Making H-Anim bodies

Won-Sook Lee, Taro Goto, Nadia Magnenat-Thalmann
MIRALab, CUI, University of Geneva, Geneva, Switzerland
wslee@email.com, goto@miralab.unige.ch, thalmann@miralab.unige.ch
http://www.miralab.unige.ch

> *As the 3D Internet continues to grow, there will be an increasing need to represent human beings in online virtual environments. Achieving that goal will require the creation of libraries of interchangeable humanoids, as well as authoring tools that make it easy to create new humanoids and animate them in various ways. (...) This standard will allow humanoids created using authoring tools from one vendor to be animated using tools from another.*
>
> *Humanoid Animation Group[11] ("h-Anim" for short) for the sole purpose of creating a standard representation for humanoids*

This paper is focused on how to create new standard Humanoid H-Anim bodies. In addition, it treats not only an animatable virtual bodies but also photo-realistic virtual twins of us. These virtual twins will populate the virtual worlds acting in different situations. It will be as if we would like to deal with kind, affordable and smart humans that will be at our disposal at any time with a lot of knowledge and help us to understand, to learn and to experience situations.

Key words: *Feature based approach, 3D Deformation, Texture mapping, Edge detection, 3D morphing, Animation, H-Anim, MPEG-4.*

1 Introduction

As H-Anim with the close relation with MPEG-4 is the new standard for representing virtual human and H-Anim compatible bodies are able to be visualized in any VRML97 compliant browser, to make an H-Anim compatible virtual humans is very important now. There are various approaches to make animatable virtual humans. Compared to face part [1][2][5][6][8][12][17][18][20], the body part has been less explored by researchers and is becoming increasingly popular. There are many common methods, which can be applied to both on the face and on the body, but we have to consider the difference between them. The difference between face and body is that firstly the body part is much bigger and secondly body has skeleton with several skin parts related to skeleton. There are some methods that concern precision and accuracy while others concern visual realism.

Laser scanner: To capture the intricacies of the human body in one pass, the *Cyberware*^{TM} Whole Body scanner [10] uses four scanning instruments mounted on two vertical towers. It has laser, CCD sensor and camera for texture capturing as well as software to merge four huge meshes from the instruments. A primary goal is to acquire as complete a model as possible in one pass. The use of multiple instruments improves accuracy on the sides of the body and in difficult-to-reach areas, such as under a person's arms. Since recent laser range finder becomes very good, the key point is the software to compensate hole problem, calibration of the texture image from different camera and polygon reduction functionality.

Silhouette in multiple views on video: Kakadiaris and Metaxas [13][14] present a novel approach to the three-dimensional human body model acquisition from three mutually orthogonal views, the shape is not their main concern though. It uses silhouette information to catch the movement of human body. Their technique is based on the spatio-temporal analysis of the deforming apparent contour of a human movement according to a protocol of movements. The technique does not use a prior model of the human body and prior body part segmentation is not assumed.

Silhouette in multiple views on photographs: Gu and et al. [7] concentrate on human body modeling for the purpose of measurement of human body for the clothes industry. They use a hexagonal supporting framework to form a closed space for imaging with 12 cameras for upper and lower body parts in six views. Additionally a slide projector with a grid pattern is used to catch the chest area using the stereo pair of the intersections of horizontal and vertical lines. The silhouette is extracted and some attached markers are reconstructed to build a feature curves of the final body. Weik et al. [21] creates flexible anthropomorphic models from multiple views by extracting silhouette. Then from different views, points of each volumetric cones are constructed, whose intersections in 3D form the final approximation of the volume model. The predefined skeleton and a flexible triangle surface mesh are adapted to the point data. The environment is again limited by homogeneous background and they treat only the upper body. Hilton et al. [9] proposed a method for capturing of realistic whole-body animated models of clothed people using model-based reconstruction from multi-view color images taken in an environment with a specially prepared background and properly controlled lighting. The images are used to morph a 3D generic humanoid model to have the shape and appearance of a specific person. It is a low-cost whole-body capture of clothed people from digital color images. The approach is simple and efficient, but limited to body part without having delicate face usable for facial animation

Stereo-video: Plänkers et al. [19] uses video cameras with stereo pair for the body part. A person’s movements such as walking or raising arms are recorded to several video sequences and automatically extracts range information using stereo and tracks an outline of body. This 2-D and 3-D information is fed to an optimizer that fits to the data.

We address here how to acquire an animatable human data with a realistic appearance. The issues involved in realistic modeling a virtual human for the real-time application purposes are as follows:

- acquisition of human shape data
- realistic high-resolution texture data
- functional information for animation of the human

It is our goal to develop a technique that enables an easy acquisition of the H-Anim avatar model having the ability to be produced at a low cost and animated properly. There are many possible methods to acquire input data for modeling from high-end laser scanner to low-end still photographs. Each of them has own advantage and disadvantage. We select to use photographs as main input. For the photographs input, we will use a frontal view, a side view and a back view. There are possibilities to build a special booth in a spacious room with a special setting. However in this case, it gives a great limitation to the potential places and users. We use simple snapshots with commercial cameras without any special environment setting. Instead of seeking a solution by limiting environment, we provide a user-friendly interface, which allows non-expert user to interactively hint to the system certain important information about the human body.

Our method is the feature-based modeling of animatable human body, so called virtual cloning or virtual-twins. The basic idea is to consider a virtual actor as a combined set of data, including the 3D shape but also the structure to animate the shape inside a virtual environment. The human modeling approach described starts from default virtual human templates (here we use H-Anim 1.1 format), including shape and animation structures, and modifies the shape to create a new virtual actor. Shape modifications are transmitted to the attached animation structures in order to keep the set of data consistent. We get an H-Anim “ready to animate” virtual humans as the output of the modeling process. Our approach optimizes the modeling of virtual humans in order to animate him/her. Feature based approach is useful to make easy input in robust way and to take the animation structure while the shape resembles the input shape.

The methodology is composed of two major components: face-cloning and body-cloning. The detailed information for the face-cloning is found in another literature [17]. The body-cloning component in section 2 uses firstly feature points specification, which enables automatic silhouette detection in an arbitrary background, and then two-stage body modification by using feature points and body silhouette respectively. Section 3 shows some post processing needed to fit to the H-Anim 1.1 format and the final integrated human model with cloned face and body together has photograph-realistic H-Anim humanoid. Section 4 shows some examples of the animation. The result can be visualized in any VRML compliant browser.

2 Body Cloning

We present a full body cloning methodology in this section. This system utilizes photos taken from the frontal, side and back of a person in any given imaging environment without requiring a special background or a controlled illuminating condition. A seamless generic body specified in the VRML H-Anim 1.1 format is used to generate an individualized virtual human. The generic body has the level of articulation three with full H-Anim hierarchy with 94 skeleton joints and 12 skin parts including head, hands and feet. As a reference, there is an option in H-Anim 1.1 format we can select a subset of 94 skeleton joints. Each skin part is related to a skeleton joint as a segment node and is saved in local coordinates. The *n* skin parts have the corresponding skeleton joints, which transform the local coordinates of the skin part *i* to global coordinate by 4x4 matrix M_i for given *i* where *0 £ i £ n*.

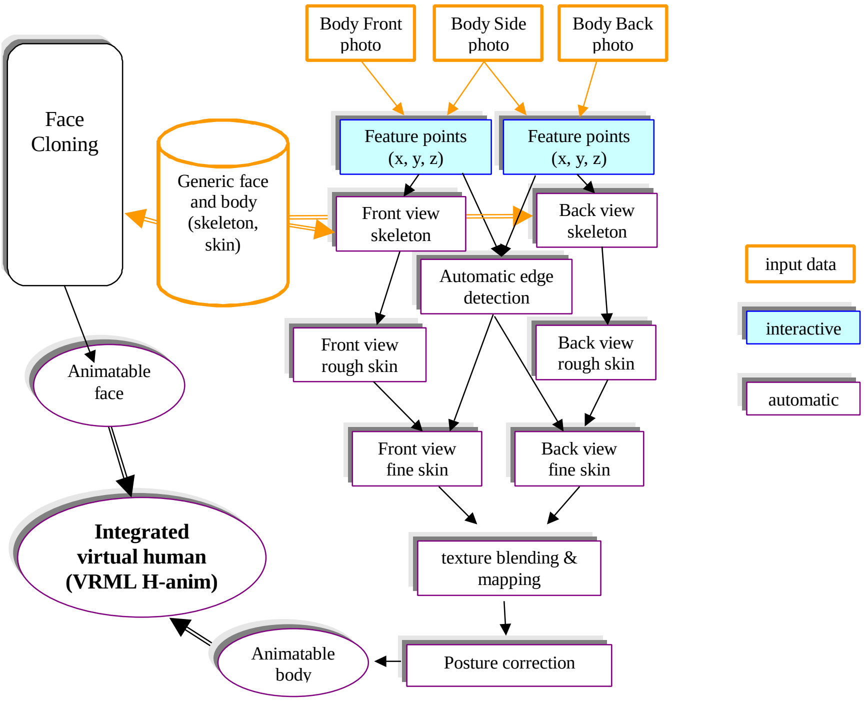


Figure 1: Overflow of body cloning.

2.1 Initial skin model using features

We focus on the simplest environment to take photos with only one camera. We take three photos, from the front, the side and back. In this case, the frontal and back views are not exact reflections of each other since we asked the person to rotate for the back view after taking the frontal view. We input the height (a unit - the meter as H-Anim definition), of the person and the image body heights are checked on the three images for normalization. Since we use arbitrary background to take photos and the size of the person is not fixed, we use interactive feature points localization on images. Here feature points are the ones we can notify the location easily from photographs such as the armpit, wrist, waist, hip, ankle, end-shoulder points, and so on. These simple feature point localization is used for skeleton modification, rough skin deformation and fully automatic edge detection later.

When we have feature points for the skin envelope (although the person put on clothes, we assume that the clothes outlines are close to the skin outlines), we can have an estimation of the skeleton. For example, the feature point correspondent to the right elbow must be located around middle position between the right end shoulder point and the outer end of the right wrist. Here we applied *piecewise affine transformation* and Barycentric interpolation to find the typical skeleton joints from feature points. Since there are 94 skeleton joints, we make a subset of joints as movable joints, which are modified by feature points while others are modified by piecewise affine transformations defined by movable joints and the skeleton hierarchy.

Then each skin part *i* is connected to a skeleton joint by a matrix M_i to be in global coordinates. We update the matrix by scaling, translation and rotation defined by the corresponding skeleton joint and the child skeleton joints. Then adjacent skin parts are not guaranteed to be continuous. The shoulder parts are usually overlapping with torso parts. Therefore, we define a freeform deformation to make a rough matched continuous body with feature points information. The control points are placed at certain required positions to represent the shape characteristics. Hence the skin model can be deformed by moving these control points, which is designed such as they have corresponding points on the frontal (or back) and the side view images, or the locations are found with certain relations. For example the boundary between the '*front_torso*' and '*right_upper_arm*' can be found from the feature points on the bottom-right corner point of the *neck* and on the right end shoulder point on the images. Furthermore, several control points are located at the boundaries between two parts, so that surface continuity is preserved when the posture of the generic body is changed. These control points are used for the *piecewise affine transformation*. We apply separate deformations for each skin part.

2.2 Fine modification using edges

The accuracy of the initial skin model is insufficient, so we modify it by using the body silhouette extracted from the pictures. Here we detect the silhouette of the human body on the pictures and then fit the skin model to the silhouette. We design a simple algorithm by making use of the feature points on the body, which serves as the heuristics for the body silhouette extraction.

2.2.1 Heuristic based silhouette extraction with any background

There are plenty of literatures available about boundary extraction or edge linking [4][15]. It can be treated as a graph-searching problem, as an optimization problem, or as an energy minimization and regularization procedure. However, these algorithms are usually inefficient due to the need for backtracking or exhaustive search. In addition, the algorithms need time to reach convergence or stable result, such as the snake algorithm. We design a simple algorithm by making use of the feature points on the body. The feature points serve as the heuristics for the body silhouette extraction.

1) First, Canny edge detector is applied to every image. Then a coloring-like linking algorithm is used to link the edgels (edge pixels) into connected segments. Due to possible noise caused by the background, the edgels generated by the background sometimes are connected to the body edgels. To avoid this potential wrong connection from occurring, the extract segments are split into short line segments. The short line segments are called as *edge segments*, and the line segments formed by consecutive feature points as *feature segments* to make the discussion easier. The *edge segments* include *feature segments*.

2) After the Canny edge detector and linking step, *edge segments* generated by the object as well as by the background are evaluated to be a proper segment. We first throw away those lying outside the vicinity of any *feature segments*. The goal now is, for each *feature segment*, to find a path that is formed by an ordered set of *edge segments* within its vicinity.

3) Weak segments remained after the vicinity check, are also removed using general body shape information to avoid abnormal edge detection. Since edge strength of the image is different for each image, proper edge will not appear in a detector of constant value. Thus, weak edge will make a noise, and it will be the obstacle to make a clear line. The weak *edge segments* compared with other segments should be removed. To remove these weak segments, dynamic deletion is done. The temporal line made from the start point to the end with only strong edge segments is created, at the beginning. Then the number of edges are increased to make the connecting line contains enough *edge segments*.

4) To link the *edge segments* into meaningful boundaries, we first look for the admissible connection for each *edge segment.* We define a connection between *edge segment* $\boldsymbol{S_1} = P_{12}-P_{11}$ and $\boldsymbol{S_2} = P_{22}-P_{21}$ as *admissible* if :

$$\boldsymbol{S_1} \cdot \boldsymbol{S_2} >= 0.0,\ a_1 < T_{sm} \text{ and } a_2 < T_{sm}$$

where P_{11}, P_{12}, P_{21} and P_{22} are the ends of the two segments under consideration; a_1 is the angle between $\boldsymbol{S_1}$ and the potential connecting segment $\boldsymbol{C_{12}} = P_{21}-P_{12}$, a_2 is the angle between $\boldsymbol{C_{12}}$ and $\boldsymbol{S_2}$, T_{sm} is the maximum angle allowed for the connection.

5) Next, in order to select the most desired connection, we define a function G^L to evaluate the "goodness" of the potential connection (see *Figure 2*):

$$G^L(\boldsymbol{S_1},\boldsymbol{S_2}) = cos(a_1)*cos(a_2)*(1+cos(a_1)*cos(a_2)*log(M_2))/(w*D_1+(1-w)*D_2)$$

where D_1 is the length of the $\boldsymbol{C_{12}}$ and D_2 is the distance from the P_{22} to the *feature segment* $\boldsymbol{L}$, M_2 is the edge magnitude of edge segment $\boldsymbol{S_2}$, w is the weight of D_1 taken as a constant in our experiments. The rationale behind this definition is that we always favor a connection that contributes to a smooth path running along $\boldsymbol{L}$, formed by segments with strong edge magnitude.

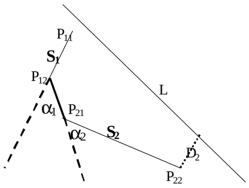


Figure 2: Link two edge segments

Thus based on G^L, we find, for the two ends of each *edge segment*, its best connection that maximizes G^L among all of its neighboring *edge segments*. Now we are ready to build the path. Starting from each *edge segment*, we connect it to its two best connections computed by G^L to form a partial path. For the *edge segments* sitting on the *head* and tail of this partial path, we connect their open ends to their respective best connections. This procedure is repeated until there is no further connection possible. So a path $\boldsymbol{P}$ consists of a sequence of *edge segments* $\boldsymbol{S_i}$, $i=1,2,..., N$.

6) Now we need another evaluation function to assess the "goodness" of a path. We define a function G^P as:

$$G^P(\boldsymbol{P}) = \sum M_i/(w_1{*}DT_1+w_2{*}DT_2+w_3{*}\sum D_i(1+\sin(\alpha_{i,1})+\sin(\alpha_{i,2}))$$

where $\sum M_i$ is the summation of the edge magnitudes, DT_1 is the distance between the path ends to the ends of the *feature segment*, and DT_2 is the average distance from the pixels on the path to the *feature segment*. D_i, $\alpha_{i,1}$ and $\alpha_{i,2}$ correspond to D_1, α_1 and α_2 in *Figure 2*, respectively. w_1, w_2, w_3 are the weights of each measurement (here we take the value *0.2*, *0.2* and *0.8* respectively).

Among all the paths found, the one $\boldsymbol{P}^*$ maximizing G^P is selected, i.e. $\boldsymbol{P}^* = arg\ max_{\boldsymbol{P}}\ (G^P(\boldsymbol{P}))$. This edge detection is not real-time and the calculation time depends on the size of the images.

2.2.2 Detailed skin modification from edges

An exact silhouette is detected by the proposed method as shown in *Figure 6* (a). After the skeleton adjustment and rough skin modification in the previous sections, the skin parts have been adjusted into proper orientation and rough size. Then this silhouette modifies the contours defining the skin surface. It can be confirmed that a visually real 3D human body model can be constructed by our method without using special environment or equipment. More detailed description can be found in another literature [16].

2.3 Texture mapping

We use only the frontal and back views for texture mapping since the two views (not the side view) are enough to cover the whole body except for the head. For the texture mapping, we give texture coordinates to points on skin envelope. Since there are two images used, we have to make a partition of the skin envelope polygons, either to the frontal view or to the back view by checking the cross product of the vertex normal with the viewing vector. To get the texture coordinates, we use a projection onto the *XY* plane in the image space. The process is as follows.

1. Deform the body with the back and side views;

2. Project back/frontal+back viewpoints onto the back view image plane to get the texture coordinates;

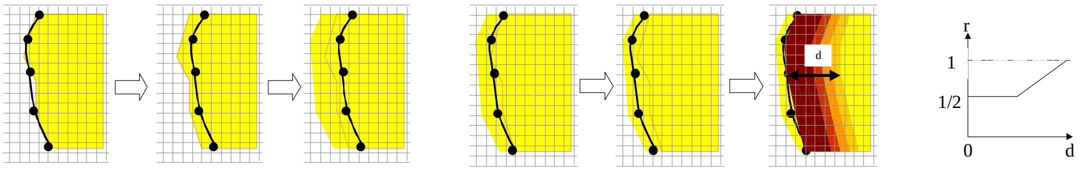
3. Deform the body with the frontal and side views;

4. Project frontal/frontal+back viewpoints onto the frontal view image plane to get the texture coordinates.

Then the proper texture mapping on both the frontal view and the back view are applied to the individualized body. However, there are still some errors on the boundary between the frontal and back views caused by digitization process and different illumination between two images. We solve this problem in two steps as follows.

1. Extension of the interior pixels on frontal and back views within certain neighborhood; Since from the previous edge detection processing, we have already known which side of the boundary is background. We search along the perpendicular direction to the edge for a foreground pixel and take its color as the color for the edge neighborhood pixels. Note it is always possible to have errors for the edge pixel location. So we move the edge to a few pixels inward position temporarily before the extension of pixels on the edge as shown in *Figure 3* (a). As result shows, this simple processing removes the noisy effect of the digitization process.

2. Blending between the frontal and back texture to remove the difference between the two images; From the feature points given in the previous processing, we can recognize semantically the various body parts, hence establish the part correspondences between the two images. We further find the pixel correspondence according to the boundary lengths. Within the neighborhood of the two pixels from frontal and back view images respectively, we use a piecewise linear blending. For example for the frontal view, $p(d) = r* p^F(d) + (1.0 - r)*p^B(d)$ with r shown in *Figure 3* (c). Note it is always possible to have errors for the edge pixel location. Therefore, before mixing of pixels on the edge, we move the edge to a few pixels outward position temporarily as shown in *Figure 3* (b).



(a) (b) (c)

Figure 3: (a) extension of the interior region for digitalization error. (b) the texture blending between the frontal view and back view. (c) piecewise linear mixing ratio of pixel color value where d is distance from the edge pixels and r is mixing ratio between the frontal and back views.

3 Post processing and Results

3.1 Elbow skeleton correction

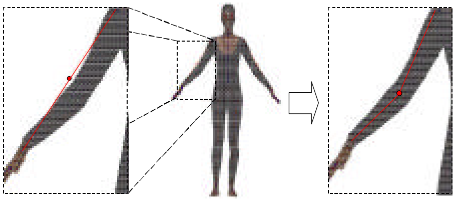


Figure 4: A problem on elbow for people with bent arms.

For some people, the elbow is bent and it creates the skeleton on the elbow region is visible coming out of the skin as shown in the left side of *Figure 4:*. The problem comes from the way to construct skeleton. The skeleton is constructed using feature points and we construct the elbow skeleton joint using shoulder and wrist feature points without using elbow points from photographs since they are not considered as feature.

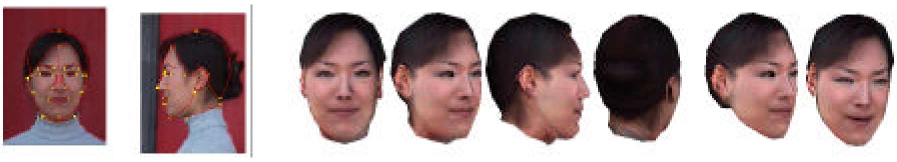
To correct the problem, we adjust the skeleton with silhouette information. So we move the elbow skeleton on the center (it can be positioned with proper biomechanical skeleton position with skin envelope) of the skin envelope of elbow that is a contour in our data structure. Since the skin parts have the corresponding skeleton joints by 4x4 matrix $\boldsymbol{M_i}$, we have to adjust the local coordinates of skins of *upper_arm* and *lower_arm*. Then the skin parts keep the same global position although the position of elbow is changed. So we calculate the $\boldsymbol{M'_i}$ with updated position of elbow skeleton. Then to get the corrected coordinates $\boldsymbol{P'_i}$ of the coordinates $\boldsymbol{P_i}$ of points on the skin, we apply the following equations.

$$\boldsymbol{P'_i = P_i\, M_i\, (M'_i\,)^{-1}}$$

The resulting skeleton is shown in the right side in *Figure 4*.

3.2 Connection between body and face

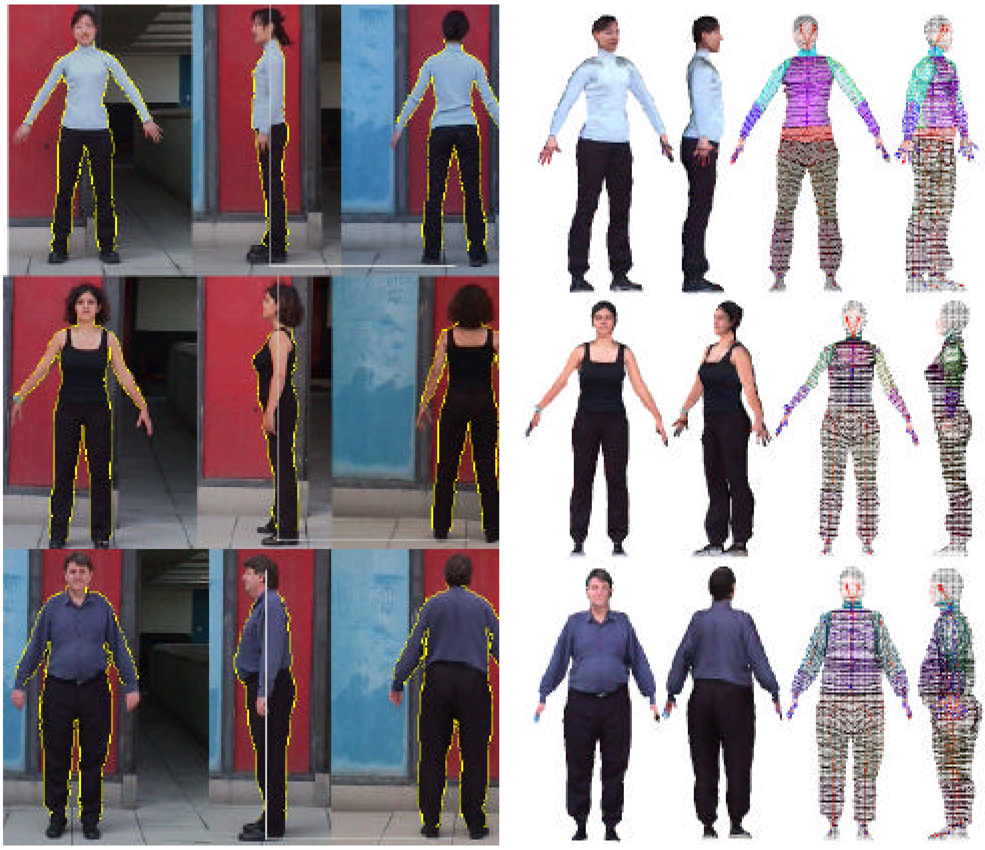
We processed face cloning [17] and body cloning separately. Even though the size of the face is much smaller, we have to keep a detailed structure and high resolution for a face since we often zoom in the face to see facial animation and communication. So we use separated images for face cloning and body cloning and these two cloning methods use different texture mapping schemes. *Figure 5* (b) shows several views of the final reconstructed *head* out of two photographs in *Figure 5* (a). When we connect this *head* with a body, we remove the *neck* since the *neck* is from the body due to the body skeleton animation for face rotation. The face animation is immediately possible as being inherited from the generic *head*. See the last face in *Figure 5* (b) to see an expression on a face. When we have a face and a body reconstructed separately, we have to connect them properly to make a smooth envelope for a perfectly smoothed body. We check the face size and location of the face on the frontal and side view body images using specific feature points. Then simple translation and scaling locate the individualized face on the individualized body and then we use an automatic sticking between them by the database built when we load the generic body where the nearest points on the neck and the head are found. The neck from the body is deformed using *piecewise affine transformation* using the edges from the head and from the body.



(a) (b)

Figure 5: (a) Two input photographs focused on the face. (b) Snapshots of a reconstructed head in several views and an example of animation on the face. The final two images shows the neck removed from the face model.

The final example bodies are shown in *Figure 6* (b).



(a) (b)

Figure 6: (a) Images with super-imposed silhouette (b) Final H-Anim 1.1 bodies with detailed individualized faces connected to the bodies. They are modified from one generic model.

3.3 H-Anim default posture

The H-Anim default posture is fixed to be able to animate. Our photographs as input have posture a bit different from the defined default posture for the reason of automatic edge detection. Therefore, after construction of the individualized virtual body, we correct the posture by modifying the skeleton joint angles for arms and legs.

3.3.1 Joint angles for skeleton

The skeleton joint coordinates in H-Anim are based on global coordinates, which means the location is indicated by *(x, y, z)* with origin *(0, 0, 0)* between two feet. This is a convenient criterion for modeling purposes, but it does not provide the direct way to animate skeleton joints where the only angles need to be updated without changing the skeleton segment length. There are three choices to represent coordinates of skeleton joints.

1) Global origin and rotation
2) Local origin on the parent joint and global rotation
3) Local origin with rotation, so called moving frame

The third method is the most proper way for animation purposes since, for instance, the arm can be bend only by updating 'flexion' value regardless the shoulder has translation or rotation. Since we do not need to deal with sophisticated animation to correct postures, we take the second method. So to correct the posture from legs and arms open posture to close posture, we change the representation of the coordination of skeleton joint from *(x, y, z)* originated on *(0, 0, 0)* to *(r, f, q)* originated on the parent's skeleton joint position by

$$x \neg x - x_{parent} \qquad y \neg y - y_{parent} \qquad z \neg z - z_{parent}$$

$$r = \sqrt{x^2 + y^2 + z^2}$$

$$x = r \sin f \cos q \qquad y = r \sin f \sin q \qquad z = r \cos f$$

Once we have the representation in *(r, f, q)* originated on the parent's skeleton joint position, it is possible to correct the angle of legs and arms. We set angles for *r_elbow*, *l_elbow*, *r_knee* and *l_knee* to be the default straight down position. Since this representation is not the moving frame, we have to correct all children joints with the difference between the original joint angles and updated joint angles. *Figure 7* (a) and (b) show the corrected skeleton into H-Anim default posture.

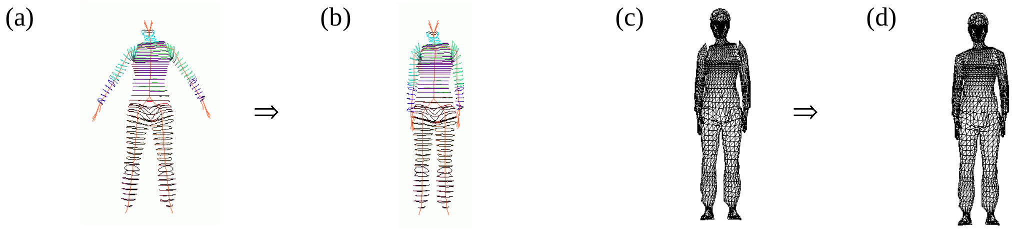


Figure 7: (a) to (b) shows correcting the joint angles for legs and arms. (c) to (d) shows correcting the skin envelope.

3.3.2 Skin adaptation

A modification of skeleton without applying skin deformation may create critical problems, especially for the shoulder part. See *Figure 7* (b) and (c) for the separated shoulder envelope. Since we do not have data of the person in an H-Anim default posture, the simplest solution is the linear interpolation between mid-shoulder point on *torso* (See *Figure 7* (c)) and end-shoulder point on *upper_arm* (See *Figure 7* (d)). More sophisticated skin connection is also possible by using splines such as *Bezier* splines or *Hermite* splines. The mid-shoulder point, end-shoulder point and the point in the second next contour on the *upper_arm* after end-shoulder point can be control points as shown in *Figure 8*. Then the shoulder part has a smoother surface.

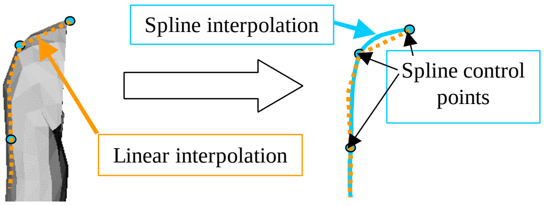


Figure 8: Spline connection for shoulder part.

For the *upper_leg* skin parts, simple connection with hip skin part is adaptable.

4 Body animation

Since the generic body had H-Anim 1.1 structure, the individualized bodies keep the same structure, which means we can animate the bodies immediately. The final bodies are exported into VRML format and can be loaded in public web browsers. *Figure 9* shows an animation example of cloned body in a virtual environment [3]. Since we are using seamless skin structure for a real-time animation, the skin and textures are smoothly connected during animation.

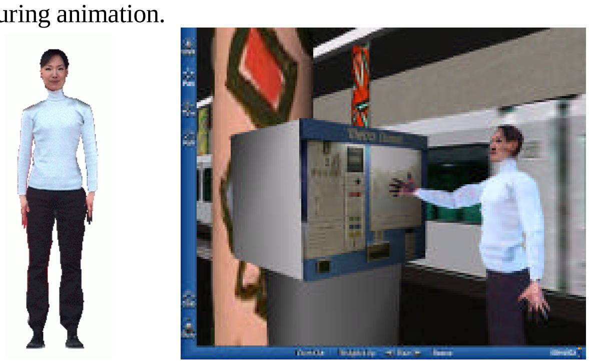


Figure 9: H-Anim humanoid in default posture and animation of a virtually cloned human in a virtual environment.

Since we keep the standard H-Anim format for cloned bodies, we can use any animation system if it is able to animate H-Anim bodies. We experiment another system to show another animation example of cloned bodies by realizing a complete pipeline for making clone of a person from the shape to movement. The pipeline includes main three steps such as (i) to clone a person from five photographs, (ii) to capture the skeleton movement of the persons using an optical motion capture system, (iii) to visualize the realistic movement of the cloned virtual human with deformation. For the motion capture, we have chosen a dancing motion, which is typically non-trivial movement to synthesize. Six cameras whose sample rate is 120Hz are used along with 25mm markers. *Figure 10* shows some snapshots of a video clip of a person, which is composed with the same camera view in the real world and in the virtual world.

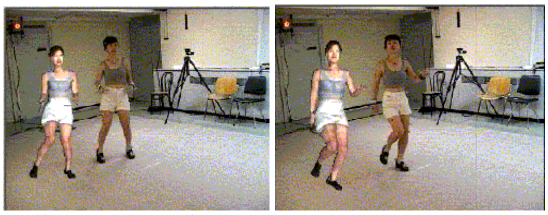


Figure 10: Another example of simulation of a human shape and motions with the same camera views in the real world and in the virtual world.

5 Conclusion

We have described the methodology and algorithms implemented for computer-aided animatable human modeling in H-Anim format. We have developed a body cloning system from photograph data. Easy input like photographs is the first priority to build the system and even the images sent by email are successfully processed to make the virtual-twin and animation. In addition continuous mesh for generic body is used for individualization and successfully implemented resulting seamless real-time animatable avatars. International standard H-Anim 1.1 for the body-cloning program is considered and the result is successfully produced in the standard format.

As the virtual human has wide applications from TV show, computer generated film production to Internet, we need to support several level of details (LOD) adapting to each application. For example, on the Internet application, lower polygonal number will accelerate the process while film production needs high resolution without considering the speed much. LOD will be also useful for real-time animation system, which will control the polygonal number depending on the distance between the camera and the virtual human shown. To support LOD is our ongoing topic.

6 Acknowledgements

The authors would like to thank to Laurent Moccozet, Nabil Sidi-Yacoub, Hyewon Seo and Jin Gu. This project is supported by Swiss National Research Foundation.

7 Reference

[1] Akimoto T., Suenaga Y. and Richard S. W., "Automatic Creation of 3D Facial Models", IEEE Computer Graphics & Applications, September, IEEE Computer Society Press, 1993.

[2] Arai K., Kurihara T., Anjyo K., "Bilinear interpolation for facial expression and metamorphosis in real-time animation", Visual Computer, Springer, Vol. 12, No. 3, 1996.

[3] Babski C., Thalmann D., "Realtime animation and Motion Capture in Web Human Director(WHD), Proc. Web3D & VRML2000 Symposium, 2000.

[4] Ballard D. H. and Brown C. M., Computer Vision, pp. 131-147, Prentice-Hall InC., 1982.

[5] Blanz V. and Vetter T. "A Morphable Model for the Synthesis of 3D Faces", Computer Graphics (Proc. SIGGRAPH'99), ACM Press, pp. 187-194, 1999.

[6] Fua P., "Regularized Bundle-Adjustment to Model Heads from Image Sequences without Calibration Data", International Journal of Computer Vision, Kluwer Publisher, In Press.

[7] Gu J., Chang T., Gopalsamy S., and Shen H., "A 3D Reconstruction System for Human Body Modeling", In Modelling and Motion Capture Techniques for Virtual Environments (Proc. CAPTECH'98), (Springer LNAI LNCS Press), pp. 229-241, 1998.

[8] Guenter B., Grimm C., Wood D., Malvar H., Pighin F., "Making Faces", Computer Graphics (Proc. SIGGRAPH'98), ACM Press, pp. 55-66, 1998.

[9] Hilton A., Beresford D., Gentils T., Smith R. and Sun W., "Virtual People: Capturing human models to populate virtual worlds". In Computer Animation (Proc. Computer Animation'99), IEEE Computer Society Press, pp. 174-185, 1999.

[10] http://www.cyberware.com/wb-vrml/

[11] http://www.H-Anim.org

[12] Ip H. H.S., Yin L., "Constructing a 3D individual head model from two orthogonal views", The Visual Computer, Springer, 12:254-266, 1996.

[13] Kakadiaris I. A. and Metaxas D., "3D Human Body Acquisition from Multiple views", in Proc. of the Fifth ICCV, IEEE Computer Society Press, pp. 618-623, 1995.

[14] Kakadiaris I. A. and Metaxas D., "Model-based estimation of 3D human motion with occlusion based on active multi-viewpoint selection", In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, IEEE Computer Society Press, pp. 81-87, San Francisco, CA, June 18-20 1996.

[15] Lai K.F. and Chin R.T., "Deformable contours Modeling and extraction", IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Computer Society Press, vol. 17, no.11, pp. 1084-1090, 1995.

[16] Lee W.-S., Gu J., Magnenat-Thalmann N., "Generating Animatable 3D Virtual Humans from Photographs", accepted in Eurographics 2000, Volume 19, Number 3, Computer Graphics Forum, Blackwell publisher, 2000.

[17] Lee W.-S., Magnenat-Thalmann N., "Fast Head Modeling for Animation", Journal Image and Vision Computing, Volume 18, Number 4, pp.355-364, Elsevier Sceince, 1 March, 2000.

[18] Lee Y., Terzopoulos D., and Waters K., "Realistic Modeling for Facial Animation", In Computer Graphics (Proc. SIGGRAPH), ACM Press, pp. 55-62, 1995.

[19] Plänkers R., Fua P., D'Apuzzo N., "Automated Body Modeling from Video Sequences" in Proc. IEEE International Workshop on Modelling People (mPeople), IEEE Computer Society Press, Corfu, Greece, September, 1999

[20] Proesmans M., Van Gool L. "Reading between the lines - a method for extracting dynamic 3D with texture", In Proceedings of VRST'97, ACM Press, pp. 95-102, 1997.

[21] Weik S., Wingbermühle J., Niem W., "Automatic Creation of Flexible Antropomorphic Models for 3D Videoconferencing", Proceedings of Computer Graphics International CGI, IEEE Computer Society Press, June 22-26, Hanover, Germany, 1998