Invited Paper

Virtual humans acting in virtual reality

N.M. Thalmann

MIRALab, University of Geneva, Switzerland

ABSTRACT

Several very complex problems must be solved in order to render three-dimensional animation involving virtual humans in their environment. In this paper, we explain several of these problems and present solutions. In the context of interactive animation systems, the relationship between the animator and the virtual humans should be also emphasized. We present an example of facial communication between the animator and virtual humans.

INTRODUCTION

The ultimate reason for developing virtual humans (also called synthetic actors) who seem real is to be able to use them in any virtual scene representing the real world. Anyway, a virtual scene, beautiful though it may be, is not complete without people...virtual people that is. Scenes involving virtual humans imply many complex problems that we try to manage since several years [1]. We slowly come to the point of simulating real-looking virtual humans, taking into account body, face and cloth deformations. In a short future, we will not be able to see any difference between a real person and a virtual one. Any environment could be simulated and consequently, we will be able to experiment in real-time any virtual environment, and to communicate with virtual humans rather naturally. Any simulation will be possible: from the infinite world to the megaworld passing through any daily live situation.

CREATION OF VIRTUAL HUMANS

The synthesis of realistic virtual humans leads to obtain and include the specific features of the character of interest. For the universally known personalities (actors) such as Marilyn, Humphrey, and Elvis, there is a less scope to make mistakes as
the deviations will be very easily detected by the spectator. In spite of this ambition to make realism, or better, imitation, this type of realism should not be confused with the photographic or the cinematographic realism.

**Sculpting the shape**
Creating a body for a virtual human is only the first step, his particular character depends on his body movements and his personality is defined by the subtle changes of his facial expressions and other gestures.

To construct these shapes, we propose the use of an interactive sculpting approach. The surfaces of human face and body are irregular structures implemented as polygonal meshes. We have introduced a methodology [2] for interactive sculpting using a six-degree-of-freedom interactive input device called the Spaceball. When used in conjunction with a common 2D mouse, full three dimensional user interaction is achieved, with the Spaceball in one hand and the mouse in the other. 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.

Typically, 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 a person and use cylinders for limbs. 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 may be more tedious and time consuming.

With this type of 3D 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.

**The selection process**
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 surface deformations.**
These tools make it possible to produce local elevations or depressions on the surface and to even out unwanted bumps once the work is nearing completion. 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. 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**
These tools make it possible to produce global deformations on 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 construction of a certain category of figures like realistic human bodies, it is often preferable to keep certain irregularities on the surface. A very smooth skin, for example, does not necessarily guarantees a more realistic appearance. In very delicate parts like shoulders or cheeks the imperfections are visible and generally we have to get rid of them right away, but we can keep some others small irregularities in order to create a figure which seems less plastic and robotized and in order to attenuate the feeling we often feel in front of computer generated human bodies. This kind of imperfections in a realistic figure give the impression of a human figure which has not been conceived only in the "designer's" head but has really been observed from the reality.

**Simplification of polygonal meshes**
The main goal of shape creation is the realism of the final virtual humans. It should look as "human" possible. This constraint implies that the number of faces used to model the body does not come into consideration. What might appear as a small drawback when rendering fixed pictures becomes a problem in animation. Algorithms applied to a virtual human for animation have some others constraints, and among these realism increases computing time the most. If we take the example of an actor dressed with synthetic clothes, one time consuming process in the animation of clothes is collision detection of the clothes with the body. The complexity of these algorithms is dependent on the number of
faces of the colliding objects. Moreover, most of the body parts involved in collision detection are covered by the clothes during the animation. There is no need for a very precise body. The solution is to reduce the number of faces used to render the synthetic actor. This kind of reduction can't be applied uniformly to the whole object. If the actor is dressed in a skirt, and short-sleeved shirt, arms and legs need to be as precise as possible since they will appear in the animation.

We need a method to reduce the number of polygons of objects used in animation according to algorithm requirements, thereby reducing computation time and space. The reduction can be locally applied and controlled by the shape information the user wants to keep. We may summarize the whole process of simplification, more details may be found elsewhere.

- Selecting a connexe region of the object by picking the triangles.
- Orthogonal projection of the region onto a plane (defined by an arbitrary point and a vector normal as an average of the normal of all triangles in the selected region). If the projection of a triangle is inside the border of the triangles already projected, it is eliminated from the region.
- Extraction from the 3D region of the elements defining shape of the region. These elements are either points or edges.
- Selection of the elements to keep inside the region.
- Deletion of all other elements.
- Triangulation of the 2D projection using the selected elements to keep and the elements defining the border of the 2D projection.
- Reconstruction of the simplified 3D region from the 2D triangulation. As the border of the region has been kept, there is no problem reconstructing the region and inserting it in the whole object.

Fig. 1 shows an example.

MOTION OF VIRTUAL HUMANS

The human animation is very complex and should be split into body motion control and facial animation. Basically a virtual human is structured as an articulated body defined by a skeleton. Skeleton animation consists in animating joint angles. There are two main ways to do that: parametric keyframe animation and physics-based animation. Another complex objective is to model human facial anatomy exactly including its movements to satisfy both structural and functional aspects of simulation.
Body motion
Basically a virtual human is structured as an articulated body defined by a skeleton. When the animator specifies the animation sequence, he/she defines the motion using a skeleton. A skeleton is a connected set of segments, corresponding to limbs, and joints. A joint is the intersection of two segments, which means it is a skeleton point where the limb which is linked to the point may move. The angle between the two segments is called the joint angle. A joint may have at most three kinds of position angles: flexion, pivot and twisting. Skeleton animation consists in animating joint angles. There are two main ways to do that: parametric keyframe animation and procedural animation based on mechanical laws. For example, to bend an arm with parametric keyframe animation, it is necessary to enter into the computer the elbow angle at different selected times. Then the software is able to find any angle at any time using for example interpolating splines [3]. In the second approach, angles are calculated by inverse kinematics, dynamics or biomechanics.

A high-level approach consists in specifying the animation in terms of tasks. With task level control, the animator can only specify the broad outlines of a particular movement and the animation system fills in the details. In task-level animation, the animator specifies what the virtual human has to do, for instance, "bring me a glass". Task-level animation requires high-level models of human actions.
With task level control, the animator can only specify the broad outlines of a particular movement and the animation system fills in the details. In task-level animation, the animator specifies what the synthetic actor has to do, for instance, "jump from here to there". Task-level animation requires high-level models of human actions. For example, we have developed a human walking model built from experimental data based on a wide range of normalized velocities [4]. The model is based on a simple kinematics approach designed to retain the intrinsic dynamic characteristics of the experimental model. This approach also allows the definition of an individualized walking action in an interactive real-time context in most cases. Fig.2 shows a frame from the film Still Walking using this walking model.

![Still Walking](image)

**Fig.2. Still walking**

**Facial animation**
Because the human face plays the most important role for identification and communication, realistic construction and animation of the face is of immense interest in the research of human animation. The ultimate goal of this research would be to model exactly the human facial anatomy and movements to satisfy both structural and functional aspects. However, this involves the concurrent solution of many problems. The human face is a very irregular structure, which varies from person to person. The problem is further compounded with its interior details such as muscles, bones and tissues, and the motion which involves complex interactions and deformations of different facial features.
Although all movements may be rendered by muscles, the direct use of a muscle-based model is very difficult. The complexity of the model and our poor knowledge of anatomy makes the results somewhat unpredictable. This suggests that more abstract entities should be defined in order to create a system that can be easily manipulated. A multi-layered approach [5] is convenient for this. In order to manipulate abstract entities like our representation of the human face (phonemes, words, expressions, emotions), we propose to decompose the problem into several layers. The high level layers are the most abstract and specify "what to do", the low level layers describe "how to do". Each level is seen as an independent layer with its own input and output.

For our facial deformations, we have extended the concept of Free Form Deformations (FFD) introduced by Sederberg and Parry, a technique for deforming solid geometric models in a free-form manner. It can deform surface primitives of any type or degree, for example, planes, quadrics, parametric surface patches or implicitly defined surfaces. FFD involves a mapping from $\mathbb{R}^3$ to $\mathbb{R}^3$ through a trivariate tensor product Bernstein polynomial. Physically, FFD corresponds to deformations applied to an imaginary parallelepiped of clear, flexible plastic in which the objects to be deformed are embedded. The objects are also considered to be flexible so that they are deformed along with the plastic that surrounds them. A grid of control points is imposed on the parallelepiped. For any point interior to the parallelepiped, the deformation is specified by moving the control point(s) from their undisplaced latticial position.

As an extension to the basic FFDs, we provide the option of including rational basis functions in the formulation of deformation [6]. These rational basis functions allow incorporation of weights defined for each of the control points in the parallelepiped grid. The advantage of using rational FFDs is that they provide one more degree of freedom of manipulating the deformations by changing the weights at the control points.

Region build-up is accomplished by interactive selection of polygons. These regions on the human face generally correspond to their anatomical descriptions, such as nose, lips, eyes, etc. The selection process can be further hastened by attaching the color attributes of the polygons. The magnitude and direction of the muscle pulled is interactively adjusted by changing the position and weight of the control points on the control-unit of a region.

Certain points in a defined region can be set "anchored" meaning they will not deform when the deformations are applied on the region. The physical characteristics of facial mesh points can also be set interactively.
The free-form deformations correspond to the lower level of our multi-layer system SMILE where, at each level, the degree of abstraction increases. The defined entities correspond to intuitive concepts such as phonemes, expressions, words, emotions, sentences and eye motion, which make them natural to manipulate. The top level requires direct input from the animator in the form of global abstract actions and their duration, corresponding to the intuitive and natural specifications for facial animation [7]. Fig.3 shows an example.

![Fig.3. Facial animation](image)

**RENDERING VIRTUAL HUMANS**

**Skin texture**
To improve the "Barbie-like" aspect of virtual humans, we propose a technique based on texture mapping of photos of real faces. A separate tool for matching the 3D facial topology on a given picture/photo of a face is developed. Only a few feature points is selected from the 3D model to exactly match the corresponding points on the picture.

Delaunay triangulation is used to connect these points. These points can be moved and displaced on the picture interactively. An interpolation scheme in a triangular domain is used to get the desired texture coordinates. As a result the picture is deformed and mapped on the 3D model. In order to map the entire head,
multiple views are needed. These pictures will be projected on a
cylinder. Then the corresponding matching will be performed
between the cylindrical projected 3D model points and cylindrical
projected pictures. By using texture mapping the quality of
rendering improves considerably. In addition, it will allow us to
put a picture of a specific person on a given 3D model.

It is useful and interesting to be able to generate synthetic
textures of different type of skins varying in color and luminance.
Samples of skin for different regions of a face are used to generate
a synthetic skin texture. Techniques of image treatment are used
to obtain characteristic features for the samples in a group, and
each group is then assigned to a portion of a face for skin texture.

Hair modeling
In the field of human simulation, hair presents one of the most
challenging problems and therefore has been one of the least
satisfactory aspects of human images rendered to date. The
difficulties of processing hair result from the large number and
detailed geometry of the individual hairs, the complex interaction
of light and shadow among the hairs, and the small scale of the
hair width in comparison with the rendered image.

Rendering an image of hair with our system involves several
steps:

- creating a database of hair segments
- creating shadow buffers from all lights
- rendering the hairless objects using all shadow buffers
- compositing the hair on the hairless image

In the hair style rendering module, the process is made step by
step. First, the shadow of the scene is calculated for each light
source $i$, as well as for the light sources for the hair shadow. The
hair shadows are calculated for the object surface and individually
for each individual hair. Finally the hair style is blended into the
scene, using all shadow buffers. The result is an image with a
three-dimensional realistic hair style rendering where complex
shadow interaction and highlight effects can be seen and
appreciated. Fig.4 shows an example.

Clothes modeling
In most computer-generated films involving virtual humans,
clothes are simulated as a part of the body with no autonomous
motion. For modeling more realistic clothes, two separated
problems have to be solved: the motion of the cloth without collision
detection and the collision detection of the cloth with the body and
with itself. A flexible or deformable object is different from a rigid
object because it cannot be considered as a whole and its movement cannot be computed from a small set of its points. The flexible object must be divided into small parts and each point is submitted to a set of local and global constraints. These constraints create forces which prevent the violation of these constraints. Solving the dynamic system requires finding an equilibrium between all these forces. Recent research deals with dynamic models for flexible or deformable objects.

Our work in cloth animation [8] is based on the fundamental equation of motion as described by Terzopoulos et al. [9] with the damping term replaced by a more accurate one proposed by Platt and Barr [10]. When a collision is detected, we pass through the second step where we act on the vertices to actually avoid the collision. For this collision response, we have proposed the use of the law of conservation of momentum for perfectly inelastic bodies. This means that kinetic energy is dissipated, avoiding the bouncing effect. We use a dynamic inverse procedure to simulate a perfectly inelastic collision. Such collisions between two particles are characterized by the fact that their speed after they collide equals the speed of their centers of mass before they collide.

The constraints that join different panels together and attach them to other objects are very important in our case. Two kinds of dynamic constraints [12, 11] are used during two different stages. When the deformable panels are separated, forces are applied to
the elements in the panels to join them according to the seaming information. The same method is used to attach the elements of deformable objects to other rigid objects. When panels are seamed or attached, a second kind of constraint is applied which keeps the panel's sides together or fixed on objects. Fig.5 shows examples.

![Fig.5. Cloth modeling](image)

INTERACTION WITH VIRTUAL HUMANS

We should also consider the relationship between the virtual human and the rest of the world. We distinguish four basic cases:

1. the virtual human is alone in the scene, there is no interaction with other objects.
2. the virtual human is moving in an environment and he is conscious of this environment.
3. actions performed by a virtual human are known from another virtual human and may change his behavior
4. not only may the animator communicate information to the virtual human but this virtual human is also able to respond it and communicate information to the animator.

Virtual humans are moving in an environment comprising models of physical objects. Their animation is dependent on this environment and the environment may be modified by these
actors. Moreover several virtual humans may interact with each other. Several very complex problems must be solved in order to render three-dimensional animation involving virtual humans in their environment. For example, we introduced a finite element method to model the deformations of human flesh due to flexion of members and/or contact with objects [12]. The method is able to deal with penetrating impacts and true contacts. Simulation of impact with penetration can be used to model the grasping of ductile objects, and requires decomposition of objects into small geometrically simple objects. All the advantages of physical modeling of objects can also be transferred to human flesh. For example, the hand grasp of an object is expected to lead to realistic flesh deformation as well as an exchange of information between the object and the hand which will not only be geometrical.

In the context of interactive animation systems, the relationship between the animator and the virtual humans should be emphasized. With the existence of graphics workstations able to display complex scenes containing several thousands polygons at interactive speed, and with the advent of such new interactive devices as the Spaceball, EyePhone, and DataGlove, it is possible to create computer-generated characters based on a full 3D interaction metaphor in which the specifications of deformations or motion are given in real-time. True interaction between the animator and the virtual human requires a two-way communication: not only may the animator interact to give commands to the virtual human but the virtual human is also able to answer him. Finally, we may aspire to a virtual reality where virtual humans participate fully: real dialog between the animator and the virtual human. The animator may now enter in the synthetic world that he/she has created, admire it, modify it and truly perceive it. Finally, computer-generated human beings should be present and active in the virtual world. They should be the synthetic actors playing their unique role in the theater representing the scene to be simulated.

COMMUNICATION BETWEEN THE VIRTUAL HUMAN AND THE ANIMATOR

For the communication between the animator and the virtual humans, we are developing a prototype system. As shown in Fig. 6, this system is mainly an inference system with facial and gesture data as input channels and face and hand animation sequences as output channels.
The development of the inference system is divided into three subsystems:

i) a subsystem for the recognition of emotions from facial expressions, head-and-shoulder gestures, hand gestures and possibly speech

ii) a subsystem for the synthesis of facial expressions and hand motions for a given emotion and speech

iii) a subsystem for the dialog coordination between input and output emotions

This last subsystem is a rule-based system: it should decide how the virtual human will behave based on the behavior of the real human. The dialog coordinator analyzes the humor and behavior of the user based on the facial expressions and gestures. It then decides which emotions (sequences of expressions) and gestures (sequence of postures) should be generated by the animation system. For the design of correspondence rules, our approach is based on existing work in applied psychology, in particular in the area of non-verbal communication.

For the recognition of emotions, our method is based on snakes as introduced by Terzopoulos and Waters [13]. A snake is a
dynamic deformable 2D contour. A discrete snake is a set of nodes with time varying positions. The nodes are coupled by internal forces making the snake acting like a series of springs resisting compression and a thin wire resisting bending. The expression forces are introduced into the equations of motion for dynamic node/spring system. Our approach is different from Terzopoulos-Waters approach because we need to analyze the emotion in real-time. Instead of using a filter which globally transforms the image into a planar force field, we apply the filter in the neighborhood of the nodes of the snake.

We only use a snake for the mouth; the rest of the information (jaw, eyebrows, eyes) is obtained by fast image-processing techniques. For the jaw, we consider that the lower part of the lower lip (using information given by the snake) is moving with the jaw i.e. if the jaw opens, the lower part goes down with the jaw. For the nose, we use the center point of the upper part of the mouth (also using the snake) and we scan upwards until an edge is detected. As we assume that the illumination is very strong, the edge should belong to the shadow of the nose. For the eyebrows, we use the same principles as the nose. We start from the forehead and scan downwards until we detect an edge. This should be the eyebrow. For the eyes, we define a rectangular region around the eyes (using the position of the nose and eyebrows) and we count the number of white points in the region. If the number of white points is under a threshold value, we consider the eye as closed. Fig.7 shows an example.

Fig. 7. Animator and virtual human
CONCLUSION

Modelling humans using computers is a very complex task. Several important problems have to be solved to incorporate realistic virtual humans in computer-generated films. However, in a few years, we will be able to recreate any person, living or dead, mix up the past with the present and the future and make this person talk and have emotions.

REFERENCES


