VIDEO MOTION CAPTURE IN VBA—VIDEO-BASED ANIMATION*

PAN Yun-he(潘云鹤), ZHUANG Yue-ting(庄越挺), LIU Xiao-ming(刘小明)

(*Microsoft Visual Perception Laboratory, Dept. of Computer Science, Yuquan Campus of Zhejiang University, Hangzhou 310027, China*)

E-mail: panyh@sun.zju.edu.cn and yzhuang,@cs.zju.edu.cn,

Received Dec.8, 1999; revision accepted Dec.29,1999

Abstract: Computer vision has very wide application in human motion capture research. This paper proposes a new approach to do motion capture in video. It is composed of image sequence based tracking of human feature points and the reconstruction of the three-dimension(3D) motion skeleton. First, every part of the human body from top to bottom is tracked on the basis of a human model. The image difference and a morph-block similarity algorithm based on subpixels are used. Then camera calibration is done using the line correspondences between the 3D model and the image. Finally the 3D motion skeleton is established by use of the model knowledge. This approach does not aim at a given mode of human motion. Rather, it analyzes large scale motion from frame to frame in complex, variational background, and sets up a 3D motion skeleton in the perspective projection. The experiment results are presented at the end of the paper.

Key words: video, feature point, image difference, camera calibration, skeleton.
Document code: A **CLC number:** TP391.4

INTRODUCTION

Motion capture plays an important role in the creation of special effects in many fields. Aside from use in films and animations, motion capture has comprehensive applications in the analysis of athlete performance, medical diagnostics, surveillance, video retrieval, etc.

Conventional motion capture can be achieved through two approaches. One is to attach many sensors to the joints of the human body, so that they will record the position of the joints at every instant. The other is to analyze a video image in the following three steps: 1) feature extraction in video frames, 2) determination of the correspondence between the features of every frame, 3) recovery of 3D motion from feature correspondences. O'Rouke and Badler(1980) analyzed 3D human motion by mapping the input images to a volumetric model. In the systems of Hogg (1983) and Rohr(1993), edge and line features are extracted from images and matched to a cylindrical 3D body model. Chen and Lee(1992) used 17 line segments and 14 joints to represent the human skeleton model. Bharatkumar et al. (1994) also used stick figures to model the lower limbs of the human body in order to construct a general model for analyzing gait in walking. Bregler and Malik(1998) recovered information on 3D human motion shown in orthographic projection by marking limb segments in the initial frame. Due to the special complexity of human motion, the existing research methods have humanly unavoidable limitations (Aggarwal et al., 1997), such as a single and quiescent background, parallelism of motion direction to the image plane, and tight human clothing. To attach sensors will cost too much money and time, restrict free movement.

We propose a new motion capture approach not burdened by the many restrictions in previous approaches. It does not aim at a given human motion mode. Rather, it analyzes large scale motion from frame to frame in complex, variational background, and finally sets up a 3D human skeleton under the perspective projection. Then this model can be used in many applications such as human animation, VR, etc. The user only needs to mark the joints of the first frame and the computer will do the rest. In par-

* Project(69803009) supported by Natural Science Foundation of China (NSFC) and Key Project of National Development Planning Committee.

ticular, we emphasize two objectives. One is to acquire the sequence of the 2D human motion skeleton by tracking the joint with the support of motion prediction and color model of the body part. The other is to use the correspondences between the 3D model and the 2D image to calibrate the camera and establish the sequence of the 3D human skeleton in the perspective projection.

The setup for video motion capture is shown in figure 1. The setup in the dashed boxes serves to achieve the two objectives mentioned above. Section 2 in this paper introduces the human model used in our approach. The human feature tracking of the image sequence and reconstruction of the 3D human motion skeleton sequence are detailed in section 3 and 4 respectively. Section 5 presents the experiment result.

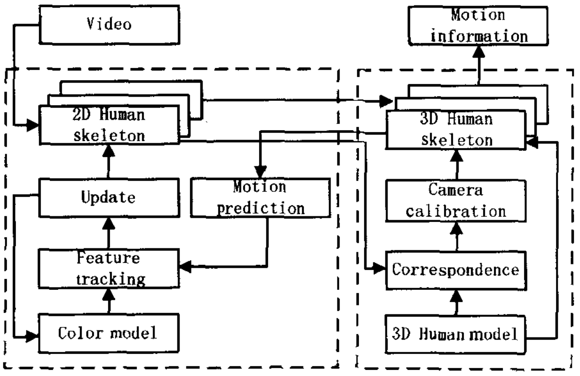

Fig. 1 The setup for video motion capture

THE HUMAN MODEL

The basic idea is to regard the 3D human body as an articulated object(Huang, 1994) and simplify the human motion to that of skeletal motion. Fig.2a shows a 3D human skeleton model containing 16 joints named 3D feature points. From the knowledge of anatomy, we can acquire the length proportion of each line in this model.

In this paper, the projection of a 3D feature point is called a 2D feature point. In the tracking of 2D image sequence, we use a block to represent the projection of a body part in the image plane (see Fig. 2b). The middle line of each block is the skeleton after projection. It divides a corresponding block into two small blocks of equal area. After the marking of the first frame, we may get a color model of each block. By searching out its new position in the subsequent frames, we can track the human skeleton on the image plane.

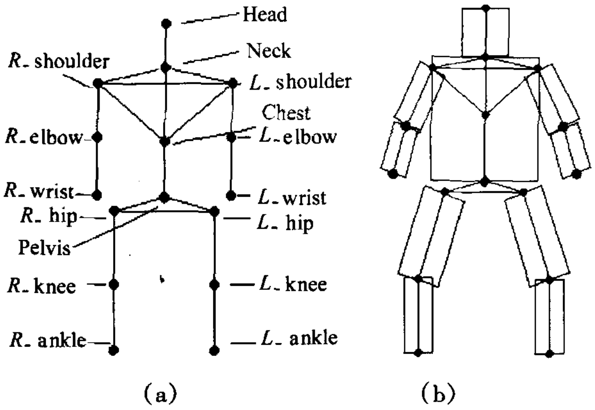

Fig. 2 Human model

(a) 3D skeleton model; (b) 2D block model

FEATURE TRACKING

Because there is little self occlusion on the human head, we can acquire its color information easily. So, beginning with the head, we track every body part from top to bottom. Now we detail the tracking of head, trunk and limb respectively.

1. Head

For every frame in the sequence, the head may move toward any direction in the next frame. To reduce the search area of calvaria point in the next frame, we introduce an image difference based global motion model to predict the calvaria point. Then we select a search path to do morph-block match around the predicted point.

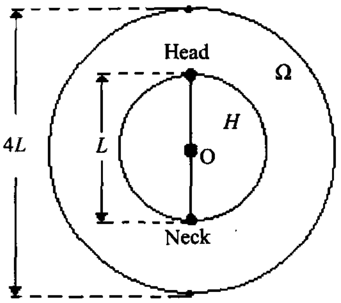

Fig. 3 Head region

Image difference Image difference is a widely used image processing technique. Here we show how it is used to estimate the motion direction and displacement of the human head in the subsequence frames. Having gotten the calvaria and neck point in the current the frame k, we approximate the head region as a circular region, H. In Fig. 3, the diameter and center of H is represented as L and O respectively. Because there is limited small motion of heads between two consecutive frames, we define the difference region as a circular one, Ω, whose center is O and radius is $2L$. Calculate the color value of pixels in difference region by the following equations:

$$\text{Difference}_{ij}[R] = |X_{ij}[\text{Red}] - X_{ij}'[\text{Red}]| \quad (i,j)\in\Omega \qquad (1)$$

$$\text{Difference}_{ij}[G] = |X_{ij}[\text{Green}] - X_{ij}'[\text{Green}]| \quad (i,j)\in\Omega \qquad (2)$$

$$\text{Difference}_{ij}[B] = |X_{ij}[\text{Blue}] - X_{ij}'[\text{Blue}]| \quad (i,j)\in\Omega \qquad (3)$$

Where X_{ij} represents the color values (RGB) of a pixel in the K frame, whose coordinate is (i,j). X_{ij}' represents them in the $K+1$ frame. Because the head may move toward any directions in the next frame, we should judge its motion direction at first, and then estimate its motion displacement. Fig. 4 shows the difference region, where H' is the head position in the next frame. We suppose that there is little variety in background between consecutive frames. So except for the two head regions, every pixel in the difference region has an approximately color value of 0. We calculate the motion direction, α, by the following algorithm:

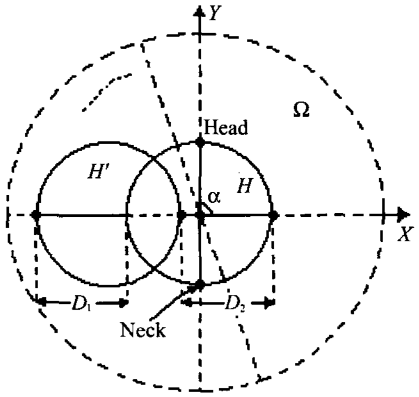

Fig.4 Image difference region

1. Beginning from $\alpha_0 = 90°$, we get 36 angle values $\alpha_{n+1} = \alpha_n + 5°$;

2. For every α_n, extend its diameter to intersect with the boundary of difference region, Ω, and obtain 36 lines whose length is $4L$;

3. Calculate the sum of every pixel on every line, L_n, by the following equation:

$$\text{Sum}_n = \sum_{(i,j)\in L_n} (\text{Difference}_{ij}[R] + \text{Difference}_{ij}[G] + \text{Difference}_{ij}[B]) \qquad (4)$$

4. Regard α as an angle, whose L_n has the largest Sum_n:

$$\alpha = \{\alpha_n \mid \text{Sum}_n = \text{MAX}(\text{Sum}_0, \cdots\cdots, \text{Sum}_{36})\}; \qquad (5)$$

The foundation of this algorithm is that the head region has the most displacement on the motion direction. Based on this, the largest sum of color of pixels is in the motion direction in difference region. After determining the motion direction, we estimate the head displacement in that direction. We obtain the color values (RGB) of every pixel on line L_n, and plot them in Fig. 5 showing a clear two-peak curve. The left peak corresponds to the difference from H' to the background, and the right one corresponds to the difference from H to the background. Actually these two differences represent the head displacement in the image plane and have approximately the same width. Our approach is to calculate the average value (denoted as a) of all pixels on the line, L_n, and then define a threshold value, p, such as $p = 0.5$. In Fig. 5, we draw a line, $Y = a \cdot p$, which can produce two lines, D_1 and D_2, while intersecting with the two-peak curve. At last we regard the average length of D_1 and D_2 as the displacement of head motion.

The above approach only fixes the motion direction on a line, but not on which end of that line. In practice, we just need to calculate the distance of D_1 and D_2 to O. If the distance of D_1 to O is larger, the head moves to the direction of D_1, otherwise moves to D_2.

Thus, by the image difference, we can obtain the estimated position of calvaria point in the next frame. When we get its precise position by morph-block match, we should weight the veracity of motion estimation. If the estimation is larger, we add the threshold, p. Otherwise minish p. By this adaptive adjustment of threshold, we enhance the accuracy of motion estimation.

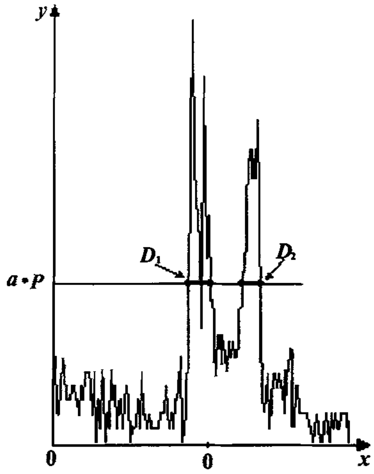

Fig.5 A two-peak curve in the motion direction

Morph-block match We have applied the image difference to predict the calvaria point in the next frame. Now we will choose a search path (Fig. 6) to do morph-block based match around the predicted point.

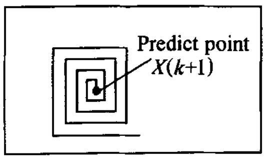

Fig.6 Search path

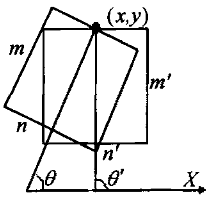

Fig.7 Two feature morph-blocks

Because we know the calvaria and 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 by the human anatomy. The color information of $m \times n$ pixels in this block is saved as the color model for the match of subsequent frames. Since the head block in the image is the projection of the human head, the head motion will change the shape of projection. For example, the head block becomes larger, when a human is moving toward the camera. So, the block match must be processed between morph-blocks. For this, we propose a morph-block weighted similarity algorithm based on subpixel.

Define a feature morph-block $A = \{(x,y), m, n, \theta\}$ (see Fig.7), where (x,y) is the intersection of one side and the middle line, m is the height and n the width of block A, and θ is the angle between the middle line and X axis. Now there is a reference block $A = \{(x,y), m, n, \theta\}$ and a comparative block $A' = \{(x', y'), m', n', \theta'\}$. To calculate their similarity, use the algorithm as below:

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 θ and $\theta + 90°$ respectively. We name the intersection of any two gridding lines as subpixel X_{ij}, $(0 \leqslant i < m,\ 0 \leqslant j < n)$. Then we use quadric linear interpolation to calculate the color of every subpixel, $X_{ij}[\mathrm{Red}]$, $X_{ij}[\mathrm{Green}]$, $X_{ij}[\mathrm{Blue}]$.

3. In block A' we depict column and row pieces of gridding lines evenly in the direction of θ' and $\theta' + 90°$ respectively. Then we use linear interpolation to calculate the color of every subpixel, $X_{ij}'[\mathrm{Red}]$, $X_{ij}'[\mathrm{Green}]$, $X_{ij}'[\mathrm{Blue}]$.

4. Calculate:

$$\begin{aligned} \mathrm{diff}_{ij} = {} & W_R \cdot | X_{ij}[\mathrm{Red}] - X_{ij}'[\mathrm{Red}] | \\ & + W_G \cdot | X_{ij}[\mathrm{Green}] - X_{ij}'[\mathrm{Green}] | \\ & + W_B \cdot | X_{ij}[\mathrm{Blue}] - X_{ij}'[\mathrm{Blue}] | \end{aligned} \tag{3}$$

$$S = 1/(W_1 \cdot \sum_{(i,j)\in b_1} \mathrm{diff}_{ij} + W_2 \cdot \sum_{(i,j)\in b_2} \mathrm{diff}_{ij}) \tag{4}$$

where W_R, W_G, W_B represent the weight of each element in RGB, b_1, b_2 represent the two regions divided in the block, 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 b_1 and the margin region as b_2, namely:

$$\begin{cases} (i,j) \in b_1 \text{ If } m/4 \leq i \leq (3/4)m \\ \qquad \text{and } n/4 \leq j \leq (3/4)n \\ (i,j) \in b_2 \quad \text{Otherwise} \end{cases} \tag{5}$$

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

For a frame sequence, we define the tracked head block in the current frame as the reference block A, and the head block in the next frame as the comparative block A'. We set θ' as $\theta - \Delta\theta \leqslant \theta' \leqslant \theta + \Delta\theta$ and m' as $m - \Delta m \leqslant m' \leqslant m + \Delta m$. Since the height and width of the head zoom in proportion, we set n' as $n - (n/m)\Delta m \leqslant n' \leqslant n + (n/m)\Delta m$. Then for every point (x, y) on the search path, we form several A' by $\{(x, y), m', n', \theta'\}$ and calculate its similarity with the head block of the current frame, A. The system records the A' which has the largest similarity. After finding the largest similarity on the past search path, the search process will continue until on the search path of the next circle it does not find a point which has a larger similarity. If it does, repeat the process mentioned in the last sentence. In the end, the recorded A' is the head block of the next frame. And for the self adaptability of the color model, we utilize linear weight to update the color model (Akita, 1984).

2. Trunk and Limb

When the head block is tracked, one feature point of trunk, the neck, is fixed on. The tracking of trunk and limb also depends on the above algorithm. But we must pay attention to two problems. First, because of the large limb motion from frame to frame, we introduce a prediction mechanism to estimate the potential limb position in the next frame, and then fix it on accurately(Liu et al., 2000)*. Then, we show how to deal with self occlusion in the tracking of limb. For example, there is relatively small similarity in the block match when in one frame the trunk occludes an upper limb. But the similarity will be larger as soon as the occlusion disappears. According to this, we also define the similarity S as the reliability of block match. In the match process of the frame sequence, we preserve the reliability of every limb match. If there are one or several low reliability frames between two relatively 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 extent and optimize the tracking performance.

RECONSTRUCTION OF 3D HUMAN MOTION SKELETON

To establish the sequence of 3D human motion skeleton in the perspective projection, we must first acquire the camera parameter, namely camera calibration in computer vision. Then we calculate the coordinate of 3D feature points on the human model by use of the pin-hole model and the knowledge of the human skeleton.

Consider two coordinate systems(Ma et al., 1998), $O_wX_wY_wZ_w$ and $O_cX_cY_cZ_c$. The former is an object space coordinate system in which the 3D feature points are located. The camera is referenced as the camera coordinate system $O_cX_cY_cZ_c$. Then every point P_w in $O_wX_wY_wZ_w$ can be translated to (u, v) on the image plane by two transformations, a rotation R and a translation t. Our goal in camera calibration is to determine R and t when some corresponding feature lines between 3D human model and image plane are given. To simplify the calculation of partial derivative, we transform the standard formulas into

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = R \begin{pmatrix} X_W \\ Y_W \\ Z_W \end{pmatrix} \tag{6}$$

$$(u, v) = \left(\frac{f \cdot X'}{Z' + D_z} + D_x, \frac{f \cdot Y'}{Z' + D_z} + D_y \right) \tag{7}$$

We substitute translation t with D_x, D_y, D_z, and represent parameter R by a rotation vector, $(W_x, W_y, W_z)^T$. We define the equation of a line, with a point (u, v) on it, by

* Liu Xiaoming, Zhuang Yueting, Pan Yunhe et al., Human three dimension motion skeleton reconstruction of image sequence. Accepted for publication In *Journal of Computer Aided Design and Computer Graphics*, Spring, 2000)

$$\frac{-m}{\sqrt{m^2+1}}u + \frac{1}{\sqrt{m^2+1}}v = d \qquad (8)$$

where d is the perpendicular distance from the origin to that line, and m is the line slope. Now the partial derivative of d to D_x, D_y, D_z, W_x, W_y, W_z will be obtained. After that, we may use Newton method to obtain six projective parameters based on the following equation:

$$\frac{\partial d}{\partial D_x}\Delta D_x + \frac{\partial d}{\partial D_y}\Delta D_y + \frac{\partial d}{\partial D_z}\Delta D_z + \frac{\partial d}{\partial W_x}\Delta W_x + \frac{\partial d}{\partial W_y}\Delta W_y + \frac{\partial d}{\partial W_z}\Delta W_z = E \qquad (9)$$

where E 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 (9) for one pair of corresponding feature lines. Given three pairs of such lines, six equations will form a linear equation group. So, 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. This choice is based on the observation that this triangle should not morphitself in motion under most conditions. In the below description, we name each feature object of this triangle as the key joint, key line, and the key triangle. In the first frame, the projection of key joints on the image plane is known by manual marking. The position of key joint 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 the camera in the object space coordinate system and let the perspective projection of key triangle superpose with the up triangle of the trunk on the image plane.

Now, corresponding to the first frame, except for three key joints, all other 3D feature points of the human model are not determined yet. The next step is to acquire the 3D feature point coordinate (X_c, Y_c, Z_c) of the human model corresponding to a known 2D feature point coordinate, (u, v). As known from the pin-hole model, to link the optical center and a projective point reguires a line, on which all the points project on the same point in the image plane. In order to locate the 3D feature point on this line, we begin with a known neighboring point and use the knowledge of human skeleton length to find a point, the distance from which to the known neighboring point is equal to the corresponding skeleton length (See the footnote on page five). Thus, with the order from center to fringe in the skeleton model, we can get all the 3D feature points coordinates in the human model.

Then we discuss how to determine the coordinates of three key joints corresponding to the subsequent frames (Hang et al., 1994). Given the key joint coordinates, $P_i^n(X_i^n, Y_i^n, Z_i^n)(i = 1 \sim 3)$, of frame n in the camera coordinate system, let us calculate the corresponding key joint, $P_i^{n+1}(X_i^{n+1}, Y_i^{n+1}, Z_i^{n+1})(i = 1 \sim 3)$, 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

$$P_i^{n+1} = \left(\frac{U_i^{n+1} \cdot Z_i^{n+1}}{f}, \frac{V_i^{n+1} \cdot Z_i^{n+1}}{f}, Z_i^{n+1}\right) \quad (i = 1 \sim 3) \qquad (10)$$

The skeleton length in the human model is invariable, which means:

$$d(P_i^n, P_j^n) = d(P_i^{n+1}, P_j^{n+1})(i, j = 1 \sim 3) \text{ and } i \neq j \qquad (11)$$

Using (10) to substitute P_i^{n+1} in (11), 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.

Finally we can calculate all the 3D feature points in human model corresponding to frame $n+1$ by the algorithm used on the first frame.

EXPERIMENT RESULTS

We have implemented a demo system on a personal computer. The video recordings in our lab were done with a single camera. Figure 8 shows one example sequence of a human sitting on a chair as seen from an oblique view. In the top row, the 16 feature points on the first image are marked by the user with a 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, our system constructed the sequence of 3D human skeleton in 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 were shot from the same viewpoint as the top ones. Then we rotate the camera rightward with 30°, shot six frames corresponding to the top ones, and showed them in the bottom row. As you see, the motion continuity and authenticity are embodied in this sitting sequence, which proves the robustness of the algorithm in ambiguity elimination. It means now we can see the person sit down from a more oblique view.

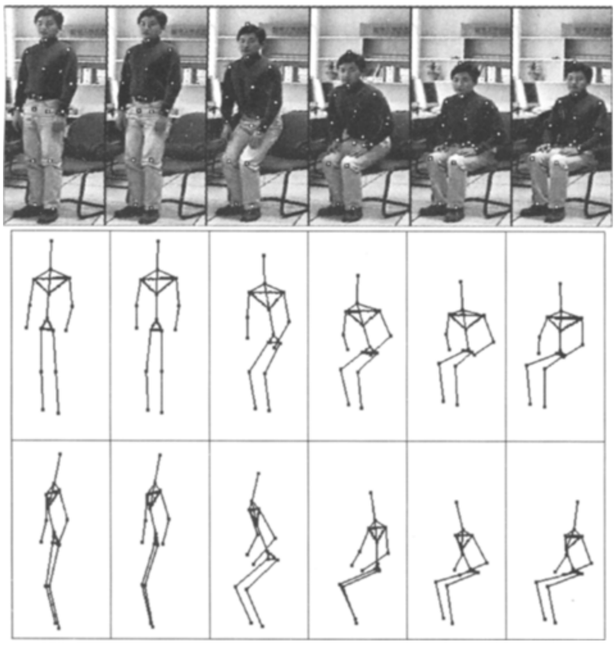

Fig.8 The experiment of human sitting down

CONCLUSION

This paper proposes a new technique to capture motion in video. It is a challenging domain to track human motion in the joint level and recover 3D motion information. Our contribution to this problem is that this approach does not pose any restrictions on human motion and finally sets up a 3D motion model in the perspective projection. To the best of our knowledge, this is the first demo system that can do such a challenging task and recover complex human motion with high accuracy. It is easy and straightforward from a user's point of view. The user only needs to mark the joints of the first frame and the computer does the rest. On the other hand, any video stream, whether it is a film, or any historical shot, such as Charlie Chaplin's walking and Karl Lewis' running, can be our material. Future work will concentrate on utilizing more 3D human skeleton motion knowledge to guide 2D feature tracking. We should also implement the texture mapping to visualize the human skeleton.

ACKNOWLEDGEMENTS

We would like to thank Tong Yiying, Zhou Kun, Wu Yi and Lu Xiqun for fruitful discussions.

References

Aggarwal, J. K., Cai, Q., 1997. Human Motion Analysis: A Review, *In*: Proc. of the IEEE Nonrigid and Articulated Motion Workshop. Piscataway, NJ,USA, p.90 - 102.

Akita, K., 1984. Image sequence analysis of real world human motion. *Pattern Recognition*, **17**(1):73 - 83.

Bharatkumar, A. G., Daigle, K. E., Pandy, M. G., et al., 1994. Low limb kinematics of human walking with the medial axis transformation. *In*: Proc. of IEEE Workshop on Motion of No-Rigid and Articulated Objects. Austin,TX,USA. p.70 - 76.

Bregler, C., Malik, J., 1998. Video Motion Capture, *In*: Proceeding of SIGGRAPH 98. USA.

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

Hogg, D., 1983. A program to see a walkirg person. *Image Vision Computing*, **5**(20):28 - 42.

Huang, T. S., Netravali, A. N.,1994. Motion and structure from feature correspondences: A review. *Proceedings of The IEEE*, **82**(2):252 - 268.

Ma Songde, and Zhang Zhengyou, 1998. Computer Vision: Compute Theory and Arithmetic Foundation. Science Press, Beijing, p.60 - 68.

Rohr, K., 1993. Incremental recogniton of pedestrians from image sequences. *In*: Proc. IEEE Conf. Comput. Vision and Pattern Recogn., Allerton Press, New York, USA. p.8 - 13.

Rourke, J.O., Badler, N.I., 1980. Model-based image analysis of human motion using constraint propagation. *IEEE PAMI*,**2**(6):522 - 536.