University of Bristol

Faculty of Engineering: Computer Science

Design and Development Report

EXO Games CGI Project

By Oliver Pope

Full animation can be found at https://youtu.be/3LHgnKOKRos.

1. Concept Design

1.1. Initial Concept Design

Initially, I decided my design would be a lightweight, minimal exosuit. I think this design most suits the Olympic theme because top athletes are always looking for marginal gains. An Olympic athlete would not want a bulky exosuit that restricted their movement and added a lot of weight. They would want a suit made from the lightest materials possible, that was moulded to their body to provide the highest level of control. With this in mind, I gathered some images online to use as inspiration and reference for my suit design **(see Appendix 5.1.1 - Figure 17)**.

I then sketched some initial designs for my exosuit **(see Appendix 5.1.1 - Figure 18)**, incorporating some aspects of the reference images into my design. My design is more of a lightweight framework rather than a full exosuit, it has strong plastic struts that are attached to disc joints. These disc joints are the points of contact between the athlete and the suit, and they would be attached with some sort of adhesive. The suit also has plates that fit onto the bottom of the athlete’s shoes which are attached to the ankle joint.

1.2. Second Iteration Design

The initial idea was that the suit would increase the force going through the athlete’s joints allowing them to run faster and jump further. In the next stage of design, I decided that the suit needed to be more high tech so I added a jetpack attached to the back of the suit **(see Appendix 5.1.2)**. However, in order to fit in with the rest of the suit design, I decided it should be able to retract into a small pack on the athlete's back when it was not in use. This would provide less aerodynamic resistance when the athlete was performing their run-up, but they would still be able to unfold and use it to jump much further. In order to keep the Olympic spirit intact, I decided that there should be some rules restricting the use of the jetpack. I decided that athletes would be limited to a short burst of propulsion from their jetpack, this way the sport would still be about the athlete’s performance and not just the suits.

1.3. Final Concept Design

In my final design, I added a helmet for safety purposes as the athlete would now be able to jump a lot higher using the jetpack. The helmet has a small retractable visor that prevents sand from getting in the athlete’s eyes when they land. I also added chest and backplates as the suit would have to be able to hold the jetpack and it looked a bit strange just being attached to a small disc joint. The chest and back plates are connected using straps over the athlete’s shoulders and around their torso under the arms **(see Appendix 5.1.3)**.

2. Geometry and Model Development

I used Autodesk Character Generator [1] to generate and download an athlete. I could then build my exosuit around this character, rather than start from scratch and have to model an athlete first. I used Autodesk Maya [2] to model and animate my geometry.

2.1. Base Exosuit Development

Initially, I decided to create a basic version of the exosuit with just the joints and basic connecting struts **(see Appendix 5.2.1)**. I would then be able to iteratively build on and improve it to be more like my final concept design.

I created the joints by resizing polygon cylinder primitives to be thin discs. I then positioned these on the athlete's joints against the skin. To create the struts, I initially resized polygon cube primitives to be rectangular structs that were the right length to connect each joint. Once these were positioned to be between the joints, I added some more detail to their shape by inserting edge loops. This allowed me to bend the struts around the athlete's leg muscles to prevent intersection. I also used edge loops and the extrude tool to create the hip struts which connect to the lower back joint, and the back struts that connect the upper back joint to the shoulder joints. Once all of my geometry was created, I decided to bevel a lot of the edges of my exosuit as they looked unnaturally sharp.

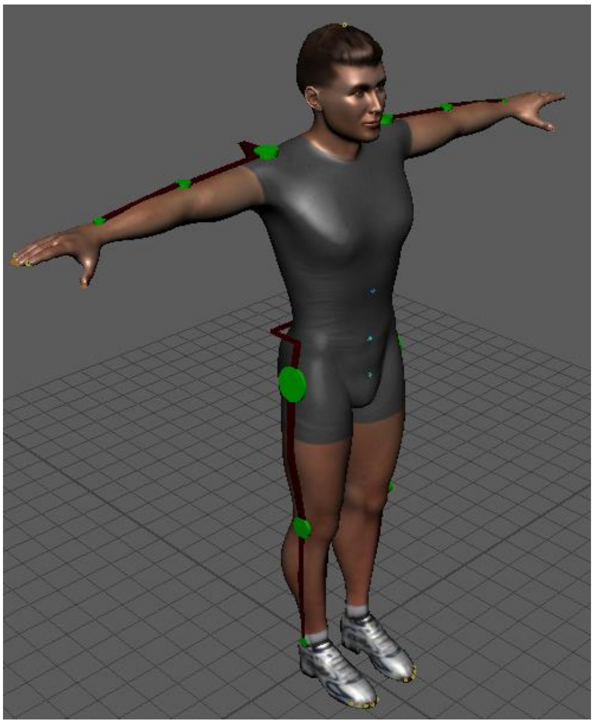
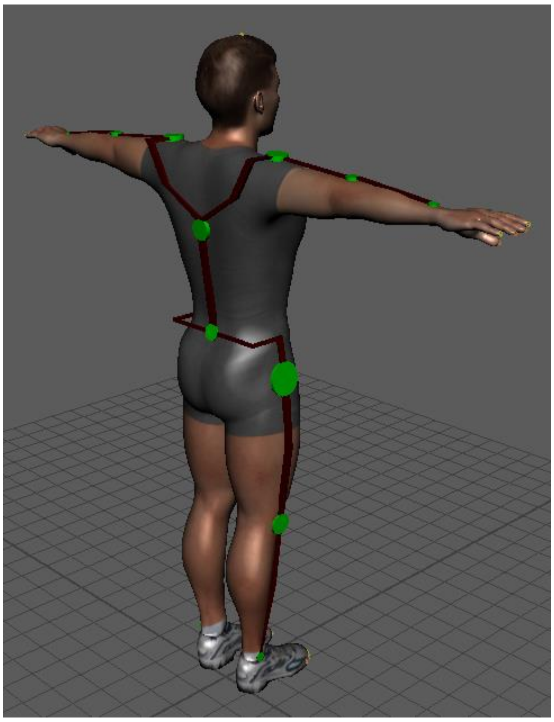

Figure 1 - Base Exoskeleton

2.2. Exosuit Shoes

In the next iteration of my development, I added the plates that fit onto the bottom of the athlete's shoes **(see Appendix 5.2.2)**. To try to create the base shape of the plate I used the *EP Curve Tool* in the orthographic view to plot the outline of the athlete's shoe as a curve. I then used this curve to create a planar surface and extruded this to create a base of the shoe. Although this did create the base of the shoe with the correct shape, it did not create geometry with clean flow lines.

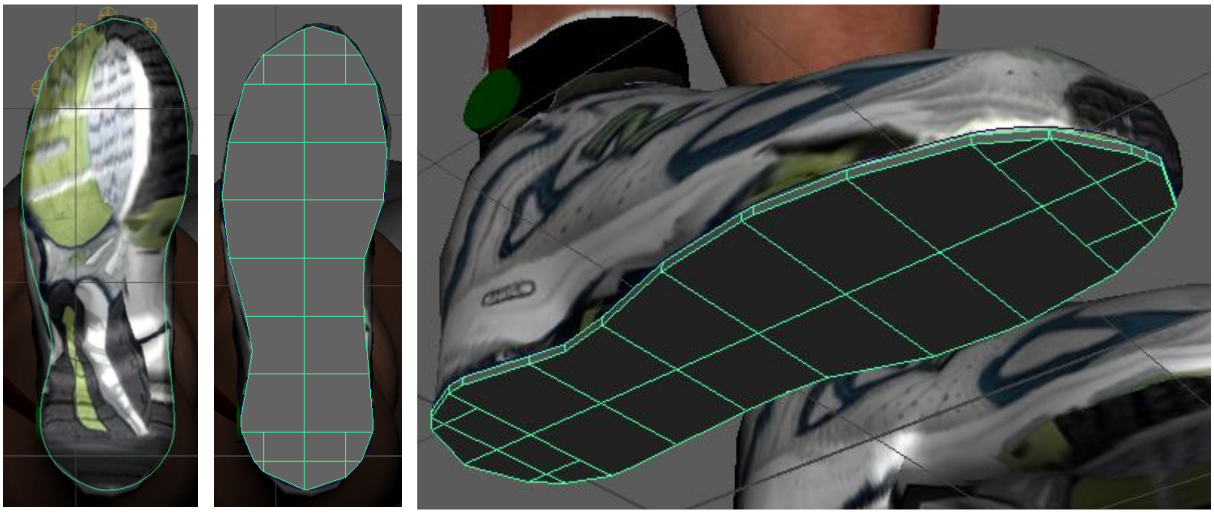

Figure 2 - Exosuit Shoes created using the EP Curve Tool

I decided to instead start off with a polygon cube primitive and reshape that to be the base of the shoe. To reshape the cube, I first resized it to cover the whole base of the shoe. I then added edge loops and moved the vertices to fit the shape of the shoe.

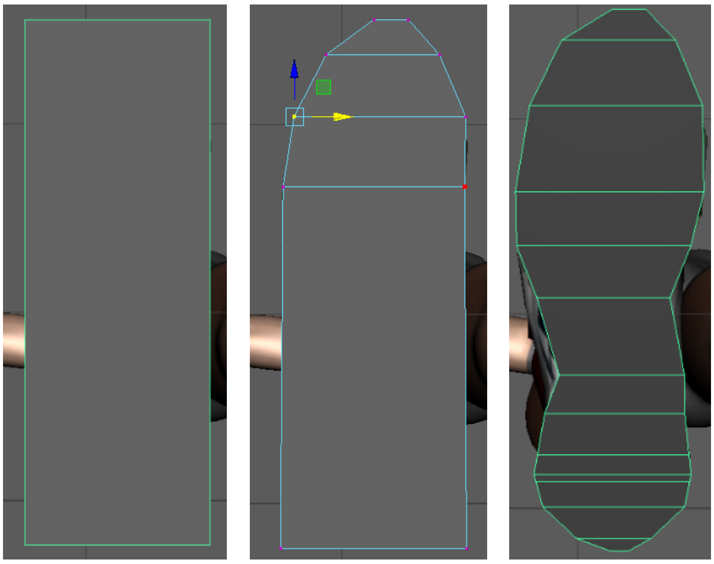

Figure 3 - Exosuit Shoes created by reshaping a cube

This method produced geometry with much cleaner flow lines. The geometry was not as detailed as the last method; however, the base of the foot is not something that is viewed as much as the rest of the model. For this reason, I decided to prioritise cleaner flow lines, if I needed to, I could have added more edge loops to match the shape more closely to the athlete's shoe. I decided that it was not necessary as it would have taken more time and there would not have been a noticeable difference in the final animated film.

Finally, I extruded some faces to create the support which connects the base plate to the ankle joint.

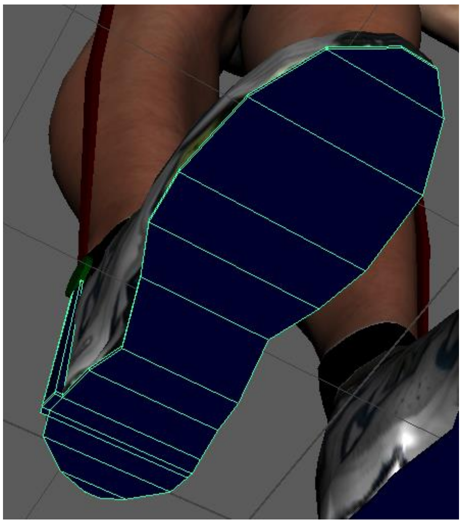

Figure 4 - Final Exosuit Shoe

2.3. Chest and Back Plates

The next section of my model that I created was the chest and back plates **(see Appendix 5.2.3)**. I created these by duplicating and extruding faces to create geometry that was perfectly moulded to the athlete's body. This resulted in clean geometry because it was based on the existing geometry of the athlete. I also then used the same technique to recreate the struts, so they were moulded to the athlete's body **(see Appendix 5.2.3 - Figure 29)**. At this stage, I decided to remove the arm sections of the exosuit because I felt they were not necessary for long jump and would not add anything to the design.

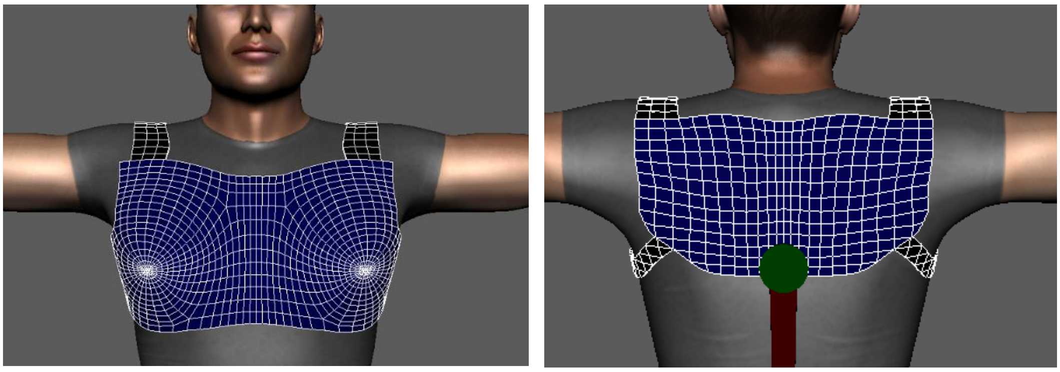

Figure 5 - Chest and Back Plate Geometry

To create the straps connecting the chest and back plates I used the same technique. This worked well for the shoulder straps and they joined perfectly with the plates because I used the existing geometry. The existing geometry did not work as well for the straps going under the athlete's arms. To create these, I used the previous technique then I moved some of the vertices to create a smoother more natural-looking strap.

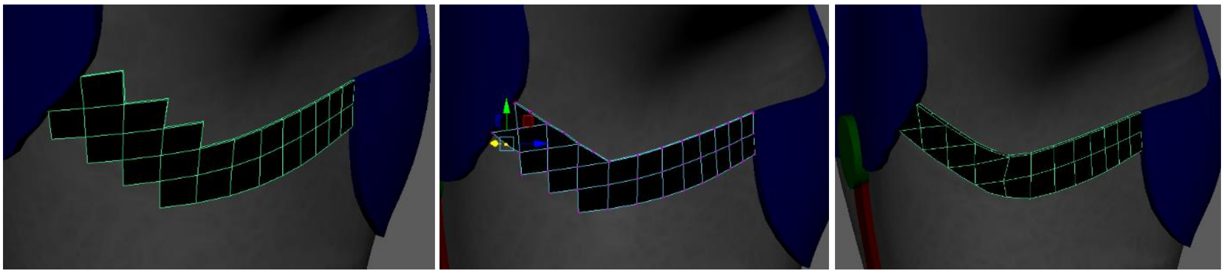

Figure 6 – Under-arm strap creation

2.4. Helmet

The helmet was the most challenging piece of geometry to create. I wanted to produce a helmet that would fit tightly to the athlete's head with a sleek design. However, I could not simply duplicate the existing geometry because the athlete's head is not a smooth surface, the athlete's ears and hair were the main issues.

First, I duplicated the athlete so I could use it without altering the original. I then selected an edge loop around the athlete's neck and used the *Detach Components* T*ool* and used *Mesh->Separate* to get the athletes head and body as two separate objects. I could then delete the rest of the athlete's body as I only needed the head to model the helmet around. I decided I would make the helmet in sections rather than as one object to overcome the issues with the ears. I created one half of the helmet then duplicated it to create the other half.

2.4.1. Main Helmet Section

The first section I created was the main part that covers the top, back and side of the head. I created the initial shape by duplicating faces from the athlete's head, leaving a space around the ear **(see Appendix 5.2.4 - Figure 30)**. I then used the same techniques as before to extrude the faces and create a helmet with some thickness. However, this did not produce a smooth helmet shape, the athlete's hair caused it to have lots of bumps. To fix this I tried using *Mesh->Smooth* with the most uneven areas selected, then *Mesh->Retopologize* as smoothing created a lot more faces in some areas than others. Although this technique produced a smooth looking helmet that I was happy with, it created very uneven topology without clean flow lines **(see Appendix 5.2.4 - Figure 31)**.

In my next attempt, I duplicated the same faces but then manually adjusted the vertices to create a smoother surface before I extruded them. This method was a little bit more time consuming, but it produced a smoother helmet and much cleaner geometry as for the most part it was the same as the original athlete. Overall, I think this was the better method and this was the helmet section I used for my final model.

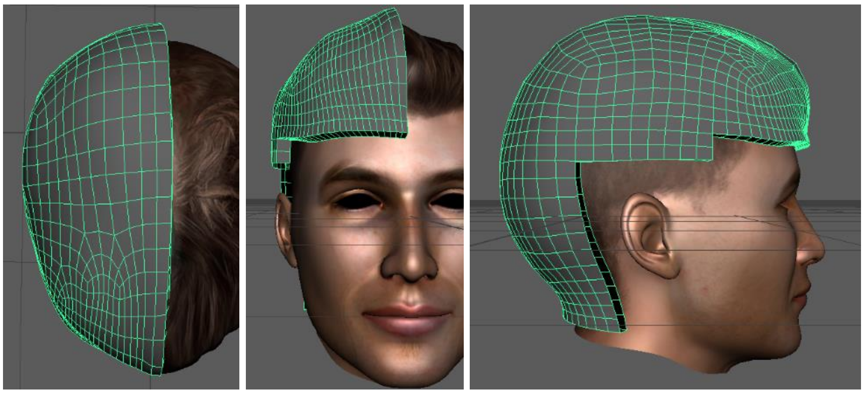

Figure 7 - Main Helmet Section

2.4.2. Helmet Ear Section

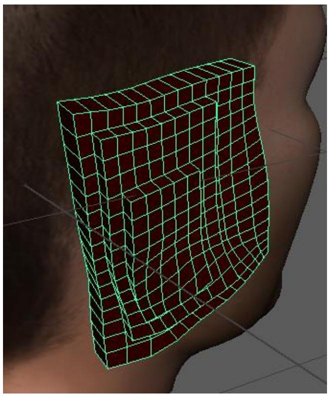

Figure 8 - Helmet Ear Section

The ear section of the helmet was another challenging part of the model to create. I decided to create a layered design with the layers building up to cover the athlete's ear **(see Appendix 5.1.3 - Figure 23)**. I created this by first duplicating a ring of faces around the athlete's ear, then selecting the inner edge loop and using *Mesh->Fill Hole*. I then extruded the mesh and used *Mesh->Retopologize* to create clean flow lines. However, I discovered if I did this the other way round it created much cleaner geometry **(see Appendix 5.2.4 - Figure 32)**. I then extruded some of the faces to create layers to cover the ear and create the finished ear section.

2.4.3. Helmet Chin Section

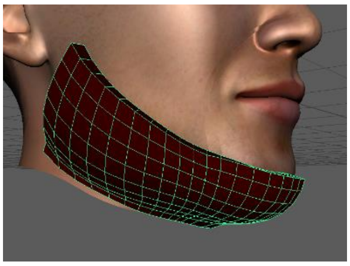

Figure 9 - Helmet Chin Section

The chin section was relatively easy to create. I simply duplicated the faces from the athlete geometry and extruded the faces to create a section of the helmet to protect the athlete's chin.

2.4.4. Complete Helmet

Once I had created all the sections of the helmet, I needed to combine them to make the complete helmet. I duplicated and extruded faces around the athlete's ear to fill the gap between the main section and the ear section of the helmet and to allow the chin section to connect to the rest of the helmet **(see Appendix 5.2.4 - Figure 33)**.

I then deleted the faces that would be hidden when these sections were joined together **(see Appendix 5.2.4 - Figure 34)** and used *Mesh->Combine* to combine all the sections of the helmet into one object. Once the sections were all combined into one object, I used *Edit*

Mesh->Bridge to connect the edges of each section together **(see Appendix 5.2.4 - Figure 35)**.

The final stage was to use *Mesh->Mirror* on the right half of the helmet and perform the same process of deleting faces that would become hidden, combining the two halves into one mesh, and bridging the gaps **(see Appendix 5.2.4 - Figure 36)**. Once I had done this, I used *Mesh->Smooth* to round off the edges and create the final helmet model.

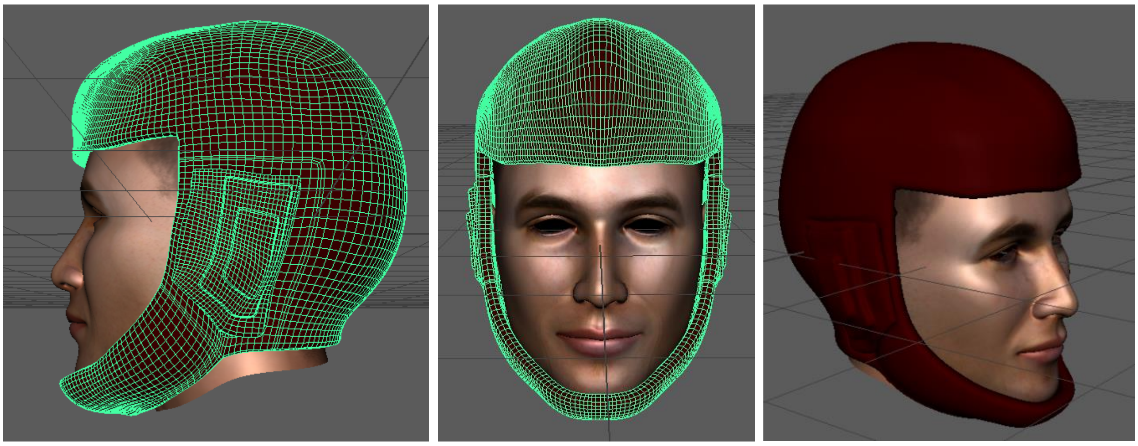

Figure 10 - Final Helmet

2.4.5. Visor

To create the foldable visor, I modelled a fully unfolded visor, a half-folded visor, and the fully folded visor, I used *Deform->Blend Shape* to allow it to change shape and created a *Set Driven Key* to control these blend shapes. I used 2 blendshapes because I wanted to create a multi-stage folding effect where the visor would fold into the centre then up into the helmet. The set driven key then allowed me to control the folding visor with one value between 0 and 1 rather than adjusting each blendshape.

I started with a polygon cube which I resized to be a thin rectangular visor. I used the *Insert Edge Loop* tool to split the rectangle up and adjusted the weight attributes of the p*olygon split ring* to make sure the edge loops were evenly distributed. After that, I extruded the centre faces to create the stalk which attaches the visor to the helmet and translated the faces to add a slight angle to the stalk.

I created the half-folded visor by selecting the vertices of each side of the visor and moving them into the middle. To create the fully folded visor I selected all the vertices at the base of the visor and moved them up **(see Appendix 5.2.4 - Figure 37)**.

Once I had created the meshes for the three stages of the visor I used *Deform->Blend Shape* to add them as blend shape inputs on the original visor shape. I created an attribute on the parent helmet object to control how folded the visor is, 0 is fully folded and 1 is fully unfolded. I used *Set Driven Key* to control this attribute so when it is set to 1 both blend shapes are set to 0, when it is set to 0.5 the first blend shape gets set to 1 and the second is set to 0, and finally when it is set to 0 the second blend shape is set to 1 and the first is set to 0. This makes the animation of the visor a lot easier because I just had to keyframe the single attribute to control the folding of the visor rather than keyframing the blend shape values.

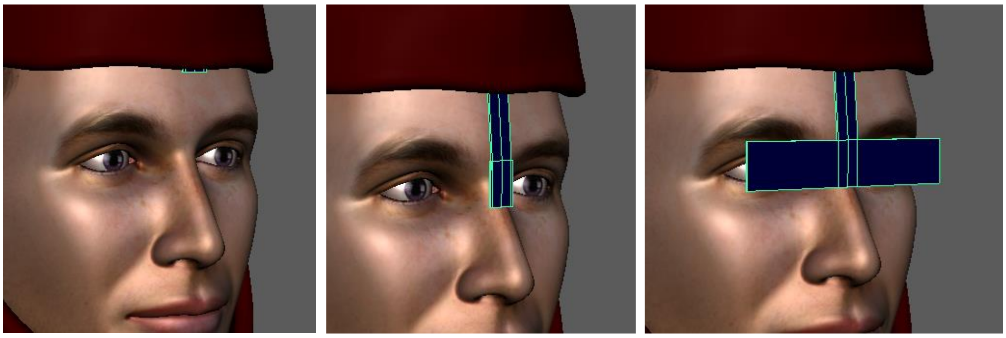

Figure 11 - Folding Visor Stages

2.5 Jetpack

I again used blendshapes controlled by a *Set Driven Key* to create a fold-away jetpack. The jetpack has more stages than the visor, I modelled four different stages of folding. I then used *Deform->Blend Shape* to allow the jetpack to change into each of these stages. I created a *Set Driven Key* to control how unfolded the jetpack is, between 1 (unfolded) and 0 (folded).

To create the fully unfolded jetpack I started with a *Polygon Cube*, I then used *Edit Mesh->Bevel* along with *Edit Mesh->Extrude* to create most of the jetpack shape. To create the thrusters on the base of the jetpack, I first had to add divisions to the face. I did this using the *Insert Edge Loop* tool, I then changed the *Split Type* attribute of the *Poly Split Ring* to Multi and increased the number of divisions. I was then able to extrude the centre of the face and bevel it. **(see Appendix 5.2.5)**

In the first stage of folding, the two thrusters on each side collapse into one. I did this using the same method I used for the visor. I selected the vertices and moved them into the new position. Using this method of duplicating the model and moving vertices ensured that the blendshape would behave as expected. It allowed me to know for certain that all of the vertices would map correctly between the two models. In the next stage, the thruster on each side folds into the centre section of the jetpack and the final stage involves the bottom of the thrusters pulling up into the jetpack.

Once I had modelled all the stages, I used the same technique I used for creating the visor to create and control the blendshapes using a set driven key. This meant I could now control the jetpack folding animation using one attribute.

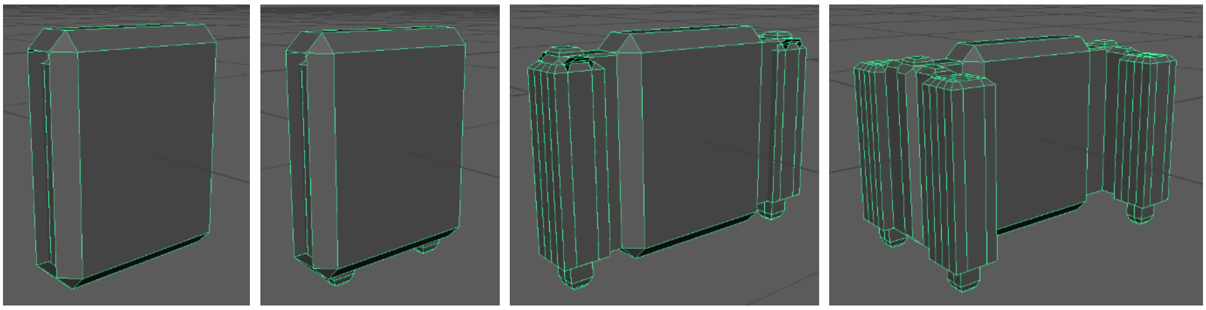

Figure 12 – Folding Jetpack Stages

2.6 Parenting and Constraints

The final step to prepare my model for animation was to parent the geometry so it would move properly. I first made sure the pivots were in the correct places so that when a piece of geometry was rotated it would look natural. For example, the leg struts should rotate around the joints and not the centre of the models **(see Appendix 5.2.6)**.

Once the pivots were all correctly set, I added parent constraints to constrain the joint geometry to the corresponding joints in the athlete's skeleton, and the helmet to the athlete's head. I then added parent constraints to constrain the rest of the geometry to the corresponding joint geometry.

These constraints controlled how the exosuit geometry would move when the athlete's skeleton was animated. This meant I was able to just animate the skeleton and the exosuit would automatically move with it.

3. Animation Design

3.1 Animation Reference

The Olympic sport I decided to animate was the long jump, I decided this at the start of the project so I could base my exosuit design around it. I watched videos of long jump events to get an overall reference for the timing and main actions that I would need to include in my animation. After watching these I decided to split my animation design into 3 main stages, pre-running, run-up, and the jump. I then watched slow-motion footage to analyse the form and screenshot the main keyframes of each stage.

The pre-running stage is a very short stage, from the start of the animation, when the athlete is stationary, to when the athlete has begun their run-up. This stage is all about the Disney Animation Principle [3], anticipation. It is the athlete's small backwards movement before they then launch themselves forward and begin their run-up. From the reference material, I noticed that athletes usually start leaning slightly forward, they then lean back so that they are upright, and finally launch forward into a run. This rocking motion helps them to build momentum and accelerate much faster.

Figure 13 - Pre-running Reference Images [4]

The run-up stage is the longest stage, however, it can be created as a cycle and then looped. I found lots of animation reference images available online for running which I used when animating my athlete.

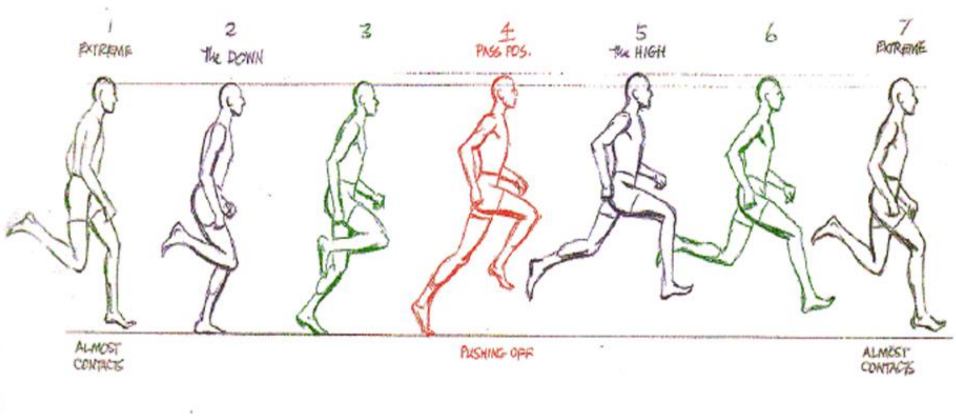

Figure 14 - Running Reference Image [5]

The jumping stage is the final stage of the animation, this stage is made up of the take-off, the flight, and the landing. Again, anticipation is extremely important during the take-off stage, without this, it looks very unrealistic. During the flight, athletes perform a running motion with their legs and swing their arms to preserve momentum. In the final landing stage, the athlete stretches their arms and legs out in front of them and leans forward. They land with their feet first and continue to slide forward until the rest of their body has reached the floor.

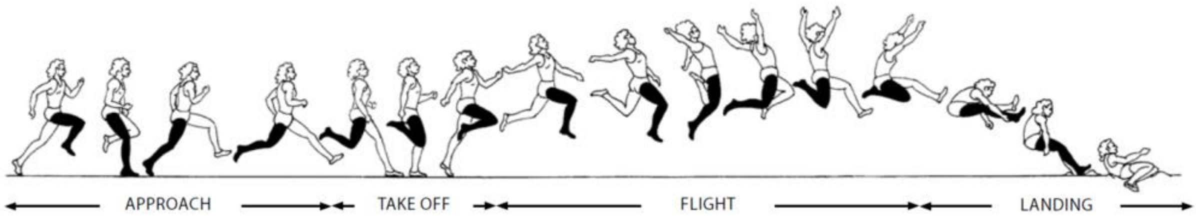

Figure 15 - Jumping Reference Image [6]

3.2 Keyframe Poses

Once I had gathered lots of reference material **(see Appendix 5.3.1)** and I had planned my animation, I started to animate my athlete. I created my animation pose-to-pose using the HumanIK [7] rig built into Maya. I used it to position my athlete into each major pose which I then keyframed **(see Appendix 5.3.2)**. One of the Disney Animation Principles is exaggeration. I used this in my animation, therefore, my poses are not perfect copies of the reference images.

The HumanIK rig allowed me to position parts of the athlete using controllers, I did not actually have to move any of the geometry manually. I used inverse kinematics to create goal-directed poses, rather than manually animating each joint down the chain. This allowed me to create the key poses of my animation very quickly. Sometimes I had to then adjust a few joints to refine the pose, but this technique was much less time consuming than using forward kinematics.

3.3 Secondary Animation

I created the secondary animation after my main animation of the athlete was complete. The secondary animation includes the folding of the jetpack and the retractable visor. Adding the secondary animation was very simple because I had already created the attributes that used set driven key to control the blendshapes. All I had to do was set keyframes on these attributes. I animated the visor to unfold as the athlete is about to start their run-up. The jetpack unfolds as the athlete is about to jump and then folds away as the athlete is about to land **(see Appendix 5.3.3)**.

I also animated the camera so before the athlete's run-up, it is zoomed in on their face. It then zooms out and pans to follow them as they run. It also tracks the athlete as if it is mounted on a rail alongside the long jump area.

4. Surface and Lighting Design

When designing the surfaces of my exosuit I had to think about the materials they would be made from. I decided that the majority of the exosuit, the struts, joints, and helmet, would be made from a plastic material. I created standard surfaces for these 3 types of object with a *metalness* value of 0.1. I decided that the helmet would be smoother than the rest of the exosuit and gave it a lower *diffuse roughness* value, this makes it glossier. I gave the straps a higher roughness value and a metalness value of 0. I decided that the jetpack would be made of metal rather than plastic, so I gave it a metalness value of 0.8 and a much lower roughness value of 0.2. I made the visor material transparent by giving it a higher *transmission weight*. **(see Appendix 5.4)**

I used an *Arnold SkyDome Light [8]* to light my scene as if it were outside. I adjusted the attributes of the *Physical Sky* to set the *elevation* and *azimuth* angle, this allows you to control the time of day and year. I set them to an elevation of 45 degrees and an azimuth of 170 degrees.

5. Appendix

5.1. Concept Design

5.1.1. Initial Design

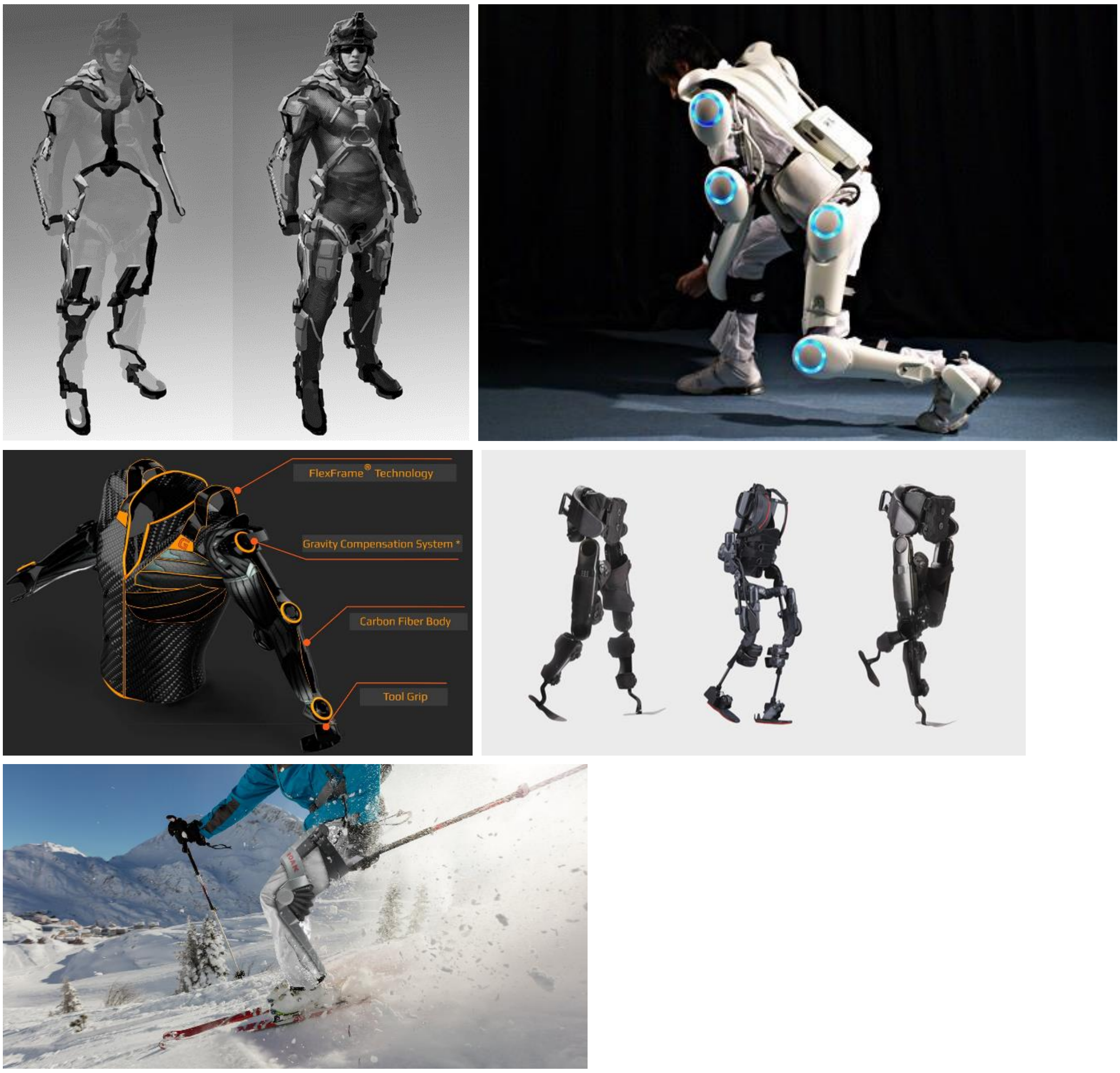

Figure 16 - Initial Design Reference Images

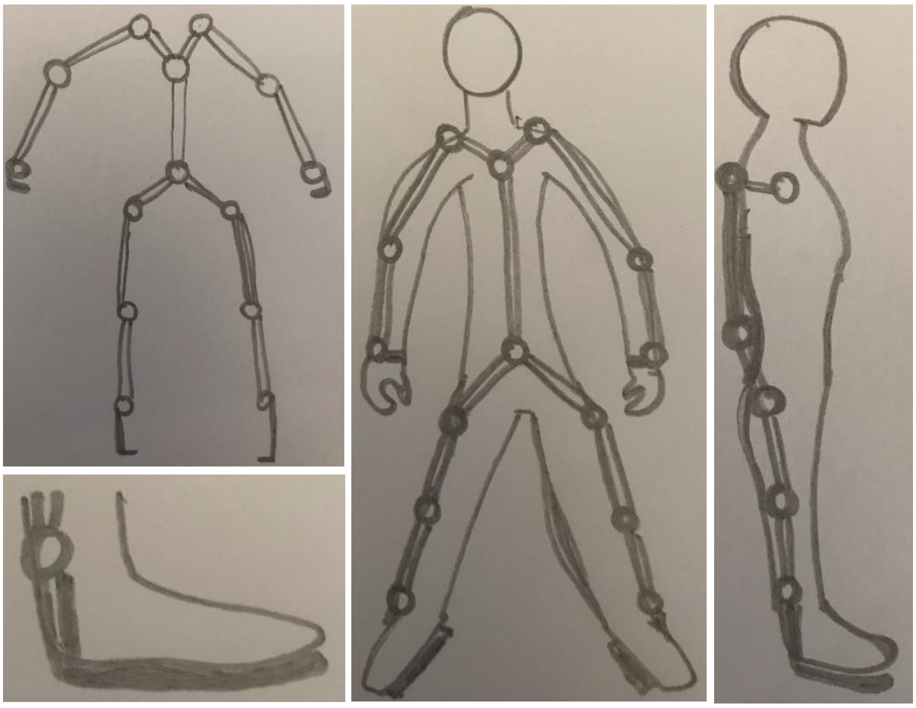

Figure 17 - Initial Design Sketches

(Top Left: Exosuit Front, Bottom Left: Exosuit Shoe, Centre: Exosuit Back, Right: Exosuit Side)

5.1.2. Second Design

Figure 18 - Second Design Reference Images

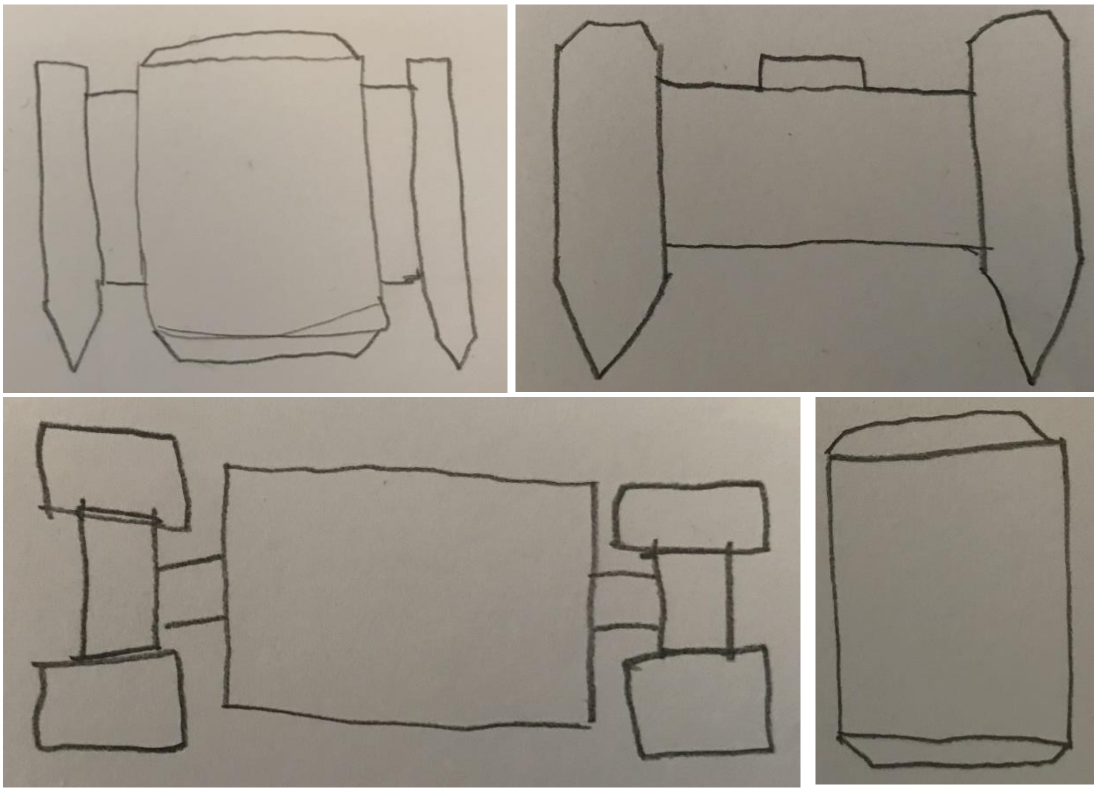

Figure 19 - Second Design Sketches

(Top Left: Jetpack Front, Top Right: Jetpack Side, Bottom Left: Jetpack Top, Bottom Right: Jetpack Folded)

5.1.3. Final Design

Figure 20 - Final Design Chest-plate Reference Images

Figure 21 - Final Design Helmet Reference Images

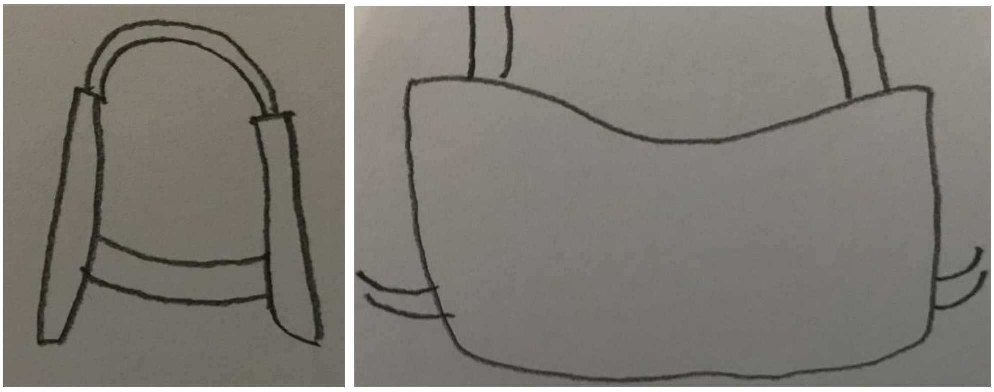

Figure 22 - Final Design Chest-plate Sketches

(Left: Chest-plate Side, Right: Chest-plate Front)

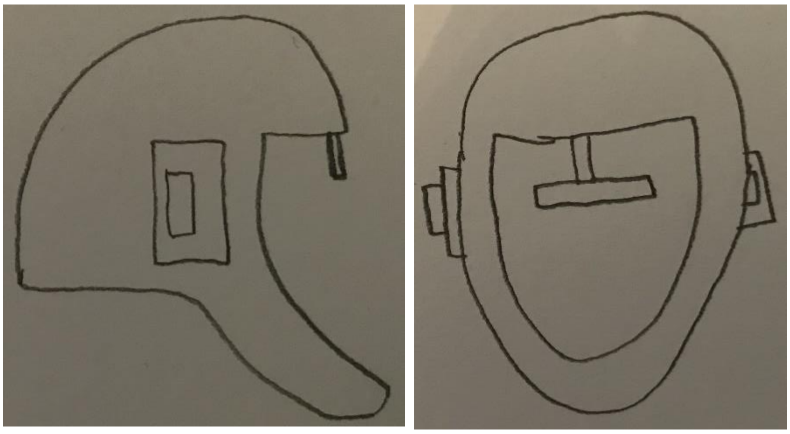

Figure 23 - Final Design Helmet Sketches

(Left: Helmet Side, Right: Helmet Front)

5.2. Geometry and Model Development

5.2.1. Base Exosuit

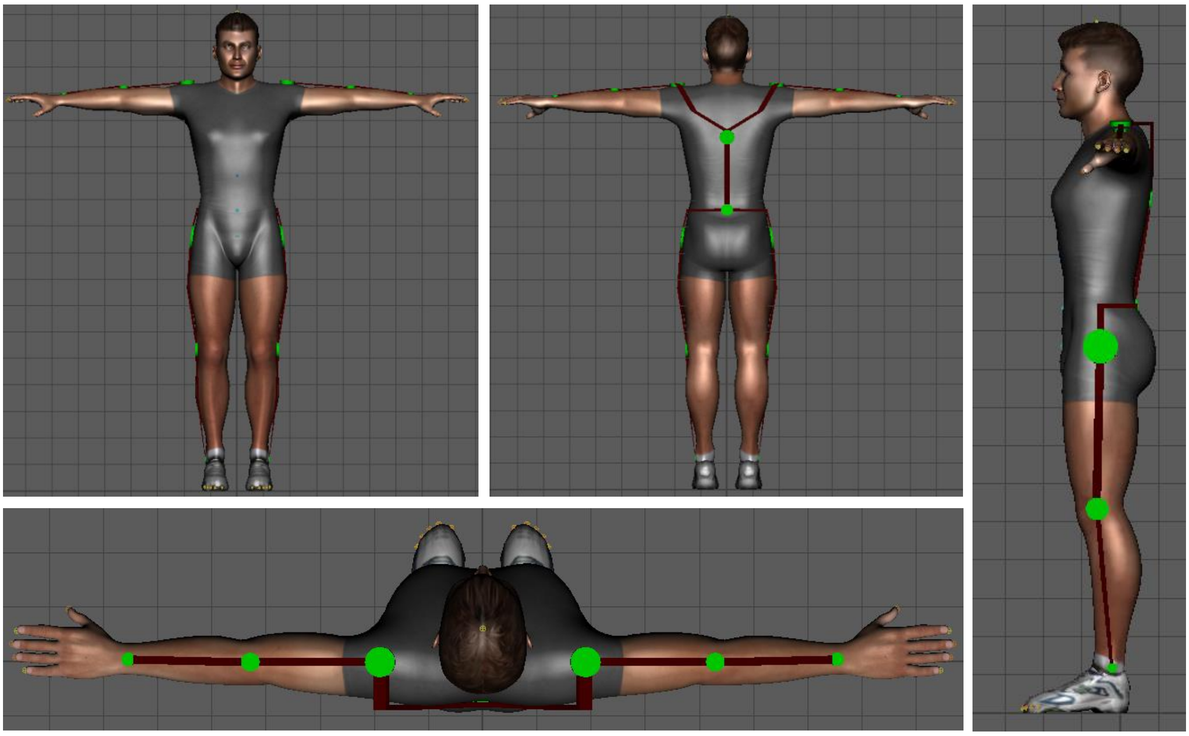

Figure 24 - Base Exosuit Orthographic Views

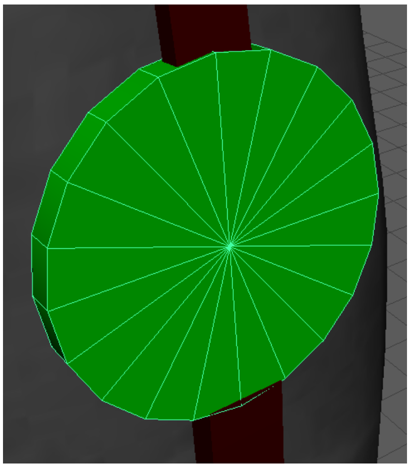

Figure 25 - Hip Joint

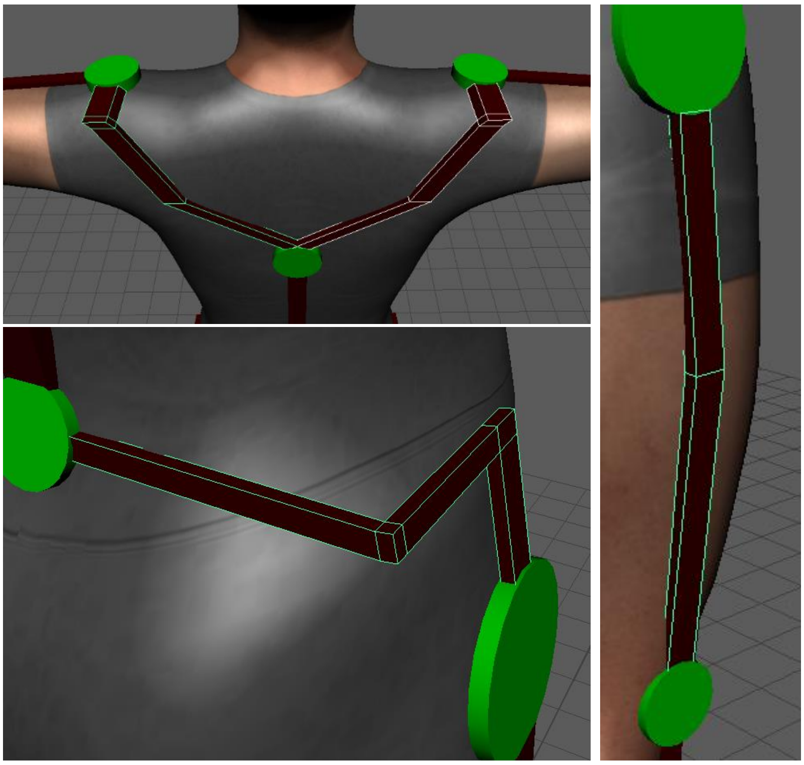

Figure 26 - Exosuit Struts

(Top Left: Shoulder Struts, Bottom Left: Hip Strut, Right: Upper Leg Strut)

5.2.2. Exosuit Shoes

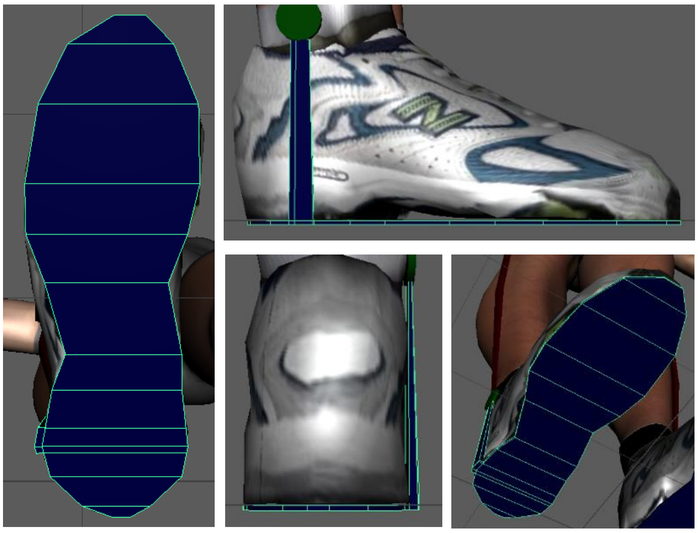

Figure 27 - Final Exosuit Shoe

5.2.3. Chest and Back Plates

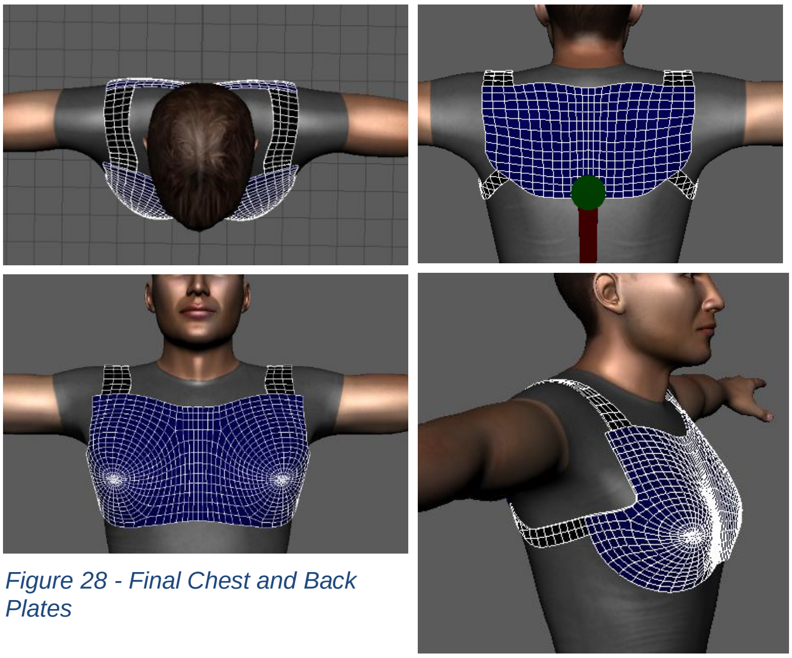

Figure 28 - Final Chest and Back Plates

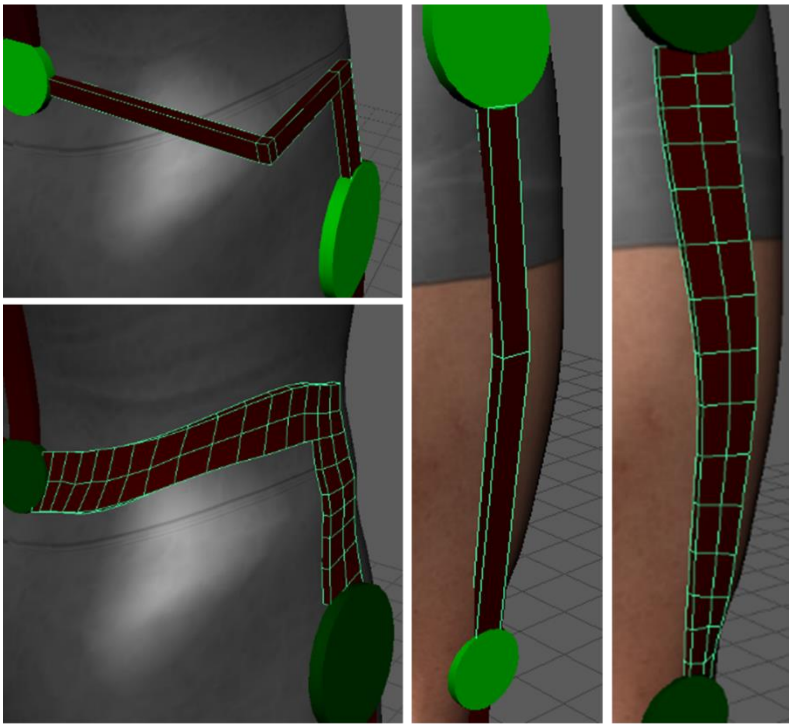

Figure 29 - Struts Before and After Changes

(Top Left: Hip strut before changes, Bottom Left: Hip strut after changes, Centre: Upper leg strut before changes, Right: Upper leg strut after changes)

5.2.4. Helmet

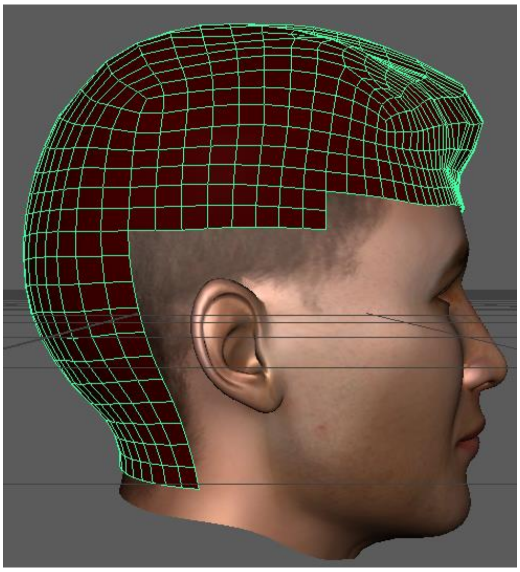

Figure 30 - Duplicated Faces From Athlete Geometry

Figure 31 - Initial Helmet Attempt Using Smoothing and Retopologizing

(Left: Uneven helmet before smoothing, Centre: Helmet after smoothing and retopologizing, Right: Unclean mesh after smoothing and retopologizing)

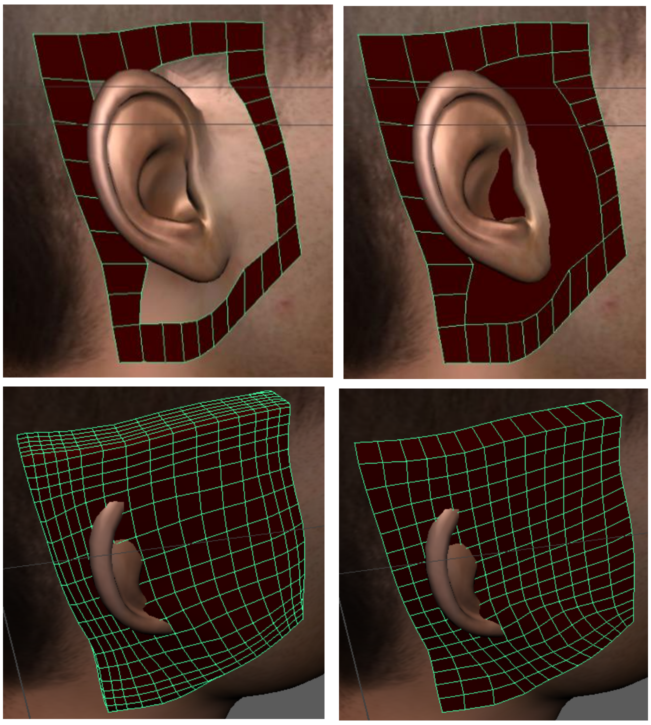

Figure 32 - Ear Cover Creation

(Top Left: Duplicated faces from athlete, Top Right: After using Fill Hole, Bottom Left: After extruding then retopologizing, Bottom Right: After retopologizing then extruding)

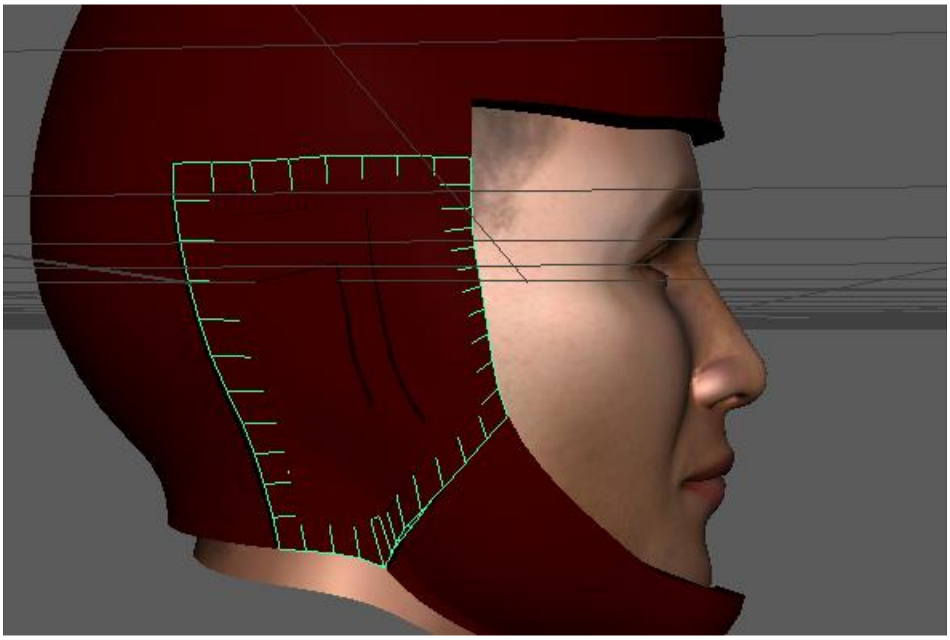

Figure 33 - Connecting Section Around the Ear

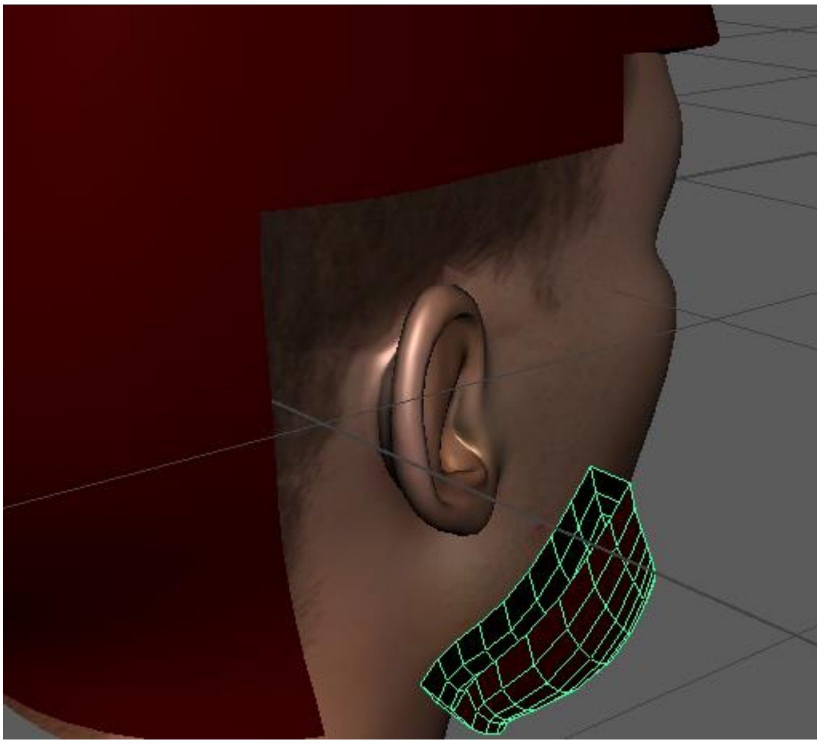

Figure 34 - Deleted Faces to Connect Sections

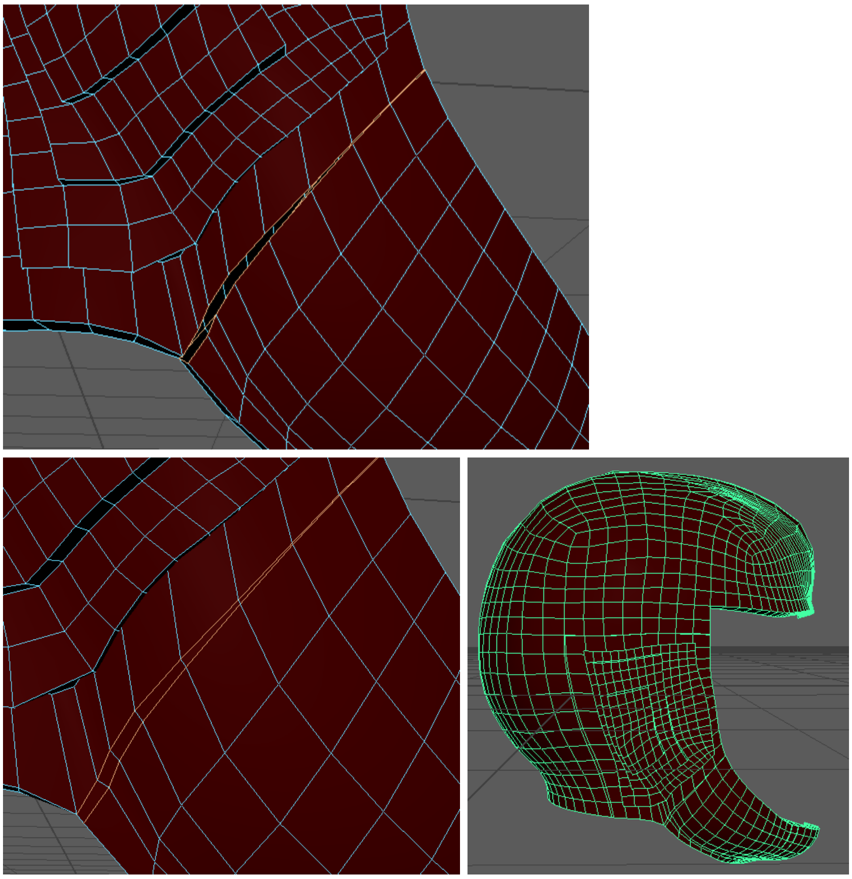

Figure 35 - Filled Gaps to Combine Helmet Sections

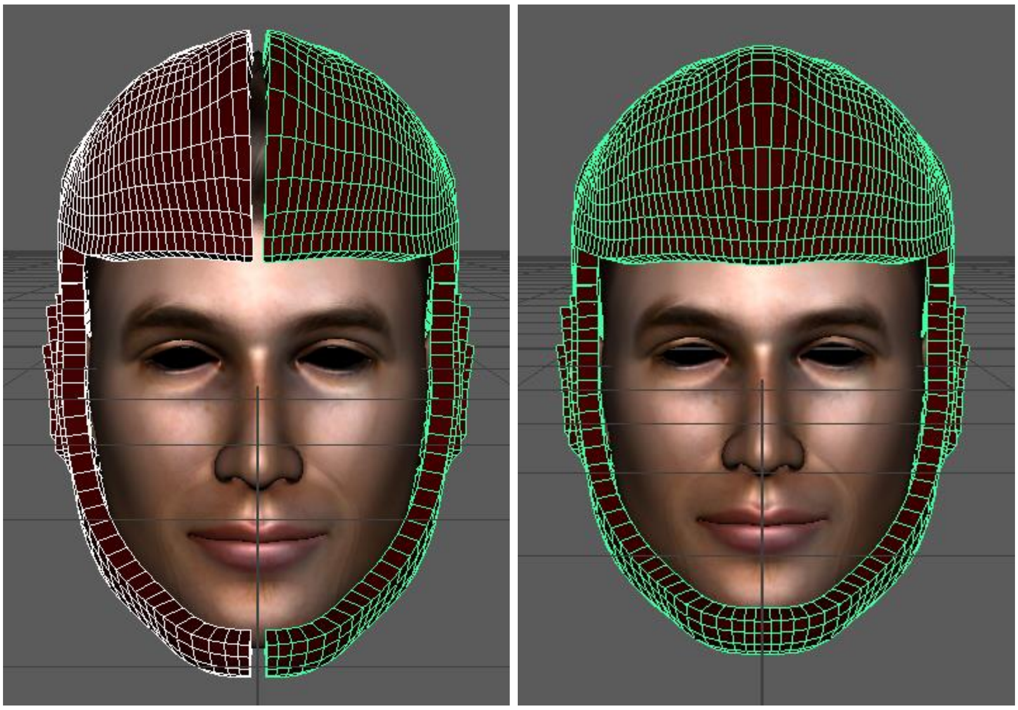

Figure 36 - Duplicated Right Side of the Helmet and Connected the Two Halves

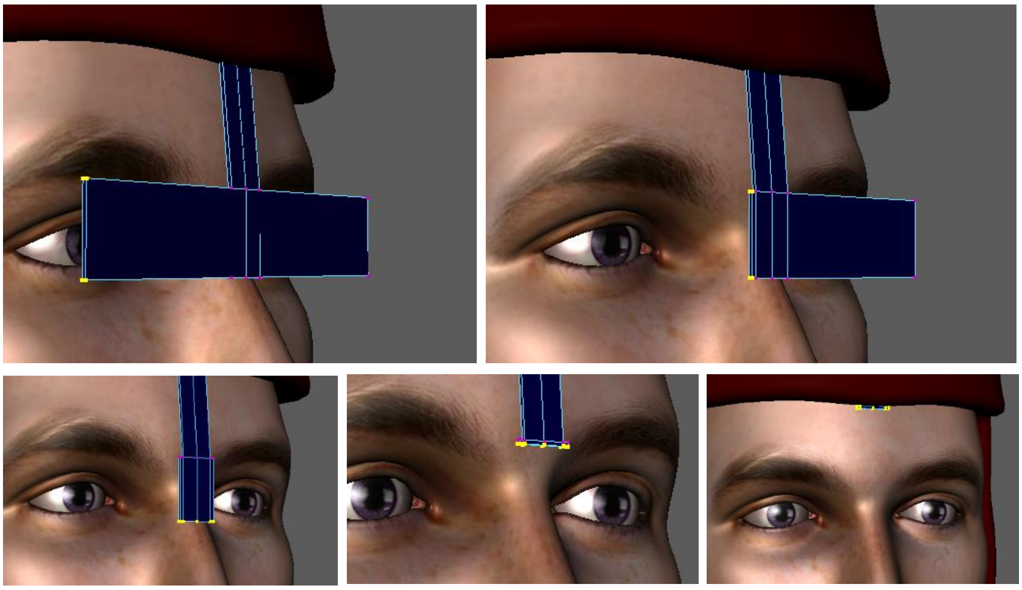

Figure 37 - Visor Stages Creation

5.2.5. Jetpack

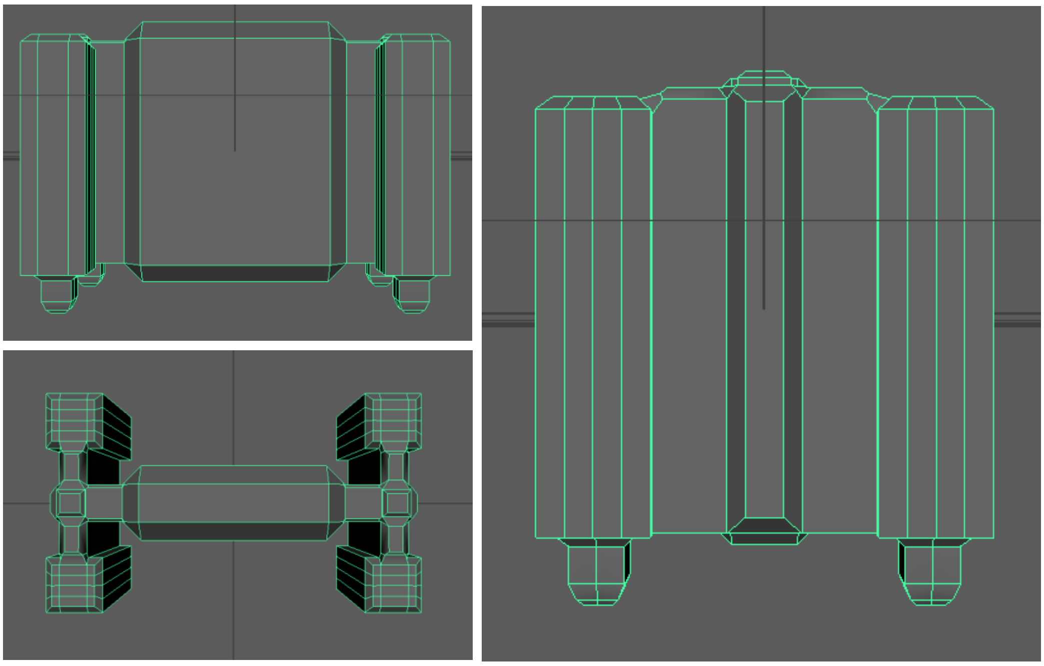

Figure 38 - Final Jetpack Orthographic Views

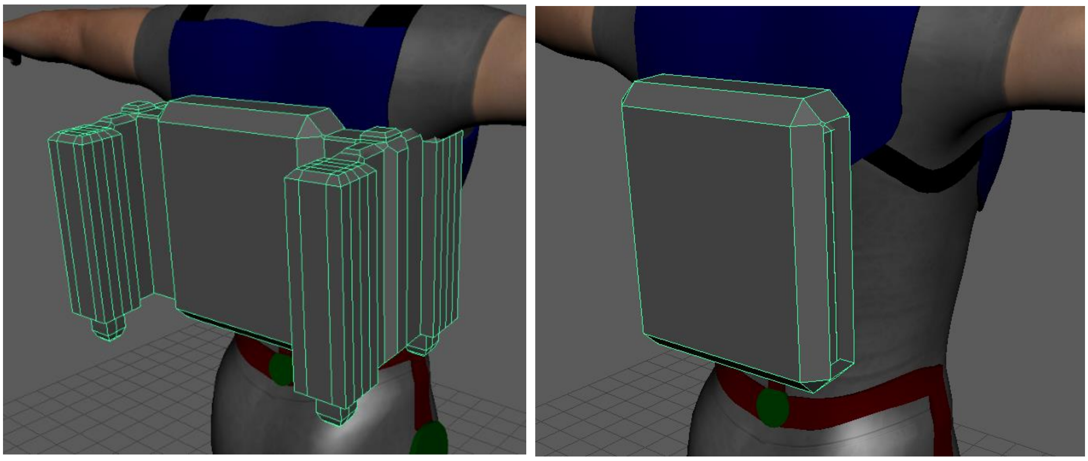

Figure 39 – Final Jetpack Unfolded and Folded

5.2.6. Parenting and Constraints

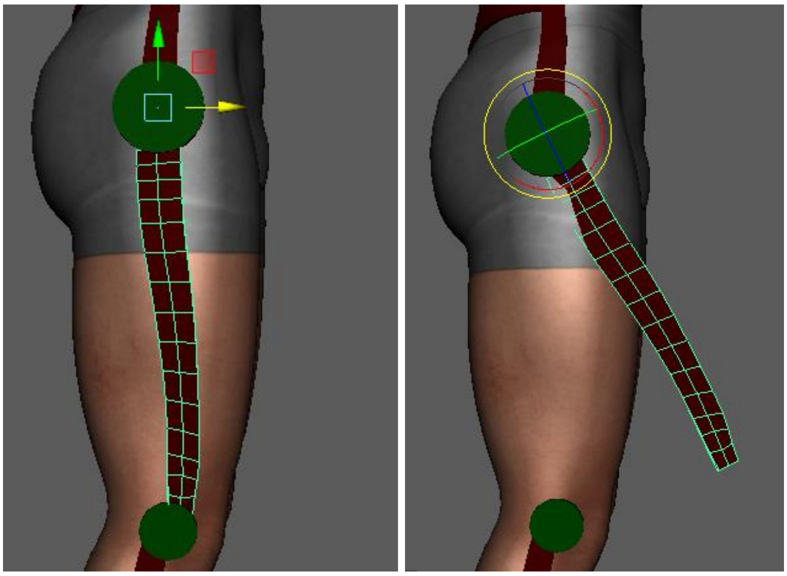

Figure 40 - Hip Pivot Position

5.3. Animation Design

5.3.1. Animation Reference

[New York 2014: Men's Long Jump | Top 3]
[Carl Lewis - Long Jump - Slow Motion]
[Juan Miguel Echevarria Long Jump 8.83m SlowMo]
[Men's Long Jump Final | Rio 2016 Replay]

5.3.2. Keyframe Poses

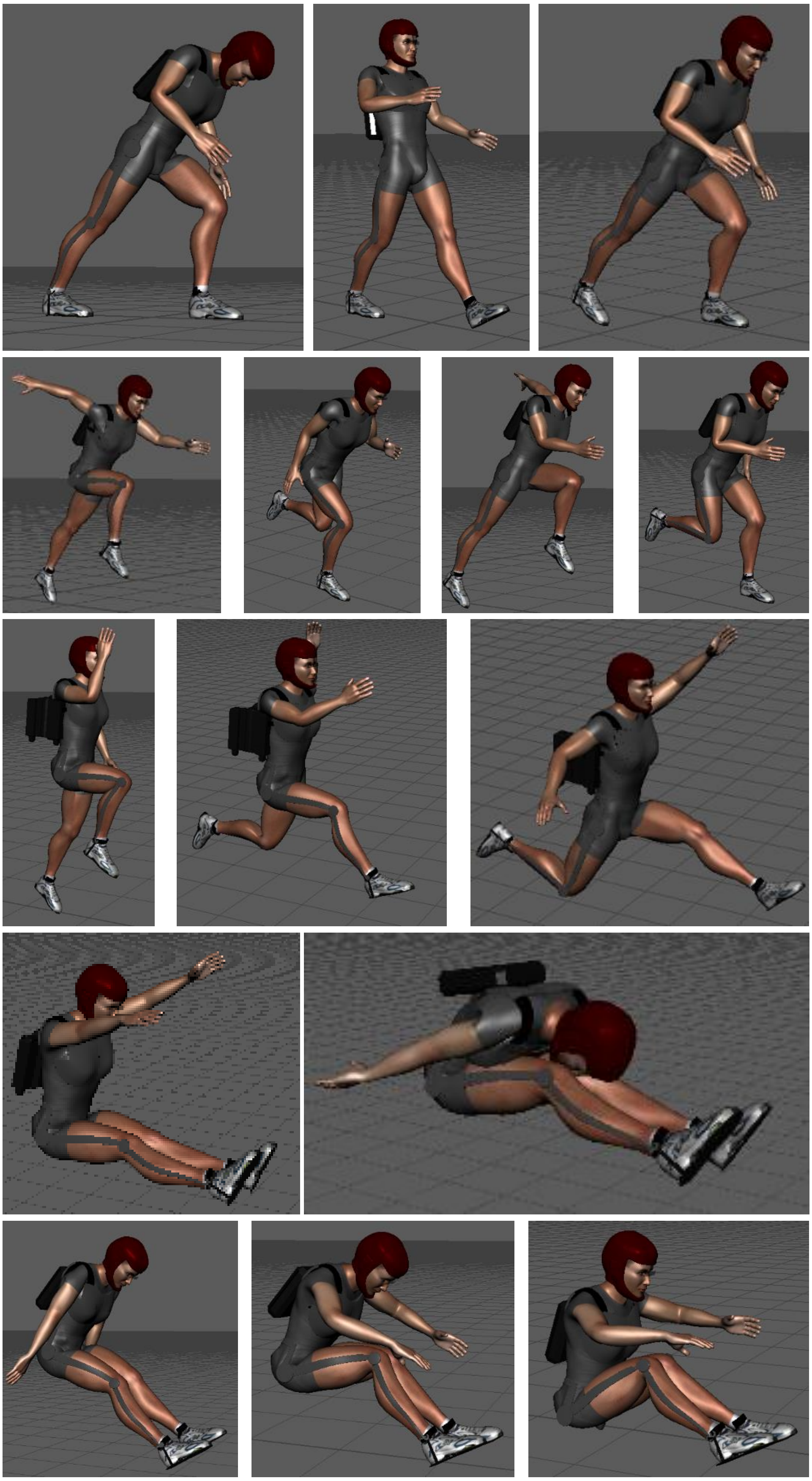

Figure 41 - Main Keyframe Poses

5.3.3. Secondary Animation

Figure 42 - Jetpack Keyframes

5.4. Surface Design

Figure 43 – Rendered Frames

6. References

[1]. Autodesk Character Generator. Available at: https://charactergenerator.autodesk.com/

[2]. Autodesk Maya. Available at: https://www.autodesk.co.uk/products/maya/overview

[3]. Disney Animation Principles. Available at:
https://en.wikipedia.org/wiki/Twelve_basic_principles_of_animation

[4]. Men’s Long Jump Final. Rio 2016 Replay. Available at:
https://www.youtube.com/watch?v=fXIbLmlUdOQ

[5]. Animation Research: Run Cycle. Available at:
https://cloytoons.wordpress.com/2015/12/06/animation-research-run-cycle/

[6]. Long Jump – The Technical Model. Available at:
https://www.aths.coach/resources/long-jump-the-technical-model

[7]. HumanIK. Available at:
https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2020/ENU/Maya-CharacterAnimation/files/GUID-EDBDA3DB-4715-40EF-9ADF-412F78BFF98E-htm.html

[8]. Arnold Renderer. Available at: https://www.arnoldrenderer.com/