Opinnäytetyö (AMK)

Tietotekniikan koulutusohjelma

Media tekniikka

2013

Jussi Lipponen

PRODUCING A 3D ANIMATED TEASER TRAILER

Case: Tale of the orbs

TURUN AMMATTIKORKEAKOULU
TURKU UNIVERSITY OF APPLIED SCIENCES

OPINNÄYTETYÖ (AMK) | TIIVISTELMÄ

TURUN AMMATTIKORKEAKOULU

Tietotekniikan koulutusohjelma | Mediatekniikka

April 2013 | 88 s.

Ohjaaja: (AMK) Keijo Leinonen

Yliopettaja FT Mika Luimula

Jussi Lipponen

PRODUCING A 3D ANIMATED TEASER TRAILER

Tämän opinnäytetyön toimeksiantajana toimi pelinkehittäjä tiimi FantasyCraft. Työn tarkoituksena oli tehdä traileri pelille, jota FantasyCraft on parhaillaan tekemässä, ja samalla tutustua ja soveltaa menetelmää missä tuotanto jaetaan esituotanto-, tuotanto- ja jälkituotantovaiheisiin.

Esituotantovaiheeseen liittyy mm. kuvakerronta. Tässä vaiheessa tuotanto suunnitellaan huolellisesti. Seuraavana on tuotantovaihe, johon liittyvät mm. 3D-mallinnus, teksturointi ja animaatio. Tämä oli työn tärkein vaihe. Siinä vaiheessa jouduttiin ratkomaan ongelmia, kuten kangas simulaattorin taipumus väreillä kun sitä käytetään hahmojen vaateisiin ja savu simulaattorin huono toimivuus suurissa mitta suhteissa. Nämä ratkaistiin rajoittamalla kankaan liike vain vaatteiden vapaasti riippuviin osin ja renderöimällä savu simulaatio erikseen pienemmän kokoisena ja sitten asettaen sen otokseen video toisto polygonina.

Viimeisenä käydään läpi jälkituotantoanto jossa pääpainona on renderöinti ja sommittelu. Näiden aiheiden lisäksi tutustutaan myös lyhyesti siihen miten animaation ääni raita kootaan eri osista kuten dialogi, ääni tehosteet ja musiikki.

Tämän opinnäytetyön tuloksena saatiin 3D animaatio traileri jonka tuotannossa jouduttiin soveltamaan kaikkea suunnittelusta ja projekti hallinasta aina edistyksellisin 3D teknikoihin.

ASIASANAT:

3D, mallintaminen, animaatio, tuotanto

BACHELOR´S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

Degree Programme in Information Technology | Digital Media

April 2013 | 88 p.

Instructor: Keijo Leinonen, B. Eng

Mika Luimula, Ph. D., Principal Lecturer

Jussi Lipponen

PRODUCING A 3D ANIMATED TEASER TRAILER

The product of this thesis was made for the game developer team FantasyCraft. The purpose was to make a trailer for the game which they are currently making, and at the same time, to explore and apply the method of splitting up the work into pre-production, production, and post-production phases.

The thesis starts with describing the preproduction phase covering topics such as storyboarding and the importance of planning. From there, it moves to the production phase where topics like 3D modeling, texturing, animation and other various steps are presented. This is the main focus of the thesis. The thesis also gives some insight into the setbacks encountered during production like for example the cloth simulators tendency to jiggle when it is used for character cloths or the smoke simulators poor visual fidelity at large scale. These were resolved by limiting the fabrics movement to the parts of the clothes that are freely suspended. The smoke issue was resolved by rendering the smoke simulation separately in a smaller size and then placing to the scene as a billboard.

Lastly, the thesis examines the post-production phase which includes rendering and compositing aspects of the production. It also briefly addresses how to utilize recorded dialog and integrate it with other audio components, such as sound effects and the score.

The final result of this thesis as expected was a completed 3D animated trailer that implemented everything from basic project management and planning to advanced 3D techniques.

KEYWORDS:

3D, modeling, animation, production

SISÄLTÖ

FIGURES

GLOSSARY

Audio Dub	The process of adding audio to a video recording without disturbing the pictures. The original audio may be replaced, or kept and combined with the new audio. (http://www.mediacollege.com/glossary/a/)
Codec	Short for compressor/decompressor. A tool which is used to reduce the size of a digital file. Can be software, hardware or a combination of both. (http://www.mediacollege.com/glossary/c/)
Crossfade	A video and/or audio transition in which one shot/clip gradually fades into the next. AKA mix or dissolve. (http://www.mediacollege.com/glossary/c/)
Depth of Field(DOF)	The zone between the nearest and furthest points at which the camera can obtain a sharp focus. (http://www.mediacollege.com/glossary/d/)
Frame Rate	The number of video or film frames displayed each second (frames per second; fps). PAL frame rate is 25 fps, NTSC is 30 fps, film is 24 fps. (http://www.mediacollege.com/glossary/f/)
Key Frame	n animation and filmmaking is a drawing that defines the starting and ending points of any smooth transition (http://en.wikipedia.org/wiki/Key_frame)
Resolution	The amount of detail in an image or signal. On a computer screen, the resolution is the number of pixels. In an analogue video signal, the resolution is the number of horizontal lines. In digital audio, the resolution is the number of samples per second. (http://www.mediacollege.com/glossary/r/)
Exposition	The is the author's providing some background information to the audience about the plot, characters' histories, setting, and theme. (http://en.wikipedia.org/wiki/Exposition_(literary_technique)#cite_ref-1)
Z-Depth	The distance of an object from the rendered camera, known in computer graphics. Z-Depth is often used to apply depth of field effects in post-production. (http://3d.about.com/od/3d-101-The-Basics/a/3d-Defined-What-Is-3d.htm)

INTRODUCTION

I will tell you, any artist that has spent enough time with a 3D animation suit will sooner or later get the urge to make something bigger and better than anything they done before. For some that thing is a still image that they tune to absolute perfection for others like me the notion of animated short is more alluring. Creating a true short animation is by no means an easy task as the need for storytelling, asset/time management, organization, acting, editing skill is increased dramatically. Indeed for someone like me who's earlier animations running times have been measured in mere seconds instead of minutes a lot of these things that I have had to worry relatively little about.

While my original intent was indeed to create an animated short for this thesis that idea was quickly scrapped in favor of an animated promotional trailer for a game. While animated shorts can share a lot in common with trailers there are also some differences especially in areas like story telling.

The trailer was to be produced for FantasyCraft's Tales of the Orbs. FantasyCraft is a group enthusiastic game developers working on their very first game and I was delighted to be able to work with such people.

OVERVIEW

In his book Animating with Blender Roland Hess talks about “Death by Natural Causes”. [1] Natural Causes is a way of describing the tendency of projects to slowly grid to a halt, fall apart and never get finished. In deed it could be argued that “natural causes” is the single biggest threat to any project and that threat increases drastically the more ambitious the final goal is. This is where preproduction, production, and postproduction come in. The terms are used to describe the time tested structure that is used in video production.

1 PREPRODUCTION PHASE

Like the name implies preproduction is something that is done before actual work on the film or in this case animation takes place. That's not to down play the importance of this stage. On the contrary it is of vital importance that this stage is executed with care as it's the bedrock on which the rest of the work rests. [1]

So what does one actually do during preproduction? There is no one right answer for that as it depends a lot on the production in question. For traditional live action movie its things like hiring the cast and film crew, selecting locations, writing the screen play etc. [2]

For a production like mine where the footage is 3D generated the focus shifts heavily towards things like concept art

1.1 Screenplay

For an animation to be more than just an extended animation test it needs to have a point, a story. Short animations are many times done in a similar fashion as novels that is to say self-contained and often comedic in nature. [1]

So what about trailers? Indeed most trailers also strive to tell a story. In the case of feature films a popular format is to condense the films story and present it in a slightly cryptic way for maximum appeal. Like in the feature film they represent they also favor the three-act structure. [3]

A rough draft of the screenplay was provided by the Fantastycraft team which feasibility was then evaluated. The content of the draft was indeed ambitious containing several characters and multiple locations. After some consideration it was decided that something had to be done to cut down the work required to realize the trailer. The solution was to substitute some of the locations and characters with hand drawn stills that would be integrated by animating parts of them as well as adding particle effects to them in order to bring more life to them.

1.2 Storyboard

Storyboards of big productions can be stunning pieces of art but that is not there primary purpose. On the contrary they don’t need to be pretty at all to serve there functioning. The storyboard is a tool, a tool that’s purpose is to make the production easier to manage. It doesn't matter if the result is just stick figures. What matters is that shots are planned in advance. Like the saying goes “if it's not on paper it does not exist”. [5]

In figure 1 the Tale of the orbs story boards is presented as t was first hand drawn.

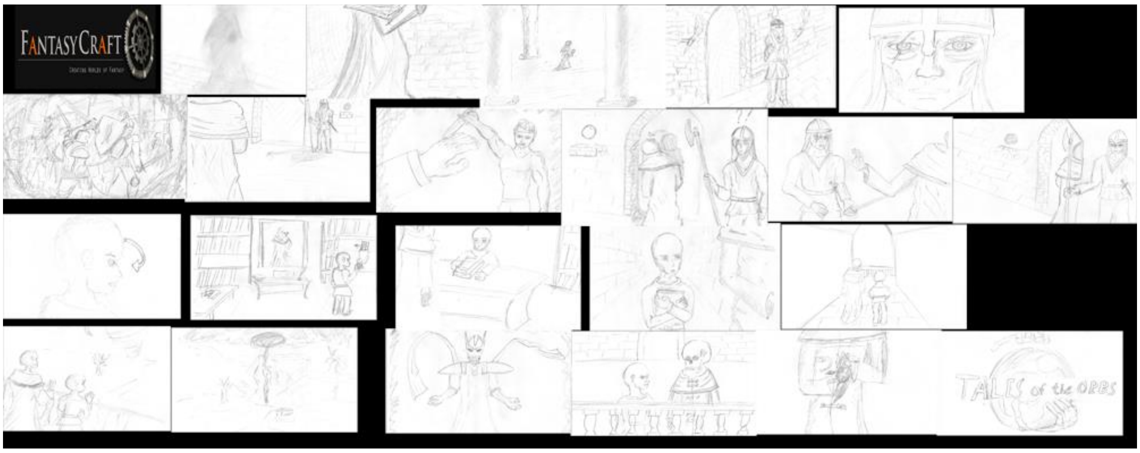


Figure 1. the tale of the orbs story board

1.3 Story-reel

The story reel is a combination of the story board and a rough soundtrack combined in a video editor. It’s a slide show that’s purpose is to give an estimate of the total running length. The rough sound track is recorded usually just by one or two people speaking the dialog and making some sound effects. Again this is not about quality it’s just a tool got evaluate the timing and running length as well as the general feel of the story. This rough sound track might be made more than once as attempts are made to close in on the wanted timings.

1.4 Character Designs

Usually some amount of concept art has been generated during the preproduction phase. This holds especially true if the story line is driven by the character as is the case in this instance. [1] Indeed as the game for which this trailer was being made has itself been in production for some time there was an excising pool of character designs that where to be used in the trailer. Therefore the initial conceptual design of characters will not be covered in this thesis and the focus will rather be on how those designs can be realized in 3D.

The first order of business when attempting to re-imagine a 2D character for 3D animation is to decide on the level of reality. Though the source art does provide a direction to work from there is often plenty of room for interpretation. Even when using the concept art as a guide there was still the possibility to go for either a highly stylized or a more photorealistic approach. Where in that range the character is to be placed depend on the action required in the story, resources and the theme. Generally the closer to photorealistic a character is the more work goes into making it. [4]

It's worth noting that attempting to make a highly photorealistic character can have a very serious pitfall. It's a phenomenon known as "the uncanny valley" where a highly realistic, but not quite perfect character appears creepy and unsettling to the viewer. While Dr. Masahiro Mori's theory of "the Uncanny Valley" is yet to be fully proven by hard science its effects do seem quite real. For example it's speculated that part of the reason why the 3D animation Final Fantasy the spirits within did so poorly at the box-office was due to the effect. While impressive for its time the movies character did not look real enough to be accepted as humans.

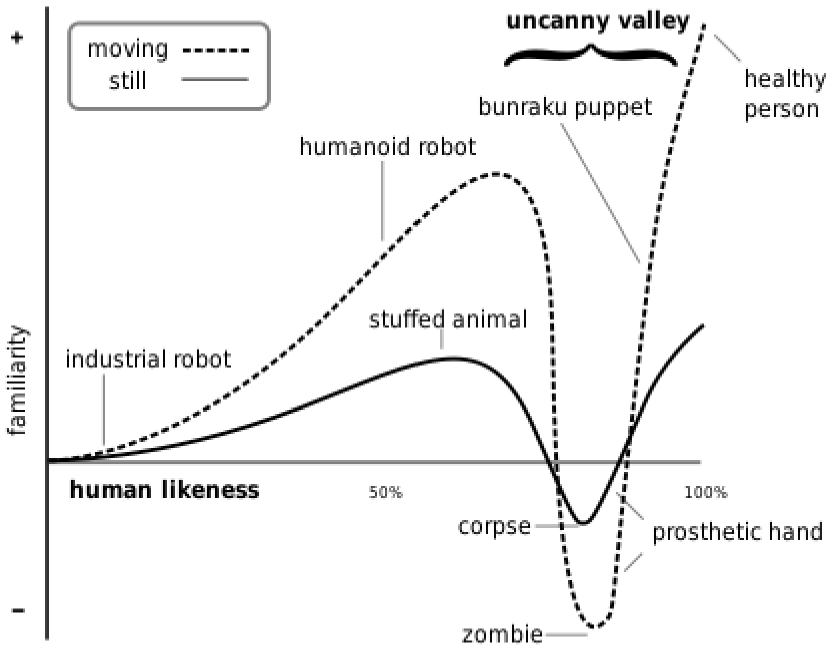


Figure 2. Mori's graph [6]

An exaggerated and more cartoonish look will keep characters away from potential pitfalls of the Uncanny Valley by staying well to the left of it on Mori's graph (Figure 2). This effectively results in 3D cartoons like the style Pixar uses in movies like The Incredibles.

With enough skill and time one can also attempt to jump the uncanny valley. While Final Fantasy failed to clear the gap the Gollum character form the Lords of the Rings faired considerably better thanks to advancements in technology. [6]

FantasyCrafts characters are not only stylized but also a sort of undead or “unliving” as they called them so in this case a little creepiness could even enhance the presences of the characters, within reason of course.

2 PRODUCTION PHASE

Creating 3D models takes numerous steps and a fair bit of time starting with modeling itself followed by UV wrapping, texturing and material settings, rigging and finishing up with the physics simulations. The steps are not always executed in that order and sometimes are even necessary to revisit already completed steps to make adjustments. In fact 3D workflows can be quite iterative

2.1 The basics of 3D

Before we dive in to the specifics of how the 3D content for this project was made let’s take a go over some of the basics of 3D graphics.

At its core 3D graphics relies on the Cartesian coordinate system that is to say it uses three axis to represent height, width and depth also known as X, Y, Z. In digital space 3D models are in fact a collection of data points that are marked out on in different coordinates in Cartesian space. [8]

In Blender you have the option to use either curves and their 3D extension surfaces or polygonal meshes.

2.1.1 NURBS surface

A Non-uniform rational B-spline, or NURBS surface is a surface composed of two or more curves in 3D space, which can be manipulated by moving handles called control points along the x, y, or z axis.

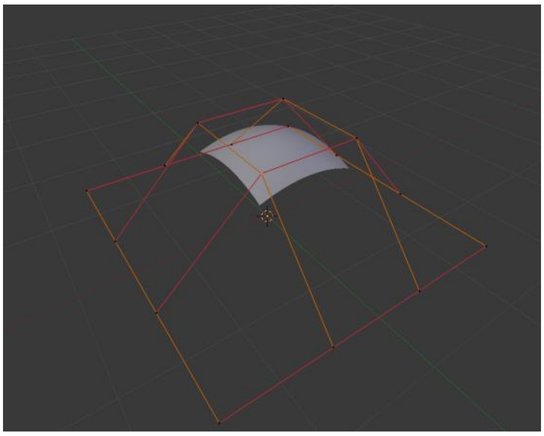


Figure 3. A NURBS surface with its control points visible.

As the software interpolates the space between curves and creates a smooth mesh between them the NURBS surfaces have the highest level of mathematical precision (See figure 3). [8]

This makes them great for CAD applications but in Blender and for 3D animation they have limited applications due to the difficulty of modeling with them in Blender and how they can't be UV unwrapped.

2.1.2 Vertices, Edges, and Faces. The polygonal model

Polygonal models or meshes as they are known in Blender are far easier to work with and thanks to the subdivision surface modifier you can achieve similar levels of smoothness as NURBS surfaces.

The parts that make up meshes are made up form vertices, edges and faces.

- Vertices, shown in Figure 4, are the most basic part of the mesh as they represent the Cartesian coordinates of the mesh. [7]
- Edges, shown in Figure 5, connect two vertices, forming the side of a face thus determining the layout of a mesh. [7]
- Faces, shown in Figure 6, are composed of three or more edges, forming either triangles quads or in the case of more than five edges N-gons. Faces in general are referred to as polygons (polys for short) and are what make up the surface of a mesh. [7]

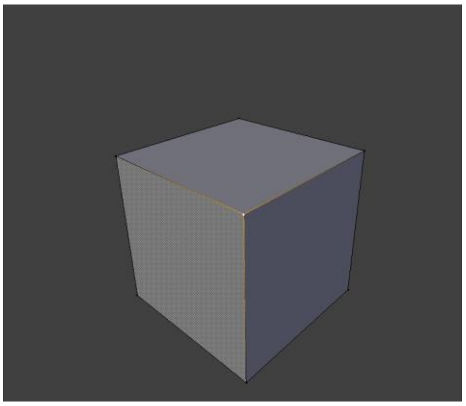


Figure 4. a highlighted vertex.

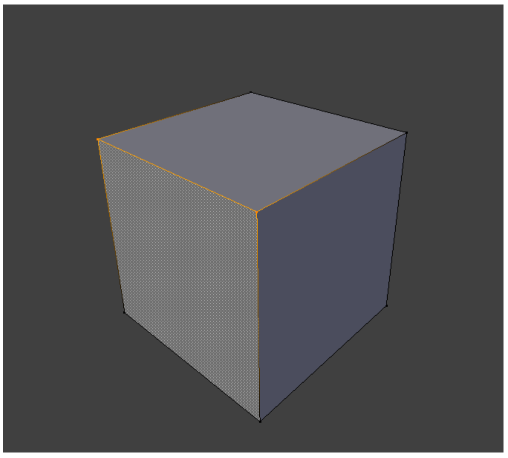


Figure 5. a highlighted edge.

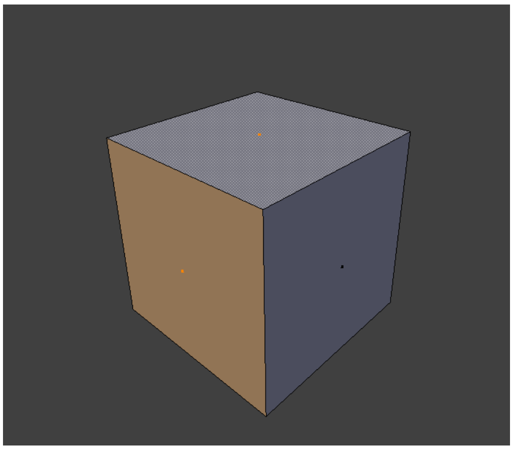


Figure 6. a highlighted face.

2.1.3 Modeling basics

There are a few different modeling techniques. The main ones that are usually used in blender are polygonal and box(primitive) –modeling as well as sculpting that we will cover later. When using box modeling you start with a simple primitive shape like a cube, sphere or tube and then proceed to extrude and deform it to make the shape you want slowly adding detail/complexity. [7]

When you do what's known as polygonal modeling you simply start with a rectangle that you then extrude and shape to the desired form. The main advantage in using polygonal modeling is that it is easier to make loops. Loops are generally regarded as one of the corner stones of character modeling as they greatly affect the way a mesh deforms. For humanoid faces this means quad faces that follow the contours and underlying musculature of a character. When done right the end result is significantly more realistic deformations when animated than would be created with pattern of faces. This is where we get it to the subject of topology. In the context of 3D it refers to the polygonal flow of the model. The same shape can be achieved with good and bad topology. What constitutes a good and bad topology? As mentioned earlier topology has an effect on how a mesh deforms but that's not all it affects. Even when the models are not being deformed one still needs to worry about how they will react to subdivision-surfaces better known as sub surfacing [1].

So what does sub surfacing do? Quite simply it adds more polygons and in the process smoothing out the model as seen illustrated with Suzanne the monkey head in figure 7.

Quads and triangles react differently to sub surfacing and often triangles give undesired results. While sub surfacing is great for organic shapes it can also result in an over smoothed look. This can be combated by placing additional edge loops where sharper shapes are wanted. Blender also supports creasing and mark sharp that can be used for different types of edges but more on those later. [1]

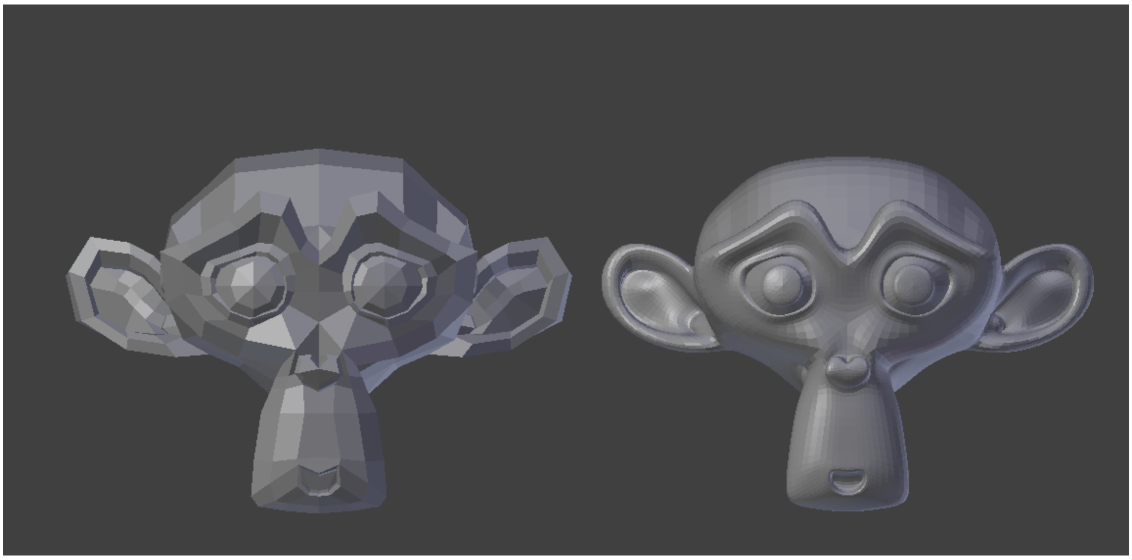


Figure 7. Suzanne’s base mesh on the left and with two times sub surfacing applied on the right

2.1.4 Material basics

A material defines the qualities of the substance that an object is made of. At its simplest it just defines the color of the surface. More often the substance is represented by its surface qualities (color, shininess, reflectance, etc.) but it can also exhibit more complicated effects such as transparency, diffraction and sub-surface scattering. These properties are often referred to as shaders.

The basic (un-textured) Blender material is uniform across each face of an object. However different faces of the object may use different materials.

As this is rather in convenient the use of textures is almost always preferred for more complicated surfaces. [10]

Texture mapping as it’s called is a method to add detail to surfaces by projecting images and patterns onto those surfaces. The projected images and patterns can be set to affect not only color, but also specularity (shininess), reflection, transparency, and can even create the illusion of 3-dimensional depth by the use of normal or bump -maps as seen in figure 8. Most often, the images and patterns are projected during render time, but texture mapping is also used to sculpt, paint and deform objects. [10]

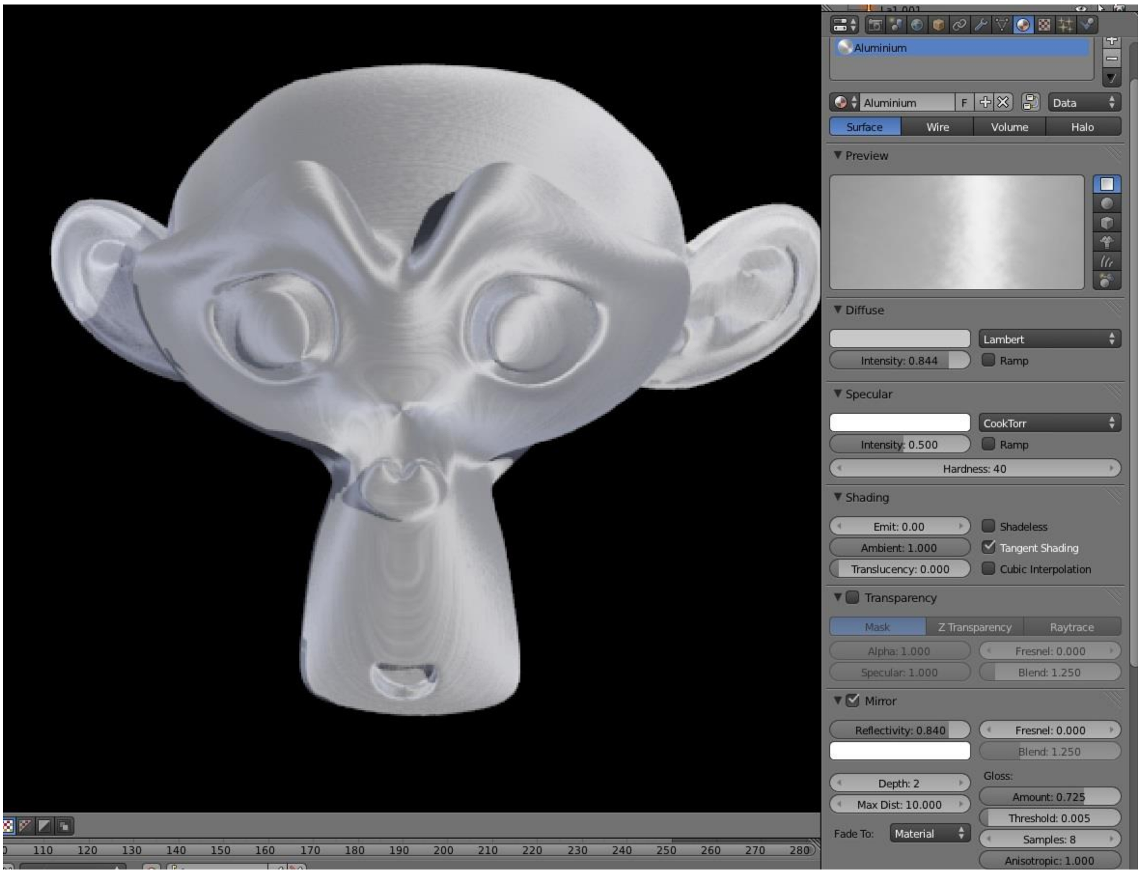


Figure 8. an aluminum material with a bump map giving brushed effect

2.2 Material options in Blender

The main material options in Blender are diffuse, specular, transparency, reflections and Subsurface Scattering.

2.2.1 Diffuse Shaders

A diffuse shader controls the general color of a material when light shines on it. Most of shaders that are designed to mimic reality and give a smooth falloff from bright to dark from the point of the strongest illumination to the shadowed areas but there are also some that do more specialized effects. [10]

All of diffuse shaders have the following options:

- Color: Select the base diffuse color of the material
- Intensity: The shader's brightness, or to put it more accurately, the amount of incident light energy that is actually diffusely reflected towards the camera.
- Ramp: Allows you to set a range of colors for Material, and define how the range will vary over a surface.

Some of the shaders have additional settings for more fine control.

2.2.2 Specular shaders

Specular shaders create the bright highlights that one would see on a glossy surface, mimicking the reflection of light sources not to be confused with actual reflections. Unlike diffuse shading, specular reflection is viewpoint dependent meaning the hot spot will be rendered differently depending on where you are looking from. Like the diffuse shader the specular also has the color, intensity and ramp settings available for all shaders and also has some with more potisons [10]

2.2.3 Bump and Normal Maps

Both Normal- and Bump Maps are used for the same purpose: to fake additional detail that isn’t actually present in the 3D geometry. This is done by modifying the normal angles, which influences how a pixels are shaded. Because only shading is being modified no shadows can be cast from the fake geometry. Another limitation is that if the camera angle is too close to parallel with the surface the illusion is broken.

While the terms Normal map and Bump map can often be used interchangeably in the contexts of Blender they mean slightly different things.

Bump maps are the simpler version and only work of intensity and only fake height. Because of this they height can be sampled form any RGB-texture though grayscale pictures are often preferred.

Normal maps are a higher quality option and work off direction. The three axes are represented by the three color channels.

- Red maps from (0-255) to X (-1.0 - 1.0)
- Green maps from (0-255) to Y (-1.0 - 1.0)
- Blue maps from (0-255) to Z (0.0 - 1.0)

The resulting textures often look something like what's seen in figure 9.

Figure 9. how a typical normal map is colored

Because of this they can really be painted by hand thy way bump maps can and instead need to be generated in some way either by baking them out from actual geometry or by using software designed to interpolate RGB images in to normal maps. [10]

2.2.4 Transparency

Transparency in blender comes in three flavors: mask, Z Buffer and ray-traced - transparency.

Z Buffer

This option uses the alpha buffer for transparent faces. The alpha value of each pixel determines the mix of the basic color of the material, and the color of the pixel determined from the objects/background behind it. To put it simply this option makes the material transparent without any fancy refractions. No very realistic but has more than its fair share of uses. [10]

Ray-traced Transparency

Like the name suggest this option is calculated using ray-tracing and as result is much slower but does have the nice refractions we expect to see in real transparent materials.

The way ray-tracing works is a ray is sent from the camera that then travels through the scene until it encounters an object. If the first object hit by the ray is non-transparent, then the ray takes the color of the object. If however the material has ray-traced transparency enabled the ray can be deflected from its course according to the Index of Refraction (IOR) of the material. In addition to the IOR there are additional options that become available for more advanced effects.

The complete list of options as it appears in the Blender 2.6 documentation:

IOR

Index of Refraction. Sets how much a ray traveling through the material will be refracted, hence producing a distorted image of its background. See IOR values for Common Materialsbelow.

Filter

Amount of filtering for transparent ray trace. The higher this value, the more the base color of the material will show. The material will still be transparent but it will start to take on the color of the material. Disabled (0.0) by default.

Falloff

How fast light is absorbed as it passes through the material. Gives 'depth' and 'thickness' to glass.

Limit

Materials thicker than this are not transparent. This is used to control the threshold after which the filter color starts to come into play.

Depth

Sets the maximum number of transparent surfaces a single ray can travel through. There is no typical value. Transparent objects outside the *Depth* range will be rendered pitch black if viewed through the transparent object that the *Depth* is set for. In other words, of you notice black areas on the surface of a transparent object, the solution is probably to increase its *Depth* value (this is a common issue with raytracing transparent objects). You may also need to turn on transparent shadows on the background object.

Gloss

Settings for the glossiness of the material.

Amount

The clarity of the refraction. Set this to something lower than zero to get a blurry refraction.

Threshold

Threshold for adaptive sampling. If a sample contributes less than this amount (as percentage), sampling is stopped.

Samples

Number of cone samples averaged for blurry refraction.

[10]

Mask

The third and final option is mask. This is not so much a transparency as it is a compositing tool as instead of making objects transparent/translucent, it hides the geometry behind it. [10]

3.2.3 Mirror Reflections

Mirror reflections are an exclusively ray-traced option. As the name states it gives materials a mirror like quality as it reflects the rays. As using two or more mirror could lead to rays bouncing back and forth the depth value is of outmost importance as it limits the maximum number of bounces allowed for a single ray thus sawing valuable rendering time.

Also like ray- traced transparency mirror reflections has a number of additional settings that dictate how the mirror surface looks.

The complete list looks like this as it appear in Blender 2.6 documentation:

Reflectivity

Sets the amount of reflectiveness of the object. Use a value of 1.0 if you need a perfect mirror; or set it to 0.0 if you don't want any reflection.

Color swatch

Color of mirrored reflection

By default, an almost perfectly reflective material like chrome, or a mirror object, will reflect the exact colors of its surrounding. But some other equally reflective materials tint the reflections with their own color. This is the case for well polished copper and gold, for example. In order to replicate this within Blender, you have to set the Mirror Color accordingly. To set a mirror color, simply click the color swatch in the mirror panel and select a color.

Fresnel

Sets the power of the Fresnel effect. The Fresnel effect controls how reflective the material is, depending on the angle between the surface normal and the viewing direction. Typically, the larger the angle, the more reflective a material becomes (this generally occurs on the outline of objects).

Blend

A controlling factor to adjust how the blending happens between the reflective and non-reflective areas.

Depth

Maximum allowed number of light inter-reflections. If your scene contains many reflective objects and/or if the camera zooms in on such a reflective object, you will need to increase this value if you want to see surrounding reflections in the reflection of the reflected object (!). In this case, a Depth of 4 or 5 is typically a good value.

Max Dist

Maximum distance of reflected rays away from camera (Z-Depth) in Blender units. Reflections further than this range fade out to reduce compute time.

Fade to

The color that rays with no intersection within the *Max Distance* take. *Material* color can be best for indoor scenes, *Sky* color (World settings) for outdoor scenes.

Gloss

In paint, a high-gloss finish is very smooth and shiny. A flat, or low gloss disperses the light and gives a very blurry reflection. Also, uneven or waxed-but-grainy surfaces (such as car paint) are not perfect and therefore slightly need a Gloss < 1.0. In the example to the right, the left mirror has a Gloss of 0.98, the middle is Gloss = 1.0, and the right one has Gloss of 0.90. Use this setting to make a realistic reflection, all the way up to a completely foggy mirror. You can also use this value to mimic depth of field in mirrors.

Amount

The shininess of the reflection. Values < 1.0 give diffuse, blurry reflections and activate the settings below.

Threshold

Threshold for adaptive sampling. If a sampling contributes less than this amount (as percentage), sampling is stopped. Raising the threshold will make the adaptive sampler skip more often, however the reflections could become noisier.

Samples

Number of cone samples averaged for blurry reflection. More samples will give a smoother result, but will also increase render time.

Anisotropic

The shape of the reflection, from 0.0 (circular) to 1.0 (fully stretched along the tangent). If the *Tangent Shading* is on, Blender automatically renders blurry reflections as anisotropic reflections.

When Tangent is switched on, the *Anisotropic* slider controls the strength of this anisotropic reflection, with a range of 1.0 (default) being fully anisotropic and 0.0 being fully circular, as is when tangent shading on the material is switched off. Anisotropic raytraced reflection uses the same tangent vectors as for tangent shading, so you can modify the angle and layout the same way, with the auto-generated tangents, or based on the mesh's UV co-ordinates.

[10]

2.2.5 Subsurface Scattering

Unlike the previous two options Subsurface Scattering (referred to as SSS) does not use ray- tracing instead relying on a separate rendering pass that is somewhat similar to a regular diffuse render. Unfortunately the said render pass hast to be done separately for each material with SSS so having many object with SSS materials will start to add up to a considerable amount of added rendering time.

Many times this is a price that has to be paid as many organic and some inorganic skins are not totally opaque right at the surface and instead some light also penetrates the skin surface, and scatters around inside, taking on the color of the insides and emerging back out to blend with the surface reflection. And this effect cannot be achieved without SSS.

2.3 3D Characters creation

Now that we have covered the basic concept of 3D models and materials in Blender lest dive strait in to the deep and start looking at how the characters for the trailer where made.

2.3.1 Modeling: The First

I decided to start with modeling the character “The First” as the Fantasycraft team had named him. So let’s see how concept was turned into an animated 3D character as seen in figure 10.

Figure 10. The First concept art(left) and in 3D rendered(right)

As the above mentioned concept art was the only thing provide for me I started by looking up some reference pictures of a human skull and setting them up in Blender. I choose to go with polygonal modeling. With the help of a front and side reference picture I modeled out first the eye socked area followed by the cheek, nose area and finally the complete skull. The jaw was made in much the same way while the teeth where individually box modeled. Many things in blender revolve around modifiers. We have already seen one of them in action: the sub surface modifier.

In figure 11 we can see it and the mirror modifier providing the right side of the mesh and how the over laying base mesh is being subdivide into a smother shape. We also see how enabling smooth shading hides the remaining visible quads.

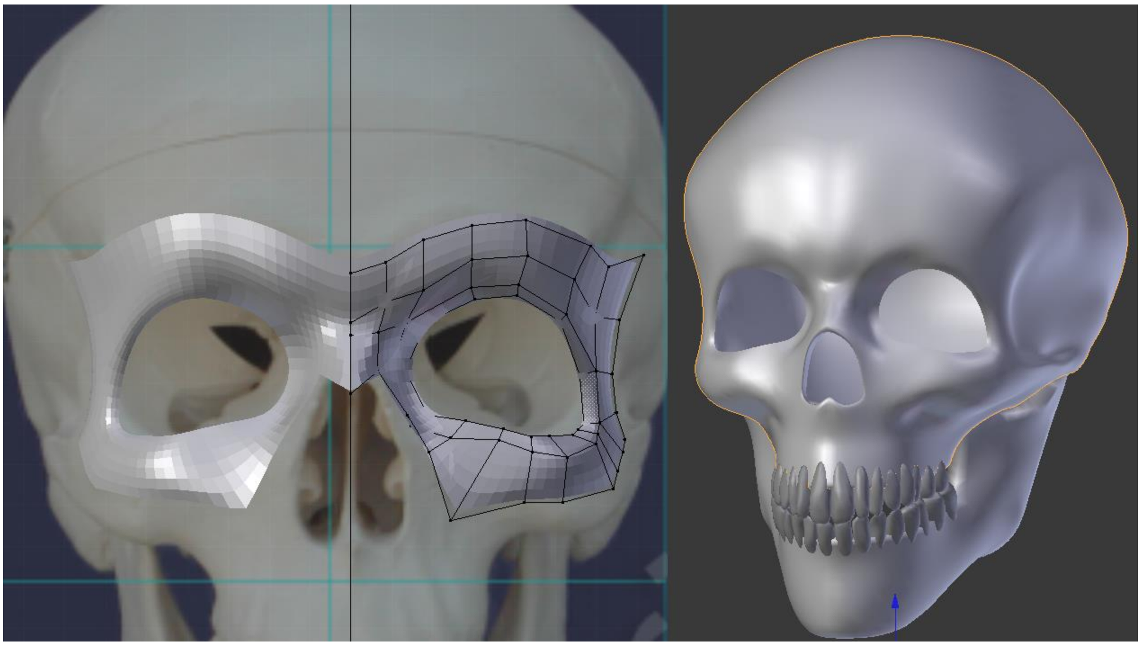


Figure 11. the initial edge loops around the eye (left) the smoothed out skull (right)

But this isn’t precisely what was wanted as we were going for a more stylized feel. A regular human skull just would not do. So after adding some more geometry in the nose and eye sockets it was time to do some sculpting.

Sculpting in Blender tries to mimic how real life clay sculpting works. You use brushes to push, pull, and nudge instead of dealing with individual elements (vertices, edges, and faces), In other words, instead of selecting a group of vertices, Sculpt Mode automatically selects vertices based on where the brush is, and moves them accordingly. Sculpting can be combined with the Multiresolution modifier so. By replacing the Sub surface modifier with the Multiresolution modifier you can now sculpt the high resolution mesh without permanently modifying the base mesh. This is a very powerful approach as it gives you the ability toggle the different levels of detail later when rendering. [10]

After the sculpting the skull had reached its final shape as seen in figure 12. In addition to the skull itself the models that would be used to make the glowing eyes effect are also seen in the picture.

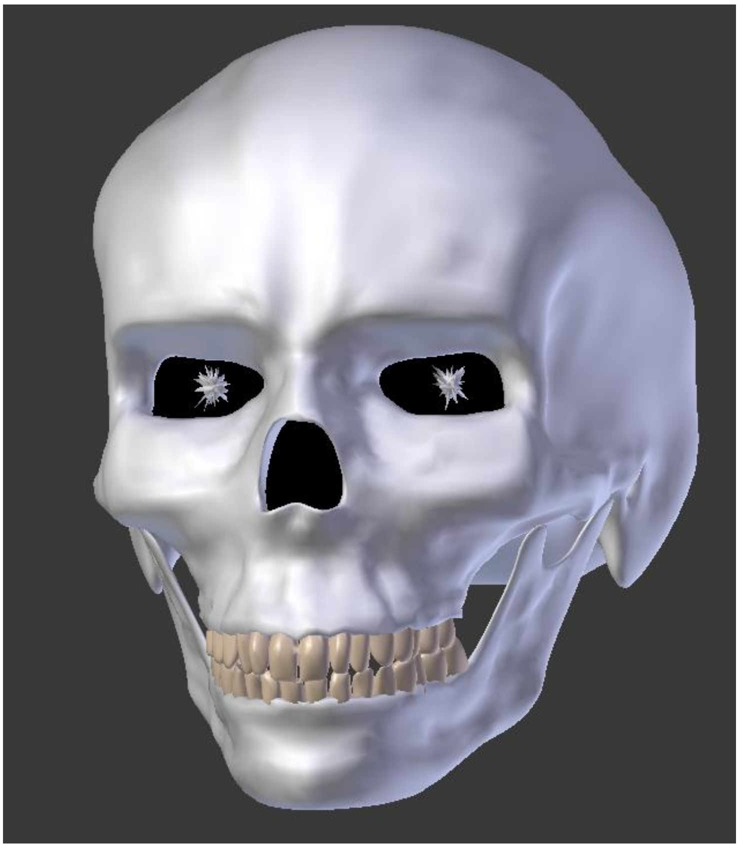


Figure 12. the fully sculpted skull of The First

After the skull was completed the hands where next. Rather unexpectedly a skeletal hand is actually easy to model. Basic box modeling got the job done fairly quickly as the bones in the hand don’t need to be very detailed. Unlike with regular human hands there was not much need to worry about the edge loops of things like knuckles or finger nails. With sub surfacing the rather simple mesh provides good enough results for this projects needs as seen in figure 13.

Figure 13. the mesh on the left and the sub surfed models on the right

The spine, ornaments, boots and bag where modeled in much the same way. For the little chain connecting the bag to the belt I used a Blender feature known as Duplication Faces or DupliFaces as it’s often shortened to. What it does is the replicate an object on each face of a parent object. [10] In this instance in meant taking the chain link model and using a subdivided plain as its parent as seen in figure 14. Only the duplicates are rendered so neither the plain nor the reference link will be visible in the final image.

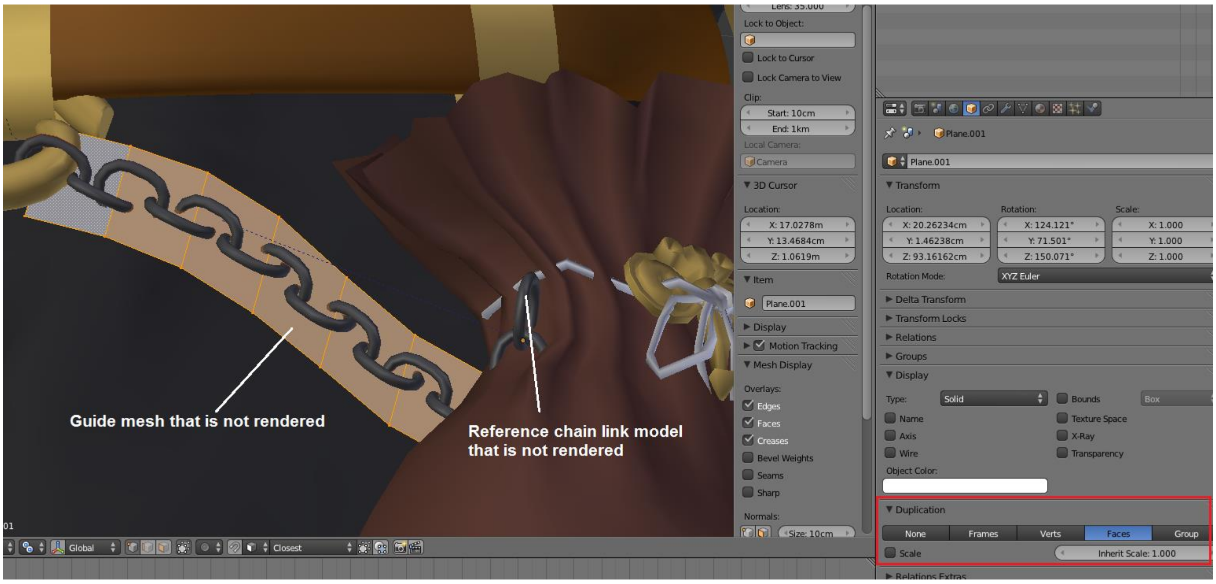


Figure 14. the guide mesh and reference chain-link

A big part of the character is the menacing black robe and this would prove to be a recurring problem area trough out the production. As the intention was to use the cloth simulation on the robe it would have to be made on a rather strict polygon budget as well as avoided having vertices positioned very close to one another as that would lead to simulation issues later. Another requirement for good simulation is that the mesh retains even as possible polygon spacing.

Due to these constraints the idea of having the gold band raised up had to be scraped. I later realized that baking normal maps or using Multi-resolution sculpting could have been used to solve this problem. What was used however was displacement mapping.

What displacement mapping does is allows the position of vertices to be manipulated by a texture. This can be used for a range of effects but in the case of the robe the goal was to make it look wrinkly as the cloth simulation could not provide the smaller details. The texture used was a procedural (this means its generated internally by an algorithm) texture called a Voronoi Crackle (see figure 15).

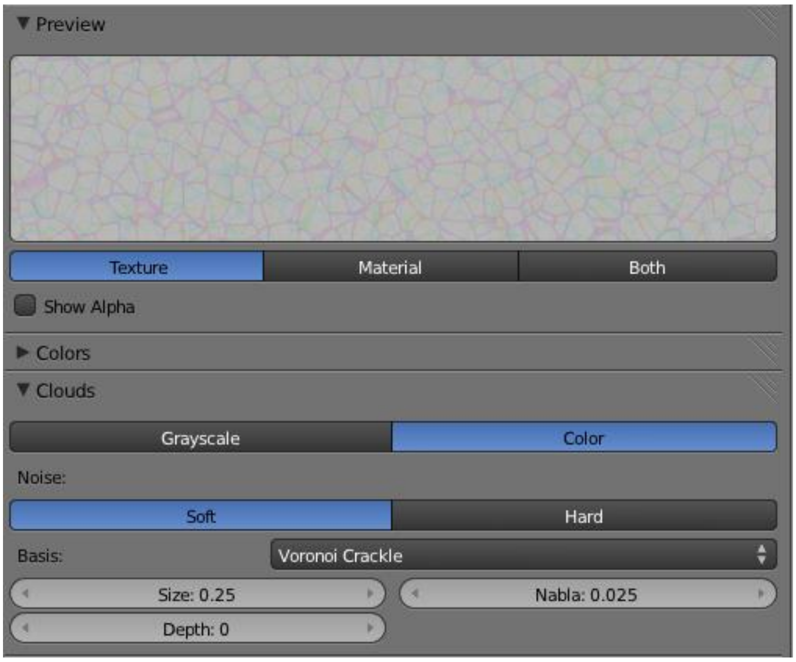


Figure 15. a procedural Voronoi Crackle texture and its settings

The displacement modifier was added after subsurface modifier so it would affect the smoothed mesh rater then the base mesh. While the results are not perfect the displacement does make the overly smooth mesh look more realistic as seen in figure 16.

Figure 16. a comparison of the un-displaced model on the left and the displaced model on the right

2.3.2 Materials: The First

Texturing is every bit as challenging as modeling and good textures can make even poor models seam better than they really are.

Blender supports a wide range of ways to map a texture to a model but the one that is used the most is UV mapping and is the most flexible way of mapping a 2D texture over a 3D object. When you do this you take the three-dimensional (X,Y & Z) mesh and unwrap it to a flat two-dimensional plain. Rather the call the axis of this new plain X and Y they are called U and V hence the name UV map.

As the skull is a continuous surface it needs to have a seam (the red line in figure 17) where the UV map can be split open and laid flat.

Now you could just export this map out to Photoshop and paint the texture there but for this skull was textured with a different technique known as projection painting.

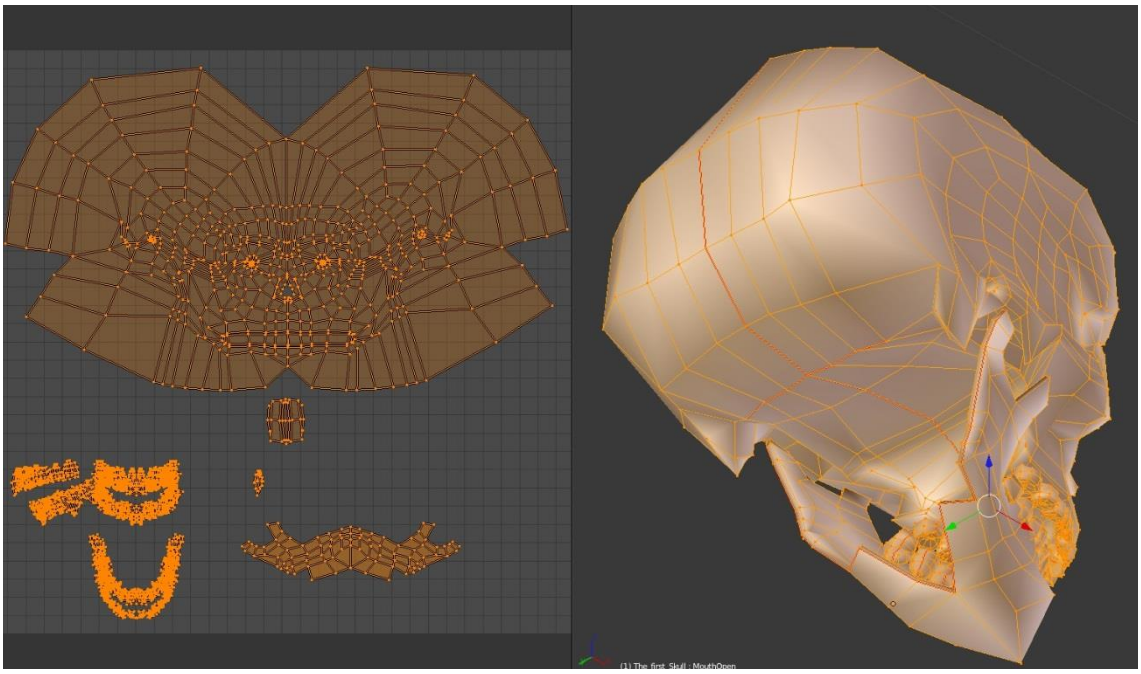


Figure 18. a uv map on the left and the model and the seam on the right

Projection painting

Projection painting allows you to pain an image directly on to the model. The image you are painting from is projected form your viewport angle on to your model. Blender automatically determines the corresponding UV-coordinate locations for you, selecting the proper pixels from the source UV-map and using them to modify the pixels in the target UV-map and the result is shown in real time. The model can be freely rotate at any time and the source images can be switched at any time. The brush itself works like a normal brush meaning you can vary its size, strength and jitter as desired. [10]

In this case the skull additional UV maps where made using “project from view”. The aim was to match the rotation of the skulls in the source images as seen in figure 18.

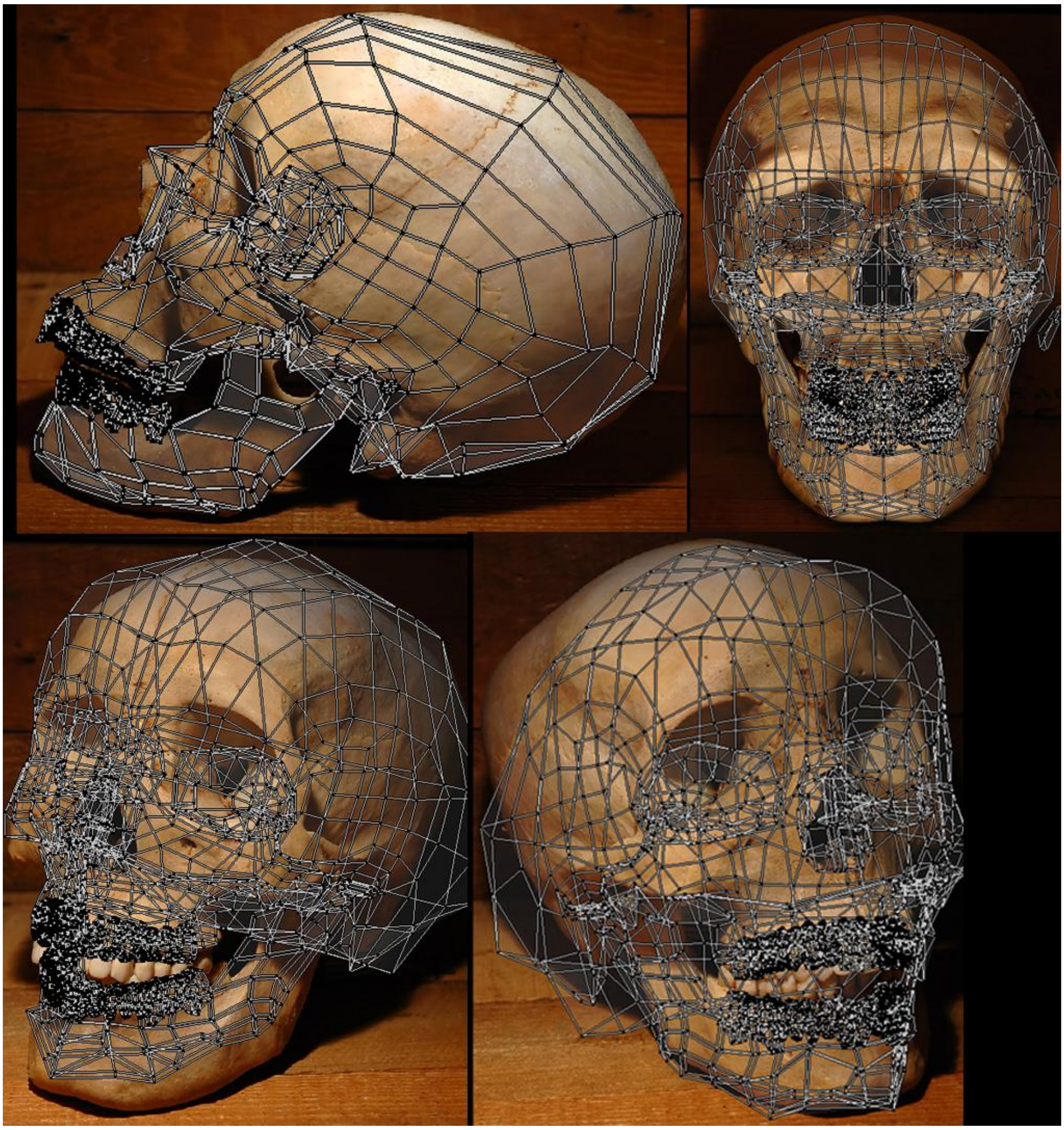


Figure 19. additional UV maps matched over the source picture

By painting form the different sources it was possible to paint around most of the skull and avoid major shadows and highlights. That painted texture was then taken into Photoshop to be cleaned up and the remaining shadows and highlights to be flattened out. This texture was also used as a bump map as is. A bump maps is basically a simplified version of a normal map and unlike a normal map only gives height information.

To further help make the texture fits its new geometry an ambient occlusion map was baked for it.

Ambient Occlusion or AO for short is a ray tracing calculation which simulates soft global illumination by faking darkness perceived in corners and at mesh intersections, creases, and cracks, where light is often defused by accumulated dirt and dust. This makes cracks, corners and points of contact darker. [10]

This information can then be baked to a texture and applied to the model for greater control or so it does not need to be calculated again when rendering.

In the below figure we can now see both diffuse map and AO map as well as both of them applied to the model.

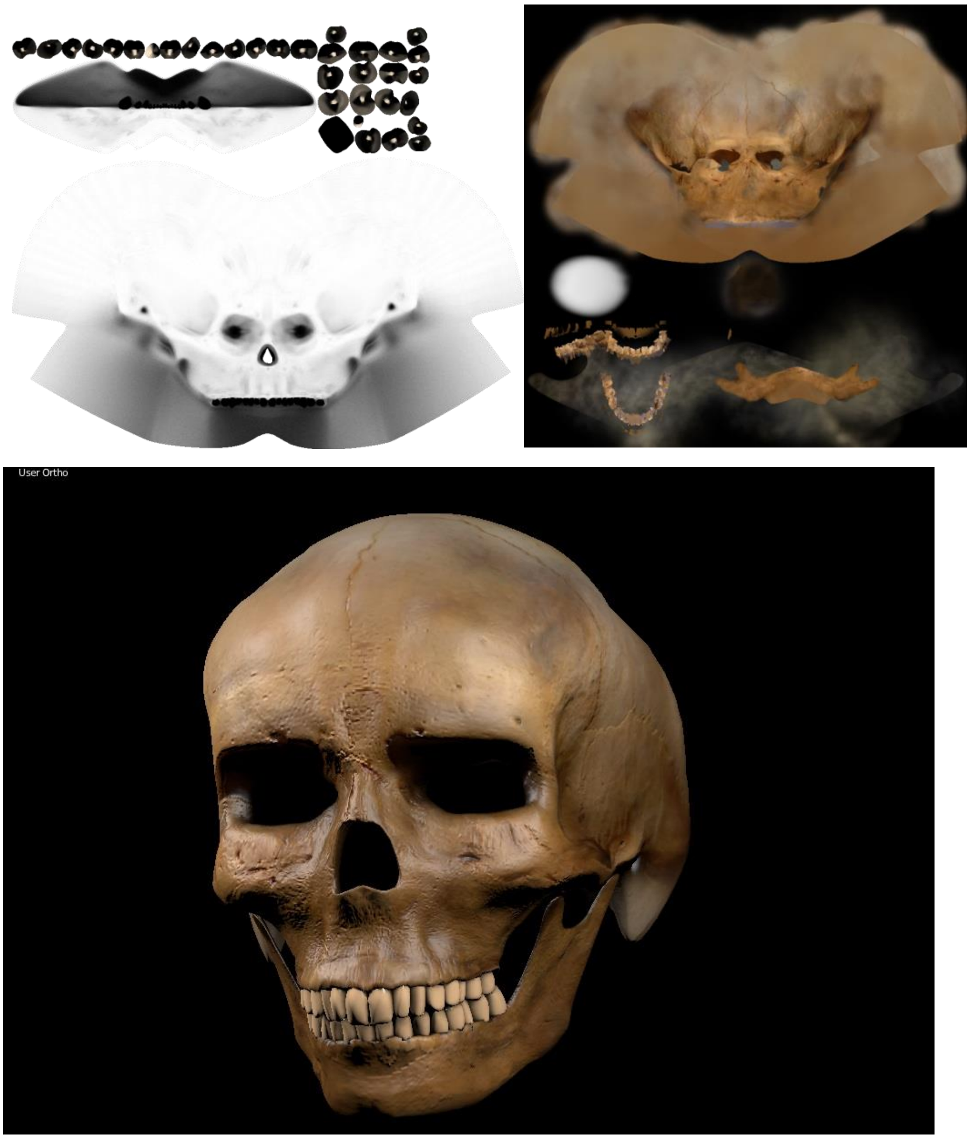


Figure 20. the AO map (top left) the diffuse map (top right) as well as the finished skull at the bottom.

The rest of the character was textured in a more traditional way mostly by applying images and procedural texture directly to the UV mapped objects.

The gold however proved to be quite challenging as much of what is perceived as gold is due to reflections much in the same manner as chrome.

Making gold shine

To make shiny object appear more metallic the plastic a good way to start is to adjust the specular color. By making the specular color a light shade of yellow as well as tinting the reflections in the same color help make the material look more gold then yellow paint or plastic. Like stated earlier reflections are important. When an actual 3D environment that can be reflected is not present a “world map” can be used as a substitute or to fill in parts where no models are present, like behind the camera. For the best result a special kind of texture is needed. In this case an angular map was used. Angular maps are these spherically shaped textures like seen in figure 20. These can be stitched together from a bunch of normal pictures, taken with special photographing rigs or a combination of the two.

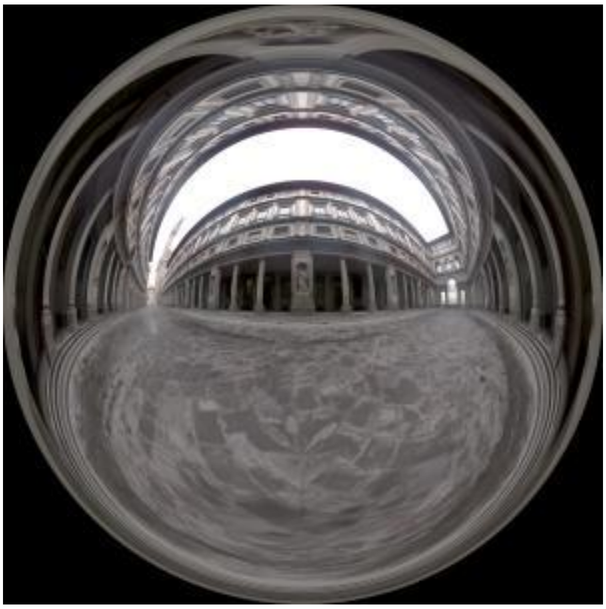


Figure 21. an angular map [11]

The image is then applied to the word where it effectively wraps around the camera in a similar fashion like a sky box but unlike a sky box it has no actual geometry and is always an unlimited distance away.

Now that there is a world to reflect the gold looks much more realistic as seen in the side by side comparison in figure 21.

Figure 22. a comparison of the reflection with the angular map on the right and without on the left.

2.3.3 Modeling: The Guard

The second character to get made was the guard. By far the most challenging and complicated of the three characters the guard took a lot of time to make.

Fantasycraft did not have a specific design for the guard. The only real requirement was that the character had a notable scar over the left eye as that was a major plot point. Fantasycraft had provided some general designs about what the race of “unliving” looked like. After looking though the different designs the solider-esque character seen in figure 22 was deemed as a suitable guide. The helmet and neck piece designs where a change to a more open model to give the character a wider emotional range as the face was now visible.

Figure 23. the concept art on the left and the final 3D model on the right

In order to speed up the modeling process a human base model was imported from MakeHuman 1.0 alpha 7. This base model was then heavily modified with traditional modeling as well as sculpting as seen in figure 23.

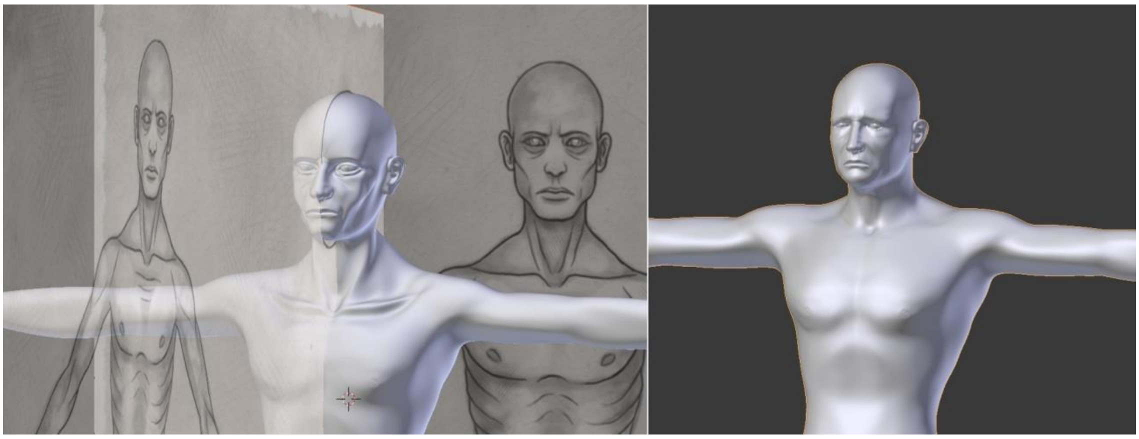


Figure 24. on the right is the model that was imported and on the left the result of the sculpting

The rest of the modeling followed conventional box and polygon work flow. For the belt a trick using arrays and a curve was used. Only two sections of the belt were modeled, the section that would make up most of the belt and the “end cap” that would finish up the belts end in a neat way. The belt section was then duplicated with the array modifier to make up length of the belt and as the modifier has a capping function it was only a matter of selecting the cap model. This array was then made to follow a curve that would wrap the belt around the character model. The curve and the arrayed section can be seen in figure 24. The clasp was modeled and added separately.

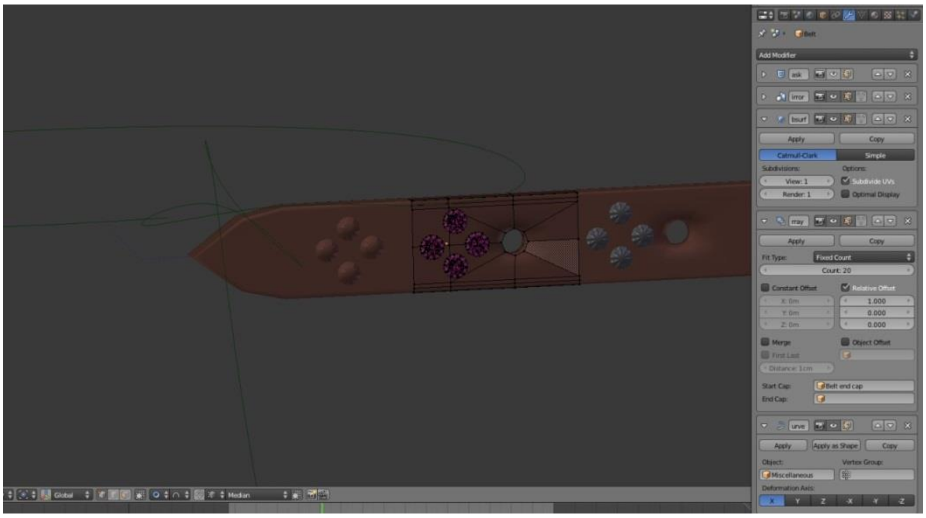


Figure 25. shows the curve and the segment used to make the belt

2.3.4 Materials: The Guard

Like for The First the bulk of the materials used on the guard where pretty simplistic, a bit of bump map here, a little reflection there but there were some materials that needed some special attention

The armor parts follow mostly the same idea as the gold but this time with scratches and dents where added to enhancing the feel that it was indeed something that had been used in combat. The chain mail however needed to be done somewhat differently. While simply modeling out all the chain links was initially considered this was quickly abandoned due to rigging and performance issue. What was done instead was a poly geometry that was baked to a normal map.

2.3.5 Baking normal maps

In Blender normal maps can be baked in one of two ways. Either from a multi-resolution modifier to the base mesh or from a separate high poly model to a lower poly model. For the chain mail the latter was used.[10]

First a few chain links where modeled out. Those where then arrayed to form a small patch of chain mail as seen in figure 25.

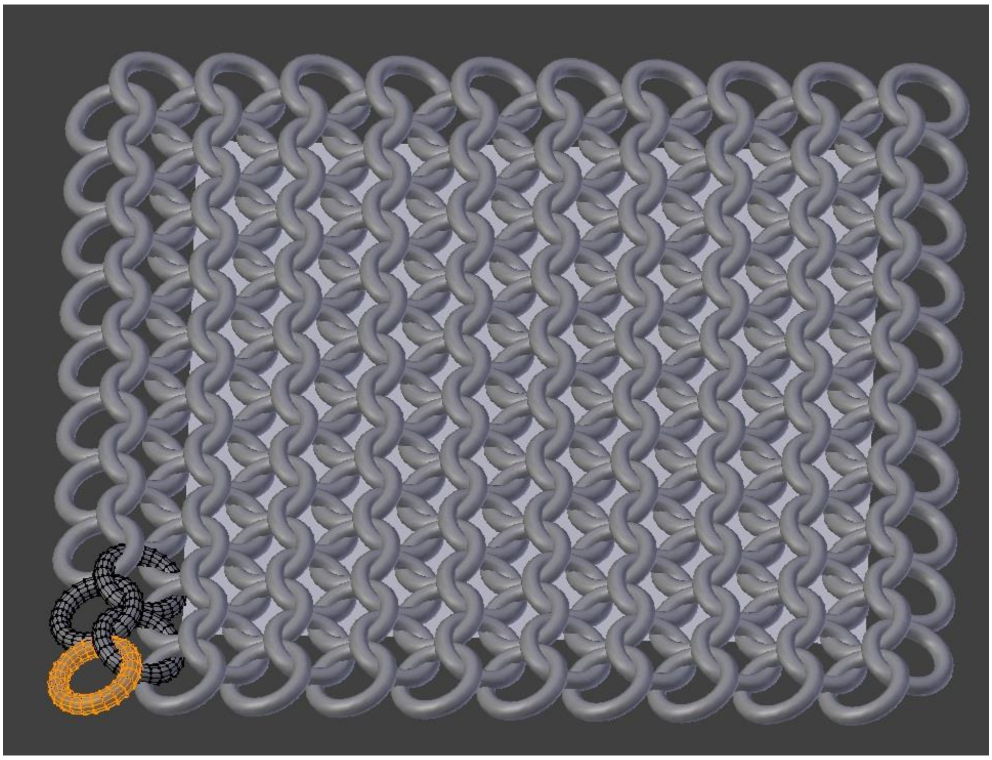


Figure 26. the high poly chain-mail patch placed over the low poly plain

Then a simple plain was added under. This would serve as the low poly sand in. Before an image can be baked a blank texture needs to be made that can then be overwritten with the baked normal map Then it the baking distance needs to be set. This part is a bit of a trial and error but after a few tries a suitable distance is found.

As the underlying plain was carefully lined up beforehand the resulting texture is tiling saving time and effort. However as chain mail has gaps in it something else was needed as well, a transparency map. To get this the normal map was brought in to Photoshop where the gaps where selected using select color. This was possible as all the gaps had been filled with the “very far away” color during baking. Then the selection was filled with black and that was it alpha map. The resulting maps as well as the final result can be seen in figure 26.

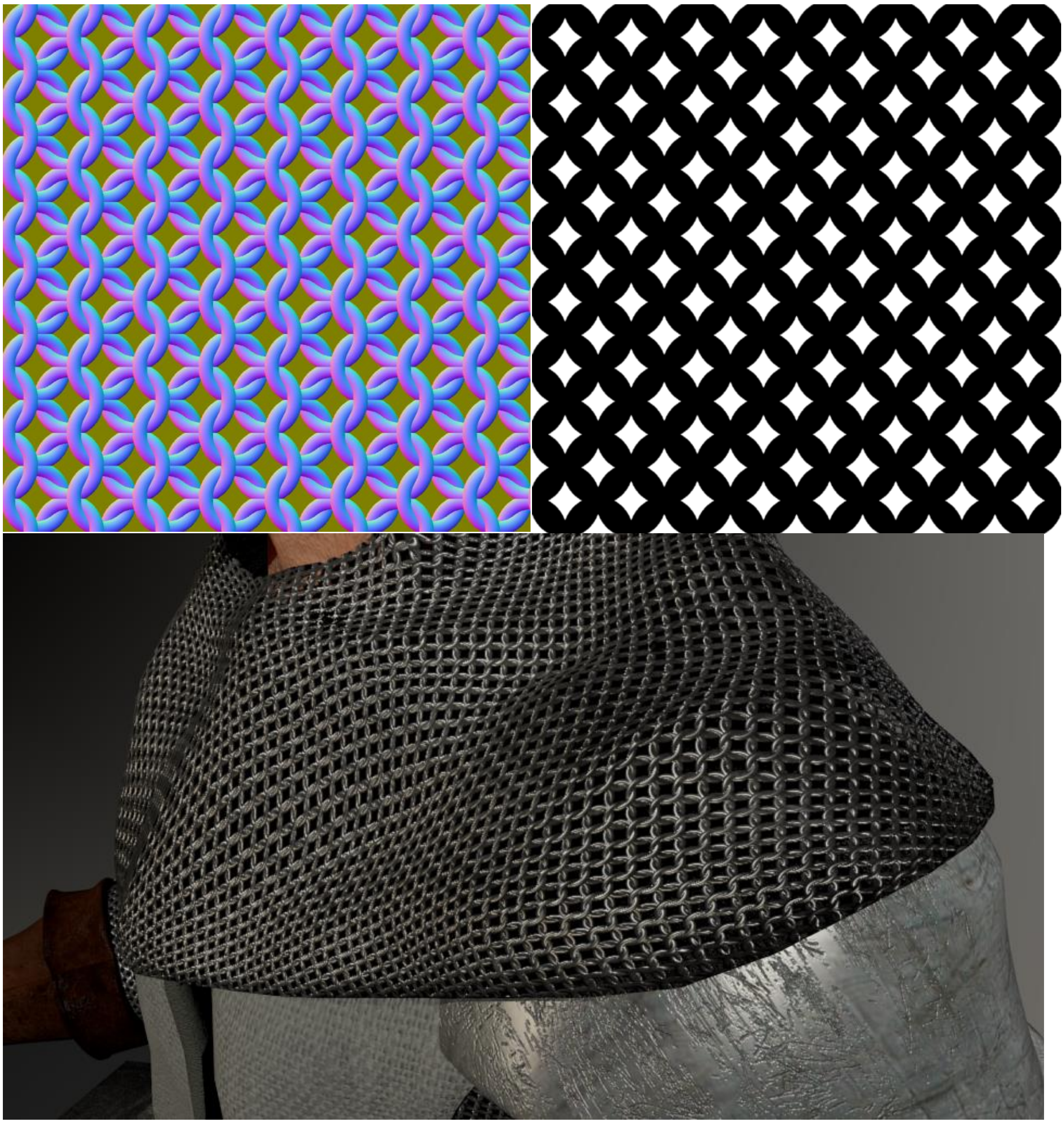


Figure 27. the normal map (top left), the alpha map (top right) and the final result bottom

2.3.6 Two Layer SSS: Making human skin

Makin convincing skin is rather difficult. Many different things need to come together in order for human skin to look realistic. While simply enabling subsurface scattering for a regular material this often leads to a more wax like. For a more realistic result a node based material was used. So what do these nodes do? Well the potential applications are too numerous to count but in this instance it allows us to combine multiple materials into a single material. That is to say we split the different properties of skin into different materials as seen in figure 27.

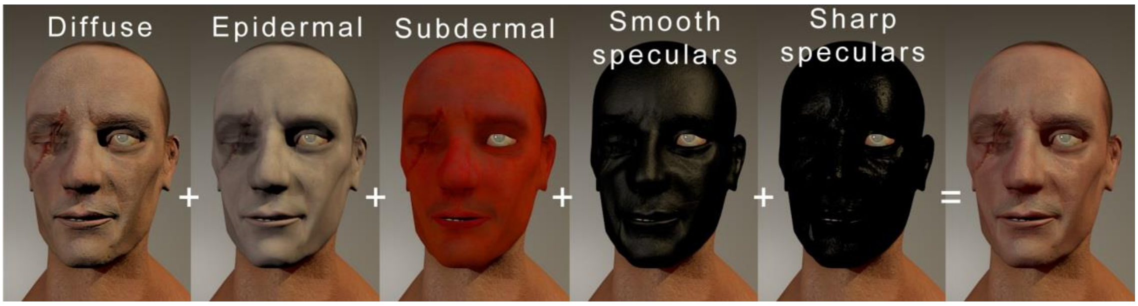


Figure 28. all the different skin node materials laid out

The materials are then assigned to nodes and setup in a node network like seen in figure 28 to make the final skin material.

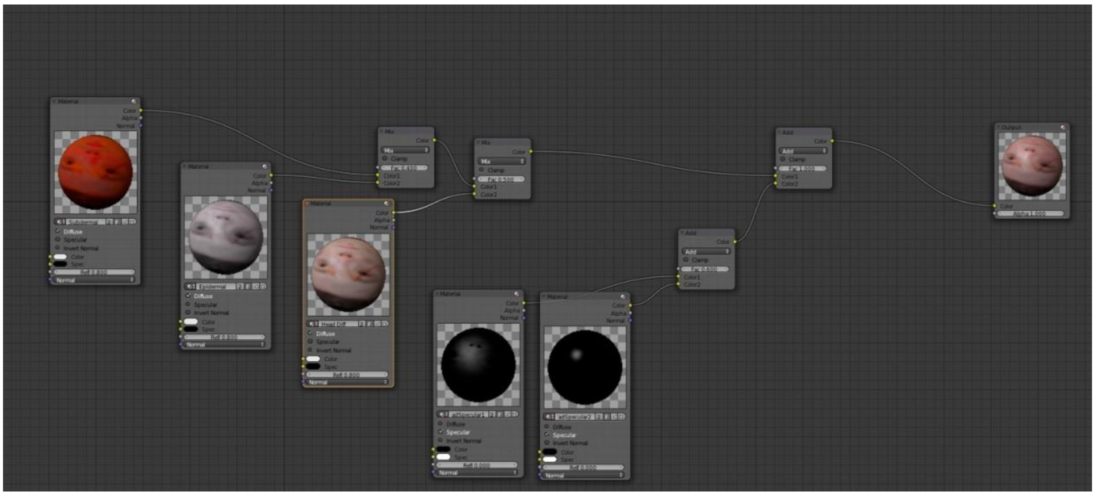


Figure 29. the material node network used to make the skin

Crucial SSS settings for skin

Before we look in depth at the different layer let's review settings that play the most vital roles.

IOR: Index Of Refraction value determines the falloff of incident light. Higher values means that light falls off faster. The effect is quite subtle and changes the distribution function only slightly. Values value in the 1.3-1.5 are known to work well in most materials. [10]

Scale: This is the major setting. What it does is determine how far light scatters through the skin. If the light scatters too far you get wax instead of skin. A scale of 1.0 means that one blender unit equals one millimeter – this is obviously way too large when the character is built on the assumption that **1** Blender unit = 1m. This means in order for light to penetrate 1mm the scale needs to be set to 0.001. [12]

RGB radius: This allows you to set the relative scattering amounts for red green and blue light, in proportion to the scale value. As different colors of light scatter different amounts under the skin these need to be adjusted. While not 100% correct the close enough results can be obtained by setting the color red to scatter the most, halve that amount for the blue and halving the blue amount yet again for the green.

Colour influence: This controls how much the RGB colour picker in the SSS settings affects the scattering and how much comes. Higher number mean more of more of the picker color is used in the scattering but even when turned to 0 it still has an effect.

Blend Texture: This setting determines how much the texture is blurred by the scattering, higher values will blur the texture more. To retain as much crispness as possible this is set to 0.

Subdermal

In our skin materials there will be 5 materials two of which are SSS materials. The SSS materials simulate the underlying layers that exist beneath the surface of human skin. The subdermal layer simulates the light that penetrates to the deep layers of the skin, and spreads out the furthest. As the light penetrates human skin it gets tinted in red and orange hues by the blood and tissues before scattering back. This is the most important layer for giving the material those rich saturated shadows that we see in natural skin. To mimic the underlying structures this layer is painted with subtle patterns of veins and capillaries as well as general redness for fatty tissues. The bony areas where there is less blood flow a more toned down color is used. [12]

Epidermal

The epidermal layer simulates the scattering that happens just beneath the skin surface. In real life this layer is mostly made up of dead skin cells and the thickness varies widely. In eyelids and lips it is much thinner so more of the subdermal layer shows through. The texture is given a bluish tint to counter act the redness underneath and the thickness is represented with the darkness. The darker an area is colored the more of the underlying subdermal layer will show trough. [12]

Diffuse

This is the skin surface itself and has no SSS. This layer represents the final color of the skin and it can be used to fine tune the pigmentation as well as add anything sitting on top of the skin like makeup or dirt and grime. Like with make up this layer can be used to fine tune the redness of the skin the same way you would apply blush with makeup. [12]

Two part specular

The reason the specular highlights are made separate from the diffuse material is that we want to use two of them. We want one that represents the small hot highlights and one that give larger smooth shine. The hot specular material is a simple black material using a standard specular map with a high hardness and will remain mostly the same regardless of the conditions. The reason its black is that it can then be easily overlaid on top of the rest of the materials in the node network using the add node.

The second material is used to represent the skins condition. If the skin needs to be wet and shiny this materials intensity is turned up and some amount of ray mirror is enabled when the skin is wet enough. In this case however is very dry so there was no need to enable it in this instance. [12]

2.3.7 Hair for the librarian

Figure 27. the completed librarian rendered

While the librarian (figure 27) shared much of the same workflow as the previously covered characters, the skin material for example is the same as the one used for guard only with new textures. There was one new challenge which this character posed and that was hair. There are many approaches to make a full set of hair.

While mesh hair that is to say a polygonal model representing hair can give very good results in this case something that was more physically correct was used: strand rendering.

In strand rendering hair strands are small polygons, which are extruded to follow the direction of the static particle line, but whose width is exactly perpendicular to the viewing angle and precisely 1 pixel wide. [13]

Getting the strands on the head

In blender the hair is part of the particle system and like a particle system it needs to have an emitter defined. While the hair can be grown strait out of the head mesh a separate emitter object (figure 30) was made so that interpolation errors could be minimized as well as to make assigning materials easier.

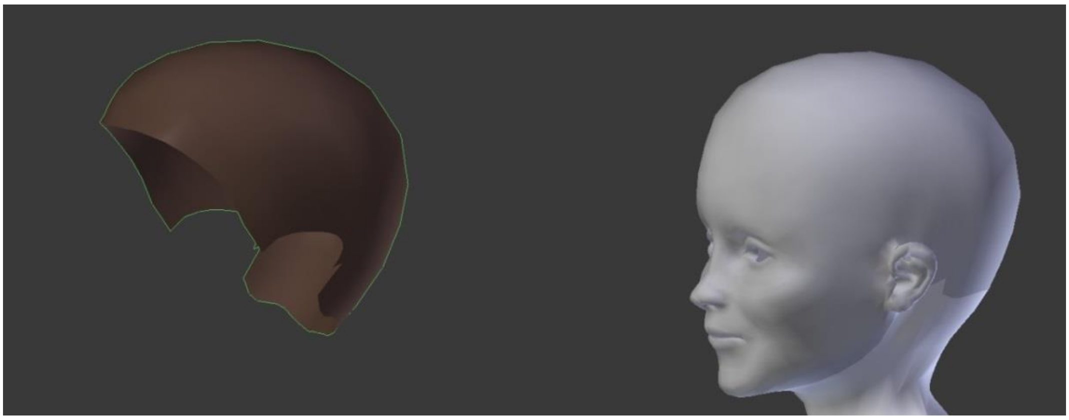


Figure 31. the scalp(left) and head model(right) side by side

To grow the hair a particle system is then added to this scalp. This particle system is the put into hair mode, doing that generates the basic set of hair according to the length and number setting like seen in figure 31. As the hair can later be freely edited the main goal here is to get a nice even distribution. There are a 1000 control strands that are further supplemented by child strands. The number of Segments can be set to about 5, the number of segments define the amount of control points when editing the hair. A small initial Normal and Tangent velocity gives the hairs a first approximately right direction. [14]

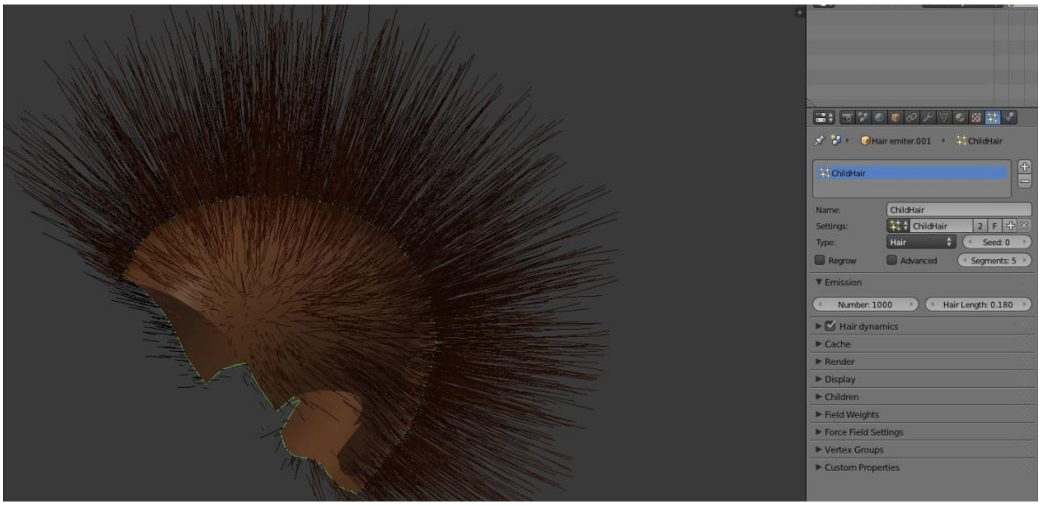


Figure 32. the hair in its initial orientation

This hair is then styled in the particle mode using comb and cut tools much in the same way hair would be styled in real life as seen in figure 32. At this point more control strands can also be added manually if needed.

Figure 33. the styling process illustrate

Though the initially generated 1000 strands are enough to define the shape of the hairdo, it’s not sufficient to actually give the correct volume of hair needed. This is where the child strands come in. Instead of using thousands and thousands of actual strands that would seriously hamper performance additional child strands are generated based on the existing strands. During the hair styling a small amount of child strands where already used to help visualize the hairdo. The child particle setting works as a multiplier so when the settings was set to 5 during the editing phase the 1000 particles where turned into 5000. For the final render the multiplier was set up to 150 resulting 150 000 rendered strands. The difference can be seen in figure 33.

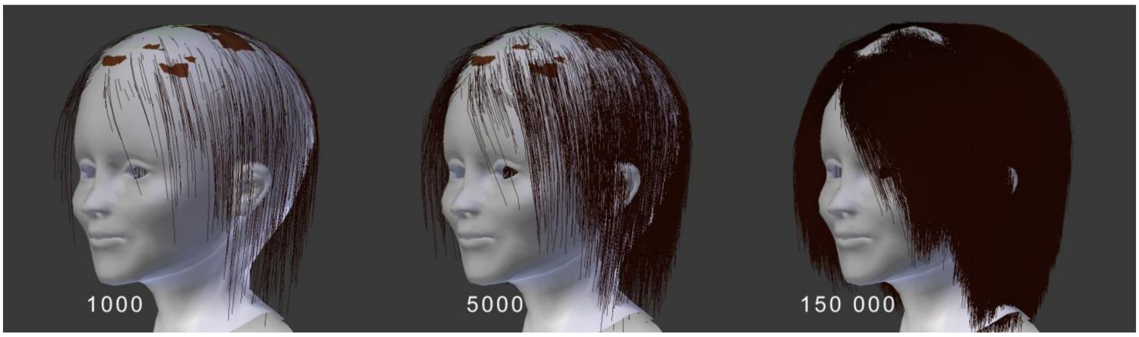


Figure 34. a illustration of how the strand count increases with the child multiper

The use of child strands also opens up additional settings (figure 34) for styling. While settings like Braid which makes children braid themselves around the parent hair or Curl that makes the children grow in a spiral around the parent hair can be used in great effect to make intricate hairs styles in this case the aim as was a more natural and messy hair. To achieve this, a combination of Clump, Shape and Roughness was used.

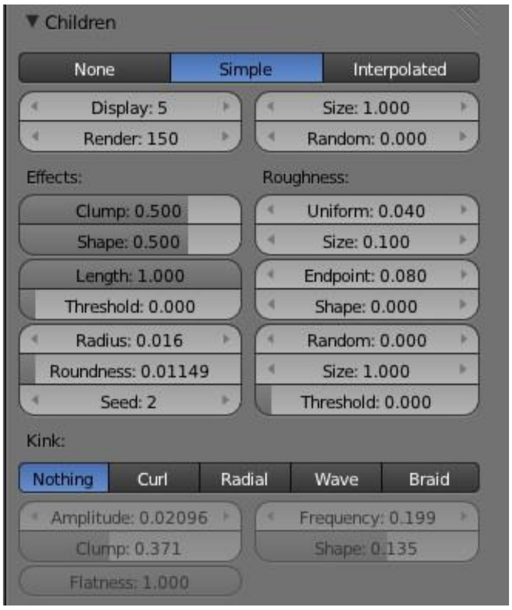


Figure 35. what the child strand controls look like

First clump is used to make the particles clump together more towards the end giving the hair a dirty/greasy feel. As just using clump results in spikes the shape setting is used to puff out the middle part of the strand group. To make the hair messy the roughness is turned up that like the name suggests makes the hair a more erratic shape. The effects can be seen in exaggerated form in figure 35. [15]

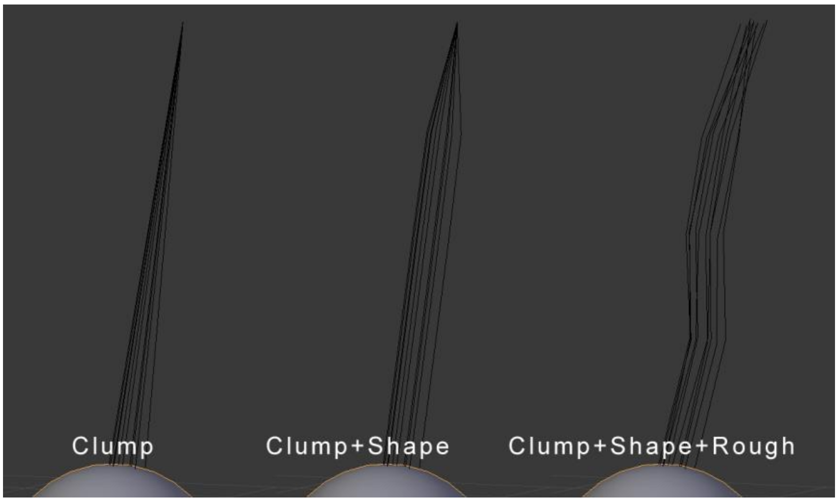


Figure 36. an lustration what the clump, shape and rough settings do

The roughness actually has multiple settings that can be used to fine tune the effect.

Uniform Rough: Adds a global roughness to the hair that affects all the particles along their whole length.

Endpoint Rough: This adds noise to the endpoints of the hairs, pushing them apart giving a frayed look.

Random Rough: This setting can be used to add stray or flyaway hairs.

Hair material

The hairs can use textures in the same way as a normal mesh surface would. In this case the strands inherit this color from the spot on the mesh which the grow form.

Additionally textures can be setup to affect the color along the length of the strand. It’s also a good idea to use texture to affect the alpha along the length. The reason for doing this is to make the hair fade out nicely at the tips instead of doing a harsh stop as seen in figure 36. [14]

Figure 37. a comparison between an alpha fade(left) and none alpha faded strand render

2.4 Rigging and the bodies

In order to be able to animate the characters they first need to be rigged. A rig is essentially digital skeleton that is then used to deform the overlaying mesh. Like a real skeleton, a rig is made up of joints and bones that can then be moved and animated to give the character the desired poses and movement. This approach is called skeletal animation.

The digital skeleton or armature as it is called in Blender is made up of what else but bones. These bones have the standard line up of transforms that is to say position, rotation and scale. The bones can also have parents that they inherit transforms from and in doing so form a hierarchy.

This hierarchy can be made to work in two ways: Forward kinematics or FK for short and Inverse Kinematics that is abbreviated to IK.

In an FK hierarchy any given joint can only affect parts of the skeleton that fall below it on the joint hierarchy, so for example when a hand is rotated the finger bones are rotated along with it. With IK a bone lower in the hierarchy is positioned and the other bones within the limits of the IK chain will be positioned automatically so the position for the last bone is possible. That is to say if the IK chain ran form the hand to the shoulder the arm would move and rotate in order to keep the hand in place when the rest of the body is moved. This behavior can be very useful for hands and feet.

2.4.1 Auto rigging with Rigify

Making a good rig can be quite labor intensive, luckily blender has a thing called Rigify. What Rigifye is an auto rigging system designed for humanoid characters, though it can be modified to work with many other types as well.

The way Rigifye works is that instead of building a rig one bone at a time like you normally would you add the human meta-rig to your scene. This is guide armature is then made to fit the character. When the meta-rig is properly lined up the actual rig is made by hitting the generate button and so a fully functional rig is generated in a fraction of the time it would take to make it from scratch. This workflow is illustrated in figure 37.

Figure 38. illustration of the Rigifye workflow

There are draw back however as the generation process relies on initial rotations to determent the correct orientations of the bones. This can cause problems if the model has joints that have very strait as Rigifye can get confused as to which way the joint is supposed to bend. Usually these types of issues can be fixed by giving the joints slightly more initial rotation and re-generating the rig.

2.4.2 Skinning

Now the rig might be fully functional but in order for it to be able to move the character the various meshes need to be linked to the bones. This is called skinning. In order for the bones to know what part of the mesh to move they need vertex weight groups. Initial crude vertex weight groups can be generated by using the Armature Defrom: With automatic weights. Now the armature is able to move the meshes but for good results these weights often need to be tweaked. To do this the weight paint mode is used. In figure 38 we see the highlighted head bones as well as the vertex weight group represented in colors. The red areas are completely controlled by the head bone while the blue areas are the region where the head bone has no effect. The yellow and green shades represent the blending where the head bone has limited effect.

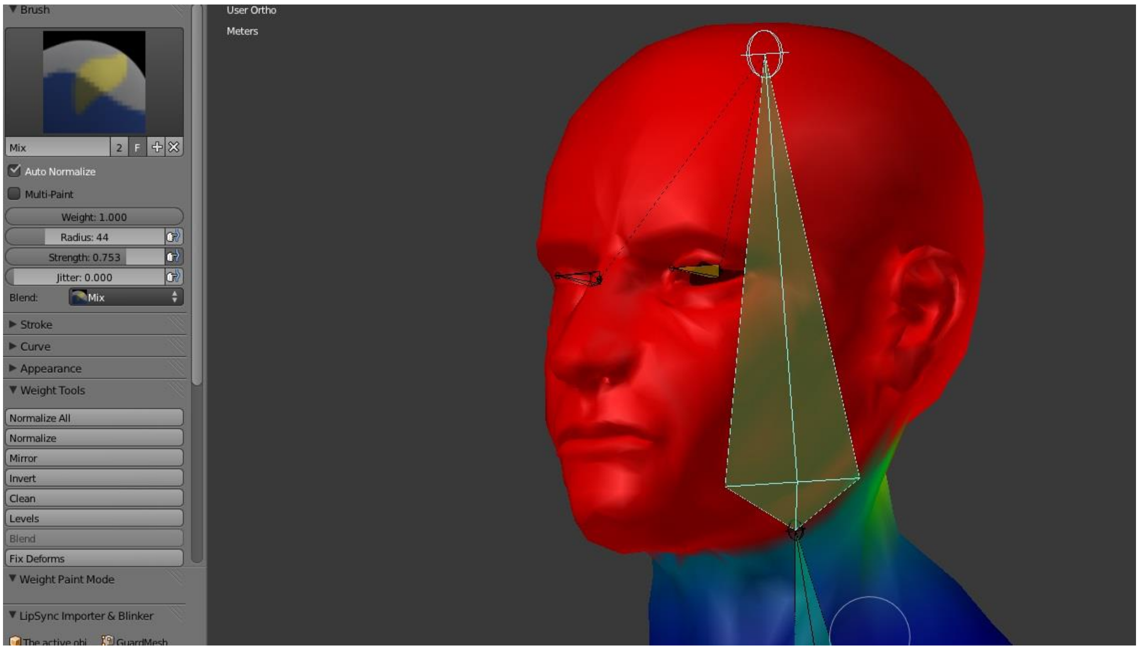


Figure 39. the head bones influence represented in red

2.5 Sets, props and linking

Before the characters can be animated they need an environment which they can interact in. At the surface it seems simple enough; just model the environment the same way you would any other object and indeed it's not the modeling that is the issue. It's managing the multiple characters, environments and objects that become the issue. Simply sticking all the assets in to one giant blend file would be unruly to say the least. This is where blenders linking system comes into play. What this allows is assembling parts form different files into one stage file.

The stage file can be completely void of any local assets or it can have everything except the characters themselves. In the case of our trailer the assets where broken up into three categories, location, props and characters. The locations files would also serve as stages so the props and characters where linked to the location files from their respective blend files.

The location files contain buildings, skyboxes, ground plains as well as all the lights and compositing node networks. In regards to the lights there are actually some exceptions as things like the glowing orbs are located in the prop files and yet they have lights of their own that get linked along with them.

Let’s take a look at how a complete set comes together. In figure 39 we first see the plain set then the set populated with the various props and lastly the final shot with the character in it.

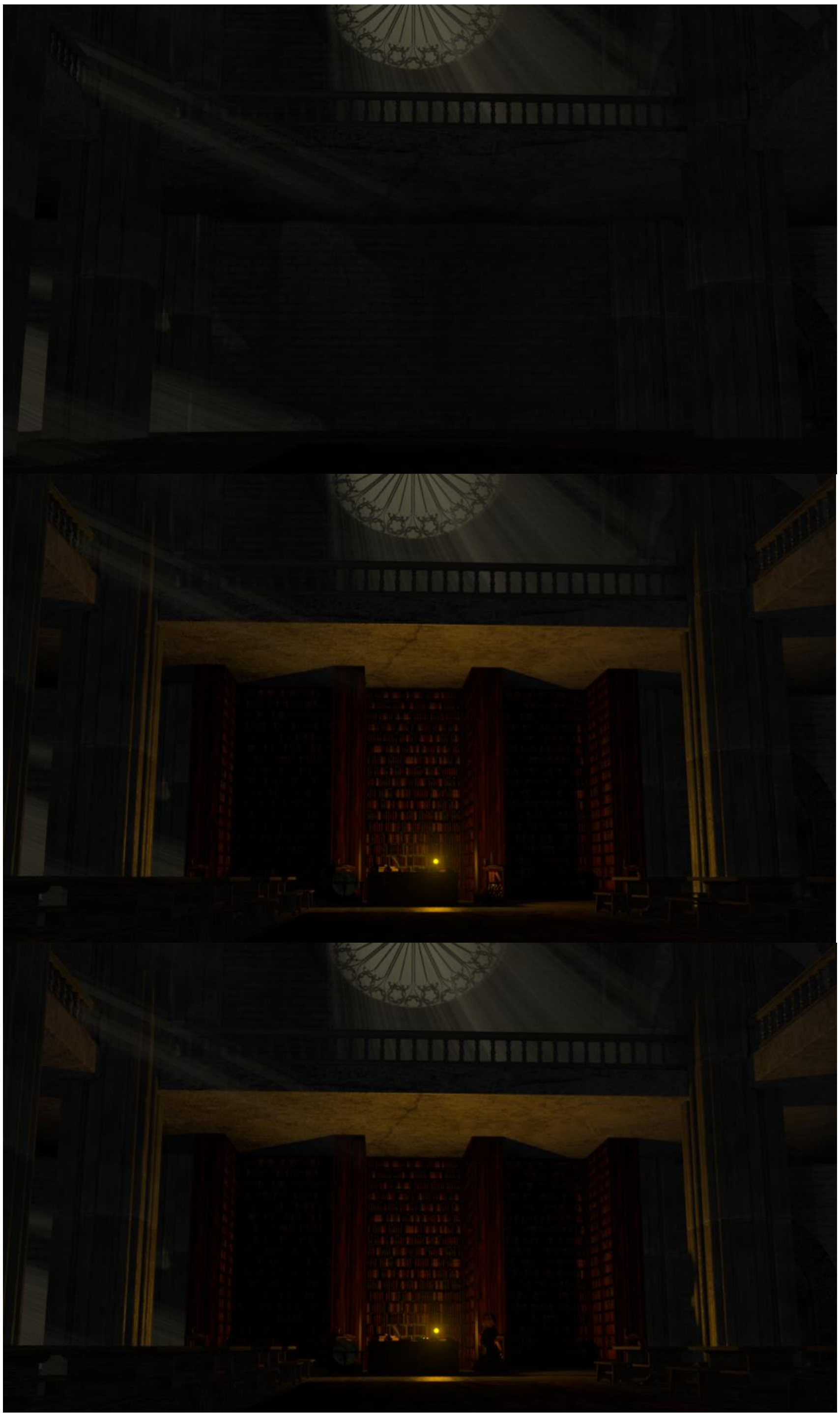


Figure 40. the bare set at the top the set with props and at the bottom the set with props and the character.

2.6 Recording the dialog

At this point in the production the temporary soundtrack made during the pre-production is slowly staring to outlive its usefulness. Although music and sound effects would be added in postproduction as they are fairly easy to sync up the dialog itself would be needed before the animation. Animating facial expressions and lip-syncing is far easier to do when you have the dialog in its final form as reference. [1]

The equipment

The equipment used to record the dialog was a Behringer B-1 condenser microphone, a miditech audiolink II USB sound interface and a pop filter as seen in figure 40. For the initial recording audacity was used due to its fast setup and ease of usage.

Figure 41. the recording equipment used

The recording session

Two people form FantasyCraft where kind enough to come over and provide voices for the characters. As there were three characters I myself was also tasked with provide one of the voices.

In order to capture good quality dialog it is necessary got get quite close to the microphone as this enables it to pick up more of the warm tone of the voice. As the person doing the speaking is so close to the microphone there is a tendency for the microphone to pick up blowing sounds due to the air people exhale when talking. This is why a pop filter is needed as it helps block the exhaled air but lets the sound waves through. Another crucial thing is to use appropriate amounts of gain. In this context gain refers to the amount of signal boost the microphone receives form the audio interface with in turn determines the input level. Too much and you get clipping, too little and the dialog will be recorded very quietly and easily be drowned out by the background static. Low recording volumes are preferable to clipping as clipping ruins the sound to a mostly unsalvageable state.

In audacity the input level is represented by the red bar (figure 39). If the red bar hits the bottom you get clipping. The blue patter is from waveform which is a visual representation of sound. This can also be used to see if the recorded sound has clipping. If the sound clips the waveform hit the edges of the edges of the timeline.

Figure 42. Audacity interface with a wavefrom in the time line

With decently strong input level and a good safety margin for clipping the dialog could be recorded. These recordings where then stored for further processing.

2.7 Animation

Animation in Blender is based on key frames and it’s not just bone that can be animated with key frames. Blender has been built around the idea that pretty much anything and everything can be animated.

In computer graphics unlike traditional animation movement can be interpolated between the key frames. In blender this interpolation is represented with Interpolation curves or IPO curves as they are most often called. As seen in figure 40 the key frame has handles that can be used to adjust the interpolation. The key frame itself can also be moved after it has been created in order to adjust the animation timing. [10]

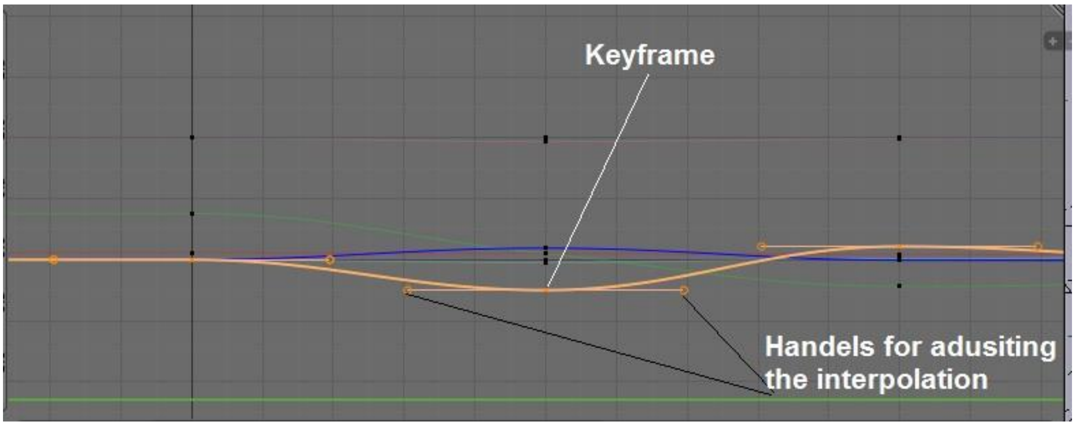


Figure 43. IPO curves in the editor

2.7.1 Walk cycle

As a walk cycle is one of those easy to learn hard to master type things it’s a good starting point to begin make animations. Like the name suggest a walk cycle is a looping animation that is designed to repeat over and over seamlessly creating the continuous walking motion. At its simplest a walk cycle can be defined by 4 key frames. The forward contact point, passing pose 1, Back Contact Point and passing pose 2 as seen in figure 43.

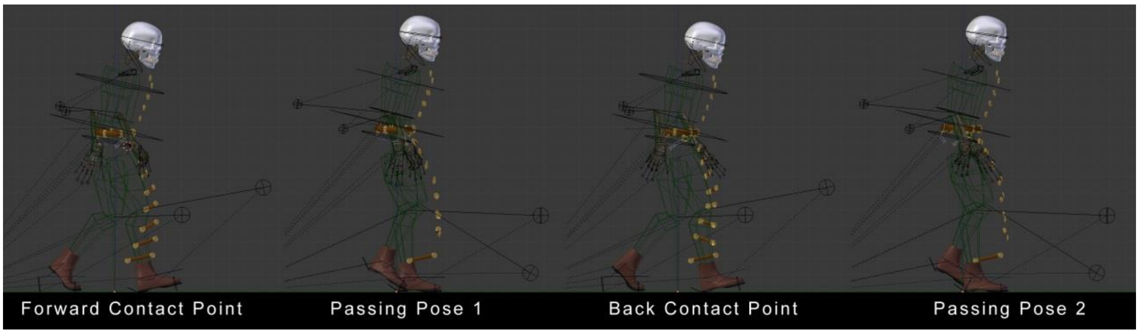


Figure 44. A breakdown of a walk cycle

This animation can then be further refined by adding additional key frames. As the walk cycle it doesn't make the character move an additional animation is needed. This is done with the Follow Path Constraint. What this does is make the rig follow and turn according to the curve. The speed at which the rig follow the curve is defined by the Evaluation Time. This is a user animated value so the rig can be made to stop and start moving as desired with the desired speed.

2.7.2 Pose to pose animation

A looping walk is good for getting the character from point A to point B but won't do much when we want interaction. The pose to pose process starts by roughing out the need main poses. In the case of opening a door this could be for example lifting up the and from the rest position on to the door handle and then a third key frame with the pulling action. This movement is then refined over and over by adding additional key frames until the movement is satisfactory. This can result in quite high numbers of key frames as can be seen form figure 44.

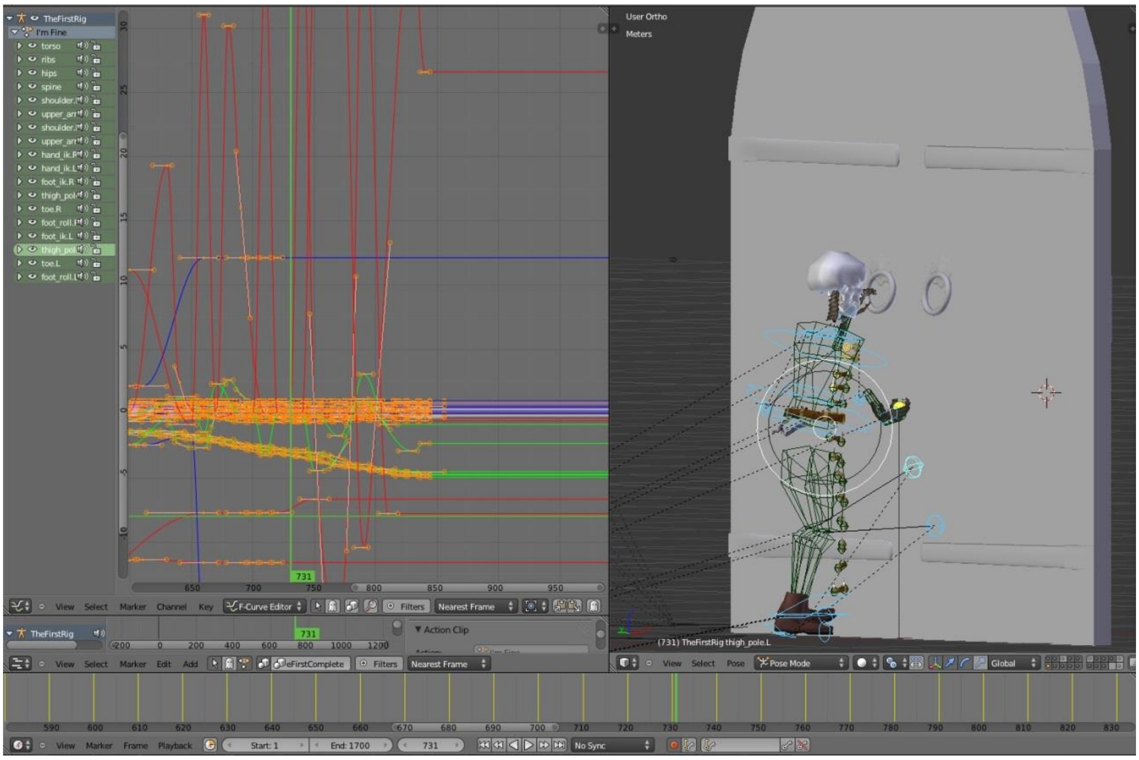


Figure 45. The first during the door opening animation

Unfortunately sometimes adding a large amount of key-frames can backfire as the desired movement is not always reached. When this happens the large amount of key-frames makes readjusting the timing very difficult. A way to avoid this is to really more on the IPO curve handles to adjust the animation but this approach also has its own limitations.

2.7.3 Non-Linear Animation Editor

The Non-Linear Animation Editor or NLA is one of the more powerful animation tools in blender. It enables the manipulation and repurpose actions, without the need for direct key frame editing by converting the key frames to easily manageable action strips. It can be used to speed up, slowdown or repeat animations making it essential or walk cycles. It can also be used to layer actions like in figure 45 where it has been used to add a different kind of hand movement to the walk cycle.

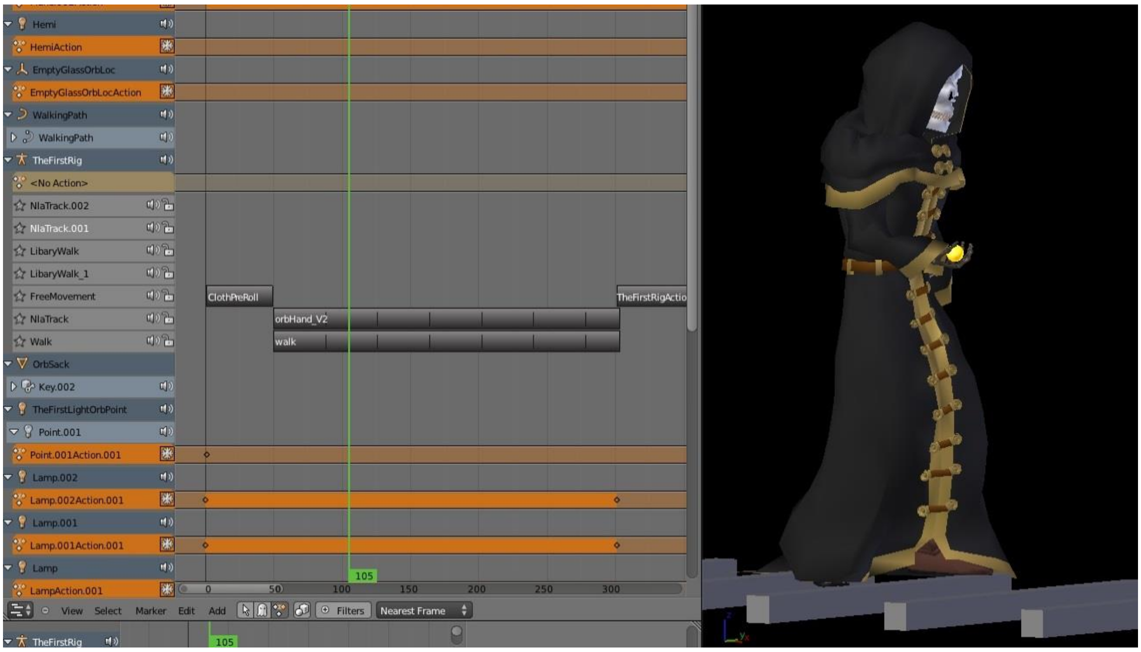


Figure 46. the walk cycle modified with the NLA editor

2.8 Lip-sync and facial animations

The body animations start to get better defined it also get more clear where facial animations will be needed and where they would simply be a wasted effort. The facial animations will be made in somewhat different manner then the body was. While it’s possible to make a face rig using convention bones there is another and arguably superior way of doing it and that is with shape keys

2.8.1 Face rig with shape keys

So what are shape keys? Well where regular keys to describe the position of a bone at one point in time a shape key describe the position of vertices within a mesh. What this means in practice is that with shape keys a mesh can be made to smoothly morph between different shapes. This is well suited for facial expressions. Furthermore as the shape keys modify vertex positions it’s possible to use sculpting tools to make the shape keys.

Even a basic face rig needs a large number of shapes.

The shapes used for the librarian and the guard where:

- Jaw down
- Jaw left
- Jaw right
- jaw jutting
- Upper lip up
- Lower lip down
- Snarl left
- Snarl right
- Squint left
- Squint right
- Eye lid top left
- Eye lid top right
- Eye lid bottom left
- Eye lid bottom right
- Brow anger left
- Brow anger right
- Brow middle up
- Brow up left
- Brow up right
- Smile right
- Smile left

Now these shape keys could be animated as is with individual sliders but due to the large number of shapes it would be very slow and counter intuitive. Instead the shapes are driven by a smaller number of control bones. In this way things like jaw down, jaw left jaw right jaw jutting can all be controlled with single control object. The finished face rig can be seen in action in figure 46.

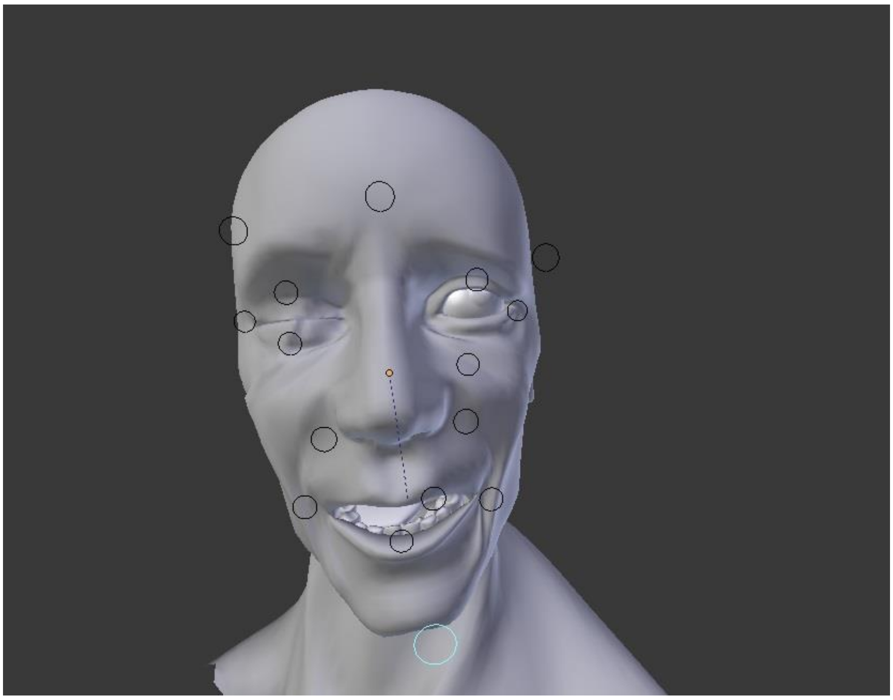


Figure 47. the guard demonstrating the new shape key based movement

2.8.2 Lip-sync with Papagayo

Now that the character can move and emote with their face it’s time to make them talk. To accomplish this we need yet more shape keys. This time they will be in the shape of phonemes that is to say mouth shapes. Below in figure 47 we see the phonemes that Papagayo is made to work with. These shapes approximate the shapes our mouths make during speech. Additional custom phonemes can be made but as the dialog is in English the standard ten should suffice.

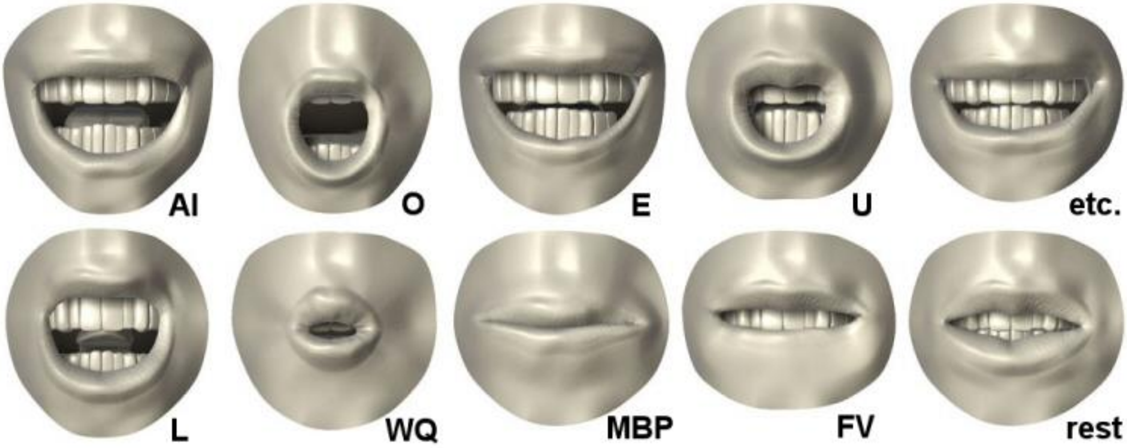


Figure 48. the ten phonemes needed for lip-syncing

So these then mouth shapes need to be made as shape keys for the character model then what? Now we are going to leave Blender for a while and do some work in Papagayo. What Papagayo does is help line up phonemes with the recorded dialog. The process is strait forward and surprisingly fast.

To create lip sync you import the dialog sound file into Papagayo then the dialog is typed in to the field. After this the words overlay the waveform and then it's just a matter of lining things up with the help of playback and the illustrated phonemes (figure 48).

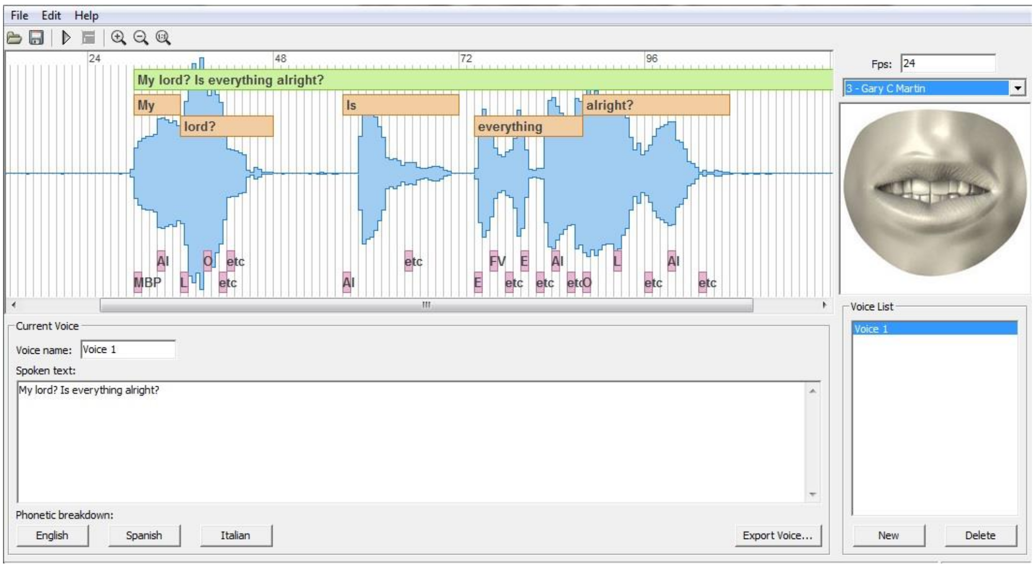


Figure 49. the Papagayo user interface

The lip-sync data is then exported form Papagayo and imported in to Blender. There are still a few settings (see figure 50) that need to be properly adjusted. Skscale basically determines how big the mouth movements are. Setting it to 1 gives maximum movement range while setting it to 0.5 will give you half the movement. Frame offset determines when the animation is supposed to starts. The ease in, hold gap and ease out effect how the shape keys will flow in and out of one another. Finding the good setting for this is largely trile and error.

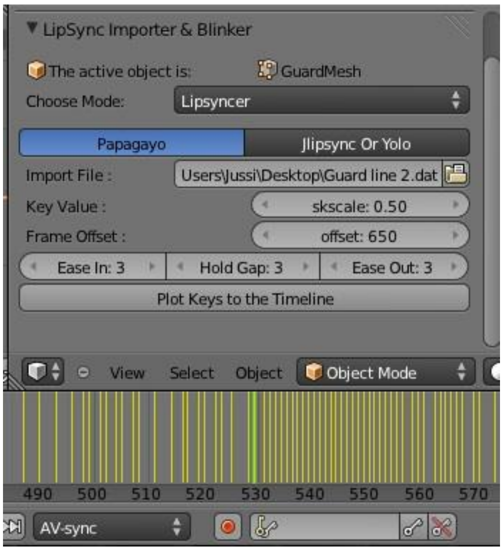


Figure 50. the lip- sync imported in blender.

After the keys have been generated they work the same as any other Blender key frames, that is to say they can be edited moved around and converted into a NLA strip.

2.9 Simulations

Simulations are a big part of this animation and also perhaps the hardest. While the simulations in Blender are currently excellent for a free 3D suite the cloth simulator in particular can be hard to work with.

2.9.1 Cloth

Some are of the opinion that Blenders cloth simulator is unusable for character clothing. This this notion is not unwarranted as Blenders cloth simulator has a very unfortunate tendency to have jiggle, stretching and jitter problems. Never the less the cloth simulator would be used for clothing in this animation as producing natural looking cloth movement using keyed animation is very hard.

To enable cloth simulation you go in to the physics section of the mesh and click enable cloth. The entire mesh is now governed by the cloth simulator. There are presets for things like rubber, cotton, silk but truth be told none of them are very realistic so the settings have to be dialed in according to the circumstances. As it has nothing to collide with it will simply fall indefinitely. Any mesh can be made to work as a collision object but high poly meshes are extremely slow to calculate so using them is to wise. Instead a dedicated low poly collision mesh is made as seen highlighted in figure 51.

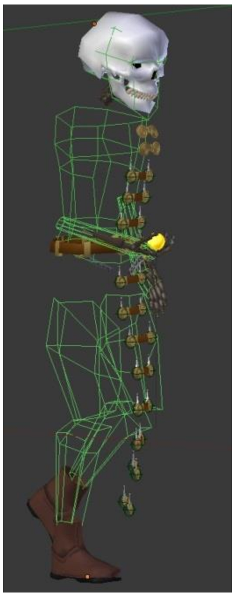


Figure 51. the collision meshes highlighted

Now the cloth will actually draip around the character but there is an issue. Ny fast moovement and the simulator can’t keep up resulting in an armo or a leg going tough the cloth. This is ont only incredably gully but also makes the simulator go highwire. This is not usable. To combat this issue most of the cloth is pinned to the rig. What pinning esenally does is mark a part of the mesh as not cloth. The pinning works with the same vertex weight groups as rigging so its time to do some more weight painting. The goal is to pin down as much as possible of the cloth and still retain enough movement to keep the cloth look.

The weights where applied as seen in figgure 52.

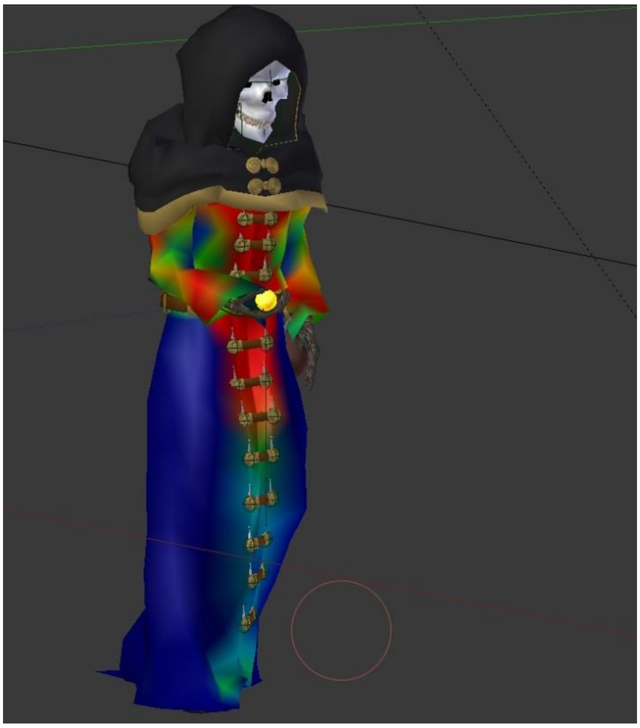


Figure 52. the pinning weights.

This helped rain in the uncooperative cloth simulator quite a bit but sill it was not producing the wanted results. Then a break trough was discovered. Turning the cloth meshes quads in to triangles improved the cloths characteristic immensely. A comparison between a regular quad mesh and a triangulated mesh deforming around a sphere can be seen in figure 53.

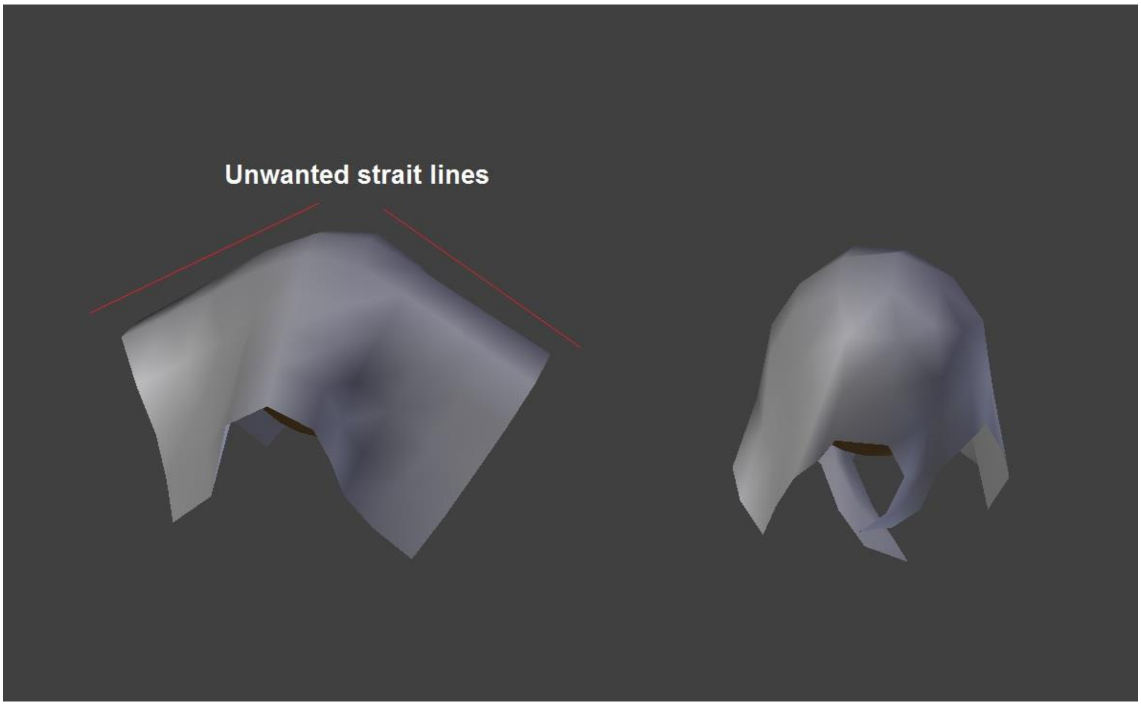


Figure 53. a comparison of a plain using quads(left) and a plain using triangles (right) when running cloth simulation

While the cloth simulator was still a hand full to use and might need several adjustments to look right for a shot it was now at least usable.

2.9.2 Hair

While hair was mostly covered already, what's left is the subject matter of dynamic hair. Hair dynamic in blender currently (2.64) is in a bit of a flux. Back in 2.49 hair could be linked to soft bodies, but that's not possible anymore. Instead hair is getting its own Hair Dynamics. So why is this problem? Well the hair dynamics in its current state does not support collisions. This causes a major problem as it either forces you to use short hair which we in the case of this animation did not want to do hair that would go through anything without any reaction or have the hair completely immobile. None of these where very appealing options. A work around was in order and indeed a work around was found.

The first step was quite similar to what was done with the cloth simulator that is to say only allow the minimum need movement by applying more weights. The second part was to make a collision mesh also much like the one used for the cloth but with one key difference. Instead of making the mesh a collusion object that would have no impact on the hair it was instead turned in to a surface type force field. Force field do affect the dynamic hair but simply having the force field would just repel the hair all over the place. In order to apply this work around successfully the minimum and maximum falloffs would have to be enabled. By setting these values really low the force field only works a very short distance from the mesh surface so when a hair strand get close it will start to get repelled. It's not true collision by any means as hairs can still quite easily penetrate the mesh surface. The key difference is that the strands don't stay inside the mesh but rated strive to move back outside it. The weight paint and force field settings can be seen in figure 54.

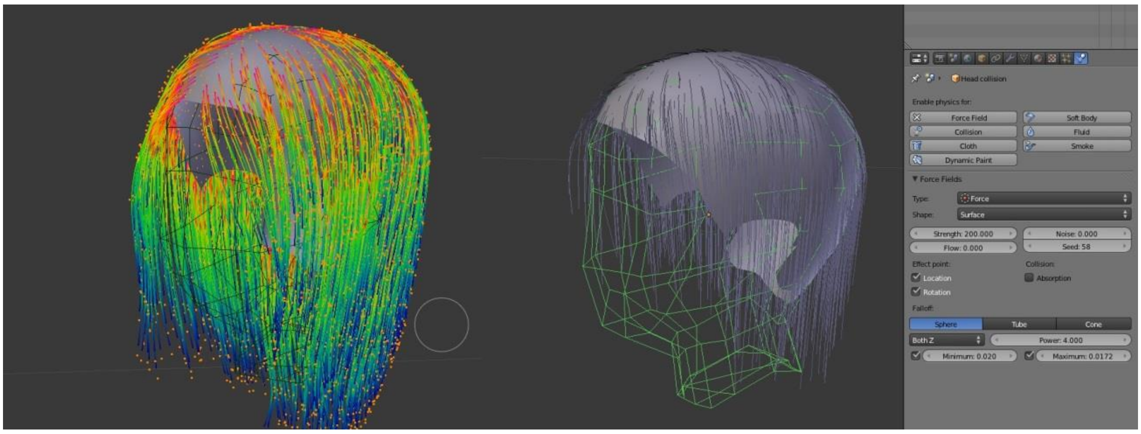


Figure 54. the hair weights and the force field settigns

2.9.3 Smoke

The final simulation that was used in this project was the smoke simulation. To make smoke in blender you need two things; a mesh that defines where smoke can exists called the domain and an emitter with a particle system where the smoke originates form that is called the flow object. This simple setup can be seen in figure 55.

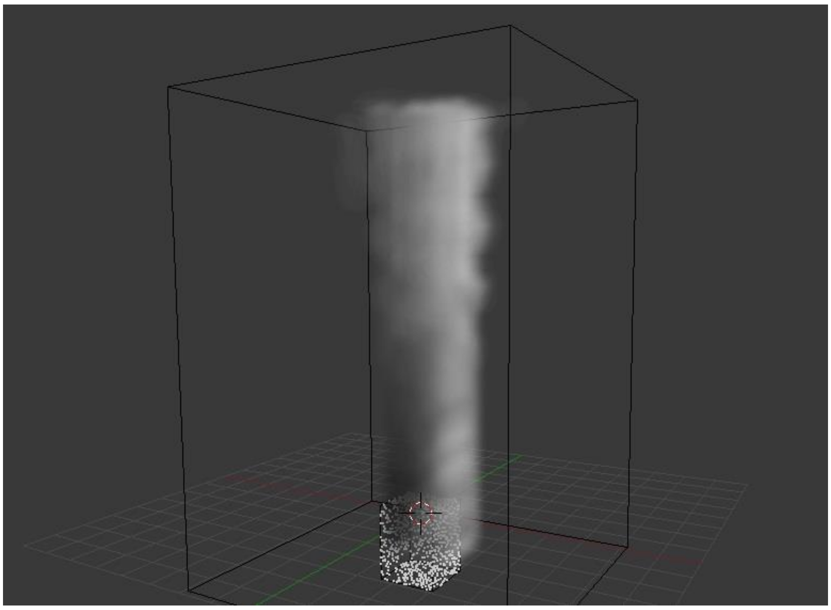


Figure 55. the most basic smoke simulation setup

The characteristics and appearance of the smoke is largely dictated by the domain and its material. The basics settings are as follows:

- Resolution: This determines the smokes level of detail. A resolution low resolution smoke can be calculated in seconds high resolutions like we would be suing would take considerably more.
- Time Scale: This can be used to speed up or slow down the smoke part of the simulation
- Border Collisions
  - Vertically Open: Smoke disappears when it hits the top or bottom part of the domain.
  - Open: Smoke disappears when it hits any wall of the domain
  - Collide All: Effectively makes the domain a closed box where no smoke can escape.
- Temperature and Density: How much Density and Temperature affect smoke motion. Higher Values make faster-rising smoke.
- Vorticity: Make the smoke more swirly.
- Dissolve: Allow the smoke to dissipate over time.
- Time: The speed of the smoke's dissipation.

- Slow: This option makes 1/Time instead of Time, making the smoke dissolve much slower. [10]

Now the aim was to use the smoke simulator to make a storm, a super cell/tornado type storm to be exact. This however soon proved to be much more difficult than first anticipated. First getting the smoke to do a smooth and consistent spin was quite hard as the force fields just would not give the wanted results. The solution was to use a spinning emmitter and then have the smoke inerith the spinning moting using the initial velocity multiplier. The emiter desing can be seen in figure 56 and the rendered result in figure 57.

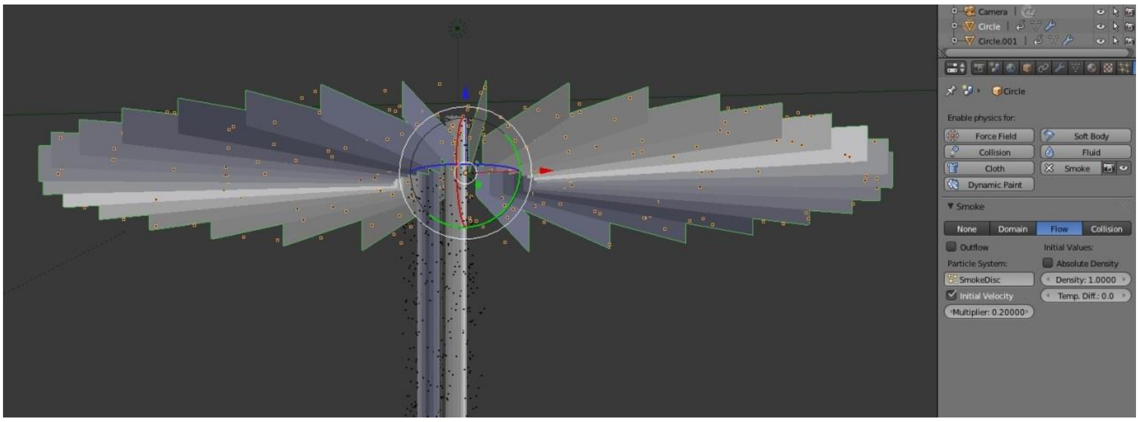


Figure 56. the rotating emitter

Figure 57. the rendered result of the smoke simulation

Here's where the second issue comes in. While the tornado looks good it's not the correct size. Everything in this project has been made in the correct scale.

This is supposed be a massive force of nature and sit in a scene with mover 2km draw distance yet this tornado is only about 90m tall. Initially this does not hit one as a big hindrance but as it was soon discovered the storm would just not scale up properly. Even with much higher resolutions once scaled up the tornado would lose all its crisp smoky properties and look like a blob. The solution for this was a rather unorthodox approach.

The complete storm animation was first rendered out as is with a transparent back ground like seen in figure 57.

This animation was then applied to simple plain that would play back the animation in the main outdoor scene as seen in figure 58.

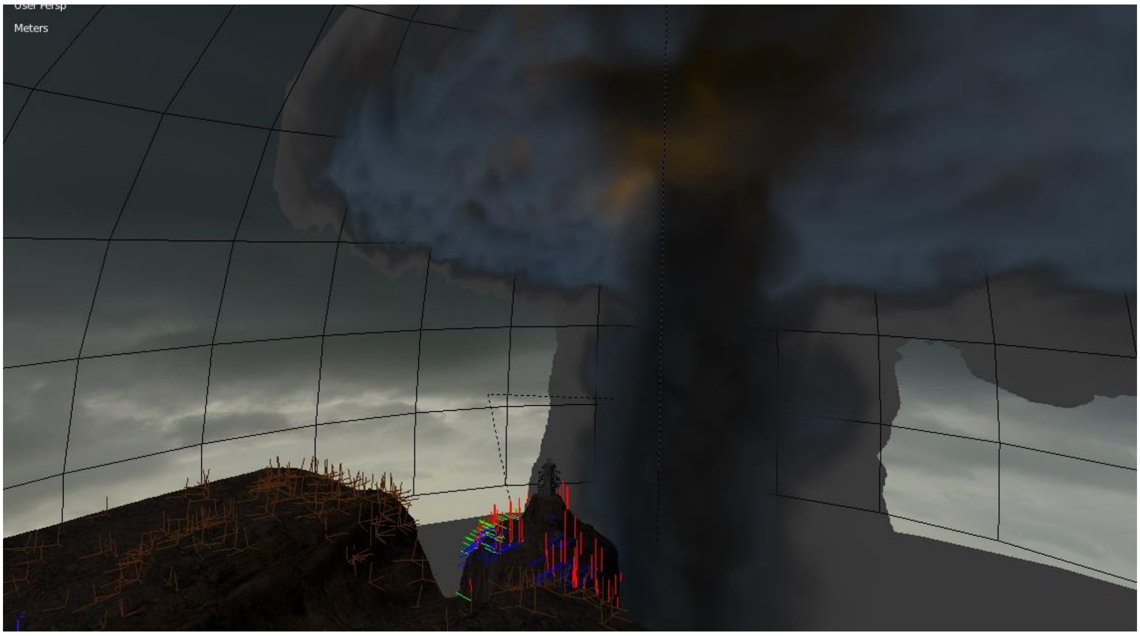


Figure 58. the strom animation applied to a plain in the scene

As the storm would be shot mostly from a perpendicular angle so the flatness and lack of parallax would be all but impossible to notice.

2.10 Lightings

While all the 3D render images that have been shown up until now have had lights and most certainly have been rendered we have yet to really address the subject.

So lest start with the basics. Blender has 5 diferent lamps:

- Point is an omni-directional point light source, similar to a light bulb.
- Spot is a directional point light source and its light is directed in a cone.
- Area is a source simulating area producing light, as windows, neons, TV screens.
- Hemi simulates a very wide and far away light source, like the sky.
- Sun simulates a very far away and punctual light source, like the sun.

[10]

For standard 3 point lighting systems the spots are a good choice and they can also be very fast as you can use buffer shadows instead of ray traced shadows with them. The Buffer shadows also allow for some special features perhaps the most interesting being halos. What the halo option essentially fakes the presence of small dust parcels that would refract light in high contras situations, this effect is called volumetric light. In figure 59 we can see a demonstration of the effect.

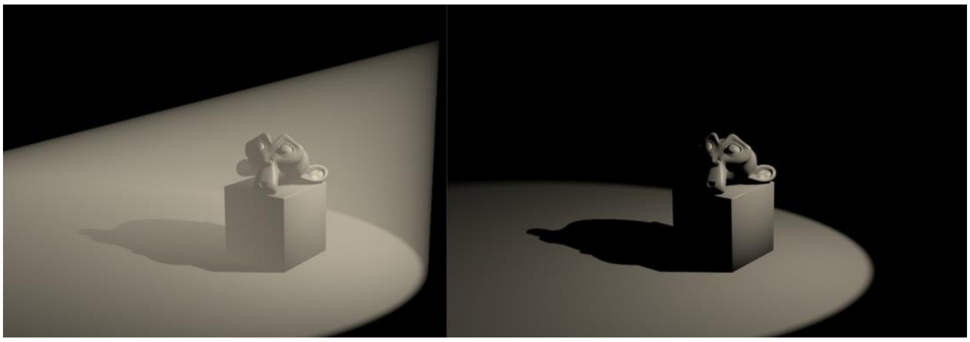
. 

Figure 59. a spot light with halo enabled on the left and the same spot with halo disabled

This is the same feature that was used to make the light rays seen in figure 60 back in the sets, props and linking section.

Many times when doing lighting is actually a good idea to use slightly different colored lights that sit on different sides of the color wheel.

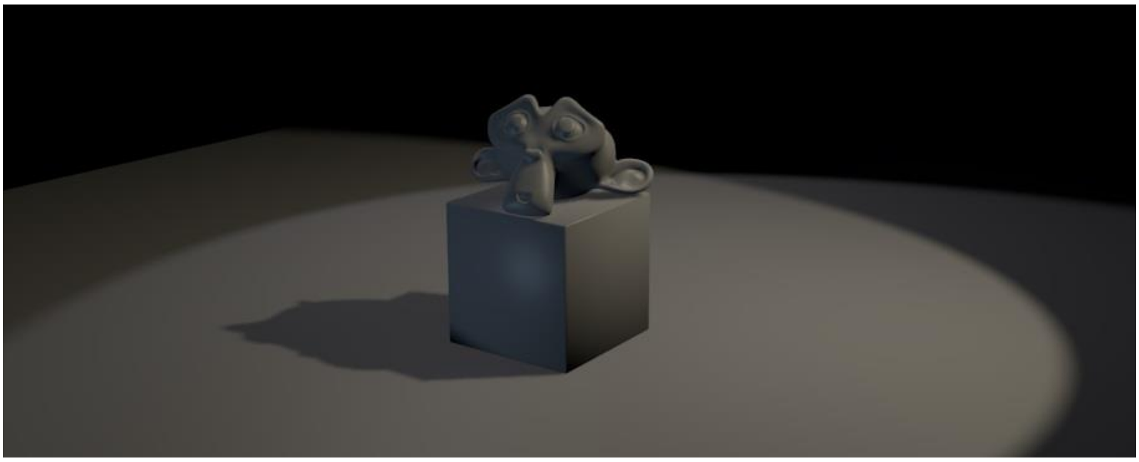


Figure 61. a demonstration of three point lighting

Here in figure 57 we have a basic three point system with a slight blue tone coming in from the left while the main light or key light as it’s called is tinted in a more yellowish color.

Now lighting a true interior and not just a studio setup is much harder. This is where Environment Lighting that provides light coming from all directions can come in real handy.

So let’s look at why interior lighting is much harder. For that let’s put Suzanne the monkey head in room with a window. We put a light source to shine in form the window. What we get is seen in figure 61.

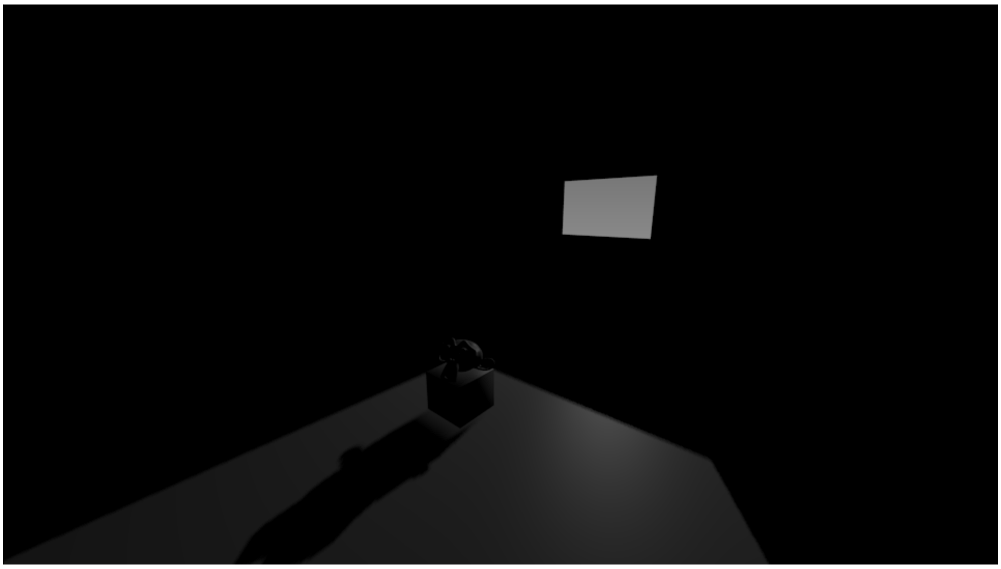


Figure 62. a room rendered with light just entering though the window

The room’s walls are totally black. This doesn’t look realistic at all does it? In the real world when light enters a room from a window it gets reflected around the room lighting up the walls but with Blenders rendering engine this does not happen so we have to fake it by adding a second light that will fill in the missing parts. What we then get is seen in figure 62.

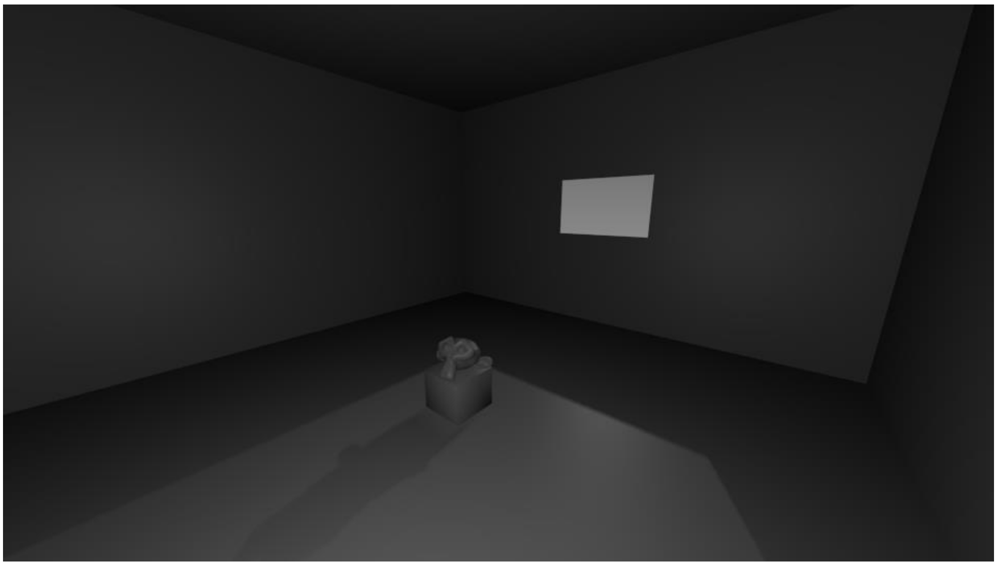


Figure 63. a room with light entering form the windows and a helper light

This simple approach of faking reflected light is what was used to light the majority of the scene in this animation.

2.11 2D sequences

Like mentioned early on in this theses there was also a 2D component in this 3D animation. The goal was to incorporate some 3D elements in to 2D images as well as to animate the stills theme selves in a slight way. Let’s see how the flat plain image in the figure 63 was turned into what’s seen in figure 64.

Figure 64. the original 2D image

Figure 65. a still form the animation

First off the sill image was made with this in mind so it has a lot of layers in its .psd file. These layers where then separated and brought into Blender as individual plains. The image plains where then made shadeless that is to say the lamps would not affect them. The plains where then spaced out so that a camera could “fly” between them. A particle system was added to give additional depth. Lastly the plains that would have movement where subdivided and rigged like a normal model would be. Figure 65 shows what the scene looks like in blender and how the different layers are spaced out.

Figure 66. the 2D plains as they are seen in the blender viewport

3 POSTPRODUCTION

Post production is generally defined as what is done after the raw footage has been recorded. In Blender I would argue that post production starts as soon as the frame has finished rendering. Why I say this is because in blender you can apply a large array of effects after the frame has finished. Effects like glares, motion blur, depth of field and many others. Many of these effects can only be applied in blender after the rendering is complete but additional information like motion vectors are still in the system memory though so the line is blurry.

3.1 Blender compositor

We have actually already used nodes but now we will use composite nodes. Compositing nodes can be used to do man many things so perhaps good to start with a really simple example. In FantasyCraft's Tale of the orbs game orbs are understandably in an important part of the story so let's take a look at the unaltered render in figure figure 66.

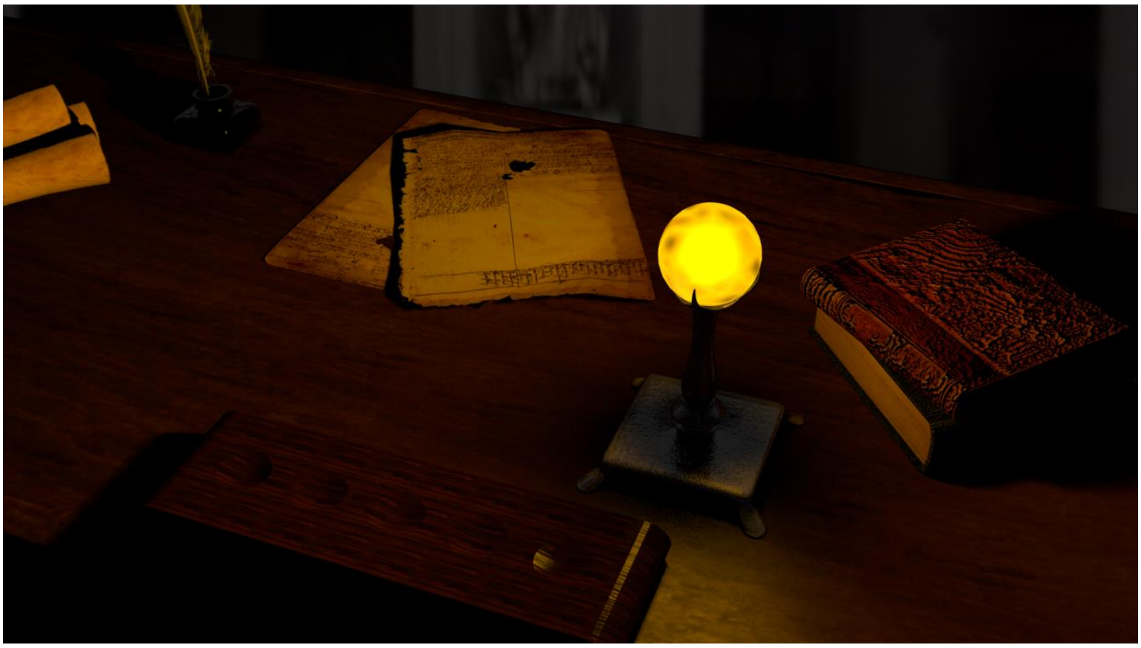


Figure 67. the basic rendered scene

It’s not bad the orb is right and shines light on to its surroundings with the help of a lamp but it lacks some visual impact. Letts see what we can do to improve it with some compositing. So first of lets go to render layer and enable the emit tick box. What this does is tells the render layer to deliver us a separate emissions pass as seen in figure 67. This emissions pass only has things in it that’s material has some amount of emit enabled in its material shadings options.

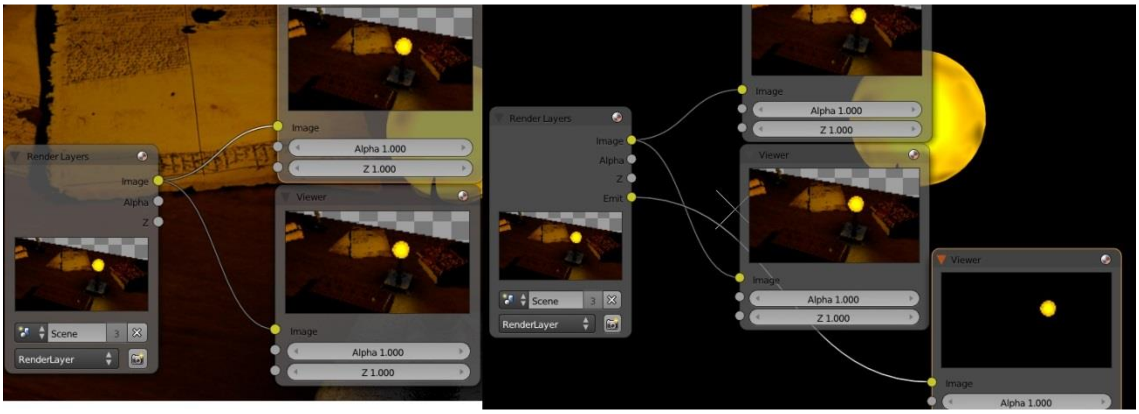


Figure 68. the standard render layer node on the left and the new render layer node with the emit output on the right

So how do we use this new information? In this case we are going to give the orb a glowing effect witch is quite easy as we just plug in the emit output to a fog glow node and then add that glow to the original image with an add node like see in figure 68.

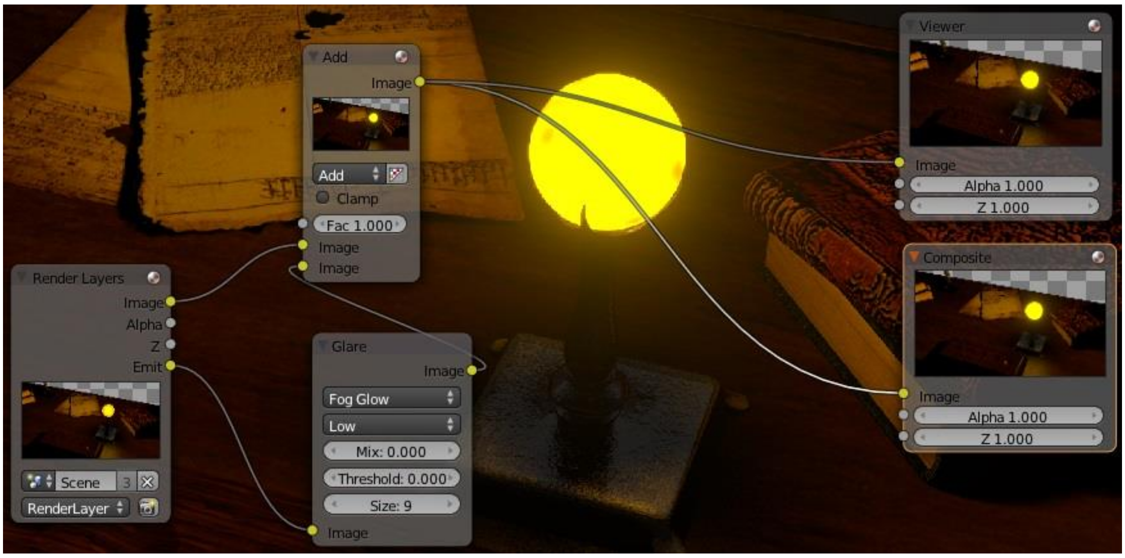


Figure 69. what the completed fog glow node network looks like

But this is only composting at its very simplest level. The balcony scene where the two characters overlook the storm the node network is considerably more complicated as seen in figure 69.

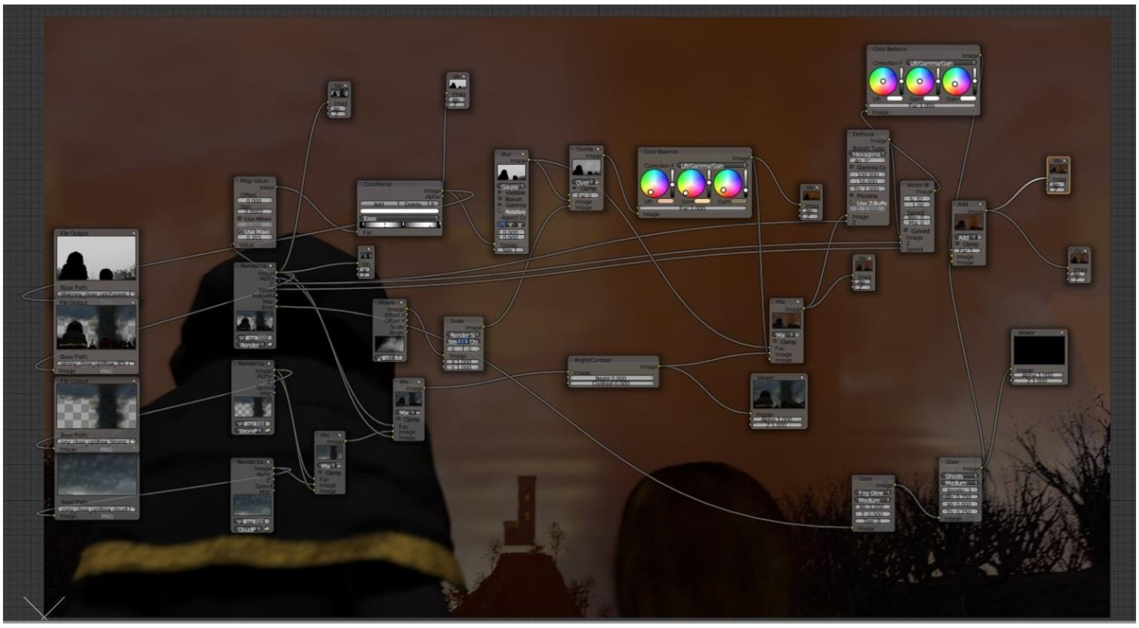


Figure 70. the node network used in the rendering of the balcony scene.

This node network does not simply do one thing but a whole range of things. First off it uses three independent render layers. This so that additional content can be injected in between the different layers as well as to give greater control over the color correction.

Probably the most notable and complicated part of this network is the mist system. The way this mist system works is that first the z-map (depth map) is pulled out of the main render layer. That z-map is then adjusted with the map value node and the color ramp node so that we get an image where the wither the color the thinker the mist we actually input a video of some smoke to give the mist some variation and life. This mist is then overlaid back on to the rest of the image using a z-buffer cut off so that the mist never goes over the characters thus appearing to be further away.

3.2 After effects

And so the footage arrives at its final destination; after effects. This is where the rendered footage is edited to one continuous video. This is also where the final visual effects are made.

Actually saying the footage arrives in after effect as ever since the first test render finished things have been constantly updated into the file that once was the story reel. As the dialog got recorded it replaced the crude track and so slowly the crude drawing got replaced by cured animation and after a while the crude animation got replaced by better animation.

So then with these now improved animations the transition also honed in. Some shots needing more work than others but there is only one shot that can really be called effects shot and that’s the very last scene where The First pulls out an glowing orb. The glow itself was already done in Blender but in this shot the game log will be presented. Many different approaches where tested in how the log would be unveiled but the one that finally go selected was the one where the glowing orb is morphs in to the orb in the logo. A lustration of the sequence can be seen in figure 70.

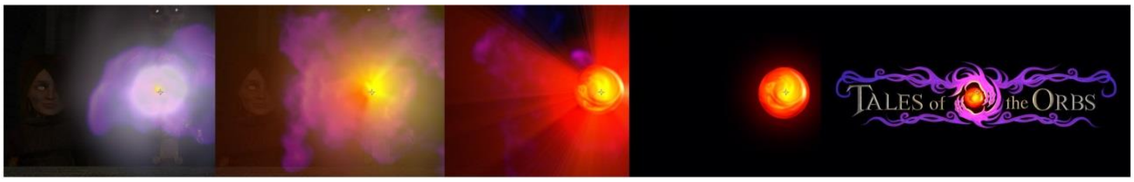


Figure 71. the reveal effect as it progresses

3.3 Audio

So all the composting and visual effects magic over so is the trailer done? No not quite yet. There is still the matter of getting good audio on to the video and audio matters. The trailer will have a proper composed score courtesy of FantasyCraft but the rest will have to be built from the ground up. The score as been around for a while already and it was used to help refine the pacing in the same way the dialog was. Now the both need to be edited and mixed so they work in harmony. First the voices are synced up to the video. Then their levels are adjusted so that the volume matches the setting in the video. Then the sound is paned to get the stereo sound effect. As the last step for the voices effects are added. Either echoes to help sell the illusion that the characters are talking in a spacious stone castle or some more radical effects like pitch distortion to get away from the more mundane speaking voices.

After the voices are done the rest of the less important sounds could be added on. Some of the sounds like battle cries where take form free to use source on the internet. It was quickly evident though that there wasn’t that much good free stuff available.

This is where the interesting and quite peculiar art of Foley comes in. Foley is really just a fancy way of saying making sound in a studio suing everything and anything. That is to say it’s easy as it’s anything but. What was surprising however was that even something as simple as taping the desk with a finger can produce foots steps that sound better than some of the free options. This is in some part due to the fact that you aren’t trying to force a preexisting sound to your video but rather the sounds you produce are synced to the movements of your video when you are making them. By layering sounds and using editing to distort the sound even simple sounds can be made to produce acceptable results.

SUMMMARY

So that was my journey through the production process. While I achieved many of the goals first set out and have had learned an incredible amount through all the setbacks and problems I still feel there is more to do. I will not lie there were times when progress was slow and my motivation was running really low, even so progress never sopped and that is one little thing I do take some pride in. That being said I came face to face with my own limitations namely very poor time management. I would even go as far as to rate is as my number one flaw. I just plain and simple can't do it right. I'm always underestimating the work load und overestimating my own vigor.

So what about the trailer then and the folks at FantasyCraft? While the trailer was "finished" in regards to this theses the people at FantasyCraft are personal friends of mine and both they and I agree that the trailer still has untapped potential. In light of this I'm inclined to continue to improve the trailer so perhaps even before FantasyCraft decides to unveil their game there will be a trailer 2.0 done.

SOURCES

[1] D. Roland Hess 2009, Animating with Blender

[2] Film production http://en.wikipedia.org/wiki/Filmmaking

[3] Trailer http http://en.wikipedia.org/wiki/Trailer_(film)

[4] Three-act structure http://en.wikipedia.org/wiki/Three-act_structure

[5] John Hart 2008, The Art of the Storyboard

[6] A WALK IN THE VALLEY OF THE UNCANNY http://www.damninteresting.com/a-walk-in-the-valley-of-the-uncanny/

[7] Jonathan Williamson 2012, Character Development in Blender 2.5

[8] 3D Defined - What is 3D? http://3d.about.com/od/3d-101-The-Basics/a/3d-Defined-What-Is-3d.htm

[9] Anatomy of a 3D Model http://3d.about.com/od/3d-101-The-Basics/a/Anatomy-Of-A-3d-Model.htm

[10] Blender 2.6 manual http://wiki.blender.org/index.php/Doc:2.6/Manual

[11] Light Probe Image Gallery http://www.pauldebevec.com/Probes/

[12] Three Layer SSS in Blender Demystified http://bensimonds.com/2010/05/31/three-layer-sss-in-blender-demystified/

[13] Hair Strand Rendering http://www.blender.org/development/release-logs/blender-240/hair-strand-rendering/

[14] Fur Rendering http://www.blender.org/development/release-logs/blender-246/hair-and-fur/

[15] Hair in Blender http://bensimonds.com/2011/02/17/hair-in-blender/

[16] Skeletal animation http://en.wikipedia.org/wiki/Skeletal_animation

[17] Tony Mullen, Bounce, Tumble and Splash! Simulating the Physical World with Blender 3D

[18]Lighting in Blender internal http://home.elka.pw.edu.pl/~mskrajno/tutorials/lighting/general/General.html