Vision-Based Registration for Augmented Reality with Integration of Arbitrary Multiple Planes

Yuko Uematsu and Hideo Saito

Keio University, Dept. of Information and Computer Science, Yokohama, Japan
{yu-ko, saito}@ozawa.ics.keio.ac.jp
http://www.ozawa.ics.keio.ac.jp/Saito

Abstract. We propose a novel vision-based registration approach for Augmented Reality with integration of arbitrary multiple planes. In our approach, we estimate the camera rotation and translation by an uncalibrated image sequence which includes arbitrary multiple planes. Since the geometrical relationship of those planes is unknown, for integration of them, we assign 3D coordinate system for each plane independently and construct projective 3D space defined by projective geometry of two reference images. By integration with the projective space, we can use arbitrary multiple planes, and achieve high-accurate registration for every position in the input images.

1 Introduction

Augmented Reality (AR) / Mixed Reality (MR) is a technique which can superimpose virtual objects onto the real 3D world. We can see the virtual objects as if they really exist in the real world, so AR provide users with more effective view [1,2]. One of the most important issues for AR is geometrical registration between the real world and the virtual world. In order to achieve correct registration, accurate measurements of the camera rotations and translations (corresponding to the user's view) are required.

For the measurements, some kind of sensors such as magnetic or gyro sensors may be used. The registration by such sensors is stable against a change in light conditions and is especially effective when a camera moves rapidly. However, the rotations and translations obtained from sensors are not accurate enough to achieve perfect geometrical registration. Furthermore, the use of sensors has some limitations in practice: user's movable area, perturbation caused by the environment, and so on. On the other hand, vision-based registration does not require any special devices expect cameras. Therefore an AR system can be constructed easily. This kind of registration relies on the identification of features in the input images. Typically artificial markers placed in the real world [3,4], prepared model [5,6,7], and / or natural features are used for the registration. Related works based on natural features have used various features: feature points [8], edges or curves. However, it is also true that few features are available for registration in the real world. Therefore, it is important how to use the few features effectively.

We focus on the planar structures, which exist in the real world naturally without artificial arrangement and put appropriate restrictions to the natural feature points. Since

F. Roli and S. Vitulano (Eds.): ICIAP 2005, LNCS 3617, pp. 155–162, 2005.

a lot of planes exist in various environments, such as indoor walls, floors, or outdoor wall surfaces of buildings etc., using these structures is very reasonable approach. Using multiple planes, we can overlay virtual objects onto wider area than using only 1 plane and the accuracy is also improved. Furthermore, using multiple planes which are in arbitrary positions and poses, we can use most planes existing in the real world. Therefore, using arbitrary multiple planes is valuable approach for the future AR applications.

Registration using planes has attracted attention recently, and Simon et al. have proposed related AR approaches [9,10,11]. In [9], they track feature points existing on a plane in the real world, estimate a projection matrix for each frame by the tracked points and overlay virtual objects onto the real input images. They also implement registration using multiple planes. In [10], they estimated the projection matrix by multiple planes which are perpendicular to the reference plane using an uncalibrated camera. In [11], they estimated the projection matrix using a calibrated camera from multiple planes of arbitrary positions and poses. In their method, the geometrical relationship among these planes and motion of the camera are calculated by bundle adjustment which is carried out over all frames.

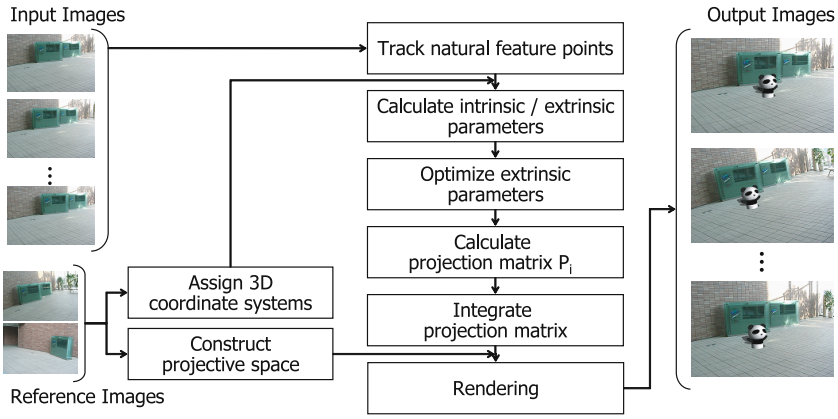


Fig. 1. Overview of the proposed method

In this paper, we propose a vision-based registration approach, which can use arbitrary multiple planes without any information on physical relationship of the planes, estimate the camera motion frame by frame, and achieve high registration accuracy. In order to use arbitrary multiple planes, we assign 3D coordinate systems for each plane independently. Since geometrical relationship among those planes is unknown, we construct "projective 3D space" for integrating those planes. This integration is main contribution of our approach. Fig.1 describes an overview of our approach. Firstly, the input image sequence in which n planes exist is taken by an uncalibrated hand-held video camera. Next, we compute each projection matrix (corresponding to the camera motion) from each plane independently. Then, the projective space is constructed by 2 reference images, which are taken at 2 views, those matrices are integrated with the space, so one camera motion is obtained. Lastly, virtual objects are overlaid onto the input images according to the camera motion.

2 Calculation of Projection Matrix

As mentioned previously, for overlaying virtual objects, accurate measurement of the camera rotations and translations (extrinsic parameters of the camera) is required. Moreover, for using an uncalibrated camera, we also need to estimate intrinsic parameters. In our approach, we assign a 3D coordinate system for each plane so that each plane is set to $Z = 0$ (see sec.2.1). Then we compute intrinsic and extrinsic parameters from a homography between the 3D real plane and the input image plane in order to obtain a projection matrix [9].

A 3D coordinate system is related to a 2D coordinate system by 3×4 projection matrix $\boldsymbol{P}$. Thus, each 3D coordinate system designed for each plane is also related to the input images by each projection matrix. If each plane's Z coordinate is set to 0, a homography $\boldsymbol{H}$ also relates between each plane and the input images.

$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \simeq \boldsymbol{P} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} \simeq \begin{bmatrix} p_{11} & p_{12} & p_{14} \\ p_{21} & p_{22} & p_{24} \\ p_{31} & p_{32} & p_{34} \end{bmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \simeq \hat{\boldsymbol{P}} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \simeq \boldsymbol{H} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \quad (1)$$

This 3×3 matrix (called $\hat{\boldsymbol{P}}$), which is the deleted third column of $\boldsymbol{P}$, is equivalent to a planar homography $\boldsymbol{H}$. The deleted column vector can be estimated by this $\boldsymbol{H}$. When the homography is calculated, we can obtain the projection matrix from it. In particular, we think dividing into intrinsic parameters $\boldsymbol{A}$, and extrinsic parameters $\boldsymbol{R}$, $\boldsymbol{t}$.

$$\boldsymbol{P} = \boldsymbol{A}\,[\boldsymbol{R} \mid \boldsymbol{t}] = \boldsymbol{A}\,[\boldsymbol{r_1}\ \boldsymbol{r_2}\ \boldsymbol{r_3}\ \boldsymbol{t}] \quad (2)$$

$$\hat{\boldsymbol{P}} = \boldsymbol{A}\,[\boldsymbol{r_1}\ \boldsymbol{r_2}\ \boldsymbol{t}] = \boldsymbol{H} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \quad (3)$$

2.1 Assigning of 3D Coordinate Systems

For using the multiple planes whose geometrical relationship is unknown, we assign a 3D coordinate system for each plane in the 3D real world independently. Each plane's Z coordinate is set to 0. This is for computing a homography and estimate a projection matrix from it.

2.2 Calculation of Homography

In order to estimate the intrinsic and the extrinsic parameters, we calculate homographies between each 3D plane ($Z = 0$) and the input image plane. Natural feature points existing on the 3D planes are tracked by KLT-feature-tracker [12] and used for computing the homography. The Homography is calculated for each 3D plane independently, so homographies and projection matrices are computed to the number of the 3D planes respectively.

2.3 Estimation of Intrinsic Parameters

By fixing the skew to 0, the aspect ratio to 1 and the principal point to the center of the image, the intrinsic parameters can be defined as in eq.(4), and the relationship to homography is represented by eq.(5). Then, we only have to estimate the focal length f.

$$\boldsymbol{A} = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \qquad \begin{matrix} (c_x, c_y) : \text{ principal point} \\ f \quad : \text{ focal length} \end{matrix} \tag{4}$$

$$\boldsymbol{A^{-1}H} = [\boldsymbol{r_1}\ \boldsymbol{r_2}\ \boldsymbol{t}] \tag{5}$$

According to the property of rotation matrix $\boldsymbol{R}$, that is the inner product of $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ is equal to 0, we can calculate the focal length f.

$$f^2 = \frac{(h_{11} - c_x h_{31})(h_{12} - c_x h_{32}) + (h_{21} - c_y h_{31})(h_{22} - c_x h_{32})}{-h_{31} h_{32}} \tag{6}$$

2.4 Estimation of Extrinsic Parameters

The extrinsic parameters of a camera consist of a rotation matrix $\boldsymbol{R}$ and a translation vector $\boldsymbol{t}$. Since $\boldsymbol{r_1}$, $\boldsymbol{r_2}$ (the first and second column vectors of $\boldsymbol{R}$) and $\boldsymbol{t}$ are already known, we should estimate only $\boldsymbol{r_3}$. Then, also according to the property of $\boldsymbol{R}$, that is the cross product of $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ becomes $\boldsymbol{r_3}$, we compute $\boldsymbol{r_3}$. Therefore, $\boldsymbol{R}$ is

$$\boldsymbol{R} = [\boldsymbol{r_1}\ \boldsymbol{r_2}\ (\boldsymbol{r_1} \times \boldsymbol{r_2})] = [\boldsymbol{r_1}\ \boldsymbol{r_2}\ \boldsymbol{r_3}] \tag{7}$$

Furthermore, the extrinsic parameters are optimized by the steepest descent method in order to improve its accuracy. We optimize errors ϵ between the initial point $\boldsymbol{x_p}$ projected by the calculated projection matrix and the point $\boldsymbol{x_h}$ by homography.

$$\epsilon = (\boldsymbol{x_h} - \boldsymbol{x_p}) \tag{8}$$

3 Integration of Projection Matrices

Our main contribution is using multiple planes whose geometrical relationship is unknown and existing in the real world arbitrarily. After assigning 3D coordinate system for each plane independently and calculating projection matrices, we integrate those projection matrices in order to use the information of multiple planes. Each projection matrix is reliable around its corresponding plane, however, as the position of a virtual object moves away from each plane, the accuracy becomes lower. Therefore, we integrate the projection matrices to compute one accurate matrix over the whole image. However, it is impossible to integrate them simply because each projection matrix is from different 3D coordinate system. Then, we construct projective 3D space based on the projective geometry of two reference images.

If there are n planes, the relationship among each 3D coordinate system assigned for each plane, the projective space, and the input images is shown in fig.2. Since this

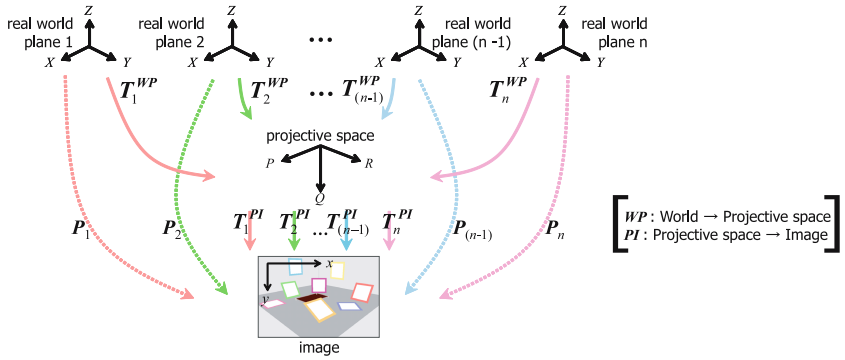


Fig. 2. Relationship among 3 coordinate sysmtes

projective space is defined by only projective geometry of two reference images, it is independent from the 3D coordinate system of the planes. Thus, $\boldsymbol{T}_{\boldsymbol{k}}^{\boldsymbol{PI}}$ are the transformation matrices between the common coordinate sysmtes (projective space → input images), and we can integrate those matrices. In this way, using projective space, we can extract the common transformation matrices from the porjection matrices calculated from different 3D coordinate sysmtems and integrate arbitrary multiple planes. This integration of multiple planes, which are different poses and exist in various positions, allows accurate augmentation onto wider area than using only 1 plane. The detail will be described below.

3.1 Construction of Projective Space

The projective space used in our method is based on the concept of "projective reconstruction" as shown in fig.3. By epipolar geometry between the reference images (cameras), the relationship between the projective space and the reference images is as follows respectively,

$$\boldsymbol{P}_{\boldsymbol{A}} = [\mathbf{I} \mid \mathbf{0}], \qquad \boldsymbol{P}_{\boldsymbol{B}} = [\boldsymbol{M}\boldsymbol{e}_{\boldsymbol{B}}], \qquad \boldsymbol{M} = -\frac{[\boldsymbol{e}_{\boldsymbol{B}}]_{\times}\boldsymbol{F}_{\boldsymbol{AB}}}{\|\boldsymbol{e}_{\boldsymbol{B}}\|^2} \tag{9}$$

where $\boldsymbol{F}_{\boldsymbol{AB}}$ is a fundamental matrix of image A to B, and $\boldsymbol{e}_{\boldsymbol{B}}$ is an epipole on the image B. Consider $\boldsymbol{C}_{\boldsymbol{p}}$ as a point in the projective space, $\boldsymbol{C}_{\boldsymbol{A}}(u_A, v_A)$ as on the image A, $\boldsymbol{C}_{\boldsymbol{B}}(u_B, v_B)$ as on the image B, we can write

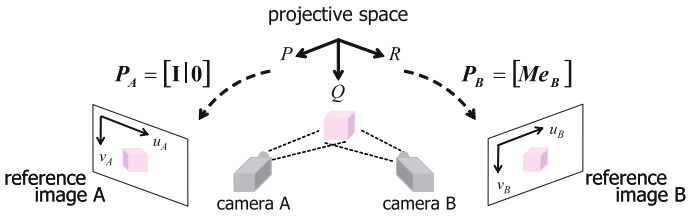


Fig. 3. Projective space by projective reconstruction

$$\boldsymbol{K}\boldsymbol{C}_{\boldsymbol{P}} = \left[\ \boldsymbol{p}^{\boldsymbol{1}}_{\boldsymbol{A}} - u_A \boldsymbol{p}^{\boldsymbol{3}}_{\boldsymbol{A}},\ \boldsymbol{p}^{\boldsymbol{2}}_{\boldsymbol{A}} - v_A \boldsymbol{p}^{\boldsymbol{3}}_{\boldsymbol{A}},\ \boldsymbol{p}^{\boldsymbol{1}}_{\boldsymbol{B}} - u_B \boldsymbol{p}^{\boldsymbol{3}}_{\boldsymbol{B}},\ \boldsymbol{p}^{\boldsymbol{2}}_{\boldsymbol{B}} - v_B \boldsymbol{p}^{\boldsymbol{3}}_{\boldsymbol{B}}\ \right]^\top \boldsymbol{C}_{\boldsymbol{P}} = \boldsymbol{0} \quad (10)$$

$\boldsymbol{p}^{\boldsymbol{i}}$ is the i th column vector of $\boldsymbol{P}$. Then, we obtain $\boldsymbol{C}_{\boldsymbol{p}} \simeq [p, q, r, 1]^\top$ by the singular value decomposition of $\boldsymbol{K}$.

3.2 Calculation of $\boldsymbol{T}^{\boldsymbol{WP}}_{\boldsymbol{k}}$

Consider $\boldsymbol{C}_{\boldsymbol{W}}(X, Y, Z)$ as a point on the kth plane in the real world and $\boldsymbol{C}_{\boldsymbol{P}}(P, Q, R)$ as in the projective space, the relationship between the two coordinate systems is

$$\boldsymbol{C}_{\boldsymbol{P}} \simeq \boldsymbol{T}^{\boldsymbol{WP}}_{\boldsymbol{k}} \boldsymbol{C}_{\boldsymbol{W}} \quad (11)$$

Since $\boldsymbol{T}^{\boldsymbol{WP}}_{\boldsymbol{k}}$ is 4×4 matrix, we can compute this matrix by the 5 (or more) corresponding points, in which any combination of 3 points must not be colinear and 4 points must not also be coplanar.

3.3 Calculation of $\boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{k}}$

When $\boldsymbol{T}^{\boldsymbol{WP}}_{\boldsymbol{k}}$ is known, we can compute $\boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{k}}$ by eq.(12), so $\boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{1}} \sim \boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{n}}$ are computed for each plane as fig.2.

$$\boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{k}} = \boldsymbol{P}_{\boldsymbol{k}} (\boldsymbol{T}^{\boldsymbol{WP}}_{\boldsymbol{k}})^{-\boldsymbol{1}} \quad (12)$$

As described previously, these matrices represent the common transformation (projective space $\rightarrow$ input images). Therefore, we can integrate these matrices. For the integration, we propose two approaches.

Maximum likelihood estimation. Using $\boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{k}}$, set of corresponding points between the projective space and the input images can be calculated for each plane. Then, we calculate $\boldsymbol{T}^{\boldsymbol{PI}}$ by the maximum likelihood estimation method using those points. This means that, if n planes exist and m set of corresponding points are calculated every plane, the integration expression becomes as follows.

$$\begin{bmatrix} X_{11} & Y_{11} & Z_{11} & 1 & 0 & 0 & 0 & 0 & -X_{11}x_{11} & -Y_{11}x_{11} & -Z_{11}x_{11} \\ 0 & 0 & 0 & 0 & 1 & X_{11} & Y_{11} & Z_{11} & -X_{11}y_{11} & -Y_{11}y_{11} & -Z_{11}y_{11} \\ & & & & & & \vdots & & & & \\ X_{nm} & Y_{nm} & Z_{nm} & 1 & 0 & 0 & 0 & 0 & -X_{nm}x_{nm} & -Y_{nm}x_{nm} & -Z_{nm}x_{nm} \\ 0 & 0 & 0 & 0 & 1 & X_{nm} & Y_{nm} & Z_{nm} & -X_{nm}y_{nm} & -Y_{nm}y_{nm} & -Z_{nm}y_{nm} \end{bmatrix} \begin{bmatrix} t^{PI}_{11} \\ t^{PI}_{12} \\ \vdots \\ t^{PI}_{33} \\ t^{PI}_{34} \end{bmatrix} = \begin{bmatrix} x_{11} \\ y_{11} \\ \vdots \\ x_{nm} \\ y_{nm} \end{bmatrix} \quad (13)$$

Merging with weights. In order to integrate $\boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{k}}$ in consideration for that each projection matrix is reliable around each plane, we employ the following integration form.

$$\boldsymbol{T}^{\boldsymbol{PI}} = \frac{1}{n} \left[\ w_1\ \cdots\ w_n\ \right] \left[\ \boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{1}},\ \ldots,\ \boldsymbol{T}^{\boldsymbol{PI}}_{\boldsymbol{n}}\ \right]^\top \quad (14)$$

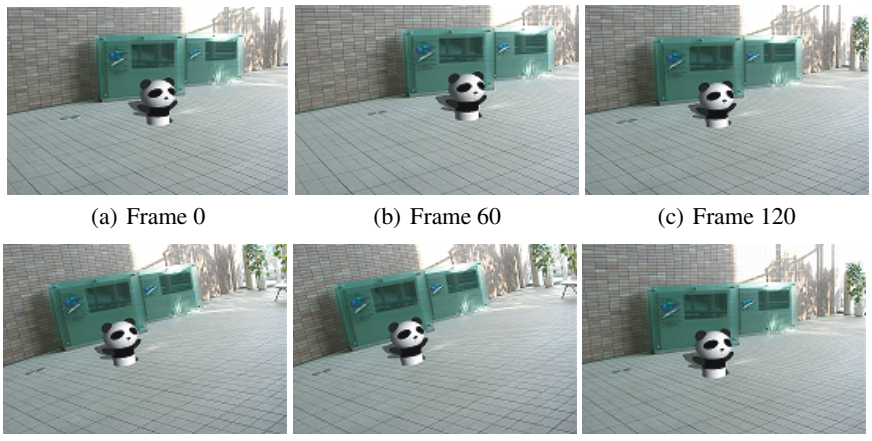
w_k is a weight parameter which is defined according to the distance from each plane to the overlaid position. This integration enables effective augmentation depending on the overlaid position. We use this one for the experiments in the next section (sec.4).

4 Experimetal Results

In this section, the experimental results are shown to prove the availability of the proposed method. We implement the AR system based on our method using only a PC (OS:Windows XP, CPU:Intel Pentium Ⅳ 3.20GHz) and a CCD camera (SONY DCR-TRV900). The input image's resolution in all the experiments is 720×480 pixels, and graphical views of a virtual object are rendered using OpenGL.

The overlaid result images produced by the augmentation are shown in fig.4. In this sequence, the 3 planes (a floor, a front display, and a back wall) are used for registration and a virtual object (a figure of panda) is overlaid on the floor plane. As shown in the figure, our approach can superimpose a virtual object onto the input images successfully.

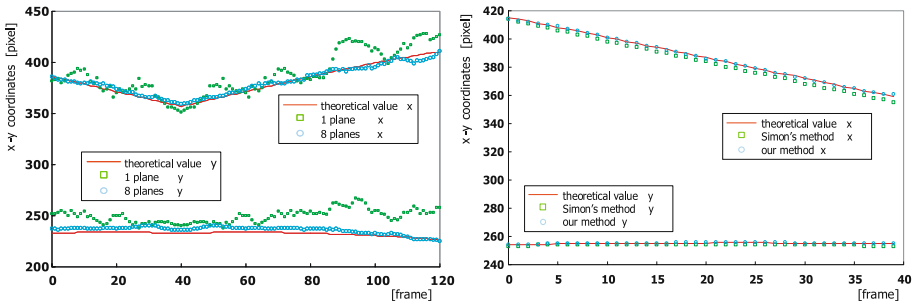
Next, in order to evaluate the registration accuracy in our method, we perform the same implementation for the synthesized images rendered with OpenGL. Since we have



(a) Frame 0 (b) Frame 60 (c) Frame 120

(d) Frame 180 (e) Frame 240 (f) Frame 300

Fig. 4. Overlaid image sequence of a virtual object



(a) 1 plane and 8 planes. (b) Related method and our method.

Fig. 5. Comparison of x-y coordinates accuracy with theoretical value

to know the exact position and pose of a camera to evaluate accuracy, we use the synthesized images. Fig.5(a) shows that the result by 8 planes has less registration errors and jitters than using only 1 plane, in spite of no information about the relationship of the planes. This suggests that increasing the number of planar structures in the scene using the proposed method can improve the registration accuracy.

We also evaluate the proposed method by comparing with one of related works by Simon [10], in which multiple planes need to be perpendicular to the reference plane (that is one of multiple planes). For the comparison, we apply the images, in which 3 orthogonal planes exist, to Simon's method and our method, and evaluate the registration accuracy. The result of the evaluation is shown in fig.5(b). Even though our method does not require any geometrical information of the plane, our method achieves almost the same accuracy with their method, in which the planes need to be perpendicular to the reference plane.

5 Conclusion

A geometrical registration method for Augmented Reality with an uncalibrated camera based on multiple planes has been proposed in this paper. The planes do not need to be perpendicular to each other. This means that any planes at arbitrary positions and poses can be used for registration. Furthermore the registration can be performed frame by frame without using all frames in the input image sequence. Thus we can construct the AR system easily, and overlay virtual objects onto the image sequence correctly.

References

1. Azuma, R.T.: A survey of augmented reality. Presence (1997) 355–385
2. Azuma, R.T.: Recent advances in augmented reality. IEEE Computer Graphics and Applications 21 (2001) 34–47
3. Billinghurst, M., et al.: Magic book: Exploring transitions in collaborative ar interfaces. Proc. of SIGGRAPH 2000 (2000) 87
4. Satoh, K., et al.: Robust vision-based registration utilizing bird's-eye with user's view. In: Proc. of the ISMAR. (2003) 46–55
5. Drummond, T., Cipolla, R.: Real-time tracking of complex structures with on-line camera calibration. In: Proc. of the BMVC. (1999) 574–583
6. Comport, A.I., Marchand, E. Chaumette, F.: A real-time tracker for markerless augmented reality. In: Proc. of the ISMAR. (2003) 36–45
7. Klein, K., Drummond, T.: Sensor fusion and occlusion refinement for tablet-based ar. In: Proc. of the ISMAR. (2004) 38–47
8. Chia, K.W., Cheok, A., Prince, S.J.D.: Online 6 dof augmented reality registration from natural features. In: Proc. of the ISMAR. (2002) 305–313
9. Simon, G., Fitzgibbon, A., Zisserman, A.: Markerless tracking using planar structures in the scene. In: Proc. of the ISAR. (2000) 120–128
10. Simon, G., Berger, M.: Reconstructing while registering: a novel approach for markerless augmented reality. In: Proc. of the ISMAR. (2002) 285–294
11. Simon, G., Berger, M.O.: Real time registration known or recovered multi-planar structures: application to ar. In: Proc. of the BMVC. (2002) 567–576
12. Shi, J., Tomasi, C.: Good features to track. IEEE Conf. on CVPR (1994) 593–600