Vision-based overlay of a virtual object into real scene for designing room interior

Harasaki, Shunsuke, Saito, Hideo

Shunsuke Harasaki, Hideo Saito, "Vision-based overlay of a virtual object into real scene for designing room interior," Proc. SPIE 4572, Intelligent Robots and Computer Vision XX: Algorithms, Techniques, and Active Vision, (5 October 2001); doi: 10.1117/12.444225

SPIE.

Event: Intelligent Systems and Advanced Manufacturing, 2001, Boston, MA, United States

Vision based overlay of a virtual object into real scene for designing room interior

Shunsuke Harasaki and Hideo Saito†

Department of Information and Computer Science, Keio University
3-14-1 Hiyoshi, Kohoku-ku, Yokohama 223-8522, Japan

ABSTRACT

In this paper, we introduce a geometric registration method for augmented reality (AR) and an application system, *interior simulator*, in which a virtual (CG) object can be overlaid into a real world space. *Interior simulator* is developed as an example of an AR application of the proposed method. Using *interior simulator*, users can visually simulate the location of virtual furniture and articles in the living room so that they can easily design the living room interior without placing real furniture and articles, by viewing from many different locations and orientations in real-time. In our system, two base images of a real world space are captured from two different views for defining a projective coordinate of object 3D space. Then each projective view of a virtual object in the base images are registered interactively. After such coordinate determination, an image sequence of a real world space is captured by hand-held camera with tracking non-metric measured feature points for overlaying a virtual object. Virtual objects can be overlaid onto the image sequence by taking each relationship between the images. With the proposed system, 3D position tracking device, such as magnetic trackers, are not required for the overlay of virtual objects. Experimental results demonstrate that 3D virtual furniture can be overlaid into an image sequence of the scene of a living room nearly at video rate (20 frames per second).

Keywords: vision based system, registration, 3D interaction techniques, augmented reality

1. INTRODUCTION

Many Augmented Reality (AR) technologies and applications have been studied in recent years.[1] One example of AR is overlaying virtual CG objects onto an image sequence captured by a video camera for films and TV programs. One of the most important issues for AR is geometric registration between the real and virtual worlds. This generates a correct view of a virtual object, and overlays it onto a view of the real world. Conventional methods for achieving geometric registration can be categorized into two groups: methods based on 3D sensing of the real world (3D measurement-based methods) and methods based on image appearance of the real world (appearance-based methods). The former category covers methods in which the location and orientation of the viewpoint and/or some points in the real world are explicitly measured with 3D positioning sensors to obtain the relationship between the real world, the virtual world, and the viewpoint.

One popular 3D positioning device is the magnetic tracker, which is used for measuring the location and orientation of the viewpoint in the system.[2] Methods that use such positioning devices can run stably and do not require any real-world information such as the positions of some markers in the real world. However, the locations and orientations that can be obtained by the existing devices are not accurate enough to achieve geometric registration with satisfactory accuracy. Furthermore, the measurement area of such devices is limited to some special environments, which prevents the use of AR systems in broader environments such as outdoors. Recent advances in computer vision research have also contributed to measure the location and orientation of the viewpoint by extracting feature points from images. Such methods can be achieved using cameras alone, but the 3D position of feature points for estimating the location and orientation of the camera (extrinsic parameters) have to be known beforehand. Additionally, intrinsic parameters also have to be known beforehand. Neumann et al. estimated the parameters of a monocular camera,[3] where metric of three feature points must be measured beforehand. Kanbara et al. estimated the location and orientation of the camera at arbitrary positions from three non-metric mesured feature points by reconstructing the 3D positions of

Shunsuke Harasaki: E-mail: harasaki@ozawa.ics.keio.ac.jp
Hideo Saito: E-mail: saito@ozawa.ics.keio.ac.jp
†PRESTO, Japan Science and Technology Corporation(JST)

Intelligent Robots and Computer Vision XX: Algorithms, Techniques, and Active Vision,
David P. Casasent, Ernest L. Hall, Editors, Proceedings of SPIE Vol. 4572 (2001)

these points using stereo cameras with a head-mounted display (HMD),[4] but the stereo cameras had to be well calibrated.

Methods that combine the advantages of a positioning sensor and computer-vision-based techniques have been suggested[5,6,7] Anabuki et al. constructed the AR application system "*Welbo*" with this combination method, where one can visually simulate the location of virtual furniture and articles in the living room with HMD.[8] The combination is stable and accurate enough for a practical system, but the applicable environment is still limited because of the limitations of the 3D positioning devices.

On the other hand, the image-appearance-based method does not provide the location and orientation of the viewpoint explicitly, but implies similar information based on two or more base images of the real world, where a virtual object is overlaid. The appearance-based methods can be achieved using cameras alone, without any positioning devices, and do not require the intrinsic and external parameters of the camera and metric measured feature points. Kutulakos et al. achieved geometric registration from two base images, assuming an affine camera.[9] Since an affine camera assumes orthographic or weak-perspective projection, the distance of the viewpoint from the object must be long relative to the focal length of the camera, and the depth range of the virtual object must be small compared to the focal length. Kobayashi et al. achieved augmentation by applying a linear algorithm based on three base images assuming affine representation.[10]

Sato shows the theoretical possibility of geometrical registration in perspective by projective reconstruction from two base images applying fundamental matrices calculated from seven or more corresponding points between the base images and the projective basis defined from five or more corresponding points of which four are not coplanar and three are not collinear in the real world.[11] Since this method assumes projective representation, it has none of the limitations of affine representation, about, for example, the distance or difference in depth. Seo et al. also achieved geometric registration by projective reconstruction.[12] In their method, intrinsic and external parameters are also estimated based on the projective reconstruction.

We propose a geometric registration method of image-appearance-based method taking into account the perspective projection by using uncalibrated cameras. The key idea of our method is related to Kutulakos' method,[9] but we extend that method to a perspective camera by applying the Projective Grid Space (PGS).[13] Although the theoretical background of our method is similar to their ones[11,12] in terms of the projective geometry based method, our method is more practical because the PGS explicitly defines the projective position in a real-world scene by using two base images. Since their methods[11,12] are based on the projective reconstruction, the position of epipoles must be explicitly estimated. However, epipole estimation is known to be unstable, so accurate registration is not easy in a practical implementation. Our method, on the other hand, does not need to estimate epipoles because of the PGS. This makes our method more accurate in practical applications.

We also constructed an AR application system, *interior simulator*, with our geometric registration method. The *interior simulator* enables the users to visually simulate the location of virtual furniture and articles in the living room and easily design the living room interior without placing real furniture and articles, viewing from many different locations and orientations in real-time.

In this paper, Section 2 presents some preliminaries of geometrical theories. Section 3 presents our geometric registration method. Section 4 describes the real-time overlay system, *interior simulator*, which is developed by applying the proposed method. Finally, concluding remarks are given in Section 5.

2. PRELIMINARIES

2.1. Relationships between the viewpoint, real world, and virtual world

In order to overlay a virtual object onto a real world image from the viewpoint, the relationships between the viewpoint, real world, and virtual world must be obtained. Basically, there are two main methods for obtaining the relationships: the 3D measurement-based method and the image-appearance-based method.

In the 3D measurement-based method, four coordinate systems are defined: 1) object orthogonal coordinate system (virtual world), where a virtual object is represented, 2) world orthogonal coordinate system for the real world, 3) camera coordinate system for the viewpoint, and 4) image plane coordinate system, where a camera image is projected, as shown in Fig. 1. For accurate projection of a virtual object onto an image plane, it is necessary

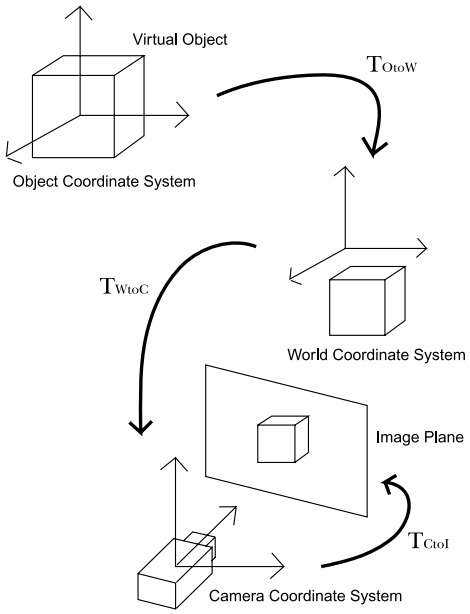


Figure 1. Coordinate system of 3D measurement-based method.

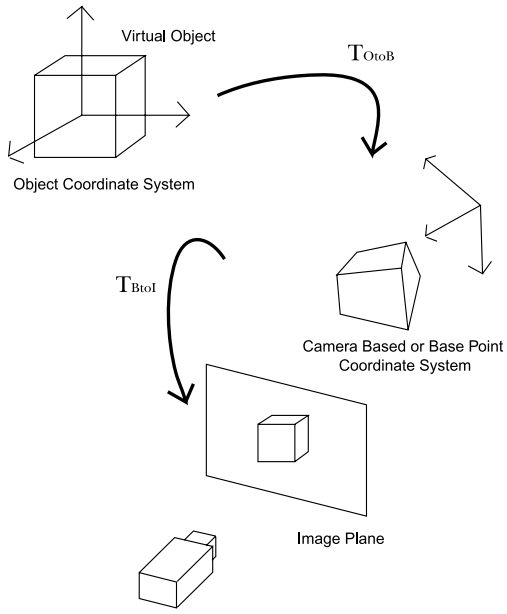


Figure 2. Coordinate system of image-appearance-based method.

to obtain the transformation matrices of object-to-world $\boldsymbol{T_{OtoW}}$ and world-to-camera $\boldsymbol{T_{WtoC}}$ and the projection matrix of the camera-to-image $\boldsymbol{T_{CtoI}}$ in the following equation.

$$\begin{bmatrix} u \\ v \\ h \end{bmatrix} = \boldsymbol{T_{CtoI}T_{WtoC}T_{OtoW}} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \tag{1}$$

where $[u, v, h]^\top$ is a 3D homogeneous coordinate in the image plane coordinate system and $[x, y, z, w]^\top$ is a 4D homogeneous coordinate in the object coordinate system. Thus, the 3D measurement-based method is a method of obtaining the three matrices.

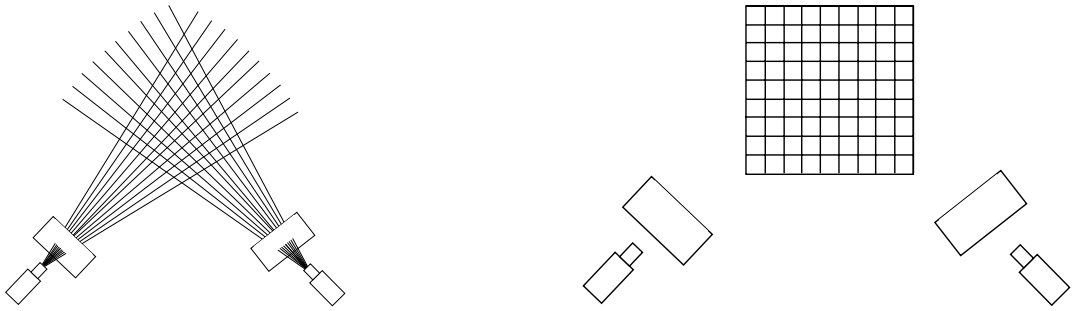
In the image-appearance-based method, on the other hand, three coordinate systems are defined: 1) object orthogonal coordinate system (virtual world), where a virtual object is represented, 2) image-appearance-based coordinate system (real world), which is constructed by two or more images captured by base cameras, and 3) image plane coordinate system, where a camera image is projected, as shown in Fig. 2. For correct registration, it is necessary to obtain the transformation matrix of object-to-base $\boldsymbol{T_{OtoB}}$ and the projection matrix of base-to-image $\boldsymbol{T_{BtoI}}$ in this equation

$$\begin{bmatrix} u \\ v \\ h \end{bmatrix} = \boldsymbol{T_{BtoI}T_{OtoB}} \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix}. \tag{2}$$

Thus, the image-appearance-based method is a method of obtaining the two matrices.

2.2. Projective Grid Space

In this section, we define the Projective Grid Space (PGS).[13] PGS is a 3D space that is constructed by two base perspective images captured by two base cameras as shown in Fig. 3. The three axes of PGS are expressed by $\boldsymbol{P}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$, which are the horizontal and vertical axes of base image 1 captured by base camera 1 and the horizontal axis of base image 2 captured by base camera 2, respectively. The voxels are not the same size as illustrated Fig. 3-(a) compared with the same size voxels of the Orthogonal Space as illustrated Fig. 3-(b).



(a) (b)

Figure 3. Coordinate systems of (a) Projective Grid Space and (b) Orthogonal Space.

Representing an arbitrary 3D space with PGS defined by two base cameras, the relationships between an arbitrary camera and the 3D space can be represented by only the geometric relationship between the camera at arbitrary position and the base cameras, which is an epipolar geometry. Therefore, no camera calibration is required to obtain the relationships between the camera and a 3D space of the orthogonal coordinate system of the real world. Thus, geometric registration for augmented reality can be achieved with uncalibrated cameras as long as the relationships among the geometric positions of the cameras are represented in PGS.

The epipolar geometry between two cameras is represented by the fundamental matrix $\boldsymbol{F}$,[14] which is a 3×3 homogeneous matrix of rank 2 with seven degrees of freedom calculated by the correspondence of seven points in the two images nonlinearly. In order to speed up the calculation of $\boldsymbol{F}$, our method uses linear algebraic with eight corresponding points.

3. PROPOSED METHOD

3.1. Outline

The following is the outline of the proposed AR method.

1. Capture two base images at different positions by uncalibrated base cameras.
2. Specify at least eight correspondeing points in the base images interactively for the linear calculation of the fundamental matrix between the base images.
3. Specify the location of the five or more points where a virtual object should be projected in the base images which construct PGS. For example, these points are the area and height of a virtual object in the base images.
4. Transform a virtual object from object space to PGS.
5. Project a virtual object onto the base images.

These initial operations $1 \sim 5$, described in Fig. 4, are performed before real-time overlay of a virtual object into an input image sequence captured by an uncalibrated AR camera at an arbitrary viewpoint.

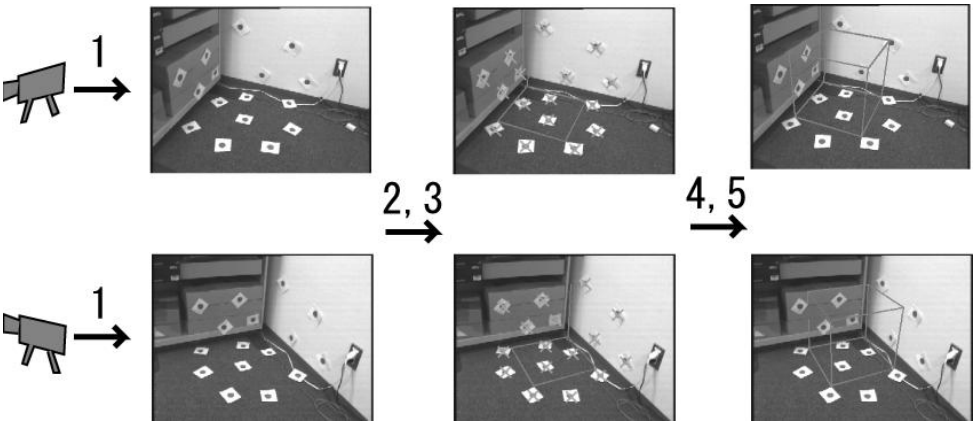


Figure 4. Outline of the initial process.

6. Capture the AR camera image (or image sequence) by uncalibrated AR camera at an arbitrary viewpoint.
7. Detect and tracking eight or more corresponding feature points in the AR camera image. These points are corresponding ones between the AR camera image and each base image for fundamental matrices.
8. Transfer a virtual object in the base images onto the AR camera image.
9. Render a virtual object.

These process of generating an overlaid AR image, described in Fig. 5, are applied to an image sequence and repeated in real-time.

Note that there is no need to know the intrinsic and external parameters of all cameras.

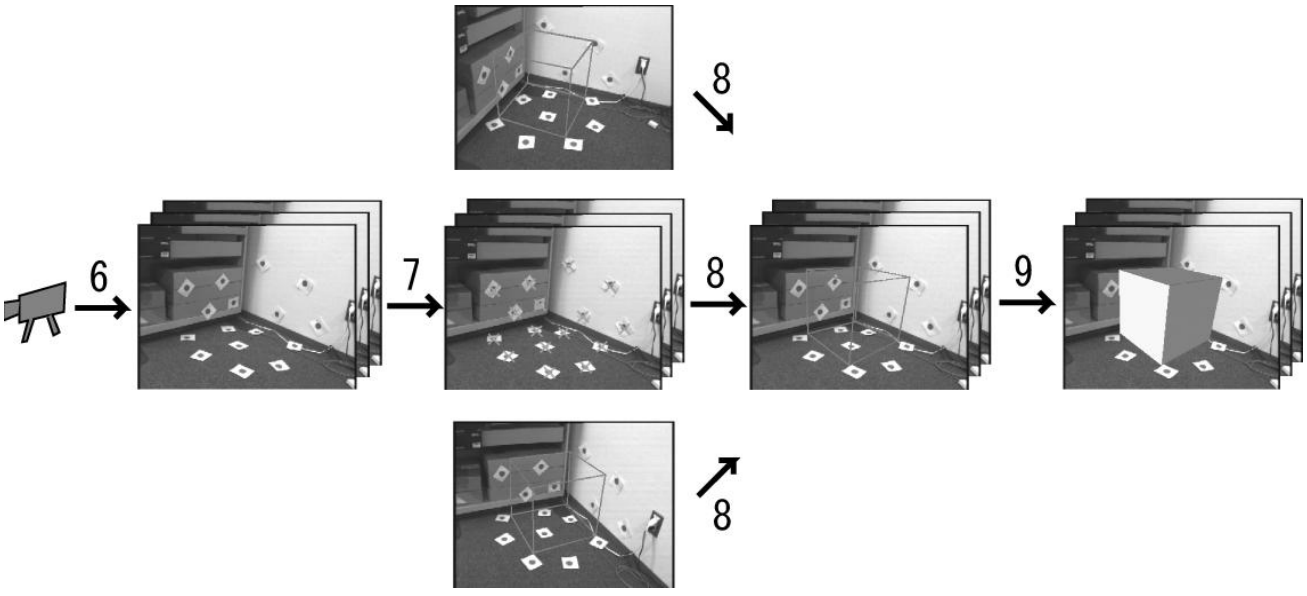


Figure 5. Outline of the real-time overlay process.

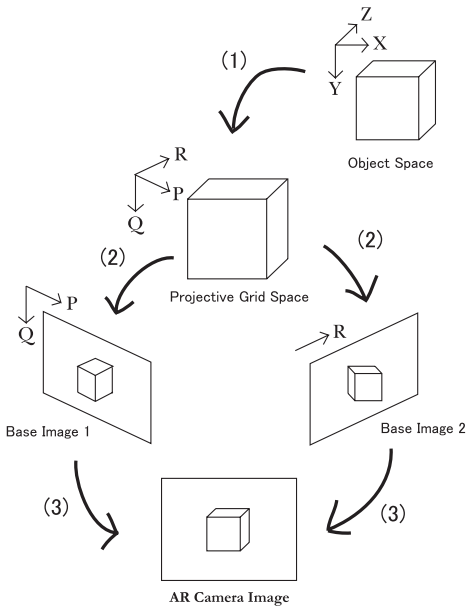


Figure 6. Overview of the process of the transformation and projection of a virtual object.

3.2. Specification and tracking of corresponding points

In order to calculate the fundamental matrices linearly, eight or more feature points have to be detected and related to each other between the base images, and each base image and the AR camera image. The procedure for specifying and tracking the corresponding points is described below.

1. Specify the corresponding markers interactively in base images and the first frame of an image sequence of AR camera.
2. Set the tracking windows to the size of 10×10 pixels around each specified position and searching within each tracking window. Then the regions of the markers can be extracted from the images. In the images of the other frames of an image sequence, on the other hand, assuming that the positions of the markers hardly move compared with the pre-frame image when the capturing speed of the AR camera is high enough, set the tracking windows to the size of 10×10 pixels around each position of the corresponding points in the pre-frame image and search within each tracking window; then the regions of each markers can be extracted.
3. Each centroid of the regions is calculated as each position of the corresponding points.

3.3. Registration

In this section, we describe the algorithm for projecting a virtual object defined in a object space onto AR camera images in an appropriate position.

As described in Section 2.1, for correct registration by the appearance-based method, the relationships between object space, camera based or base point space, and image plane are required. The registration procedure is overviewed below (shown in Fig. 6).

1. Transformation of a virtual object from object space into PGS.
2. Projection of a virtual object from PGS onto base images.
3. Transfer of a virtual object from base images onto an AR camera image.

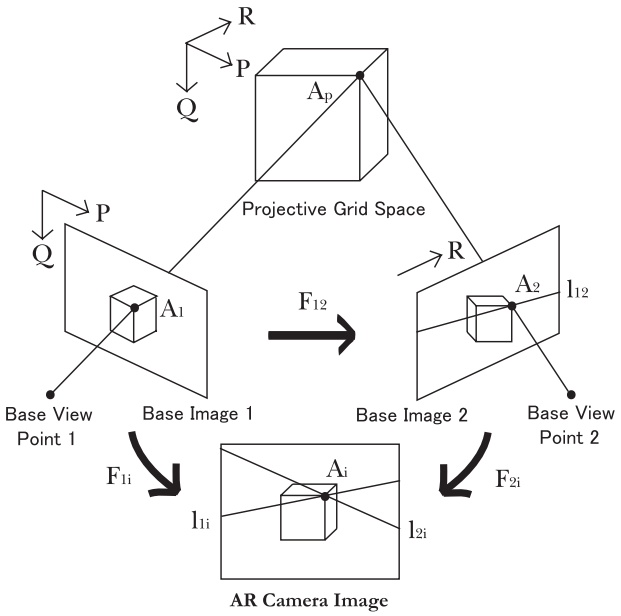


Figure 7. Projection and transfer of a virtual object onto each image.

3.3.1. Transformation from object space into PGS

In this section, we describe the transformation of a virtual object from object space, which is defined as the orthogonal 3D coordinate system where the origin is the center of a virtual object, into PGS, where the origin is the center of base images. The relationship between object space and PGS is expressed by

$$\boldsymbol{A_{P_i}} \sim \boldsymbol{T_{OtoP}} \boldsymbol{A_{O_i}} \tag{3}$$

where $\boldsymbol{A_{O_i}} \sim [x_i, y_i, z_i, 1]^\top$ is a 4D homogeneous point in object space, $\boldsymbol{A_{P_i}} \sim [p_i, q_i, r_i, 1]^\top$ is a point in PGS, and

$$\boldsymbol{T_{OtoP}} = \begin{bmatrix} t_{11} & t_{12} & t_{13} & t_{14} \\ t_{21} & t_{22} & t_{23} & t_{24} \\ t_{31} & t_{32} & t_{33} & t_{34} \\ t_{41} & t_{42} & t_{43} & t_{44} \end{bmatrix} \tag{4}$$

is the transformation matrix from object space into PGS.

Since $\boldsymbol{T_{OtoP}}$ has 15 degrees of freedom because of $t_{44} = 1$, $\boldsymbol{T_{OtoP}}$ can be calculated from at least five corresponding points between object space and PGS.

3.3.2. Projection from PGS onto base images

In this section, we describe the projection from PGS onto base images, using the property of PGS and the fundamental matrix as illustrated in Fig. 7, where $\boldsymbol{A_P} \sim [p, q, r, 1]^\top$ is a 4D homogeneous point in PGS and $\boldsymbol{A_1} \sim [u_1, v_1, 1]^\top$ and $\boldsymbol{A_2} \sim [u_2, v_2, 1]^\top$ are 3D homogeneous points projected $\boldsymbol{A_p}$ onto base images 1 and 2, respectively. By the property of PGS, $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ are described by

$$\boldsymbol{A_1} \sim [p, q, 1]^\top, \boldsymbol{A_2} \sim [r, v_2, 1]^\top. \tag{5}$$

Since the epipolar line l_{12} corresponding to $\boldsymbol{A_1}$ is described in base image 2 by the fundamental matrix $\boldsymbol{F_{12}}$ between base images, v_2 can be calculated.

3.3.3. Transfer from base images onto AR image

In this section, we describe the transfer from base images onto an AR camera image, using the fundamental matrices as illustrated in Fig. 7, where $\boldsymbol{A_i}$ is 3D homogeneous point projected $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ onto the AR camera image. Since the epipolar lines l_{1i} and l_{2i} corresponding to $\boldsymbol{A_1}$ and $\boldsymbol{A_2}$ are described by the fundamental matrices $\boldsymbol{F_{1i}}$ and $\boldsymbol{F_{2i}}$ between each base image and the AR camera image, $\boldsymbol{A_i}$ can be calculated as the cross point of the epipolar lines.

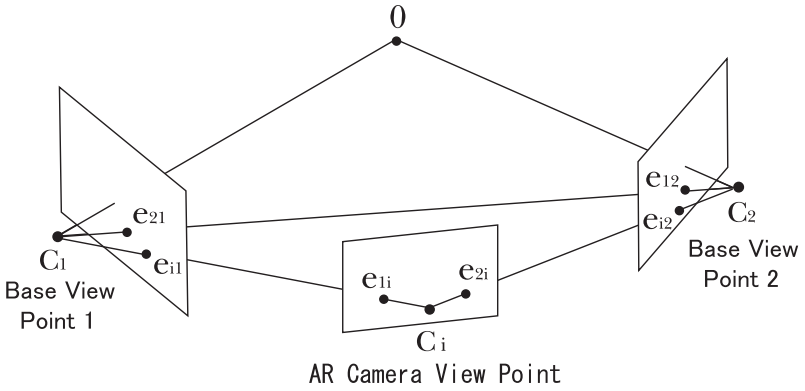


Figure 8. Each position of cameras in Projective Grid Space.

3.4. Rendering

Since the hidden surfaces of a virtual object have to be removed for correct rendering, our method uses the Z buffer algorithm. In this algorithm, the distance between a viewpoint and a virtual object must be required as the Z value. By referring to the Z value, we can detect the nearest surface of a virtual object from the viewpoint of the AR camera. Then, by painting the pixels with the color of the nearest surface, a correct view of a virtual object can be generated. Thus, the relative position between a viewpoint and a virtual object must be obtained in order to apply the Z buffer algorithm.

The relative position can be obtained by applying each property of PGS and the fundamental matrices in our method. Now since the position of a virtual object can be obtained in PGS as described in Section 3.3.1, all we have to do is obtain that of the viewpoint in PGS. As illustrated in Fig. 8, presenting each position of the viewpoint of base images as $\boldsymbol{C_1} \sim [C_{1_p}, C_{1_q}, C_{1_r}, 1]^\top$ and $\boldsymbol{C_2} \sim [C_{2_p}, C_{2_q}, C_{2_r}, 1]^\top$, respectively, in PGS, the epipole of base image 1 on base image 2 as $\boldsymbol{e_{12}} \sim [e_{12_u}, e_{12_v}, 1]^\top$, and the epipole of base image 2 on base image 1 as $\boldsymbol{e_{21}} \sim [e_{21_u}, e_{21_v}, 1]^\top$, we can represent them as $\boldsymbol{C_1} \sim [C_{1_p}, C_{1_q}, e_{12_u}, 1]^\top$, and $\boldsymbol{C_2} \sim [e_{21_u}, e_{21_v}, C_{2_r}, 1]^\top$, where C_{1_p}, C_{1_q} and C_{2_r} are arbitrary real numbers and e_{12_u}, e_{21_u} and e_{21_v} are calculated from the fundamental matrix $\boldsymbol{F_{12}}$. Thus the position of the base viewpoint $\boldsymbol{C_1}$ and $\boldsymbol{C_2}$ can be obtained in PGS. On the other hand, the position of the AR camera is represented as $\boldsymbol{C_i} \sim [e_{i1_u}, e_{i1_v}, e_{i2_u}, 1]^\top$ in PGS calculated from $\boldsymbol{F_{1i}}$ and $\boldsymbol{F_{2i}}$. After all, since each position of a virtual object, the base viewpoints, and the viewpoint of the AR camera can be obtained in PGS, a virtual object is correctly rendered on all images by the Z buffer algorithm.

4. EXPERIMENTS AND DISCUSSION

We implemented the augmented reality system, *interior simulator*, based on our method using only a PC (OS: Windows NT, CPU: Intel Pentium III 750 MHz, RAM: 512 MB) and a CCD camera (JAI Corporation: CV-M70). The images used in all the experiments were 320×240 pixels, and graphical views of a virtual object were rendered with OpenGL library. The markers used in this system were red and round, and we did not measure the size and position of the markers in the real world.

In *interior simulator*, virtual objects such as furniture or articles are overlaid into an image sequence of the real world. The users can visually simulate the location of virtual furniture and articles in the living room and easily design the living room interior without placing real furniture and articles, viewing from many different locations and orientations in real-time.

The process of *interior simulator* is presented below. First, base image 1 and 2 are captured by an uncalibrated camera from different positions with markers placed in the real world as shown in Fig. 9. Then a virtual object is registered in the base images with specifying the eight or more corresponding points for the fundamental matrix and the locations of the five points where a virtual object should be placed in the real world. Fig. 10 shows the base images where a frame box was overlaid as a virtual object. Next, the AR camera image sequence is captured by a hand-held uncalibrated camera with only the markers placed, generating overlaid AR camera images with tracking the markers in real-time.

(a) Base image 1 (b) Base image 2

Figure 9. Base images.

(a) Base image 1 overlaid a frame of box (b) Base image 2 overlaid a frame of box

Figure 10. Geometrical registrated base images.

We registered a virtual chest as a virtual object into a real world scene with the developed *interior simulator*. Fig. 11 shows some images of (a) the input AR camera image sequence, (b) with a virtual frame box registered and (c) with a virtual chest rendered. In addition, the system runs nearly at video rate (20 frames per seconds on the average).

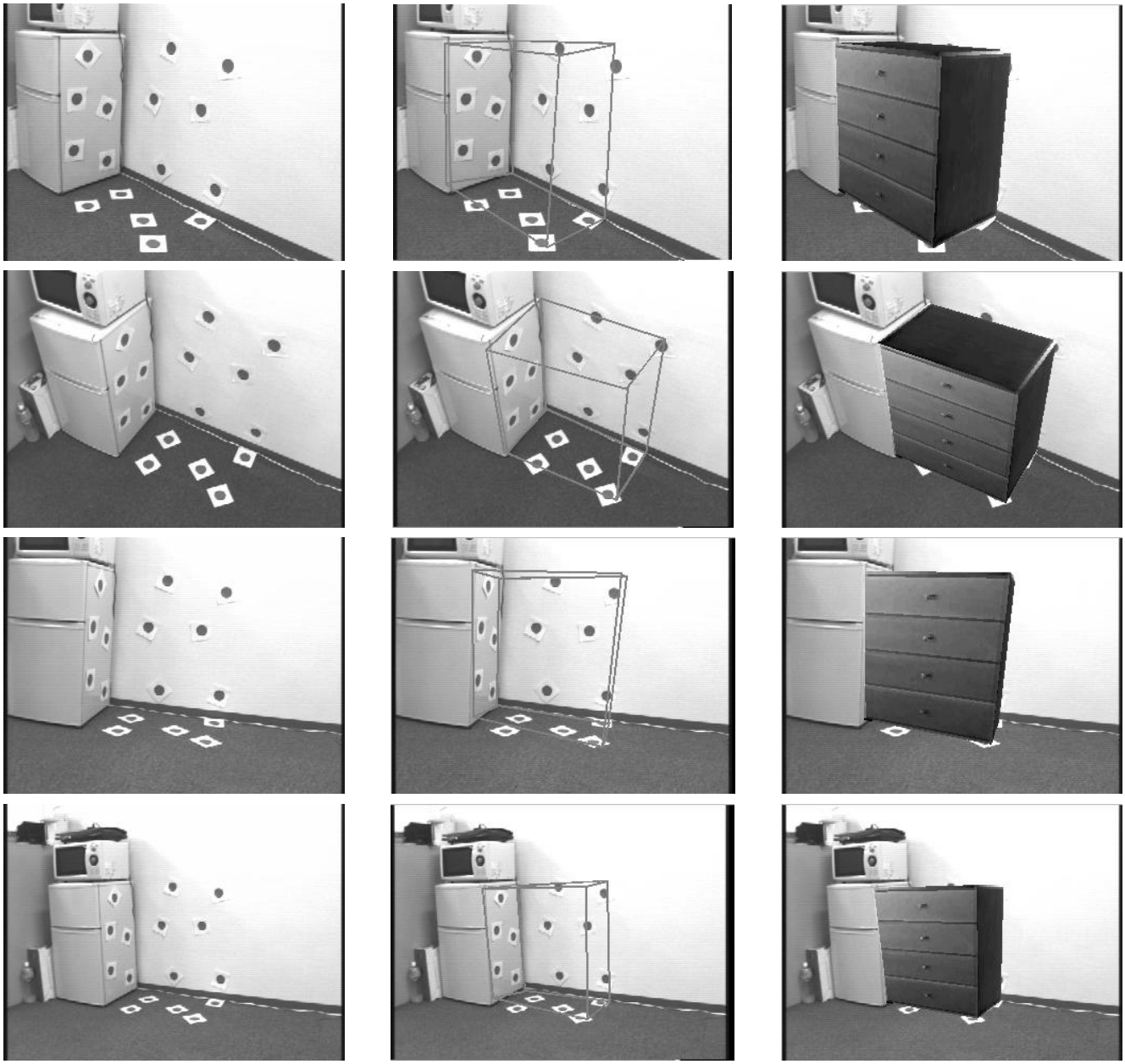
The number of the corresponding points of markers for calculating the fundamental matrices is 16 and that for calculating $\boldsymbol{T_{OtoP}}$, which are the vertices of a box, is 5.

In the augmented reality system of our method, a rendered virtual object moves a few pixels shakily every a few frames, because the corresponding points can not be tracked accurately. The centroids of the red round markers are used as the corresponding points in our system. Since the accurate regions of the markers may not be extracted because of the flickering of the fluorescent light, the extraction of the centroids of the markers may be unstable. Thus the calculation of the fundamental matrices may be unstable, so shakiness may be generated. The accurate tracking of feature points in real time is generally a hard problem. Although the positioning error of the markers is not so large and acceptable for most cases, we will figure out a fast and accurate tracking algorithm for improving the overlaying accuracy in feature. Furthermore, we will also implement an algorithm to track arbitrary feature point in input images rather than the red round markeres, so that we can use the *interior simulator* in arbitrary environment.

5. CONCLUSION

We proposed a geometric registration algorithm based on image-appearance-based method for augmented reality. By applying the fundamental matrix and Projective Grid Space, we were able to generate the correct perspective view of a virtual object and overlay it onto a view of the real world in an appropriate position using uncalibrated cameras.

We constructed an augmented reality system, *interior simulator*, that runs nearly in real time (20 frames per second) by applying our method with an image sequence and tracking the non-measured markers as the corresponding points between the images.



(a) the input AR camera image (b) with a virtual frame box registered (c) with a virtual chest rendered

Figure 11. Some geometric registration and rendering results of a virtual chest.

Since the tracked feature points are not measured in our method, we expect to construct a system with no markers in the future. To make such a system, we must figure out methods where natural feature points are tracked accurately and the corresponding points are specified automatically.

REFERENCES

1. R. T. Azuma, "A survey of augmented reality," *Presence* **6**, pp. 355–358, 1997.
2. M. Bajura, H. Fuchs, and R. Ohbuchi, "Merging virtual objects with the real world: Seeing ultrasound," *Commun of the ACM* **36**, pp. 52–62, 1993.
3. U. Neumann and Y. Cho, "A self-tracking augmented reality system," *Proc. VRST '96* , pp. 109–115, 1996.
4. M. Kanbara, T. Okuma, H. Takemura, and N. Yokoya, "A stereoscopic video see-through augmented reality system based on real-time vision-based registration," *Proc. IEEE Virtual Reality 2000 Int. Conf. (VR2000)* , pp. 255–262, March 2000.
5. M. Bajura and U. Neumann, "Dynamic registration correction in video-based augmented reality system," *IEEE Computer Graphics and Applications* **15**, pp. 52–60, 1995.
6. A. State, G.Hirota, D.Chen, W.Garrett, and M.Livingston, "Superior augmented reality registration by integrating landmark tracking and magnetic tracking," *Proc. SIGGRAPH'96* , pp. 429–438, 1996.
7. K. Sato, H. Yamamoto, and H. Tamura, "Registration of physical and virtual spaces in mixed reality systems," *Proc. VRSJ Annual Conf.* **2**, pp. 161–164, 1997.
8. M. Anabuki, H. Kakuta, H.Yamamoto, and H. Tamura, "Welbo:an embodied conversational agent living in mixed reality space," *CHI2000 Extended Abstracts* , pp. 10–11, 2000.
9. K. N. Kutulakos and J. Vallino, "Affine objet representations for calibration-free augmented reality," *Proc. IEEE Virtual Reality Ann. Int. Symp. (VRAIS'96)* , 1996.
10. T. Kobayashi, G. Inoue, L. Quan, and Y. Ohta, "A unified linear algorithm for a novel view synthesis and camera pose estimation in mixed reality," *Proc. IEEE Virtual Reality 2000 Int. Conf. (VR2000)* , March 2000.
11. J. Sato, *Computer Vision - Geometry of Vision - (in Japanese)*, Corona Publishing, 1999.
12. Y. Seo and K. Hong, "Calibration-free augmented reality in perspective," *IEEE Trans. Visualization and Computer Graphics* **6**, pp. 346–359, 2000.
13. H. Saito and T. Kanade, "Shape reconstruction in projective grid space from large number of images," *Proc. IEEE Computer Vision and Pattern Recognition* **2**, pp. 49–54, 1999.
14. Z. Zhang, "Determining the epipolar geometry and its uncertainly: A review," *Research Report, INRIA Sophia-Antipolis, No.2927* , July 1996.