To appear in: Proceeding of The 7th ACM International Multimedia Conference (Multimedia'99). Orlando, Florida, USA. October 30-November 5, 1999.

Video Based Human Animation Technique

Xiaoming Liu Yueting Zhuang Yunhe Pan

Institute of Artificial Intelligence, ZheJiang University
HangZhou, 310027
P. R. China
86-571-7951853

Liuxm@icad.zju.edu.cn Yzhuang@icad.zju.edu.cn Panyh@sun.zju.edu.cn

ABSTRACT

Human animation is a challenging domain in computer animation. To aim at many shortcomings in conventional techniques, this paper proposes a new video based human animation technique. Given a clip of video, firstly human joints are tracked with the support of Kalman filter and morph-block based match in the image sequence. Then corresponding sequence of three-dimension (3D) human motion skeleton is constructed under the perspective projection using camera calibration and human anatomy knowledge. Finally a motion library is established automatically by annotating multiform motion attributes, which can be browsed and queried by the animator. This approach has the characteristic of rich source material, low computing cost, efficient production, and realistic animation result. We demonstrate it on several video clips of people doing full body movements, and visualize the results by re-animating a 3D human skeleton model.

Keywords

Human animation, video, feature point, kalman filter, camera calibration, skeleton, motion library.

1. INTRODUCTION

Computer technique has entered many domains of society and become the focus of people's attention. In the area of art, computer has begun to aid animator and given birth to computer animation since 1960's. For decades of years, researchers have been exploring an easy, effective way to fabricate computer animation for its wide application prospective. Now, no matter whether in children cartoon or in films such as TIANTIC, people can experience the realistic vision flavor, which is brought by computer animation. The representation of human body and its motion is the most challenging domain in computer animation. This paper will propose a new video based human animation technique.

1.1 Conventional techniques

The conventional human animation is also named as joint animation where the joint motion can be controlled by forward kinematics or inverse kinematics. Forward kinematics can acquire the position of several related body parts by specifying the joint rotation angle. Inverse kinematics can compute the position of middle joints by specifying the ending joint position. Although it is easier than the former, its solution cost will be expensive with the increase of complexity. We can see that not only the fussy work of animator, but also a large computation cost is needed in joint animation. Because this approach is just a simulation of real human motion, it always lacks of reality.

In contrast to joint animation, the motion capture based animation is becoming increasingly popular. An actor performs the desired motion, and a set of devices is used to record the body's joint configuration. This data is mapped onto a 3D human model stored in the computer. Though this technique can result in more realistic human motion than the former, the process of motion capture usually costs much time and money. Sometimes it should be done inside a studio. Sometimes magnetic sensors are wired to the actor, which greatly restricts the free movements and results in unrealistic motion.

1.2 Our approach

With regard to many shortcomings in conventional techniques, we try to bring forward an efficient, economical and motion unrestricted human animation technique. Inspired by the research of motion analysis in computer vision, a video based human animation technique is proposed in this paper. Given a clip of video, we can acquire 3D human motion information and construct an entry of motion library. Then an animator can utilize the existing motion information to re-animate the data or create new data. In particular, we will show how to:

- Acquire the sequence of 2D human motion skeleton by tracking human joint with the support of motion prediction and the color model of body part.
- Use the correspondences between 3D model and 2D image to calibrate camera, and construct the sequence of 3D human motion skeleton under the perspective projection using the pinhole model and the human anatomy knowledge.
- Establish motion library by annotating multiform motion attribute automatically.

The architecture of video based human animation is shown in figure 1. The contents in three dashed boxes are the focuses listed above. In contrast to motion capture based animation, our approach does not require any markers, or sensors to be attached to human joint, which ensures the free motion of human. Except for the video recording, it does not cost anything. It is easy and straightforward from a user's point of view. In order to get the sequence of 3D human skeleton, the animator only needs to mark the joint in the first frame of video and the computer does the rest.

On the other hand, any video clip, whether it is a film or any historical shot, can be our material, which means that its source is much wider than that of motion capture. In contrast to joint animation, this approach has a low computation cost, yet with a more realistic human motion instead. It also frees the animator from tedious routine.

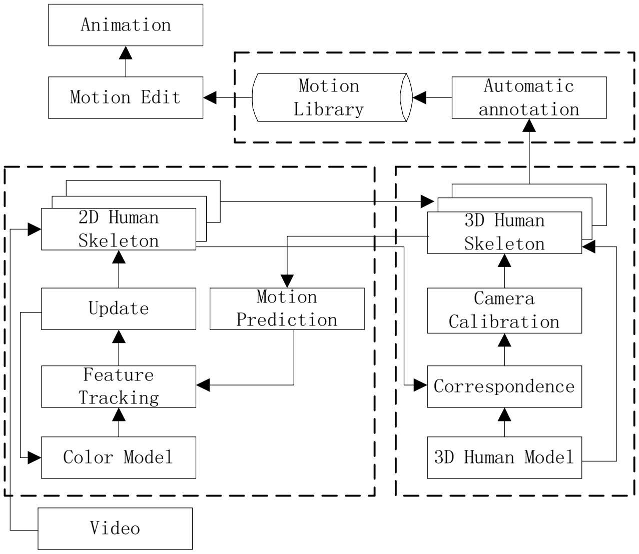

Figure 1. The architecture of video based human animation

The paper is organized as follows. Section 2 reviews the techniques of human motion analysis in computer vision. Section 3 introduces the human model used in our approach. The human feature tracking of image sequence and the construction of 3D human motion skeleton are detailed in section 4 and 5 respectively. Section 6 introduces how to construct motion library automatically. Section 7 shows the experiment results. Finally we give the conclusions and future directions.

2. A BRIEF SURVEY

In the computer vision domain, many researchers have made great efforts in analyzing and recognizing human motion in a video. The video may be shot by one single camera or by several cameras from different viewpoints simultaneously. Their approaches usually follow three steps: 1) feature extraction in video frames, such as body part segmentation, joint detection and identification, 2) correspondence between the features of every frame, 3) recovery of 3D human structure and motion from feature correspondences. O'Rouke and Badler[12] conducted 3D human motion analysis by mapping the input images to a volumetric model. The systems of Hogg[6],Rohr[11] were specialized to a one-degree-of-freedom walking model. Edge and line features were extracted from images and matched to a cylindrical 3D body model. Chen and Lee[5] used 17 line segments and 14 joints to represent the human skeleton model. Various constraints were imposed on the basic analysis of the gait. Bharatkumar et al.[3] also used stick figures to model the lower limbs of the human body. Their goal was to construct a general model for gait analysis in human walking. Akita[2] focused on the model-based motion analysis for real image sequence. A key frame sequence of stick figures indicated the approximate order of the motion and spatial relationships between the body parts. Bregler and Malik[4] recovered the 3D human motion information under the orthographic projection by marking the body segments in an initial frame. For the special complexity of human motion, the existing research methods laid much limitation on human, such as a uniform and quiescent background, parallelism of human motion direction to the image plane, and skintight clothing of human[1].

From the view of motion analysis, our approach removes many restrictions as in the previous approaches. For example, it does not aim at a given human motion mode. Rather, it analyzes large motion from frame to frame in complex, variational background, and finally sets up a 3D human skeleton model under the perspective projection. Then this model can be used in many applications such as human animation, virtual reality in which a 3D scene is needed.

3. THE HUMAN MODEL

The basic idea is to regard the 3D human body as an articulated object[1], which consists of rigid parts connected by joints. For example, the upper limb is composed of two rigid parts, the up arm and the low arm, which are connected by the elbow. Thus we simplify the human motion to the motion of skeleton and result in a 3D human skeleton model as show in figure 2.a. It contains 16 joints, which are named *3D feature points* in this paper. In our approach, we first use rigid constraints between the 3D feature point, which means that the length of each line in the 3D model remains constant. Furthermore, from anthropometric data we know that the length of each line is fractions of body height. So the constraint about length proportion of each line can also be added to our algorithm.

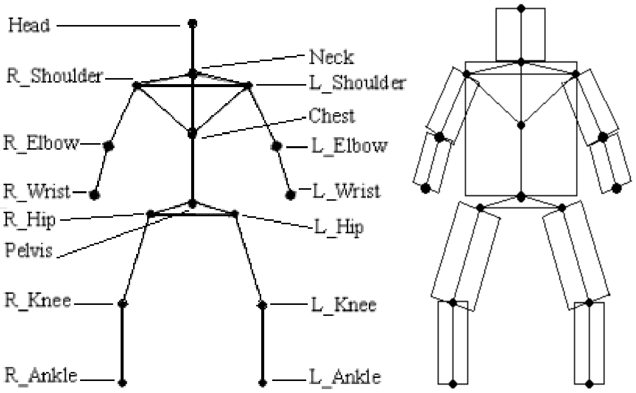

Figure 2. The human model: (a) a 3D skeleton model (left) (b) a 2D block model (right)

This paper names the projected point of a 3D feature point as a *2D feature point*. When the 3D human model is projected, the space relationship of each line in this model is consistent. But the projective line in the image plane may have a scale change. In the tracking of 2D image sequence, block is used to represent the projection of body part in image plane (see figure 2.b) for there is much color information on the body part. The middle line of each block is the skeleton after projection. It divides a corresponding block into two small blocks of equal area. After we mark each joint in the first frame, we may get the color model of each block in this frame. Then, if the new position of each block in the subsequent frames can be found, we will get human skeleton sequence on the image plane. Because of the high reliability of user marking, feature extraction is actually combined with feature correspondence into one step, i.e. finishing the feature correspondence during the course of tracking feature point. Having got this skeleton sequence, we can get 3D human motion information using computer vision technique.

4. FEATURE TRACKING

Because there is little self-occlusion on the human head, its color information can be acquired easily. After the head block is tracked, one feature point of trunk, the neck, is also fixed. So, beginning with head, we track every body part from top to bottom. Now we detail the tracking of head, trunk and limb respectively.

4.1 Head

For every frame in the sequence, the head may move toward any directions in the next frame. If a local search mode is used, the result may be not global optimal. If a global search mode is used, it suffers low efficiency. So what we adopt is the combination of these two. To reduce the search area of head point in the next frame, we introduce Kalman filter based global motion model to predict the motion of head point. Then in order to fix on the head accurately, we select a search path to do morph-block based match around the predicted point.

4.1.1 Kalman filter

Regarding the sequence of motion images as a dynamic system[9], the head point can be described by the following equation:

$$P=P' + \eta \qquad (1)$$

The coordinate $P=(x,y)^T$ is the tracked head point. P' is the actual coordinate. η is a 2D gaussian random noise with mean value 0 and covariance R. We use thrice polynomial to represent the motion trajectory of point P. The state vector is defined as

$$S=(P,P',P'') \qquad (2)$$

where $P'=(x',y')^T$,and x' ,y' represent the velocity of point P in the X,Y directions respectively. $P''=(x'',y'')^T$, where x'',y'' represent the acceleration of point P in the X,Y directions respectively. The state equation is defined as

$$S(k+1) = F \bullet S(k) + G \bullet n(k) \qquad (3)$$

where $$F = \begin{bmatrix} I_2 & I_2 \bullet \mathrm{T} & \frac{1}{2} I_2 \bullet T^2 \\ 0_2 & I_2 & I_2 \bullet T \\ 0_2 & 0_2 & I_2 \end{bmatrix} \qquad G = \begin{bmatrix} \frac{1}{2} I_2 \bullet T^2 \\ I_2 \bullet T \\ I_2 \end{bmatrix}$$

K=0,1,2,... represents the serial number of the frame, I_2 is a 2×2 unit matrix, 0_2 is a 2×2 zero matrix, and T is the time interval between frames. $n(k)=(\ n_x(k),n_y(k)\ )^T$ describes the acceleration noise in the x,y directions. Suppose $n(k)$ conform to the gaussian distribution with even 0 and covariance Q. This state equation shows that P is doing varied-acceleration linear motion in all the x,y directions. In practice, we track the coordinate of point P, i.e. $X(k)=p(k)$. So the measurement equation is:

$$X(k) = H \bullet S(k) + \eta(k) \qquad (4)$$

where $H=[I_2,0_2,0_2]$ is a 2×6 matrix. In the above conditions, we get the recursive equations of kalman filter as follows:

- State vector prediction equation:

$$S_k{}' = F \bullet S_k \qquad (5)$$

- State vector covariance prediction equation:

$$P_k{}' = F \bullet P_{K-1} \bullet F^T + G \bullet Q \bullet G^T \qquad (6)$$

- Kalman filter gain matrix:

$$K_k = P_k{}' \bullet H^T \bullet (H \bullet P_k{}' \bullet H^T + R)^{-1} \qquad (7)$$

- State vector covariance update equation:

$$P_k = P_k{}' - K_k \bullet (H \bullet P_k{}' \bullet H^T + R) \bullet K_k{}' \qquad (8)$$

- State vector update equation:

$$S_k = S_k{}' + K \bullet (X_k - H \bullet S_k{}') \qquad (9)$$

Kalman filter consists of initialization, prediction and update. The flow chat is shown in figure 3. Our experiments show that using kalman filter to predict the head point has a good performance.

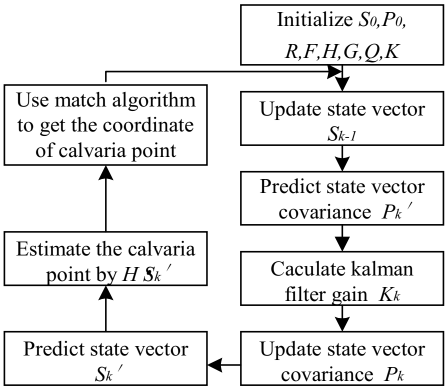
Initialize S_0,P_0, R,F,H,G,Q,K
Update state vector S_{k-1}
Predict state vector covariance $P_k{}'$
Caculate kalman filter gain K_k
Update state vector covariance P_k
Predict state vector $S_k{}'$
Estimate the calvaria point by $H\ S_k{}'$
Use match algorithm to get the coordinate of calvaria point

Figure 3. The flow chart of Kalman filter

4.1.2 Morph-block based match

We have applied the kalman filter to predict the possible position of head point in the next frame. Then in order to fix on the head accurately, we choose a search path (figure 4) to do morph-block based match around the predicted point.

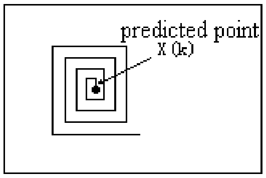

Figure 4. Search path

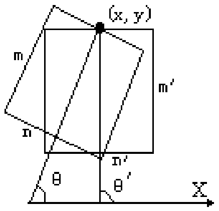

Figure 5. Two feature morph-blocks

Because we have known the head and the neck point in the first frame, the height (m) of the head block is the distance between these two points and the proportion of height to width (n) can be acquired in anatomy. The color information of $m \times n$ pixels in the block is saved as the color model for the matching of subsequent frames. Since the head block in the image is the projection of human head, the head motion will change the shape of projection. For example, the head block becomes larger which is likely to happen when human is moving toward the camera. So, the block match must be processed between morph-blocks. For this, we propose a *weighted morph-block similarity algorithm based on subpixel*.

Define a feature morph-block A={(x,y),m,n, θ }(see figure 5), where (x,y) is the intersection of one side and the middle line, m is the height of block A, n is the width of block, and θ is the angle between the middle line and X axis. Now there are a reference block $A=\{(x,y),m,n,\ \theta\}$ and a comparative block $A'=\{(x',y'),m',n',\theta'\}$. To calculate their similarity, we use the algorithm as follows:

1. If $m \times n < m' \times n'$, Then *row*=*m*, *column*=*n*; Else *row*=m', *column*=n';
2. In block A we depict *column* and *row* pieces of gridding lines evenly in the direction of $arctg\ \theta$ and $arctg(-1/\theta)$ respectively. The intersection of any two gridding lines is named as subpixel X_{ij},($0 \leqslant i<m,\ 0 \leqslant j<n$). Then we use quadric linear interpolation to compute the color of every subpixel, $X_{ij}[Red]$, $X_{ij}[Green]$, $X_{ij}[Blue]$.
3. In block A' we depict *column* and *row* pieces of gridding lines evenly in the direction of $arctg\ \theta'$ and $arctg(-1/\theta')$ respectively. Then we use linear interpolation to compute the color of every subpixel, $X_{ij}'[Red]$, $X_{ij}'[Green]$, $X_{ij}'[Blue]$.
4. Calculate:

$$\begin{aligned} diff_{ij} = {} & W_R \bullet |X_{ij}[Red]-X_{ij}'[Red]| \\ & + W_G \bullet |X_{ij}[Green]-X_{ij}'[Green]| \\ & + W_B \bullet |X_{ij}[Blue]- X_{ij}'[Blue]| \end{aligned} \tag{10}$$

$$S = 1/(W_1 \bullet \sum_{(i,j)\in b1} diff_{ij} + W_2 \bullet \sum_{(i,j)\in b2} diff_{ij}) \tag{11}$$

where W_R, W_G, W_B represent the weight of each element in *RGB*, *b1,b2* represent the two regions divided in block, and W_1, W_2 represent the weight of each region in the whole block. In the case of the head, we define the center region as *b1* and the marginal region as *b2*, respectively:

$$\begin{cases} (i,j)\in b1 & If \quad m/4 \le i \le (3/4)m \quad AND \quad n/4 \le j \le (3/4)n \\ (i,j)\in b2 & Otherwise \end{cases} \tag{12}$$

Here we have $W_1>W_2$. This weighted morph-block similarity measure is based on the observation that the marginal region of head has a more salient change of color in motion, however the center region has a relative small change. S is used to represent the similarity of two morph-blocks.

Note that the 3D human skeleton of frame *t-1* has already been established when human joint is tracked in frame *t*. Now we introduce how to predict the initial height of head block in frame *t* using the 3D human motion. The model of projected image height of head is illustrated in figure 6. H is the actual height of head in the 3D human model, f is the focal length of the camera, and O is the optical center of the camera. A point (x_t,y_t) in frame t is related to point (X_t,Y_t,Z_t) in the 3D camera coordinate system by $(x_t,y_t)=(X_t \bullet f/Z_t, Y_t \bullet f/Z_t)$, where Z_t is the distance of the head from the camera in frame *t*. Thus we have:

$$h_t=H \bullet f/Z_t \tag{13}$$

where h_t is the height of head in frame *t*. Assume the head is approaching or moving away from the camera at a locally constant velocity V, i.e.

$$Z_t = Z_{t-1} + V \bullet T \tag{14}$$

Using (14) to substitute Z_t in (13), we may get the initial height h_t in frame *t*.

For a frame sequence, we define the tracked head block in current frame as a reference block A, and the head block in the next frame as a comparative block A'. θ' is set in the range of [$\theta-\Delta\theta$, $\theta+\Delta\theta$]. By the previous prediction algorithm we get an estimated height of the head h_t and set m' in the range of [$h_t-\Delta m$, $h_t+\Delta m$]. Since the height and width of head zoom in proportion, n' is set in [$h_t(n/m)-(n/m)\Delta m$, $h_t(n/m)+(n/m)\Delta m$]. Starting from the predicted point, for every point (x,y) on the search path, we form several block A' by $\{(x,y),m',n',\theta'\}$ and calculate its similarity with the head block of current frame, A. The system records the block A' which has the largest similarity. After finding the largest similarity, the search process will continue until it does not find a block, which has a larger similarity, on the search path of next one circle. If does, repeat the process mentioned in the last sentence. In the end, the last recorded A' is the head block in the next frame. And for the self adaptability of color model, linear weight is utilized to update the color model[8].

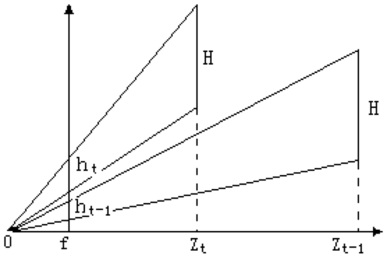

Figure 6. Prediction of the Head Height

4.2 Trunk and limb

The tracking of trunk and limb also depends on the above algorithm. But we must pay attention to other two problems. Firstly, because of the large limb motion from frame to frame, we introduce a prediction mechanism to estimate the possible limb position in the next frame. As the example of thigh, the relative angle from knee to hip is preserved for every frame. While doing prediction, we calculate the average value of such angles in the previous two frames, use it as the initial angle θ', and fix on θ' in the scope of [$\theta'-\Delta\theta$, $\theta'+\Delta\theta$]. Our experiment shows that this prediction mechanism can reduce the search area of block match for the large motion. Secondly, we show how to deal with self-occlusion in the tracking of limb. For example, there is relative small similarity in the block match of an up limb when in one frame the trunk occluded that up limb. But the similarity will be larger as soon as the occlusion disappears in one subsequent frame. According to this, the similarity S is also defined as the reliability of block match. In the block match process of frame sequence, the reliability of every limb match is preserved. If there are one or several low reliability frames between two relative high ones, we use the joint coordinate of high ones to obtain the joint of low ones by linear interpolation. Our experiment shows that it can deal with self-occlusion to a certain degree and optimize the tracking performance.

5. CONSTRUCTION OF 3D HUMAN MOTION SKELETON

To establish the sequence of 3D human motion skeleton under the perspective projection, we must first acquire the camera parameter, i.e. camera calibration in computer vision. This paper uses Newton method to solve this problem by the correspondences between 3D model and 2D image. Then we calculate the coordinate of the 3D feature point on the human model using the pinhole model and the proportion knowledge of human skeleton. In the frame sequence, the assumption of motion continuity is applied to eliminate the ambiguity of 3D motion information effectively.

5.1 Linear model based camera calibration

As shown in Figure 7, consider two coordinate systems, $O_wX_wY_wZ_w$ and $O_cX_cY_cZ_c$. The former is an object space coordinate system in which the 3D feature points are located. Thus P_w is a point in this coordinate system with coordinate (X_w,Y_w,Z_w). The camera is referenced to the camera coordinate system $O_cX_cY_cZ_c$. In particular, we assume that the image plane is perpendicular to the O_cZ_c axis and at location Z_c=f. Point coordinate on the image plane is obtained by the perspective projection and denoted by $P(u,v)$. Thus every point P_w in $O_wX_wY_wZ_w$ can be translated to (u,v) on the image plane with two transformations. Firstly, $O_cX_cY_cZ_c$ is obtained by a rotation R and translation t of the coordinate system $O_wX_wY_wZ_w$. The 3D coordinate of P_c is related to that of P_w by

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = R\left(\begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} - t\right) \tag{15}$$

Secondly, through the prospective projection, the projective point of P_c is at P whose coordinate is given by

$$(u,v)=\left(f\bullet X_c/Z_c,\ f\bullet Y_c/Z_c\right) \tag{16}$$

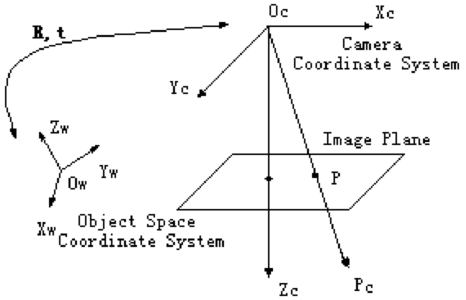

Figure 7. Projective transformation

Our goal in camera calibration is to determine R and t when some corresponding features between 3D human model and 2D image plane are given. In the above two equations, it is difficult to calculate the partial derivative of u,v to unknown parameters. So we transform them into

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = R\begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} \tag{17}$$

$$(u,v) = \left(\frac{f\bullet X'}{Z'+Dz} + Dx\ ,\frac{f\bullet Y'}{Z'+Dz} + Dy\right) \tag{18}$$

The meaning of R in the above equation is the same as that of equation (15). We substitute translation t with Dx,Dy,Dz. p' is a 3D point with coordinate (X' ,Y' ,Z'). These two representations are equivalent when t and Dx,Dy,Dz are related by:

$$t=R^{-1}\bullet[-Dx(Z'+Dz)/f,-Dy(Z'+Dz)/f,-Dz]^T \tag{19}$$

This paper represents rotate parameter R by a rotative vector, $(W_x, W_y, W_z)^T$, whose direction is equal to that of rotative axis and whose module is equal to the rotative angle. Now the projective parameter may be represented by a vector, (Dx,Dy,Dz,Wx,Wy,Wz) and the partial derivative of u,v to them can be calculated expediently. We use corresponding feature lines between 3D human model and image plane to calculate projective parameters. We define the equation of a line, with a point (u,v) on it, by:

$$\frac{-m}{\sqrt{m^2+1}}u + \frac{1}{\sqrt{m^2+1}}v = d \tag{20}$$

where d is the perpendicular distance from the origin to that line, and m is the line slope. From (20) we can get the partial derivative of d to u,v. Combining with previous calculation, the partial derivative of d to Dx,Dy,Dz,Wx,Wy,Wz will be obtained. After that, we may use Newton method to calculate a revisional vector, h=[ΔDx, ΔDy, ΔDz, ΔWx, ΔWy, ΔWz].

This method is details as follows. Firstly, beginning with the initial values of projective parameters, (Dx,Dy,Dz,Wx,Wy,Wz), let the 3D model project to the image plane according to the current parameters. Secondly, calculate the error between the projective line and the feature line on the image plane, which results in the following equation:

$$\frac{\partial d}{\partial Dx}\Delta Dx+\frac{\partial d}{\partial Dy}\Delta Dy+\frac{\partial d}{\partial Dz}\Delta Dz+\frac{\partial d}{\partial Wx}\Delta Wx+\frac{\partial d}{\partial Wy}\Delta Wy+\frac{\partial d}{\partial Wz}\Delta Wz=Ed \tag{21}$$

where Ed is the perpendicular distance from the end points of a 2D feature line to the projective line. Because there are two end points on one line, we can get two equations such as (21) for one pair of corresponding feature lines. So given three pairs of such lines, six equations will form a linear equation group. Its solving will lead to the revisional vector h. Then h is added to current projective parameters for revising projective parameters. Thus, we may solve the linear equation group again. All the Ed values in that equation group will be smaller than a predefined threshold after several iterations, which means that the six projective parameters have been obtained.

There are at least three pairs of corresponding lines needed in Newton method. In the human model, we choose the line between left and right shoulders, and the two lines between the chest and two shoulders. These three lines constitute a steady isosceles triangle of human up trunk. This choice is based on the observation that this triangle should not morph-itself in human motion under most situations. In the below description, each feature object of this triangle is named as a *key joint*, *key line*, or *key triangle*. In the first frame, the projection of key joints on the image plane is known by manual marking. The key joint position in the object space coordinate is specified by our system. As long as the proportion of each key line accords with the anatomy, we can always find the location and orientation of camera in the object space coordinate system and let the perspective projection of key triangle superpose with the up triangle of trunk on the image plane.

5.2 Construction of 3D human skeleton corresponding to the first frame

From the above work, six projective parameters have been obtained. Now corresponding to the first frame, except for three key joints, all the other 3D feature points of human model are not determined yet. The next step is to acquire the 3D feature point coordinate $P_c(X_c,Y_c,Z_c)$ of human model corresponding to a known 2D feature point coordinate, $P(u,v)$. As known from the pinhole model, to link the optical center and a projected point will get a radial, on which all the points project on the same point in the image plane. In order to locate the 3D feature point on this radial, we begin with a known neighboring point p and use the knowledge of human skeleton length *len* to find a point, the distance from which to p is equal to *len*.

Now we detail our algorithm by the example of inferring unknown right elbow point P_c from known neighboring point, the right shoulder point $P_c{}'$. The coordinate of right elbow point in the camera coordinate system is (X_c,Y_c,Z_c). (u,v) is the coordinate of P_c in the image plane. There is an equation as follow:

$$d(P_c', P_c) = Len \qquad (22)$$

where $d(P_c{}' ,P_c)$ represents the distance between $P_c{}'$ and P_c. *Len* is the up limb length in the 3D human model. By combining (22) with (16), we can get an equation, which has only one variable, Z_c. This equation can visualize as a line intersected with a sphere with center $P_c{}'$ and radius *Len*(see figure 9). According to three possibilities, intersection, tangency and apartion, of a space line intersected with a sphere, the solution of this equation has also three cases:

1. Two solutions. It means there are two possible positions for the elbow point. This ambiguity in the course of modeling from 2D to 3D is caused by this ill-posed problem itself. Two methods are used to eliminate the ambiguity. Firstly, we can utilize diversified human anatomy constraints. For example, the low arm can not extend backward when the up arm extends forward. Secondly, brightness information may be used. In the two solutions, one is always close to the optical center of camera and the other is far. We make an assumption that an image region, which is closer to the optical center, has a relative higher brightness. We choose a small region around the feature point on the shoulder and the elbow respectively, turn the *RGB* color model to *HLS* model for every pixel in these two regions, and calculate the mean values of *L* weight for two regions. If the value of elbow is larger than that of shoulder, we select the solution closer to the optical center, and otherwise we select the farther one. Our experiment shows that the combination of these two methods can eliminate the ambiguity effectively.
2. One solution. From it, we may get a unique point, which is just the position of 3D right elbow point.
3. No solution. There are two reasons: one is the tracking error of 2D feature point, the other is that the skeleton proportion of this person does not accord with the ordinary anatomy. The system will adjust the skeleton length for renewal computation.

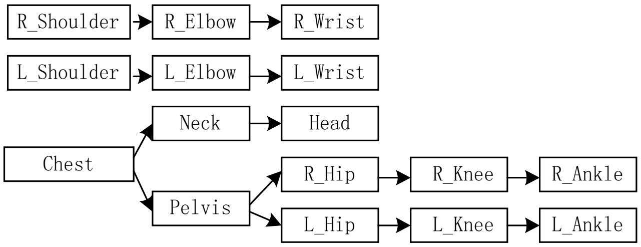

Figure 8. The solution order of 3D feature point

Now, with the solution order from center to margin shown in figure 8, we can get all the 3D feature point coordinates of the human model in turn.

5.3 Construction of 3D human skeleton corresponding to the subsequent frames

In the last subsection, we have constructed the 3D human motion skeleton for the first frame. In fact, as soon as the coordinates of three human key joints for every frame are known, the corresponding 3D human model can be obtained by the algorithm introduced in 5.2. Now we discuss how to determine the coordinates of three key joints in the subsequent frames[7].

Given the key joint coordinates, $P_i^n(X_i^n,Y_i^n,Z_i^n)$ $(i=1\sim3)$, of frame n in the camera coordinate system, let us calculate the corresponding key joint, P_i^{n+1} $(X_i^{n+1},Y_i^{n+1},Z_i^{n+1})(i=1\sim3)$, of frame $n+1$. The corresponding 2D feature point in the image plane is (U_i^{n+1},V_i^{n+1}). The relation of P_i^{n+1} and (U_i^{n+1},V_i^{n+1}) can be described as

$$P_i^{n+1} = \left(\frac{U_i^{n+1} \bullet Z_i^{n+1}}{f}, \frac{V_i^{n+1} \bullet Z_i^{n+1}}{f}, Z_i^{n+1} \right) (i=1\sim3) \qquad (23)$$

As mentioned in section 3, the skeleton length in the human model remains constant, which means:

$$d(P_i^n, P_j^n) = d(P_i^{n+1}, P_j^{n+1})\ (i,j=1\sim3)\ AND\ i<>j \qquad (24)$$

Using (23) to substitute P_i^{n+1} and P_j^{n+1} in (24), we will get a nonlinear equation group, which has three variables and may be solved by the grads method. Thus, we obtained the key joint coordinates of frame $n+1$ in the camera coordinate system.

Then the algorithm mentioned in 5.2 is used to calculate all the 3D feature points in the human model corresponding to frame $n+1$. Here a key assumption is made: the human motion has the property of continuity. While the ambiguity appears, we calculate the distances of the two solutions to the 3D feature point of frame n respectively and select the solution with a smaller distance. Our experiment shows that this method has excellent performance. The continuity and authenticity are embodied in the long sequence of human motion. In the example of figure 9, there are two possible positions, *P1* and *P2*, while locating the right elbow point corresponding to frame $n+1$. From the dashed line in figure 9 we know that the right elbow point in frame n is P' . Thus we select *P1* as the right elbow point of frame $n+1$ for it is closer to point P' .

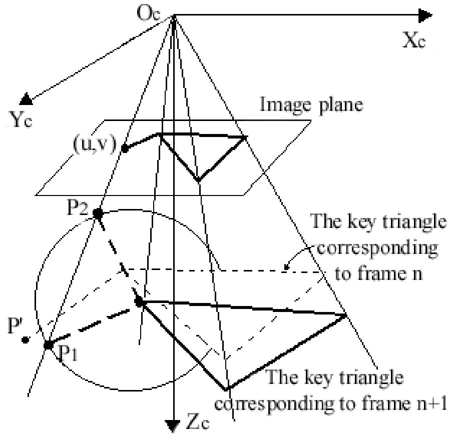
Oc
Xc
Yc
Image plane
(u,v)
P2
The key triangle corresponding to frame n
P'
P1
The key triangle corresponding to frame n+1
Zc

Figure 9. Ambiguity elimination in the subsequent frames

6. ESTABLISH MOTION LIBRARY AUTOMATICALLY

From the works of previous two sections, the 3D human motion information has been obtained. To support the animator with this information, we should establish a motion library, which contains diversified motion information. But at present this information only manifests as the coordinate of every human joint in the 3D space. While establishing library, we expect that not only these coordinates, but also various descriptions about motion information, such as motion type, initial position, motion space, etc, should be preserved in the entry of library. Having these descriptions, the animator can browse and query the motion library by the key word of these annotations. Apparently it is very useful for the animator to comprehend and utilize the motion knowledge in library.

Here this type of entry with motion annotation in natural language is accomplished by our system automatically. Natural language provides a large dictionary of movement-related terms. These terms are commonly used to define movements and their attributes; different forms of the same movement are distinguished by specifying different attributes related to a motion verb. We propose a motion grammar to classify motion in library based on [10]. This grammar has the goal of providing a multi-dimension, qualitative and quantitative description of human motion in natural language terms, and does not intend to be an exhaustive dictionary for human motion description. Additional entries may be added, following the general classification scheme. The motion grammar is detailed as:

--The symbol *::=* represent *Definition*.

--The symbol | represent *Or*.

--Tokens in symbol <> are *non-terminal symbols* of the grammar.

--Tokens starting with an upper case letter are *terminal symbols* of the grammar.

<action>::=<initialPosition><motion><finalPosition>

<initialPosition>::=Erect | Stooped | Knelt | Supine | Seated | Others

<finalPosition>::=Erect | Stooped | Knelt | Supine | Seated | Others

<motion>::=<transition> <qualitativeAspects> | <locomotion> <space> <time>

<transition>::=Raise | Fall | Sit | Lag | Stoop | Kneel | Others

<qualitativeAspects>::= Slowly | Suddenly

<locomotion>::=Walk | Jump | Descend | Ascend | Hop | Run | Others

<space>::= <Direction> <Trajectory> <Gradient>

<direction>::= Forward | Backward | Left | Right

<trajectory>::= Straight | Zig_zag

<gradient>::= Horizontal | Vertical | Ascending | Descending

<time>::= <speed> <speedData> <acceleration> <accelerationData>

<speed>::= Slow | Fast

<acceleration>::=Constant | Accelerating | Decelerating

The meaning of some terms are:

- initialPosition and finalPosition : verbs represent the start and end posture of human motion respectively.
- transition : verbs imply that human motion has not any displacement, but just an onsite transition from an initial posture to a final posture, for example, “Stoop”.
- Locomotion : verbs specifying a spatial displacement from an initial position to a final position, for example, “Walk”.
- qualitativeAspects are attributes specifying the way in which a motion is performed, for example, “Sit slowly”.
- space and time attributes relate to the description of locomotion.
- speed is the description of motion speed and acceleration, where speedData and accelerationData are two quantitative description of the speed attribute. As known from the pinhole model, if the human model is two times farther away from the image plane, but twice as big, and translated at twice the speed, we would get exactly the same two images. Therefore, if we suppose that the height of human model is 17.5 CM and the motion speed is 1 CM/Sec, the motion speed will be 1 M/Sec when the actual human height is 1.75 M. The motion speed of human model can be calculated by the speed of pelvis point. Thus, while establishing the motion library, we calculate the quantitative speed and acceleration and add them to library as motion attributes.

Now we will show how to determine the entries in grammar. In the example of “initialPosition”, we define several angles according to 3D human skeleton model.

- θ (Pelvis):= The projective angle of the pelvis skeleton to the plane of hip triangle.
- θ (Knee):= The angles from the left and right thigh skeleton to the respective shin skeleton.
- θ (Ankle):= The angles from the left and right shin skeleton to the *OZX-plane* in the camera coordinate system.
- θ (Thigh):= The angles from the left and right thigh skeleton to the *OZX-plane* in the camera coordinate system.

The “initialPosition” of human skeleton can be determined by the decision tree shown in figure 10.

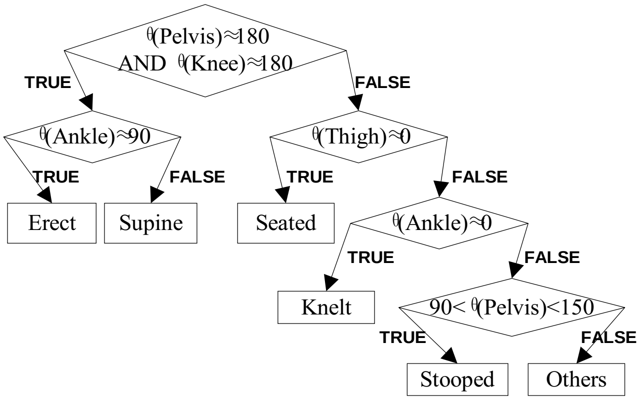

Figure 10. Decision tree

In this figure, the symbol $\approx$ represents equality approximately. In practice, we use a threshold to control it. For example, θ *(Pelvis)*

≈90 is equal to 90-w< θ(*Pelvis*)<90+w *and* w=20. It can be seen that this is a production system, which utilizes a combination of several qualifications to deduce the motion attribute. Other attributes' deduction is similar to this one. For example, if the ankle point has not space displacement, the motion is in the category of "transition". The attribute of "Direction" can be known from the change of X and Z coordinate values of pelvis point. At last, we obtain a section of natural language description composed by various attributes in grammar.

7. EXPERIMENT RESULTS

Based on the above algorithms, we have implemented a demo system, *Video Based Human Animation(VBHA),* using Visual C++ and Open-GL on personal computer. *VBHA* can construct 3D human skeleton sequence and establish motion library automatically. We apply it to video recordings in our lab and a video clip of actor's dancing in MASK.

The video recordings in our lab were done with a single camera. Figure 11 shows one example sequence of a human sitting down on a chair seen from an oblique view. In the top row, the 16 feature points on the first image are marked by the user with the mouse. After the hand-initialization we applied the program to a sequence of 25 image frames. We could successfully track all body joints in the video sequence. The other five frames of the top row are the 5th, 10th, 15th, 20th and 25th frame of the clip respectively. At the same time, *VBHA* construct the sequence of 3D human skeleton under the perspective projection. We define a virtual camera to simulate the camera used in practice. In the middle row, we show six images of constructed skeletons, which are shot in the same viewpoint as the top ones. Then we rotate the camera right with 30° , shoot six frames corresponding to the top ones, and show them in the bottom row. As you see, the motion continuity and authenticity are embodied in this sitting sequence, which proves the robustness of algorithm in ambiguity elimination. It means now we can see the person sit down from a more oblique view. At last, annotation of this motion entry is described as follows by *VBHA* automatically:

- initialPosition: Erect
- transition: Sit
- qualitativeAspects: Slowly
- finalPosition: Seated

In figure 12 we show an example of a human walking toward the camera. The five frames of the top row are the 1st, 5th, 17th, 23th and 41th frame of the clip respectively. The initialization and the meaning of the middle row are the same as before. But in the bottom tow, we show the skeletons when the virtual camera is posed from various viewangles. These five images are shot by the camera rotating left 15° , right 45° , right 30° , left 60° and left 30° respectively. In this case, the human body has an obvious scale change on the image plane. Here our height prediction and morph-block similarity algorithm take effect and result in a good performance in the tracking of 41 frame sequence. For this motion entry, *VBHA* annotate it with:

- initialPosition: Erect
- locomotion: Walk
- space: Forward Straight Horizontal
- time: Slow 0.8M/Sec Constant 0

As shown in figure 13, the last experiment material is a video clip from the film MASK with a length of 2 second (48 frames). In the top row these five ones are the 1st, 5th, 17th, 26th, 39th frame of the clip respectively. If you have seen this film, you must know that this actor dances very fast. So here our motion prediction algorithm helps us to track every body part, especially the low limb in the sequence. In the middle row, we show images, which are shot in the same viewpoint as the top ones. In the bottom row, these five images are shot by the camera rotating right 30° , left 30° , right 15° , left 15° and left 150° respectively. These two rows reflect the various motion configurations vividly. For this motion entry, *VBHA* annotate it with:

- initialPosition: Erect
- locomotion: Others
- space: Left Straight Horizontal
- time: Slow 0.5M/Sec Constant 0

For more information of our experiment results, please visit: http://icad.zju.edu.cn/~liuxm/animation.html.

8. CONCLUSIONS

This paper has presented a video based human animation technique. Given a clip of video, we can acquire 3D human motion information and construct an entry of motion library. Then an animator can utilize the existing motion information and produce a new human animation. This approach has the characteristic of rich source material, low computing cost, efficient production, and realistic animation result.

It is challenging to animate realistic human motion. Our contribution to this problem is that we open up a huge resource of archived human movement captured on video for use by 3D computer animators, and propose the basic algorithms for this animation technique. With this approach, any video clip, whether it is a film or any historical shot, such as Charlie Chaplin's walking and Karl Lewis' running, can be the source material, which means the animator can utilize much more motion information than before. It is easy and straightforward from a user's point of view. Just to mark the joints of the first frame, he will see the animated human motion from any viewangle. On the other hand, from the viewpoint of motion analysis, this approach does not pose any restrictions on human motion. Rather, it analyzes large motion from frame to frame in complex, variational background, and finally sets up a 3D human skeleton model under the perspective projection, which make it possible to construct a 3D scene for the animation. To the best of our knowledge, this is the first demo system that is able to process such a challenging task and recover complex human motion with high accuracy.

From the experiment, we found the tracking performance will be weakened if there are too much self-occlusion on the human body. So, our future work will concentrate on utilizing more knowledge of 3D human skeleton motion to guide the 2D feature tracking and adding some feedbacks from the animator. As we see, as soon as the mapping of video to model is obtained, a lot of work may be carried out based on it. One of such work is rendering the model using another new video. What we are planning to do is a virtual Chaplin, who may simulate any gesture that real Chaplin has performed in the video.

9. ACKNOWLEDGEMENTS

Our work was sponsored by the National Natural Science Foundation of China. We would also like to thank Yiying Tong, Kun Zhou, Zhiqiang Lao, Jun Yang, Yi Wu and Xiqun Lu for fruitful discussions.

10. REFERENCES

[1] Aggarwal, J. K. and Cai, Q. Human Motion Analysis: A Review. In Proceedings of the IEEE Nonrigid and Articulated Motion Workshop 1997. IEEE, Piscataway, NJ, USA.

[2] Akita, K. Image Sequence Analysis of Real World Human Motion. Pattern Recognition, Vol.17 No.1, 1984.

[3] Bharatkumar, A. G., Daigle, K. E., Pandy, M. G., Cai, Q. and Aggarwal, J. K. Low limb kinematics of human walking with the medial axis transformation. In Proc. Of IEEE computer Society Workshop on Motion of No-Rigid and Articulated Objects, Austin,TX,1994, 70-76.

[4] Bregler,C. and Malik,J. Video Motion Capture. UC Berkeley, Technical Report CSD-97-973. http://www.cs.berkeley.edu/~bregler/bregler_malik_sig98.ps.gz.

[5] Chen, Z. and Lee, H. J. Knowledge-guided visual perception of 3D human gait from a single image sequence. IEEE Trans. On Systems, Man, and Cybernetics, 22(2),1992, 336-342.

[6] Hogg, D. A program to see a walking person. Image Vision Computing, 5(20), 1983.

[7] Huang, Thomas S. and Netravali, Arun N. Motion and Structure From Feature Correspondences: A Review. In Proceedings of The IEEE Vol.82 No.2, Feb. 1994, 252-268.

[8] Liu Mingbao, Yao Hongxun, Gao Wen, Real-time human face tracking in color images. Chinese Journal of Computer, Vol.21 No.6 June 1998, 527-532.

[9] Ma SongDe, Zhang ZhengYou, Computer Vision-Compute Theory and Arithmetic Foundation, Scientific Press. Jan.1998.

[10] Maiocchi, Roberto and Pernici, Barbara. Directing and animatated scene with autonomous actors. The Visual Computer, 6,1990, 359-371.

[11] Rohr, K. Incremental recogniton of pedestrians from image sequences. In Proc. IEEE Cpmput. Soc. Conf. Comput. Vision and Pattern Recogn, New York City, June 1993.

[12] Rourke, J.O' and Badler, N. I. Model-based image analysis of human motion using constraint propagation. IEEE Trans. Pattern Anal. Mach. Intell.,2(6), November 1980, 522-536.

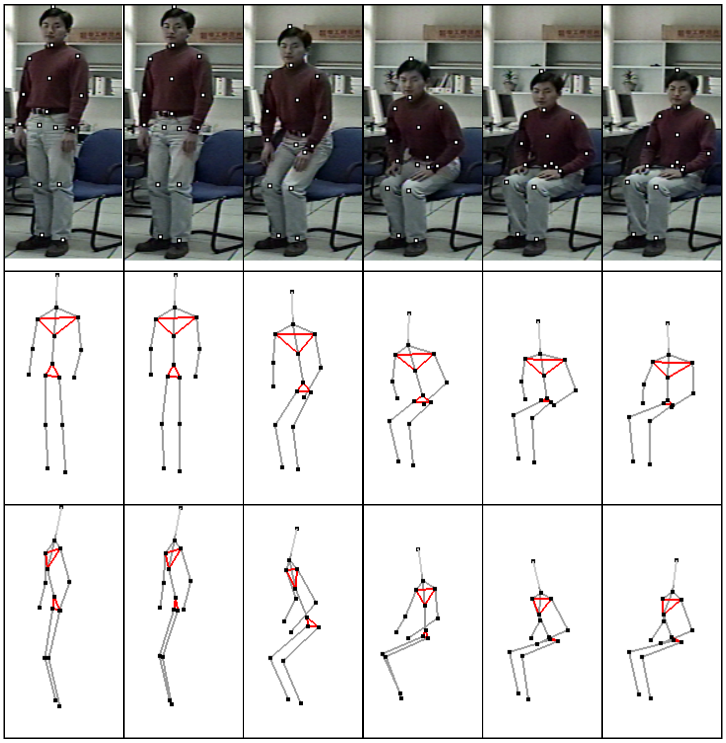

Figure 11. The human body's sitting down: The top row shows tracked frames, the middle row shows constructed 3D human skeletons in the same viewpoint, and the bottom row shows the skeletons from right 30° viewangle.

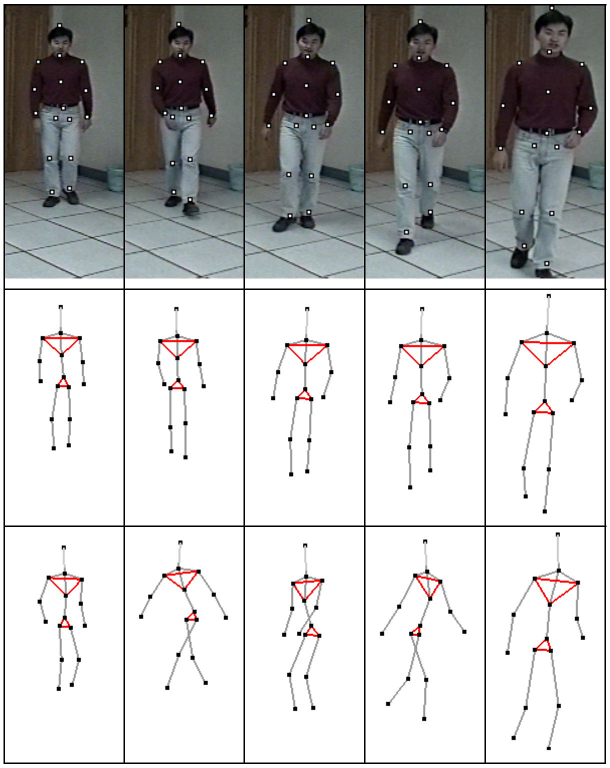

Figure 12. The human body's walking: The top row shows tracked frames, the middle row shows constructed 3D human skeletons in the same viewpoint, and the bottom row shows the skeletons from various viewangles.

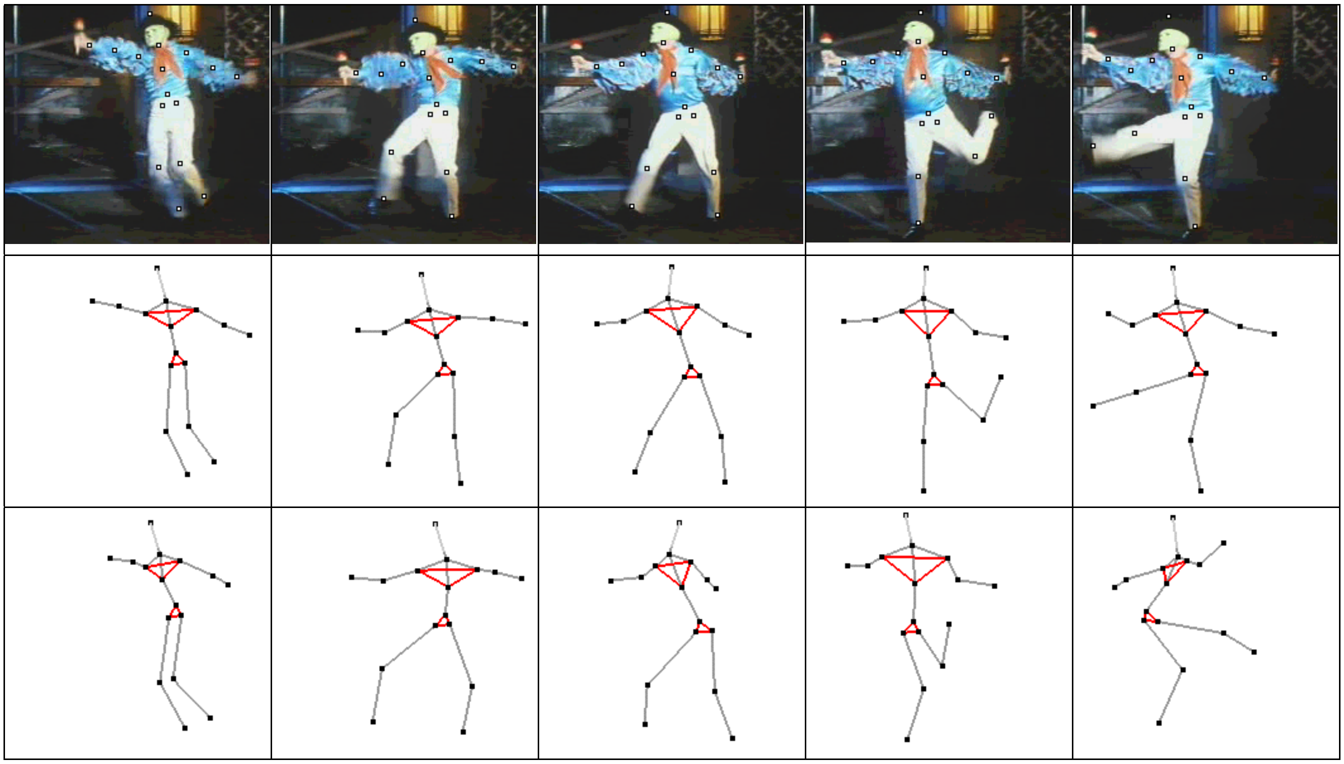

Figure 13. The experiment of MASK: The top row shows tracked frames, the middle row shows constructed 3D human skeletons, and the bottom row shows the skeletons from various viewangles.