

**A Documentation of the Modelling and Rigging
Techniques Used to Create a Virtual Toy
Transformer and the Possible Applications for
it as a Marketing Tool**



By Alison Farmer

March 2006

Contents Page	Page No.
Abstract	2
Introduction	2
Modelling the Virtual Toy	3
Rigging the Virtual Toy	11
-Setting up the Skeleton	12
-Setting up the FK Transformation Rig	15
-Setting up the Vehicle Rig	18
-The Problems with Creating the IK Rig	18
Possible Uses of the Virtual Toy as a Marketing Tool	24
Critical Analysis	29
Appendix A	31
Appendix B	38
References	40

Abstract

The objective that I set out to achieve for this project was to create an accurate replica of a toy Transformer robot using 3D computer modelling software and to build a working mechanical rig that would enable him to transform. This allowed me to explore the techniques used to model mechanical, robotic type characters and to understand and gain knowledge about the most apt methods for rigging them. On top of this, I decided to explore the possible uses and applications of my virtual toy model as a marketing tool for the toy company. My objectives provided me with a project that was to be both challenging and innovative. The practical side of this report is targeted at an audience who understand the concepts and terminology of 3D computer modelling and rigging in Alias Maya.

Introduction

This report documents the techniques and solutions that I have applied to the practical aspects of the project and is also a discussion about the theoretical applications of 3D computer visualization as a marketing strategy, in the context of my virtual toy. I aim to bring to light some of the most useful methods and approaches to mechanical modelling and also to tackle the many issues that can occur with mechanical rigging as well as the solutions that helped to overcome them.

I feel that there is an abundance of information from books and on the internet about the different approaches to organic modelling and the best methods for setting up human and creature rigs, but a lot less is written about how these practices are applied to a mechanical type character. I had recently, however, acquired two books in a series released by the CGToolkit company, namely 'The Art of Rigging: Volume 1' and 'The Art of Rigging: Volume 2' by Kiaran Ritchie. The 2nd volume goes into depth about how to set up a robotic type character for animation and discusses the implementation of some custom MEL scripts to speed up and improve upon the rigging process. I will be describing how I employed the techniques that I learnt from these books, to the virtual Transformer model and also how I applied my own knowledge and understanding to solve some of the more complex rigging issues unique to this mechanical character.

During working on the practical aspects of this project and building the virtual toy, I was inspired to carry out research into the possible ways in which my virtual toy could be used as a marketing tool for the actual toy Transformer product. I will be reviewing some of the applications that I considered for the virtual toy and will also be documenting my findings from researching into the ways in which toy companies can use computer animation as an effective means of marketing their products.

Modelling the Virtual Toy

One of the main challenges for me with this project was to be able to produce an accurate replica of the toy Transformer in 3D. I thought this would be a useful technique to practice as it translates to the way the industry pipeline works; a modeller receives the designs of a character or environment structure and has to follow these designs to a high degree of accuracy in order to satisfy the client. This is innovative and a new experience for me as I have only ever modelled characters that I have designed myself. This means that in the modelling stage you can still be quite free and loose about how you want your character to turn out. I find that when you are modelling your own character design, the design stage can cross over into the modelling stage as you begin to see your character emerge as a three dimensional object. You can then ‘sculpt’ at will any changes that you feel may improve upon your original design.

With this particular toy character the accuracy of the modelling was even more important due to the fact that the toy transforms from a standing robot into a futuristic jet in its vehicle mode (see **Figure 1**). This meant that each part of the virtual model had to be an accurately scaled 3D version of the parts that make up the toy, so that they all fold up and connect together realistically to form the jet vehicle.

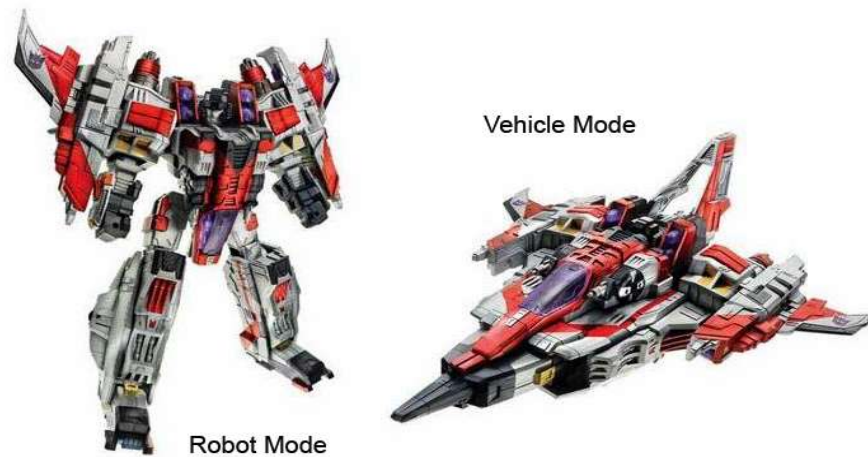


Figure 1. This shows the toy Transformer, 'Starscream', that I will be recreating using Maya as the 3D modelling software. It shows the toy before transformation in robot mode and after transformation in vehicle mode. (Image taken from www.toysnjoys.com)

To be able to make this possible, it was necessary for me to purchase the actual toy allowing me to take measurements directly off of it. Taking measurements for my model from 2D images (for example drawings and photos from websites) would not have allowed me to achieve the accuracy that I had aimed for. Having the toy in front of me made it a lot easier for me to obtain an understanding of how the toy worked and what parts folded up to achieve the transformation.

It is interesting to think whether if I had been modelling from blue prints of the Transformer, would I have found it harder to create my virtual model? Although I would have had accurate 2D drawings to work from I think it would have been a lot harder to build the 3D model as there is only so much you can interpret from a 2D image. This is an issue that I have become aware of through my own experience of 3D computer modelling and seeing the way fellow students work. I find that it is common practice to scan in a front and side view of your character into Maya and build your 3D model by lining it up with the two views. This often means you can become so fixated on how well your model lines up with these 2D image planes that you can overlook the model as a whole in its 3D form and restrict your creative mind from seeing what looks right and wrong.

I have found with recent work and through undertaking this project that having a 3D clay sculpture or in this case an actual toy figurine, the modelling process is made less complex and the final result that you

achieve is often more successful and aesthetically pleasing. In the case of this project, I think it has enabled me to produce a 3D representation of the toy to a higher degree of accuracy than if I was to use blue prints or accurate 2D images to work from. However, it is not always the case that, as a modeller, you will have a maquette to work from. In industry, for example a client may not provide this kind of material. It is then very important that that the model sheets you have to work from are clear and concise.

“A good model sheet includes very tight and accurate drawings that are used by the modellers as reference planes inside the 3D modelling environment.” Ritchie K, ‘The Art of Rigging: Volume 2’, 2005

We can see from the character design in **Figure 2** of the SP-23 robot that building a maquette for the modeller to work from would be impractical due to the complexity of the character.

This project also offered me the challenge of modelling a non-organic character. This was a new learning experience as I had only ever modelled organic, creature-like characters in my previous work. The word non-organic used in this context describes the model as being a non living or industrial object, for example a car or a chair and the word organic refers to a natural or living being, such as a plant or a person. I have found that there are big differences in the ways in which you approach modelling a mechanical object compared to an organic one. I have also learnt about particular modelling tool options in Maya that have assisted me in reproducing an accurate model of the Transformer.

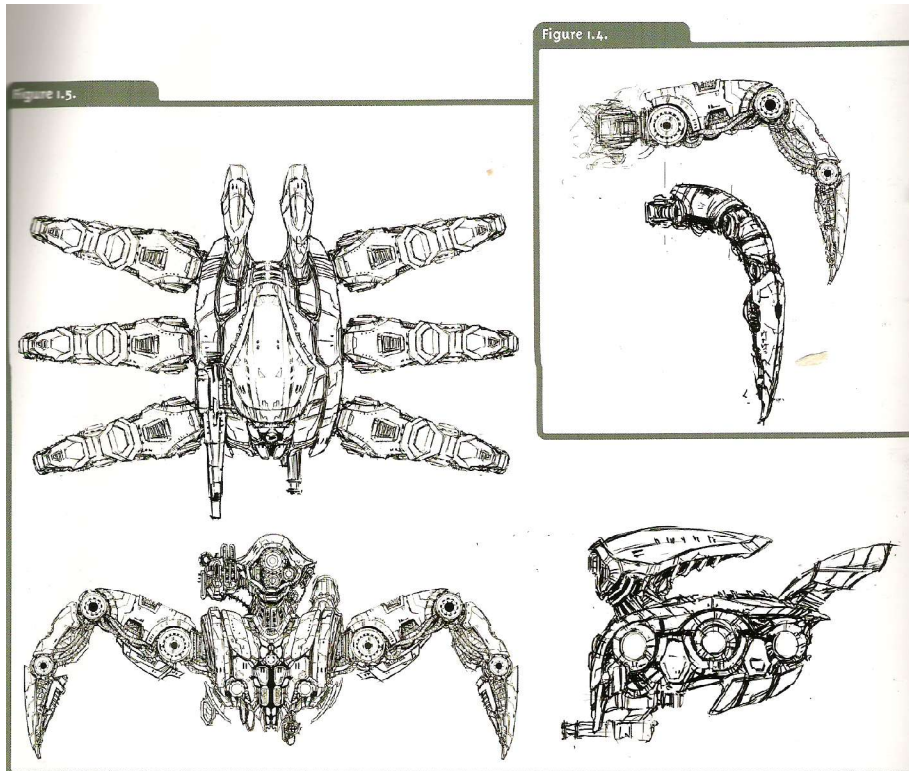


Figure 2. Model sheet for SP-23 robot, featured in ‘The Art of Rigging: Volume 2’. This is comprised of a top, front and side view, as well as some close ups of the complex robotic leg. This gives the modellers enough information to accurately reproduce the form of the robot in 3D modelling software.

I discovered the best way to model each part of the transformer was to start with a polygon cube in Maya and scale it to match the exact measurements of the particular piece, then work the detail into it. This technique is known as box-modelling. This is quite different to how I would tackle the first stages of modelling an organic character. If you were to use a cube as the basis of your mesh for an organic model, it is often the case that even after adding significant detail the character tends to retain a box like shape to it which is very unnatural. Topology is also an issue with organic modelling. It is important that the edges of the model flow in the right direction so that it is possible to achieve realistic deformations when it comes to the rigging and skinning of the model. So generally, starting with a predefined shape to model an organic character is a bad idea due to the fact that it will already come with an existing topology. This is not an issue with mechanical modelling, as even though the model will be rigged, the separate parts will not deform in any way. This is because the model will not be bound as skin to the rig. Instead, each separate piece is parented to the

corresponding joint to allow for animation. This will be discussed further in the Rigging chapter.

To be able to make the model an accurately scaled reproduction of the toy in Maya, it was necessary for me to constantly check the lengths of each part throughout the modelling process, using Maya's Distance Tool. This creates two locators and displays the distance between them to the user. To measure the lengths of the edges of each piece to a high degree of accuracy, I would snap one locator to the vertex at the start of the edge I wished to measure and the other locator to the vertex at the end of that edge (see **Figure 3**).

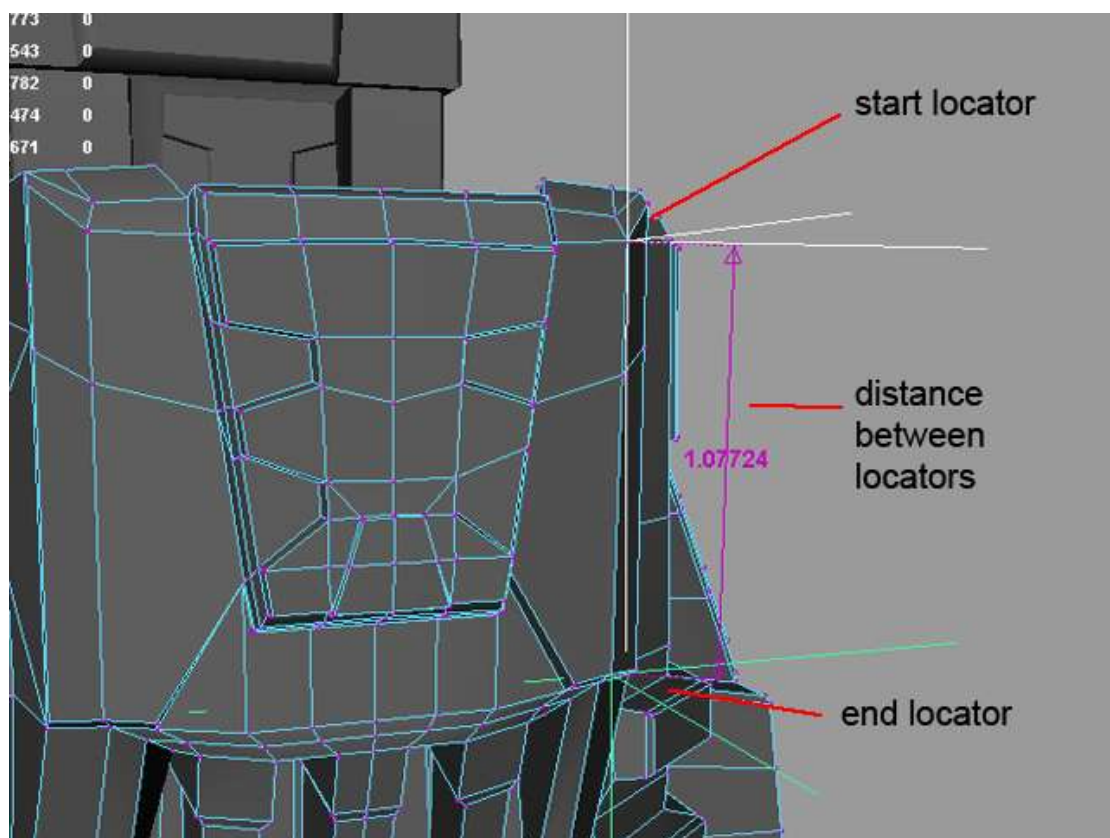


Figure 3. Distance Tool - This is used accurately by snapping the start locator to the vertex at the point you wish to measure from, then snapping the end locator to the vertex at the point you wish to measure to. Maya will display the length in units between these two locators.

To work in detail to the model it is necessary to split your geometry to increase the number of edges and faces on the polygons surface. This enables you to make extrusions at the appropriate place on a particular part of your model. To create accurate and even edges and faces on the parts of the toy model, I utilised the 'Snap To Magnets' button in Maya's Split Polygon Tool settings. Once this option is turned on, you

can control the 'Number Of Magnets' and the 'Magnet Tolerance' attributes (see **Figure 4**). By setting the magnet tolerance to 100% means that wherever you click on the edge that you wish to split, the split will happen at the closest magnet point. If the Number of Magnets is 1 (as in **Figure 4**) then there will be one magnet point along that edge and it will be at an equal distance from the start and end points of that particular edge, i.e. it will sit half way along it. Increasing the number of magnets will allow for splits to occur at increasing numbers of fractional divisions, e.g. setting the number of magnets to 2 will enable the edge to be split into thirds. By using magnets it is therefore possible to create even splits around an object at a range of different fractional divisions. I found this aided my ability to reproduce the detail in each of the toy parts with a high level of accuracy.

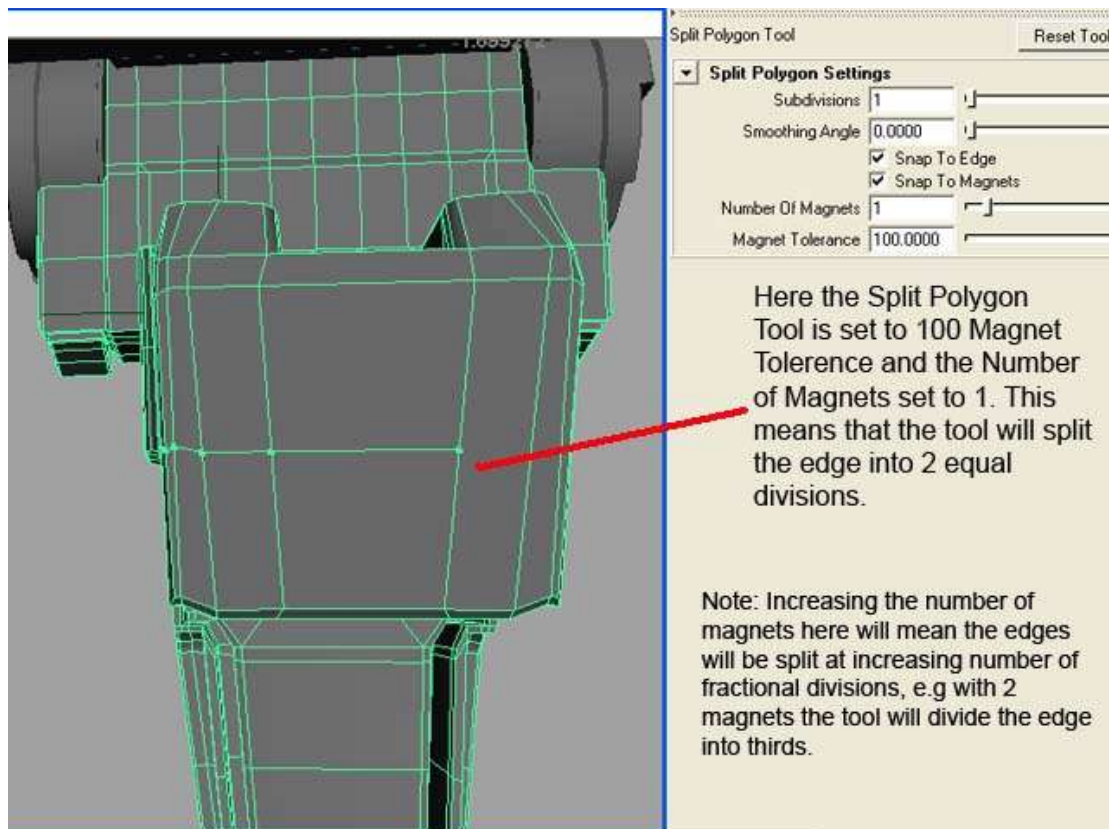


Figure 4. Split Polygon Tool – Using this tool with the 'Snap To Magnets' option box checked on, allows you to control where this particular tool will split an edge. By adjusting the Number of Magnets and setting the Magnet Tolerance to 100%, enables you to split the edge at a range of fractional divisions.

I have found that when modelling hard-edged, industrial surfaces, it is important to consider the normals of the edges on the model. It is possible to achieve very different looks to a surface by softening or

hardening the normals (see **Figure 5**). By selecting the desired edges and making the normals ‘All Hard’ will make the edges appear sharp in shaded mode. By making the normals ‘All Soft’ will soften their appearance and effectively smooth the surface.

Throughout the modelling of this toy Transformer, I had not needed to consider the poly count of the final model as it was my aim to produce a highly detailed accurate reproduction of the toy. I was working to achieve the same level of detail that is held in the actual toy product. As I have progressed with this project, however, I have begun to think about what this 3D visualization of the toy could be used for. In researching into the toy industry and the ways toy companies market their products, I have realised some possible applications for my virtual model. I will be discussing this at a later stage in this report.

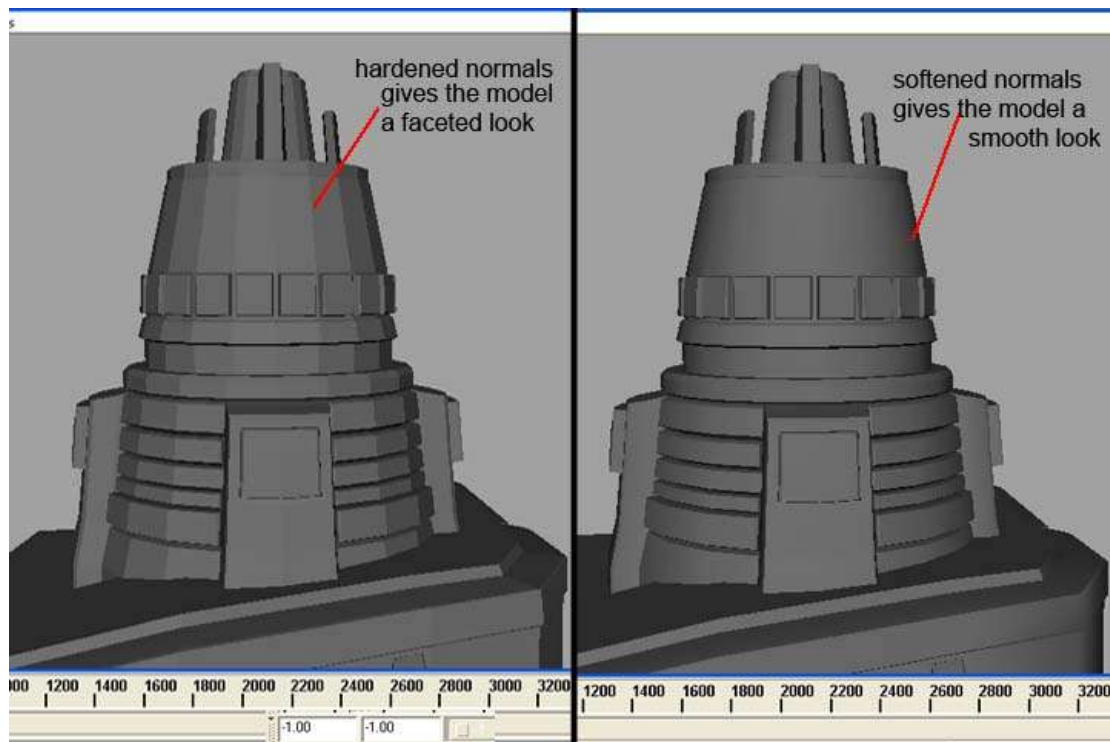


Figure 5. Soften/Harden Normals – It is possible to achieve distinctly different looks to a model by softening or hardening the normals. We can see from this image that making the normals hard gives the edge a sharp look and making the normals soft gives the model a much smoother appearance.

Most of the uses for this virtual toy that I have arrived at would require it to be of a much lower level of detail than what I have achieved for my model. An example of this would be if it were to be used on a website in a Flash animation or game. Although my model is too

detailed for such applications as these, it would be an easy process to reduce the model in terms of poly count and remove a lot of the models detail to make it suitable for such uses.

The final completed model of the Transformer can be seen in robot mode, in **Figure 6** and in vehicle mode, after transformation, in **Figure 7**.



Figure 6. This is the finished model of the Transformer in robot mode, showing the front and back views.

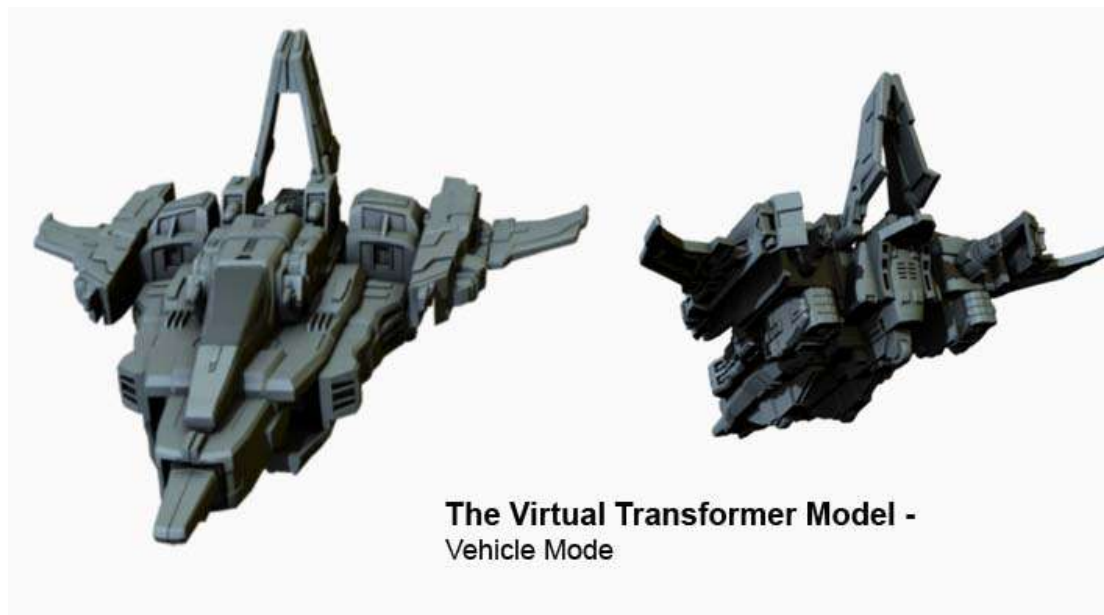


Figure 7. This is the finished model of the Transformer in Vehicle mode, showing the front and back views.

Rigging the Virtual Toy

Similar to the way I had only ever built organic models before this project, I had also only ever rigged organic characters, so setting up a mechanical model to be animated was an innovative challenge for me. I had recently acquired some very interesting and useful books that teach the theory and practices of rigging used by professional technical directors. The books are called, 'The Art Of Rigging: Volume 1' and 'The Art of Rigging: Volume 2', written by Kiaran Ritchie, Jake Callery and Karim Biri and released by CG Toolkit. These books were initially purchased to assist with my Major Project and while some topics covered will significantly help me with this project, there are also many new ideas covered that I would like the opportunity to experiment with. In 'The Art of Rigging: Volume 2', the process of setting up a rig for a complex, robotic character (the SP-23 robot as shown in **Figure 2**) is discussed in detail. Having worked through both books and tutorials covered, I have felt encouraged to apply the new skills and techniques that I have learnt onto a mechanical model of my own. This project offers me the opportunity to do this and to improve and further my understanding of the rigging process.

Not only was there the challenge of setting up a rig for a mechanical model but I also had to think about how it could be built to allow for



Figure 8. The Citroen Transforming Car - This TV commercial was produced by Spy Films in Toronto, with animation undertaken by a Vancouver based VFX company, The Embassy.

the virtual toy to transform. A very successful example of a transformable rig in action can be seen on the Citroen C4 television commercial (see **Figure 8**). Seeing this advert inspired me to want to research into methods of developing transforming rigs and hence this being one of the reasons why I have chosen to build a virtual toy Transformer. Using a Transformer toy was also ideal for my project as it meant I did not need to consider how the transformation would work (as the toy makers had designed

this already), thus allowing me to have more time to focus on building the complex rig.

The main issue I had to consider when rigging the Transformer, was that the rig needed to cope with three different states; robot mode, transformation mode and vehicle mode. I decided that the best way to do this would be to build three separate rigs, one for each mode. This was necessary because there were very different controls and setups needed for the different states. It was much more practical and user friendly to have these three separate rig setups instead of attempting to incorporate all the controls needed for the robot, the transformation and the vehicle onto one single skeleton.

When the Transformer is in robot mode the animator would find it much easier to animate with a rig setup with inverse kinematics (IK) controls. The animator can select these controls to move the legs and arms and set keys on them, for example to animate the Transformer walking. Unfortunately, due to the way in which the toy is built, it was difficult to accomplish a working IK rig and I will discuss the reasons for this at a later stage in this report. For the transformation state of the virtual toy, it was more suitable to have a rig setup with forward kinematics (FK). The FK rig is made more practical for the animator by the creation of sliders that can control how much each part of the

model has transformed. This allows the animator to set keys on these slider controls to animate the transformation. For the last state, the rig only needed to be able to handle the translation and rotation of the jet. The animator can select one main control which can be used to control the jets movement and which also holds attributes to control the rotary motions of the wing flaps and the guns. To switch between the different rigs as the virtual toy transforms, for example from robot mode to transformation mode, the transformation skeleton would need to be positioned to match the robot skeletons position and the visibility of the robot rig would be switched off at the frame where the transformation begins.

Setting up the Skeleton

It is very important when setting up the skeleton for a mechanical character that the joints are placed accurately.

“Rigging mechanical characters (or vehicles) requires some extra attention. You may be familiar with placing joints by hand. This works fine for organic characters, but it can present a problem for mechanical rigs.” Ritchie K, 2005. Pg 101.

The joints should be placed at the pivot points of each piece and if they are placed slightly off-centre it can cause the mechanical parts of the model to look wobbly and broken when animated (see **Figure 9**). To ensure that I was placing my joints at the exact centre of the Transformers mechanical joints, I used Maya’s snapping tool. Using this I was able to snap the joints to a vertex that was at the exact centre of rotation for that particular piece. For some parts of the model, however, it was not possible to do this, i.e. if there was no vertex lying at the exact centre of rotation. An example of where this occurred is in placing the joint for the knee.

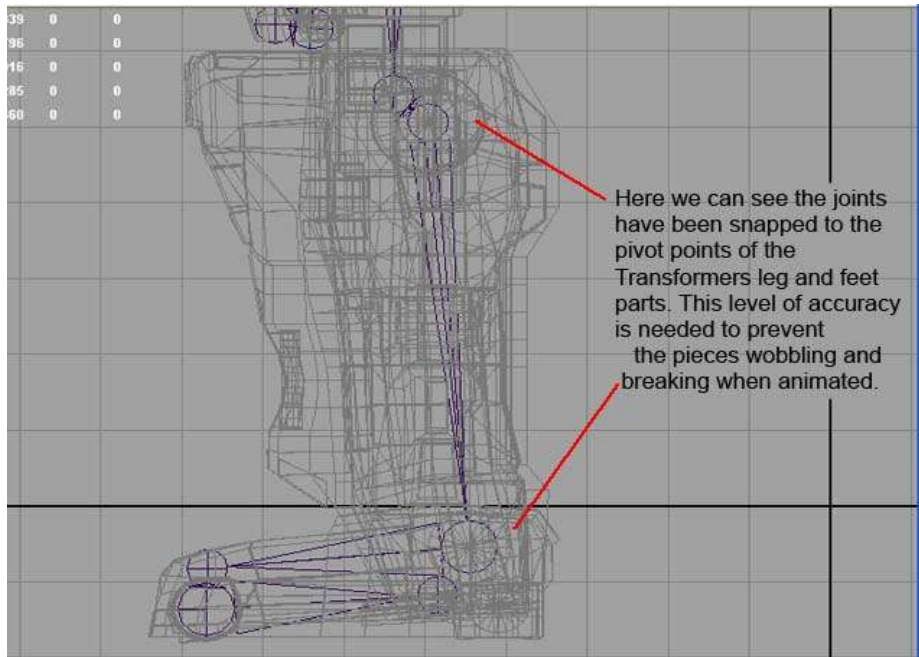


Figure 9. Accurately Placing Joints. This shows how I have snapped the joints to the pivot points of the separate parts of the Transformer toy to ensure precise placement and prevent the leg from looking broken and wobbly when animated.

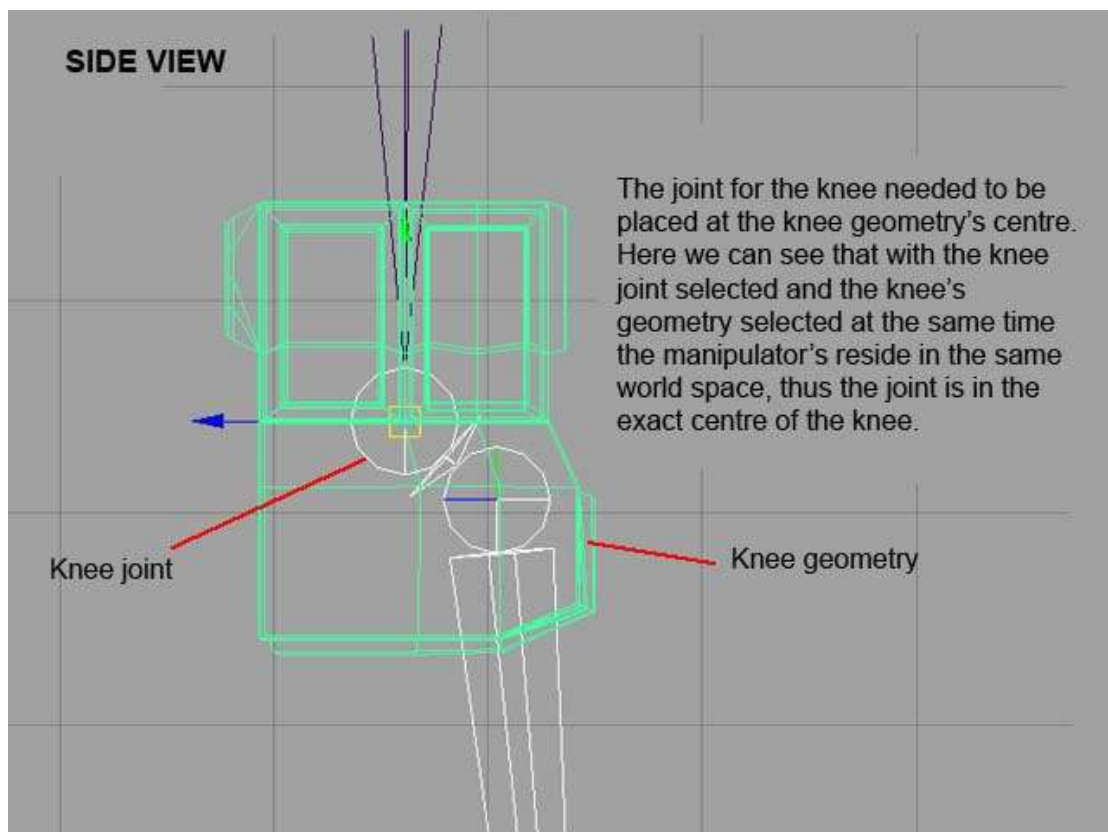


Figure 10. This shows how the 'cgTkSnapTo Pivot' script was used to place the joints accurately, in the correct pivot position.

For the knee, I needed to snap the joint to the centre of the knee's geometry. Firstly, I centred the pivot on the knee piece using the Maya command 'Centre Pivot'. I then needed to snap the joint to the knee's pivot and to do this I was able to use a script that was supplied with 'The Art of Rigging: Volume 2', called 'cgTKSnapToPivot' (see Appendix A.I). This is a fairly simple script that allows you to select a base object and a target object and on running the script it will query the world space location of the base objects pivot and move the target object to that position. We can see from **Figure 10**, that the knee joint has been snapped to the pivot of the knee's geometry as from selecting them both the manipulator's lay directly on top of each other.

After accurately placing the joints, I needed to check the orientations of their axes to make sure that they are lined up with the orientation of the pivots of each part of the model. This is just as important as placing the joints precisely, as the joint may be sitting at the correct pivot point but if its axes are not oriented correctly, it would mean that all three axes would need to be rotated to achieve the desired rotation for each part of the model, which is very difficult and impractical for an animator. If the axes are lined up correctly we can just select the desired axis to rotate around and the geometry will move realistically. In order to rig the Transformer so that it moves according to its mechanical structure, I needed to consider which axes are used and what motion is restricted at each joint.

"In many cases, mechanical joints are restricted to one free axis. This is the case with almost every moving part on your car (things like wheels. Wipers, doors, seat-backs, steering wheel etc...). When rigging these kinds of objects, it is important that you do not give the animator the option of twisting the piece in the wrong axis. Joints that have limited axes should have their unused axes locked and hidden from the animator." Ritchie K, 2005, pg 119.

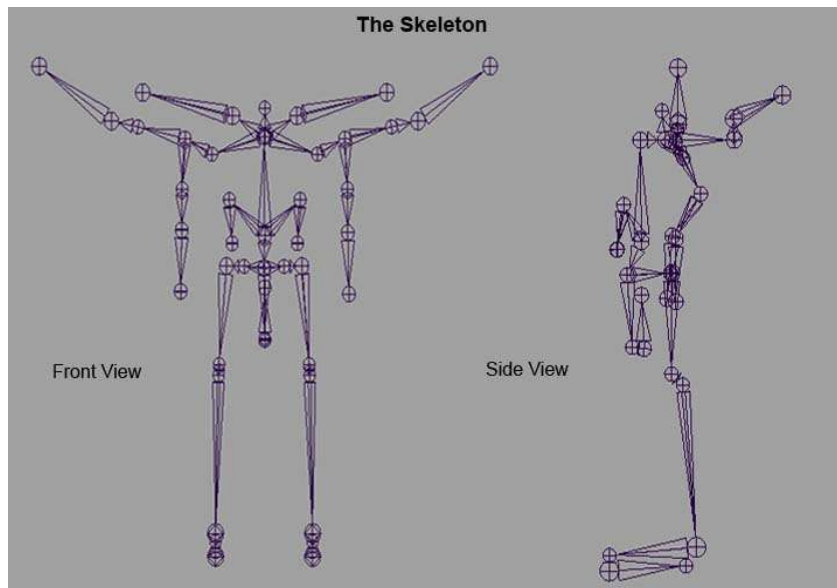


Figure 11. The Skeleton – This shows the bone layout for the final The Transformer rig. The skeleton joint layout can be seen in **Figure 11**.

Setting up the FK Transformation Rig

Once I had accurately placed the joints and correctly oriented them, I decided to setup the FK Transformation rig first. I initially needed to check that the virtual model could fold up like the toy to transform into the jet. I found this a very thrilling part of the project. It was exciting to rotate each part of the virtual toy Transformer and to then have an entirely different looking model that correctly resembled the jet that the real toy transforms to. It was quite a satisfying process and must resemble how a child would feel playing with the toy. This is most likely the main reason for this toys success and its main selling point. To setup the animation rig for this skeleton, to allow it to be animated in an FK fashion, I decided to apply some of the techniques which I had learnt from ‘The Art of Rigging: Volume 2’ tutorial DVD and utilise some of the mel scripts to make setting up the rig faster and more user friendly. Instead of creating curves to use as controllers, a way of using mechanical characters geometry directly as controller objects is explained. The idea is that the animator can click on the various body parts to manipulate the rig (see **Figure 12**).

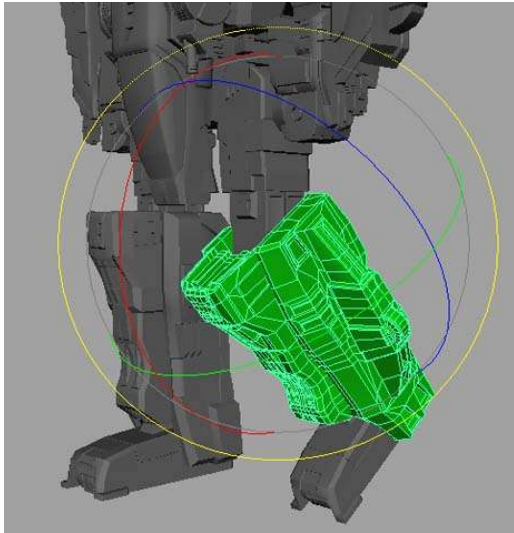


Figure 12. This shows how the body parts of the transformer are used as control objects for the FK rig.

To setup the Transformers body parts as control objects, I used a mel script (cgTKAttachControl) provided with the book (see Appendix A.II). This sped up the process of creating the controller objects for the Transformer. It works by selecting the part of the virtual models geometry you wish to make a controller and then shift selecting the underlying joint. On execution of the script an orient constraint relationship is setup between the controller and the joint. It will also parent constrain the controller

object to the joint. In doing so it will make sure that the controller object's rotation axes are still properly aligned to the joint and that its transforms are set to zero. I made use of another MEL script here, called cgTKDynSelector which adds a 'dynSelction' attribute to each of the controller objects (see Appendix A.III). The script then works by changing the colour of the control that has been selected. This is useful for the animator to clearly see which part of the robot they are affecting.

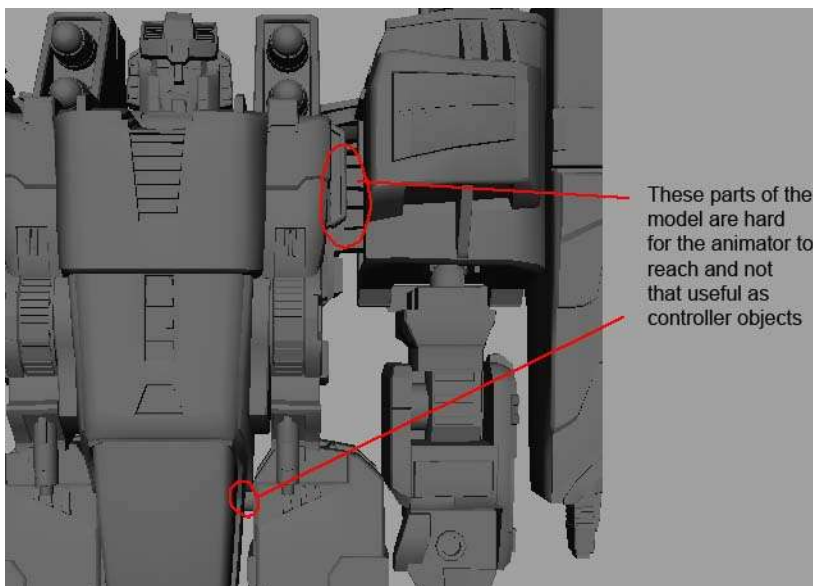


Figure 13. This shows the parts of the Transformer that would be hard for an animator to use as control objects as they are very small and hidden between other parts of the models geometry.

This setup worked nicely for most parts of the Transformer, however, some pieces on the model are very small and quite hard for an animator to select an animate, for example, the parts that attach the Transformers upper leg and

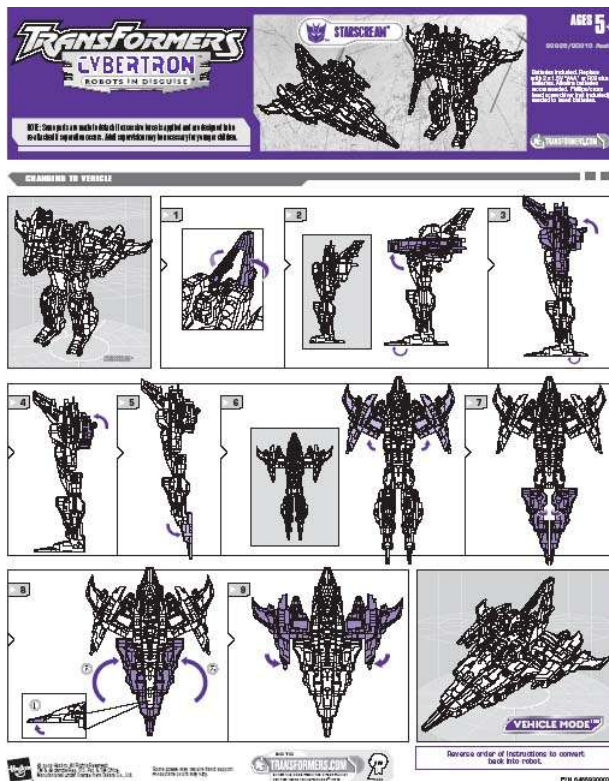
arm to its torso
(see **Figure 13**).

I decided a way of getting around this issue would be to add some float attributes to the upper arm and leg geometry that control the rotations of the joints. I then connected the rotate X channels of the arm and leg connector joints, to their respective attributes on the upper arm and leg pieces. This was achieved using the connection editor to create a link between them. I used this technique on other parts of the Transformers body to make the rig much easier to pose for an animator. To control the guns, for example, the animator simply needs to select the torso and in the channels box for the torso control there are attributes to rotate the guns to there desired position.

To make the transformation rig even more user friendly for the animator, I decided to use a mel script (provided by ‘The Art of Rigging: Volume 2’ book) that sets up a pick walking ability between the controller objects. The script, ‘cgTKPickWalker’ creates a new set of hotkeys that the animator can use to pick walk through the rigs controls (see Appendix A.IV).

I have made sure that all unused attributes in the channel box have been locked and hidden from the animator for each controller. At this stage, the animator could take the rig and set keys on the controllers to animate the series of rotations needed to transform the robot. The transformation steps can be seen in **Figure 14**. If the animator was unaware of how to transform my virtual model, then just having the FK rig as it was at this stage would make it extremely hard for the animator to key the transformation. To solve this issue and to make the rig usable to anybody, I decided to add some custom float attributes that control to what extent different areas of the Transformer have transformed. These are set up using Maya’s ‘Set Driven Key’ option.

So an example of this would be to set up one of the arms to transform, I created a float attribute on the torso controller, named it ‘Transform



Arm', and set it to range from 0 to 10. Using the set driven key option in Maya, I loaded the 'Transform Arm' attribute as the driver and loaded all of the control objects that make up the arm into the driven section. I then set keys between the 0 to 10 value of the 'Transform Arm' attribute to control the order that each part of the arm folded up in. This enables an animator to slide between values to make the arm transform. I setup similar attributes and keys for the other parts of the Transformer so that an

animator can set keys on these sliders to control which part of the Transformer is transforming at each frame. In **Figure 15**, we can see the FK Transformation rig in action and the list of attributes attached to the torso control.

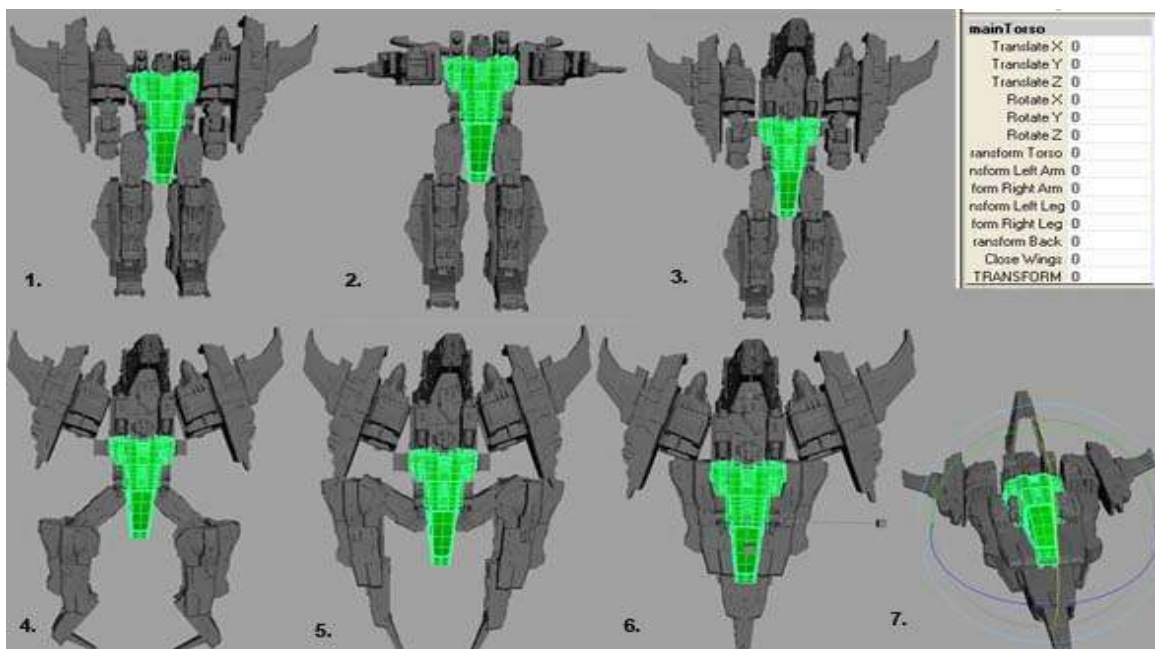


Figure 15. This shows the sequence of the virtual Transformer, transforming into the jet. This is controlled using the attributes attached to the torso control and sliding between 0 and 10. When these attributes are at 0, no transformation has happened (as in Step1). When they are set to 10 the virtual toy is fully transformed (as in Step 7).

Setting up the Vehicle Rig

The rig for the Jet vehicle that the virtual toy transforms into did not need to consist of many controls. It needed to allow for translation and rotation of the jet and to also allow for control of the guns and the side wings. I decided that the most suitable setup for this particular rig was to create a controller object in the form of a Nurbs curve that the animator would select to move and rotate the jet. I then added attributes to this control object that could be used as sliders to control the rotation of the guns and the opening and closing of the side wing flaps. I discovered an efficient and simple technique for linking the attributes on the controller with the appropriate motions that I required from the gun and the wings. This was to create a link between the appropriate joints rotation channel and the corresponding attribute on the controller by using the connection editor. This meant that when the attributes were animated the joints rotated in the axis that they had been linked to. The setup for this rig can be seen in **Figure 16**.

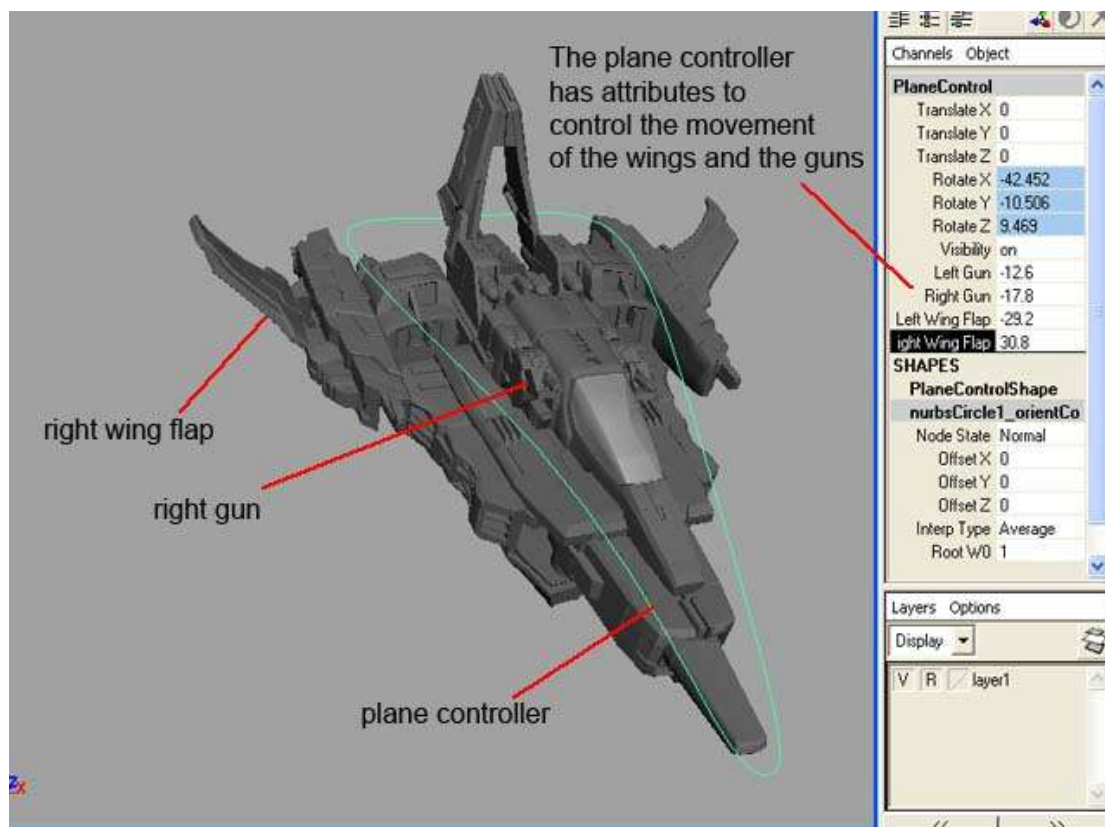


Figure 16. The Vehicle Rig. This shows the rig set up for the Transformer in vehicle mode. We can see the attributes that are attached to the main control to allow for animation of the left and right guns and wing flaps.

The Problems with Creating the IK Rig

The use of inverse kinematics to drive the motion of a character's joint chain provides an easy more practical set up for an animator to control. It allows for a more intuitive way of positioning your character than with forward kinematics, in particular for goal-directed movements. The reason for this is that an animator can focus on the goal they wish the joint chain to reach and not have to consider the rotation of each of these joints in the chain to reach the desired position, as an IK handle will calculate this for them. This setup works effectively on organic characters, where the joints can move relatively freely to reach a desired pose. The toy Transformer, however, is built in such a way that the movements of each of its parts are restricted, often to one rotational axis, due to its mechanical nature.

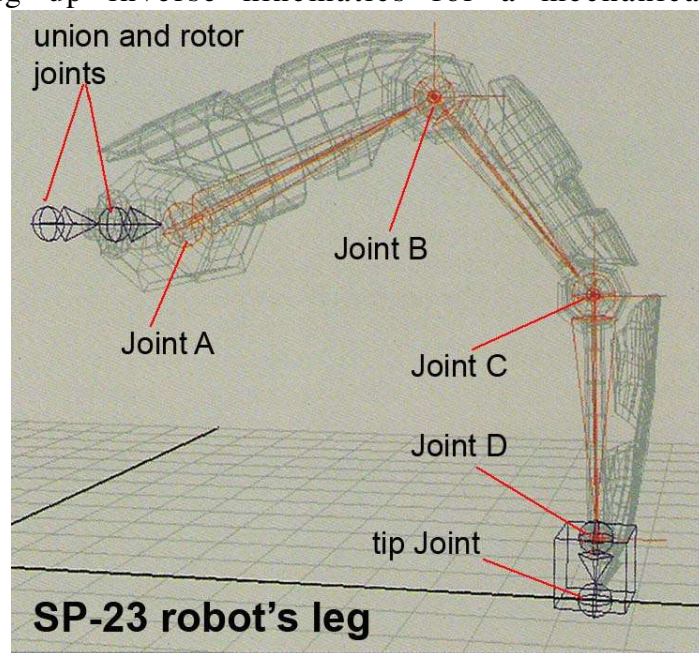
It is evident that setting up inverse kinematics for a mechanical character is not a medial task, as discussed in the chapter on 'Rigging the IK Leg', for the SP-23 robot's leg, in 'The Art of Rigging: Volume 2'.

"The robot's leg is problematic because of its mechanical nature.

Using IK on a leg with so many mechanical constraints

proves extremely difficult. The problem is that Maya's IK solvers do not

support joint limits. That is, an IK solver will attempt to twist a joint in any direction in order to solve the joint chain." Ritchie K, 'The Art of Rigging: Volume 2', 2005



For the SP-23 robot, that is used to demonstrate mechanical rigging techniques in this book, an IK setup was achieved by using pole vector, aim and orient constraints. These constraints helped to create a leg rig that did not break the mechanical nature of the robot's leg. When applying similar techniques to my virtual toy robot, however, I encountered some complex issues that were not dealt with in the book

and which caused the IK leg controls to make the leg rig behave in an odd, non-user friendly nature.

Figure 17, shows the SP-23 robot's leg rig setup. Three IK handles were placed along the chain of joints that runs from joint A to the tip joint. This system of IK handles could be utilised as the joints from A to D all rotated about the same axis. This meant that rotating the IK handle at its end effector at Joint D in the correct axis, caused the rest of the joint chain to follow suit and rotate about the corresponding axes. The main problem with this mechanical leg rig occurred when translating the controller cube, as the union and rotor joints did not follow and resulted in the leg appearing to be broken. The techniques used to resolve this problem involved the use of constraints between the rotor and union joints and the controller cube. The problem with setting up inverse kinematics on my Transformer was that, unlike with the SP-23 robot's leg, the joints along the main leg chain were each restricted to different rotations. This can be seen in **Figure 18**.

SP-23 robot's leg, featured in 'The Art of Rigging: Volume 2'. pg 121

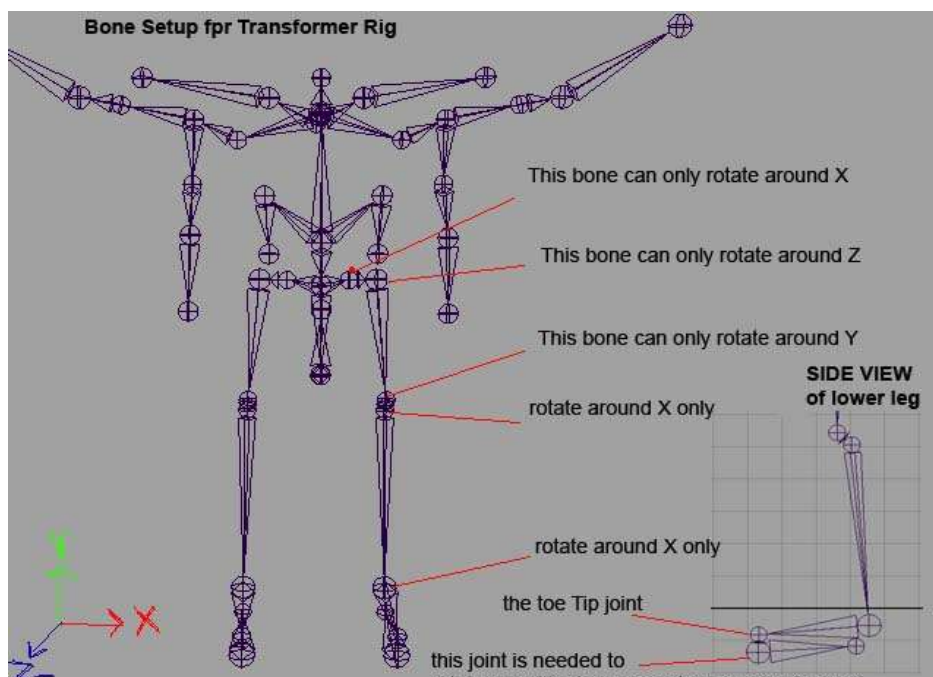


Figure 18. This shows the different rotational restrictions of the joints along the Transformer's leg.

After attempting some different setups for the leg, I managed to create an IK setup, which worked to an extent. The joints never rotate about the wrong axes, and by moving the foot controller, the animator can pose the leg into position, although this sometimes leads to an

unexpected leg movement, especially at extreme poses and when the control is moved in more than one axis.

I went about setting up the IK leg, first by opening up the joint limitation options for each joint and setting limitations on their rotation channels so that each joint would only rotate in the desired way. I then created separate IK handles between each joint along the chain (see **Figure 19**).

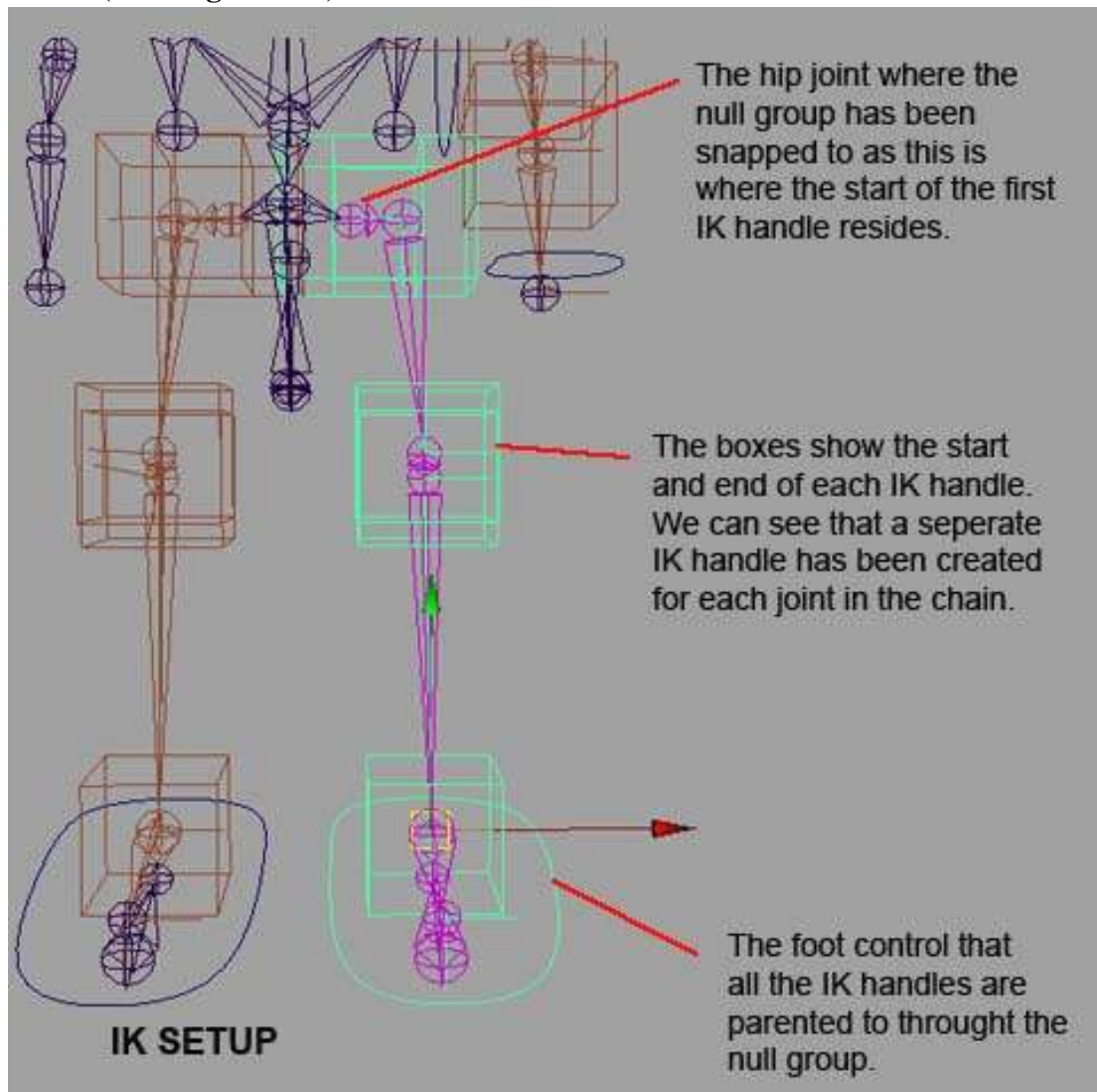


Figure 19. This shows the IK setup that gave me the best results with the Transformer leg, although this setup is still not ideal for an animator to work with.

This seemed the most suitable action to take as there are never more than three joints along the Transformer's leg rig that rotate about the same axis. I then created a null group and snapped its pivot to the first joint at the hip, where the start of the first IK handle resided. I created a controller object and snapped this to the ankle. I then parented all the

IK handles to the null group and parented this to the leg control object. The type of motion that I attained with this can be seen in **Figure 20**.

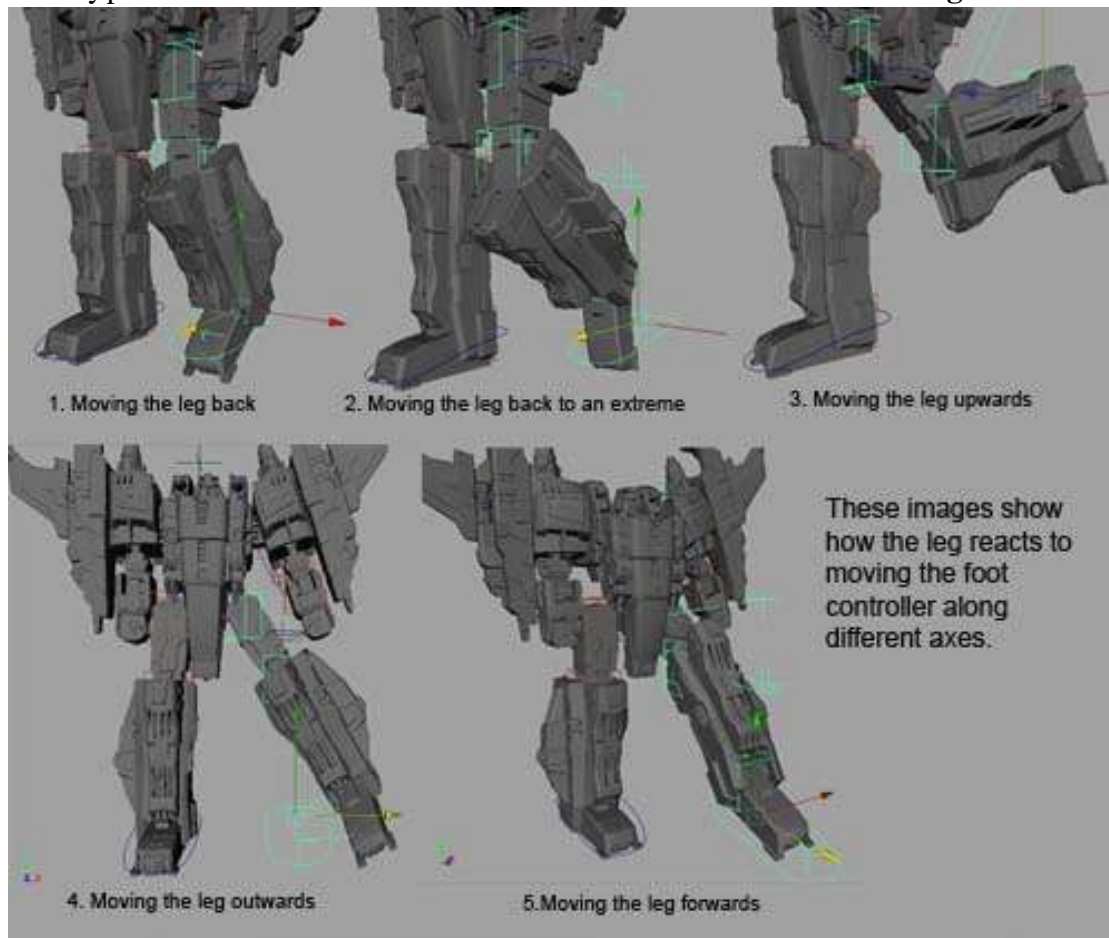
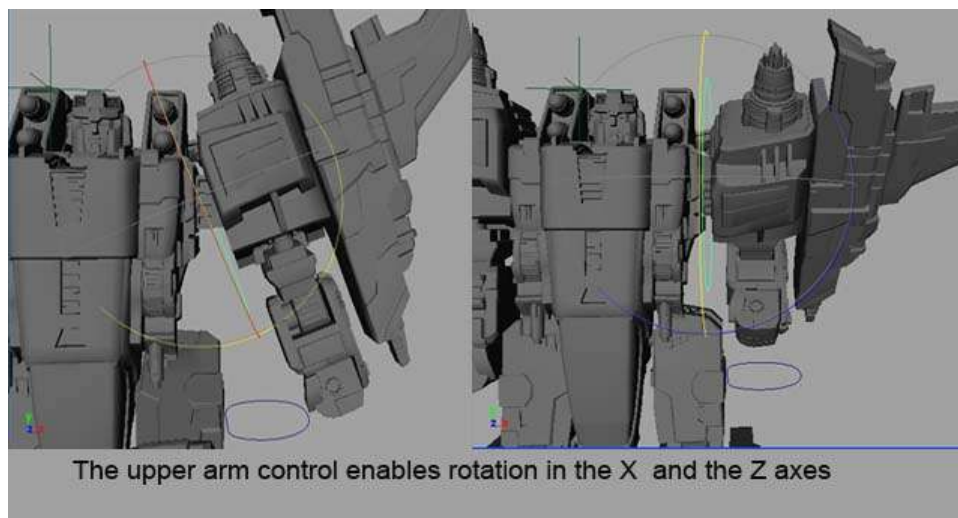


Figure 20. This shows how the IK leg setup reacts to moving the controller object in different directions.

The IK setup seems to work well, until the controller is moved to an extreme and some flipping occurs at the knee. Also moving the control about in more than one axis sometimes moves the leg into an unexpected position. The flipping of the knee could be fixed using a pole vector constraint and any unexpected movements could be controlled by implementing an IK/FK switch. This would allow the animator to pose the leg using the IK rig and to then switch to an FK rig to obtain more control over the position of the leg. When using an IK/FK switch the final position of the joints is determined by blending between the IK pose and the FK pose.

The arms were setup with inverse kinematics but by using two controllers, one placed at the shoulder and one at the hand. By rotating the shoulder control the animator is able to pose the upper arm. As with the leg a separate IK handle was created between each joint along the arm chain. For the upper arm control, a null group was created and

snapped to the pivot of the shoulder joint. The two IK handles that are used to control the upper arm were parented under this null group. I then parented this group to the shoulder control object and restricted



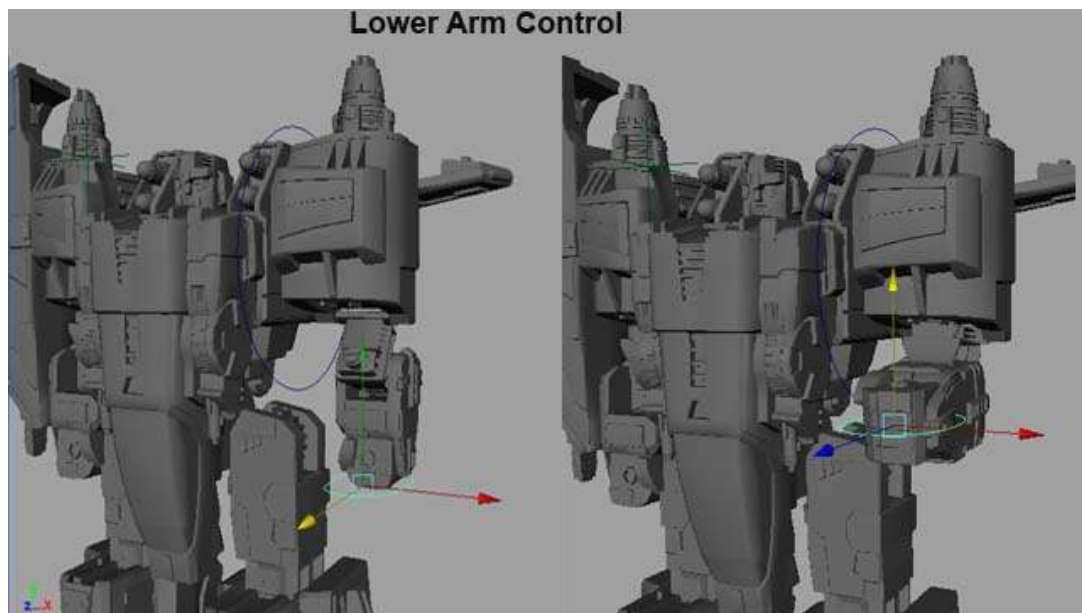
the

Figure 21. This shows the poses that can be achieved by animating the upper arm control.

channels on this so that the animator could only rotate it in the X and Z axes to correspond with how the joints at this part of the arm can move. Because I had limited the rotation of these joints in the attribute editor (as I had done previously with the leg joints), this meant that rotating the control in the Z axis only affected the joint in the upper arm that this motion was possible for. Likewise, by rotating the controller in the X axis, the correct joint would respond to this action and allow for rotation of the upper arm about X. **Figure 21** shows how the upper arm control affects the rig.

The lower arm control was snapped to the joint at the base of the hand. With the limits of the joints rotations restricted, a similar setup was accomplished with the lower arm. A null group was created and snapped to the elbow joint. The IK handles that were placed between the joints in the lower arm were then parented to this null group and again this was parented to the lower arm controller object. We can see how the lower arm rig reacts to translating the lower arm control

backwards and forwards in **Figure 22**.



Due to time constraints, I was unable to fully complete the IK rig setup for the transformer and implement an IK/FK switch. I also wish to develop the rig further and make it more user-friendly at a later stage.

I think it is important to realise that the Transformer has been designed to be a toy and when a child plays with the robot, he/she is effectively rotating each part of the toy at a time to pose it, in an FK fashion. Of course, this does not mean that if the Transformer were to be designed for digital animation that the designer would consider such issues as rigging. The SP-23 robot used in ‘The Art of Rigging’ had been designed with digital animation in mind, but purely for the purposes of being able to cover some complex rigging issues.

“In most real-world cases, the designs are conceived with little or no regard for practicality when it comes to digital animation. If this were the case, designers would be limited and their creativity becomes stagnant.” Ritchie K, 2005, pg 3.

Possible uses of the Virtual Toy as a Marketing Tool

During the process of creating my virtual toy Transformer and setting it up for animation, I began thinking about how this 3D visualization could be used by the toy company to market their product. Computer visualization is used as a marketing tool by some design firms, in particular architectural design companies.

“Animation for architects, makes use of the computer-generated model and the computers processing power to create animation walkthroughs, which can greatly enhance a client’s understanding and speed up the approval process” Curtis B. Charles, 1996, pg 71.

It works as a successful marketing approach for these companies as it gives a clearer image to the client of what the final construction will be like. With this principle in mind, I thought about how the virtual Transformer could act in such a way as to promote its toy counter part.

In an attempt to gain some direct feedback on the extent to which my 3D computer model could be applied as a marketing tool, I sent an email to the toy company, Hasbro, who make these particular Transformer toys (see Appendix B). After some deliberation as to whether I was a student or an inventor (the company does not deal directly with outside inventors as the toy designs are all created in house), my email was forwarded onto the design team at Hasbro. I sent with it some renders of my toy model and also a short video clip of the transformer, so they could see what I had achieved. My main question was to find out whether they would consider using such 3D visualization in any area of marketing their product. As of yet, I have not received any further response from the company regarding my last email to them.

I came up with a range of different possible uses of my own that I thought would be beneficial to the toy company. The toy Transformer product came with a set of carefully drawn out instructions that gave a step by step guide to transforming the toy from robot mode to vehicle mode (see **Figure 14**). One application of my virtual toy could be used as a means for creating the diagrams for such instruction manuals. With my virtual toy set up for animation, it is relatively quick and simple to render a still image of the toy at each stage of the transformation process. I considered this to be beneficial to the toy company as it would enable them to produce higher quality instructions, without being labour intensive or time consuming.

To gauge the length of time it would take to put together a similar set of instructions, I resolved to go ahead and recreate some using my own 3D model and posing it for each consecutive stage of the transformation. I rendered out clear images of the Transformer, using

Mental Ray and made use of its ambient occlusion texture node. I plugged the output of this node into the colour output of the material on my model. The ambient occlusion node effectively lightens the light areas and darkens the dark areas of the shader at render time. By increasing its samples to 100, I was able to render clear, sharp, high quality images to use as the diagrams for the instruction page. I then created the basic structural layout in Photoshop and layered all my rendered images together to create the final instruction page (see **Figure 23**). The instructions took approximately 6 hours of work to obtain my renders and create the structure in Photoshop. I was still unsure at this stage, whether the company would find this particular application of any use to them.

The Hasbro Transformer website has a link to the television commercial that they are currently airing to show off their Transformer toy products. The advert was clearly low-budget and would have been

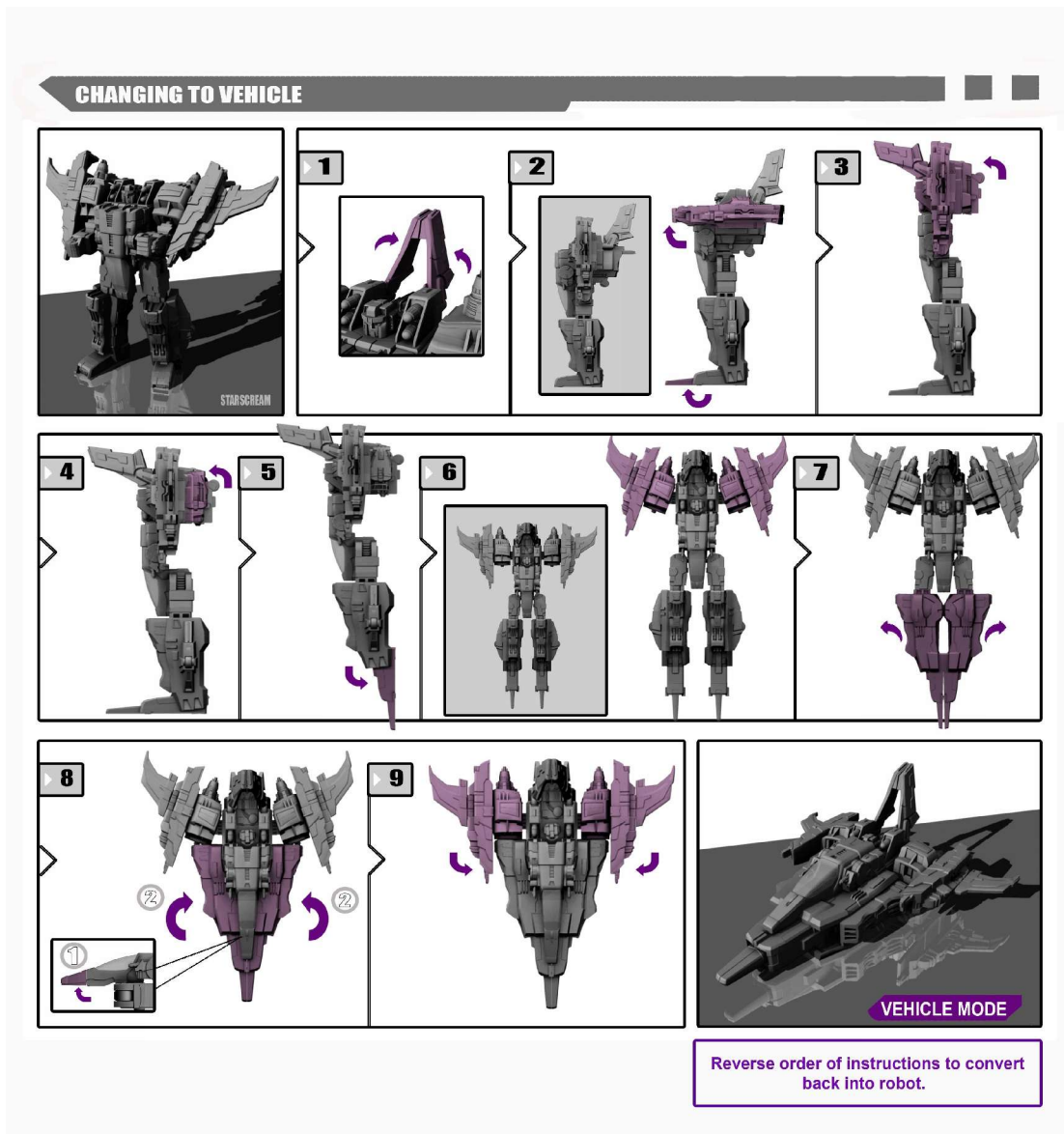


Figure 23. This is the Instruction page that I recreated using my 3D virtual model as the diagram images.

considerably quick and easy to make. Some 2D animation was used in a short section of it, but this was of a fairly low standard. The bulk of the advert showed the toy product being posed by hand and showed off its new, 'exciting' features. I was interested to find out whether the company may find a use for computer animation in their adverts as this could significantly improve the quality of their commercials and may in turn aid in the marketing of their toys. Not only would the use of good quality computer animation improve upon the aesthetics of the advert, it seems that it tends to appeal to the modern market and may act at successfully grabbing their attention more so than just displaying the actual toy figure.

When playing with a toy a child is in effect using his/her imagination to bring the toy to life and pretend that the toy talks, walks and emotes. This concept led me to indulge upon the notion that computer visualization and animation could be an extremely effective marketing tool for the toy industry, as it is in essence used to bring inanimate designs to life. If a child was to see a toy animated and moving like a living being, it may excite them more to the idea of owning the actual toy. This does not necessarily have to occur on a television commercial. The toy company could market their product, using computer animation, by means of other media such as developing a computer game or creating a Flash based interactive website. One particular web-based application for my virtual toy could be as an interactive feature, by which it would be possible to control the Transformer and move it about to transform it

To discuss the ideas that I had come up with and to gain a professional opinion on them, I got in touch with Mike Molesworth, a lecturer for the Interactive Media Production course at Bournemouth University. He gave me an interesting insight into the factors that toy companies, like Hasbro, consider when marketing a toy and how most of them will go about it. It became apparent to me after going through the different possible applications for my virtual toy, that cost is always a major consideration for a company in any marketing strategy. Particularly with toy companies, it seems that they will always opt for cheap marketing solutions as they know that most toys are sold on a fad basis and the market changes so quickly with trends. They often do not want to inject much time and money into these products as their shelf life is often very short with toy crazes coming and going continuously.

This is not the case for all toy companies, however, particularly the larger more established companies. Lego have been a very successful toy company and their product stays continuously popular with children, throughout the changing trends. They can afford to spend money and put effort into utilising new and upcoming marketing ideas. An example of how they have recently used computer animation as a tool to sell their product is with the recent Star Wars Lego computer game (see **Figure 24**). Lego hope that by playing the game and enjoying it, people will want to go out and purchase the Star Wars Lego toys, as they will appeal to them more. They also may want to buy the toys to be able to recreate what they played in the game.



I **Figure 24. The Star Wars Lego video game is an example of a successful application of computer visualization and animation for marketing the Lego toy.**

think the most favourable use of my virtual toy Transformer would be as a character in a computer game. The Transformer is a well established and globally famous toy, so a game would be suitable as the toy has an existing fan base for which such an application would appeal to. Of course, this would not be an achievable project for one person and a company would need to be hired to carry this off. My virtual model could be used as a means to demonstrate to the Hasbro toy company how the characters could look and move as computer generated models in a video game.

Researching into other well known and established toy companies marketing techniques, such as Barbie and Action Man, I discovered they all seem to use Flash based websites (see **Figure 27**). By using animations and incorporating games and interactivity with Flash, the websites become more fun and appealing to the younger market and most probably make the toy more appealing the to a child visiting the website.

“Three-dimensional animation in the Web environment is used for online video games. It is also used as a way to display products, to feature logos and to cybertravel through locations. Viewers can now turn an automobile upside-down and inside out on a web page” Shires, J. 2001, pg 80



Figure 27. This is the home page of the Barbie website. The website uses Flash in such a way that the site is made more appealing to children. The user can click on the different objects within the room to link them to different parts of the site. Barbie is animated to guide the user at each new link. (Image taken from <http://barbie.everythingirl.com>)

The Transformer website could be improved by allowing for more interactivity and implementing my virtual toy in Flash could be one way of achieving this. The user could click on different parts of the toy to move it about, pose it and make it transform. If my virtual model

were to be used for the Transformer website, I would need to significantly reduce the level of detail on it from what it has currently.

On discussing the use of my Transformer model in a television commercial, Mike Molesworth, pointed out to me that it is often the aim of a companies advert to display their product as much as possible to the viewer. This would mean that any computer generated toy would need to be photo realistic for the companies to want to include it, which would result in steeper costs for them too as they would need to employ a computer effects company to work on the advert. However, the detail created in the model does lend itself to photorealism, and with the right shader and light setup this goal could easily be achievable.

With regards to the instruction page that I created (see **Figure 23**), I discovered that the actual instructions that came with the toy (see **Figure 14**), would have been inexpensive for the company to produce as it is the case that such documents are often made in countries where the labour is cheap. My instruction page may be of better quality than the one supplied with the actual toy product, but the company would probably not see this as a good enough reason to change to using computer visualization here as the costs would be significantly larger.

Critical Analysis

Through undertaking this project, I have been able to gain an insight and develop an understanding for the techniques and methods used in modelling mechanical characters and in setting them up for animation purposes. The practical aspects of this project have provided me with interesting challenges and the ideas and techniques that I have explored were all innovative for me in the sense that I had only ever before learnt about and practised organic modelling and rigging. It was good practice for me to learn about modelling an accurate replica of the toy model. I felt this was a valuable skill to learn as such accuracy and precision may be required of me in the future when working on a 3D virtual model for a client in industry.

The final model of the Transformer that I produced, I feel was a success. I believe that one of the main reasons for the model's success was due to the fact that I had the 3D toy Transformer to work from as a reference. Being able to look around the actual toy and have a three

dimensional view of it, made translating it to a high degree of accuracy in Maya much more feasible than trying to obtain this level of precision from 2D model sheets. With future modelling projects, I feel that it would be a worth while practice to create a sculpture or maquette of the character to be modelled before attempting to build it in any 3D computer modelling software. This should enable me to create more complex models to a higher standard aesthetically and also improve upon the believability of my character models as 3D beings. Each time I work on a creating a virtual model in Maya, my ability at modelling improves and I am continually gaining more practice at this skill. The techniques learnt in the process of modelling this transformer have been documented in this report, in the hope that they will be of use to anyone creating robotic or other non-organic models in the future.

After acquiring and working through both volumes in ‘The Art of Rigging’ series, I was keen to apply the new approaches to rigging that I had learnt about to a character model of my own. Modelling the Transformer, provided me with the opportunity to do this, but also presented me with more complex rigging issues such as making an animatable, transforming rig. Through coming up with solutions of my own to solve some of the rigging problems unique to the Transformer, I have gained further confidence in my ability as a character rigger. I have a clearer understanding of setting up the skeleton and the importance of correctly orienting joints as well as the need for accurately placed joints for mechanical characters. I also have a better grasp of setting up inverse kinematics on joint chains where the joints have limited rotations which a new concept to me before this project.

Due to the time constraints of this project, I was unable to develop the Transformer’s IK rig in robot mode as far as I had intended to. If I had planned my time more proficiently, I would have been able to set the IK rig up to be more animator friendly. To do this, I would add constraints to the controller objects so the animator can only key the correct channels and also implement an IK/FK switch, to allow for more control when posing the robot virtual toy. I now understand the importance of planning ahead to realise what issues may occur within a project and to leave adequate time to come up with the solution. This is a valuable lesson that I can carry over to other projects that I work on.

Although I did not write any of my own custom MEL scripts for this project, it has helped me to gain knowledge about this area of Maya. I have a much clearer idea of how it can be used to solve otherwise unsolvable problems within Maya. Becoming more familiar with MEL scripting in this project has opened a window for me to go ahead and utilise MEL as a tool to solve possible technical issues with my Major Project.

The research that I carried out into the extent that 3D computer visualization and animation is used by toy companies as a means to market their products, allowed me to reflect upon the different possible applications for my virtual toy as a marketing tool. I found it interesting to discuss my ideas with Mike Molesworth, and have come away from this project with a deeper insight into how computer animation is used outside of the Film and Games industries. I am keen to expand on this project and finalise the rig for the virtual Transformer so that I can go ahead and create an animation with him. Once I have put together an animation and coloured the virtual model, I am keen to try again to contact Hasbro (perhaps by telephone as this is more direct) and try and arrange a meeting with a member of the marketing team to discuss my project and what, if any, possible uses they can see for it.

Word Count: 9,134

Appendix A

These are the MEL scripts that I used at the rigging stage of this project.

I.

```
//                                cgTkSnaptoPivot.mel
//
//    Directions
//    1.) Select a target object, then shift select
//        the object you wish to move.
//    2.) Execute 'cgTkSnapToPivot'.
```

```
global proc cgTkSnapToPivot()
```

```

{
    //Get current selection.
    string $currentSel[] = `ls -sl`;
    string $base = $currentSel[0];
    string $target = $currentSel[1];

    //Find the world location of the base object's pivot.
    float $basePivot[] = `xform -query -worldSpace -rotatePivot $base`;
    //Move the target object.
    move -absolute $basePivot[0] $basePivot[1] $basePivot[2] $target ;
}

```

II.

```

                                cgTkAttachCtrl.mel

//
//      Copyright © 2005 CGTOOLKIT. All rights reserved.
//      www.cgtoolkit.com
//
//      Description:
//
//This script was designed to attach a controller object
//to a joint. After properly orient constraining the joint,
//the script will attach the controller into the heirarchy.
//
//
//      Directions:
//      1.) Select controller.
//      2.) Shift select joint.
//      3.) Execute 'cgTkAttachCtrl;' from the command
line.
//

global proc cgTkAttachCtrl()
{
    string $sel[] = `ls -sl`;
    string $controlObj = $sel[0];
    string $joint = $sel[1];

    if ( (! `objectType -isType "joint" $sel[1]`) || ($sel[0] == "" ) )
    {

        warning "Please select a controller object, then shift select a
joint.";
    }
}

```

```

}else
{
    //Freeze the transformations on the controller
    select $controlObj;
    FreezeTransformations;
    performFreezeTransformations(0);

    //Snap Pivot to Joint
    float $worldSpaceOfJoint[] = `joint -q -a -p $joint`;
    xform -a -ws -rotatePivot $worldSpaceOfJoint[0]
$worldSpaceOfJoint[1] $worldSpaceOfJoint[2] $controlObj;

    //Create Null
    string $nameOfGroup = `group -em`;
    $nameOfGroup = `rename $nameOfGroup ($controlObj + "Null")
`;

    //Snap Null to Joint
    move -rpr $worldSpaceOfJoint[0] $worldSpaceOfJoint[1]
$worldSpaceOfJoint[2] $nameOfGroup;

    //Orient the Null to the Joint
    delete `orientConstraint -offset 0 0 0 -weight 1 $joint
$nameOfGroup`;

    //Parent the control object under the null
    parent $controlObj $nameOfGroup;

    //Freeze xforms on control again
    select $controlObj;
    FreezeTransformations;
    performFreezeTransformations(0);

    //Orient constrain the joint to the curve.
    orientConstraint -offset 0 0 0 -weight 1 $controlObj $joint ;

    //Parent constrain the null into the heirarchy
    select $joint;
    string $nameOfParentJoint[] = `pickWalk -d up`;

    if ($nameOfParentJoint[0] == $joint)
    {
        warning "Controller was setup, but it was not parent
constrained into heirarchy. (Possibly root a joint?)";
    }else

```

```

        {
            if(!`objectType -isType "joint" $nameOfParentJoint[0]`)
            {
                warning "Controller was setup, but it was not
parent constrained into heirarchy. (Possibly root a joint?)" ;
            }else
            {
                parentConstraint -mo -weight 1
$nameOfParentJoint[0] $nameOfGroup;
            }
        }
    }
}

```

III.

```

//      Copyright © 2005 CGTOOLKIT. All rights reserved
//
//      www.cgtoolkit.com
//
// - Dynamic Selection Script Node -
// This is a script node, that when embedded in your scene file,
// will allow you to create controller objects that turn green
// when the animator selects them.
//
//      Directions:
//      - Open the Expression Editor
//      - Choose "Select Filter > By Script Node Name"
//      - Copy the entire contents of this script and paste it into
//        the expression editor.
//      - Click "Create".
//      - Change the "Execute On" Type to "Open/Close"
//      - Now save your file and re-open.
//      - Any objects that have an attr named 'dynSelector'
//        will turn green when selected.
//
// Create the script node
int $selectionScriptJobA = `scriptJob -killWithScene -event
"SelectionChanged" "cgTkDynamicSelectionChanged"`;

// Create green shader
if( ! `objExists dynSelectorGreen` )
{
string $greenShaderName = `shadingNode -asShader lambert`;

```

```

$greenShaderName = `rename $greenShaderName "dynSelectorGreen"`;
setAttr ($greenShaderName + ".colorG") 1;
setAttr ($greenShaderName + ".colorB") 0;
setAttr ($greenShaderName + ".colorR") 0;
}

//Loop through and find all of the dynSelector objects in the scene.
string $all[] = `listTransforms -geometry`;
global string $allDynSelectors[];

$i=0;
for ($obj in $all)
{
if (`attributeExists "dynSelector" $obj`)
{
$allDynSelectors[$i] = $obj;
$i++;
}
}

//Declare the procedure that is executed when the selection changes
global proc cgTkDynamicSelectionChanged()
{
//Use some string arrays to hold the selections.
string $sel[] = `ls -sl`;
string $selectedDynSelectors[];
global string $allDynSelectors[];
int $i = 0;

//Loop through and find all of the dynSelector objects that are also selected.
$i=0;
for ($obj in $sel)
{
if (`attributeExists "dynSelector" $obj`)
{
$selectedDynSelectors[$i] = $obj;
$i++;
}
}

//Select all of the dynSelectors and apply the initialShadingGroup.
if (`size($allDynSelectors)`)
{
select -r $allDynSelectors;
hyperShade -assign "initialShadingGroup";
}
}

```



```

}
//Select all of the selected dynSelectors and apply the green shader.
if (`size($selectedDynSelectors)`)
{
select -r $selectedDynSelectors;
hyperShade -assign "dynSelectorGreen";
}

//Reset the users selection.
select -r $sel;
}

```

IV.

```

//
// Copyright © 2005 CGTOOLKIT. All rights reserved
// www.cgtoolkit.com
//
// - Custom Pickwalker -
// This script will allow a setup artist to specify custom
// pickwalk behavior for iconic representation.
//
//

global proc cgTkCustomPickWalkLoadObj()
{
    string $sel[] = `ls -sl`;

    if (!( 2 > (`size $sel`) > 0))
    {
        warning "Please select one and only one object to load.\n";
    }else
    {
        button -edit -label ($sel[0]) currentObjButton;

        if (`attributeExists "cgTkPickWalkup" $sel[0]`)
        {
            button -edit -enable 1 -label (`getAttr ($sel[0] +
".cgTkPickWalkup"`)`) upButton;
        }else
            button -edit -enable 1 -label "Up" upButton;

        if (`attributeExists "cgTkPickWalkleft" $sel[0]`)
        {

```

```

        button -edit -enable 1 -label (`getAttr ($sel[0] +
".cgTkPickWalkleft"`)`) leftButton;
    }else
        button -edit -enable 1 -label "Left" leftButton;

    if (`attributeExists "cgTkPickWalkright" $sel[0]`)
    {
        button -edit -enable 1 -label (`getAttr ($sel[0] +
".cgTkPickWalkright"`)`) rightButton;
    }else
        button -edit -enable 1 -label "Right" rightButton;

    if (`attributeExists "cgTkPickWalkdown" $sel[0]`)
    {
        button -edit -enable 1 -label (`getAttr ($sel[0] +
".cgTkPickWalkdown"`)`) downButton;
    }else
        button -edit -enable 1 -label "Down" downButton;
    }
}

global proc cgTkCustomPickWalkLoad(string $direction)
{
    string $sel[] = `ls -sl`;
    string $loadedObject = `button -query -label currentObjButton`;

    if (!(2 > (`size $sel`) > 0))
    {
        warning "Please select one and only one object to load.\n";
    }else
    {
        button -edit -label ($sel[0]) ($direction + "Button");

        if (!(`attributeExists ("cgTkPickWalk" + $direction)
$loadedObject`))
            addAttr -ln ("cgTkPickWalk" + $direction) -dt "string"
$loadedObject;

        setAttr -type "string" ($loadedObject + ".cgTkPickWalk" +
$direction) $sel[0];
    }
}

global proc cgTkPickWalkUp ()
{

```

```

string $sel[] = `ls -sl`;
string $destinationObj;

if (`attributeExists "cgTkPickWalkup" $sel[0]`)
{
    $destinationObj = `getAttr ($sel[0] + ".cgTkPickWalkup")`;
    select -r $destinationObj;
}

}

global proc cgTkPickWalkDown ()
{
    string $sel[] = `ls -sl`;
    string $destinationObj;

    if (`attributeExists "cgTkPickWalkdown" $sel[0]`)
    {
        $destinationObj = `getAttr ($sel[0] + ".cgTkPickWalkdown")`;
        select -r $destinationObj;
    }
}

global proc cgTkPickWalkLeft ()
{
    string $sel[] = `ls -sl`;
    string $destinationObj;

    if (`attributeExists "cgTkPickWalkleft" $sel[0]`)
    {
        $destinationObj = `getAttr ($sel[0] + ".cgTkPickWalkleft")`;
        select -r $destinationObj;
    }
}

global proc cgTkPickWalkRight ()
{
    string $sel[] = `ls -sl`;
    string $destinationObj;

    if (`attributeExists "cgTkPickWalkright" $sel[0]`)
    {
        $destinationObj = `getAttr ($sel[0] + ".cgTkPickWalkright")`;
        select -r $destinationObj;
    }
}

```

```

global proc cgTkSetCustomHotKeys()
{
    //Up
    nameCommand -ann "Custom Pick Walk Up" -c "cgTkPickWalkUp"
cgTkPickWalkUpNameCommand;
    hotkey -keyShortcut "Up" -ctl -name
"cgTkPickWalkUpNameCommand";

    //Down
    nameCommand -ann "Custom Pick Walk Down" -c
"cgTkPickWalkDown" cgTkPickWalkDownNameCommand;
    hotkey -keyShortcut "Down" -ctl -name
"cgTkPickWalkDownNameCommand";

    //Left
    nameCommand -ann "Custom Pick Walk Left" -c "cgTkPickWalkLeft"
cgTkPickWalkLeftNameCommand;
    hotkey -keyShortcut "Left" -ctl -name
"cgTkPickWalkLeftNameCommand";

    //Right
    nameCommand -ann "Custom Pick Walk Right" -c
"cgTkPickWalkRight" cgTkPickWalkRightNameCommand;
    hotkey -keyShortcut "Right" -ctl -name
"cgTkPickWalkRightNameCommand";

    print "Your hotkeys have been setup. You can now use the custom pick
walking. (CTRL + DIRECTION)";

}

////////////////////////////////////
//                               MAIN WINDOW
////////////////////////////////////

global proc cgTkPickWalker ()
{
    if (`window -q -ex pickWalkerWindow`) deleteUI pickWalkerWindow;

    //Main Window
    window -title "CG Toolkit - Custom Pick Walker v1.0" -w 610 -h 150
pickWalkerWindow;

    //Menu Layout
    menuBarLayout;

```

```

        menu -label "Edit";
            menuItem -label "Auto Setup Hotkeys" -c
"cgTkSetCustomHotKeys";
            setParent..;

//Main Column Layout
columnLayout;

        frameLayout -label "Instructions:" -labelAlign "center"
-collapsible 1 -collapse 1 -borderStyle etchedIn;
            columnLayout;

                text "1.) Select an object and load it into the
middle. Then load an object for each direction.";
                text "2.) To use the hotkeys, you must setup your
Maya preferences. Edit > Auto Setup Hotkeys.";
                text "3.) Press CTRL + Direction to pickwalk.";

            setParent..;
        setParent..;

        separator -w 600 -h 15;

//Button layout
rowColumnLayout -nc 3 -cw 1 200 -cw 2 200 -cw 3 200 ;
        text " ";
        button -enable 0 -label "Up" -c
("cgTkCustomPickWalkLoad up") upButton;
        text " ";
        button -enable 0 -label "Left" -c
("cgTkCustomPickWalkLoad left") leftButton;
        button -label "Load a Object" -c
"cgTkCustomPickWalkLoadObj" currentObjButton;
        button -enable 0 -label "Right" -c
("cgTkCustomPickWalkLoad right") rightButton;
        text " ";
        button -enable 0 -label "Down" -c
("cgTkCustomPickWalkLoad down") downButton;
        setParent..;

        separator -w 600 -h 15;

//Show Main Window Command
showWindow pickWalkerWindow;
}

```

Appendix B

This is the documentation of my correspondence with Hasbro

From: Alison Farmer [<mailto:alifarmer12@hotmail.com>]
>Sent: 23 February 2006 15:15
>To: Consumer_affairs
>Subject: 3D Visualisation of 'Starscream' Cybertron Transformer toy

Note - I appreciate that this e-mail may not be suitable for the customer services department. If you could let me know of a suitable e-mail address to send it to, or forward it to the appropriate people, I would greatly appreciate it.

Hi,

I'm a student studying Computer Animation and Visualisation at the National Centre for Computer Animation (Bournemouth University, UK). I am currently researching the possible use of computer animation in the toy industry, and as part of this I have created a virtual model the "Starscream" transformer toy entirely in 3d. This allows us to animate an exact replica of the toy with a great deal of ease, similiar to the "Toy Story" films - it can walk, jump, run, transform into a plane - essentially do anything that the toy can do in real life, and with work could easily be made to look photo real.

The reason for this contact, therefore, is to gauge how useful this would be as a marketing tool for you, the sellers of this product. Please understand that this is only for my own project, and not related to anything commercial in any way. I would really appreciate it if you could take the time to answer the following questions -

Would you find this useful in any area of marketing this toy? If not, why not?

If you would, where can you see it being used? (eg web sites, web/TV adverts, TV series, toy assembly instructions etc)

What else would you like to see done with this technology?

If the time taken to produce this is a factor in your answers the time taken, I have made estimates into how long each stage of development takes (assuming one person is working) -

Creation of the "virtual" toy - 50 hours

Setting up of the toy for transformation and animation - 10 hours

Animation of the toy (any movement other than transforming) -
Dependant on movement, on average around 20 hours work per minute of animation

Colouring of the toy - 10 hours

Please find attached a zipped archive containing 2 stills (in .jpeg format) of the transformer, aswell a short 10 second video (in .avi format) of a very early test of the transformer in action. At your request, more stills and videos could easily be created.

Thank you so much for your time,
Yours,
Alison Farmer

From: "Consumer_affairs" <Consumer_affairs@Hasbro.co.uk>
To: "Alison Farmer" <alifarmer12@hotmail.com>
Subject: RE: 3D Visualisation of 'Starscream' Cybertron Transformer toy
Date: Fri, 24 Feb 2006 11:41:30 -0000

Thank you for your e-mail.

Hasbro thank you for inviting them to consider your idea, however, the company maintains a very strict policy against reviewing such unsolicited proposals from individual inventors as everything is developed in-house.

Apologies for not being able to help you on this occasion.

Kind Regards
ASK HASBRO

-----Original Message-----

From: Alison Farmer [<mailto:alifarmer12@hotmail.com>]
Sent: 24 February 2006 12:42
To: Consumer_affairs
Subject: RE: 3D Visualisation of 'Starscream' Cybertron Transformer toy

Thank you for such a prompt response. I fully understand why Hasbro takes the stance that it does.

I would, however, like to confirm that I am not an inventor, I'm just a student at Bournemouth University in the UK, and that my request was only in the interests of a course project, as any response good or bad from Hasbro regarding the work I've produced would be very valuable in my project report. If necessary I can provide a letter of confirmation from the course leaders to prove that this is just a student project.

I appreciate that this is an unusual request, and hope that you are able help.

Thank you very much for your time,
Yours,
Alison Farmer

Thank you for your e-mail.

I have forwarded your email onto our Design Team. If we are able to assist you in anyway, I will contact you with their response.

Kind Regards
Hasbro Consumer Affairs

References

Books

BROWN, P. 1989. *Computers in Art, Design and Animation*. New York: Springer-Verlag New York Inc.

CHARLES, C.B AND BROWN, K.M. 1996. *Multimedia Marketing for Design Firms*. New York: John Wiley & Sons

DWELLY, B. GRAFT, L. MOGK, C. AND REISBERG, D. 2003. *Learning Maya 5|Character Rigging and Animation*. Canada: Alias|Wavefront, a division of Silicon Graphics Limited.

KUPERBERG, M. 2002. *A Guide to Computer Animation: For TV, Games, Multimedia and Web*. Oxford: Focal Press

MOOIJ, M. 2005. *Global Marketing and Advertising: Understanding Cultural Paradoxes*. 2nd Ed. California: Sage Publications

MOUTINHO, L., CURRY, B., DAVIES, F. AND RITA, P. 1994. *Computer Modelling and Expert Systems in Marketing*. London: Routledge.

RITCHIE, K., CALLARY, J. AND BIRI, K. 2004. *The Art of Rigging: Volume 1*. Canada: CGToolkit.

SHIRES, J. 2001. *Careers in Computer Animation*. 1st Ed. New York: The Rosen Publishing Group Inc.

SLOANE, A. 1996. *Multimedia Communication*. Maidenhead, Berks: McGRAW-HILL Publishing Company.

WAGSTAFF, S. 2004. *Getting a Job in Computer Graphics: Real Advice from Reel People*. Alameda, CA: SYBEX Inc.

YADIN, D. 2001. *Creative Marketing Communications: A Practical Guide to Planning Skills and Techniques*. London: Kogan Page Limited

Websites

SINGH, R. 2004. *The Importance of Industry Based Training in Animation*. [na]:Animation 'xpress. Available from: http://us.indiantelevision.com/anex/y2k4/industry_issues/tony_hr_1.htm [Accessed 27th February 2006].

[na]. 2004. Citroen C4's Transformer Dance | Car Model: Citroen C4. [na]. Available from: www.carpages.co.uk/citroen/citroen_c4_13_11_04.asp. [Accessed 27th February 2006]

[na]. 2005. N/A. Available from: <http://www.hasbro.com/> [Accessed 15th February 2006]

[na]. 2005. N/A. Available from: <http://www.hasbro.com/transformers/> [Accessed 15th February 2006]

[na]. 2006. British Toy and Hobbies Association. [na]. Available from: http://www.btha.co.uk/about_us/general_information.php [Accessed 3rd March 2006]

[na]. 2006. N/A. Available from: <http://barbie.everythinggirl.com/> [Accessed 1st March 2006]

[na]. 2006. N/A. Available from: <http://www.lego.com/starwars/default.aspx> [Accessed 1st March 2006]

[na]. 2006. N/A. Available from: www.actionman.com [Accessed 1st March 2006]

[na]. 2006. *IK legs for Transformer Toy*. Available from <http://forums.cgsociety.org/showthread.php?t=323409> [Accessed 1st March 2006]

