NATIONAL CENTER FOR COMPUTER ANIMATION

Muscle and soft tissue simulation within a character pipeline based on Autodesk Maya

Master Thesis

Johannes Saam

10/10/2007

A muscle simulation tool for realistic character deformation inside Maya based on its internal mechanisms.

I dedicate this Master thesis to someone who left a lot to behind to come explore the world with me, Marlen.

TABLE OF CONTENTS

1. Introduction 4

2. Related work 5

3. Biological background 6

3.1 Bones 6

3.2 Muscles 6

3.3 Connective tissue 7

3.4 Skin 8

4. The character Pipeline 9

4.1 The different steps 9

4.2 How to connect the steps 9

4.3 Efficiency 10

5. Simplification 11

5.1 Muscles 11

5.2 Skin 12

5.3 Final simplificated muscle model 13

6. The system 14

6.1 Technical background 14

6.1.1 Maya DAG setup and Utility nodes 14

6.1.3 Maya`s skinning features 15

6.2 The development process 15

6.3 Muscle generation 16

6.4 Muscle modeling 17

6.5 Secondary motion 18

6.6 Skinning 19

6.7 User interaction 19

7. Results 21

7.1 A simple example 21

7.2 The cave troll 22

7.2.1 The trolls arm 22

7.2.2 The Trolls belly 24

8. Conclusions and future work 25

9. Appendix 26

9.1 Special MEL Code examples 26

Bibliography 27

1. INTRODUCTION

One of the most demanding areas in 3D computer graphics is the realistic reproduction of creatures of all kind. Many different tasks in a graphics pipeline work together to get this job done. In this master thesis the focus is on the rigging and skinning bit. Starting from a finished model a set of tools based on Autodesk`s Maya was developed to help an *artist*[1] to introduce realistic skin movement on top of bones. The human skin is a very complex organ and produces very different subtle effects that make or break character`s realism.

To get a production ready tool varying criteria have to be fulfilled. Due to the complexity the problem has to be simplified. The human body has a vast verity of different muscle types and styles. Also the muscle count is very high. These muscles have to be classified and abstracted to simpler models in order to keep the system fast and efficient. The same scheme applies to the skin itself. The connection between the skin and the underlying tissues is very complex and exposes too many parameters in order to be controllable.

After simplification a Maya muscle model was developed to represent muscles inside the body and to build the base for realistic skin deformation. The skin is then connected with the muscles for final deformation and special control objects are introduced for skin properties adjustment.

The main goals of the system are speed, stability and flexibility for special changes. Every character presents different challenges and therefore the system is open for changes to adapt to any tasks that may be needed.

[1] The words *artist*, *animator* and *user* are used interchangeably. They describe the user of systems to deform characters in a visually pleasing way.

2. RELATED WORK

Character skinning and animation is a wide area in computer graphics. It was approached over the years from many researchers. But all those early attempts were not geared to usability. Smooth skinning (1) was introduced by Mohr and Gleicher. It is implemented in all major animation packages today. The technique is described later in 6.3.1. Turner was the first looking on the human anatomy to create a system that is more realistic. (2) He divided the problem in layers of fat muscles and bones. This approach is still state of the art but has evolved over time. The modeling of muscles is approximated by simple ellipsoids. This estimate is not accurate enough to solve many motions. Scheepers tried to generate muscles based on ellipsoids but took it one step further by introducing fusiform and multi belly muscles. (3) These muscle types are described in 5.1. All these techniques were computationally expensive and also counterintuitive for artists. Other methods like pose spaced deformations (4) and non linear binding (5) gave the control back to the artist. This control is needed in a movie production to fulfill the demands of directors and producers. Another different approach to smooth binding was done by Yang (5). He introduced a curve skeleton. A curve replaces the ordinary bone setup to deform the character. This gives more flexibility and is easier to set up. The muscle setup presented here uses the curve skeleton`s idea as a base for deformation of the muscles. Further information is provided in 6.3. In order to create photo real deformations the simulation of internal structures is needed. To speed up that process Yang created a method to generate muscle shapes automatically based on the skin mesh layer. (7) The need of artist control is obvious. Saam and Jiang created a method to skin characters in a fast and production friendly way.(8) The system that they created took Scheepers and Thalmans ideas from the past and geared them to the modern production pipeline. The muscle simulation system presented here is based on this idea to maintain speed and user control at any step inside the pipeline.

3. BIOLOGICAL BACKGROUND

To understand the complexity of the problem and to take advice from nature a brief overview of the anatomical features of the human body considering all the different parts that make the skin move is following.

3.1 BONES

All mammals have an endoskeleton that gives them strength and flexibility. Bones are rigid organs with several tasks. Bones protect inner organs, for example the skull protecting the brain. They also give shape to the body and support its inner structure. A major task is the propagation of forces over longer distances to make the body move. The production of red and white blood cells is also done mainly in the bones. A bones` internal structure is very complex and reminiscent of a three dimensional honey-comb. It consists of a light weight material formed by calcium phosphate. Several different structures combined form the bone. (1) (2)

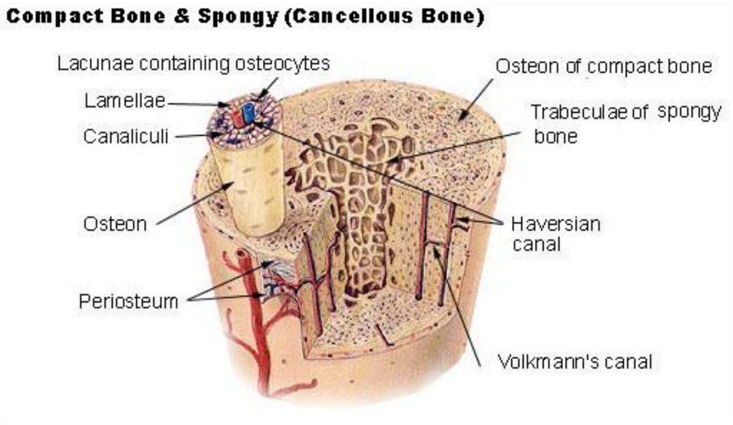

Compact Bone & Spongy (Cancellous Bone)
Lacunae containing osteocytes
Lamellae
Canaliculi
Osteon
Periosteum
Osteon of compact bone
Trabeculae of spongy bone
Haversian canal
Volkmann's canal

Figure 1 the human bone structure

3.2 MUSCLES

The muscles are the contractile tissue inside the body to produce motion. Three different types of muscles move internal organs or lead to locomotion. The *skeletal musculature* is moved voluntary. This means that the human brain controls its motion on a decision based level. Big muscle groups are always skeletal muscles. The skeletal musculature is the kind of muscles that the animator is interested. It moves the body and is visible from the outside. The two other types of muscles are non voluntary. They move inner organs like the digestive system and are divided in *smooth* and *cardiac musculature*. These muscles are important for the body to survive but not as important for animation as the skeletal musculature. (1) (3)

The anatomy of a muscle is like the bone, a layered structure. A muscle contains vast amounts of fibers. These fibers, also called myofibrils, contain a protein structure called myosin and actin. Those cells contract on stimulation from nerve endings and result in motion.

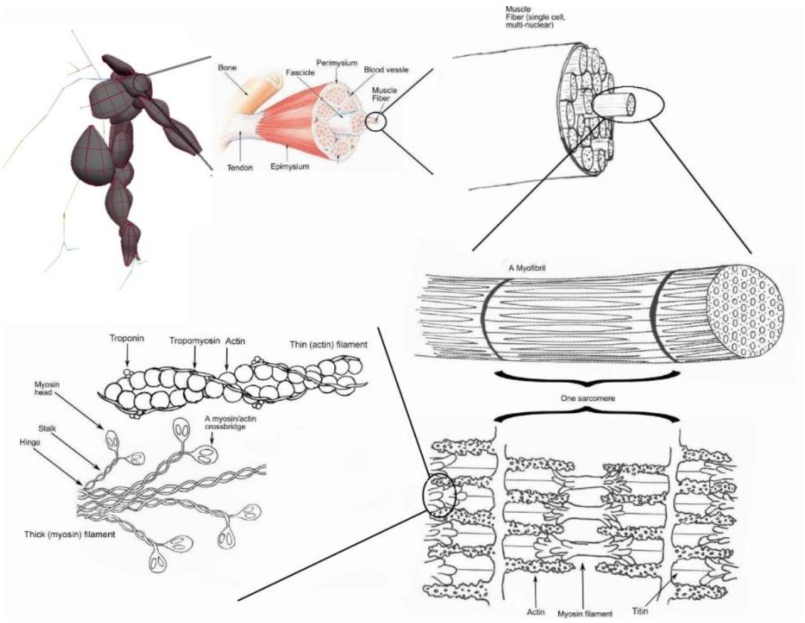


Figure 2 the muscle structure

3.3 CONNECTIVE TISSUE

Connective tissue is a rather vague expression for all different kinds of cells and proteins that are found between muscles and the outer skin. One kind of tissue called the *dense connective tissue* forms ligaments and tendons. This can be simulated easily by the system with replacing it by virtual muscles. A rather unimportant part for animation is the *reticular connective tissue*. This tissue provides the body with skeletal support. The two other connective tissues are very important for 3D computer graphics. *Areolar connective* tissue holds the inner organs in place and connects the muscles with the outer skin and fat layers. It mainly consists of long protein fibers like collagen. Collagen fibers are like little springs inside the body to hold the skin in place. The older the body becomes the looser these fibers get and the skin starts to get more lose, too. *Adipose tissue* connects the skin with the collagen fibers. It provides the skin with cushioning against heavy movement and lubricates the skin so it can slide on top of the muscles. These tissues must be observed in much heavier detail to be able to archive realism in the skinning sector of computer graphics. (1) (4)

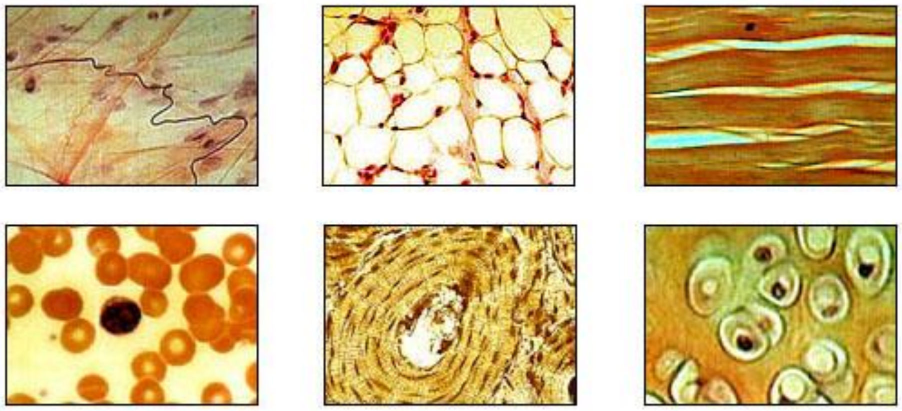


Figure 3 different connective tissues

3.4 SKIN

The skin is the biggest organ of the human body. It protects the body from peripheral influences like temperature and impact. The skin has the ability to replace itself approximately every 20 days completely. Heavily damaged skin heals itself by producing scar tissue. That tissue is less flexible and protects the damaged areas.

The anatomy of the skin is again layered. The dermis is the underlying structure that contains hair roots and sweat glands. Those glands end in pores that translate the fluids outwards. The epidermis is the top structure, which provides feedback to the body. Many different nerve endings are placed along the epidermis and form a network of cells. (1) (5)

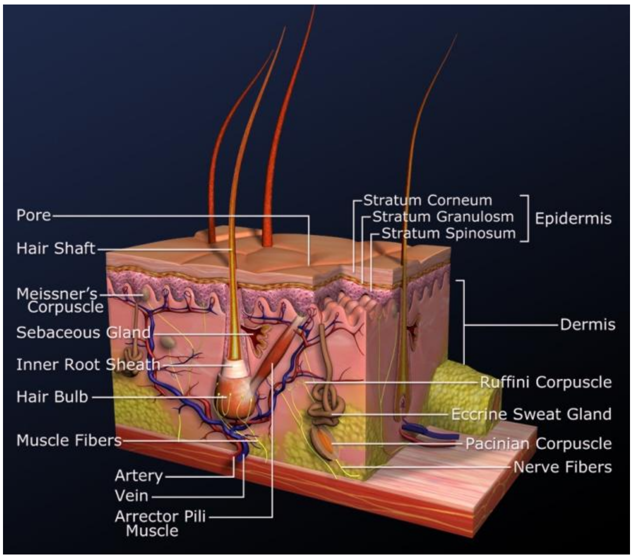


Figure 4 the skin layers

4. THE CHARACTER PIPELINE

Every job in 3D graphics has to be pipelined. The bigger the task, and the complexity, the bigger is the need for a special pipeline to keep the work manageable.

4.1 THE DIFFERENT STEPS

After the modeling of a character is completed the character has to be moved on as quickly to the animation phase to keep production times as short as possible. Virtual bones have to be inserted in the character and intelligent ways to control their movement have to be implemented. With that animation controls in place the animation phase can begin. In the mean time the deformations have to be established. Any movement causes the skin to react in different ways and solutions have to be found for all kinds of cases. This leads to a linear pipeline as seen below. (6)

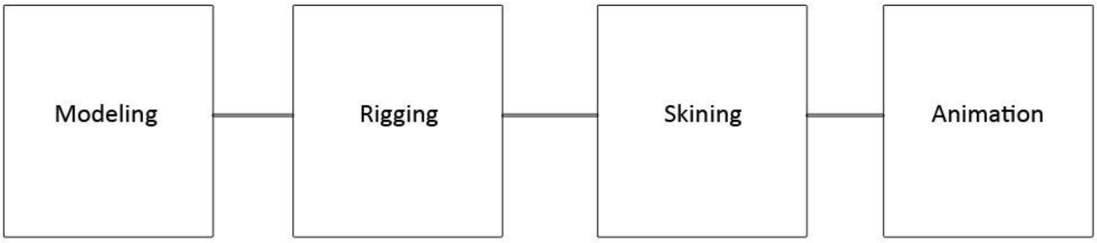


Figure 5 the linear pipeline

4.2 HOW TO CONNECT THE STEPS

To be able to be free and open for changes along the way the pipeline has to be opened up from a strictly linear way to a more complex model. Rigging and skinning can be parallelized even to a certain extent with animation by introducing another step called joint placement. With the joints in place the rigging and skinning can be started. The problem is to go back and forth between those steps to agree on a final product. In Maya for this project it was realized by duplicating the whole joint hierarchy and generating a bound skeleton and a rig skeleton. These two different hierarchies where then split up to generate the animation rig on the rig skeleton. In the mean time skinning could be done on the bound skeleton. To link the two joint chains up a fairly simple MEL script was written to connect the rig with the bound skeleton via a parent constraint. This type of constraint inherits the transformation of one object to another one. Even animation can be started by providing the rig hierarchy with a representation of the model. An appropriate way of doing so is to cut the geometry in small pieces that then can then be parented with the same constraint techniques to the joints. Now an animator has a relatively close idea how his motion will look at the end without having to wait for the sometimes very time consuming step of skinning. Another “must have” in a modern character pipeline is the possibility to go back and forth between the different steps to find solutions specific for the production. Additional animation controls might be needed or extreme motion may lead to different deformation. By having two different joint hierarchies changes can be made additional setups can be inserted as long as the main bound, and rig joints can be linked and unlinked by scripts. Several methods can be used to identify the corresponding joints. For example naming conventions or joint labels are used for this project. All this ideas lead to another pipeline as follows. (6)

s

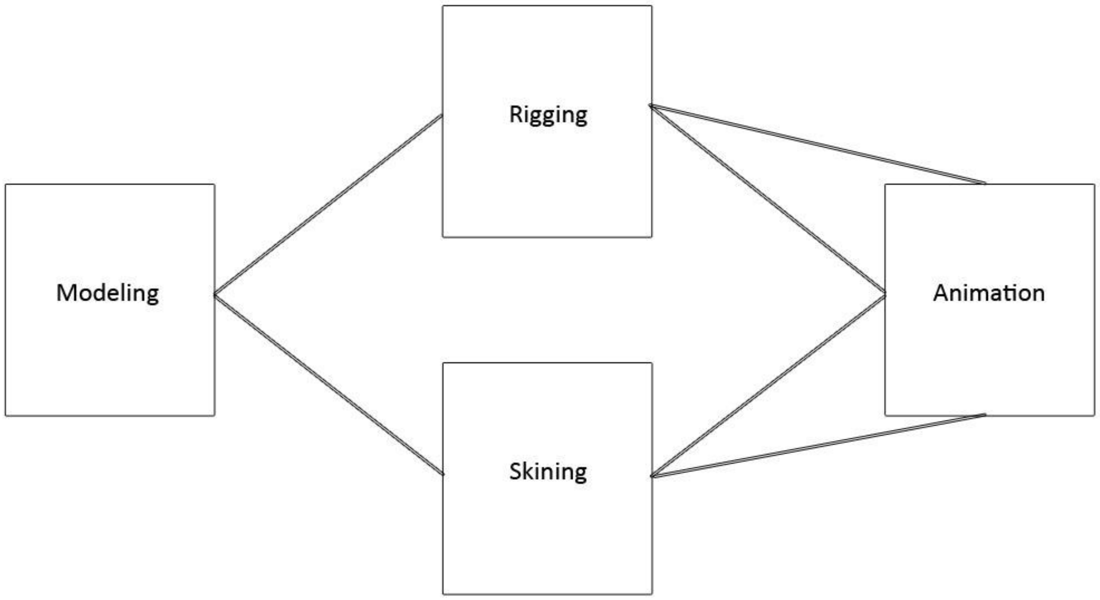


Figure 6 the non linear pipeline

4.3 EFFICIENCY

To keep a fast working speed the pipeline has to be slim and easy to use. The small script runs very fast and new joints can be added on top without reconsidering the script. A strong naming convention is used to specify the corresponding joints. At any time the two skeletons can be attached, detached and reset. This allows fast turnaround times and an easy way of parallelizing the process. Also a unique way of organizing MEL scripts was developed. The internal "what is" command can be used on the name of any script to return the directory in which the script is placed. From there custom folders can be searched and scripts can be sourced on the fly to make sure every script is hooked up to the system. Refer to the MEL code section in the appendix for further information.

5. SIMPLIFICATION

To be able to come up with a smart and fast system the problem of muscle and skin motion has to be simplified to a certain extend. The high goal is maintaining the visual realism by cutting down on complexity. Visual pleasing results are always the number one objective, but in order to keep it manageable and expose practical controls to the animator it has to be broken down to a smaller level in the following areas.

5.1 MUSCLES

There is a variety of different muscles in the human body. Mainly two forms are important to influence the skin. Fusiform muscles are simple ellipsoid shapes that span between one origin and one insertion point. (3)

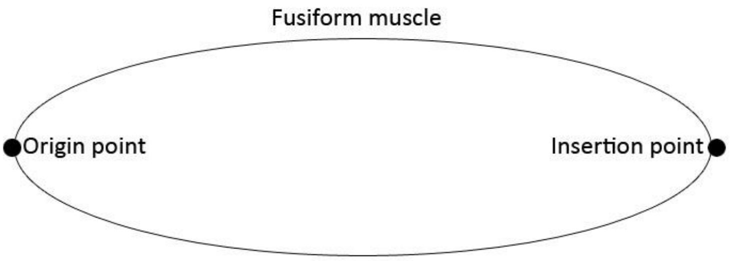


Figure 7 fusiform muscle

Multi belly muscles are more flat muscle shapes that are connected on a broader curve along a bone. These muscles are more complex and pose a much bigger task to the system then fusiform muscles. (3)

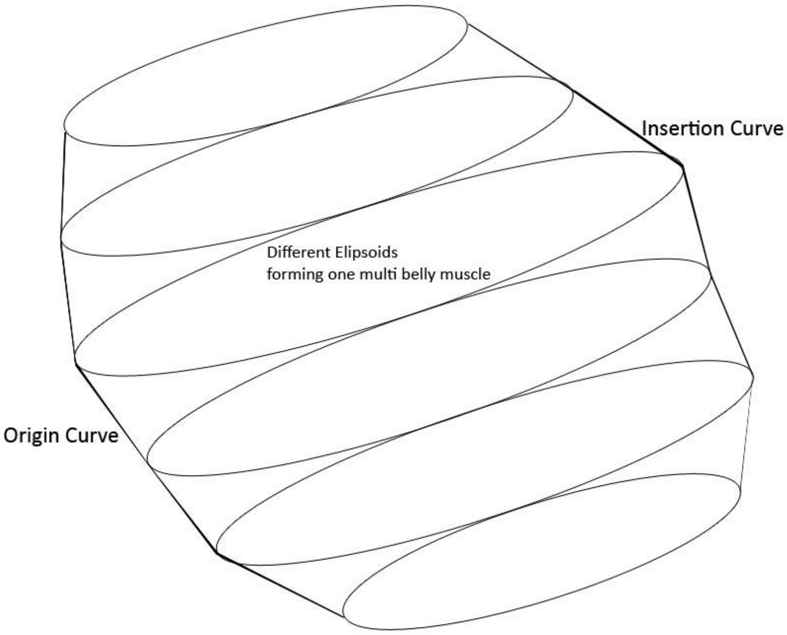


Figure 8 multi belly muscle

Fusiform muscles are simple and must not be simplified. Multi belly muscles on the other hand have to be broken down in smaller pieces. One multi belly can be represented and approximated by several fusiform muscles that are aligned along a curve. This might not be physically correct but feeds the purpose and doesn't have to be exposed to the animator. The animator controls one major muscle that drives several simpler muscles to minimize computation times.

Another important fact is that the human body consists of many muscle groups. The leg for example has almost a hundred small and big muscles that make the many different motions of the leg possible. This amount of complexity is not needed and can also be broken down. Depending on the need for a production muscle groups can be put together to bigger muscles that approximate the shape. By doing that the main underlying motion is maintained but the system gets simplified a lot. Visual artists use that a lot in fine art. (7)

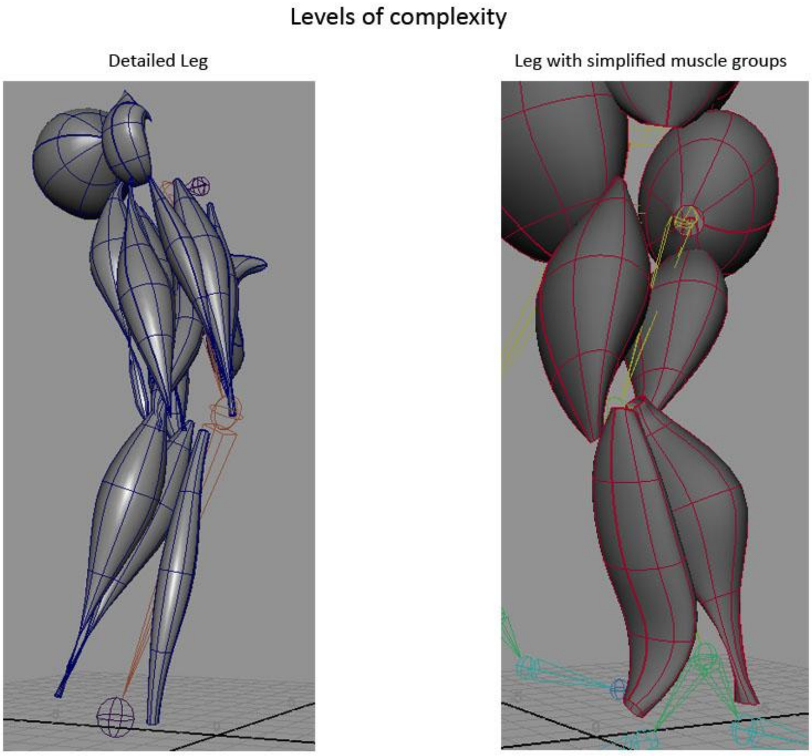


Figure 9 different levels of complexity

5.2 SKIN

In order to simulate the skin and all its connective tissue with veins, nerve fibers and organs the complexity raises to an absolute impossible level. The main features that can be seen from the outside are jiggling and skin sliding. Jiggling is the secondary motion that happens on drastic changes in motion and is due to the mass of flesh moving inside the body and due to connective tissue and organs moving underneath the skin. To achieve that motion the muscle itself is made dynamic and simulated with gravity and mass with a simple mass – spring system. The skin is then hard bound to the muscle surface to transmit that motion. An extra layer of skin jiggling is not needed because the muscle itself deals with it. In order to simulate fat or organs moving fake muscles are introduced that may not be in a human body but have visually the same effect. The belly region is a good example. A fat belly might consist of a bigger more round muscle to fit the shape of the body. Together with a realistic amount of muscle jiggling on that “fake muscle” the motion of fat and organs pushing the inside of abdomen are can be simulated realistically.

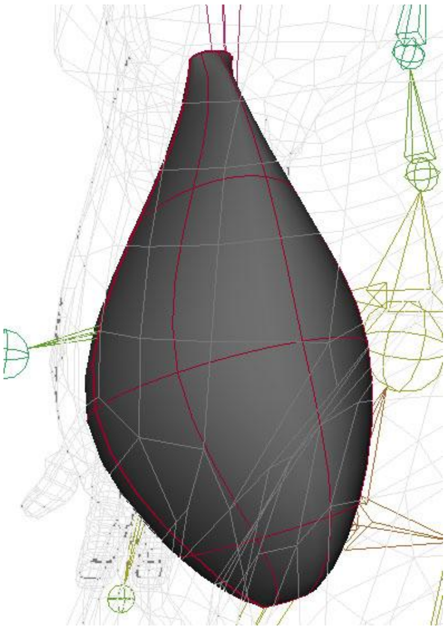


Figure 10 A Fake belly muscles

5.3 FINAL SIMPLIFICATED MUSCLE MODEL

As a result of 4.1 and 4.2 a final muscle model is created with the following characteristics.

Only one origin and insertion point was chosen. This makes controls and attachment to the skeleton easier. Between these two points a NURBS curve was created. This curve is referred from now on as action curve. The curve is controlled by control objects at the origin, the insertion and one point in the middle of the curve. With those objects the curve can be modified in length and shape via clusters at control points on the curve. The tangents at the origin and insertion point can also be adjusted.

Along the action curve five cross section curves were created. Those cross sections define the actual shape of the muscle. Starting with circles any shape can be archived by modeling. The cross sections fall off in size from the center of the curve to the attachment points. The cross sections are aligned along the action curve and rotated according to its tangent. To finalize the shape of the muscle the cross sections get lofted to form a single NURBS object. Cross sections get scaled according to the arc length of the curve to maintain volume. The shape of the whole object is maintaining the volume precise enough when the individual cross sections maintain their volume. (7)

To simulate the different jiggling motions a soft body simulation was then attached to the muscle. Particles were instanced to the control vertices of the muscle and then simulated with gravity to give mass and inertia.

The skin was then bound ridged to the muscles. By doing so the muscle motion is transferred direct to the skin and makes it jiggle and move in a realistic way.

6. THE SYSTEM

With the theoretical system in place the next step is the implementation in Maya. To find the most efficient and easy to use way several iterations of refinement where taken.

6.1 TECHNICAL BACKGROUND

In order to provide enough information about Mayas mechanisms a brief overview of the tools used is given as follows.

6.1.1 MAYA DAG SETUP AND UTILITY NODES

Maya works node based. Generally speaking Maya is a system based on nodes with attributes that are connected. Those small individual nodes form complex structures and are organized in a direct acrylic graph (DAG). Every maya object contains at least four nodes to show in the viewport. To explain that setup the DAG setup is compared with the construction of a building: The first node is a so called input node. The input node describes the shape of the model like a blueprint of a building. That input node feeds into a shape node that actually creates the object based in its input information. This can compared to a building that gets made according to the blueprint. Every shape node is connected with a transform node that set the model in the world space and manages its transformation. This is basically the position of the building in the neighborhood. The last node is a shading group node that holds material properties of the object to display it. This can be compared to the paint and coating of the building. Every step in the construction process can contain as many sub nodes as desired and every level of complexity can be assembled. (8)

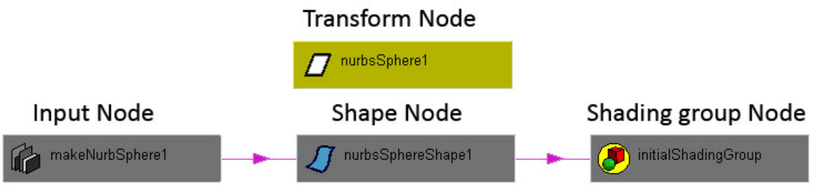


Figure 11 Maya DAG setup

One special sub set of nodes in Maya are called utility nodes. Those nodes where introduced primarily for shading. They contain math nodes that deal with float, vector and matrix math. All those utility nodes are fast in computation and easy to set up. The system is based on those nodes entirely. (9)

Node name	*Usage*
multiplyDivide	Performs multiplication, division and power operations on float values or vectors.
plusMinusAverage	Performs addition, subtraction and average calculation on float values or vectors.
vectorPeoduct	Performs a dot or cross product between vectors or a product between a matrix and a point or vector.
Reverse	Reverses the float or vector value.
Clamp	Clamps the float or vector value to a given range.

setRange	Remaps a range of float or vector values to a new range by specifying minimum and maximum values.
Condition	Works like an *if() {…}* statement by comparing input float or vector values and outputting a new value.
colorBlend	Blends two vector or float values based on a single float value between zero and one.
distanceBetween	Calculates the distance between two points in any given space.

Table 1 Some of Maya`s utility nodes

6.1.2 MEL SCRIPTING

MEL stands for Maya embedded language, Maya`s internal scripting language. MEL is used to control everything inside Maya. Every bit of user interface or user interaction is MEL code. Using a unified scripting language for configuring a whole program it becomes very open and programmable. MEL can control every process inside Maya. MEL was used entirely to develop the system. Recently Autodesk introduced Python as an alternative scripting language for Maya. Python is bound into the Maya API entirely. Even plug-ins can be written in Python. This special plug-in type is called *scripted plug-ins.* (10)

6.1.3 MAYA`S SKINNING FEATURES

Maya provides the user with the standard way of skinning meshes to bones. It divides skinning into two different approaches.

Rigid skinning attaches every vertex of the mesh to a single bone. This is easy to setup but results in simple deformations. Special lattice deformers can be introduced to deal with areas of high creasing. This technique is too simplified and isn’t open enough to be used in this system.

Smooth skinning is an alternative way of binding the skin. Every vertex of the mesh is moved by an unlimited amount of joints. These joints get weighted with a special weight map. This grayscale map contains values from zero to one for every vertex. The sum of all bones influencing this point is one and so all the different components can be blended. This process is the commonly used way of skinning. To be able to use this technique for the system another feature of Maya is used. Smooth skinning also allows the animator to specify special NURBS objects to influence the skin. These objects are treated exactly like bones but operate on a surface level. Every point gets attached to one corresponding point on the surface of the NURBS object. These influence objects feed the purpose of the system. (9) (8)

6.2 THE DEVELOPMENT PROCESS

Due to the fact that the whole system is dependent on Maya`s utility nodes the first step was to generate that system inside the user interface. This was done by manually creating nodes and connecting them. This is a very flexible and quick way to test ideas and techniques. After several iterations to refine the system MEL scripts where written to automatically generate all those setups and exposing the important controls to the user. All these scripts are then packaged together in one tool to generate, model, simulate and skin the muscles.

As the system is based on Maya`s internal nodes only an advanced user can always step in and configure the system for special needs inside a custom pipeline.

6.3 MUSCLE GENERATION

On the generation of the muscle a NURBS curve is created with five control vertices. This number was derived by several tests and five is the perfect average between complexity and simplicity. The first and last vertices are placed at the origin and insertion point. Another vertex is placed exactly in the middle of the points. The last two are placed at 12.5 and 87.5 percent of distance between the origin and insertion. These numbers where necessary because the tangents at the two main points need to be controlled. By moving these control points along the tangent of their parent control objects the action curves tangents can be controlled.

After the creation of the action curve all its control vertices where added to individual clusters and parented to three control objects. One for the origin, one for the insertion, and one for the middle point of the curve. The middle controller is held in the exact center of the two other controllers with a point constraint to a null object parent to the middle controller. This null is introduced to be able to offset the middle controller on top of its constraint.

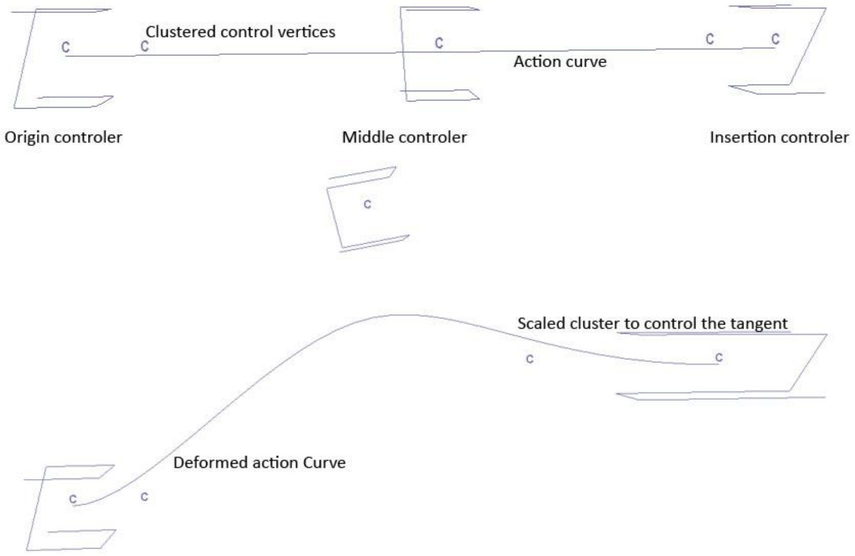


Figure 12 the action curve and its controls

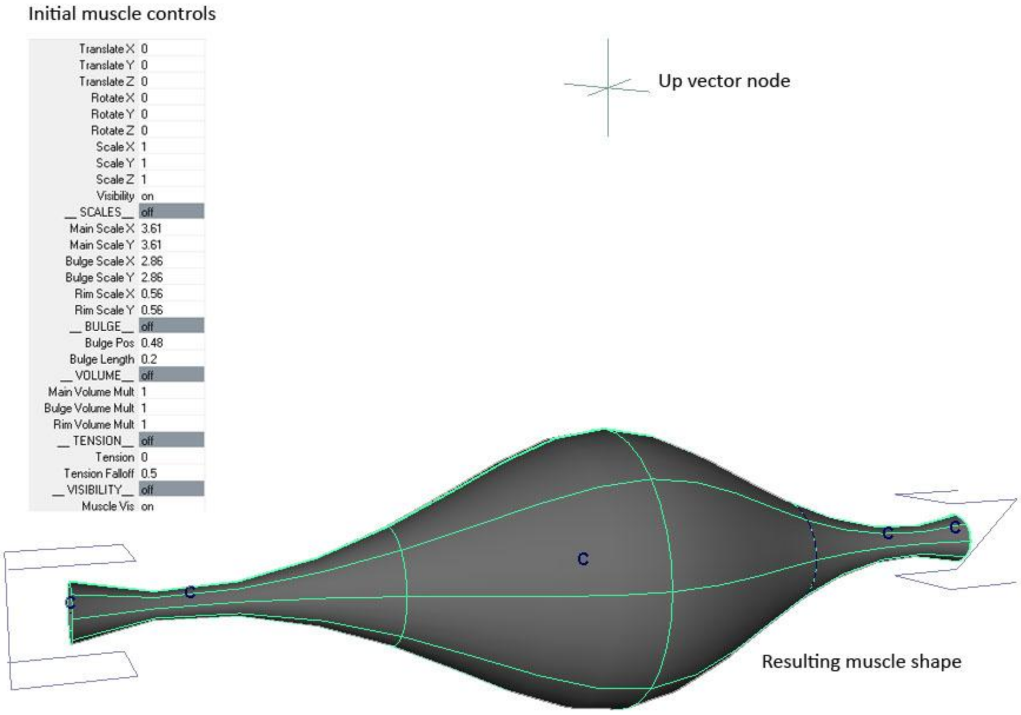
Once the action curve is in place five NURBS circles are attached to it and aligned along the curves tangent. In order to control the muscle bulge and shape several control variables are introduced. The *bulge position* controls together with the *bulge scale* attribute the position of the three middle cross sections. The position along the curve is maintained by using a *pointOnCurveInfo* node. This node generates the position of any point on a given parameter along a curve. The first and last cross section is aligned with the main control objects and scaled with a custom attribute called *rim scale*. The middle cross section is controlled by the *main scale* attribute and the cross sections that form the bulge of the muscle by the *bulge scale* attribute. On top of that control works a mechanism to maintain the volume of the muscle. The muscle maintains its overall volume when its cross sections maintain their volume. To efficiently approximate the volume, the arc length of the curve is constantly checked with a *curveInfo* node. The resulting length is divided by the initial arc length and then used as a scaling factor for each cross section. (11) (12)

$$S = \frac{Sc}{\frac{Li}{Lc}}$$

Where S is the final scale factor, Sc is the control attributes scale value Li is the initial arc length and Lc is the current arc length.

The orientation of the cross sections is accomplished with a tangent constraint. This type of constraint orients the curves according to the curve tangent at any given point. To control the orientation on a higher level an up vector node is introduced by generating a vector from the middle controller to a null object anywhere in the scene. This information is then transferred into world space and fed into the tangent constraints. All the transformations are done on null objects on a higher hierarchical level then the cross sections to be able to offset them at any later stage manually.

Now all the controls are in place and the cross section curves are lofted with a simple Maya loft node to one resulting NURBS object. Due to construction history the muscle object is updated interactively.



Translate X	0
Translate Y	0
Translate Z	0
Rotate X	0
Rotate Y	0
Rotate Z	0
Scale X	1
Scale Y	1
Scale Z	1
Visibility	on
__SCALES__	off
Main Scale X	3.61
Main Scale Y	3.61
Bulge Scale X	2.86
Bulge Scale Y	2.86
Rim Scale X	0.56
Rim Scale Y	0.56
__BULGE__	off
Bulge Pos	0.48
Bulge Length	0.2
__VOLUME__	off
Main Volume Mult	1
Bulge Volume Mult	1
Rim Volume Mult	1
__TENSION__	off
Tension	0
Tension Falloff	0.5
__VISIBILITY__	off
Muscle Vis	on

Figure 13 the final muscle after creation with its controls

6.4 MUSCLE MODELING

Once the muscle is generated it has to be sculpted to the desired shape. This is done by translating the main control objects and using the bulge controls in the first place. For simple cases of fusiform muscles this leads already to very nice results. The up vector must be positioned at first place and then the other control objects can be adjusted accordingly to model the muscle.

In some cases the target shape is more complex and need extra tweaking. In those affairs the control vertices of the particular cross sections can be tweaked. With that in place there is virtually every shape possible. To be able to tweak whole cross sections at once the cross section curves can be moved and their internal shape can be changed by the animator.

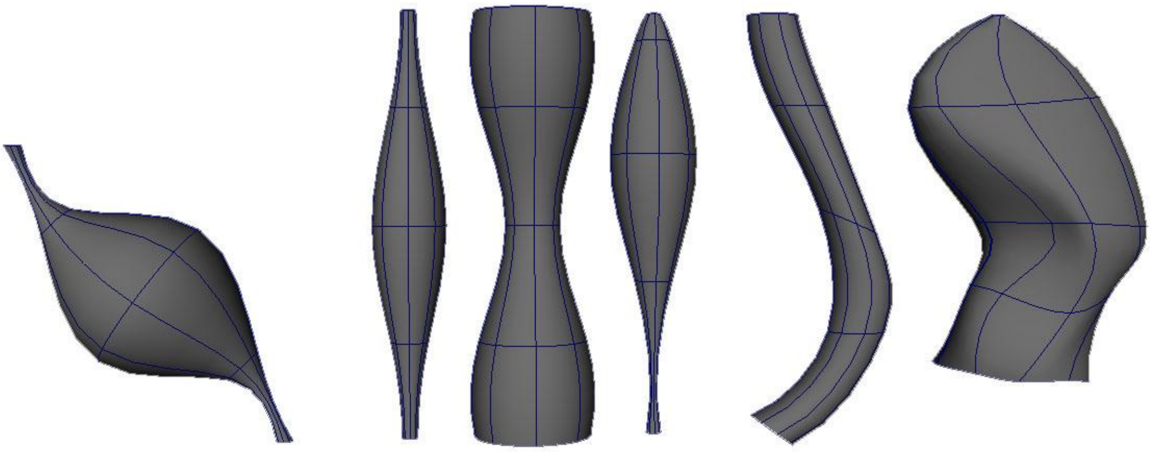


Figure 14 different muscle shapes

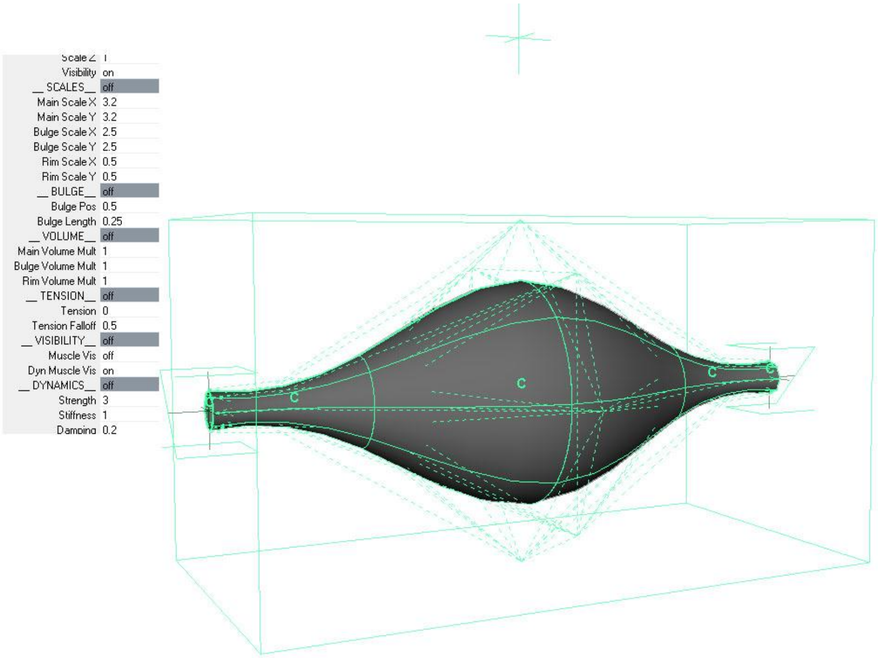
6.5 SECONDARY MOTION

After the muscle is brought in to its desired shape and attached via parent constraints to the bones chosen, additional realism can be brought into the system by secondary motion. This motion simulates the muscle and soft tissue jiggle of flesh and fat underneath the skin. This motion is archived by using Maya`s internal mechanisms for soft body simulation. The muscle shape is duplicated and then particles are instanced to every control vertex of the shape. By doing so, forces can be attached to the particles and drive the muscle shape.

To keep the muscle in an overall shape and pretend unrealistic motion two tools are used.

The first one is a so called goal object. The original NURBS muscle shape is used as one. The particles try to stick to the goal object as close as possible considering inertia and forces applied to them. The goal weight can be controlled for realistic results. Also the goal smoothness can be controlled. This attribute describes the amount of time that passes by for the particles to react. Lower goal smoothness makes the particles stick to the goal very drastically. Higher goal smoothness is used to make the particles move more loose on top of the goal. This rather abstract attribute is converted into a more user friendly *jiggle strength* attribute. This attribute is connected to the goal smoothness and provides the animator with the control over the amount of jiggling happening in the simulation.

The second mechanism to control the particles is *springs*. The springs are connected from every particle to its two nearest neighbors. These springs have damping and friction values to control their characteristics and help the system to stick together. This results in very flesh like motion. When one end of the muscle moves the motion is driven outward in waves. Springs are very fast to compute inside Maya and help increasing the realism a lot.



Scale Z	1
Visibility	on
__ SCALES__	off
Main Scale X	3.2
Main Scale Y	3.2
Bulge Scale X	2.5
Bulge Scale Y	2.5
Rim Scale X	0.5
Rim Scale Y	0.5
__ BULGE__	off
Bulge Pos	0.5
Bulge Length	0.25
__ VOLUME__	off
Main Volume Mult	1
Bulge Volume Mult	1
Rim Volume Mult	1
__ TENSION__	off
Tension	0
Tension Falloff	0.5
__ VISIBILITY__	off
Muscle Vis	off
Dyn Muscle Vis	on
__ DYNAMICS__	off
Strength	3
Stiffness	1
Damping	0.2

Figure 15 the final muscle with dynamics

6.6 SKINNING

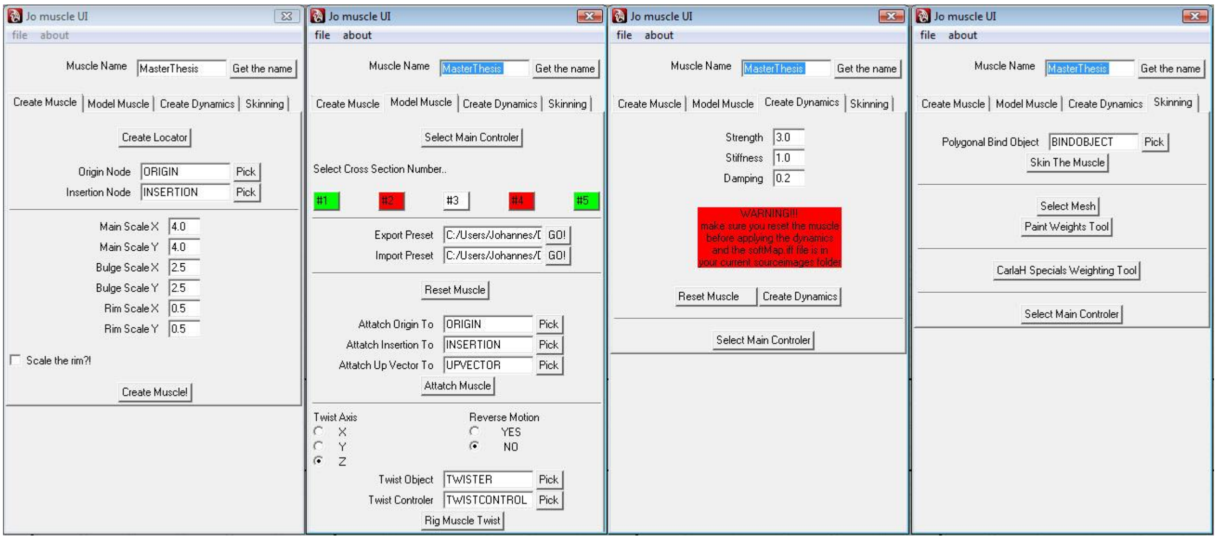
The connection between the muscles and the skin has one precondition. The skin has to be smooth bound to a Maya skeleton before the muscles can be brought in. The muscles are then used as influence objects to the skin cluster. By doing so, every production situation can get the most out of the system. Sometimes just one muscle is needed to introduce extra realism for certain shots or poses. The system is highly flexible and relies only on Maya`s internal tools. This enables artists to use the system without training because they should know the tools already.

After the muscles are linked with the skin cluster, custom weight maps must be painted to restrict the muscles influence on the skin to the area of interest. Muscles work as small details or skin a whole character. At any point the user can step in and increase the authenticity of the character by adding more details.

This method is very simple and fast to compute. More advanced techniques consider shrink wrapping or whole dynamic simulations for skin sliding on top of the muscles. These techniques are very heavy in computation and can be considered on a big production. Simulations of a scale like that are hard to handle and require many resources. Simple smooth skinning is enough for the most cases and therefore chosen for the system.

6.7 USER INTERACTION

To keep all the tools centralized in one spot a custom user interface was written to guide the user through the whole process. The UI is separated in the four parts explained above. At first the muscle name is put into the system. The name can be picked at any time by selecting the muscle and pressing the “get name” button.



Muscle Name: MasterThesis

Origin Node: ORIGIN
Insertion Node: INSERTION
Main Scale X: 4.0
Main Scale Y: 4.0
Bulge Scale X: 2.5
Bulge Scale Y: 2.5
Rim Scale X: 0.5
Rim Scale Y: 0.5
□ Scale the rim?!

Select Cross Section Number..
Export Preset: C:/Users/Johannes/D
Import Preset: C:/Users/Johannes/D
Attatch Origin To: ORIGIN
Attatch Insertion To: INSERTION
Attatch Up Vector To: UPVECTOR
Twist Axis: ○ X ○ Y ● Z
Reverse Motion: ○ YES ● NO
Twist Object: TWISTER
Twist Controler: TWISTCONTROL

Strength: 3.0
Stiffness: 1.0
Damping: 0.2

WARNING!!!
make sure you reset the muscle
before applying the dynamics
and the softMap.iff file is in
your current sourceimages folder

Polygonal Bind Object: BINDOBJECT

Figure 16 the interface

The first tab consist all the control for creating the muscle. An origin and insertion point is chosen by selecting two transform nodes and picking them with the UI. After that the muscle`s initial parameters are adjusted and the muscle is created.

In the next tab the initial shape of the muscle is modeled. The main control object contains all the parameters needed and can be selected with one click. Different cross sections can be selected and then shaped with Maya`s internal tools, already known by the user. Presets can be saved and reloaded. This enables more experienced users to save out parameters they know will work in certain situations. That can be a huge time saver on production. After the muscle has its desired shape it is rigged to the skeleton. Any transform node can be brought in the system by picking it and then attaching the controls automatically to it. The optional muscle twist can be rigged without any effort from the user. After the user picks the control objects the rig is brought into place on the fly.

Tab number three deals with secondary motion. The parameters for the dynamic simulation can be initiated and the soft body object is created. A warning text assures that the user resets the muscle before adding dynamics to it. This is a necessity for the following steps, otherwise results could be unpredictable. Again the main controller can be selected within the UI and the recently added dynamic attributes can be changed here.

At the end of the process is the skinning tab. After an already skinned poly object is picked the muscle can be added to the skin cluster. The user can now choose to paint the weights for the new object or use the well known tool from the Maya community named “CarlaH Component editor slider”. This tool helps to adjust weights on a per point basis. It was written by Carla Heinze and is freely available on www.highend3d.com.

With that set of tools the main functions can be accessed easily and tweaked to a certain extend. Due to the fact that the system consists only of internal Maya nodes, advanced users can apply their own techniques to it between every step.

7. RESULTS

With the system in place now several different results can be presented. The muscle system can be used to simulate muscle movement underneath skin to increase realism. Any fat tissue can be simulated with the same approach to get the subtle motion in areas of loose skin on characters. Following examples describe the result that can be archived with the system.

7.1 A SIMPLE EXAMPLE

In the first example a simple polygon cylinder is skinned to two bones and one muscle to demonstrate the basic of the system.

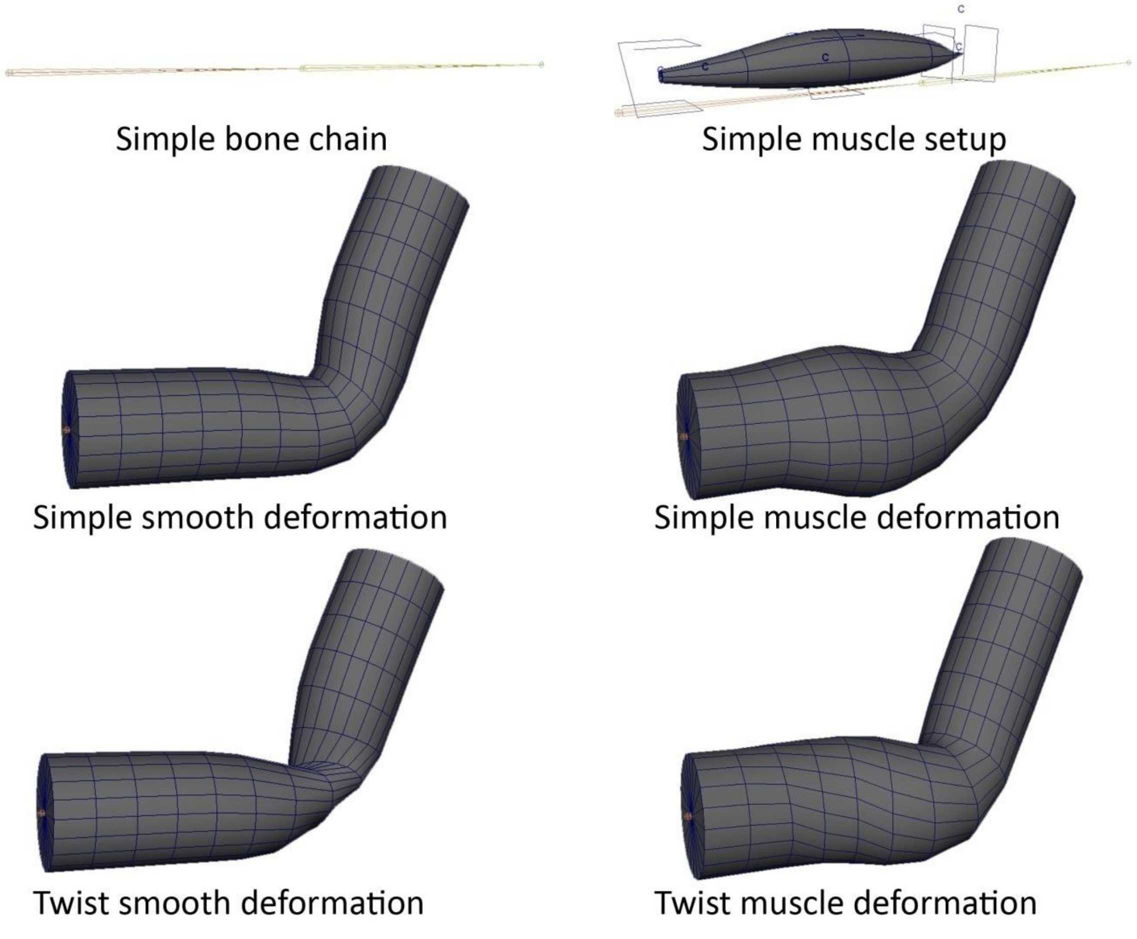


Figure 17 the simple setup

The muscle creates a bulge around the middle of the first bone. The deformations occur smooth and natural. With the muscle rigged for twisting deformation, the most difficult deformation a rotation and a twist in the bones leads to soft blending between the two bones with a natural bulge. On fast motion the jiggle inside the muscle simulates certain “fleshiness”. This is a simple example in which just one muscle leads to more realistic twisting and bending and the volume is preserved at any time.

7.2 THE CAVE TROLL

The cave troll is a final model for a three meter tall creature. The model was skinned in a normal smooth bind and then analyzed. The most important muscle groups where identified and muscles where placed to increase the realism in deformations. The troll was modeled by Ritchie Moore.

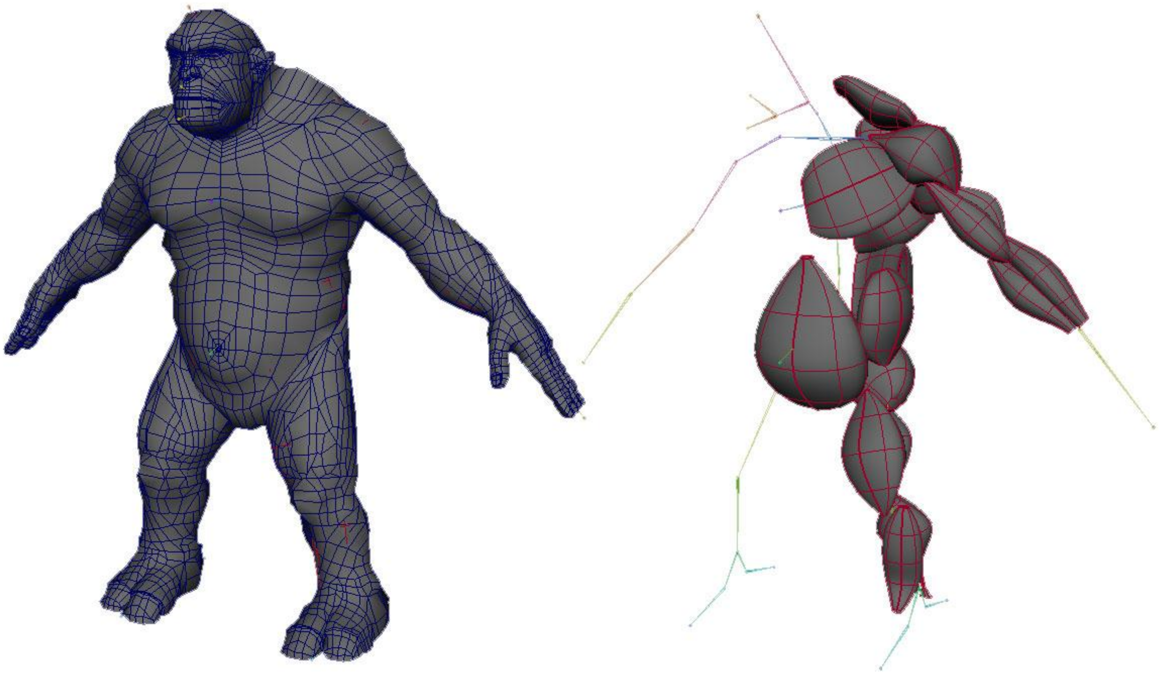


Figure 18 the troll

For comparison purposes was the troll rigged only on one side with muscles. The other side remained completely smooth bound to see the difference.

As an example for the different deformations that can be archived with the system some muscle groups were chosen to show the possibilities.

7.2.1 THE TROLLS ARM

The muscles on the trolls arm where packaged two only to five muscles that drive the skin in a realistic way. The forearm consists of two muscles that represent the ulna and radius muscles that give the forearm its characteristic twist around the bones. Traditionally this is very hard to archive. Normally additional roll bones are set into the joint hierarchy to simulate that twist. The system is capable of handling this complex motion by twisting with the forearm bone whole being attached to the elbow joint on the origin side.

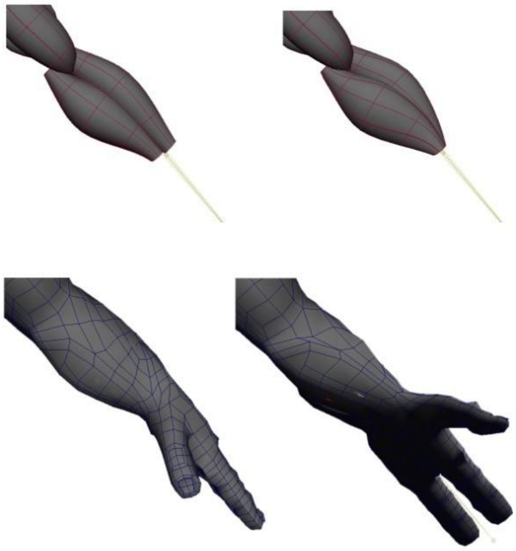


Figure 19 the forearm

The upper arm is also rigged with two muscles. They represent the biceps and triceps muscles. The muscles move in the direction of the shoulder when they contract and maintain the volume by bulging away from the bone. To keep the elbow in a realistic motion a rigging trick is performed on the triceps. The tangent controller is called with a set driven key. When the forearm bends forward the tangent is straightened to simulate the muscle sliding over the bone and leaving space for the elbow.

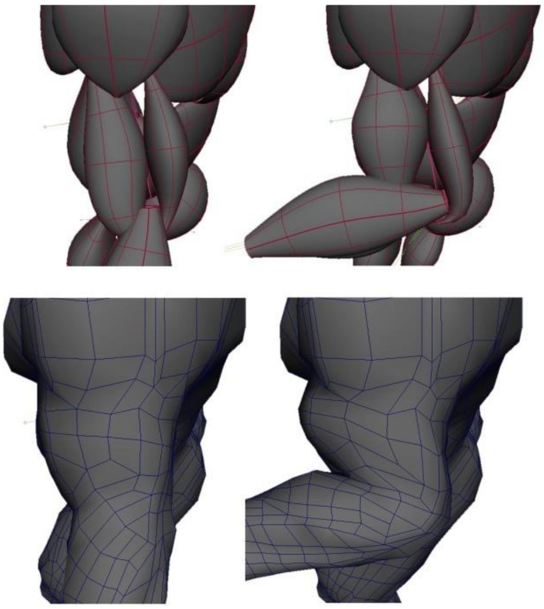


Figure 20 the biceps and triceps

7.2.2 THE TROLLS BELLY

The troll`s abdomen area is a good example for the fat tissue simulation capabilities of the system. The troll has a rather fat belly. The muscle underneath the huge fat and connective tissue layer don't really matter that much. In order to simulate those layers a muscle was chosen that suits that problem much better. One single muscle was shaped to match the belly area. Also the loose skin on the side of the stomach, known as "love handles", was represented by a single muscle. This muscle has very high jiggle strength values. That leads to a rather loose and fat like behavior. The muscle was skinned with a big falloff to the rim areas to blend the slower motions together with the stiffer pectoralis and quadriceps area.

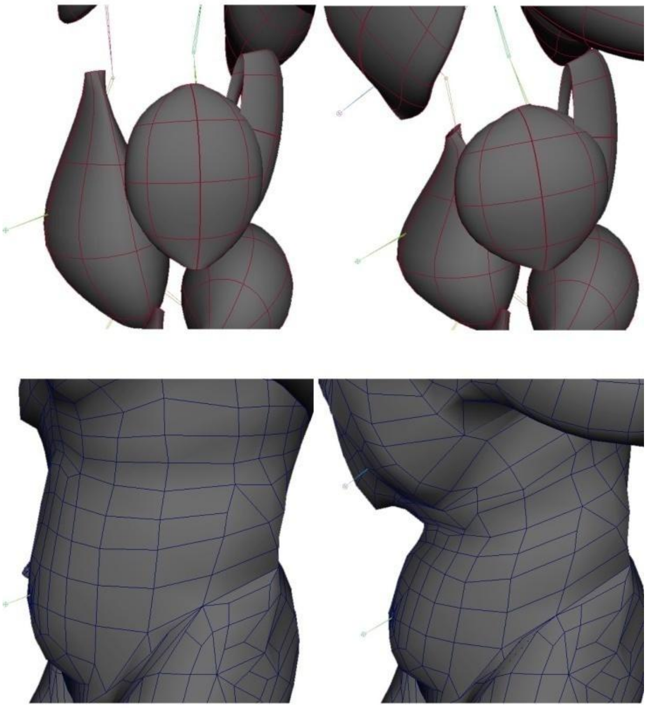


Figure 21 the belly

The results on this fake of fat and tissue simulation where rather interesting. With very little effort, only two muscles, that area could be simulated very visually pleasing. This shows the capability of the system in various cases on creatures in a modern character pipeline.

8. CONCLUSIONS AND FUTURE WORK

The emphasis with this system lies on simplicity and user friendliness. Muscle systems in current productions are very complex tool operated by several departments to archive high end effects. The average user has no chance to apply the complex simulations on smaller productions and projects. This system is easy to use and very intuitive. It can be added on top of already existing rigs to increase the believability. Even one muscle to simulate the abdominal area of a rather fat character can increase the realism by a very high percentage. Animators can apply dynamics in their every day environment and need no further training on the toolset. More advanced users can dig deeper into the system and profit from it. The system relies entirely on internal Maya nodes. Special setups can be placed on top of the standard mechanisms to complete tasks that don't ship with the standard package. The troll`s elbow setup is an example for that. With the use of a simple set driven key the muscle seems to bend around the bone. The rigging for the cave troll took only two days based on a skinned character. Deformations where enhanced a lot and weights painting becomes more intuitive when weights are painted around the muscles. The muscles give a visual feedback in real time.

In modern games animation is produced in real time to provide the player with multiple animations based on his actions. A fast and easy system could be utilized here to deform the character in real time more believable and with more visual weight.

But with all its simplicity the system has limitations. Skin sliding can only be simulated by rigging fakes. These fakes work well on the usual areas like the elbow and the knee. But areas with big bones that slide underneath the skin like the side of the ribcage or the hip, the system can only approximate the motion. A more elaborate system should simulate the connective tissue with mass spring systems and different integration methods to get closer to the biological roots of the problem. Cloth like simulations that make the skin slide over the bones and muscles are very expensive but with increasing computing power simulations could get more affordable even for smaller productions.

Muscles can be modeled in almost any shape by manually tweaking the cross sections. Automatic muscle generation based on the skin surface would speed up the set up process and depending on the accuracy of the automatic detection increase the realism. In general time-investment pays off in the skinning process. The more time is invested to generate complex muscles the closer the system will get to the reality.

9. APPENDIX

9.1 SPECIAL MEL CODE EXAMPLES

The following code explains the pipeline trick to source scripts in a folder on the same hierarchical level as the current script. This method is very helpful for any organization of MEL scripts inside Maya.

```
[…]
// big cheat to be sure that the mel files in the muscle subfolder get sourced
// no matter where the files are on disk

// get the path to the proc by doing a whatIs
string $cheat = `whatIs "joMuscle"`;
// and substituting the unnecessary stuff with nothing
$cheat = `substitute "Mel procedure found in: " $cheat ""`;
$cheat = `substitute "joMuscle.mel" $cheat ""`;
// and appending my path to it
$cheat = $cheat + "joMuscleScripts/";
// get the *.mel files in the folder
string $myScripts[] = `getFileList -folder $cheat -filespec "*.mel"`;
//step through the files and source them all one by one
for( $script in $myScripts )
{
        // using a eval command with a concatenated string in order to bypass
        // a known Maya bug with the source command
        string $sourcer = "source \"" + $cheat + $script + "\"";
        print $sourcer;
        print "\n";
        eval $sourcer;
}
[…]
```

Another bit of code that is worth noticing is a bit of the skinning script. In order to allow the user to switch between the original muscle and the dynamic muscle an extra blend shape is created. This is necessary to give the user the opportunity to either see the results of the skinning in real time, or simulate the dynamics on one model. In order to be able to switch back on forth the blend shape is created between the final muscle model and the dynamic muscle. The resulting shape is then used for skinning. By controlling the blend shape the input can be chosen by the user.

```
[…]
// get all the shape nodes for the blend shape creation
string $binderShape[] = `listRelatives -s $binder[0]`;
string $muscleShape[] = `listRelatives -s $muscle`;
string $dynMuscleShape[] = `listRelatives -s $dynMuscle`;

// create the blend shape front of chain
string $blendShape[] = `blendShape -frontOfChain -n ( $muscleControl + "_blendShape")
$muscleShape[0] $dynMuscleShape[0] $binderShape[0]`;

// add the control attributes to the muscle main controller
addAttr -ln __BLENDER__ -at bool $muscleControl;
setAttr -e -keyable true ( $muscleControl + ".__BLENDER__" );
setAttr -lock true ( $muscleControl + ".__BLENDER__" );

addAttr -ln resultVis -at bool -dv 1 $muscleControl;
setAttr -e -keyable true ( $muscleControl + ".resultVis" );
connectAttr ( $muscleControl + ".resultVis" ) ( $binder[0] + ".visibility" );

addAttr -ln static_dyn_blend -at double -min 0.0 -max 1.0 -dv 0.0 $muscleControl;
setAttr -e -keyable true ( $muscleControl + ".static_dyn_blend" );
connectAttr ( $muscleControl + ".static_dyn_blend" ) ( $blendShape[0] + "." +
$dynMuscleShape[0] );

// create a reverse node to control the blend shape from both sides
$reverser = `createNode reverse`;

// connect the control attributes to the blend shape node
connectAttr ( $muscleControl + ".static_dyn_blend" ) ( $reverser + ".inputX" );
connectAttr ( $reverser + ".outputX" ) ( $blendShape[0] + "." + $muscleShape[0] );
[…]
```

BIBLIOGRAPHY

1. **Horst Bayrhuber, Ulrich Kull, Hermann Linder.** *Linder Biologie Gesamtband 22. Auflage.* s.l. : Schroeder, 1998.

2. **wikipedia.** wikipedia bone. [Online] 2007. [Cited: september 3, 2007.] http://en.wikipedia.org/wiki/Bone#Alternatives_to_bony_endoskeletons.

3. —. wikipedia muscle. [Online] 2007. [Cited: september 6, 2007.] http://en.wikipedia.org/wiki/Muscle.

4. —. wikipedia connective. [Online] 2007. [Cited: september 1, 2007.] http://en.wikipedia.org/wiki/Connective_tissue.

5. —. wikipedia skin. [Online] 2007. [Cited: september 4, 2007.] http://en.wikipedia.org/wiki/Skin.

6. *Advanced Character Pipeline and Tools.* **Eric Miller, Paul Thuriot.** s.l. : Siggraph, 2003.

7. **Gordon, Loise.** *Zeichnen lernen - Portrait und Anatomie des Menschen.* Muenchen : Augustus, 2000.

8. **Autodesk.** Maya Online Help. [Online] 1998-2007.

9. **Gould, David.** *Complete Maya Programming. An Extensive Guide to MEL and C++ API.* s.l. : Morgan Kaufmann Series in Computer Graphics, 2002.

10. **Autodesk.** *Autodesk Maya API white paper.* [PDF file] 2007.

11. **Kiaran Richie, Oleg Alexander, Karim Biri.** *The Art of Rigging Volume 1.* s.l. : CG Toolkit, 2005.

12. —. *The Art of Rigging Volume 2.* s.l. : CG Toolkit, 2006.

13. **Johannes Saam, Jon Macey.** *Production Friendly Skinning.* s.l. : NCCA, 2007.

14. *Automatic muscle generation for character deformation.* **Xiaosan Yang, Jian J. Zhang.** s.l. : whirley inter science, 2006.

15. *Realistic Skeleton Driven Skin Deformation.* **Xiaosan Yang, Jian J. Zhang.** s.l. : NCCA, 2006.

16. *Curve Skeleton Skinning For Human And Chreature Characters.* **Xiaosan Yang, Jian J. Zhang.** s.l. : Whiley inter science, 2006.

17. *Interactive Constraction And Animation of Kayered Elastically Deformable Characters.* **Russell Turner, Enrico Gobbetti.** Oxford : The Eurographics Assotsiation, 1998.

18. *Anatomy-Based Modeling of the Human Musculature.* **Fredi Scheepers, Richard Parent, Wayne Carlson, Steven May.** s.l. : Siggraph, 1997.

19. *Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-Driven Deformation.* **J.P. Lewis, Matt Cordner, Nickson Fong.** s.l. : Siggraph, 2000.

20. *Building efficient, accurate character skins from examples.* **Alex Mohr, Michael Gleicher.** s.l. : Siggraph, 2003.