The Making of the Xian Terra-Cotta Soldiers

Nadia Magnenat Thalmann
Igor Sunday Pandzic
Jean-Claude Moussaly

MIRALab
University of Geneva
{Nadia.Thalmann,Igor.Pandzic,Jean-Claude.Moussaly}@cui.unige.ch
http://miralabwww.unige.ch

Abstract

This paper describes the making of a short film on the Xian terra-cotta soldiers using our integrated HUMANOID software. The method of creating and animating the soldiers' faces is first presented. Then, we show how our approach based on metaballs and spline surfaces was used for designing and deforming soldiers' bodies. For the animation of the bodies, we describe the motion control methods. Clothes for the soldiers are then described as well as horses and decor design. For the rendering, we explained our strategy using parallel machines. Finally, problems of integration are addressed.

Keywords: Computer animation, Virtual Humans, face animation, body animation, Xian soldiers

Figure 1: Original terra-cotta statues

1. The Xian Project

Excavation of the grave complex of the Ch'in emperor Shi Huang Ti in Xian in the 1970s has revealed a vast field of life-size terra-cotta statues depicting soldiers, servants, and horses, estimated to total 6,000 pieces [1]. A sense of great realism is conveyed in the figures with remarkable simplicity. The figures were modeled after the emperor's real army, and each face is different. They were supposed to guard the crypt and protect their ruler on his afterlife journey. The figures were carefully fashioned to resemble one of his real infantries, with well-outfitted officers, charioteers, and archers, as well as youthful foot soldiers. Now faded with the passage of time, the army was originally painted in a wide variety of bright colors. Figures 1 to 3 show the terra-cotta statues and their discovery.

Figure 2: Original terra-cotta statues

The Xian project is intended to recreate and give again life to this army using computer-generated techniques. This idea came when two authors of this paper came to China in 1992 and visited the Xian site. We immediately were very fascinated by the realism of these soldiers and found out that our software was best to recreate these soldiers.

The goal of the Xian project is first to make a pilot to be able to show and to test the feasability of a 90 minutes film. For the present time, the pilot is 90 seconds and includes the modelling and the animation of the soldiers. The short scenario that was developed was the following: We see first a scene with the 3D terra-cotta soldiers inside the earth, exactly where they have been found. It is dark with a starry sky. The day is coming so more and more light is appearing. This suddenly awakes one terra cotta soldier. He is extremely astonished to see the scene around himself. He notices the presence of another soldier near him and also his head which is on the ground. He took the head on the ground and put it on the next soldier's body. This latter start to live again. They look at each other, and in the same time, all the army is slowly coming to life. They start to walk again, but the two first soldiers decide to let them go...

Figure 3: Discovery of the statues

2. Creating and Animating the Soldiers' Faces

2.1 Sculpting the Soldiers' Faces

Soldier faces are all different and have details which make the design a tedious task. Methods based on local deformations are certainly the best for modeling human shapes. However, the designer-machine interface is essential. In this section, we show how a our soldiers may be produced with a method similar to the modeling of clay, work which essentially consists of adding or eliminating parts of the material, and turning around the object when the main shape has been set up.

In traditional sculpture, the sculptor selects a specific material corresponding to the type of the object wished. The chosen material constrains the type of treatment. We can consider the different actions involved for different types of materials:

- For sculpture in clay or wax, the sculptor sets up a wooden or metal framework with the model's size, proportions and movement and then he adds matter in order to find the preliminary shape, and establish the volume of the model. His hands are dipping into the matter in order to describe the model's form, more and more in detail.

- For materials such as stone, marble or wood, the sculptor is looking for the model's form in the interior of a mass. The form comes after eliminating parts.

- For materials such as metal, but also any other heterogeneous materials coming from the industry or the nature, the sculptor creates a new sculpture assembling the elements together and giving a very different form from the beginning.

The operations conducted in a traditional sculpture can be performed by computer for computer generated objects. Our sculpting software, developed in the framework of the HUMANOID project from a prototype [2] is based on an interactive input device called the SpaceBall. This allows the user to create a polygon mesh surface. When used in conjunction with a common 2-D mouse such that the SpaceBall is held in one hand and the mouse in the other, full three-dimensional user interaction is achieved.

The SpaceBall device is used to move around the object being sculpted in order to examine it from various points of view, while the mouse carries out the picking and deformation work onto a magnifying image in order to see every small detail in real time (e.g. vertex creation, primitive selection and local surface deformations). In this way, the user not only sees the object from every angle but he can also apply and correct deformations from every angle interactively.

With this type of 3-dimensional interaction, the operations performed while sculpting an object closely resemble traditional sculpting. The major operations performed using this software include creation of primitives, selection, local surface deformations and global deformations.

To create a new sculpture, the sculpting process may be initiated in two ways: by loading and altering an existing shape or by simply starting one from scratch. For example, we will use a sphere as a starting point for the head of our soldiers. We will then add or remove polygons according to the details needed and apply local deformations to alter the shape. When starting from scratch, points are placed in 3D space and polygonized. However, it is more tedious and time consuming.

To select parts of the objects, the mouse is used in conjunction with the SpaceBall to quickly mark out the desired primitives in and around the object. This amounts to pressing the mouse button and sweeping the mouse cursor on the screen while moving the object with the SpaceBall. All primitives (vertices, edges and polygons) can be selected. Mass picking may be done by moving the object away from the eye (assuming a perspective projection) and careful, minute picking may be done by bringing the object closer.

Local deformations are applied while the SpaceBall device is used to move the object and examine the progression of the deformation from different angles, mouse movements on the screen are used to produce vertex movements in 3D space from the current viewpoint. These local deformations make it possible to produce local elevations or depressions on the surface and to even out unwanted bumps once the work is nearing completion. The technique is intended to be a metaphor analogous to pinching, lifting and moving of a stretchable fabric material. Pushing the apex vertex inwards renders a believable effect of pressing a mould into clay.

Global deformations may be also applied to the whole object or some of the selected regions. For example, if the object has to grow in a certain direction, it can be obtained by scaling or shifting the object on the region of interest.

In the making of Xian soldiers, we have started modeling the faces using a sphere (Fig. 4). This process allows the user to keep a better understanding about the final proportions of a human's head. Scaling deformations were first applied to the sphere to give an egg shape aspect, then various regions selected with triangles were moved by translation. At this point, vertices were selected one by one or a few of them and then lifted or moved to the desired locations; however, these kind of manipulations requires a minimum of previous practicing before undertaking such task. Once the egg shape became more human, we split this one in half in order to work more efficiently and without being disturbed by the picking of vertices on the other side of our model. Then the modeling of specific regions was starting, sculpting and pushing back and forth vertices and regions for the making of the nose, jaws, eyes and various landmarks that we had to exaggerate. We were also able to keep various files separated for the nose as an example, so that you can import a region into your main file, a little bit like a cut and paste but in three dimensional world. After these time consuming operations, we obtained a nice half face of the soldier to which we applied a reversed scaling on X axis to produce the other half. The two sides were carefully merged together by picking vertices and merged two at a time which finally gave us our first soldier's face.

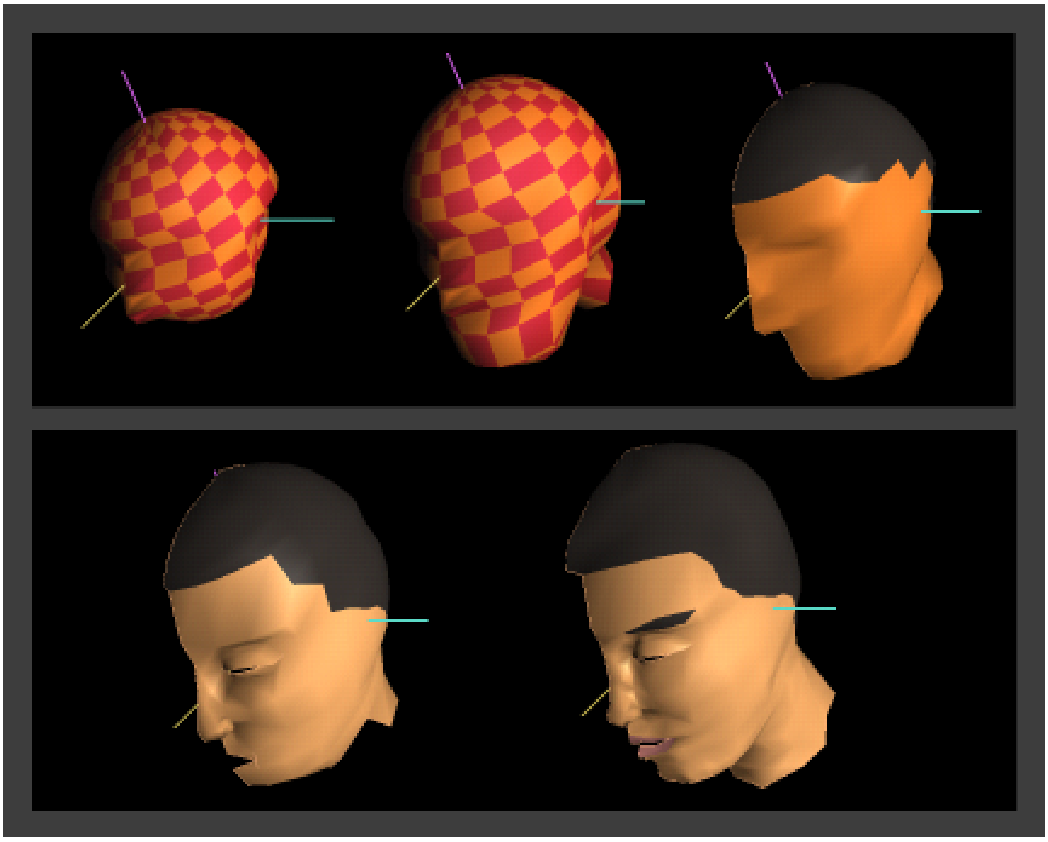

Figure 4: Creation of a soldier head from a sphere

2.2 Our Facial Animation Approach

For the facial animation of the soldiers, we use a multi-level approach, which divides the facial animation problem into a hierarchy of levels that are independent of each other. The higher levels use higher degree of abstraction for defining the entities such as emotions and phrases of speech, by those facilitating animators to manipulate these entities in a natural and intuitive way. A synchronization mechanism is provided at the top level that requires animators to specify the highest level entities: emotions, sentences and head movement with their duration. The system provides default values for duration in case they are not specified. These entities are then decomposed into lower level entities and sent through the pipeline of control at lower levels of the hierarchy. The temporal characteristics of animation are generally controlled at higher levels and the spatial characteristics are controlled at lower levels in our multi level system. For our model [3], the skin surface of a human face that is an irregular structure, is considered as a polygonal mesh. The muscular activity is simulated using rational free form deformations, an extension of original free-form deformations [4]. We employ rational basis function for the trivariate tensor product of Bernstein polynomials. The inclusion of weights for each control point provides an extra degree of freedom for controlling and manipulating the deformations. To simulate the muscle actions on the skin of human face, we define regions on the facial mesh corresponding to the anatomical description of the regions of face where a muscle action is desired. A region is a set of selected polygons in the mesh. A parallelepiped control unit then can be defined on the region of interest. The deformations obtained by actuating muscles to stretch, squash, expand and compress the inside volumes of the facial geometry are simulated by displacing the control points of the control unit. The region inside the control unit deforms like a flexible volume, according to the displacement and the weight at each control point.

The system provides convenient interactive facilities to specify regions, the displacement and weights of control points. We use the 'Ball and Mouse' metaphor described for the sculpting process employing a six degree freedom device, the SpaceBall and the mouse which enable an easy access for various geometric entities in 3D in different orientations.

At the higher level, we need a mechanism of synchronization to ensure smooth flow of emotions and sentences with head movements. A language HLSS (High Level Script Scheduler) [5] is used to specify the synchronization in terms of an action and its duration. From the action dependence the starting time and the terminating time of an action can be deduced.

3. Creating the Soldier Bodies

Modeling and deformation of human body shapes is an important but difficult problem. The human form is very complex, it comes in a variety of sizes and has two main types (or gender); it includes a skeletal frame which supports muscle, fat and flesh, all enclosed by a skin which can slide, stretch, and fold over this volume. The torso, head, arms, and legs are relatively large, roughly symmetrical collections of articulated components which vary with age, gender and race factors. Attempting to model and animate such a structure is one of the most difficult and challenging problems in computer graphics. Moreover, since our eyes are especially sensitive to the human figure, computer generated images must be extremely convincing to satisfy our demands for realism.

Our goal with the soldiers is to make realistic and efficient human modeling and deformation capabilities for many different bodies without the need of physical prototypes or scanning devices. These capabilities will make possible the automatic creation and animation of a rich variety of human shapes.

The metaball technique is inherent to interactive design. One can start with a rough shape consisting just a few metaballs. Then adding details by simply editing metaballs: add, delete, transform, adjust the parameters. However, the metaball technique suffers from two serious drawbacks: first, it requires considerable skill to construct complex objects by properly generating numerous overlapping functions. Second, interactive shape design demands quick feedback when the designer is editing blobs. Unfortunately polygonalization or ray tracing is usually required to visualize a soft object which is very expensive.

In order to enhance the modeling capability of the metaball technique, we devised some ways to reduce the number of metaballs needed for a desired detail shape, and to allow the designer to manipulate blobby surface at interactive speed. A cross-sectional based isosurface sampling method, combined with the B-spline blending, enable us to achieve those two goals.

From a practical point-of-view, we have written an interactive metaball editor for shape designers. We start the shape design by first creating a skeleton for the organic body to be modeled. Metaballs are used to approximate the shape of internal structures which have observable effects on the surface shape. Each metaball is attached to its proximal joint, defined in the joint local coordinate system of the underlying skeleton which offers a convenient reference frame for position and editing metaball primitives, because the relative proportion, orientation, and size of different body parts is already well-defined.

In our editor, the workstation can display shaded or wireframe color metaballs (Fig.5) and high resolution body envelope (Fig.6) near real time. Metaballs are displayed as ellipsoids either with effective radius or threshold radius. The "threshold" mode shows the visible size of metaballs, while the "effective" mode shows how far the metaball will influence. Some widget panels are used to interactively adjust the size, weight, position, orientation of metaballs. SpaceBall or trackball enables the user to rotate models around in space for different viewing. By turning on/off various display entities of different layers, the designer can selectively check skeleton, metaballs, cross-section contours, and skin envelope simultaneously. The designer can interactively create, delete, pick, joint attach/detach metaballs. A file format is designed which can store both joint hierarchy and metaball information. Models can be saved to a file and load in later for successive sculpting. The designer can get quick feedback of the resulting skin form.

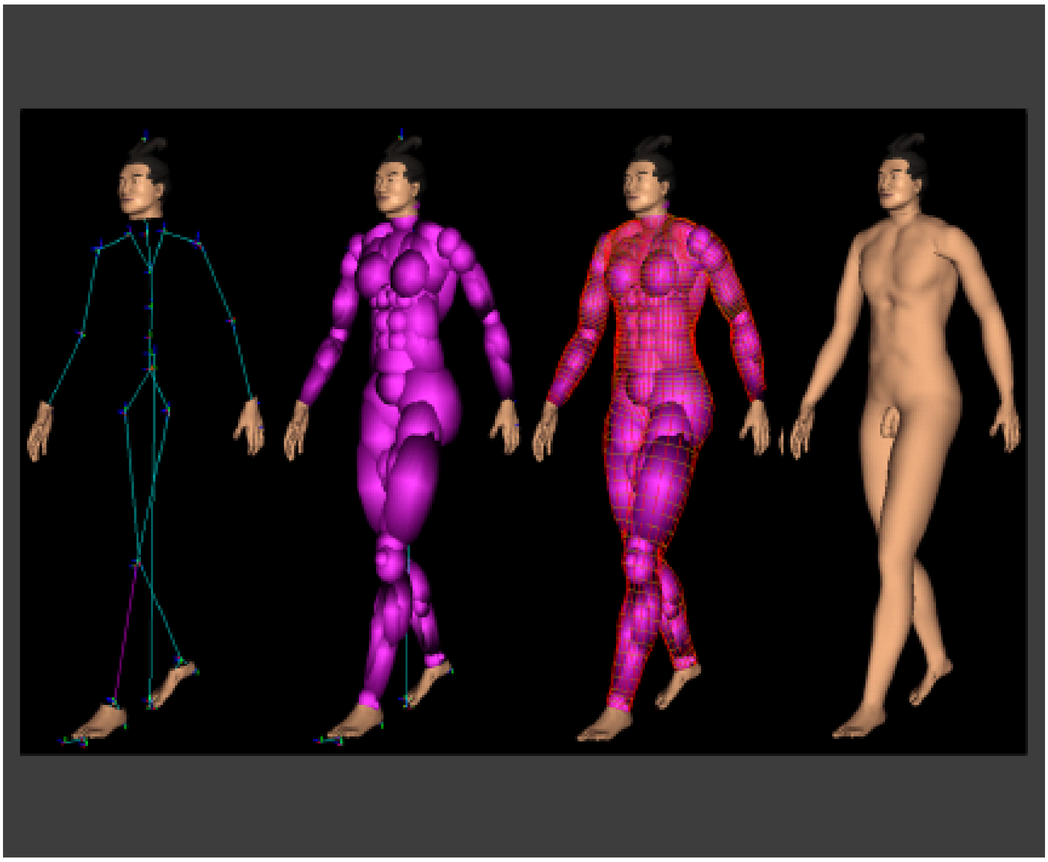

Figure 5: Metaball-based body in sading and wireframe

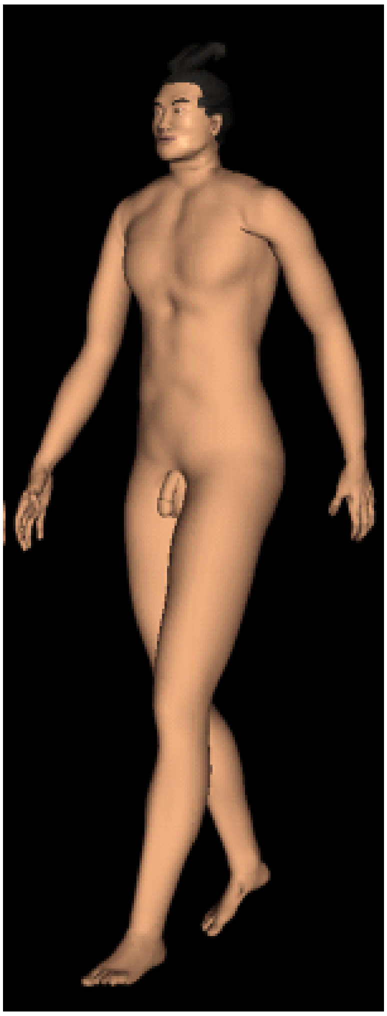

Figure 6: High resolution envelope

For the Xian soldiers, photo references and various information about the statues were taken into consideration such as the scaling of the height and shoulders for example. The metaballs hierarchy was taken from a standard model which we already had produced, we then modified the metaballs positions and shapes to fit the needs of our soldiers anatomy. The head, hands and feet were modeled with the software techniques mentioned in chapter 2 and then attached to our body envelope. This last manipulation was executed interactively by importing the head to the body envelope and then by selecting the vertices located at the joint connection of the neck. When recognition about the location of these vertices has been made, the head could be attached to the body envelope (Fig. 7), the same procedure for the hands (Fig.8) and the feet was used.

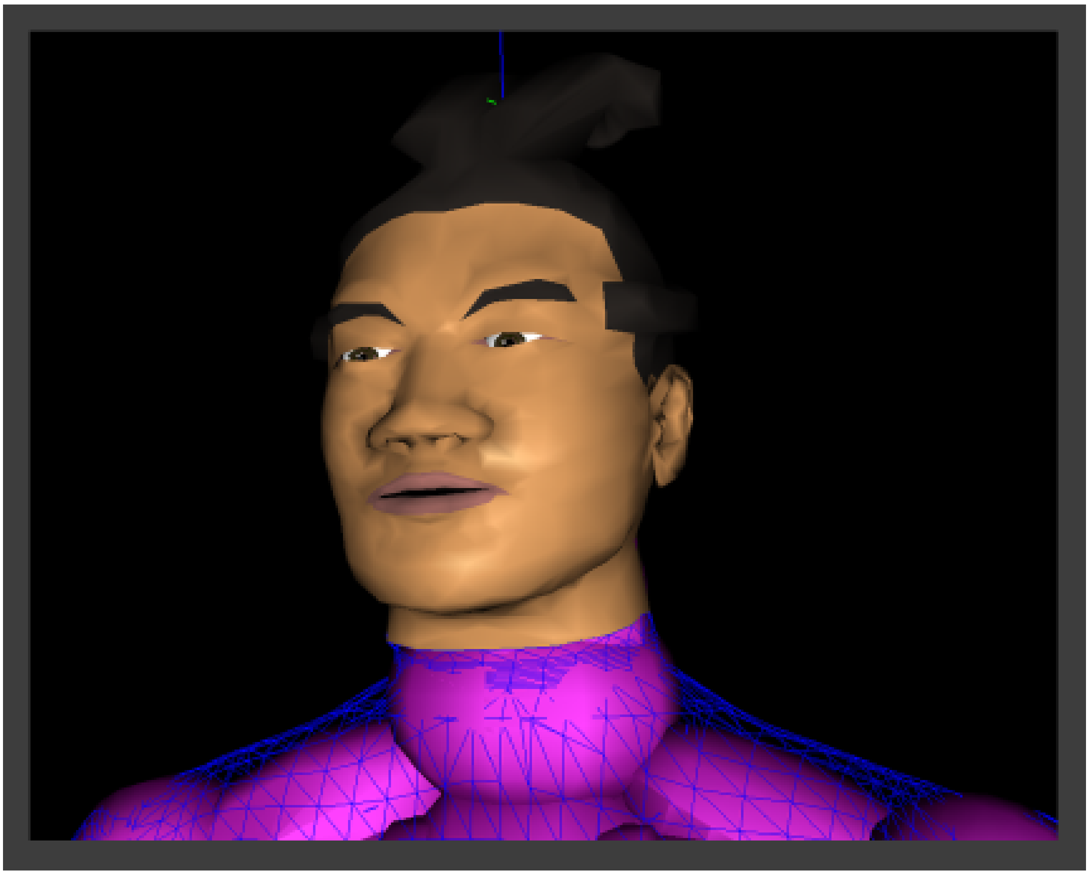

Figure 7: Head attached to the body

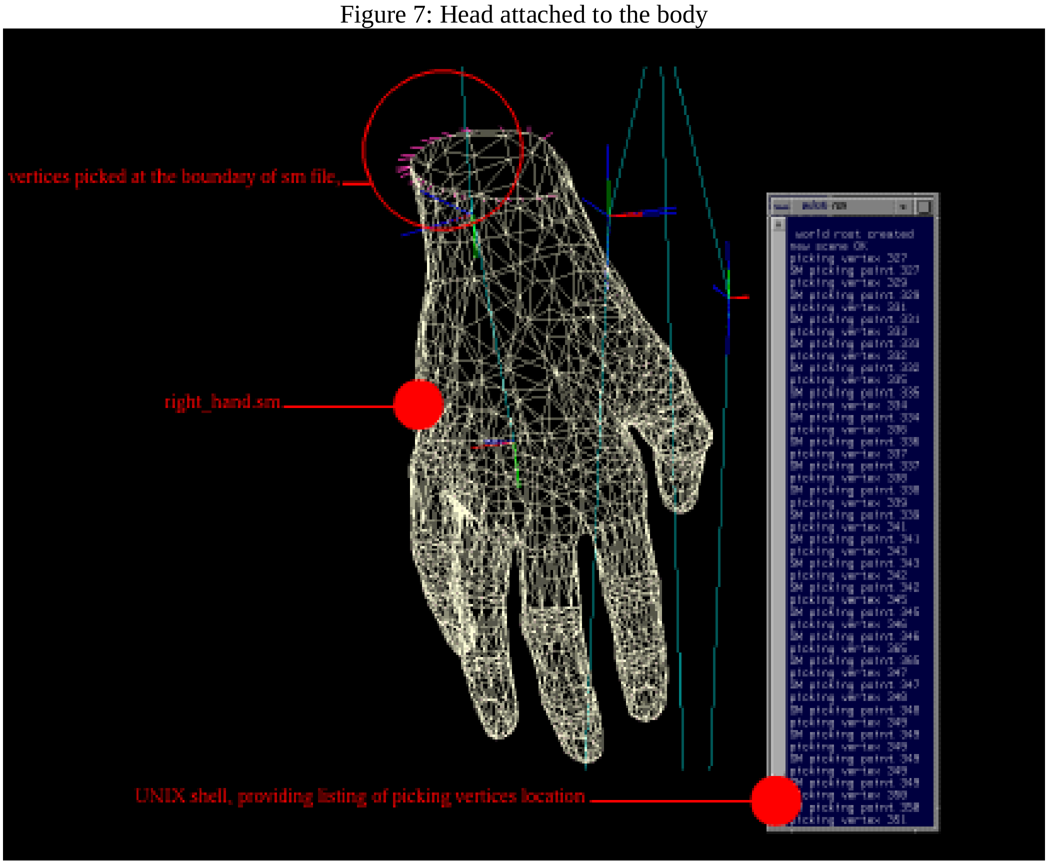

Figure 8: Hand attached to the body

4. Animating the Soldiers Bodies

For creating the body animation of the soldiers, we have used the TRACK [6] motion control system. As there is no general method applicable to complex motions, only a combination of various techniques may result in a realistic motion with a relative efficiency. Consequently, the TRACK system has been based on integrated methods. The TRACK system has two major goals, first integrating a wide range of motion generators within the unified framework of multiple track sequences, and second, providing a set of tools for the manipulation of these entities. TRACK is an interactive tool for the visualization, editing and manipulation of multiple track sequences. A sequence is associated with an articulated figure and can integrate different motion generators such as walking, grasping, inverse kinematics, dynamics and key framing within a unified framework. The system provides a large set of tools for track space manipulations and Cartesian space corrections. This approach allows an incremental refinement design combining information and constraints from both the track space (usually joints) and the Cartesian space. We have dedicated this system to the design and evaluation of human motions for the purpose of animation. For this reason, we also insure the real-time display of the 3D figure motion with a simultaneous scan of the 2D tracks. The interface design and the interaction device integration are realized with the Fifth Dimension Toolkit [7].

The function of TRACK can be basically divided into two parts: the motion generation, and the track editing. The corresponding user interfaces are Motion Control Panel and Track Manipulation Panel. The TRACK Main Control Panel is relatively simple. Its function is to select between Motion Control Panel or Track Manipulation Panel. Both of them can activate a series of sub-control panels. The structure of the whole panels can be divided into three levels. There is one special Message Panel that will display the system status and helpful message.

For the movie production we have established the motion path and various movements of each soldier from a previous storyboard study. We then started to animate skeletons one by one and finally recorded the track editing.

5. Creating Clothes for Soldiers

The clothes creation [8] process closely resembles the real-life clothes design performed by a tailor. The cloth panels are first designed ("cut") in two dimensions, as a tailor would do, and texture is defined for each one. Then the seams are defined. In a second step, the clothes are placed in 3D around a body. We use mechanical forces along the seaming lines to bring them close, wrapping the panels around the body. Finally, seaming is performed and the panels are merged together to form the garment as a single object.

When the actor is animated the simulation is performed and the movements of soft surfaces composing the clothes are calculated using collision detection and a mechanical model. The animated clothes are recorded frame by frame for integration in the final animation sequence.

The seaming and simulation system consists of a single program that provides facilities for importing various objects coming from different animation sources, and for animating soft objects of any kind using a mechanical model and collision detection for interaction with the environment.

The underlying mechanical model has been designed to be as general as possible. Thus, non regular triangular meshes can be used, giving full freedom for the shape of the soft surfaces and allowing complex cloth designs to be handled as individual objects, without complicated techniques for maintaining surface panels together. The collision detection algorithms were designed for efficiency, particularly for self-collisions, taking into account surface curvature for computing collisions. The surface thickness is simulated using proximity detection. Collision inconsistencies are detected and corrected to give a robust computation on multilayer collision situations.

Using physics to animate clothing is a real challenge and so was the making of the wardrobe to be worn by soldiers. Careful research had to be made in order to produce a visual output that corresponds to the real Qshuang statues.

6. Horses and Decor

The horses and the decor were created (Fig. 9 and 10) using the sculpting software described in Section 2, beginning with some simple primitives. The decor was sculpted out of a single plane, and the horses out of several cylinders and spheres.

The facial animation software was adapted to animate the horse. This was possible because the software is based on the Free Form Deformations which can be applied to user-defined regions of an object. So, it was enough to define the appropriate regions on the horse body, i.e. legs, neck etc. and apply the Free Form Deformations to them.

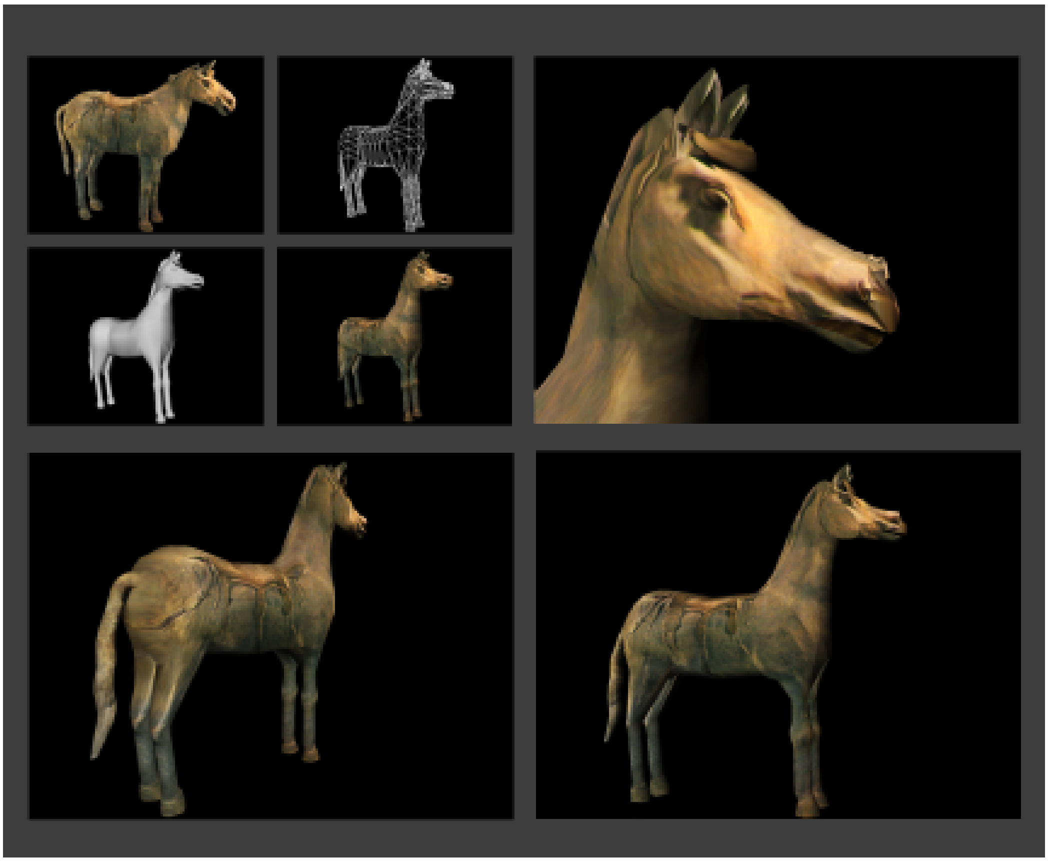

Figure 9: The making of the horses

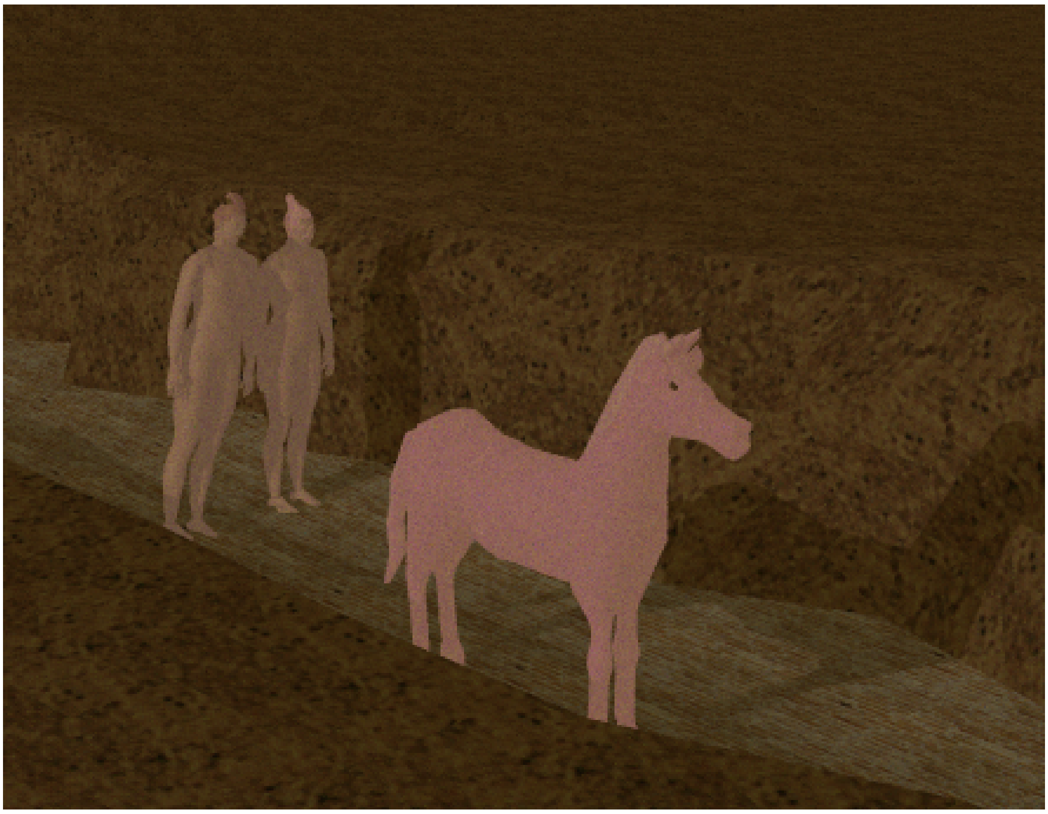

Figure 10: Horse and decor

7. Rendering

7.1 The Problematics of Rendering

Considering the number of actors and objects, as well as their level of detail, it was obvious from the very beginning of this project that the demands on rendering will be enormous. In order to perform the rendering in a reasonable time, we have chosen to develop parallel versions of the public-domain raytracing software Rayshade. The parallel versions of this software, running on the IBM SP2 and the CRAY T3D, provide us with fast rendering of the complex scenes.

We also had to develop our own interactive tool for material and texture management. Texture mapping on highly irregular surfaces is a particularly interesting problem, because the standard orthogonal or spherical mapping does not give good results.

7.2 Material and Texture Management

We have developed an interactive tool called Material Manager to manage material properties and textures. All the standard material properties (ambient, diffuse etc.) are supported, as well as multiple textures. It is possible to map two or even more textures on the same objects, e.g. one color texture mapped on diffuse and ambient coefficients and one bump texture. The lights can also be manipulated.

The Material Manager visualizes the objects with materials and textures in real time using SGI Graphics Library for a quick preview, and the Rayshade rendering can be spawned interactively from the user interface to see the final rendering results.

Texture mapping on highly irregular surfaces (e.g. horse's body) is problematic because the standard orthogonal or spherical mapping does not give good results. In this case, we map the texture on each triangle independently, keeping the same texture scale. This technique is appropriate for the soldiers (Fig.11) made of terra-cotta because the terra-cotta texture is quite homogeneous, it attaches seamlessly on the triangle boundaries. On the soldiers' faces we use slight color modulation of the texture to highlight particular regions such as eyes and eyebrows. For the clothes, the texture coordinates are calculated while the cloth panels are flat. The texture then stays correctly mapped when the clothes are deformed. To produce the relief on the soldiers' armors, we use specially designed bump textures together with the color textures.

Figure 11: Textured head

7.3 Special Effects

For the effect of dawn, with the starry night sky turning into early morning sky, we have used two large concentric spheres containing the whole scene. On the outer sphere the night sky texture is mapped and on the inner one we map the day sky. The inner one is completely transparent in the beginning, than it gradually becomes more opaque until finally it completely hides the outer sphere. The lights increase the intensity as the sky becomes clearer to produce the daylight effect. The sky images were produced by hand painting, scanned and worked on with Adobe PhotoShop and then, mapped on the spheres.

8. Integration

The body models (without hands, feet and head) are created using the metaball-based modeler Body Builder. All other parts of the body, as well as the decor and other objects, are created using the sculpting software Surface Manager. Facial animation script is produced using the interactive tools, then the Face software produces the animation based on the script. The Track software is used to animate the bodies. It produces the animation of the skeleton in the form of joint trajectories. Based on the Track output, Body Builder deforms the bodies to produce natural looking movements, at the same time integrating the animated heads, as well as hands and feet. The hand deformation algorithm is integrated in Body Builder.

The camera manipulation is done using Wavefront (this is the only commercial software package we used). The objects, actors and trajectories have to be transformed into Wavefront format to manipulate the camera, then the final camera trajectory definition is obtained by conversion from Wavefront to a custom format.

The clothes are designed in 2D using the Garment2D clothes modeler. The animation of the clothes is done by Garment3D clothes simulator, based on the animated bodies. The result are the files with animated clothes.

Material Manager is a software that allows to define the material properties for all the objects, including color and bump textures. It is also used to define the light positions and intensities. Based on this light definition, the Makelight utility serves to animate the lights, i.e. to change their position and intensity. This utility is also used to create the effect of the night sky passing into day sky.

Finally, the actors, clothes, scene, objects, camera trajectory, materials and lights are passed to the Rayshade converter that produces the rayshade files. These files are rendered by Rayshade to produce the final images for recording.

Acknowledgments

This project has been implemented with the students of the course “Informatique et Communication Multimedia” from the University of Geneva. This course is part of a common graduate curriculum which includes the Swiss Federal Institute of Technology, the University of Geneva, the University Lyon-1 and Lyon-2 and the “Ecole des Mines de Saint-Etienne”. We would like to thank the students of this course for their enthusiasm, time and effort to collaborate to the making of the scenario, the design and the animation of the soldiers. We would like also to thank the research people of MIRALab and Swiss Federal Institute of Technology for their assistance in this project when needed. The software used for creating this film has been sponsored by the Swiss National Research Foundation, the ESPRIT Project HUMANOID and the Federal Office for Education and Science.

Nadia Magnenat Thalmann is full Professor of Computer Science at the University of Geneva, Switzerland and Adjunct Professor at HEC Montreal, Canada. She has served on a variety of goverment advisory boards and program committees in Canada. In 1987, she was nominated woman of the year by the Montreal community in Quebec. She has received several awards, including the 1985 Communications Award from the Government of Quebec, the Moebius Award from the European Community in 1992, and the British Computer Society Award in 1993. Dr. Magnenat Thalmann received a BS in psychology, an MS in biochemistry, and a Ph.D in quantum chemistry and computer graphics from the University of Geneva. She has written and edited several books and research papers in image synthesis and computer animation and was codirector of the computer-generated films *Dream Flight*, *Englantine*, *Rendez-vous à Montreal*, *Galaxy Sweetheart*, *IAD*, *Flashback*, *Still Walking* and *Fashion Show*. She has served as a chairperson of Graphics Interface '85, Computer Graphics International (CGI'88), and the annual workshop and film festival in Computer Animation held in Geneva. She is co-editor-in-chief of the *Visualization and Computer Animation Journal*, associate editor-in-chief of the *Visual Computer*, editor of the *Computational Geometry Journal* and the *CADDM* journal. She is the President of the Computer Graphics Society (CGS).

Igor Sunday Pandzic is a researcher and PhD student at MIRALab, University of Geneva. He received his Diploma in Electrical Engineering from the University of Zagreb, Croatia and the Diploma of Visualization and Graphics Communication from the Swiss Federal Institute of Technology in Lausanne in 1994.

Jean-Claude Moussaly is designer at MIRALab, University of Geneva. He received a Bachelor of Fine Arts from the University of the Arts in Pennsylvania in 1987 and the Diploma of Visualization and Graphics Communication from the Swiss Federal Institute of Technology in Lausanne in 1994.

References

1 Terra-Cotta
2 LeBlanc A, Kalra P, Magnenat-Thalmann N, Thalmann D (1991), Sculpting with the "Ball & Mouse" Metaphor, Proc. Graphics Interface '91, Calgary, Canada, pp. 152-159.
3 Kalra P, Mangili A, Magnenat-Thalmann N, Thalmann D (1992) Simulation of Muscle Actions using Rational FreeForm Deformations, Proc. Eurographics ‘92, Computer Graphics Forum, Vol. 2, No. 3, pp. 59-69.
4 Sederberg TW, Parry SR (1986) Free Form Deformation of Solid Geometric Models, Proc. SIGGRAPH '86, p p. 151-160.
5 Kalra P, Mangili A, Magnenat-Thalmann N, Thalmann D (1991) SMILE : A Multilayered Facial Animation System, Proc IFIP WG 5.10, Tokyo, Japan (Ed Kunii Tosiyasu L) pp. 189-198.
6 Boulic R, Huang Z, Magnenat-Thalmann N, Thalmann D (1994) Goal-Oriented Design and Correction of Articulated Figure Motion with the TRACK System, Computers and Graphics, Vol. 18, No. 4, pp. 443-452.

7 Turner R, Gobbetii E, Balaguer F, Mangili A, Magnenat-Thalmann N, Thalmann D (1990) An Object-Oriented Methodology using Dynamic Variables for Animation and Scientific Visualization, Proceedings of CGI'90, pp.317-327.

8 Carignan M, Yang Y, Magnenat Thalmann N, Thalmann D (1992) Dressing Animated Synthetic actors with Complex Clothes, Proc. SIGGRAPH'92, Computer Graphics, Vol. 26, No 2, Chicago, pp. 99-104.