コンピュータビジョンと
イメージメディア 109－12
（1998．1．23）

人物の姿勢の3次元推定のための能動的空間インデクシング法

スヅハンシュセンワル†、大谷 淳‡、イリス フェリミン‡

† コロラド大学コロラドスプリング校

‡（株）ATR 知能映像通信研究所・第一研究室

〒619－02 京都府相楽郡精華町光台 2-2

email: semwal@redcloud.uccs.edu
{ohya,firis}@mic.atr.co.jp

あらまし 本稿では、複数カメラを用いて、仮想環境における人物を追跡する手法を提案する。提案手法は、前処理と人物の姿勢の3次元推定との、2つの主要処理モジュールから構成される。前処理において、3次元空間に対してあらかじめインデックスを与えておく。3次元推定処理においては、人物の全身におけるいくつかの特徴点の3次元位置が、前述のインデックス情報に従い求められる。実験により本提案手法の有効性を示す。

キーワード: インデクシング、仮想環境、3次元姿勢推定、特徴点、複数カメラ

An Active Space Indexing System for 3D Estimation of Human Postures

Sudhanshu Semwal†, Jun Ohya‡, Iris Fermin‡

† Univ. of Colorado at Colorado Springs

‡ ATR Media Integration and communications Research Laboratories

email: semwal@redcloud.uccs.edu
ohya@mic.atr.co.jp

Abstract We present a method for unencumbered tracking participants in a virtual environment using multiple cameras. The method consists of two main modules: preprocessing and 3D posture estimation. In the preprocessing step the 3D space (we call active space) is indexed in advance, and in the estimation step, the 3D positions of some significant points of the body of a participant, are obtained based on that indexed space.

Key words: indexing, virtual environment, 3D posture estimation, significant points, multiple cameras

1. Introduction

Virtual environments pose severe restrictions on algorithms for tracking and posture estimation due to the foremost requirement of real-time interaction. There are many related works in human posture estimation in a virtual environment. These can be broadly divided into two main categories: encumbering and unencumbering [1, 2, 3]. This is perhaps the most important choice in designing a virual environment as it determines the style and quality of interaction of a participant. In an encumbering virtual environments a participant must wear some tracking devices, for example, optical [4, 5], mechanical [6, 7], bio-controlled and magnetic trackers [8, 9, 10, 11, 12]. On the other hand, in an unencumbering system the participant must not wear any special devices [13]. Most of the camera-image based systems belong to this category.

Reasonable work space volume, robustness to measure errors and occlusion, confortness for participants are desired properties for a virtual environment. Mostly magnetic trackers have been used because they are relatively inexpensive, allow large work area, however can exhibit tracking errors up to 10 cms [14]. Optical and camera-image based tracking has problem of occlusion [2]. Our motivation is to develop unencumbering virtual enviroment using multiple cameras, in which the 3D position of a participant can be estimated.

1.1 Related Works

Camera-based techniques are well suited for developing unencumbering applications and many researchs have been done in the area of computer vision for the estimation of 3D positions using information from stereo vision and motion analysis [15, 16, 17], however problems related to camera calibration must be solved.

Jain *et.al.* [18] combine techniques of computer vision and graphics for processing the video-streams offline, non-iteractively, using color-based discrimation between frames to identify pixels of interest, which are then projected to find the voxels in a 3D grid space based on color of the region of interest. The marching cube is then used to construct the iso-surface of interest.

The virtual kabuki system [19, 20] uses thermal images from one camera and estimates the 2D joints positions in real time by using the silhouettes and 2D-distance transformations. Other points such as the knee and the elbow are estimated using genetic algorithm. Once extracted the posture is rendered to a kabuki-actor. Only 2D positions are estimated and problems due to lower thermal conductivity of clothes can appear.

Blob models are used in the Pfinder system [21]. This system uses one camera to capture a participant in a static environment. A multi-blob model is created, for the person based upon the color information, for estimating the 2D contours and postures. The estimation is based mainly on the color-changes. In the following sections, we present our method for estimating the 3D positions of a participant called Active-Space-Indexing as a part of our whole system for posture estimation, the Scan&Track system.

2. Scan&Track System

We are developing an unencumbering virtual environment called Scan&Track system which uses image sequences from multiple cameras. In Fig. 1 is detailed the modules of the Scan&Track system: body silhouette and significant points extraction, correspondences and tracking points, 3D indexing reconstruction (Active Space).

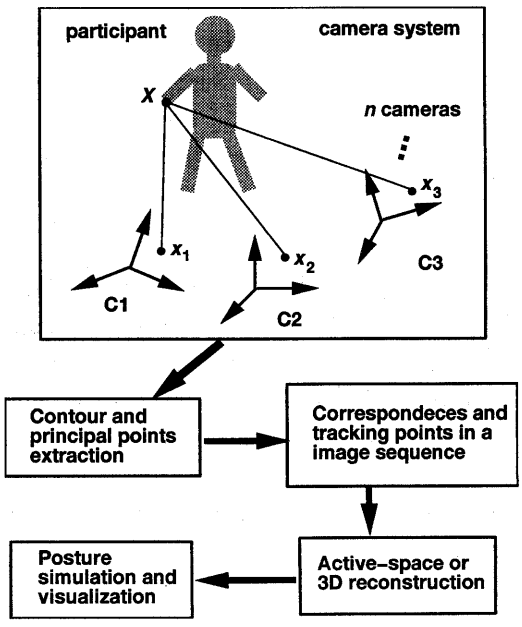

Figure 1: *Scan&Track system diagram*

In this paper, we assume that the contours, significant points and correspondences from the three

cameras C1, C2, and C3 are determined for the three images Im1, Im2, Im3, respectively. These contours would be used for estimating the 2D-location of the significant points, in the images from the three cameras. Let x_1, x_2, x_3 be the projection of a 3D significant point X, for the three camera frames, respectively. We call the triplet $\{x_1, x_2, x_3\}$ an imprint-set of the point X. If a 3D point is visible from multiple cameras, then the location of the imprint of the 3D point can be used to estimate the 3D position of the point. From now on, we assume that the imprint-sets are determined for each visible principal points.

2.1 Camera Calibration, Space-Linearity and Over-Constrained System

There are many works on reconstruction of the 3D position from motion which uses a minimal number of points to guarantee the uniquess of the reconstruction [15, 17, 22]. However, in these approaches they must to solve *a priori* the camera calibration problem. Furthermore, other approaches such affine reconstruction, self calibration have been proposed to avoid the camera calibration problem [23, 24], but many of the solutions are up to scale reconstruction. We plan to use several points, during preprocessing, and create our model based system on that, but then we do not require these points to be present for calibration or reference during the actual tracking.

Our motivation to use several points is to subdivide the active-space, so that only a few points are used locally to estimate the position during tracking, similar to the piece-wise design for surfaces and contours [25]. In our approach, by subdividing the 3D active-space into small, disjoint voxels or 3D cells, we use only a small number of points, which are the vertices of the 3D-voxel, for estimating a 3D position inside that voxel. In this way, only a few points are use to determine a 3D position of a point. The major advantage of using a set of voxels is that the non-linearity due to camera calibration is minimal. In particular, we can assume linear motion inside the voxel, and a 3D voxel will be also projected linearly on a camera image frame. Thus, the effects of camera distortion would be minimal. This assumption also allows the use of linear interpolation for estimating the position of a point during tracking.

2.2 Depth Information and Planar Slices

Consider the projections of two points p, q in Fig. 2. Depending upon the viewpoint from multiple cameras frames, the projections of these two points change, particulary their relationship changes as we move from left to right. Thus, the spatial information is not preserved in the projection plane. To recover the spatial information of a point from multiple views, we must to solve problems related with correspondences, and recovery depth from stereo cameras.

Next, we shall consider two points P, Q on a planar slice and the same set of multiples views, as shown in Fig. 3. Note that the spatial relationship of the projections of the points P and Q, for all planes remain same in the same planar slice hemi-sphere. Thus, it is easier to deal with points in a planar slice, as shown in Fig. 4. In Fig. 4 is shown that the point P is located above the line $L1$ and left to the line $L2$, and the Q is located above the line $L1$ and rigth to the line $L2$. Furthermore, slices can be stacked for estimating the depth information.

For simplicity, each slice in Fig. 5, has the 4×4 partitioning. Depending on the complexity of the scene, other partitioning are possible. Let a triplet (x_1, x_2, x_3) be the projection of a 3D point X on the cameras frames C_1, C_2, C_3, respectively. We can recover the depth information by processing all slices and searching for the voxel in which this point can be. This process is explained in the following section.

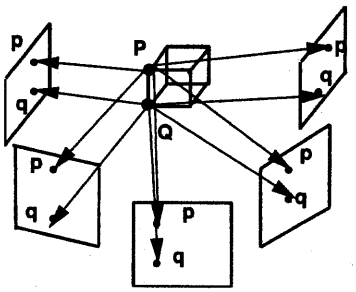

Figure 2: *Projection relation between camera frames*

3. Creating an Active Space

In this section is described the process to create an active-space indexing. A preprocessing is achieved to acquire the space coordinates of the planar grid slice model. Once the indexing is created, then we are able to determine the posture of a participant in a fixed stage.

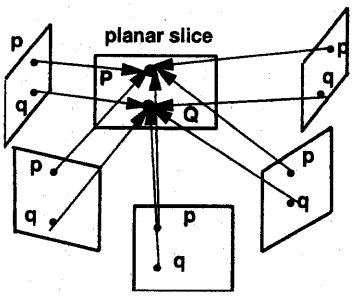

Figure 3: *Relation between projections in a planar slice*

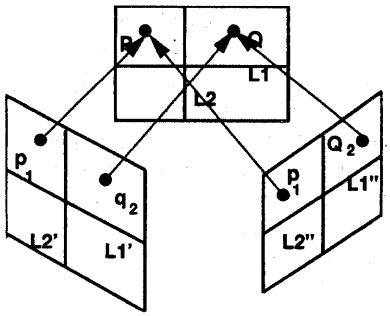

Figure 4: *Projection in a grid of a planar slice*

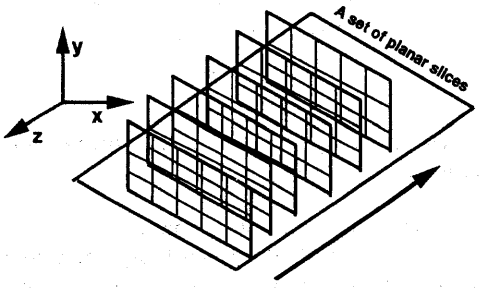

Figure 5: *Set of planar slices*

3.1 Preprocessing to Create an Active-Space Indexing

We fixed an stage which we will recorded in slices using the grid pattern shows in Fig. 6. This grid is 47 × 47 points pattern. To record the slices we use three cameras at the same time. At each time the grid is moved forward to backward at constant distance, in this case the translation is 10 cms. (Fig. 5). The total number of slices recorded for an stage are determined beforehand, depending in the size of the stage.

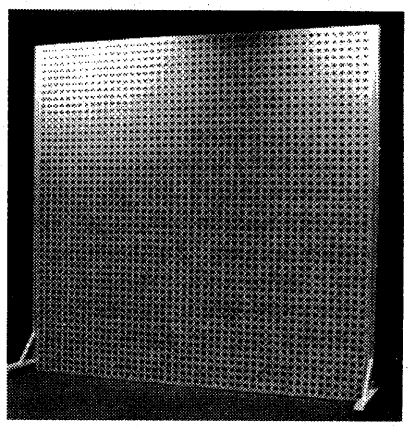

Figure 6: *Grid pattern 47× 47 points, which can be moved and its dimension is 240cm×240cm*

To create the indexing mechanism, we specify a set of horizontal and vertical lines to cover the rows and columns of the grid pattern. During the prepocessing, we do the following for each slice:

1. For all the three camera-images, four corner points on the grid parttern are picked using a mouse. These corner points define a 2D-extent of the projected grid pattern.

2. Following the x and y directions, from left to right and bottom to up, the horizontal and vertical lines are estimated, by determine the end points of each line (black points).

Thus, all straight lines have been defined then we can determined the intersection of vertical lines with horizontal lines to find the location of the grid-circles.

3.2 Finding a 2D-index during 3D Tracking

For every slice we find a set of horizontal and vertical lines during the preprocessing. During tracking, given a pixel coordinate of a 2D point on a slice, we can quickly find the grid-index using these horizontal and vertical lines. First, we check is the point is outside of the area defined by the four corner points of each slice recorded for a specific camera. If the point is inside then we find the grid-index for the given pixel by searching the set of vertical lines for the x-index, and horizontal lines for the y-index. Since the lines are specified from left to right, we find two consecutive vertical lines $\mathbf{v}$ and $\mathbf{v_{+1}}$ such that the given point is on or between the two lines. A similar algorithm is used to determine the y-index, by finding two consecutive horizontal lines, $\mathbf{h}$ and $\mathbf{h_{+1}}$ such the given point is on or above line $\mathbf{h}$, and below line $\mathbf{h_{+1}}$. For our experiment we have 47 horizontal and 47 vertical lines.

To estimate the location of a 3D point given its imprint-set $\{x_1, x_2, x_3\}$, we have implemented the following algorithm.

3.3 3D-Voxel Estimation by using Active-Space Indexing

Given a triplet of points $\{x_1, x_2, x_3\}$ corresponding to a 3D point, for the three cameras frames, respectively. We perform the following for every slice:

1. For an imprint-point, use the vertical lines collected for the corresponding camera frame during the preprocessing, to find the 2D horizontal x-index.

2. Perform the same as in step 1 to determine the y-index by using the set of horizontal lines for each camera frame, respectively.

3. Perfom the above two steps for each given triplet for every slice and for the three camera frames, respectively. Let $I1$, $I2$ and $I3$ be the indices for the left, center and right cameras, respectively.

For every triplet $\{x_1, x_2, x_3\}$, we collect $I1$, $I2$, and $I3$ points for each slice as shown in Fig. 7. We assume that the point X is between to slices k and $k+1$. These three indices define a triangle on every slice. Simple ray-optics suggests that the area would be decreasing as the ray converge at the point X and then start increase as the rays diverge. We have implemented a linear algorithm to determine the slice with the minimum triangle area. Let k be the left slice with triangle area A_L, and $k+1$ the right slice with triangle area A_R as the rays diverge. We have that the slice k is the nearest to the point X so the points's 3D cell index would be (i, j, k). We chose $i = I1_x$ and $j = I1_y$ in our implementation. Here i could also be an average of x-indices of $I1$, $I2$ and $I3$ for slice k. It can be notice that the z coordinate is obtained since we know the displacement between slices.

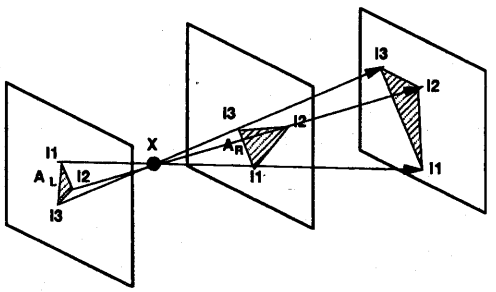

Figure 7: *Finding the 3D-voxel*

4. Experiments and Results

In the preprocessing step we recorded slices by using three cameras at the same time, which are fixed during the tracking. Setting the grid pattern an initial position, the subsequent slices are recorded moving the grid by 10 cms from the previous position. Figures 8 and 9 are two examples of this preprocessing. In these examples, eight slices were recorded for each camera. In Figures 8 and 9 also the estimation of the 3D voxels are shown. From three views of a participant, manually three imprints $\{x_1, x_2, x_3\}$ are provided as shown in Figs. 10 and 11. After a 3D voxel of an imprint is located, then the 3D information is determined by linear approximation. This 3D information can be rendered to a synthetic actor for visualization of the results. We can specify as many triplets as desired.

The technique works because there is enough shift in the projection of the points due to the cameras positioning that the depth discrimination is possible. Unless the three camera angles are identical or the grid is very wide, it is highly unlikely that more than two slices have the same area formed by $I1, I2, I3$. In fairness to our method, this would mean that the resolution of the sliced

active space needs to be increased, or the camera angles need to change. It can be concluded that given a imprint set, a unique 3D cell index can be found.

As the number of slices and the number of lines increase in future applications, we can apply binary search based on the area of the triangle on the slices.

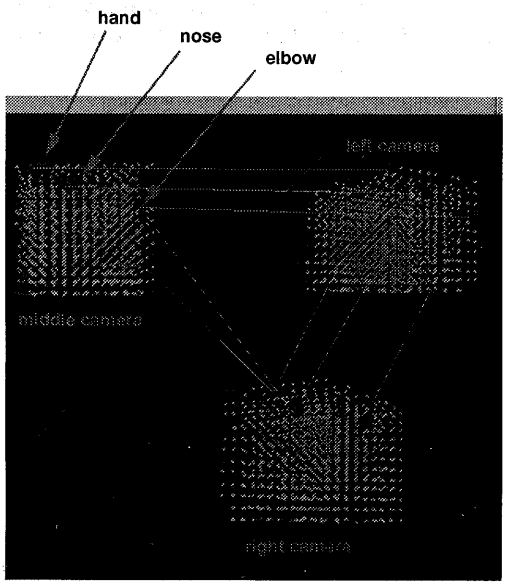

Figure 8: *Example 1, processed slices and voxel estimation*

5. Evaluation of the Active-Space Indexing

The accuracy of the system is related to the correspondence of the points. Once the triplet (x_1, x_2, x_3) is given, an active space voxel can be obtained in constant time using the active space indexing mechanism since the number of slices is constant. Further refinements of the estimated position are possible to achieve higher accuracy by using another set of cameras.

There is a different linear interpolation which will considerably improve the accuracy of our estimated for a point X within the voxel (i, j, k). Notice that the middle camera is placed perpendicular to z-axis and parallel to the grid during the preprocessing. The x and y coordinates of S can be further refined by using the relationship between $I2$ and x_2 in the middle camera frame and

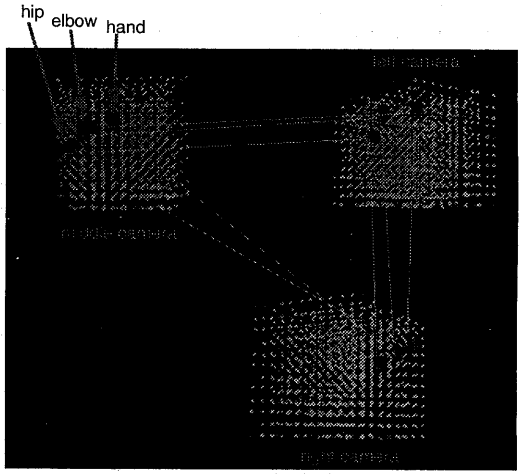

Figure 9: *Example 2, processed slices and voxel estimation*

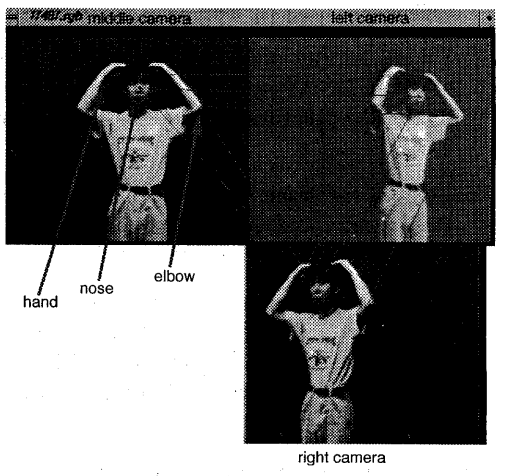

Figure 10: *Example 1, given imprints from three camera frames*

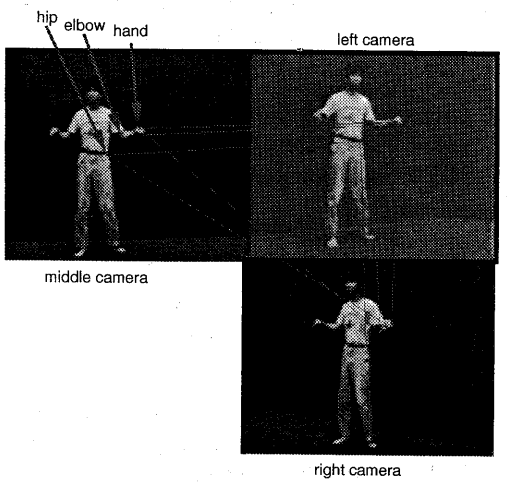

Figure 11: *Example 2, given imprints from three camera frames*

the assumption of space linearity within the voxel as discussed earlier. For instance, we know that a voxel occupies a 5× 5×10 cm in the 3D space. If a 2D cell of the middle camera contains 100 pixels, in a 10×10 array, then we expected the accuracy to be around 0.5 cm along x and y axes, respectively. Similarly, $I1$ and x_1 from the left camera and $I3$ and x_3 for the right camera, can further refine the z coordinate of point X. The accuracy of an estimation depends upon the number of pixels inside the cells $I1$, $I2$, and $I3$ and the physical size of the voxel (i, j, k). Since 3D space can be thinly sliced further, and multiple set of cameras can also be added, we expected to considerably increase the accuracy of the active-space indexing, when this interpolation method is implemented.

The active space indexing system is extremely robust as this mechanism can always find a 3D cell for given imprint sets.

Registration: This is related to the *swimming* problem encountered in virtual environments. Virtual objects, when placed on a tracked-position, tend to swin because of the slight variations in estimating the 3D position of the tracked-position. This leads to the well known *swimming* effect which is very evident in systems using magnetic trackers. When we use multiple cameras, the same corresponding points on the image will produce the same result, because the camera and slices remain in the same position for the duration of the tracking. However, although rare, voltage fluctations can account for a change in size of the image, and can create *swimming* effect. In addition, mechanical vibrations, could cause the camera image to jiter. In fairness, we do not know of any tracking device where voltage and mechanical vibrations would not have an effect on tracking.

6. Conclusions

In the Scan&Track system, we have provided a framework for unencumbered tracking based upon multiple image sequences. We implemented a new algorithm for 3D estimation of human postures called active-space indexing method. The imprint sets of the points, which are three 2D points corresponding to the same 3D point in three image frames, respectively, are the input to the active-space indexing. Assuming that we have this imprint set the active-space indexing determines the 3D cell index of that point in constant execution time.

The accuracy of the 3D position estimation depends on the estimation of the 3D voxels, which depends on the resolution of the image. Now we are implementing the construction of the voxels by image processing technique. Active-space indexing avoids the calibration of the cameras, since the grid pattern give us the infomation related to the 3D coordinates of the space.

Acknowledgments

The first author would like to thank Dr. R. Nakatsu, President of ATR Media Integration & Communications (MIC) Research Laboratories for making this research possible. Thanks are also due to Dr. K. Masse, MIC, Head of Department 2, for discussion on the Pfinder method being used in his laboratory. Special thanks for all members of MIC Department 1, for variety of discussion and help.

References

1 Meyer, K., Applewhite, H.L., Biocca, F.A., A survey of position trackers, *Presence*, 1(2), pp. 173-200, 1992.

2 Sturman, D.J., Zeltzer, D., A survey of glove-based input, *IEEE Computer Graphics*,14(1), pp. 30-39, 1994.

3 Speeter, T.H., Transforming human hand motion for tele-manipulation, *Presence*, 1(1), pp. 63-79, 1992.

4 Newby, G.B., Gesture recognition based upon

statitical similarity, *Presence*, **3**(3), pp. 236-244, 1994.

5 Gottschalk, S., Hughes, J.F., Autocalibration for virtual environments tracking hardware, *Proceedings of SIGGRAPH*, pp. 65-71, 1993.

6 Rashid, R.F., Towards a system for the interpretation of moving light display, *IEEE Transaction on Pattern Analysis and Machine Intelligence*, **2**(6), pp.574-581, 1980.

7 Sutherland, I.E., A head-mounted three dimensional display, *ACM Joint Computer Conference*, **33**(1), pp. 757-764, 1968.

8 Brooks, F.P., Ouh-Young M.J., Batter, J.J., Kilpatrick, P.J., Project GROPE - Haptic display for scientific visualization. *Proceedings of SIGGRAPH*, **24**(4), pp. 177-185, 1990.

9 Biosignal processing and biocontroller: Technology overview, *Video Presentation by Control Systems, Inc.* 430 Cowper Street, Palo Alto, CA 94301, USA.

10 Semwal, S.K., Hightower, R., Stansfield, S., Closed form and geometric algorithms for real-time control of an avatar, *Proceedings of IEEE VRAIS96*, pp. 177-184, 1996.

11 Stansfield, S., Miner N., Shawver, Rogers, D., An application of shared virtual reality to situational training, *Proceedings of IEEE VRAIS95*, pp. 156-161, 1995.

12 Boulic R., Capin T.K., Huang Z., Kalra P., Lintermann B., Magnenat Thalmann, N., Moccozet L., Molet T., Pandzic I.S., Saar K., Schnitt A., Shen J., Thalmann D., The HUMANOID environment for interactive animation of multiple deformable human character, *Eurographics'95*, 1995.

13 Krueger, M.W., *Articial Reality II*, Addison Wesley Publishing Company, Reading, MA, 1991.

14 State, A., Hirota, G., Chen, D.T., Garret, W.F., Livingston, M.A., Superior augmented reality registration by integrating landmark tracking and magnetic tracking, *Proceedings of SIGGRAPH*, pp. 429-438, 1996.

15 Faugeras, O., *Three-Dimensional Computer Vision: A Geometric Viewpoint*, MIT Press, Cambridge, MA, 1996.

16 Grimson, W.E., *Object Recognition by Computer: The Role of geometric Constraints*, MIT Press, Cambridge, MA, 1990.

17 Maybank, S., *Theory of Reconstruction from Image Motion*, Springer-Verlag, 1993.

18 Moezzi, S., Katkere, A., Kuramura, D.Y., Jain, R., Immersive Video, *Proceedings of IEEE VRAIS96*, pp. 17-24, 1996.

19 Iwasawa, S., Ebihara, K., Ohya, J., Morishima, S., Real-Time estimation of human body posture from monocular thermal images, *International Conference on Computer Vision and Pattern Recognition*, pp. 15-20, 1997.

20 Ohya J., Kishino F., Human posture estimation from multiple images using genetic algorithms, *Proceedings of 12th IAPR*, pp. 750-753, 1994.

21 Wren C., Azarbayejani A., Darrell T., Pentland, A., Pfinder: Real-time tracking of the human body, *SPIE Conference on Integration Issues in Large Commercial Media Delivery Sytems*, **2615**, 1995.

22 Ullman, S., *The Interpretation of Visual Motion*, MIT Press, Cambridge, MA, 1979.

23 Zhang Z., Deriche, R., Faugeras, O., Luong, Q., A robust technique for matching two uncalibrated images though the recovery of the uknown epipolar geometry, *Artificial Intelligence*, **78**, pp. 87-119, 1995.

24 Sapiro, L., Zisserman, A., Brady, M., 3D Motion recovery via affine epipolar geometry, *International Journal of Computer Vision*, **16**, pp. 147-182, 1995.

25 Farin, G., *Curves and Surfaces for Computer Aided Geometric Design*, Academic Press, San Diego, USA, 1992.