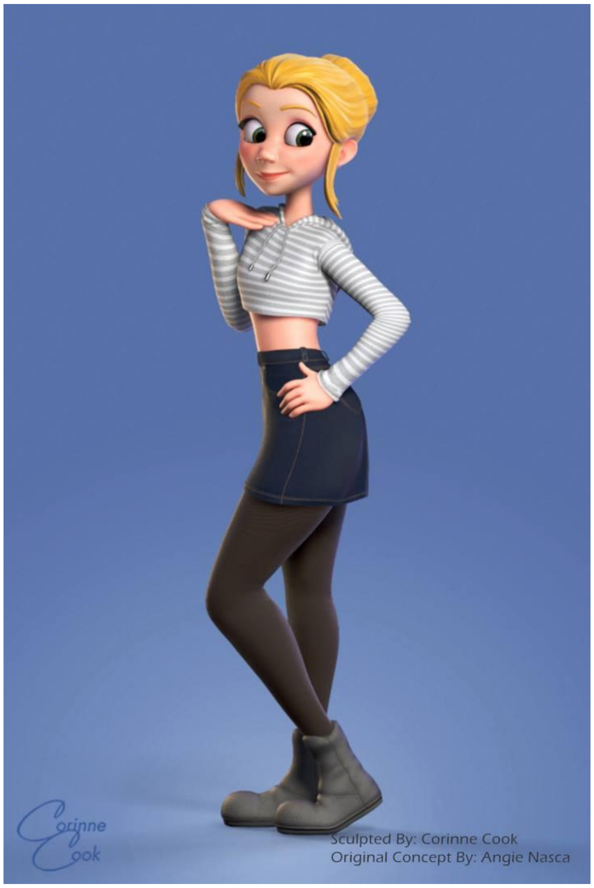


How to create stylized 3D characters for low-end devices

B.Brink

405511

Creative Media and Gaming Technology

Art and Technology

Saxion University

Information

How to create stylized 3D characters for low-end devices

Student:

Name: Brendan Brink

E-mail: brndnbrnk@gmail.com

Telephone: +31 625580749

Student number: 405511

Education: Art and Technology

School: Saxion University

Year: 2018/2019

Company Supervisor

Name: Keesjan Nijman

E-mail: keesjan.nijman@gmail.com

Telephone: +31 618569240

Company: Conceptlicious

Saxion Coach:

Name: Herman Statius Muller

E-mail: h.statiusmuller@saxion.nl

Telephone: + 31 623470579

School: Saxion University

Date of Submission: 22nd January 2018

Abstract

This research paper is about how to create stylized 3D game characters that are intended for low-end devices such as smartphones and tablets. The research is commissioned by the company Conceptlicious in order for a project for their client and for streamlining the workflow in future projects. This researched dives in to pipeline, workflow and design principles needed for stylized 3D character creation. This research provides in-depth knowledge about all the different stages in the pipeline for creating 3D game characters. The result shows that sculpting a high-poly model and then retopologizing it to a low-poly model is the most current and optimal workflow for creating 3D characters for low-end devices, however, this may depend on various aspects like the style of the game, the target device and the gameplay of the game. This research also shows that the biggest performance issue for low-end devices is caused when a 3D model has a high poly count and high-resolution textures. When creating the retopology and UV maps this researched proves that these should be done manually if good deformation for animation is required.

Preface

Prior to the report, I would like to express my gratitude to those who made this project possible, helped to resolve issues and uncertainties, and the university for providing the graduation subject.

I would like to thank Conceptlicious's employees like Rene Stam, Keesjan Nijman, Adam Butler and Arnoud Poll Jonkers for giving me the opportunity to work with them on this project and use it for my thesis. Without them, this whole project would not be possible. I also would like to thank Keesjan Nijman and Adam Butler for providing feedback on my thesis and guidance to resolve any issues and uncertainties.

For providing guidance during the graduation process I would like to thank my graduation coach Herman Statius Muller from Saxion University. And finally, I would like to thank all the interns for all the hard work they have given during this project and for living up the workspace.

Glossary

3D model	A mathematical representation of a three-dimensional object within computer software.
3D modeling	The process of creating 3D models.
Base mesh	A low-resolution 3D model which can be used as a starting point when 3D modeling or sculpting.
Bottleneck	A bottleneck within an application is when the capacity of the application is severely limited by a single component.
Decimate	A process of reducing the polygon count with minimal shape changes.
Deformation	The action or process of deforming or distorting.
Game Asset	Any single content intended for a game, like 3D models, textures, animations, and audio.
Game character	A character within a video game.
Game engine	A software-development environment designed for making video games.
Game Performance	How well a game performs on a device.
Gameplay	The specific way in which players interact with a game.
Pipeline	The processes in which an asset goes through before it is finished.
Plugin	A software component that adds extra features.
Software	A program that is used to operate computers.
Stylized	If something is stylized, it is represented with an emphasis on a particular style, especially a style in which there are only a few simple details
Target device	The device on which an application or game is intended to run on.
Tessellation	Tessellation is a process to manage polygons by dividing them into a suitable structure for rendering.
Triangulate	The process of dividing something into triangles.
Workflow	The steps in the progress of creating something.

Table of contents

1 Introduction

The technology for creating 3D game characters has developed a lot since the introduction of 3D games. With modern technology, a game character can look so real that it's almost indistinguishable from real-life, in contrast to the classic game characters like Lara Croft from Tomb Raider and Link from The Legend of Zelda in the early days of 3D games. However, while modern game consoles and computers can handle these realistic graphics, the technical restriction of low-end devices such as smartphones and tablets make it so that they are unable to handle these graphics. Furthermore, 3D characters are often not realistic at all and are stylized to look more like cartoons, this is even more so with low-end devices because of their restriction. This paper serves as a research document for the different techniques and methods in creating 3D characters for low-end devices. It covers the reason for this research, the general pipeline, the workflow for creating 3D characters, the design principles for creating stylized characters, how 3D characters can be optimized for low-end devices, and what the optimal tools and techniques are for creating 3D characters.

1.1 Reason

The biggest reason for this research is that I really like character design especially stylized and cartoon characters. While I have created a lot of 2D characters I only scratched the surface when it comes to creating 3D characters, especially when they need to be ready for games. I wanted to learn more about the design principles for character design, the pipeline, the workflow for creating 3D characters, and the techniques and methods for creating stylized 3D game characters. At the company Conceptlicious they make serious games for multiple companies, institutions, and organizations. Their clients use these games for training simulations, education and marketing purposes. One of these clients is ZGT hospital and they asked Conceptlicious to develop a prototype for a serious game for their patients who have been in surgery. Patients aren't motivated to do their physical exercises which result in all sort of health issues after they have been discharged from the hospital. The consequences of physical inactivity are a decrease in endurance, muscle mass and strength and cognitive functions. This results in an increased risk of complications and functional decline. According to the research of the hospital, the cause of physical inactivity includes an inadequate mobility-friendly environment and inadequate knowledge, mindset and actions among patients and relatives as well as healthcare professionals. The prototype of the game should help with motivating the patients to do their exercises, and well designed and developed characters can help with keeping people engaged in a game.

1.2 Preliminary problem statement

The goal of the client is to have a proof of concept for the serious game. Character creation is a big part of developing a game, therefore further research is required on how to create characters for games. Since this game will be played on a smartphone or tablet, there are limitations on the methods and workflow for creating 3D characters which need to be considered. Some methods and workflows still might be the same as creating 3D characters for high-end machines so those still need to be examined as well.

2 Theory

This chapter is a summary of the theory about the general pipeline and workflow for creating 3D characters based on literature found online. The pipeline for creating a 3D game character can differ depending on several factors like, the art style of the game, if the character is the main character or not, the technical limitations of the device the game is played on, how the character will be animated, if the character is a hard surface character like a robot or if it's going to be an organic character. This chapter will explain the general pipeline for creating a 3D game character before it's ready to be implemented into the game engine, however, the workflow may differ between game studios because of the different software the studio uses.

2.1 Concept art

The first step in creating a 3D game character is making concept art, where there are several documents made by a concept artist called model sheets. The model sheets depict the character from different angles which helps a 3D modeler with creating the character. The 3D modeler can use these sheets as a reference within the software he or she uses. The modeler will attempt to recreate the character in 3D as accurate as possible, but sometimes the 3D modeler could make changes either to increase the appeal of a character by using certain design principles or because of technical restrictions. (Pluralsight, 2014; Pluralsight, 2015; Wikipedia, 2018)

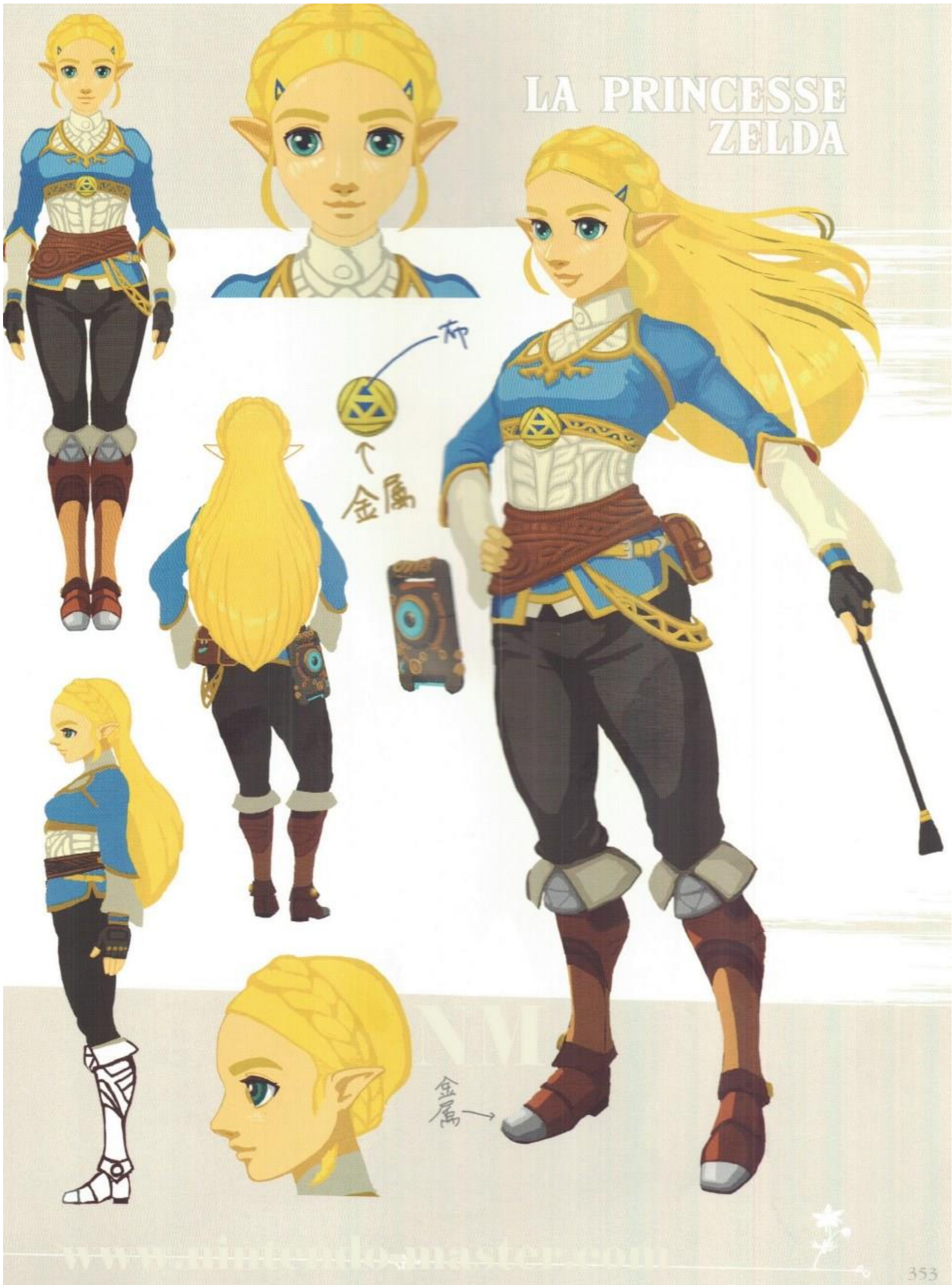


Figure 1- Character model sheet from Zelda Breath of the Wild

2.2 3D Modeling

The method of creating a 3D model can vary depending on which program is used, what kind of model it is, and the preferences of the 3D modeler. For example, when creating hard surface models like cars, motors, machinery or things man-made in general, techniques like box modeling or edge modeling are often used with software like Autodesk Maya, 3ds Max, and Blender. With these programs you can create polygon models, also known as meshes, by creating and manipulating polygons, edges, and vertices. Polygons, also known as faces, are created when three or more edges are connected to each other. The face is what fills up the empty space between the edges and thereby making it visible. Vertices are the smallest component of a mesh and it’s simply a point in three-dimensional space. By connecting two vertices or points together you create an edge. Polygons with three vertices are known as triangles, polygons with four vertices are called quads and polygons with more than four vertices are known as n-gons. (Fabian, 2017)

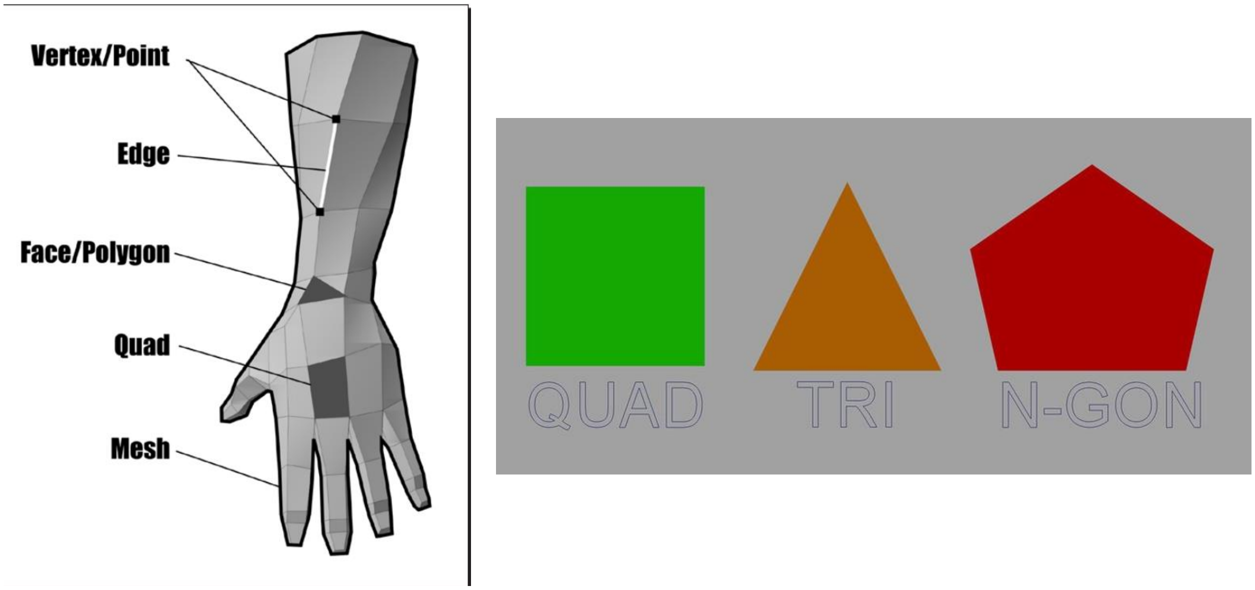


Figure 2 - A 3D model and its components and examples of quads, triangles, and n-gons.

With box modeling, the modeler starts with a primitive object like a cube, cylinder or sphere and then adjust the shape until the desired appearance is achieved. Often the modeler starts with a low-resolution also known as a low-poly model, which has as few polygons possible to create a certain shape and keeps subdividing and refining until the desired detail is achieved creating a high-poly model.

Edge modeling is another method for creating a polygon model and is usually used in conjunction with box modeling. With edge modeling, the modeler creates polygon models by placing loops of faces and filling in the gaps in between. This method is mostly used when doing the retopology of a model, which will be explained later in this chapter.

Another method for creating a 3D model is by digital sculpting with programs like ZBrush, Mudbox or 3D coat. With digital sculpting, meshes are created by using digital clay which the modeler can mold and shape using a drawing tablet almost exactly like a sculptor would do with real clay. This method is usually done when creating organic models like characters and creatures and it allows the modeler to work with high-resolution meshes which can contain millions of polygons.

Another method for creating 3D models is using 3D scanners or photogrammetry, where real-life objects are scanned, and the raw data is used to generate a mesh. This method is usually used when the models need to be as realistic as possible. The downside of this method is that it can’t be used when there is no real-world object to scan, like aliens, spaceships or cartoon-like characters. (Slick, 2017)

2.3 Retopology

Topology is an important aspect of a 3D model. It refers to the layout of a model, or in other words, how the polygons are distributed on a 3D model. Good or clean topology is when all the polygons are evenly distributed across the model. A model has to have good topology, especially when used in games because they are rendered in real time and therefore need to have as few polygons as possible. Bad topology can cause several problems like low frame rates, unpredictable subdivides and bad deformation. Depending on what the model is used for, the requirements for having a good topology can differ per 3D model though. For example, using triangles is acceptable when a model is static and won't deform. However, if a model is going to be animated and deformed, it's advisable to avoid triangles as much as possible, because this can cause bad deformation. Using n-gons should be avoided at all times. (Pluralsight, 2014b; Polycount Wiki, 2017; Taylor, 2015a)

For 3D characters, the way the polygons are distributed is very important. For instance, more polygons should be used in places that are going to deform a lot, like the face and joints of the character. When using a more traditional method like box modeling and edge modeling while creating a character, the modeler has to keep topology in mind. The loops of quads have to be distributed in certain ways around the muscles in order to subdivide and deform well. (Amin, 2013; Slick, 2017)

The current workflow is to sculpt the high-poly model first, not worrying about topology at first and to retopologize the high-poly model later. When retopologizing, the modeler can use edge modeling techniques on top of the high-poly mesh to draw polygons manually, thereby creating the low-poly mesh with the right topology. Almost every 3D software has retopology tools to manually draw polygons on the high-poly mesh, but some are more efficient than others. Some 3D software like Zbrush and 3D coat have options to automate the retopology process, though these methods are usually not good for characters that need to be animated. (Spencer, 2011)

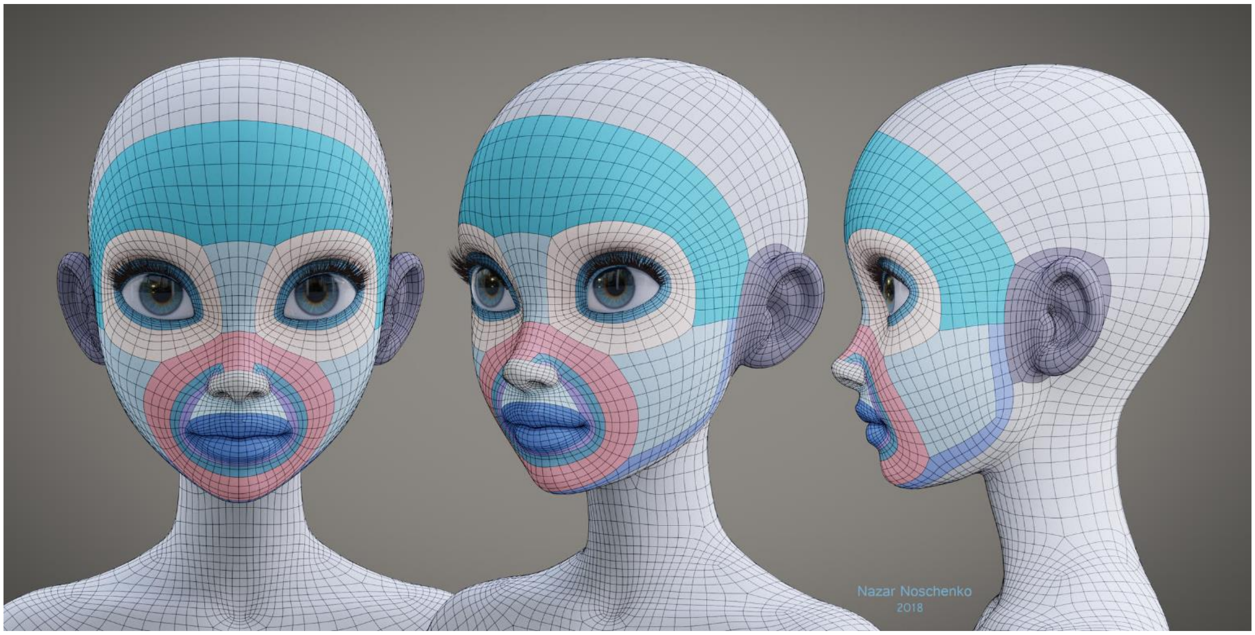


Figure 3- Example of clean and good topology of a face.

2.4 UV mapping

When the low-poly model is finished, it is ready to be UV mapped. In order for a 3D model to look like something other than just a three-dimensional shape, it needs to have a texture map. A texture map is a 2D image with more texture details like color, shading, and specular information for the 3D model. In order for the computer to know how to project the texture map onto the 3D model, the polygons need to be mapped into a UV map. The process of doing this is called UV mapping. The letters “U” and “V” stand for the axes of the 2D image since “X”, “Y” and “Z” are used for the axes of the three-dimensional space. All polygons, vertices, and edges of the 3D model are stored in the UV coordinates of the UV map, and when a texture is applied on the UV map, it will appear on the corresponding coordinates of the 3D model.

A good analogy is like cutting up a box at the edges and unfold it, making it flat. A modeler can do the same with a 3D model, creating cuts and seams in the model and unfold it into the UV map.

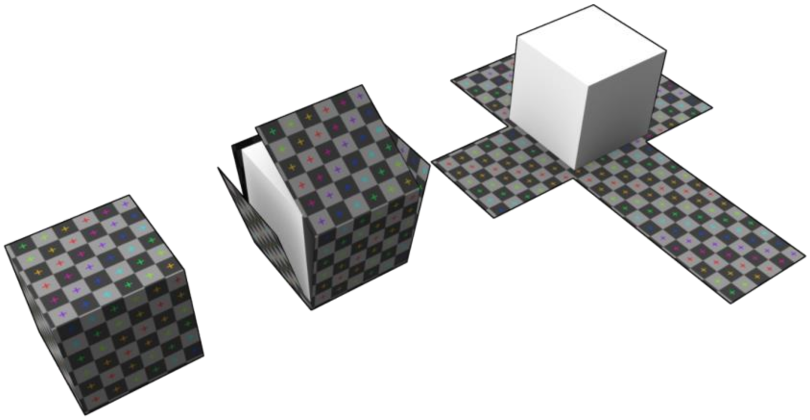


Figure 4- Unfolding a 3D box.

There are several ways creating a UV map automatically, but when the 3D model is too complex, this causes the projection to be distorted and the textures to look stretched or squashed, and it's usually not very organized and has a lot of unnecessary seams. For this reason, when creating a UV map for complex 3D models like a character, it's done manually. A UV map doesn't have to be a single piece like in the box example but can contain several pieces called UV islands. Having more islands generally means that the textures will be less distorted, but doing this creates more seams and it's advisable to keep the seams as low as possible with acceptable distortion because visible seams don't look good with a texture. (Blender, n.d.; Paulino, n.d.; Pluralsight, 2014c)

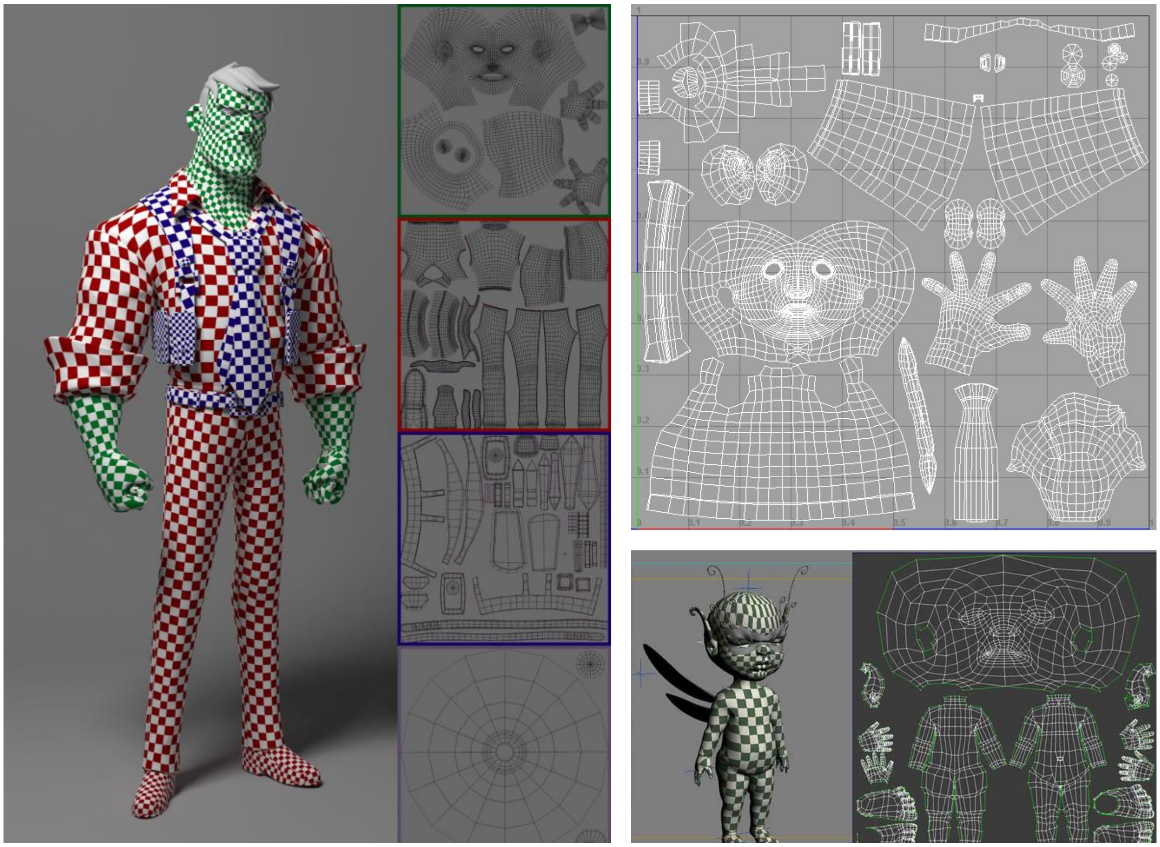


Figure 5- Examples of UV maps for characters.

2.5 Texturing

After the UV mapping is done, the 3D model is ready to be textured. There are multiple ways for creating a texture map, like hand painting it in a digital painting program like Photoshop, or painting it on directly with a 3D painting program like 3D Coat or Substance Painter. Another way of creating a texture map is a process called baking, where the details of the high-poly model created earlier are projected on its low-poly model counterpart. The baking can be done in programs like Substance Painter or Marmoset Toolbag 3. (Fisher, 2013; Giroux, 2016) There are several texture maps created with the baking process, each with their own purpose and by combining each individual map they create different effects. The most used textures maps are color maps, bump or normal maps, height maps, and specular maps. Other maps such as ambient occlusion maps and emission maps are also commonly used and they affect the way a 3D model is lit.

Albedo and diffuse maps are color maps that contain the colors of the surface of a 3D model. An albedo map usually only contains the base colors of the model with no lighting information and is used in combination with other maps to create lighting information like highlights and shadows. A diffuse map already contains the lighting information and can therefore sometimes be used as the only texture map for a model.

There are several types of specular maps, which control how reflective or glossy a surface is. A specular color map simply controls the brightness of the highlights, but can also be used with RGB values to colorize the highlights for surfaces with more complex reflective properties. A gloss map is combined with a specular color map and controls how wide or narrow the specular highlight appears, making it a glossy or matte surface.

A way to create extra detail on a low-poly model without adding extra geometry is to use maps like bump maps or normal maps. With these maps, extra detail is created by creating an illusion of depth using a simple lighting trick. Bump maps are an older version which only contains grayscale images with different gradients of black and white, where white areas will appear as if they pull out of the surface and black areas appear to be pushing into the surface. Normal maps are a newer version of bump maps and also create the illusion of depth without adding extra geometry. While bump maps only use grayscale values, normal maps use RGB values that correspond with the X, Y, and Z axis in three-dimensional space. There are two kinds of normal maps, tangent-space normal map, and object-space normal maps. Each has their own advantages and disadvantages, but the most important one is that tangent-space normals are better for meshes that deform. Since normals maps are very difficult to create and edit using a digital painting software, these maps are usually baked. Heightmaps or displacement maps are similar like bump maps and only use grayscale values. The difference with these maps is that instead of creating an illusion of depth, they actually modify the shape of the model.

Ambient occlusion maps are grayscale images which provide information about what areas of the model receive high or low indirect lighting. White areas receive full indirect lighting and black areas none. Sometimes ambient occlusion maps are combined with the color map to create shading information and like normal maps, these maps, are usually baked.

Emission maps can create an effect that part of the model is emitting light, without actually functioning as a light source. This could be useful for creating an effect like a glowing computer monitor or brake lights for a car. (Polycount Wiki, 2015; Polycount Wiki, 2015)

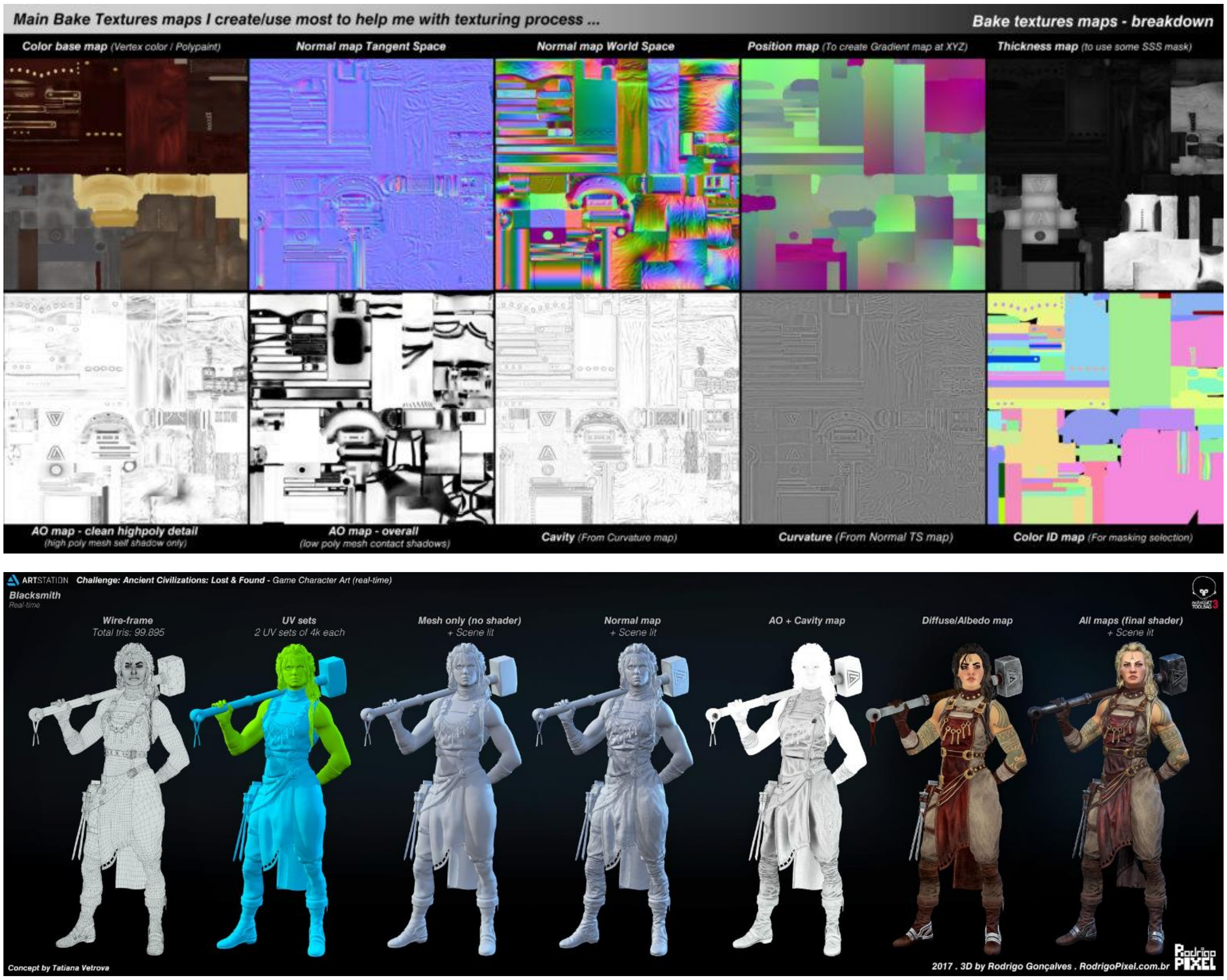

Main Bake Textures maps I create/use most to help me with texturing process ...
Bake textures maps - breakdown
Color base map (Vertex color / Polypaint)
Normal map Tangent Space
Normal map World Space
Position map (To create Gradient map at XYZ)
Thickness map (to use some SSS mask)
AO map - clean highpoly detail
(high poly mesh self shadow only)
AO map - overall
(low poly mesh contact shadows)
Cavity (From Curvature map)
Curvature (From Normal TS map)
Color ID map (For masking selection)
ARTSTATION Challenge: Ancient Civilizations: Lost & Found - Game Character Art (real-time)
Blacksmith
Real-time
Wire-frame
Total tris: 99.895
UV sets
2 UV sets of 4k each
Mesh only (no shader)
+ Scene lit
Normal map
+ Scene lit
AO + Cavity map
Diffuse/Albedo map
All maps (final shader)
+ Scene lit
Concept by Tatiana Vetrova
2017 . 3D by Rodrigo Gonçalves . RodrigoPixel.com.br
Rodrigo PIXEL

Figure 6 - Examples of different kind of texture maps and the effects they have on a character.

2.6 Rigging

Before a 3D character model can be animated it has to be rigged. Rigging is basically creating joints, bones, and controls to a 3D model so it can be animated. A rig can be either basic or complex depending on several factors like the technical limitations of the target device or how realistic a character needs to be animated. A character rig is made out of a digital skeleton made out of bones and joint. Digital bones and joints work almost in the same way as an actual human body. The joints are the articulation points where the model can turn around parts of its body. For example, an arm consists of three joints. One at the shoulder, one at the elbow and one at the wrist. Bones are the parts that connect the joints together but sometimes the term joint and bones are used interchangeably. Joints need to be placed in a certain hierarchy in order to work properly with inverse and forward kinematics. By using forward kinematics the characters rig will follow the hierarchal chain. This gives more control but also means that each joint needs to be positioned independently. With inverse kinematics, however, a joint within the rig's hierarchy can influence its parent's position. For example, if you would move the wrist joint with forward kinematics the elbow and shoulder joints need to be positioned separately, while with inverse kinematics the position of the elbow and shoulder joints will be calculated.

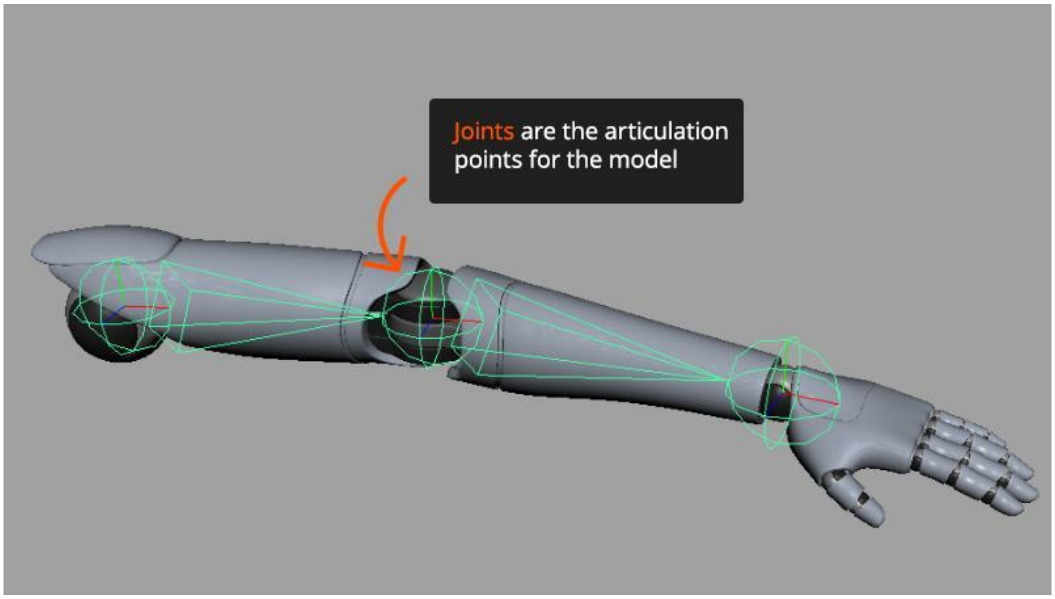


Figure 7- Example of joints.

A rigger has several tools to make it easier for the animator to control the characters rig. One of these tools is to create control curves, which are sort of handles that an animator can use for controlling all the different components of the rig. Another tool is to setup constraints so that joint only have a limited degree of freedom. This way a neck can’t rotate 360 degrees or arms and legs can only bend along one axis. To make it even easier for the animator, the rigger can utilize driven keys, which allows the animator to control multiple joint together using one control. An example would be one control to open and close all the fingers.

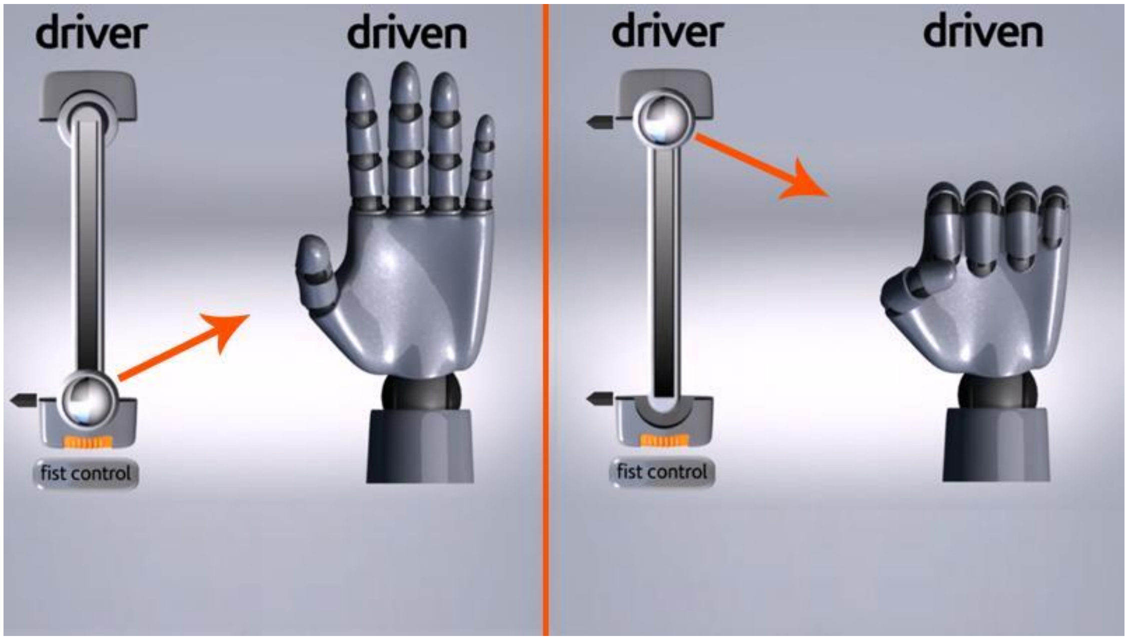


Figure 8 - Example of driven keys.

The rigger can also create blend shapes. Blend shapes are mostly used to control the face because this would be nearly impossible to control with joints and bones. Blend shapes allow the animator to change the shape of one object into the shape of another object by using a control slider. For example, there could be a control slider controlling one eyebrow to move up and down. Another tool a rigger can use are deformers, which can be used to manipulate a large section of a mesh. Squashing and stretching is an animation principle commonly used in exaggerated cartoon animations. With the use of deformers, squashing and stretching could easily be implemented in the characters rig.

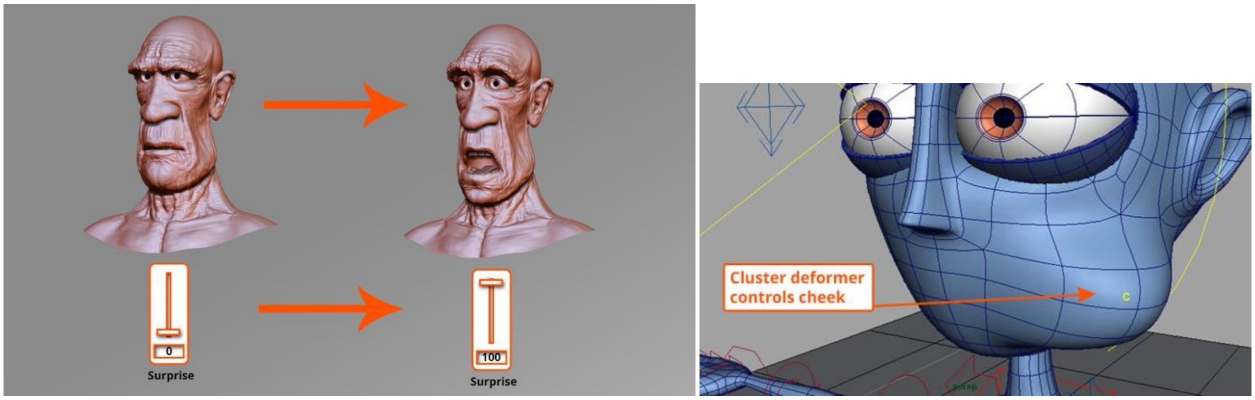


Figure 9 - Example of blend shapes.

The process of binding the rig to the 3D mesh so that it actually controls it is called skinning and after it's been skinned the mesh needs to be weight painted. Weight painting is the process of manually setting up how much influence a joint has on a specific part of the mesh so that it will deform well. For example, if the shoulder joint has too much influence on the torso it could make the torso deform unrealistically. (Pluralsight, 2014d; Slick, 2018b)

2.7 Animation

The last step in the pipeline before it’s ready to be implemented within a game engine is to create the animations. The tools to create animations within in a 3D program depends on which software is used, but almost every program makes use of time sliders and graph editors. With the time slider, an animator can preview the animation by moving through all the frames. An animation can be created by adding keyframes on different frames on the time slider. A keyframe contains values like position, rotation, and scale of an object. If two keyframes have different values, the software will transition between those two keyframes. The values of the frames between the two keyframes are calculated and created by the computer and these frames are known as tweens. (Autodesk, n.d.-a)

With the graph editor, an animator can edit the interpolation between keyframes. The interpolation between two keys is represented on a graph with lines called curves. These curves can be edited which creates different effects on the animation. For example, a straight line would mean that the interpolation is linear, creating a linear change of values between keyframes. If the line is curved, this would create a non-linear change of values between keyframes. (Autodesk, n.d.-b)

The most important principles of animation are created by Frank Thomas and Ollie Johnston in “ The Illusion of Life: Disney Animation". In total there are 12 principles that help with creating animations for characters. The first principle is timing and spacing, where timing refers to the time between two keyframes and spacing refers to the difference between those two keyframes. An example would be a ball moving from one point to another, where timing is how long the ball takes to reach the other position, and spacing would be the distance it travels between those two points. If the timing would be the same but the spacing would be different, the ball either moves slowly because the distance is short or would move fast because the distance is large.

The second principle is squash and stretch and refers to the flexibility of an object. Squash and stretch also happens a lot in real life but is mostly used in exaggerated cartoon animations as mentioned before. A great example that is often used is a bouncing ball, where if the ball is falling or bouncing back up the ball would be stretching, and when the ball hits the floor the ball would be squashed. The squash and stretch principle is often used to exaggerate expressions and movement on cartoon characters.

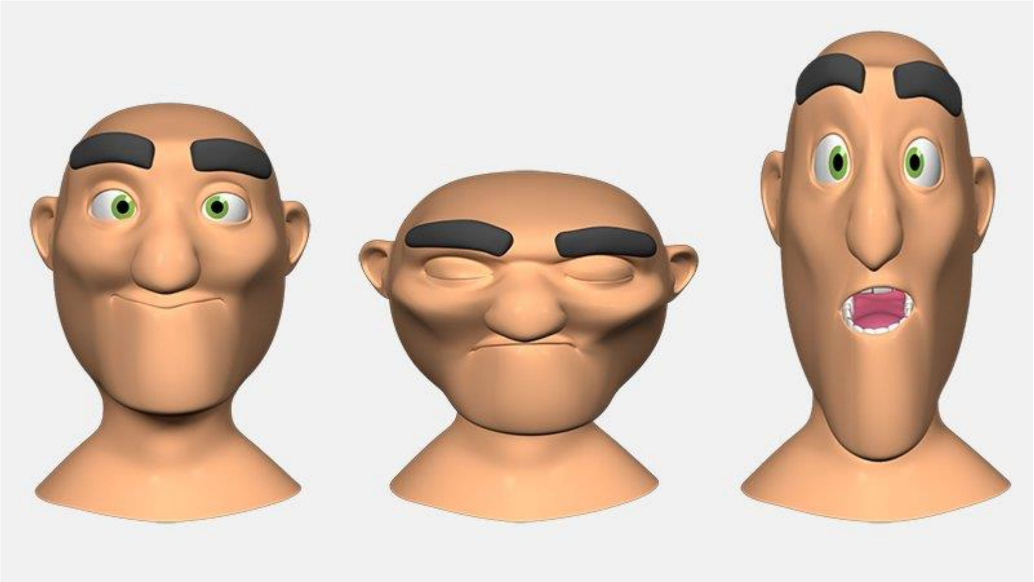


Figure 10 - Example of the squash and stretch principles used on a face

Exaggeration actually is one of the twelve principles of animation. Exaggeration is used to push certain movements giving it more appeal. Appeal is also one of the principles and by adding more appeal to a character the audience can relate or connect more with that character. However, appeal is mostly created when the character is designed and created.

Anticipation is another principle which is used to prepare the audience for an action that is about to be performed. An example would be a character bending its knees before jumping in the air. Another important animation principle is called ease in ease out, which refers to the acceleration and deceleration of a moving object or character. If the movement would be linear, it would become very unrealistic. An example could be a car that starts driving at full speed and stopping instantly. In order to make it look realistic, the car needs to gain speed when starting and decelerating when coming to a halt. An animator can create this principle by editing the animation curves mentioned earlier. To make it even more realistic animators should implement the follow through and overlapping action principle. Follow through relates to the idea that separate body parts won't stop at the same time. An example could be when a character stops walking, his arms would still be moving even though his feet already stopped moving. Overlapping is very similar but it refers to that separate body parts won't move at the same time. For example, if a character makes a waving motion, the shoulder first starts to move and the other parts of the arm would drag along after the shoulder. Most things in life move with an arc expect with machinery like a robot, and this principle should also be implemented when animating a character. An example of an arcing movement is when a character turns its head, it will dip down and move back up in an arcing motion.

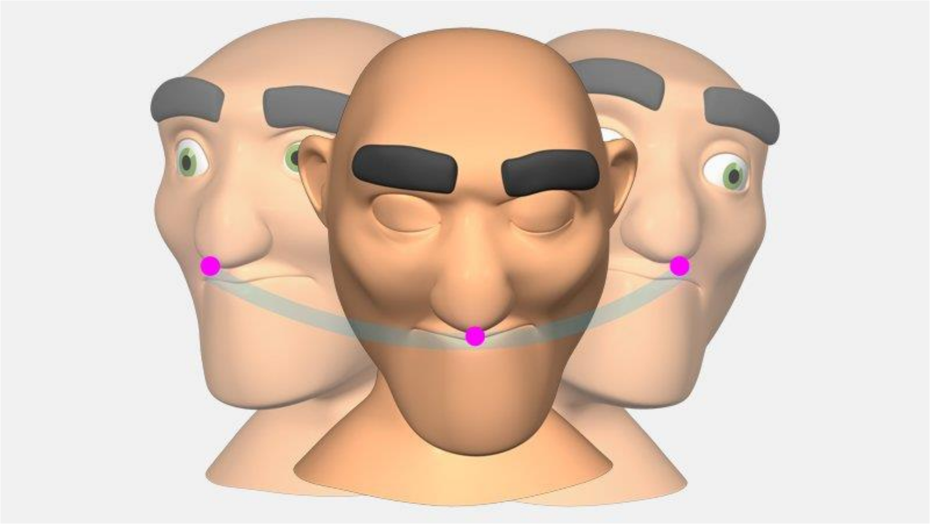


Figure 11- Example of the principle arc.

To bring more life to an animation, a second action should be added to the main action. Second actions should be subtle and emphasize the main action. An example could be a character walking down the street while whistling. The main action would be the character walking and the secondary action is the whistling, which emphasizes that this character is in a happy mood. Another principle of bringing more life to an animation is staging. Staging refers to setting up the scene, which can include the characters positions, the lighting within a scene, foreground and background elements, and camera angles. The idea of staging is to make the purpose of the animation as clear as possible to the audience. For example, if there is a fighting scene you could add shaking camera movements to add more action to the scene.

The principle straight ahead and pose to pose refers to the methods for animating. Pose to pose is when you keyframe one pose and then another pose and afterwards fill in the intervals. With straight ahead each frames is drawn out separately.

The last principle is called solid drawings and is a principle for traditional animation about anatomy, balance, three-dimensional shapes, light, and shadow, etc. A traditional animator would learn all these aspects from taking drawing classes and drawing from life. For 3D animators is mostly about posing your character in a way that’s not boring. A way to create interest in the pose of a character is to avoid mirroring its pose which makes it looks stiff and unappealing. For example, a character that has both hands in his or her hips makes it symmetrical and boring, but having one hand on the hip and putting more weight on one leg would make the pose more appealing. (Pluralsight, 2014e; Wikipedia, 2019)

3 Definition of the problem

The research done in the previous chapter briefly explains the general pipeline and workflow for creating game characters, however, some of these methods might be different for creating characters suited for low-end devices such as smartphones and tablets. Also since the game is not going to have a realistic style but a more cartoon-like style, the workflow, tools, and techniques might be different then previous explained. Therefore, further research will focus on how to create stylized game characters for low-end devices. Further research will also dive deeper into methods and techniques for creating the 3D models, topology, UV maps etc. with the style and platform in mind.

4 Main and sub-questions

4.1 Main question

How to create stylized 3D game characters for low-end devices such as a tablet, so that they are ready to be implemented into Unity?

4.2 Sub-questions

- What design principles need to be kept in mind when creating stylized characters?
- Which methods can be used for optimizing 3D game characters for low-end devices?
- What are the optimal tools and techniques for creating 3D characters for low-end devices?

5 Method

For answering each sub-question several methods will be used. This chapter will explain per sub-question which method is used, how the data obtained will be presented, and why the method is chosen.

5.1 What design principles need to be kept in mind when creating stylized characters?

To answer this question literature study will be used. Resources that are going to be used are websites and books. The data obtained with this method will be summarized and explained in short. This method is chosen because this gives a clear overview of the design principles.

5.2 Which methods can be used for optimizing 3D game characters for low-end devices?

For answering this question desk research will be applied. First information will be gathered on how to optimize 3D game characters, secondly, these theories will be tested through a series of technical experiments which will be evaluated afterward. This method is used because this will give clear data on what optimization techniques there are and which work for 3D characters and which do not.

5.3 What are the optimal tools and techniques for creating 3D game characters for low-end devices?

The method used to answer this question will be a combination of literature study and desk research. The tools and techniques will be investigated and tested out by following along several courses online in creating 3D characters. The most common tools and techniques will be explained and evaluated. The courses that have been followed are, 3D Character Workshop by

Shane Olson, How to retopologize a head like a boss and how to retopologize the rest of the body by Danny Mac, Getting started with Substance Painter 2018 from Substance Academy, and Getting started with ZBrushCore by Pablo Munoz. This method is chosen because practical research in combination with literature study should give clear answers to the question.

6 Scope

Since the topic of this research is quite broad, not every method and technique will be researched because this will simply be beyond the scope of this research. Also rigging and animations won't be researched within in full detail within in this research, since this will also be too much for the amount of time given for this research, and most of the animations are made by an external party. Also not every software will be discussed in full detail because there are too many programs that can be used when creating 3D models and each software has a lot of tools.

7 Result

7.1 What design principles need to be kept in mind when creating a stylized character?

Even though a character is mostly designed during the concept art phase, it’s still important for a character modeler to know about the design principles for character design, because a modeler can enhance these principles and help bring the character to life. However, before going into details what these principles are, there is still the question of what stylized means. While stylization is quite a vague and arbitrary term, it can still be categorized into different categories.

7.1.1 Stylization categories

The first category would be realistic. This form of stylization is as close as realistic as possible. The proportion of the characters are very close to realism and so are the textures. Even though characters with this type of stylization are as close to realism, they still are very well designed. As the graphical power of gaming consoles improves over the years, what was considered realistic a few years ago isn’t considered realistic today, and what is considered realistic today probably won’t be in the future.

The second category would be stylized realism or enhanced realism. While characters with this type of stylization still have proportions close to realism, their shapes and silhouettes are simplified and exaggerated. Textures are also smoother and less realistic with simplified and enhanced colors but are still very detailed. Great examples are the characters from the game “Overwatch”.

The third category is exaggerated stylization. With this category shapes and silhouettes are exaggerated so much they don't look realistic anymore. With this style, the artist can decide what details need to be enhanced and what needs to be removed, depending what the model is supposed to resemble. Proportions of the model also don't come anywhere near realism, but textures sometimes can still be very detailed and close to realism.

The fourth and final category is abstract or minimalistic stylization. With this category shapes and silhouettes are simplified so much they almost only consist of primitive shapes. Textures are also very flat and have little to no detail and mostly consist of colors only. To create these different kinds of stylizations an artist can use different design principles, which will be discussed next. (Aava, 2017; Anhut, 2016; Olson, n.d)

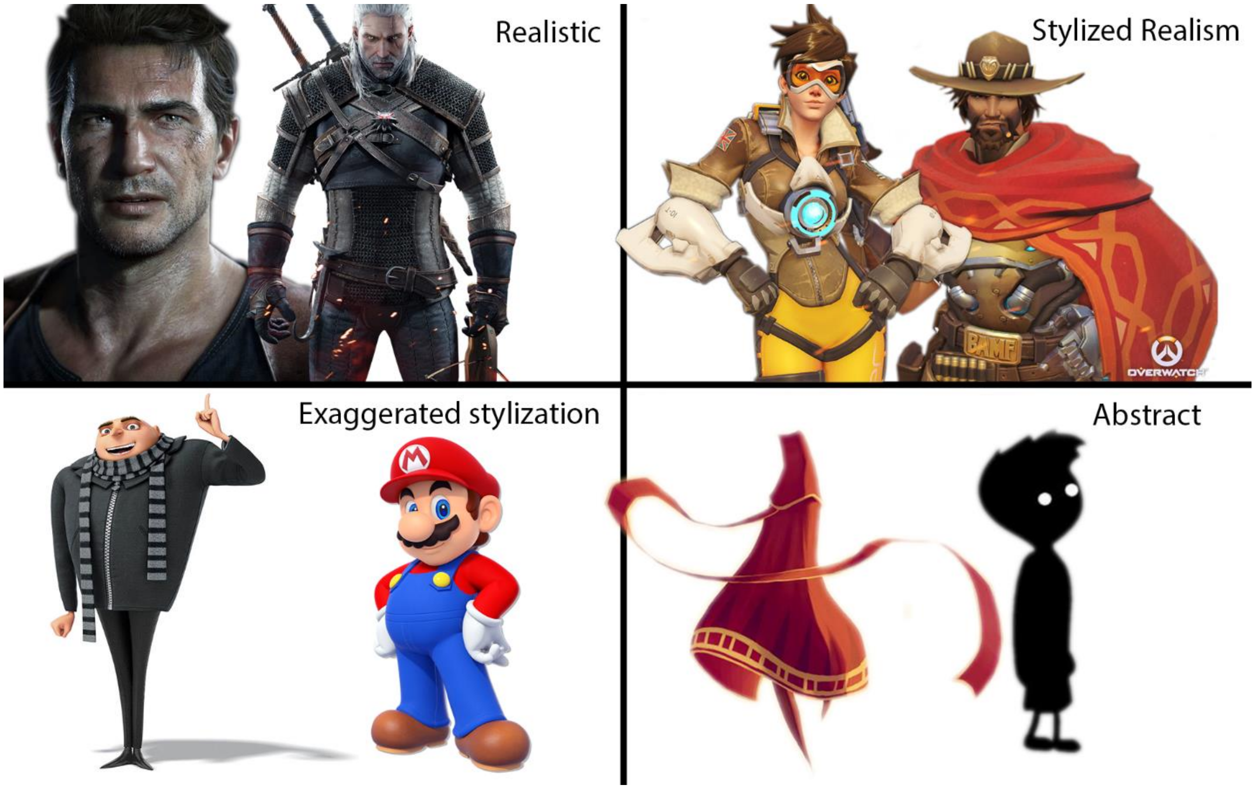


Figure 12- Examples of different style categories.

7.1.2 Shape Language

Through the use of shapes, you can communicate a lot of a character's personality. This is best done by repeating simple and primitive shapes like spheres, cubes, and triangles. For example, a character that is cute or friendly is usually depicted with round shapes like spheres. A character that has a lot of cubes or blocks can depict a stubborn or strong character. Using a lot of triangles can make a character look devious or cunning and evil. By combining shapes you can communicate several aspects of a character’s personality.

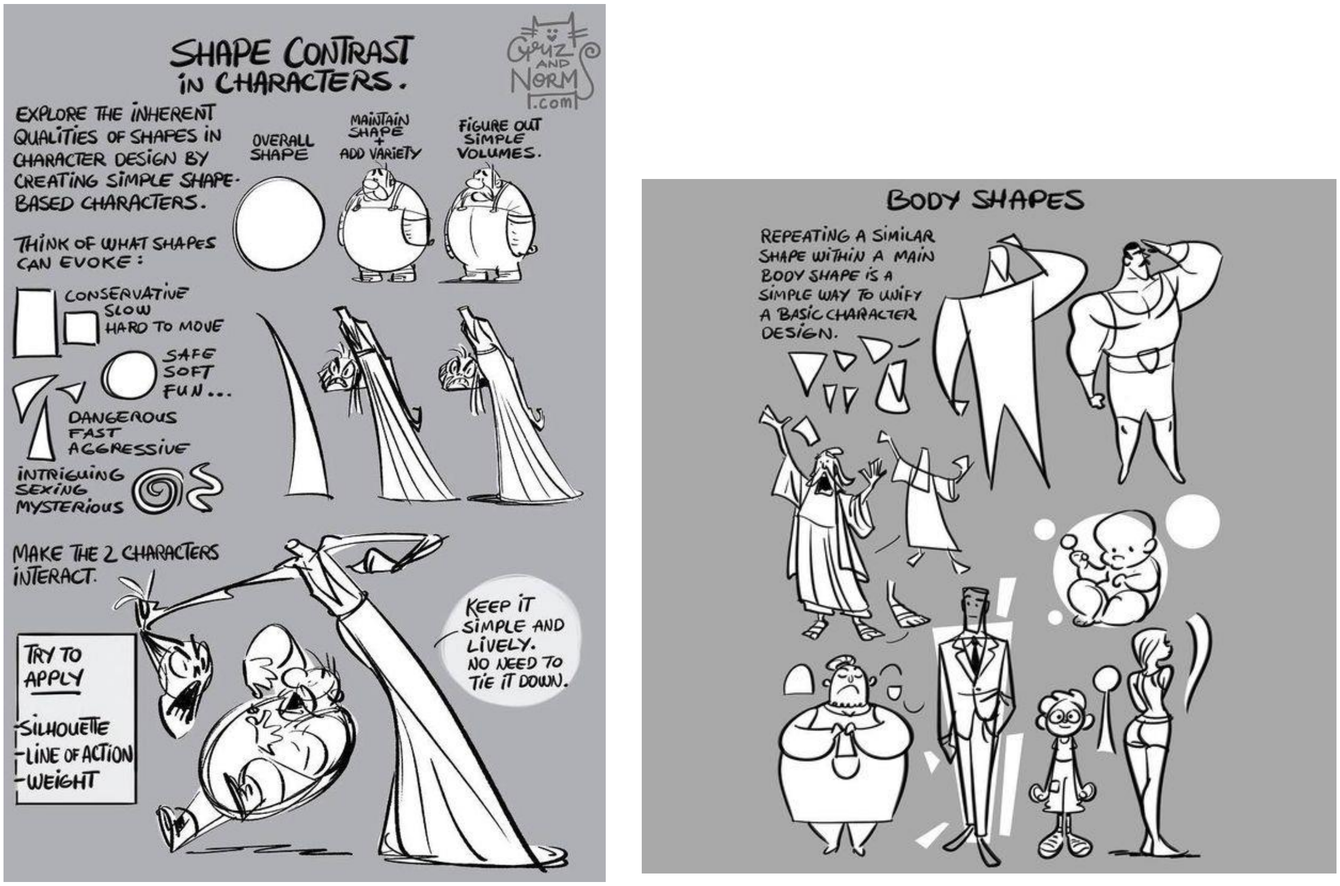
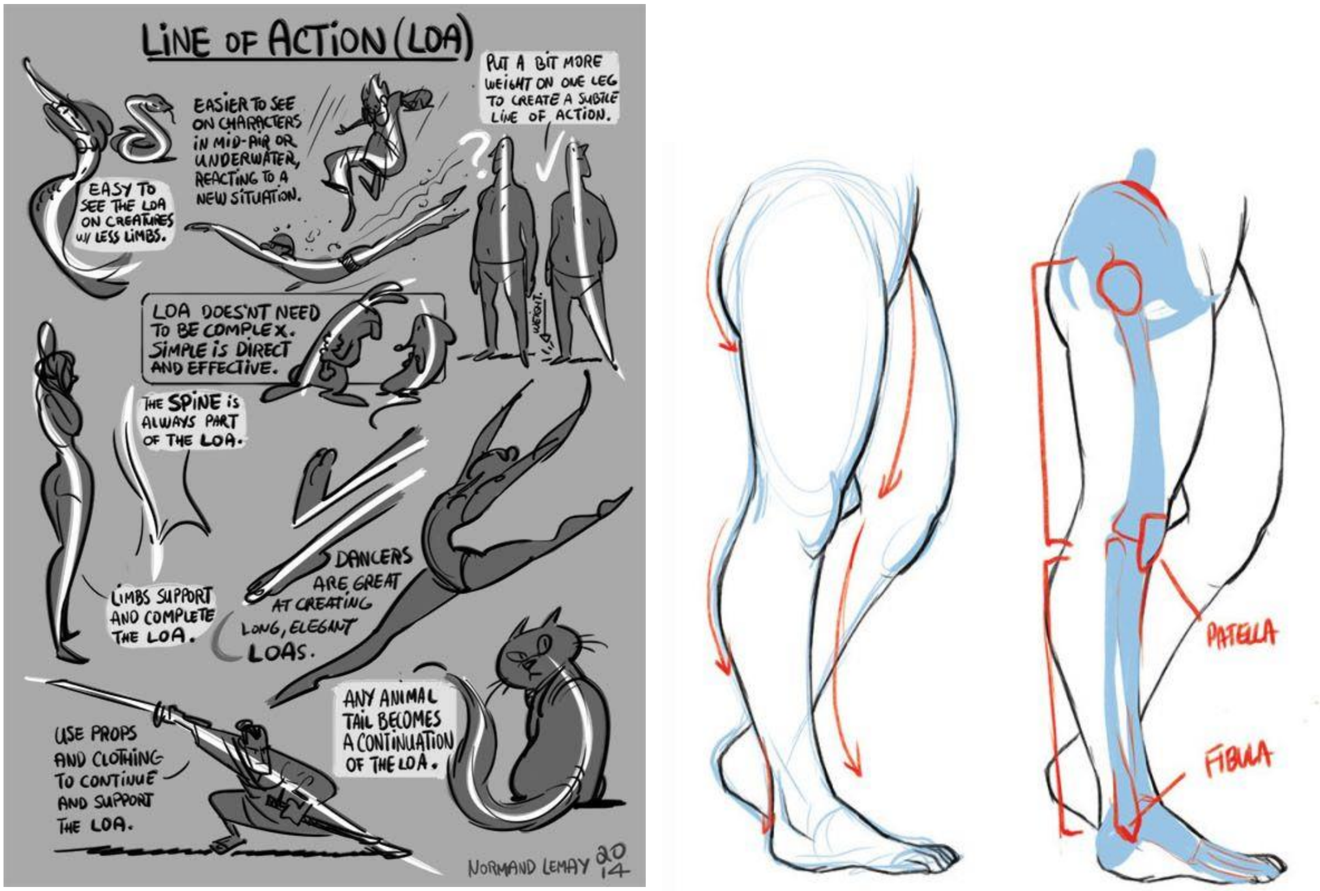

SHAPE CONTRAST IN CHARACTERS.
GRIZ AND NORM .com
EXPLORE THE INHERENT QUALITIES OF SHAPES IN CHARACTER DESIGN BY CREATING SIMPLE SHAPE-BASED CHARACTERS.
OVERALL SHAPE
MAINTAIN SHAPE + ADD VARIETY
FIGURE OUT SIMPLE VOLUMES.
THINK OF WHAT SHAPES CAN EVOKE:
CONSERVATIVE SLOW HARD TO MOVE
SAFE SOFT FUN...
DANGEROUS FAST AGGRESSIVE
INTRIGUING SEXING MYSTERIOUS
MAKE THE 2 CHARACTERS INTERACT.
KEEP IT SIMPLE AND LIVELY. NO NEED TO TIE IT DOWN.
TRY TO APPLY
-SILHOUETTE
-LINE OF ACTION
-WEIGHT
BODY SHAPES
REPEATING A SIMILAR SHAPE WITHIN A MAIN BODY SHAPE IS A SIMPLE WAY TO UNIFY A BASIC CHARACTER DESIGN.

Figure 13 - Examples of shape language used on characters.

7.1.3 Rhythm and flow

A character can look stiff and boring without rhythm and flow. This principle is mostly used in posed characters rather than a character that is in a neutral pose, but rhythm and flow can also be found within the body's anatomy. Muscles have a rhythm to them and flow into each other. A great way to depict rhythm and flow is the use of the line of action. This line usually goes through the whole body of a character from head to toe, and represent a character's main gesture. For a character modeler, it's important to recognize rhythm and flow to create appealing characters.



LINE OF ACTION (LOA)
EASY TO SEE THE LOA ON CREATURES W/ LESS LIMBS.
EASIER TO SEE ON CHARACTERS IN MID-AIR OR UNDERWATER, REACTING TO A NEW SITUATION.
PUT A BIT MORE WEIGHT ON ONE LEG TO CREATE A SUBTLE LINE OF ACTION.
WEIGHT
LOA DOESN'T NEED TO BE COMPLEX. SIMPLE IS DIRECT AND EFFECTIVE.
THE SPINE IS ALWAYS PART OF THE LOA.
LIMBS SUPPORT AND COMPLETE THE LOA.
DANCERS ARE GREAT AT CREATING LONG, ELEGANT LOAS.
USE PROPS AND CLOTHING TO CONTINUE AND SUPPORT THE LOA.
ANY ANIMAL TAIL BECOMES A CONTINUATION OF THE LOA.
NORMAND LEMAY 2014
PATELLA
FIBULA

Figure 14 - Examples of the line of action and the flow within a leg.

7.1.5 Silhouette

If every detail, lighting, and color is removed from a character you are left with a silhouette. A strong silhouette is important for a character because this is what makes a character easy to recognize. Almost every movie or game character is still recognizable if you only see their silhouette. With 3D characters, this might be more difficult to achieve from all angles but a modeler still needs to be mindful of it. A good silhouette is also important when animating so it’s clear to the viewer what the character is doing.

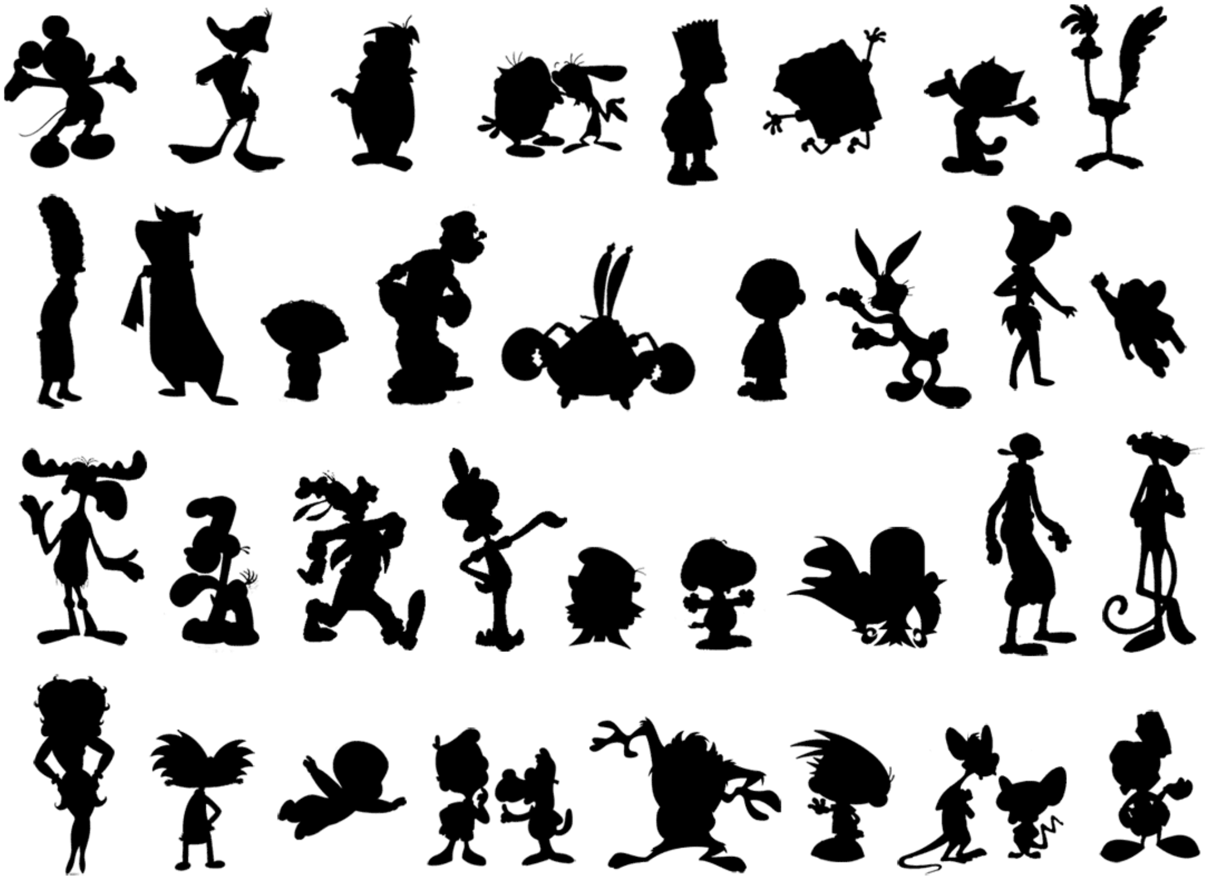


Figure 15 - Examples of silhouettes from famous characters.

7.1.6 Straights against curves

Another way of adding interest to a character is to use straights against curves. By using opposing surface shapes you create an asymmetry to a character or object, which makes it look more appealing. With a character the straight lines are usually at the bony sides of the body part and the curves at the fleshy sides. When talking of straight lines it doesn't mean they are perfectly straight, they still have a little curve to them to look more interesting.

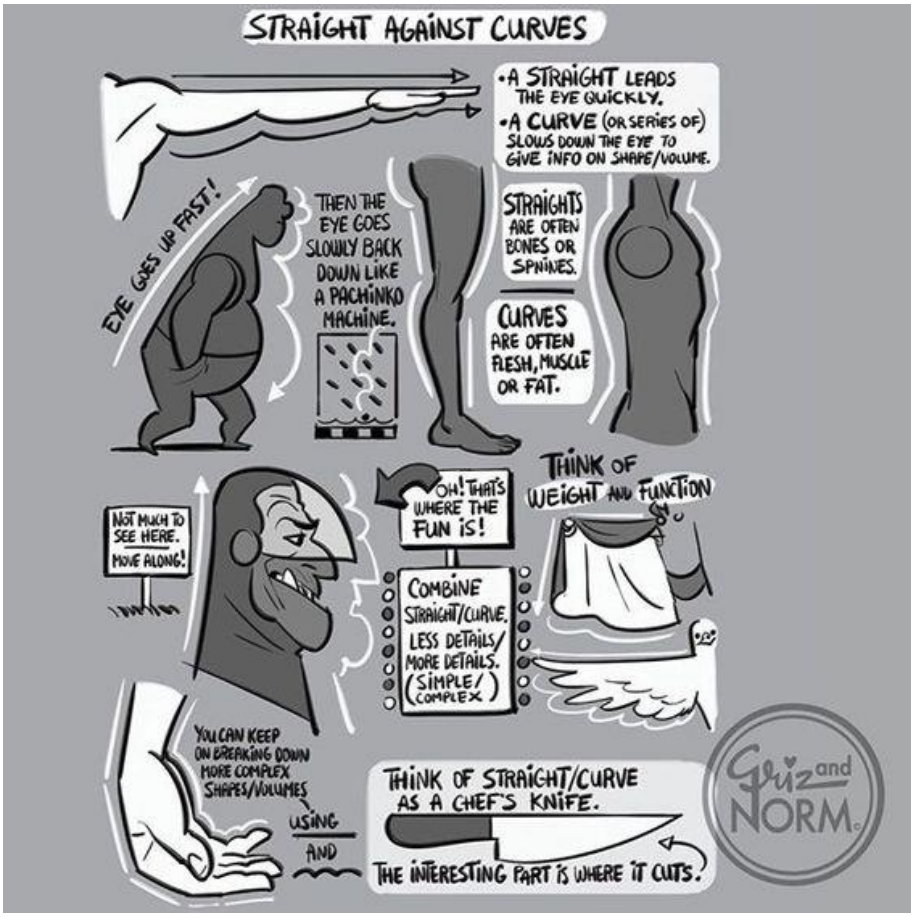

STRAIGHT AGAINST CURVES
•A STRAIGHT LEADS THE EYE QUICKLY.
•A CURVE (OR SERIES OF) SLOWS DOWN THE EYE TO GIVE INFO ON SHAPE/VOLUME.
EYE GOES UP FAST!
THEN THE EYE GOES SLOWLY BACK DOWN LIKE A PACHINKO MACHINE.
STRAIGHTS ARE OFTEN BONES OR SPNINES.
CURVES ARE OFTEN FLESH, MUSCLE OR FAT.
THINK OF WEIGHT AND FUNCTION
NOT MUCH TO SEE HERE. MOVE ALONG!
OH! THAT'S WHERE THE FUN IS!
COMBINE STRAIGHT/CURVE. LESS DETAILS/MORE DETAILS. (SIMPLE/COMPLEX)
YOU CAN KEEP ON BREAKING DOWN MORE COMPLEX SHAPES/VOLUMES USING — AND ~
THINK OF STRAIGHT/CURVE AS A CHEF'S KNIFE.
THE INTERESTING PART IS WHERE IT CUTS.
Griz and NORM

Figure 16 - Examples of using straight against curves.

7.1.7 Variety and interest

Making the character of an object different creates variety. A good design is achieved by having a good balance between variety and unity. Elements should be different enough to look interesting, while also be alike enough to be perceived as one. When designing characters it’s good to avoid repetitive shapes, patterns or parallel lines. This is another way to create asymmetry since nothing organic is perfectly symmetrical. Great ways to create variety are changing the size and color of an object, using s-curves instead of parallel lines, using different surface thickness, and having different stroke weights. Another great way to create variety and interest is to use areas of detail and areas of less detail.

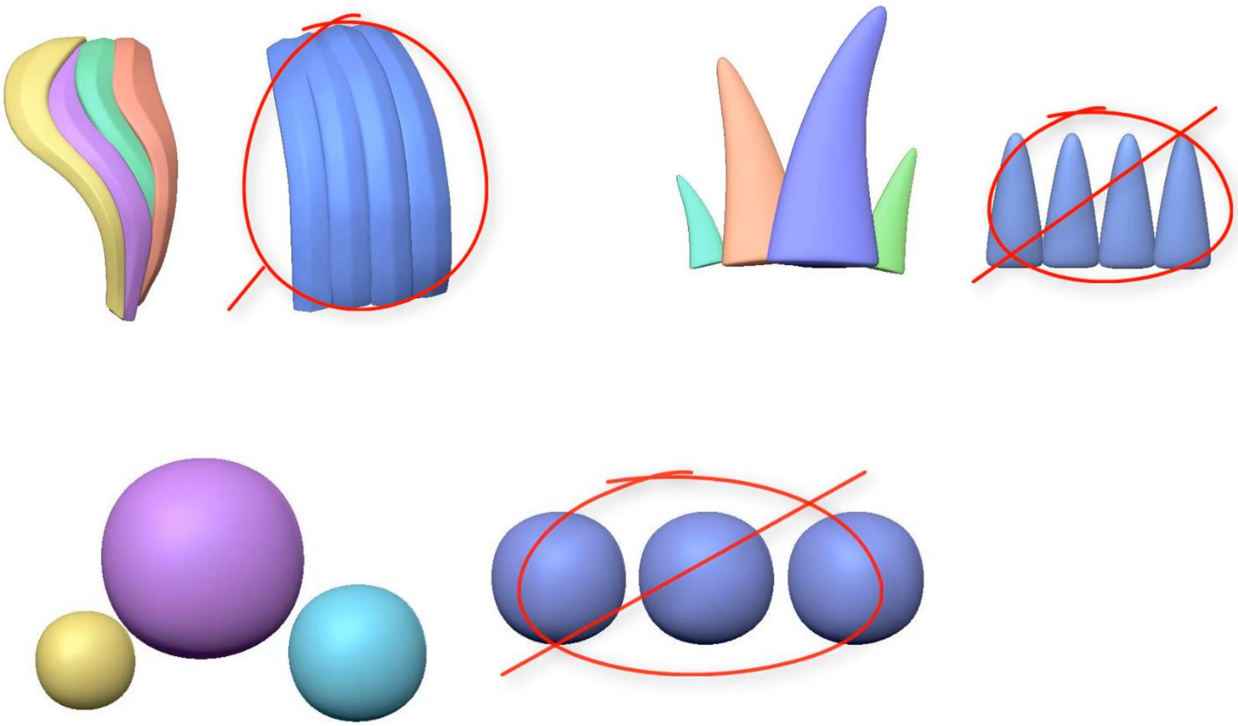


Figure 17 - Examples on how to use variety.

7.1.8 Simplification and exaggeration

As mentioned earlier most art styles are created by simplifying and exaggerating. By taking something complex and making less of it you simplify it. It’s easier to simplify real-life characters rather than those that are already been designed since there is more to simplify. Exaggeration can be achieved by making large things even larger, and small things even smaller. Great examples of exaggeration can be found in caricatures. Another way to create exaggeration is by pushing a pose to its limits.

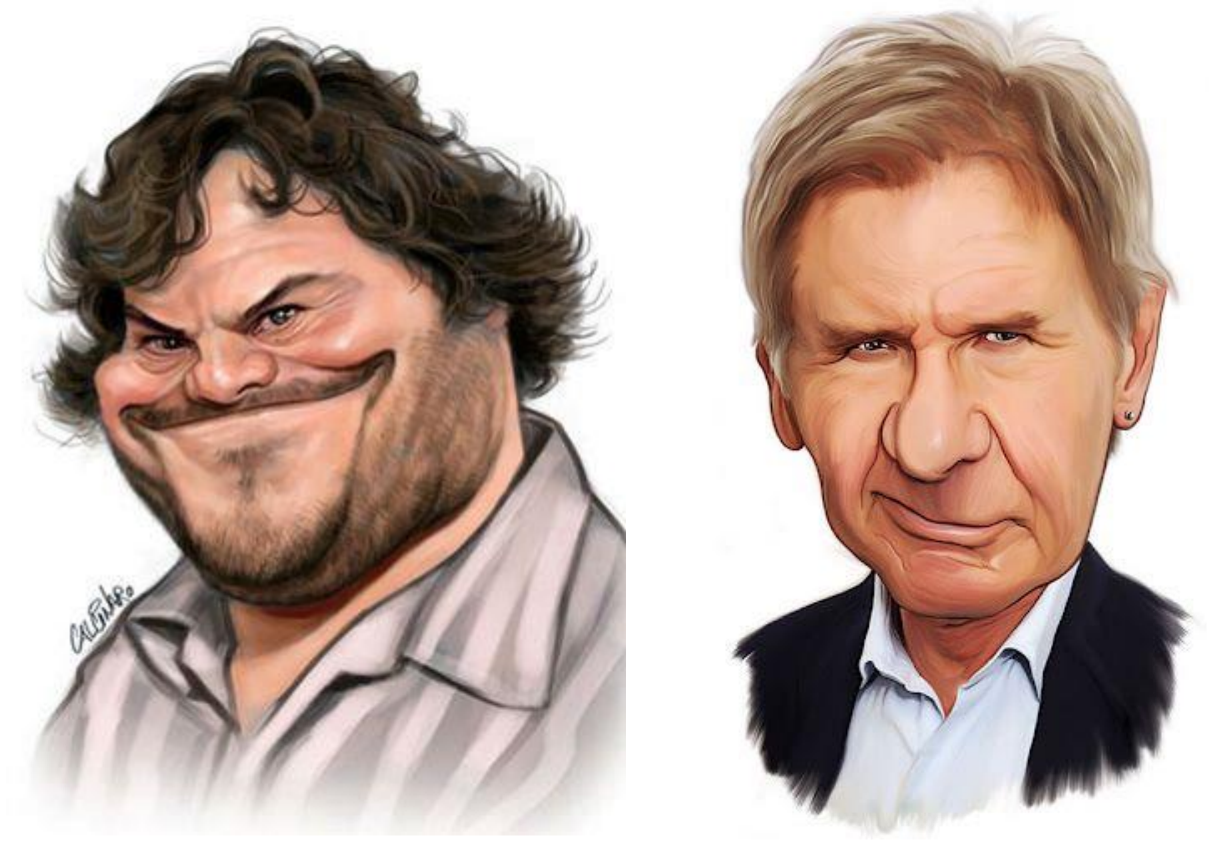


Figure 18 - Examples of caricatures.

7.1.9 Proportions

With character, proportions refer to the relative size of the body parts and are usually measured in heads. Ideally, a character is seven and a half heads tall but sometimes eight heads tall with superheroes. The fewer heads you use for a character's height, the younger they will look. It's important to know what real proportions look like in order to exaggerate and stylize them. A common trick used is to make the female character one head smaller than the male characters. (Olson, n.d.; 3dtotal Publishing, 2018)

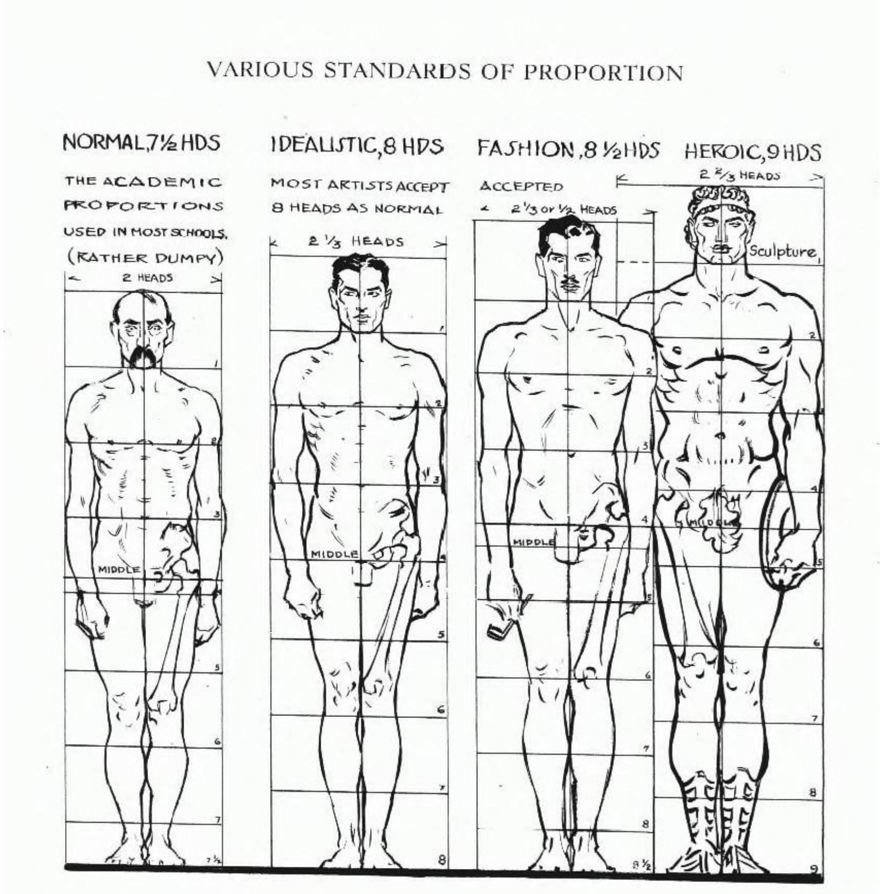


Figure 19 - Example of human proportions.

7.2 Which methods can be used for optimizing 3D game characters for low-end devices?

Because low-end devices are limited in performance, there are more technical restrictions when creating 3D assets for them. This chapter explains which aspects need to be optimized for 3D characters intended for low-end devices.

7.2.1 Polycount

The biggest performance bottleneck in games for low-end devices is the number of polygons the device can render in real-time. It is advised to use about 300 to 1500 polygons per mesh for mobile devices, but in reality, it depends on the game. For instance, if there are a lot of characters on screen at the same time, it would be a good idea to keep the polycount somewhere between those numbers but if there is going to be only one character on screen at a time, the number of polygons could well exceed 1500. (Gosch, 2017; Unity Technologies, 2017)

Thus when creating 3D characters for low-end devices the polygons should be low and this is done either in the modeling stage or the retopology stage, but there are a few things to keep in mind when creating low-poly characters. One of which is to only add polygons that add to the silhouette. Small details that don’t add to the silhouette can be left out and added into the different texture maps. Another thing to keep in mind is having good topology for deformation. As mentioned earlier more polygons are needed around the joints in order to deform well and that triangles should be avoided when a model is going to deform. However, since a model will be triangulated when being exported out of the 3D application and imported into the game engine, it actually can be beneficial to use triangles instead of quads at joints like the fingers, knees, and elbows. This method is used to save a few polygons with better deformation. (Polycount Wiki, 2015b; Silverman, 2013)

7.2.2 Textures

The size of the textures and the resolution of the textures also have a big impact on the performance of the game. A bigger size and resolution means that the textures take up more memory space. While modern game engines can support texture sizes up to 8129 pixels in height and width, modern mobile devices can only support textures up to 4096 pixels, and older devices are even limited to smaller textures sizes. Since the memory on mobile devices is limited the size and amount of textures are also limited. In addition to less memory, the use of normal maps and alpha maps can also have a negative impact on the performance of the game on older mobile devices. However, there are a few techniques that can be used to optimize the performance, while keeping an acceptable visual quality. (Unity Technologies, 2013)

A common way to save memory space for textures is to use tiling textures. Tiling textures are textures that can be repeated in any direction, which can be used to cover large areas such as walls. Since the detail of the texture can be repeated when using tiling textures, the texture size can be smaller and can be used for multiple 3D models, meaning they take up less memory space.

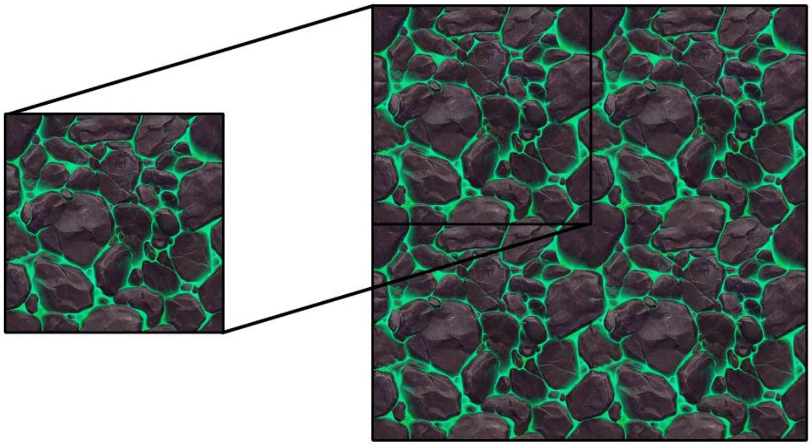


Figure 20 - Example of tiling textures.

Unfortunately, most of the time 3D characters are unable to make use of tiling textures since they are unique, however, they can make use of a texture atlas. A texture atlas basically is a texture containing multiple textures in one. To apply a texture to a 3D mesh within Unity, the textures need to be added to materials, which then is applied to the 3D mesh. For mobile devices, it’s advisable to use as few materials as possible. By combining multiple textures within a texture atlas, one material can be used for multiple 3D meshes. For characters, it might be desired to use more than one material so that different shaders can be used for different parts, but usually, no more than two or three materials per characters are needed. A downside of using a texture atlas, however, is that even though multiple textures are combined into one, which does save memory, the resolution of each texture will be lesser since they are divided by the number of textures used in the atlas. (Mäkelä, 2011; Silverman, 2013)

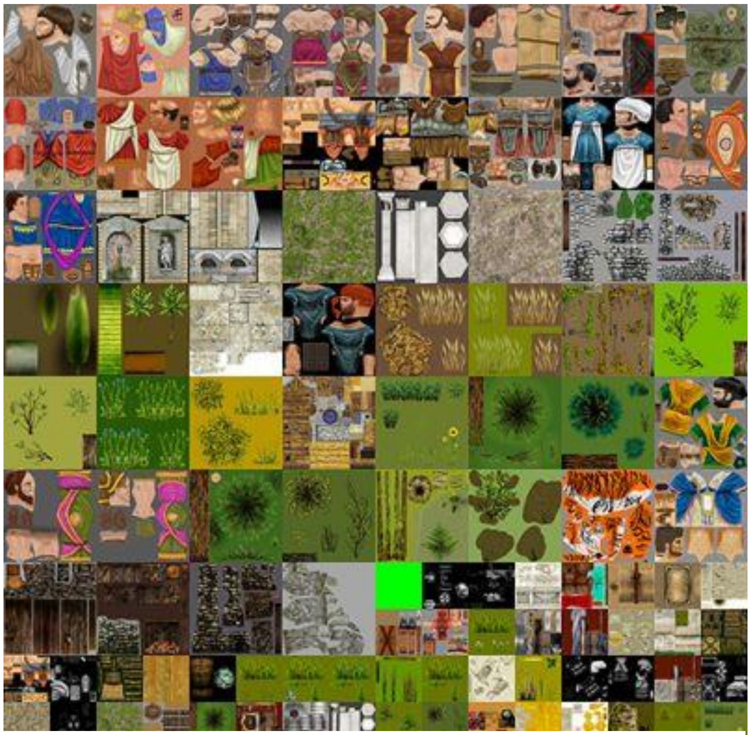


Figure 21- Example of a texture atlas.

Another common technique to optimize textures is to make use of mirrored UV shells. If parts of a 3D mesh are symmetrical, the UV shells for those parts can be flipped and stacked onto each other. This creates more space on the UV map since each part is essentially cut in half, and the free space can be used to place other parts or to enlarge the UV shells to allow more detail. Since characters are symmetrical most of the time, this technique is often used when creating UV maps for characters. (Mäkelä, 2011)

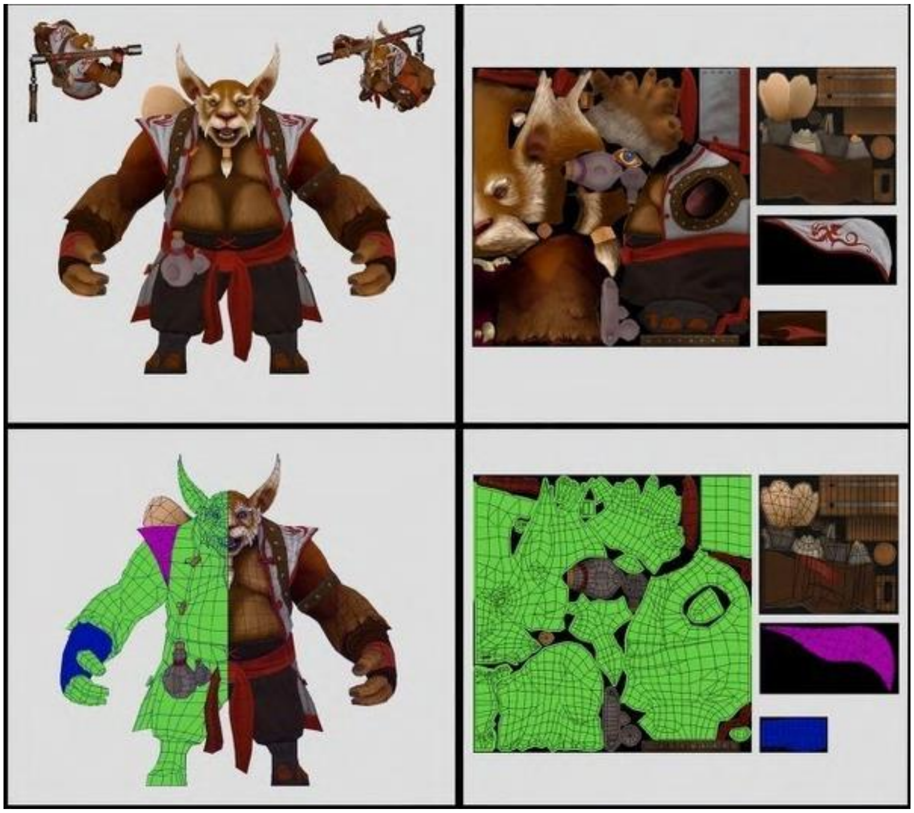


Figure 22- Example of mirrored UVs.

What also can be done is to only use a color map like an albedo or diffuse map in order to save memory. This way the material would be less complex and improve the performance. For this to still look good, the lighting information that comes from the normal map, specular map, and ambient occlusion map should be either hand painted into the color map or baked in. However, this is usually only done when an object is not moving and the lighting in the scene doesn’t change since the light information on the color map won’t change with it if the lighting does change. This probably will not be the best solution for characters, since they usually will move around a lot but it can be used if performance is a big issue. (Blender, 2017; Valve, n.d.; Polycount, 2013a; Polycount, 2013b)

It’s even possible to not use any textures at all and still have color information within a 3D mesh. Polygons can contain color information by coloring the individual vertices, this is called vertex colors. However, if two vertices next to each other have different colors, the two colors would evenly blend into each other. Even though vertex colors use less memory then textures, in order to have an area with two colors the area needs to be split in half, creating a higher polycount which could impact performance again. Another downside is that little detail can be added with vertex colors on low-poly models since this depends on the density of the model. (Alkemi, 2013)

A better solution to only use colors as textures is to make use of a palette texture. A palette texture is a texture with areas of different colors that can be used by multiple meshes. Because palette textures only have areas of individual colors, UV shells can be stacked on top of each other without having to worry about repeating patterns or texture distortion, which makes the process of creating a UV map also much simpler. However, using a palette texture with a lot of colors can get complicated. (Chung, 2014; Nadezhdin, 2018)

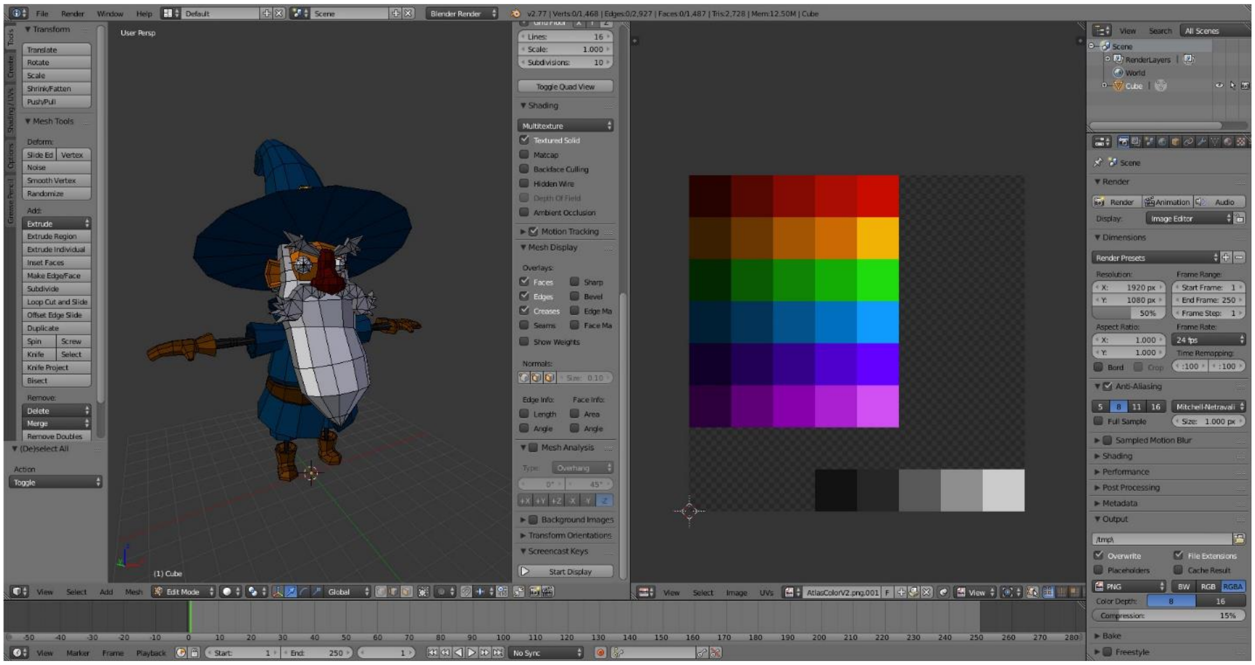


Figure 23- Example of a color palette texture.

7.2.3 Rigging and animations

For rigging and animations, there are only a few things that can be done to optimize performance. The first thing is to use as few bones as possible for a character rig. For high-end games, a rig can contain about fifteen to sixty bones, but for games intended for low-end devices, the bones should stay below thirty. Before binding a rig to the character make sure that the character is one mesh, this will limit the draw calls. In addition, it’s also advisable to keep forward and inverse kinematics separate, because when an animation is exported the IK nodes are baked into forward kinematics and Unity has no need for them. By keeping the IK nodes separate it’s easier to remove them later. Also, try to not use too many keyframes cause this will keep the animation file small, and the animations should be baked when exporting them. (Gosch, 2017; Unity Technologies, 2017)

7.2.4 Experiments

To test the theory that triangles can be used for good deformation, a little experiment is conducted within Maya, where three arms with different topology at the elbow joint are rigged and bend to check the deformation.

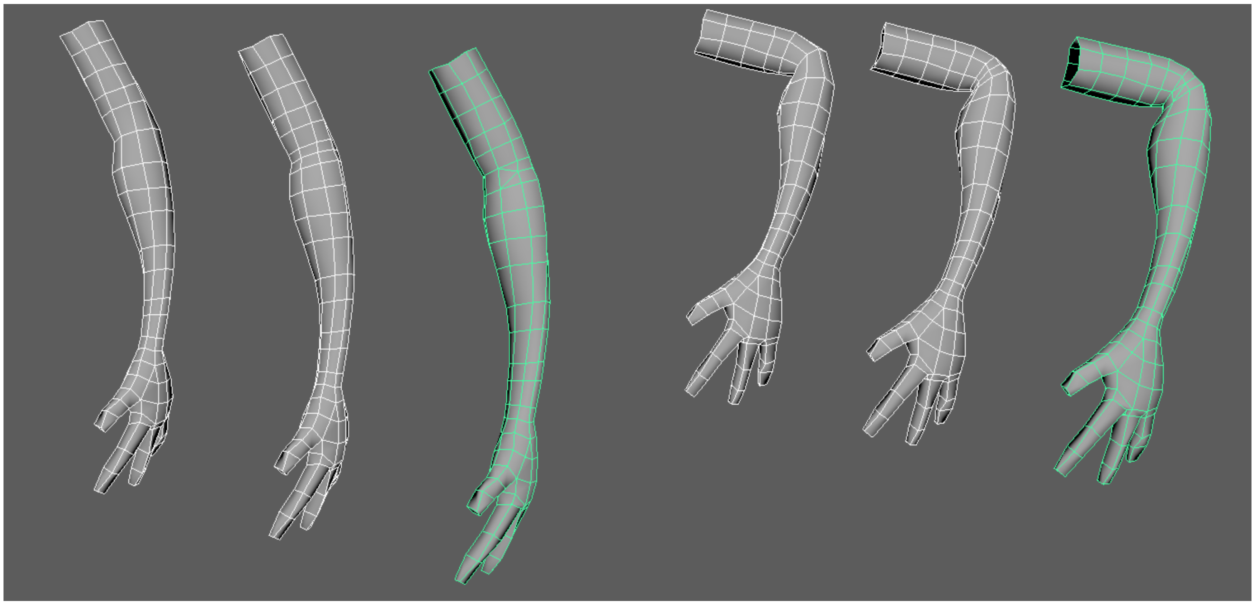


Figure 24-From left to right. The first arm with only one edge loop at the elbow causes bad deformation. The Second arm with two extra edge loops at the elbow with good deformation. The third arm is with triangles at the elbow saving a few polygons with better deformation.

The result confirms that using triangles at certain points of the joints can actually be beneficial for deformation while keeping the polycount low.

To test the theory about the performance when using a high poly mesh, multiple materials and high-quality textures on low-end devices explained in the previous chapter, an experiment has been conducted. Multiple models are tested on a Samsung Galaxy S8 smartphones and the performance is measured by measuring the average frames per second, the stability of the frames per second, the average GPU usage, and the maximum GPU usage.

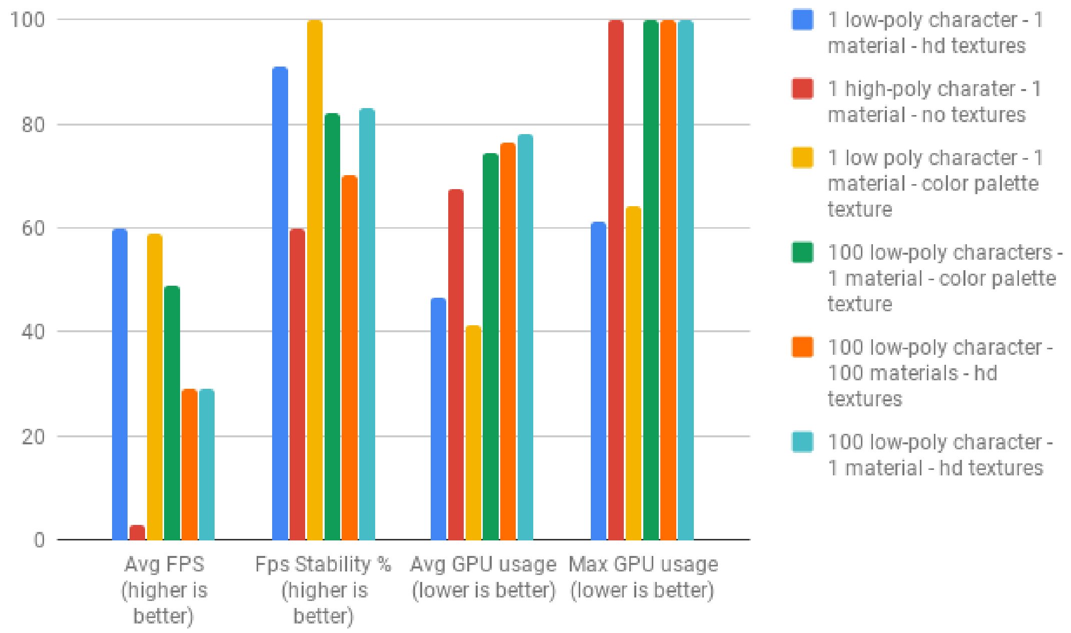


Figure 25 - Graph of performance test.

This result shows that the number of polygons within a model is one of the biggest bottlenecks when it comes to performance, this is indicated by the red chart. When it comes to the number of materials being used there is hardly any difference measured between one material with an HD texture or a hundred materials with HD textures. However, using one material with a low-resolution texture atlas on multiple models does seem to optimize performance, this indicated that using high-resolution textures has more impact on performance then the amount of materials.

7.3 What are the optimal tools and techniques for creating 3D characters?

This chapter will go through the process of creating a 3D character with the optimal tools and techniques. The process is divided into four stages, sculpting, retopologizing, UV mapping, and baking and texturing. The workflow used is high-poly to low-poly which might not be the common method used when creating 3D characters for low-end devices, but this workflow is the current method.

7.3.1 Sculpting

When creating the actual mesh there are a lot of tools, methods, and techniques. The software tested is ZBrush since this is one of the most used sculpting programs in the gaming industry and has a lot of tools. This chapter is divided into three sections, first there is a quick overview of the most common tools used within ZBrush, then several methods for creating the base mesh will be explained and compared, and finally, the tools and methods for the detailing phase will be explained.

Tools

Boolean and Live Boolean

Boolean is a method for creating 3D meshes by either combining or subtracting multiple Subtools. Live Boolean is a special mode for Zbrush where the result of boolean actions will be shown as a preview and are intractable before creating the new boolean mesh.

Brush

The most used tool when sculpting a 3D mesh. With brushes, you can add or subtract geometry to a mesh. There are a lot of different brushes with different effects and brushes can also be created and customized.

Creasing

When a mesh is subdivided the edges of the mesh will be smoothed. With creasing, you can add extra geometry to the edges, which preserves sharp and hard edges.

Dynamesh

When sculpting, polygons can get stretched and distorted if they are deformed a lot. When applying Dynamesh, Zbrush will rebuild the geometry for the selected Subtool and fix these issues.

Dynamesh Master

Dynamesh Master is a plugin that can be added to Zbrush. With this plugin, you have more control when using Dynamesh. When using Dynamesh the resolution slider controls how dense the Dynameshed mesh will be, but a resolution is an arbitrary number. With Dynamesh Master you can give an exact amount of polygons wanted for the new Dynameshed mesh.

Dynamic Subdiv

When using Dynamic Subdiv, Zbrush will divide the selected Subtool using the parameters set in the Dynamic Subdiv menu. The difference with Dynamic Subdiv as supposed to regular subdividing is that it's a preview until the apply button is pressed. Another reason for using Dynamic Subdiv instead of regular subdividing is that certain actions and tools will not work when there are multiple subdivisions.

Polygroups

Polygroups are the way Zbrush organizes polygons an object within a Subtool and are represented with different colors. Polygroups make it easy to edit different parts of a Subtool.

Primitives

Primitives are basic shapes like cubes, spheres, and cylinders which can be used for a base mesh. Within ZBrush, there are several ways to create primitives, like using an Insert Multi Mesh brush or the initialize menu.

Project

When using the Project function, Zbrush will project all sculptural detail from one Subtool onto another similar Subtool. This method is great for reapplying details on a lower resolution mesh.

Sculptris Pro

With Sculptris Pro mode activated, all brushes that are compatible will decimate each brush stroke and add triangle tessellation to each brush stroke. With this mode, there is no need to use Dynamesh or Subdivide if you want to add detail to a model because it will be possible to add more polygons at specific places that need detail.

Spotlight

Spotlight is a feature used to transfer images onto the surface of a model, but it can also be used as a way to display reference images within Zbrush.

Subdivide

When subdividing a model, Zbrush will divide each polygon into four, two horizontal and two vertical, thus quadrupling the number of polygons. This allows adding more detail to a mesh.

Subtools

Subtools are separate polygons objects. Each sub tool can have as many polygons your computer can handle. Because Subtools are separate you can only sculpt on one Subtool at a time.

Gizmo / Transpose Tool

The Gizmo is a tool that is used in most 3D software. It is used to move, rotate and scale meshes. In addition to the Gizmo, ZBrush also has the Transpose Tool which is a unique tool that can do all the things the Gizmo can do and more but works a little different.

ZModeler

The Zmodeler Brush is a special brush that has its own unique features. With the Zmodeler Brush, you can perform specific actions on vertices, edges, and polygons. This tool is the equivalent for box modeling techniques within other 3D software and is a great tool for creating hard surface models.

ZSpheres

With ZSpheres you can sketch out a model using spheres that are connected to each other. It’s a unique tool within Zbrush for creating base meshes, particularly useful for creating organic models. ZSpheres cannot be sculpted on like polygon meshes because they are unique objects but once the desired basic shape is created with them, they can be turned into a single polygon mesh.

ZRemesher

The ZRemesher tool is an automatic retopologizing tool within ZBrush. It will create a new polygonal structure for your model, with a controlled edge flow and global polycount value.

Methods and Techniques

A common method for creating a 3D character is to go from big to small, first creating the big shapes and blocking out a character and then create small details. The character block out is also known as a base mesh. There are a few techniques for creating the base mesh but the two most used are ZSpheres and the Insert Primitive method also known as the Insert Multi Mesh method since it uses IMM brushes. Both methods are tested and compared to each other.[3]

Method	**Pros**	**Cons**
ZSpheres	Fast	Less control to create clean shapes
	Simple to use	Can have unwanted results [2]
	No additional tools needed	Don’t see the result right away (have to press preview in order to see the polygon mesh that will be created with the ZSpheres)
		Takes some time getting used to
IMM	Nice clean shapes	Little slower compared to ZSpheres
	Simple to use	Needs additional tools
	Creates nice transitions between body parts	
	More control to create clean shapes	
	Easy to edit	

When blocking out the character with the IMM method there are several tools and methods used to create the base mesh.

Tools	**Method and Technique**
IMM Primitive Brush	The IMM brushes are used to insert the different body parts.
Move Brush	The Move Brush is used to edit large areas of a primitive shape.
Transpose Tool / Gizmo	The Transpose Tool or Gizmo are used for scaling, moving and rotating the primitive shapes.
Subtool	During this stage it is useful to only use one Subtool because ones the block out is finished it will be Dynameshed into one single base mesh for the detail phase.
Polygroup	Since only one Subtool is used it is useful to make use of polygroups to edit one primitive shape at a time.
Dynamic Subdiv	Dynamic Subdiv is used to add more detail while still being able to use other tools.

When the character base mesh is finished the next phase is to add detail and clothing. Creating the details is the most time-consuming part during the sculpting phase. Because ZBrush is amazing at creating really good details it is a common mistake to start with detailing too soon. The best stylized characters are good because of their simplicity and that is why the block out phase is so important. Most of the detailing is done by using different kinds of brushes but some other tools are also used during this phase. It is also important to keep all the design principles in mind when creating the details and this is where the skill and trained eye of the artist really comes into play. The next table shows which tools are used during this phase and what they are used for.

Tool	**Method**
Dynamesh	Dynamesh is used to combine all the primitive shapes into one high-resolution mesh.
Smooth Brush	The Smooth Brush is used to smooth out the surface of the mesh.
Fill Brush	The Fill Brush is used to add more geometry to body parts that need more volume.
Pinch Brush	The Pinch Brush is used to add more creases in certain parts of the mesh
Move Brush	The Move Brush is used to edit large shapes of the mesh.
ZRemesher	ZRemesher is used to reconstruct the topology of a Dynameshed mesh, this can help with keeping the model clean and easy to edit.
Project	Sometimes when ZRemesher is used some small details are lost. By first copying the old mesh and use ZRemesher you can project the old mesh onto the new ZRemeshed version to get those details back.

There are two common techniques for creating clothing and accessories. The first one is to duplicate the base mesh, then mask the out the part that will become the piece of clothing, and delete the rest. Now the piece that is left over can be sculpted further into a piece of clothing. The other method is to use a combination of the Topology brush and the ZModeler brush. With the Topology brush, you can draw out polygons onto the surface of the base mesh. Every square drawn will become a polygon when you are finished. The brush size depends on how thick the polygons will be, but for more control, it is better to have a brush size of one which creates single sided polygons. Afterward, you can add thickness with the ZModeler brush by extruding the polygons.

7.3.2 Retopology

As mentioned earlier there the topology of a 3D character is really important in order to deform well when it is going to be animated. Some 3D software has automatic or semi-automatic tools for retopologizing, but these are usually not suitable for characters that need to be animated. However, these tools are still tested and the result is shown in the first part of this chapter. The second part of this chapter explains the tools and methods for creating the topology manually.

Automatic retopology

For testing out the automatic or semi-automatic retopology tools, two programs are used. The first software tested out is ZBrush and the second one is 3D Coat. As explained in the previous chapter ZBrush has a way of retopology with the tool ZRemesher, but it can also be used to semi-automate the process for retopologizing the high-poly model with just a few extra steps.

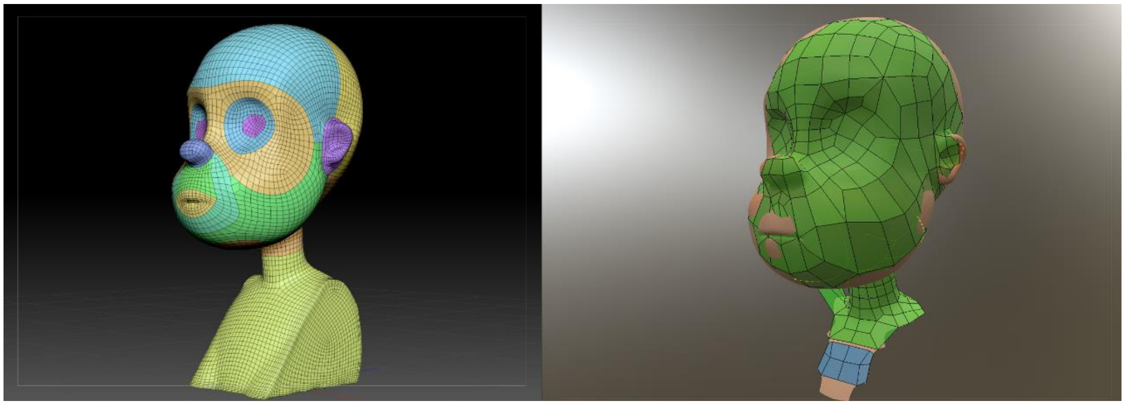


Figure 26- Image of the result when using automatic retopology tools. On the left the result of Zbrush and on the right the result of 3D Coat.

The result is quite good with Zbrush but it is far from perfect to be used as a mesh that can be animated since the edge flow is not correct. Another note is that the number of polygons is still too high for a low-poly mesh. With 3D Coat a similar process can be done by using the Autopo function, but the result is even worse.

Manual Retopology

Almost every 3D software has tools to retopologize manually and most of them are similar. Maya has the Quad Draw tool which is an excellent tool for retopology but it has one major issue, which is that it can get very slow during using it. This is mostly because Maya is typically not used with high-resolution models. 3D Coat, however, is able to handle high-resolution models and its retopology tools work similar to Quad Draw and other software like TopoGun, therefore the manual retopology tools are tested and explained using 3D Coat.

Manual Retopology Tools

Brush

The brush tool can be used to move multiple vertices together, the amount of vertices is determined on the brush size. However, the brush tool is most used to smooth out the spacing between vertices, creating evenly sized polygons.

Strokes

With the strokes tool, you can draw the polygons much like the Topology brush in ZBrush. Loops can easily be created by drawing from outside the model to create cylinders.

Points/Faces

With the Points/Faces tool, individual vertices can be placed on the surface of the high-poly model. When three or four vertices are created the empty space can be filled up creating a face. With this tool, you can also split an entire edge loop.

Quads

With the Quads tool, faces can be created by clicking an existing edge and adding two vertices. It is also possible to click existing vertices to connect the face to them.

Methods and Techniques

When creating the topology for a character the way the edge loops flow is important for good deformation. One of the hardest part of a human character to retopologize is the face, but there are a few techniques and method in making this process easier. The first technique is to make use of poles. Poles are used to break an edge loop and to connect edge loops together. Typically there are two kinds of poles, N-poles which are vertices that are connected to three edges and E-poles which are vertices that are connected to five edges. However, poles can create pinching when deformed so it is important to only use them when necessary and try to hide them at places that are not visible, like under the hair. Poles can also be used to divide poly loops, for instance breaking two poly loops into one or three loops into one. This way you can create less density for places that do not need it, like for instance the back of a character. Another way to break an edge loop is to use triangles, but triangles should only be used in certain ways without causing bad deformation as explained earlier. (Topology-guides, 2018)

The easiest method for retopologizing a character is to look up reference about how the poly loops are distributed, and then to create each loop first and connect them afterward. This method can be used to create the face and body topology. Another good technique is to have a number of edge loops that can be divided by two so that it can be evenly distributed and divided. For example, the loop around the arm should consist of either four, six, eight or sixteen loops, depending on the polycount target. It is also good to first keep the polycount as low as possible and divide everything later for better distribution.[4]

7.3.4 UV mapping

Tools

The process of creating a UV map is similar in most 3D software and so are the tools. There are ways to automate the process but these don’t give a good result because they use a lot of unnecessary seams. The three most used tools for creating UVs manually are creating seams or also called cutting seams, deleting seams or also known as sewing, and unfolding the UV shells. Another useful tool to check if the texture is stretching or squishing is the checker box texture.

Method and Techniques

The techniques and methods for creating UV maps manually are pretty straightforward. Try to keep the number of seams as low as possible and hide them on places of the model that are not visible. Keep everything that needs to have a different texture apart from each other, like hair, clothing, skin etc. It is also possible to use multiple UV maps for one model but not preferable when creating models for low-end devices since this will cause the model to have more materials in the game engine. When creating seams try not to put them on edge loops that have poles because this can cause distortion in the texture. If the model is symmetrical which characters usually are, the shells that are similar can be stacked onto each other in order to save UV space.

7.3.5 Baking and Texturing

The process of creating normal maps, ambient occlusion maps, specular maps etc. is called baking and are a part of physically based rendering or PBR textures. To create PBR textures there are multiple software that can be used but the ones most used are Substance Painter and Marmoset Toolbag 3. In addition, Photoshop can also be used but this is not the optimal software for creating PBR textures.

Method and Tools

The baking process is usually done by importing the high-resolution mesh made during the sculpting phase and the details are then baked onto the low-poly version. This process is pretty much the same in both Substance Painter and Marmoset, but Marmoset has an intuitive way of editing the cage of the model, which can fix bugs easily. However, Marmoset has no tools for creating textures other than using the baking techniques.

If further editing must be done to create a different effect for texturing, this has to be done in Substance Painter. Within Substance Painter, texture can be edited by using all sorts of brushes that can be used to add color, add detail to the normal map, or any of the other maps. These brushes can be used on the 3D model itself or can be used on the 2D texture. Different effects can be used on top each other by using layers. There are also procedural methods that can be used to create all sorts of effects like, blurring or masking certain part of the model. Another tool to quickly add different effects is to make use of Smart Materials. Smart Materials basically are pre-made effects that can be added to the model. There are all kinds of Smart Materials to quickly add different effects like skin, wood, scratched metal etc. However, all these Smart

Materials are mostly used to create realistic effects as possible and are not so much used for stylized art.

8 Conclusion and discussion

It takes a lot of work to create 3D characters and the workflow in creating them can differ very much. However, it seems that the workflow and pipeline for 3D characters intended for low-end devices are not so much different as that as for high-end devices. In fact, this research shows that the methods and techniques used for high-end devices can also be applied to those that are intended for low-end devices. However, the research performed in this paper does not include much older models of low-end devices, which may show different results. Another thing to take into account is the fact that the technology for creating 3D models keeps on developing and that the methods and techniques described in this research will be outdated within a couple of years.

9 Recommendations and graduation products

For creating stylized characters it is recommended that all the design principles should be kept in mind during the entire process of creating them and not only when the characters are designed. For 3D characters intended for low-end devices, the biggest bottleneck is the number of polygons they contain and it is recommended that they should be kept at a minimum. Optimizing should not be done after the creation of the 3D characters but should be taken into account before creating them. The maximum polygon amount, the resolution of the textures, what kind of textures should be used, and how many materials per characters are going to be used all need to be established beforehand. This should be determined by the target device, the style of the game, and how many models will be shown on screen at the same time.

The high-poly to low-poly workflow is highly recommended when creating 3D characters intended for low-end devices, because this allows baking the high-poly details onto the low-poly model, giving it much higher detail. When using this method and start out by creating a base mesh to sculpt on, the best method for creating this base mesh is to use the IMM method. When creating the retopology of a 3D game model, it is recommended to do this manually. Automatic or semi-automatic methods don’t give good results that can be used for animation and rigging. When creating the topology of a 3D character manually it is advised to first create the necessary edge loops for good deformation, and connect these loops to each other by filling in the gaps as necessary, since this method is the least time-consuming. Just like with creating the doing the retopology, it is recommended that creating the UV maps should be done manually. And finally, the baking could be done by either using Substance Painter or Marmoset Toolbag and for editing textures, it is advised to use Substance Painter. The models created can be found online on sketchfab.com and the links are in the annex.[1]

10 Sources

- 3dtotal Publishing. (2018). Creating Stylized Characters. Worcester, United Kingdom: 3DTotal.com.
- Aava, K. (2017, December 12). Realistic vs. Stylized: Technique Overview. Retrieved January 6, 2019, from https://80.lv/articles/realistic-vs-stylized-technique-overview/
- Alkemi, A. (2013, August 15). A game of Tricks II – Vertex Color | Alkemi [Forum post]. Retrieved January 6, 2019, from http://www.alkemi-games.com/a-game-of-tricks-ii-vertex-color/
- Amin, J. (2013, July 24). Maya Modeling: Polygonal Modeling Theory. Retrieved December 18, 2018, from https://3dtotal.com/tutorials/t/maya-modeling-polygonal-modeling-theory-jahirul-amin-character-face
- Anhut, A. (2016, January 30). Stylized Realism - Howtonotsuckatgamedesign. Retrieved January 6, 2019, from http://howtonotsuckatgamedesign.com/2016/01/stylized-realism/
- Autodesk. (n.d.-a). Animation Concepts. Retrieved January 5, 2019, from http://help.autodesk.com/view/3DSMAX/2016/ENU/?guid=GUID-ADB8D6B2-392D-49BF-B941-1414FBD97202
- Autodesk. (n.d.-b). Graph editor. Retrieved January 5, 2019, from http://help.autodesk.com/view/MAYAUL/2017/ENU/?guid=GUID-6D38EAEA-6032-471E-BD0E-54A74D4443C0
- Blender. (n.d.). Overview — Blender Manual. Retrieved January 5, 2019, from https://docs.blender.org/manual/en/dev/editors/uv_image/uv/overview.html

- Blender. (2017, January 7). Combine a Mat map and a Normal map in Blender? [Forum post]. Retrieved January 6, 2019, from https://blender.stackexchange.com/questions/70685/combine-a-mat-map-and-a-normal-map-in-blender
- Chung, C. (2014, August 15). Cartoony Visuals for a Silly 3D Destructive Feline Game [Forum post]. Retrieved January 6, 2019, from https://www.tumblr.com/privacy/consent?redirect=http%3A%2F%2Fcatlateraldamage.tumblr.com%2Fpost%2F94829888452%2Fcartoony-visuals-for-a-silly-3d-destructive-feline
- Fabian, F. (2017, May 16). Key 3D Modeling Terms Beginners Need to Know. Retrieved December 3, 2018, from https://i.materialise.com/blog/en/3d-modeling-terms/
- Fisher, A. (2013, August 13). Create a Game Character: Jouster - part 8. Retrieved January 5, 2019, from https://3dtotal.com/tutorials/t/create-a-game-character-jouster-part-8-adam-fisher-character-human-female
- Giroux, S. (2016, August 18). Create game-ready textures using Substance Painter. Retrieved January 5, 2019, from https://www.creativebloq.com/how-to/create-game-ready-textures-using-substance-painter
- Gosch, P. (2017, July 6). 3D model optimization for mobile devices. Retrieved January 6, 2019, from https://codefluegel.com/en/3d-modelloptimierung-fuer-mobile-endgeraete/
- Mäkelä, N. (2011, November 14). Low-poly Tips 2 – Game Art Asset Optimization | CGmascot. Retrieved January 6, 2019, from http://www.cgmascot.com/design/low-poly-tips-2/

- Nadezhdin, K. (2018, March 29). Going back to low poly palettes. Retrieved January 6, 2019, from https://medium.com/@cyrilltoboe/going-back-to-low-poly-palettes-1689fbe84ce1
- Olson, S. (n.d.). Character Design [Video file]. Retrieved January 6, 2019, from https://www.3dcharacterworkshop.com/products/3d-character-workshop/categories/193560
- Paulino, P. H. (n.d.). 6 Tips to improve your uv mapping workflow. Retrieved January 5, 2019, from http://www.paulhpaulino.com/6-tips-to-improve-your-uv-mapping-workflow/
- Pluralsight. (2014a, December 26). How to Setup Reference Images Easily in 3ds Max. Retrieved December 3, 2018, from https://www.pluralsight.com/blog/tutorials/setup-reference-images-easily-3ds-max
- Pluralsight. (2014b, October 1). Why Are Ngons and Triangles so Bad? Retrieved December 4, 2018, from https://www.pluralsight.com/blog/film-games/ngons-triangles-bad
- Pluralsight. (2014c, January 19). Understanding UVs - Love Them or Hate Them, They're Essential to Know. Retrieved January 5, 2019, from https://www.pluralsight.com/blog/film-games/understanding-uvs-love-them-or-hate-them-theyre-essential-to-know
- Pluralsight. (2014d, January 14). Key 3D Rigging Terms to Get You Moving. Retrieved January 5, 2019, from https://www.pluralsight.com/blog/film-games/key-rigging-terms-get-moving

- Pluralsight. (2014e, August 11). Understanding the 12 Principles of Animation. Retrieved January 5, 2019, from https://www.pluralsight.com/blog/film-games/understanding-12-principles-animation
- Pluralsight. (2015, January 17). Create 3D Models More Quickly with a Model Sheet. Retrieved December 3, 2018, from https://www.pluralsight.com/blog/film-games/create-3d-models-3x-faster-with-a-model-sheet
- Polycount Wiki. (2015a, May 24). Texture types - polycount. Retrieved January 5, 2019, from http://wiki.polycount.com/wiki/Texture_types
- Polycount Wiki. (2015b, May 15). Limb Topology - polycount. Retrieved January 6, 2019, from http://wiki.polycount.com/wiki/Limb_Topology
- Polycount Wiki. (2016, May 12). Texturing - polycount. Retrieved January 5, 2019, from http://wiki.polycount.com/wiki/Texture
- Polycount Wiki. (2017, March 7). Topology - polycount. Retrieved December 18, 2018, from http://wiki.polycount.com/wiki/Topology
- Polycount Wiki. (2018, May 28). FaceTopology - polycount. Retrieved December 18, 2018, from http://wiki.polycount.com/wiki/FaceTopology
- Polycount. (2013a, November 26). Is it possible to bake normal maps into diffuse maps? [Forum post]. Retrieved January 6, 2019, from https://polycount.com/discussion/128428/is-it-possible-to-bake-normal-maps-into-diffuse-maps
- Polycount. (2013b, August 17). [MAX] Baking Normal map lighting info into diffuse map... [Forum post]. Retrieved January 6, 2019, from

https://polycount.com/discussion/124757/max-baking-normal-map-lighting-info-into-diffuse-map

- Silverman, D. (2013, March 5). Attention Required! | Cloudflare. Retrieved January 6, 2019, from https://gamedevelopment.tutsplus.com/articles/3d-primer-for-game-developers-an-overview-of-3d-modeling-in-games--gamedev-5704
- Slick, J. (2017, October 17). 7 Common Modeling Techniques for Film and Games. Retrieved December 3, 2018, from https://www.lifewire.com/common-modeling-techniques-for-film-1953
- Slick, J. (2018a, September 14). Polygonal 3D Modeling Common Box and Edge Workflows. Retrieved December 3, 2018, from https://www.lifewire.com/polygonal-3d-modeling-2139
- Slick, J. (2018b, November 13). How Are 3D Models Prepared for Animation? Retrieved January 5, 2019, from https://www.lifewire.com/what-is-rigging-2095
- Spencer, S. (2011). ZBrush Character Creation: Advanced Digital Sculpting (2nd ed.). Indiana, United States of America: Wiley.
- Taylor, J. (2015, September 29). Why are triangles bad when modeling? Retrieved December 18, 2018, from http://www.methodj.com/why-are-triangles-bad-when-modeling/
- Topology-guides. (2018, July 2). Moving and Manipulating Edge Poles. Retrieved January 20, 2019, from https://topologyguides.com/post/175483457845/moving-and-manipulating-edge-poles

- Unity Technologies. (2013, October 25). Mobile Max Texture Size - Unity Answers [Forum post]. Retrieved January 6, 2019, from https://answers.unity.com/questions/563094/mobile-max-texture-size.html
- Unity Technologies. (2017, March 29). Unity - Manual: Modeling characters for optimal performance. Retrieved January 6, 2019, from https://docs.unity3d.com/550/Documentation/Manual/ModelingOptimizedCharacters.html
- Valve. (n.d.). Dota 2 Workshop - Color Texture Light Baking - Dota 2 - Workshop - Knowledge Base - Steam Support. Retrieved January 6, 2019, from https://support.steampowered.com/kb/8700-SJKN-4322/dota-2-character-texture-guide
- Wikipedia. (2018, November 29). document used to help standardize the appearance, poses, and gestures of an animated character. Retrieved December 3, 2018, from https://en.wikipedia.org/wiki/Model_sheet
- Wikipedia. (2019, January 3). 12 basic principles of animation - Wikipedia. Retrieved January 5, 2019, from https://en.wikipedia.org/wiki/12_basic_principles_of_animation

11 Annexes

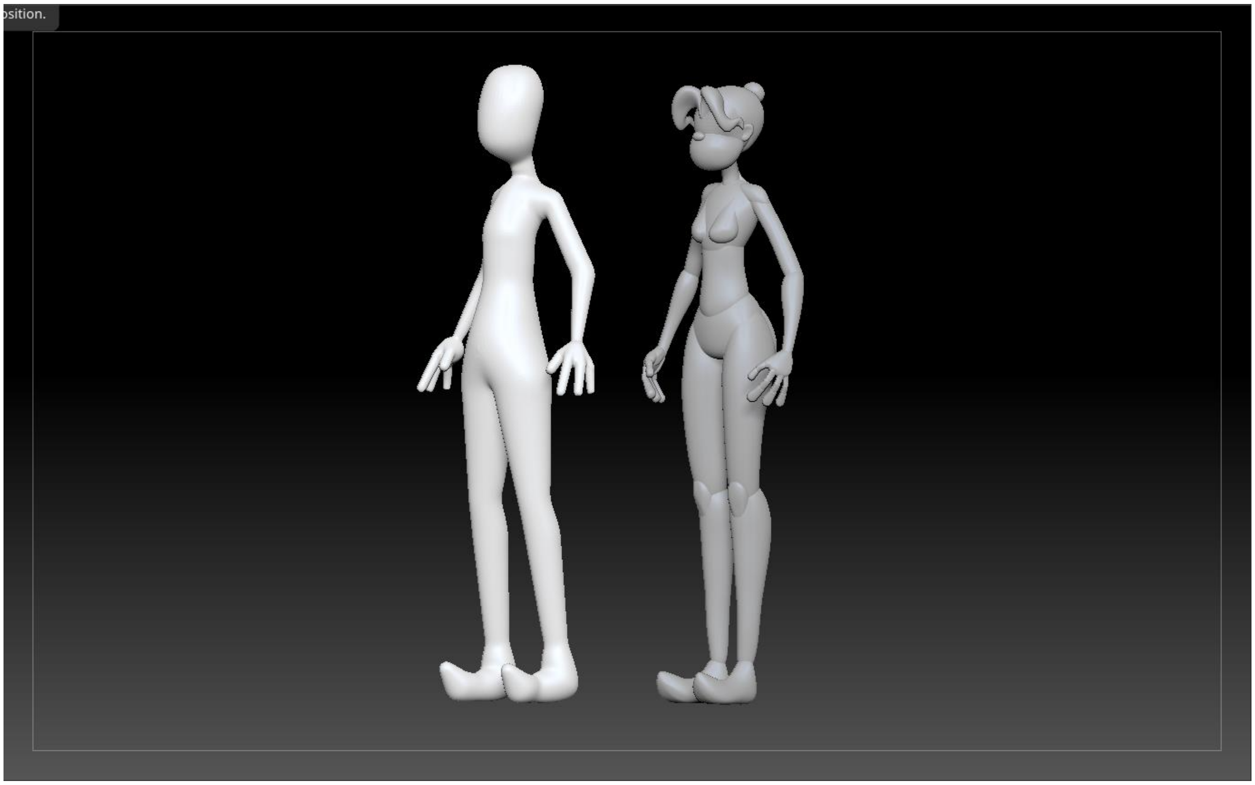
[1]Links to graduation products

https://skfb.ly/6Gu6v

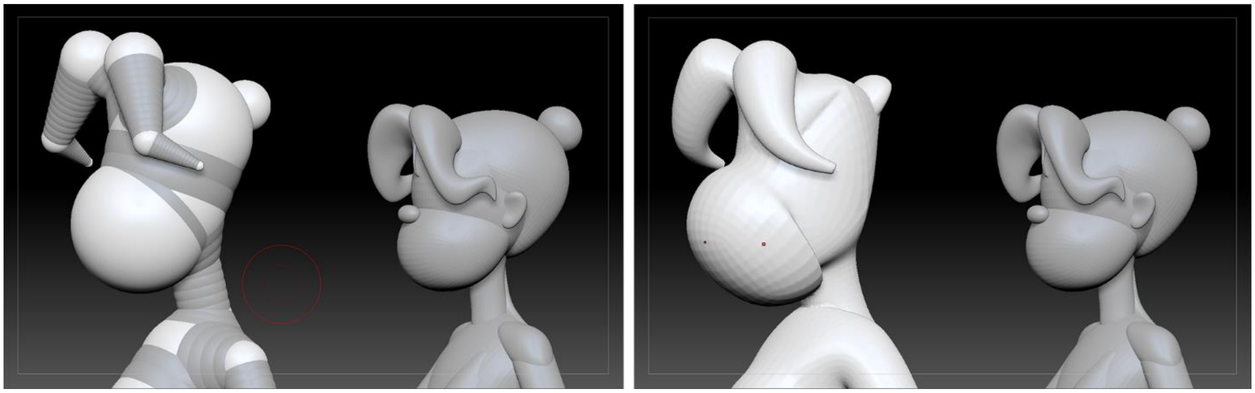
https://skfb.ly/6GunY

https://skfb.ly/6G9QV

[2] Image of ZSpheres compared to the IMM method.



[3] Image of the ZSpheres bug.



[4] Example retopology process.