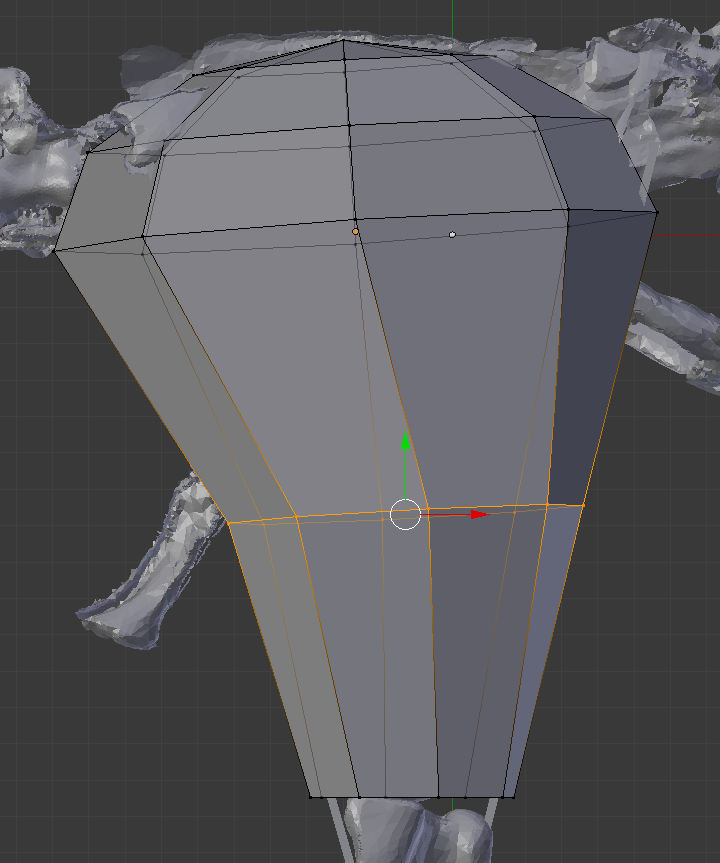
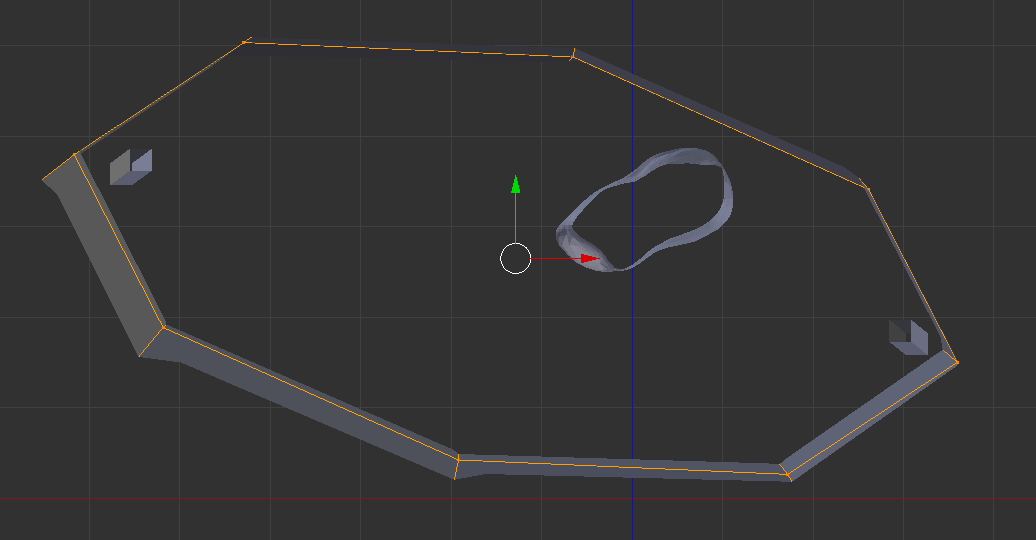
To complete the thigh section, joint the knee hoop to the hemisphere, then manually create polygons to fill the gap between them:



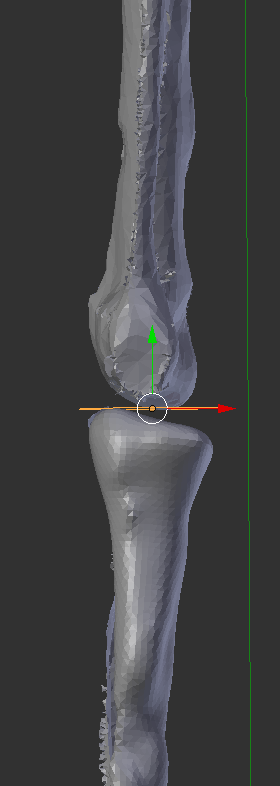
Next, use **subdivide** to split the polygons linking the hemisphere to knee hoop in two, proximo-distally. To do this you will need to select **edges** rather than vertices (which is what blender defaults to). To switch to **edge select** mode, go into **edit mode**, and press the button that looks like a small grey cube with one yellow edge (bottom right-centre of viewport window). Select the edges of the linking polygons and subdivide (**w**).

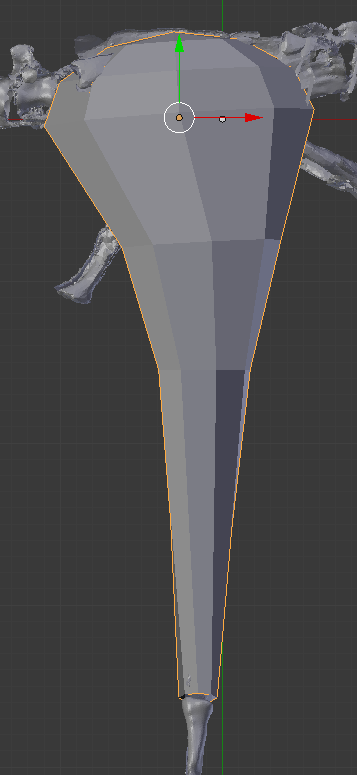
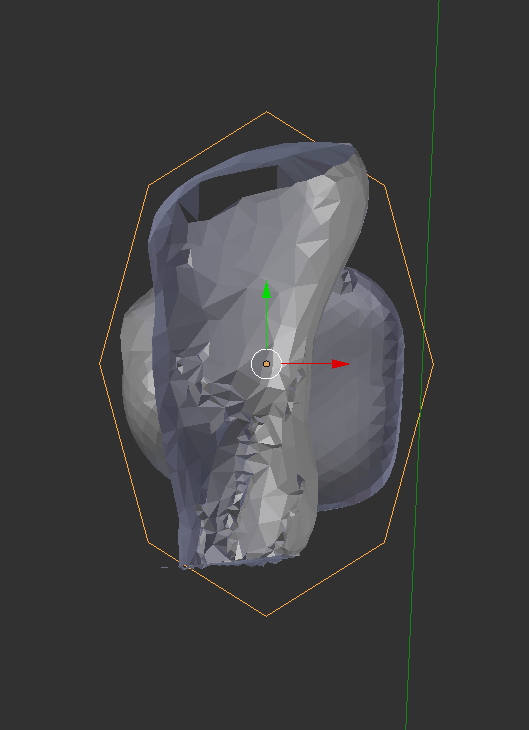


This essentially creates a new outline hoop halfway down the thigh. Use an **alt+b** cross-section to check that the new hoop covers the ‘muscle’ guides. Rotate, move and scale so that it only just covers them (this is a minimal thigh):

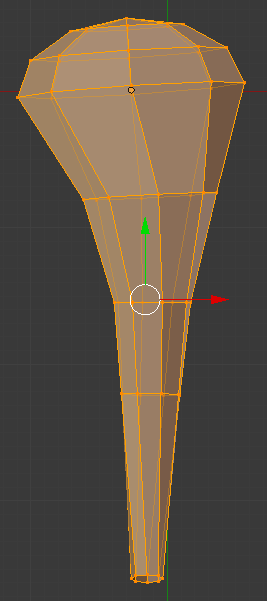


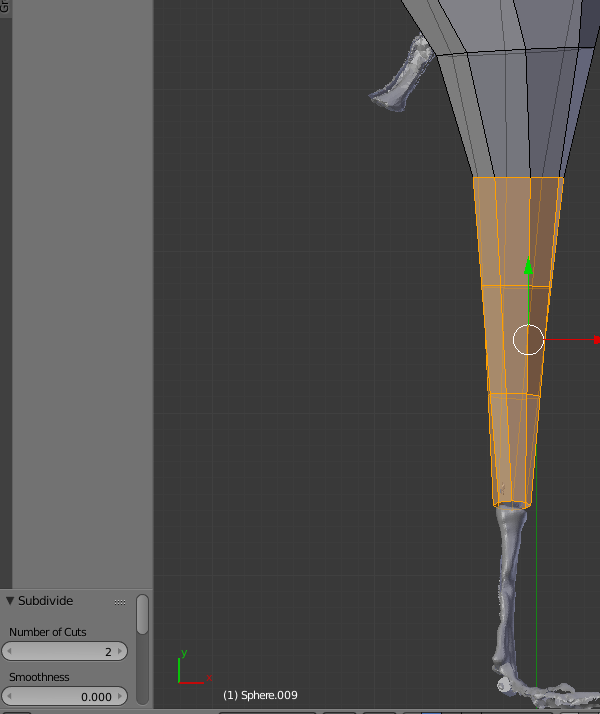
Next, add a hoop fitted to the ankle, positioned at the distalmost extent of the tibiotarsus, and create polygons to link it to the knee hoop:



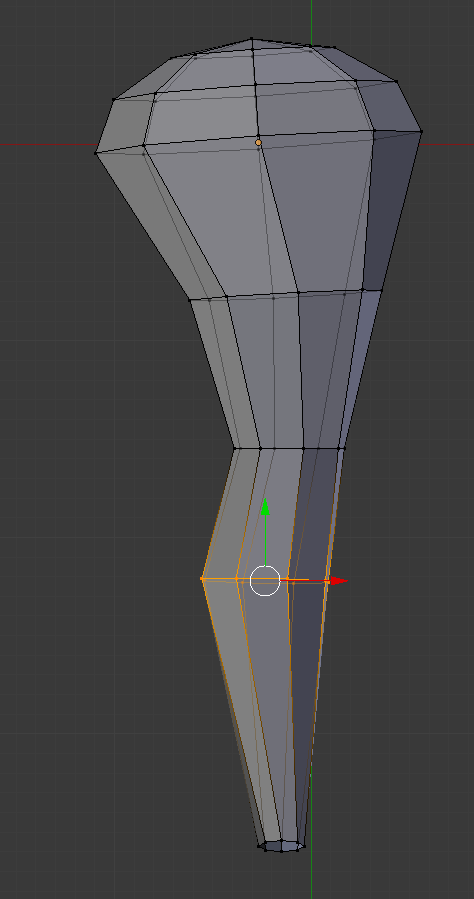


This results in a calf section that is probably too skinny, so we will add a hoop 1/3rd of the way from the knee to the ankle to represent the bulge of the gastrocnemius muscles. The easiest way to do this is to **subdivide** the edges linking the knee to the ankle **with two cuts**. To do this, go into **edit mode** and **edge select mode** (see above), select the edges linking the knee to ankle hoop, and then go into the **subdivide** menu (**w**), then hit **subdivide**. This will subdivide the edges once, but it will also open the **subdivide** **toolbox** at the bottom of the toolbox at the left of the view area. In the subdivide toolbox change the **number of cuts** to **2**. Then, select and delete the lower of the two new hoops that have been made, then manually create polygons to stitch the hull back together:

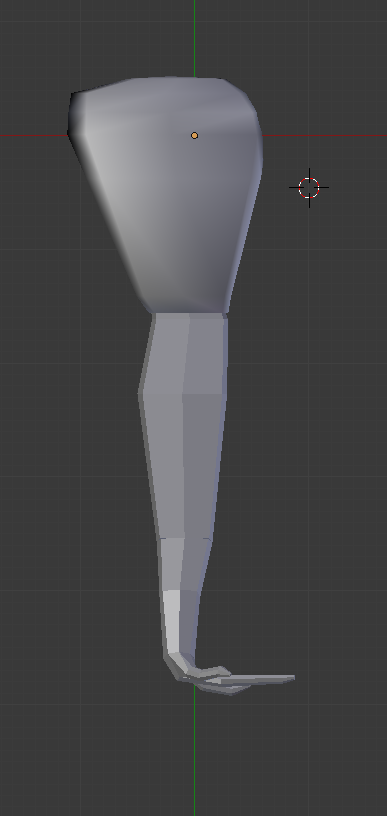




Next, select the vertices making up the mid-calf hoop and **scale by 1.5**, making sure that you have the pivot set to **active element**. After this, move the enlargened hoop caudally so that the cranial margin of the calf is straight:



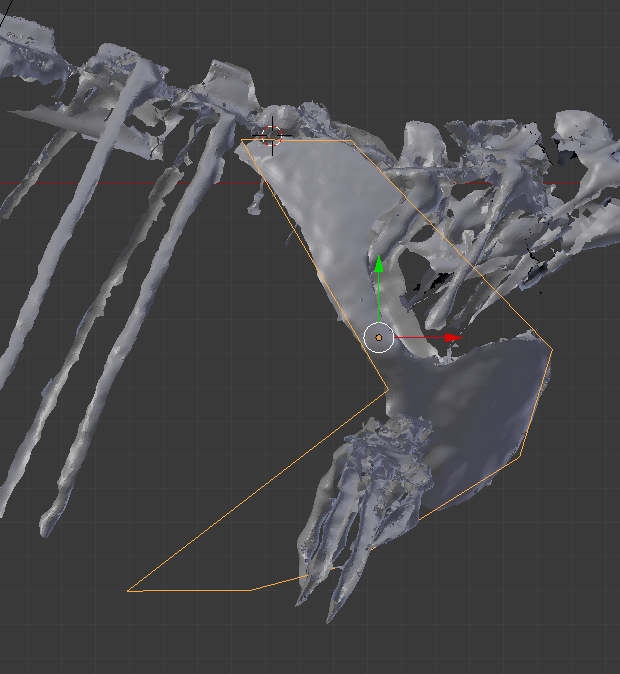
The metatarsus and pedal segments are simply modelled by fitting hoops around the bones and connecting them, either using lofting or manually created polygons:



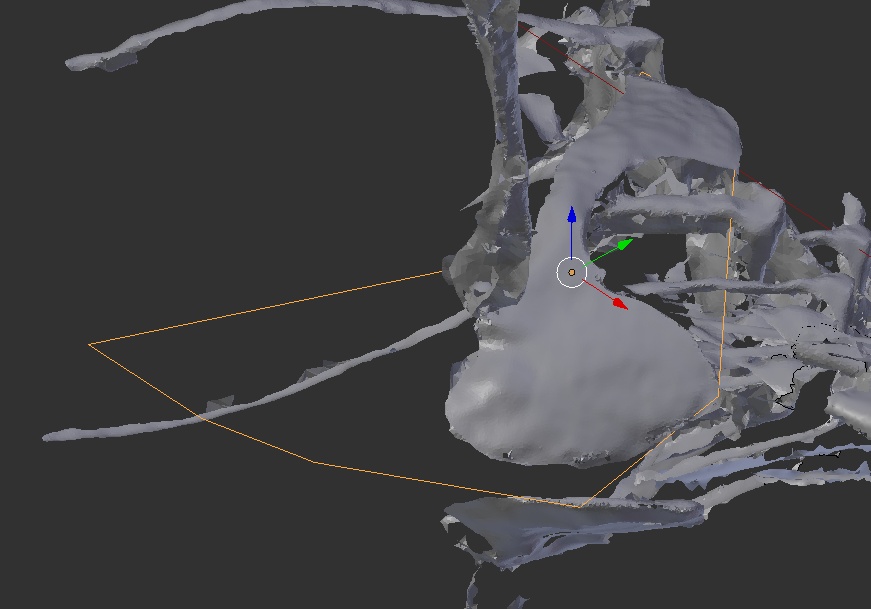
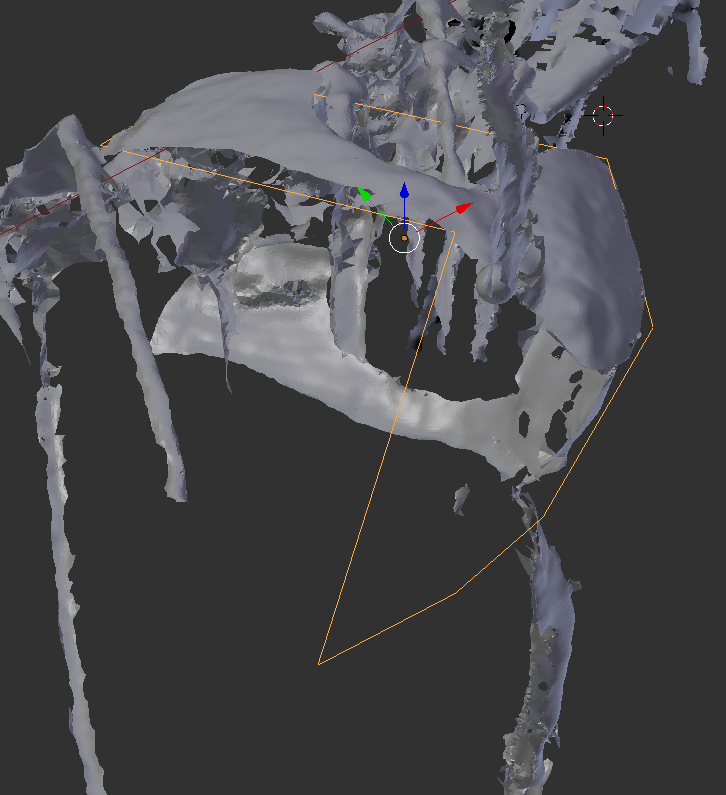
*Arms**(pectoral limbs)*

Assumptions: Cross-sectional profile of limb is elliptical. Minimum extent of pectoral and scapular muscles is constrained by a straight line drawn from the midline of the ribs to the deltapectoral crest for the former and a straight line from the dorsal border of the scapula to the deltapectoral crest.

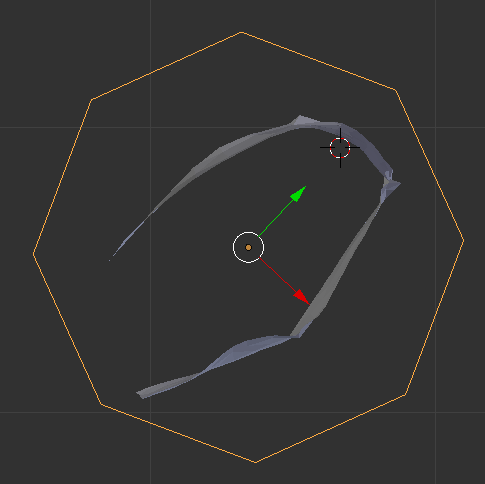
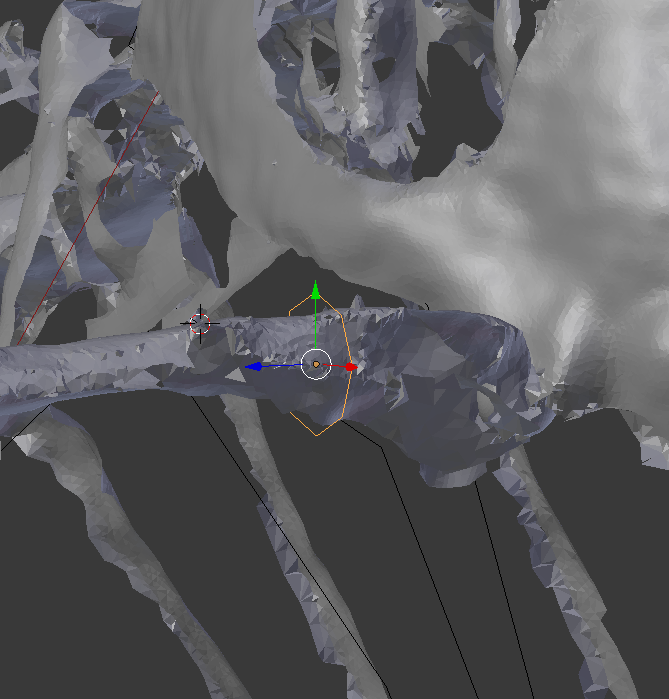
First of all, we will create a shape approximating the outline border of the pectoral girdle, starting in lateral view. The caudal extent of the pectoral muscles is poorly constrained – I have usually reconstructed it as protruding a short distance back from the caudal edge of the scapula as shown:



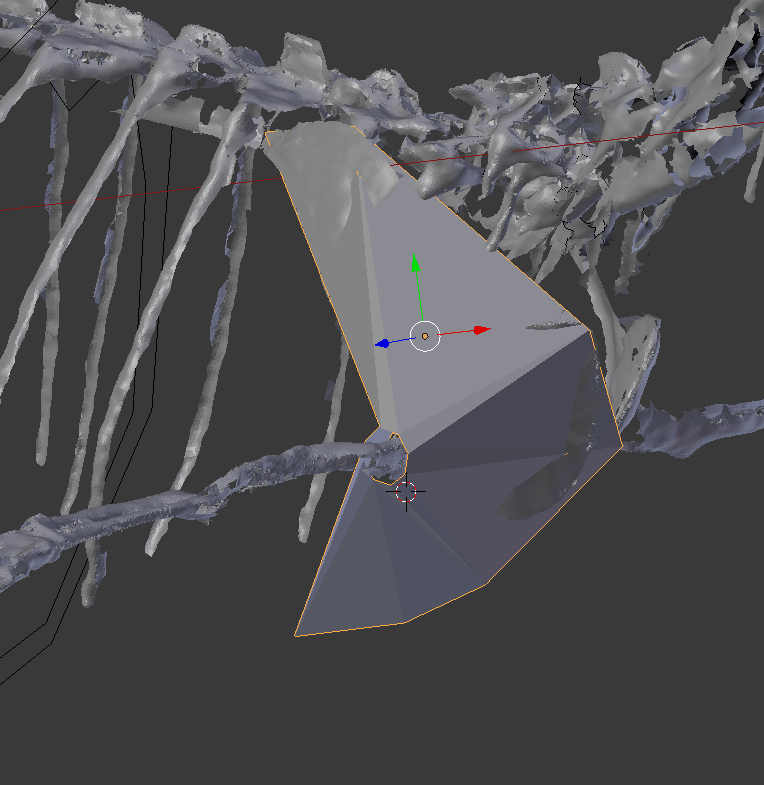
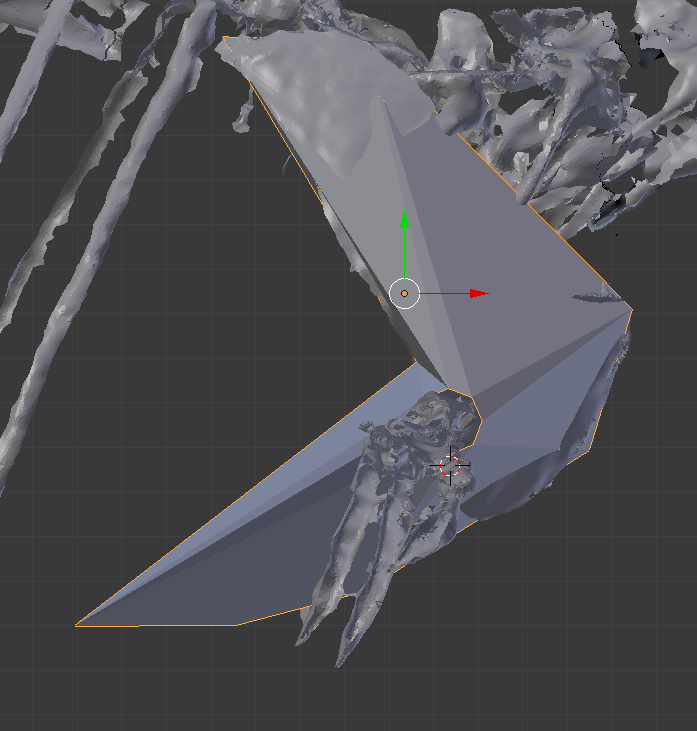
You can use as many vertices in this hoop as you need to fit around the borders of the pectoral girdle. Once fitted in lateral view, switch to a 3D perspective and fit the verts to the edges of the girdle. The ventral series of points representing the ventral border of the pectorals should be moved to the midline:



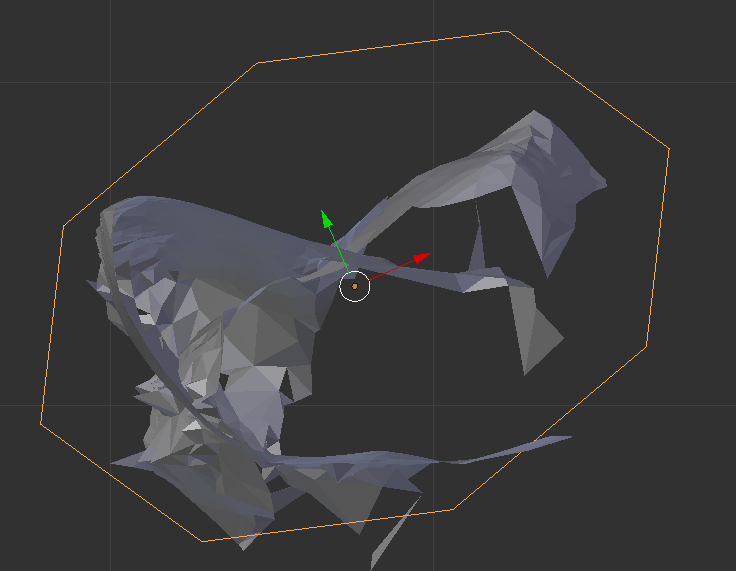
Next, fit a normal octagonal hoop around the deltopectoral crest on the humerus, maintaining its circularity (i.e. no non-uniform scaling):



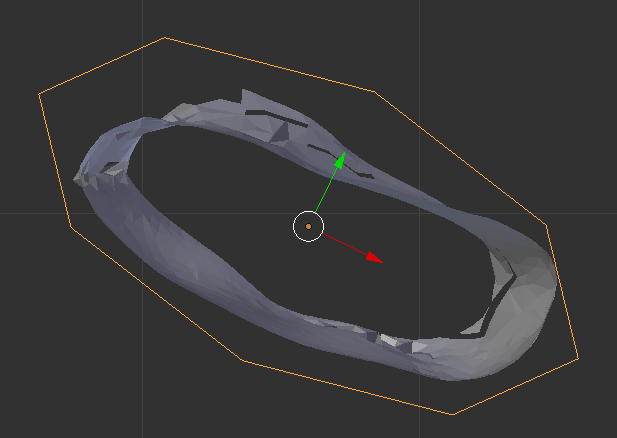
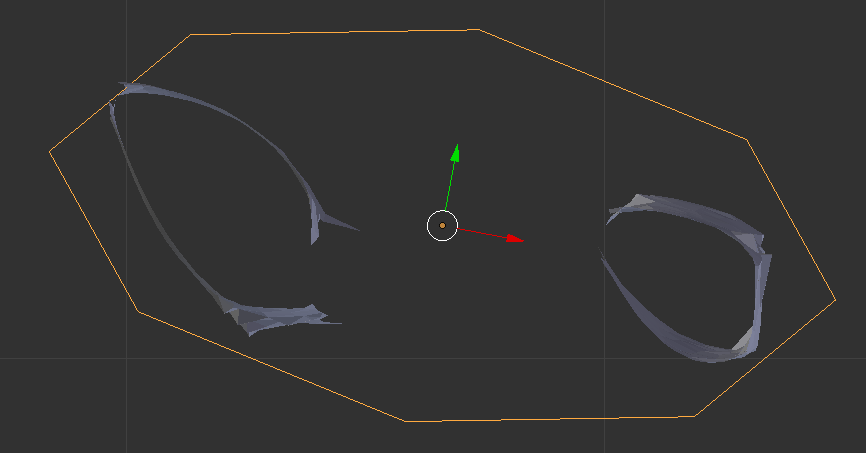
Now, stitch the pectoral girdle hoop to the deltopectoral hoop with manually created polygons (NB this is supposed to be very tightly fitted):



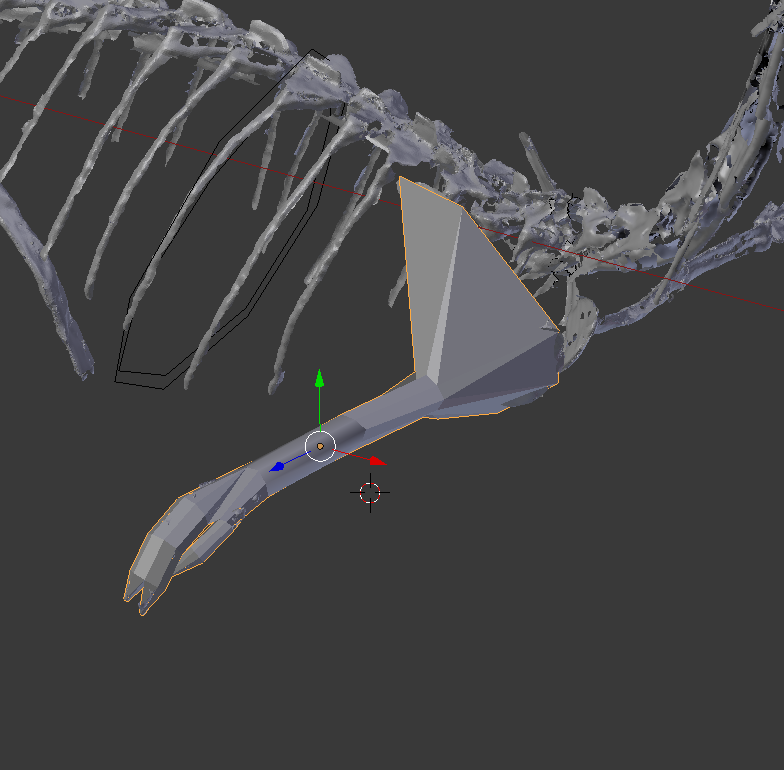
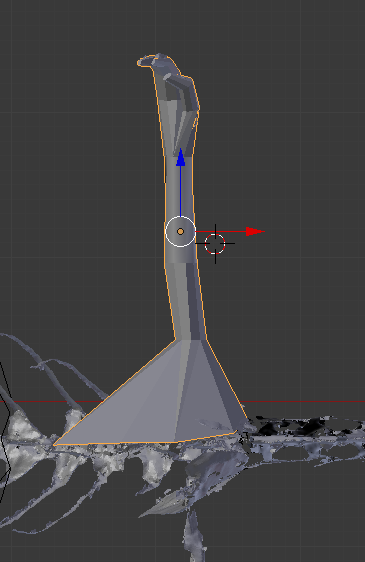
Next, fit a hoop to the elbow, located at the distalmost extent of the humerus:



Fit a hoop to the widest point of the antebrachium, and to the wrist:



Join the added loops to the pectoral girdle shape either by lofting or manually creating polygons. The metacarpus and manual segments are simply modelled by fitting hoops around the bones and connecting them in the same manner:

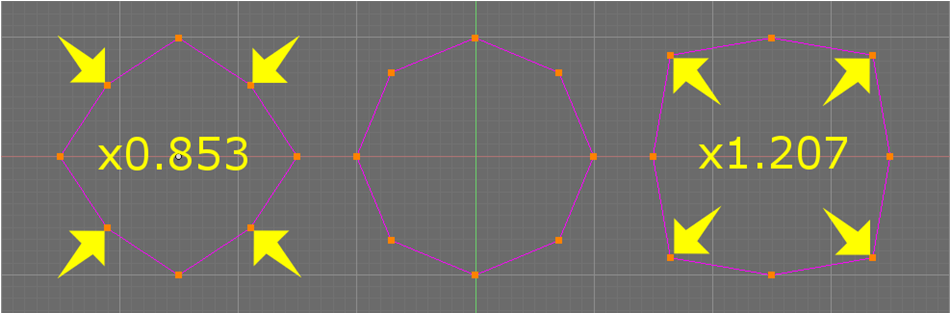


**4: Maximum and Minimum segment iterations**

You will have noticed that there are a lot of areas of the reconstructed body volume that are poorly constrained by the underlying skeleton – most of the body, in fact. While we consider our assumptions about body shape to be reasonable, they are likely to be inaccurate to varying degrees. For that reason, we believe it is important to create a range of iterations of the mass model within sensible boundaries, and see how data from these various models alter the results for mass properties you may obtain from them (i.e. a form of sensitivity analysis)—and thus how specific your conclusions might be. To do this, create maximum and minimum volume versions of each of the body segments except the head – neck, chest, tail, pelvic and pectoral limbs.

To create our maximum and minimum models, we scaled the outline hoops the were used to make each segment (assuming that the length of each element – tails, limbs etc – was well-constrained by the skeleton, but the girth was not). We used both uniform and non-uniform scaling to change not only the size but also the shape of the cross-section outline, in a crude version of the superellipse method used by Motani (*Motani, R., 2001. Estimating body mass from silhouettes: testing the assumption of elliptical body cross-sections. Paleobiology, 27(4), pp.735-750*).

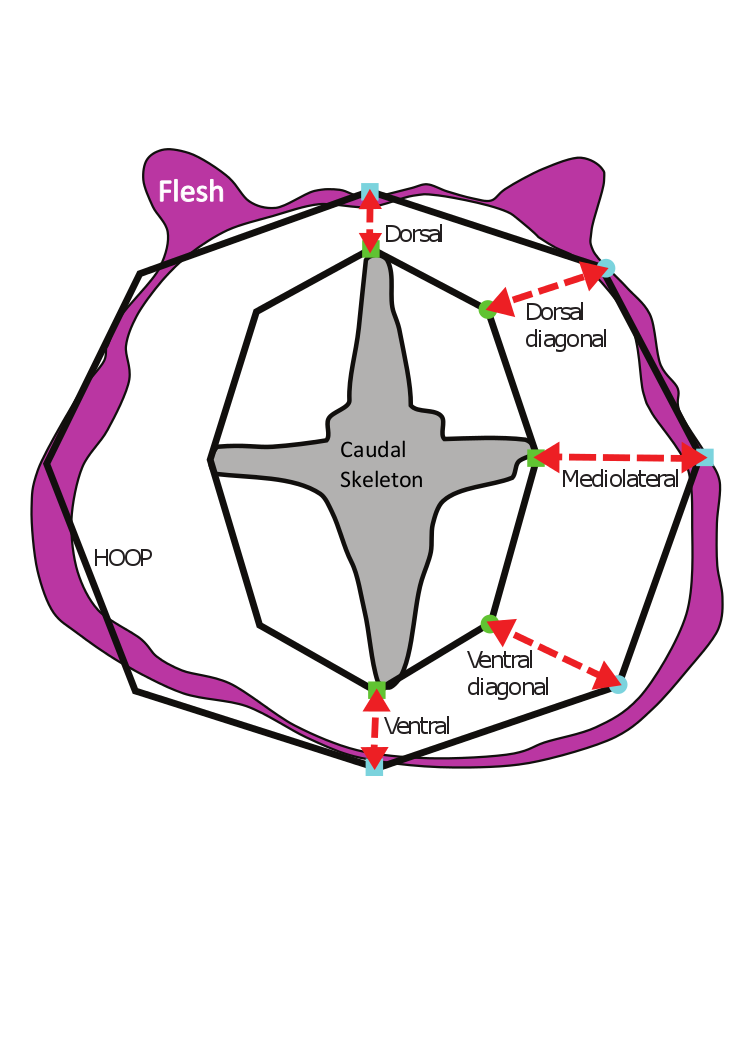
We decided that +/- 20% was a plausible maximum amount of variation in outline size, and that outline shape plausibly could vary from more square to more diamond in profile. Each hoop was therefore scaled uniformly by + or – 20% (except with exceptions, noted below), and then had the **diagonal vertices** scaled as shown to approximate ‘more diamond’ and ‘more square’ profiles (the values shown create shapes halfway between a circle and a diamond, and halfway between a circle and a square):



Specific instructions for each segment are below:

*Scaling the head*

The head is considered to be well-constrained by the skull, and so it was not scaled.

*Scaling the neck*  
  
The neck is poorly constrained and so was fully scaled (all hoops +/- 20% and diagonals scaled by 1.207 / 0.853). **NB: when scaling the hoops, make sure the pivot point is in the centre of each hoop. This will be between the two mediolateral vertices (see figure below):**

**As the hoops are now part of the mesh volumes, you will have to manually select the two mediolateral vertices, use snap to position the 3D cursor between them, and then scale using the 3D cursor as the pivot point.**

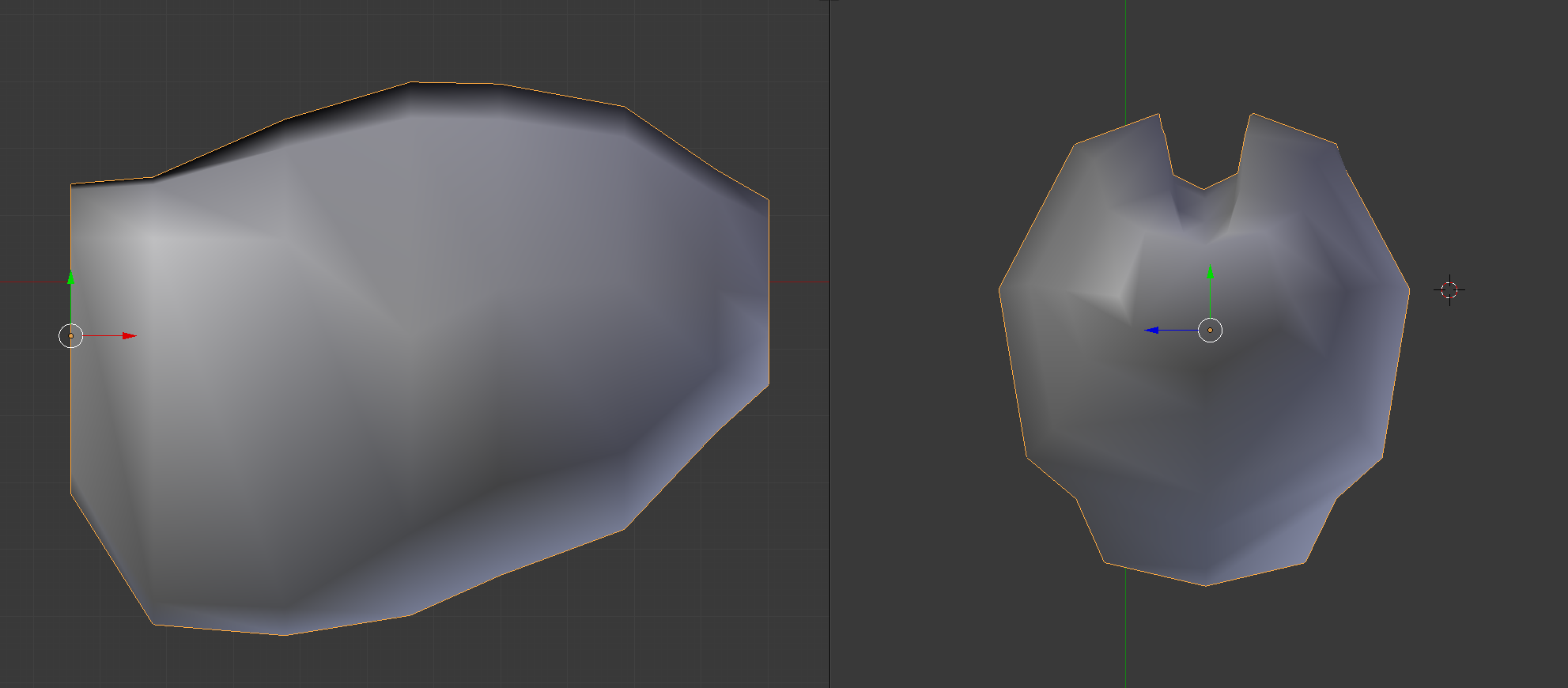
The pharyngeal cavity will also be scaled by +/- 20% in the same way. We assume that it is circular in cross section, however, and so do not scale the diagonal vertices to alter cross-section shape.

*Scaling the body*

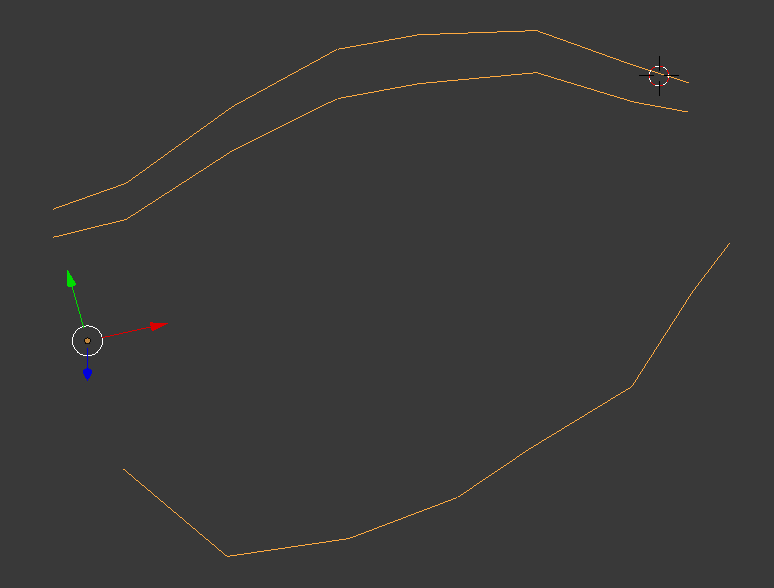
The body outlines are fitted to the ribs, and so already represent the minimal model. To create a maximal version, all hoops are scaled + 20% and diagonals scaled by 1.207.

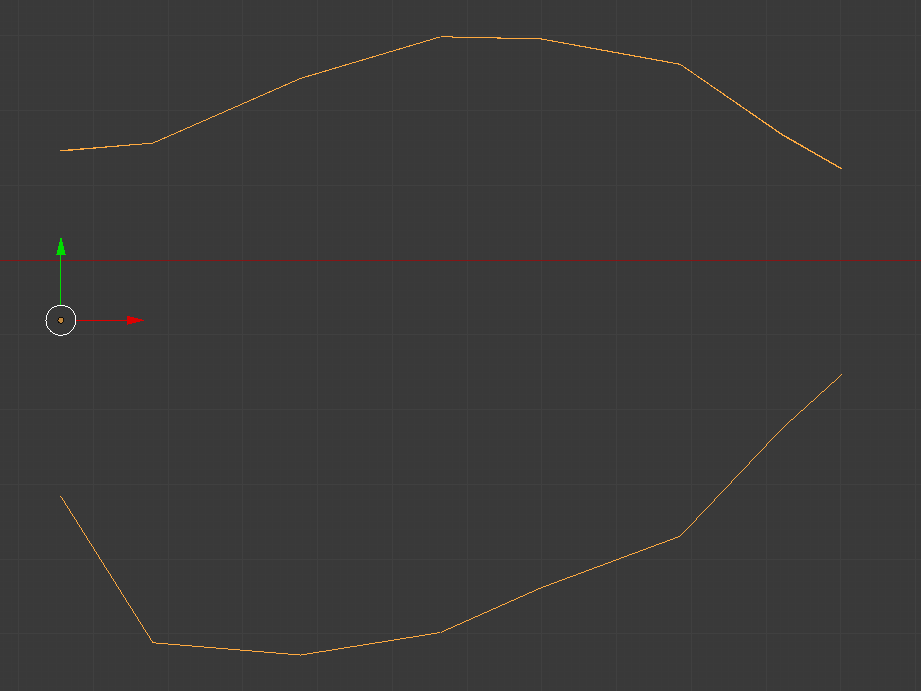
*Scaling / creating the lungs*

The lungs represent a special case, as we assume that these fill a set portion of the chest cavity. You should have modelled the inside of the chest cavity so far, which will look something like this (shown in lateral and frontal view):

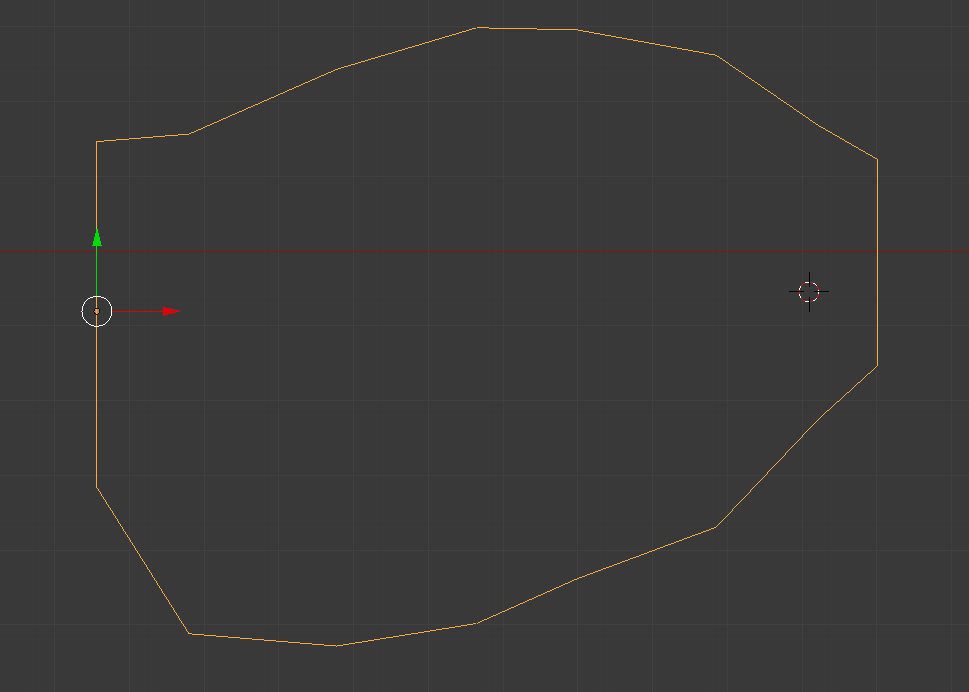


We are going to create two copies of the lungs, one that fills the dorsal 50% of the chest cavity and one that fills the dorsal 25% (NB if you are working with a non-theropod, you might want to fill the cranial 50% etc.). To do this, we first create a 2D outline of the chest cavity by copying the chest cavity mesh, going into edit mode, and deleting everything except the outline edges in lateral view. This will probably leave you with an upper and lower edge that are unconnected, and in fact are three edges when viewed in any other view:

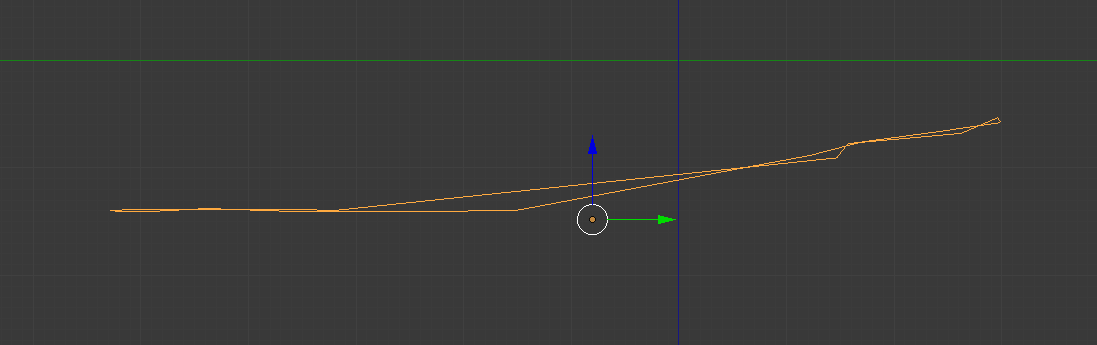




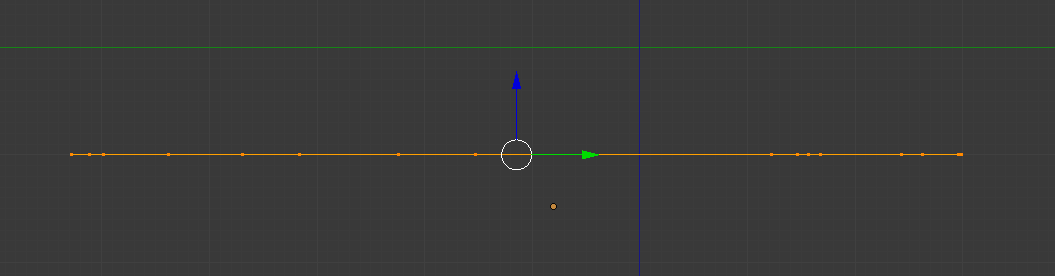
If so, delete one of the extra edges, then join the upper and lower parts together using **f** (f creates an edge if only two vertices are selected):



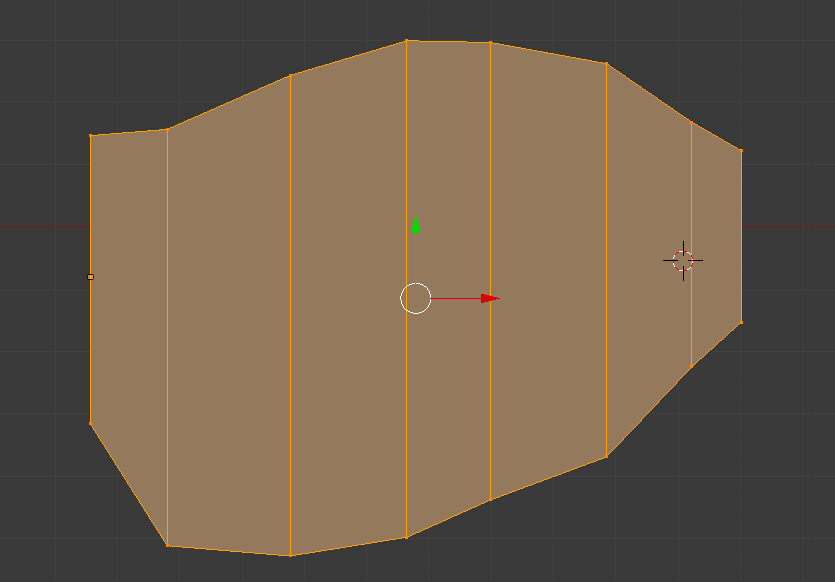
This will still be distorted if you look at it from a different view:



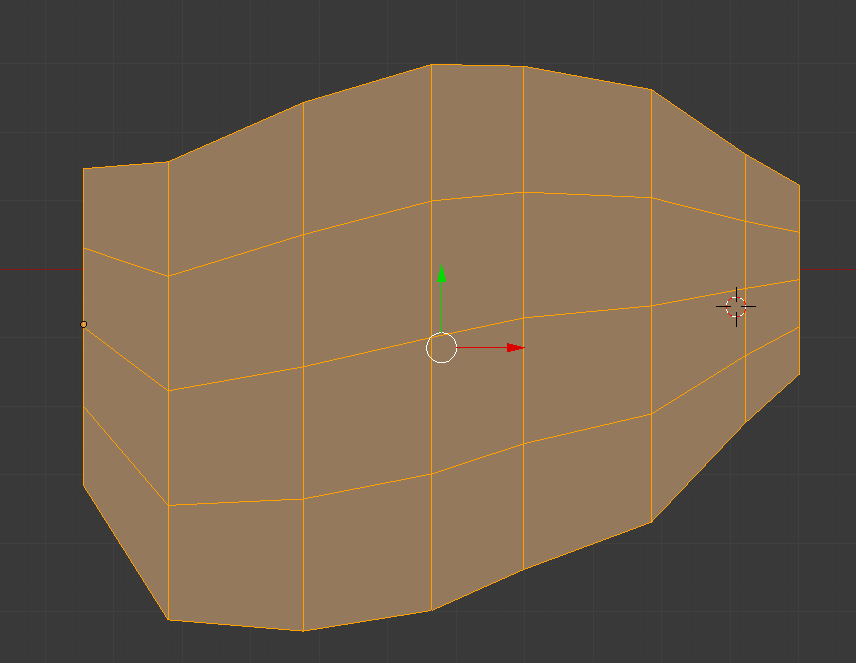
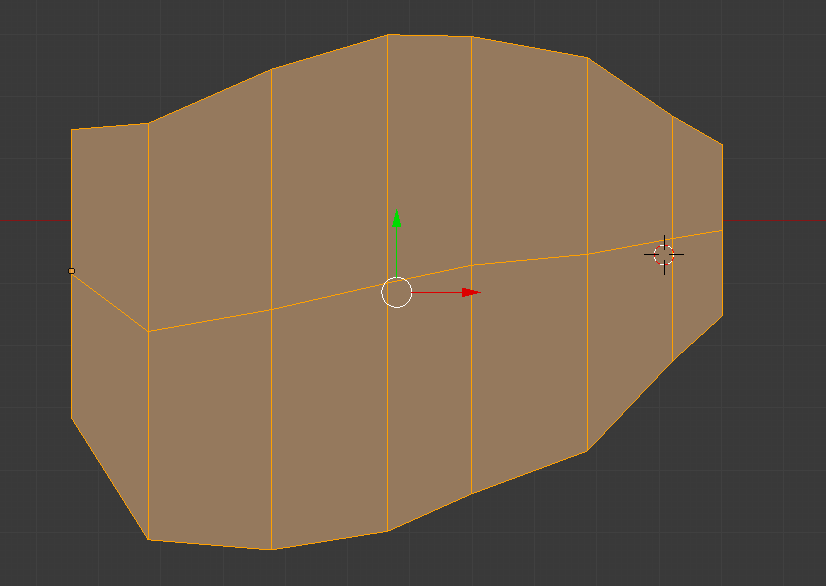
And so it needs to be flatted out. To do this easily, go into edit mode, select all vertices and scale in **z (or whatever the mediolateral axis is) by 0**:



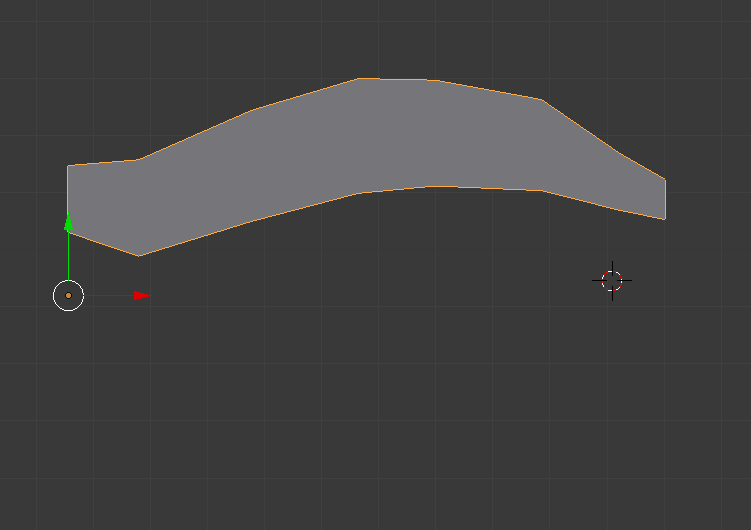
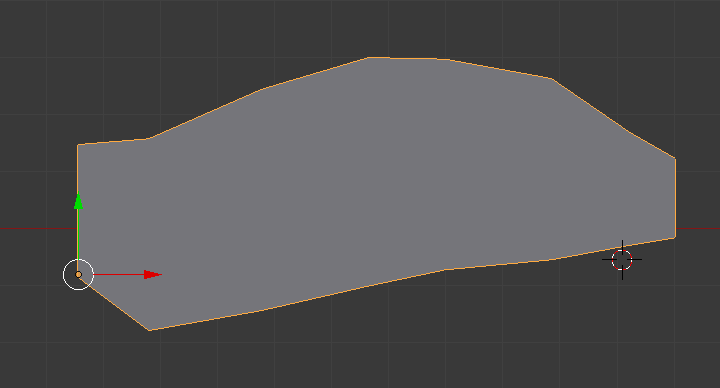
Now it is flattened out, we need to turn the flat profile into a series of planar polygons. Select upper and lower pairs of vertices (i.e. sets of four) and make square polygons out of the outline:



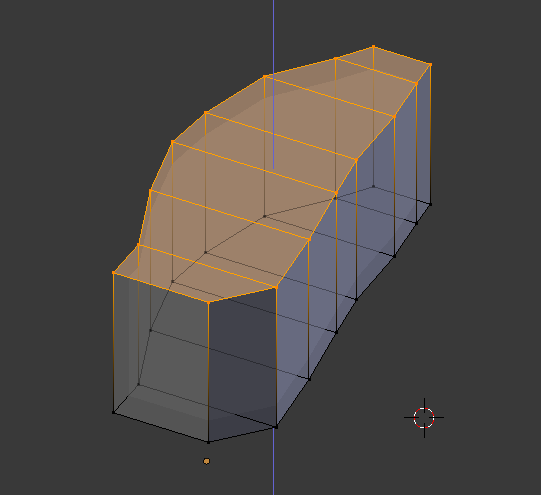
Now we are going to make two copies of this, and divide those polygons by 50% (for our half lung) and 25% (for our quarter lung) using the **subdivide** menu, with one cut for the 50%, and three for the 25%:



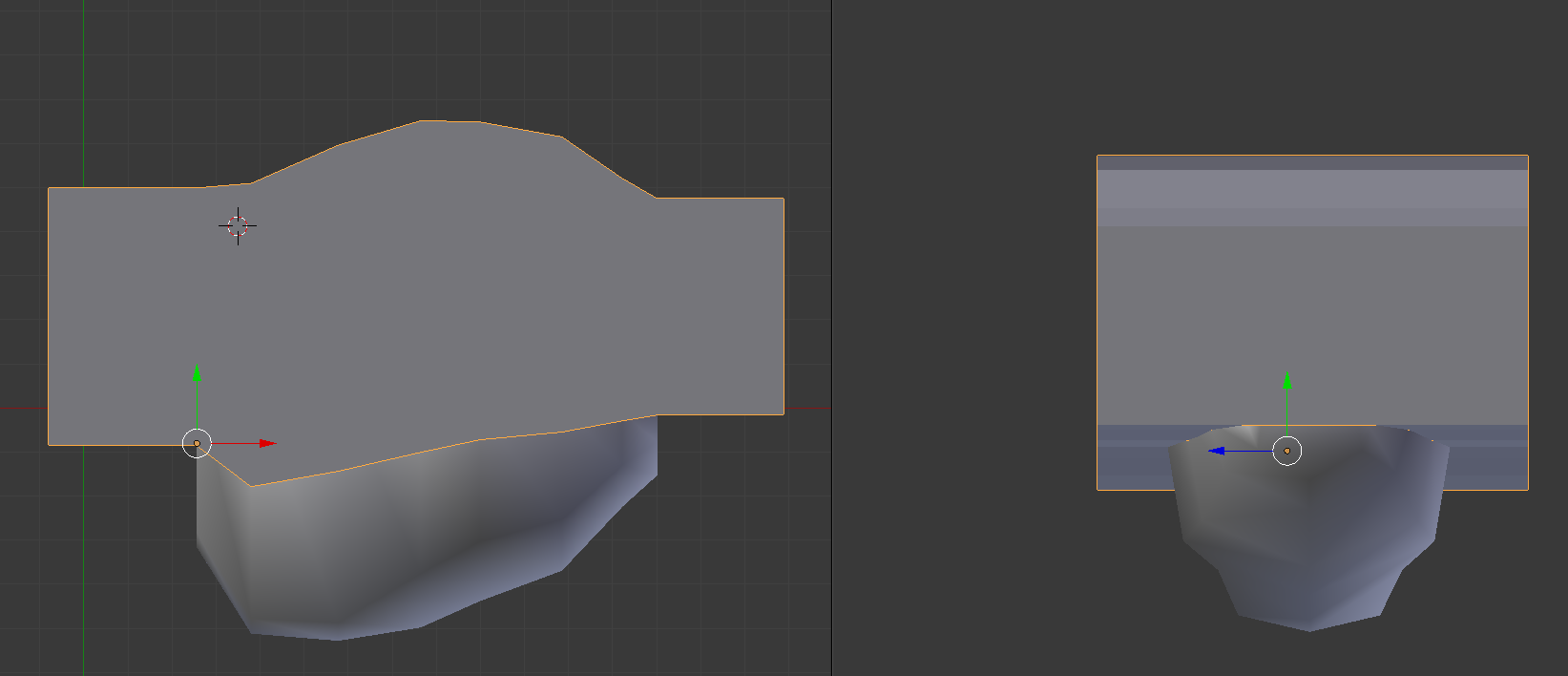
Next, delete the unwanted bottom half (for the 50%) and three quarters (for the 25%) of the shape:



Next, we are going to use the **extrude** function to turn these 2D outlines into 3D shapes, which we will eventually use to divide the lungs up using a **Boolean intersection**. Go into edit mode, select all the polygons of one of the shapes, and extrude using the **e** key:



The aim is to make a box that covers the top half of the lungs, so extrude the top and sides too, and move the base out to make something like this:



Next, make a copy of the chest cavity and a copy of the box (just in case). Select one of the copies of your chest cavity. Now, make another view area, and turn it into a **properties window** by clicking on the box in the top (or bottom) viewport menu that it furthest to the left and selecting **properties** from the popup list. There should be an **add modifier** button visible, click on it to open the **modifier menu**. From the **generate** sub-menu pick a **Boolean** modifier. Change the **operation** to **intersect**, and then use the **dropper icon** in the **object menu** to pick the box thing you just made. The results should be something like this:



Use manually created polygons to stitch up any holes (Blender’s boolean algorithm isn’t perfect). Repeat the process for the 25% lung.  
  
*Scaling the tail*

The tail is poorly constrained and so should be fully scaled (all hoops +/- 20% and diagonals scaled by 1.207 / 0.853).

*Scaling the pectoral and pelvic limbs*  
  
The limbs as constructed are already considered to be fairly minimal models, so to make the minimal versions, scale the diagonals (by 0.853) only. For the maximal versions, fully scale the hoops (+20% and diagonals by 1.207).

**5: Maximum and Minimum model iterations**

Next, we are going to combine these max and min segments to make max and min models. For our work, we wanted to establish the bounds on CoM position (maximally caudal, cranial, dorsal and ventral plausible positions of the whole-body CoM), as well as body mass (maximal and minimal mass). We assume that the head is reasonably reconstructed; hence it does not vary here as min/max versions. The models are made as follows:  
  
*Max Caudal CoM:* Max tail, max pelvic limbs, min body, max lungs, min pectoral limbs, min neck, max pharyngeal cavity, head.

*Max Cranial CoM:* Min tail, min pelvic limbs, max body, min lungs, max pectoral limbs, max neck, min pharyngeal cavity, head.

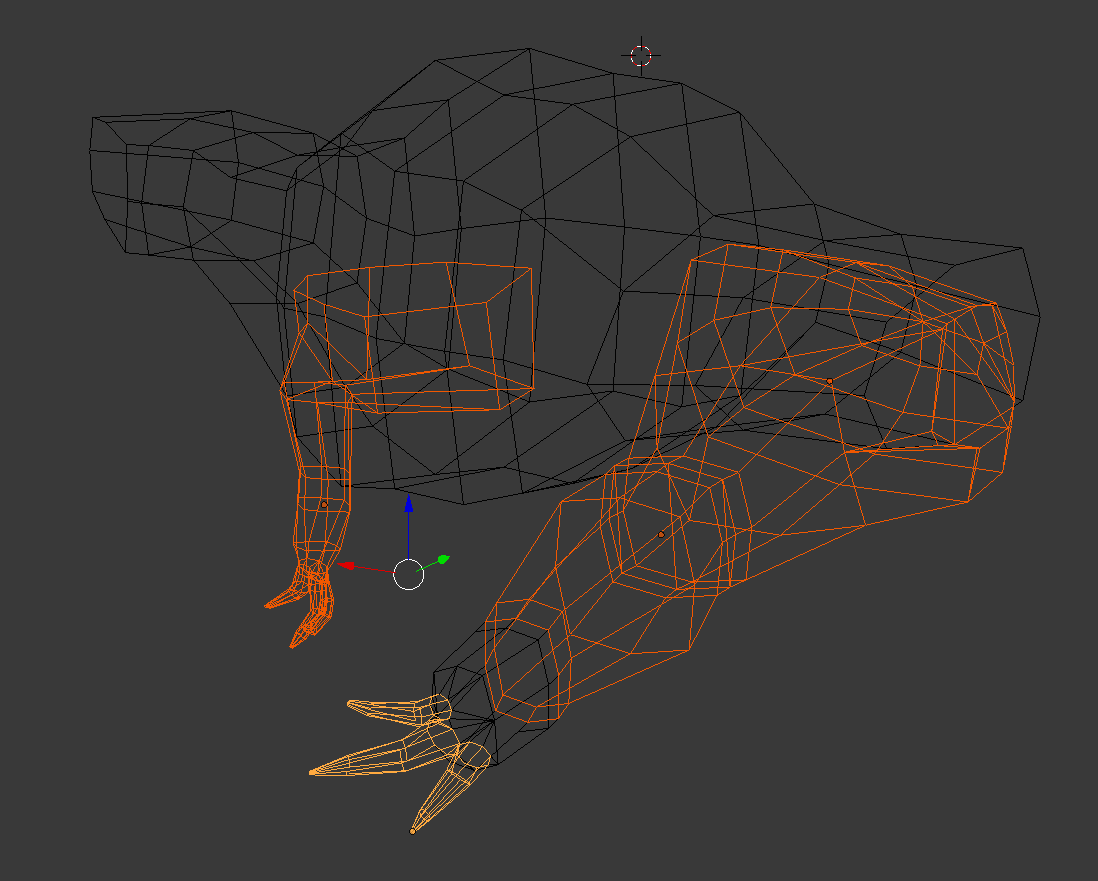
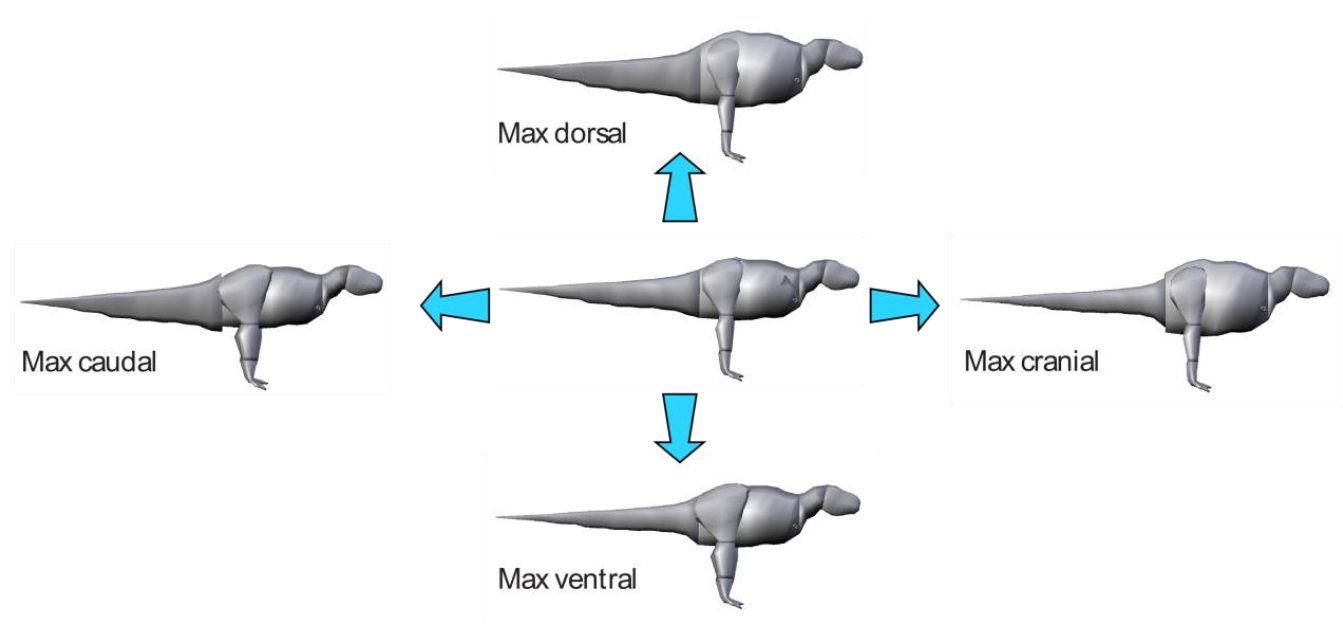
*Max Dorsal CoM:* Max tail, min pelvic limbs, max body, min lungs, max pectoral limbs, max neck, min pharyngeal cavity, head.

*Max Ventral CoM:* Min tail, max pelvic limbs, min body, max lungs, min pectoral limbs, min neck, max pharyngeal cavity, head.

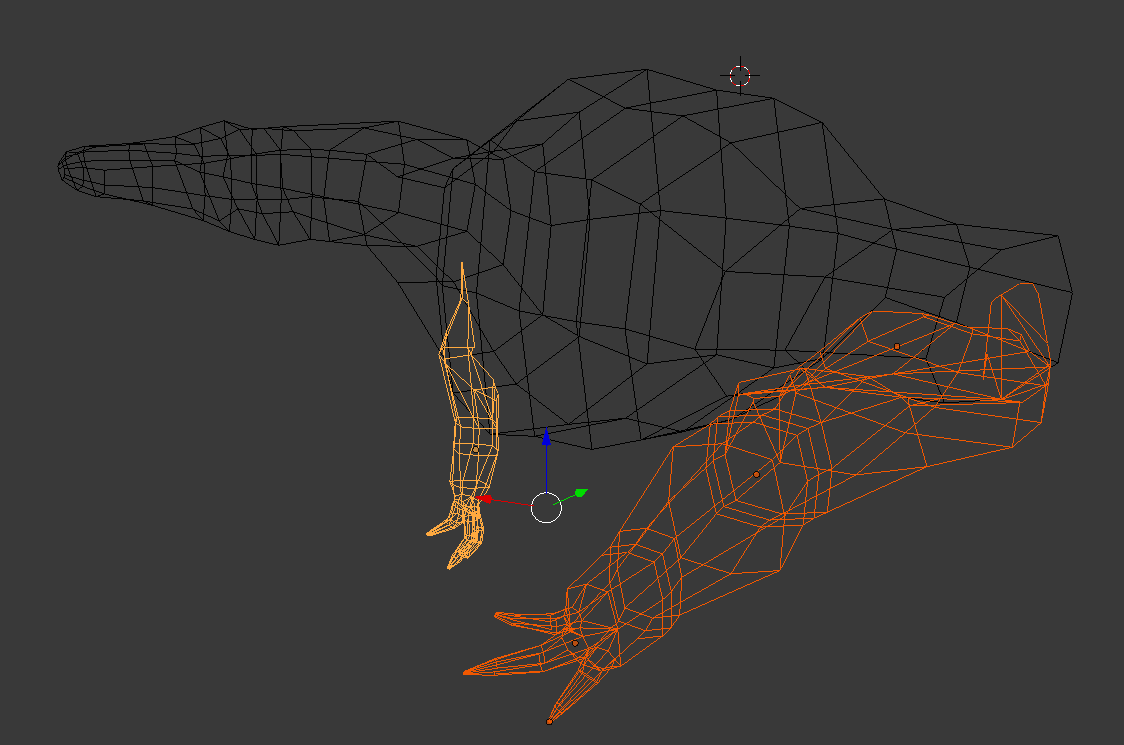
*Max Mass:* Max tail, max pelvic limbs, max body, min lungs, max pectoral limbs, max neck, min pharyngeal cavity, head.

*Min Mass:* Min tail, min pelvic limbs, min body, max lungs, min pectoral limbs, min neck, max pharyngeal cavity, head.

One thing that you will notice when making these model iterations is that **the limbs intersect the body**. This is bad, as it means the volume where the limbs and body intersect will be counted twice in the analysis of mass properties (essentially becoming twice as dense):



To avoid this, use a **boolean subtraction** to remove the intersecting area from the pectoral and pelvic limbs for each model (**NB boolean modifiers destroy both meshes, so don’t forget to make copies before you do it!**). You should end up with something like this:



Now, you should be ready to **export all these meshes as objs** and run the mass analysis! (see **SOP on mass properties Matlab code** for more details).

Most relevant paper:

Allen, V.A., Bates, K.T., Li, Z., Hutchinson, J.R. 2013. Linking the evolution of body shape and locomotor biomechanics in bird-line archosaurs. Nature 497:104–107. doi:10.1038/nature12059 (explanatory blog post here <http://whatsinjohnsfreezer.com/2013/04/24/3d_dinosaurs/> and webpage here <http://www.rvc.ac.uk/SML/Projects/Evolution3DDinos.cfm>). Open database provided at: Dryad Digital Repository- doi: 10.5061/dryad.hh74n