Arbitrary Viewpoint Video Synthesis From Multiple Uncalibrated Cameras

Satoshi Yaguchi and Hideo Saito

***Abstract*—We propose a method for arbitrary view synthesis from uncalibrated multiple camera system, targeting large spaces such as soccer stadiums. In Projective Grid Space (PGS), which is a three-dimensional space defined by epipolar geometry between two basis cameras in the camera system, we reconstruct three-dimensional shape models from silhouette images. Using the three-dimensional shape models reconstructed in the PGS, we obtain a dense map of the point correspondence between reference images. The obtained correspondence can synthesize the image of arbitrary view between the reference images. We also propose a method for merging the synthesized images with the virtual background scene in the PGS. We apply the proposed methods to image sequences taken by a multiple camera system, which installed in a large concert hall. The synthesized image sequences of virtual camera have enough quality to demonstrate effectiveness of the proposed method.**

***Index Terms*—Fundamental matrix, projective geometry, projective grid space, shape from multiple cameras, view interpolation, virtual view synthesis.**

I. Introduction

THE synthesis of new views from images can enhance the visual-entertainment effect of a movie or broadcast for a television viewer. One way of enhancing the visual effect is through virtual movement of viewpoint, which makes viewers virtually feel that they are in the target scene. Recent applications of this effect can be found in the futuristic movie "The Matrix," and the SuperBowl XXXV broadcast by CBS in 2001 which used the EyeVision system. Virtualized Reality [8], a pioneering project in this field, has achieved virtual viewpoint movement for dynamic scenes by using computer vision technology. Whereas "The Matrix" and EyeVision use the switching effect of real images taken by multiple cameras, computer-vision-based technology can synthesize arbitrary viewpoint images to create a virtual viewpoint movement effect.

We aim to apply virtualized reality technology to actual sporting events. New-view images are generated by rendering pixel values of input images in accordance with the geometry of the new view and a three-dimensional (3-D) structure model of the scene, which is reconstructed from multiple-view images. The 3-D shape reconstruction from multiple views requires camera calibration, which is carried out in order to relate the camera geometry to the object space geometry. For camera calibration, the 3-D positions of several points in Euclidean space and 2-D positions of those points on each view image must be measured precisely. For this reason, when there are many cameras involved in the production of an event, a lot of effort must be expended to perform the calibration. This is especially true in the case of a large space, such as a stadium, where it is very difficult to set many calibration points whose positions have to be precisely measured for the entire area. We have already proposed a new framework for shape reconstruction from multiple uncalibrated cameras in a projective grid space (PGS) [15], in which coordinates between cameras are defined by using epipolar geometry instead of calibration.

In this paper, we present a method for generating arbitrary views from image sequences taken from multiple uncalibreated cameras. The shape-from-silhouette (SS) [2], [14] method is applied to reconstruct the shape model in the PGS. Then, the dense corresponding relation between the images derived from the shape model is used to synthesize intermediate appearance view images. We demonstrate the proposed framework by showing several virtual image sequences generated from corrected multiple-camera image sequences captured in a large space $(110 - \mathrm{m\ (L)} \times 50 - \mathrm{m\ (W)} \times 25 - \mathrm{m\ (H)})$.

II. Related Works

View synthesis from stereo images has long been a topic of study [17]. Once the disparity between a pair of stereo images is obtained, it can be modified to obtain intermediate images. However, a hole, where no pixel value can be assigned from the original stereo pair, generally appears in synthesized view images because of occluded regions.

One method devised for removing such a hole caused by an occlusion is to use a completely closed 3-D shape model of the object, which can be obtained by using shape scanning technology [4], [24] or recovered from multiple-view images [6], [19], [23]. Such a framework for generating new views from the recovered 3-D model of an object and its texture map on the 3-D model surface is generally called model-based rendering (MBR). MBR can handle the occlusion problem, but registration errors in the texture map on the constructed 3-D model may cause blurring of the synthesized virtual images.

Alternatively, image-based rendering (IBR) [1], [3], [5], [7], [10], [11], [18] has recently been developed for generating new-view images from multiple-view images without using a 3-D shape model of the object. Because IBR is essentially based on 2-D image processing (cut, warp, paste, etc.), the errors in 3-D shape reconstruction do not affect the quality of the generated images as much as they do for the model-based rendering method. This implies that the quality of the input

Manuscript received February 19, 2002; revised July 2, 2002. This paper was recommended by Associate Editor I. Gu.

The authors are with the Department of Information and Computer Science, Keio University, Yokohama 223-8522, Japan (e-mail: yagu@ozawa.ics.keio.ac.jp; saito@ozawa.ics.keio.ac.jp).

Digital Object Identifier 10.1109/TSMCB.2003.817108

images can be well preserved in the new view images, however, we will have to ignore the occlusion effects.

Appearance-based virtual-view synthesis [16] takes into account the advantages of MBR and IBR. This 3-D shape model, which is recovered from multiple images, provides the required information for the IBR process, such as correspondence map, occluded area, etc., for the input images. The precise and dense correspondences make it possible to generate virtual views at arbitrary viewpoints without losing pixels even in partially occluded regions. Image-based Visual Hull (IBVH) [12] is another virtual view synthesis method that has the advantages of MBR and IBR. In IBVH, the hull shape of the object is represented by the intersection of silhouettes on the epipolar lines of one base camera. Such image-based representation contributes to high-speed rendering with conventional image rendering hardware. IBVH is difficult to manipulate reconstructed objects and virtual objects in the virtual space however, because the explicit 3-D shape model is not represented. The concept of a visual hull was originally proposed by Laurentini [9]. Although the visual hull reconstructed from silhouette images cannot represent an actual 3-D shape, the visual hull can be used as an approximation of the actual 3-D shape in some cases, such as IBVH [12] and the method presented in this paper.

The method presented in this paper extends the appearance-based virtual-view synthesis to the projective reconstruction framework in PGS. By applying the PGS to the similar virtual view synthesis technique, the strong camera calibration required in conventional work can be avoided.

III. Projective Grid Space

Reconstructing a 3-D shape model from multiple-view images requires a relationship between the 3-D coordinate of the object scene and the 2-D coordinate of the camera-image plane. Projection matrices that represent this relationship are estimated from measurements of 3-D/2-D correspondences obtained at a set of sample points. Since the 3-D coordinates are defined independently from the camera geometries, the 3-D positions of the sample points must be measured independently from each camera geometry. This procedure is called camera calibration [22]. Calibrating all of the each camera in a multiple-camera system requires a lot of work [8], [23]. Reconstructing a 3-D shape model from multiple-view images requires a relationship between the 3-D coordinate of the object scene and the 2-D coordinate of the camera image plane.

In our method, a 3-D point is related to a 2-D image point without estimating the projection matrices in a PGS [15], which is determined by using only the fundamental matrices [25] representing the epipolar geometry between two basis cameras. Because the 3-D coordinate in a PGS is dependently defined from the camera-image coordinates, the 3-D position of the sample points does not have to be measured. Therefore, the PGS enables 3-D reconstruction from multiple images without the need to estimate the projection matrices of each camera.

Fig. 1 shows the PGS scheme. The PGS is defined by the camera coordinates of the two basis cameras. Each pixel point (p, q) in the first basis camera image defines one grid line in the space. On the grid line, grid-node points are defined by the horizontal position r in the second image. Thus, the coordinates P and Q of PGS are decided by the horizontal coordinate and the vertical coordinate of the first basis image, and the coordinate R of the PGS is decided by the horizontal coordinate. Since the fundamental matrix $\boldsymbol{F}_{12}$ limits the position in the second basis view on the epipolar line $\boldsymbol{l}$, r is sufficient for defining the grid point. In this way, the projective grid space can be defined by two basis view images, whose node points are represented by (p, q, r).

We should note here the potential problem in this PGS framework. If the epipolar lines are nearly parallel, the epipolar lines transferring scheme fails to determine accurate intersection points. In such a case, we cannot recover a correct 3-D shape model and synthesize intermediate view images based on this PGS framework. This situation can be avoided by distributing the camera system so that the epipolar lines are not parallel between cameras.

IV. Model Reconstruction

Under the PGS framework, we reconstruct a 3-D shape model of the dynamic object by using the SS method. (We assume that the silhouette has been previously extracted by background subtraction.)

In the conventional SS method, each voxel in a certain Euclidean space is projected onto every silhouette image with projection matrices (which are calculated by accurately calibrating every camera [2], [14]) to check whether it is included in the object region. In applying the SS method in the PGS, every point in the PGS must be projected onto each silhouette image. As described in the previous section, the PGS is defined by two basis views, and a point in the PGS is represented as $A(p, q, r)$. Point $A(p, q, r)$ is projected onto $a_1(p, q)$ and $a_2(s, r)$ in the first basis image and the second basis image, respectively. Point a_1 is projected as the epipolar line l on the second basis view. Point a_2 on the projected line (Fig. 1), is expressed as

$$\boldsymbol{l} = \boldsymbol{F}_{12} \begin{bmatrix} p \\ q \\ 1 \end{bmatrix} \tag{1}$$

where $\boldsymbol{F}_{21}$ represents the fundamental matrix between the first and second basis images.

The projected point in an ith arbitrary real image is determined from two fundamental matrices, $\boldsymbol{F}_{1i}$, $\boldsymbol{F}_{2i}$ between two basis images and the ith image. Since $A(p, q, r)$ is projected onto $a_1(p, q)$ in the first basis image, the projected point in the ith image must be on the epipolar line $\boldsymbol{l}_1$ of $a_1(p, q)$, which is derived by $\boldsymbol{F}_{1i}$ as

$$\boldsymbol{l}_1 = \boldsymbol{F}_{1i} \begin{bmatrix} p \\ q \\ 1 \end{bmatrix}. \tag{2}$$

In the same way, the projected point in the ith image must be on epipolar line $\boldsymbol{l}_2$ of $a_2(r, s)$ in the basis image, which is derived by the $\boldsymbol{F}_{2i}$ as

$$\boldsymbol{l}_2 = \boldsymbol{F}_{2i} \begin{bmatrix} r \\ s \\ 1 \end{bmatrix}. \tag{3}$$

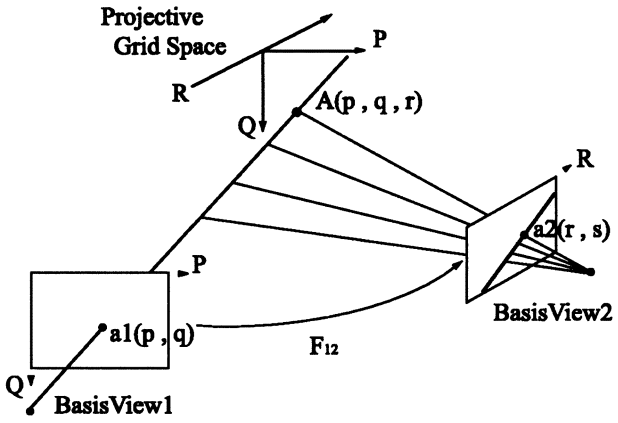

Fig. 1. Definition of projective grid. Point $A(p, q, r)$ on the projective grid space is projected to $a_1(p, q)$ and $a_2(r, s)$ on the first and second basis images.

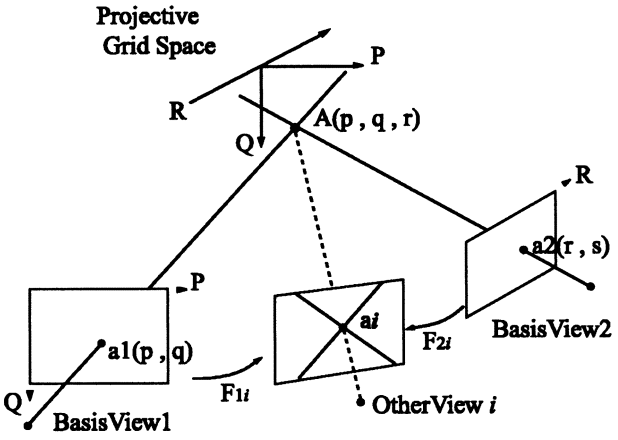

Fig. 2. Projection of point in space onto an image. Point $A(p, q, r)$ on the projective grid space is projected to the intersection of two epipolar lines in the image of other view i.

The point where epipolar lines $\boldsymbol{l}_1$ and $\boldsymbol{l}_2$ intersect is the projected point of $A(p, q, r)$ onto the ith image (Fig. 2). In this way, every projective grid point is projected onto every image, where the relationship can be represented by only the fundamental matrices between the image and two basis images.

The process for reconstructing 3-D shape model is outlined as follows.

First, two cameras are selected as basis cameras, and then the coordinate of the PGS is determined. Every voxel in a certain region is projected onto each silhouette image with the proposed scheme, as shown in Figs. 1 and 2. The voxel that is projected onto the object silhouette for all images is considered an existent voxel, while others are considered nonexistent. Thus the volume of the object can be determined in the voxel represented in the PGS. In this process, the order in which the existence of a voxel is checked is important for reducing the computational complexity, because the cost of computing the projection of a voxel onto an image is not the same for all the images in the proposed scheme. Since the vertical and horizontal coordinate of the first basis-view image are equivalent to P and Q coordinates in the PGS, projecting each voxel onto the first basis view image requires no calculation involving a fundamental matrix. In the second basis view image, the projected point is decided by calculating only one multiplication of a fundamental matrix to determine the epipolar line. This implies that the calculation for projection onto the second basis view becomes half compared with projecting the other images. Therefore, the order in which the existence of a voxel is checked should be Basis view 1, Basis view 2, and so on.

After the voxel existence determination, the implicit surface of the voxel representation of the object is extracted by using the Marching Cubes algorithm. Finally, the object model is reconstructed as a surface representation in the PGS.

V. Virtual View Synthesis

An arbitrary view image from a 3-D shape model can be generated by texture mapping onto the 3-D shape model [8], [23] or by morphing from the point correspondence of some reference images calculated using the model [1], [3], [16], [18]. In the former, the texture of the images are projected onto the 3-D shape model, and then re-projected onto the image. In this procedure, however, the generated images are likely to suffer from rendering artifacts caused by the inaccuracy of the 3-D shape. Therefore, we apply the latter procedure to generate arbitrary view images.

A. Arbitrary View Synthesis

Arbitrary view images are synthesized as intermediate images of two or three real neighboring reference images. If two reference images are selected, a virtual viewpoint can be taken on the line between the two real reference viewpoints. If three are selected, the virtual viewpoint can be taken from the inside of the triangle formed by the three real viewpoints. Therefore, if a number of cameras are mounted on the surface of a hemisphere enclosing the target space and any three of them form a triangle effectively, the virtual viewpoint can be moved freely all around the half sphere.

For the synthesis of arbitrary view images, intermediate images are synthesized by interpolating two or three reference images. The interpolation is based on the related concepts of view interpolation [3]. First, an image depicting the depth (a depth image) of the 3-D model is rendered on each reference image. To render the depth image, the 3-D positions of all the vertices on the surface representation of the 3-D shape model in the PGS are projected onto each reference viewpoint by applying the smallest depth value to the points projected onto the depth image. The depth d of the surface point from the reference viewpoint can be calculated by the following equation:

$$d = \sqrt{(p - p_c)^2 + (q - q_c)^2 + (r - r_c)^2} \tag{4}$$

where (p, q, r) and (p_c, q_c, r_c) represent the 3-D position on the surface in PGS and the viewpoint of the reference image, respectively. The 3-D position of the viewpoint can be determined by using the epipolar geometry of the cameras in the following procedure.

As shown in Fig. 3, the viewpoints of the two basis cameras and the other cameras are indicated by C_1, C_2, and C_i, respectively. Since the first basis camera viewpoint C_1 can be projected onto everywhere of the first basis images (Image 1), the P and Q components of C_1 can not uniquely be determined. Thus, we take the center point of the first basis camera, such

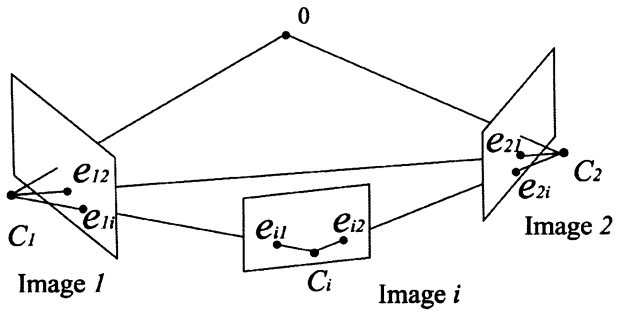

Fig. 3. Position of the viewpoint of each camera in PGS.

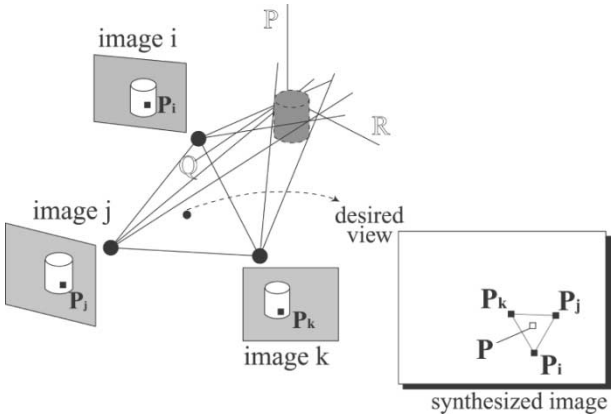

Fig. 4. Synthesis of desired view from three neighboring view images.

as $(X1_c, Y1_c)$. The R component of C_1 is the X component of the projected point of C_1 onto the second basis image, thus the epipole of the first basis camera in the second basis camera e_{21} determines the R component of C_1. Therefore, the 3-D position of the viewpoint of the first basis camera is defined as $C_1(X1_c, Y1_c, e_{21_x})$.

In the same way, if e_{12} is the epipole of the second basis camera in the first basis camera, then the P and Q components of the 3-D position of the second basis camera viewpoint C_2 are represented as e_{12_x}, e_{12_y}, respectively. We also define the R component of C_2 by the center position of the second basis camera. Then the 3-D position of the viewpoint of the second basis camera is defined as $C_2(e_{12_x}, e_{12_y}, X2_c)$. On the other hand, the viewpoint of the other cameras C_i can be represented as $C_i(e_{1i_x}, e_{1i_y}, e_{2i_x})$, by using the epipoles of the two basis cameras in camera i.

After rendering these depth images of the reference viewpoints, an intermediate viewpoint image is synthesized as follows. Let w_1, w_2, w_3 represent the weighted values of the interpolation of the reference view images. First, each vertex on the 3-D surface model is projected onto all the reference viewpoints, and the projected points are indicated as $\boldsymbol{P_i}$, $\boldsymbol{P_j}$ and $\boldsymbol{P_k}$, which are shown in Fig. 4. Then, the pixel position on the interpolated view image $\boldsymbol{P}$ for the vertex is calculated by the following equation:

$$\boldsymbol{P} = \frac{w_1\boldsymbol{P}_i + w_2\boldsymbol{P}_j + w_3\boldsymbol{P}_k}{w_1 + w_2 + w_3}. \tag{5}$$

Next, visibility of the vertex from each reference viewpoints is checked by comparing the depth from the reference viewpoint

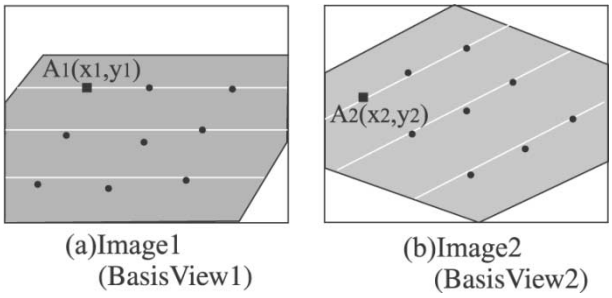

Fig. 5. Extracting the correspondence points between the two basis view images.

to the vertex with the depth value in the depth image at the reference viewpoint. If they are not equal, the vertex can be regarded as invisible from the reference viewpoint. Let v_1, v_2, v_3 represent the visibility (1: visible, 0: invisible) of the reference viewpoint. If the vertex is visible from at least one reference viewpoint, the color value is determined as the following equation:

$$I(\boldsymbol{P}) = \frac{v_1w_1I_i(\boldsymbol{P}_i) + v_2w_2I_j(\boldsymbol{P}_j) + v_3w_3I_k(\boldsymbol{P}_k)}{v_1w_1 + v_2w_2 + v_3w_3} \tag{6}$$

where $I_i(\boldsymbol{P_i})$, $I_j(\boldsymbol{P_j})$ and $I_k(\boldsymbol{P_k})$ are the colors of the projected points, and $I(\boldsymbol{P})$ is the interpolated color.

By changing the weighting ratio, the virtual viewpoint can be moved inside of the triangle.

The interpolation strategy presented here can synthesize geometrically correct intermediate images only if the viewing directions of the reference cameras are parallel to each other. Actually, they cannot be parallel because we assume all the cameras are directed at the common objective space. Thus, the interpolation strategy implies the geometrical approximation such that the reference cameras are parallel. In this paper, we assume that the distortion caused by the approximation is not obvious in the camera placement shown in Fig. 9. If we need to synthesize a geometrically correct interpolation between the reference viewpoints, we need to take into account the homographic transformation of the image plane that occurs among the reference cameras and the interpolated viewpoint, as Seitz *et al.* proposed in [18].

B. Synthesizing the Floor Plane

We also propose here a method to synthesize a floor plane in this projective method. Since the floor plane is removed at the step where the silhouette image is made, the SS method only provides the 3-D shape of an object without its background. We generate more realistic images, by synthesizing a floor plane image.

Since the coordinate axes of the PGS are defined by two basis cameras, a line and a plane can not be represented in the PGS by the same form of the equation in Euclidean space. Therefore, we need to represent the floor plane by using basis views. We synthesize the floor plane from more than three points extracted from the real background image. In the following, we explain the details of the procedure.

Several correspondence points on the floor region between the two basis view images are extracted as shown in Fig. 5. Those points are picked out manually in our experiment. From

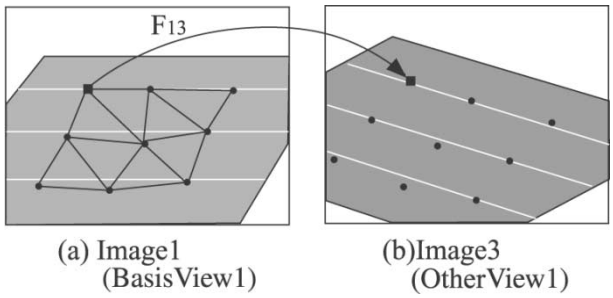

Fig. 6. Projecting the vertex of Delauney triangles onto the other view image using fundamental matrices.

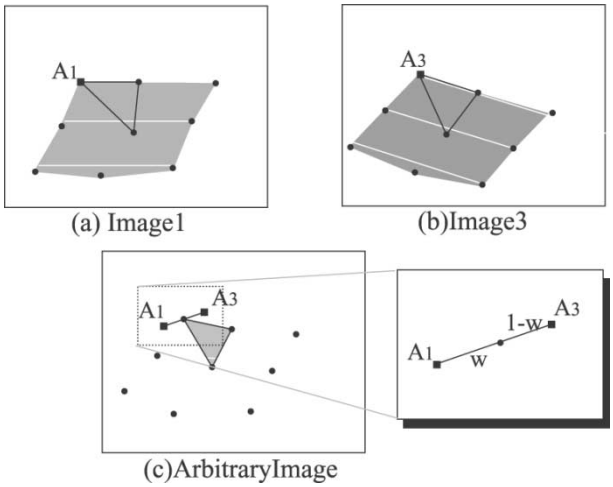

Fig. 7. Synthesizing the floor plane on the arbitrary view image.

Fig. 8. Event hall at B-con Plaza.

the definition of the PGS, the coordinate of correspondence point A is $A1(x1, y1)$ on the first basis view, and $A2(x2, y2)$ on the second, thus the coordinate of A in the PGS becomes $A(x1, y1, x2)$. Since the coordinate of a point in the PGS is fixed, the point can be projected onto every input view image with the fundamental matrices in the same way as stated before.

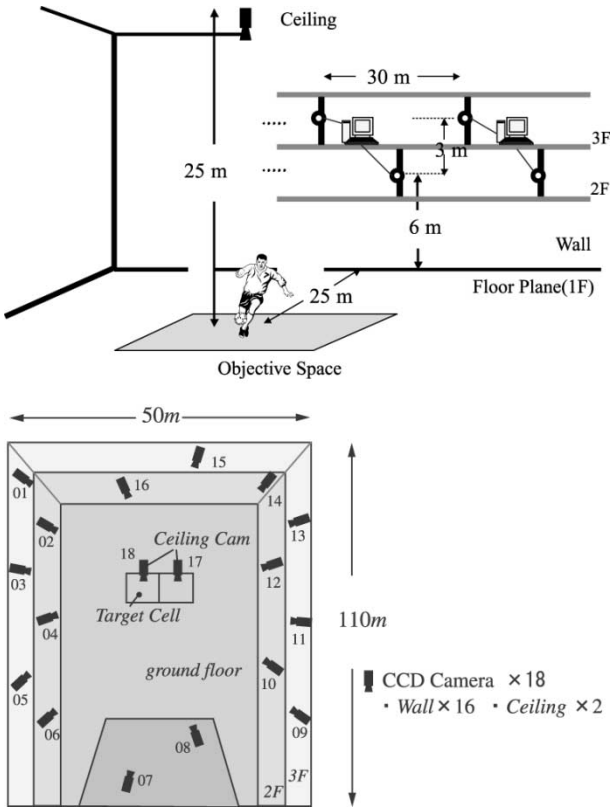

Fig. 9. Camera placement in our system.

Fig. 10. Feature point extraction for estimating fundamental matrices.

The points on the floor are triangulated so that the floor plane can be represented by Delauney triangulation in the first basis view image. The vertices of the triangle mesh are projected onto two interpolating background images using fundamental matrices as shown in Fig. 6.

Synthesizing the background image of an arbitrary viewpoint is done using the same interpolation strategy described in the previous section. The background images require the correspondence of all the points between the two references. According to the correspondence of the vertices of each triangle mesh, affine transforms between the two background images are calculated for each triangle mesh. Since the affine transforms of all the triangle meshes provide pixel-wise correspondence between the two reference background images, all the pixel positions and values are determined in accord with to the way expressed in (5) and (6), as shown in Fig. 7.

Although the use of affine transform is not perspectively correct, we ignore such perspective errors because the distance between the object and the camera is relatively large in the present experiment. In the case of such an approximation that cannot

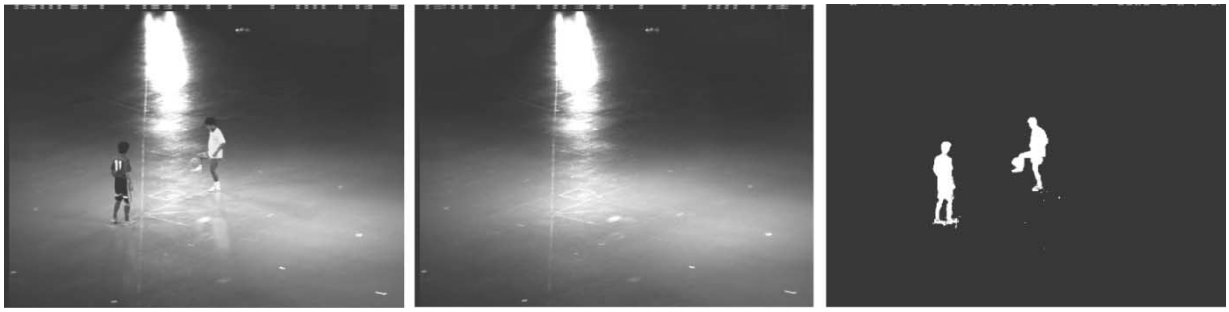

Fig. 11. An example of background subtraction. (a) Input image, (b) background image, and (c) background subtracted image.

be satisfied, we can replace the affine transform with a homographic transform that provides a perspectively correct transformation of the plane between two view images. Adding one point of correspondence inside each triangle mesh allows the homographic transform to be derived.

We should mention that the object region and the floor region have to be synthesized separately. They cannot be rendered simultaneously because the rendering method for each is different. However, it is clear that the floor plane is behind the object, therefore the floor plane is rendered first, and the object is rendered in the foreground.

VI. Experiment

A. Setup

We constructed a multicamera motion picture capturing system in the large event hall at B-con Plaza in Beppu City, Oita, Japan. The hall is $110-\mathrm{m}\,(\mathrm{L})\times 50-\mathrm{m}\,(\mathrm{W})\times 25-\mathrm{m}\,(\mathrm{H})$. We mounted 16 cameras on the walls and two on the ceiling, capturing PC's were connected to every two adjacent cameras, and all PCs were controlled by system control PC. All of those cameras were fully synchronized by pilot signal; all the video signals could be digitized and captured as a color image (640 $\times$ 480 bmp format) sequence at the full video rate (30 fps).

The interior of the event hall looking out from the inside is shown in Fig. 8. The camera positions are shown in Fig. 9. Two cameras are mounted on the ceiling, and 16 cameras are mounted on the wall at two different heights above the ground plane, as shown in Fig. 9(a). The fields of view of the wall cameras are almost equally spaced, as shown in Fig. 9(b). We call the captured objective area a cell. We performed experiments for two cases: a single cell and two cells. In the two-cell case, the cameras were categorized into two groups, so that each group captures a particular area of each cell.

Since the PGS is defined by the basis cameras, the geometrical settings of the base cameras affects the results obtained by the proposed method. We select two basis cameras so that the angle of the viewing direction between the basis cameras is close to 90° to make the axes P, Q and R almost perpendicular to each other. Voxel density in the PGS is not homogeneous because of the perspective effect induced by the basis cameras. Since the field of view of the cameras used in this experiment is less than 10°, the voxel density in the PGS of the objective area is roughly homogeneous. However, if the cameras are close to the objective area, then voxel density will vary depending on the distance from the viewpoint of the basis camera, which might cause poor rendering quality.

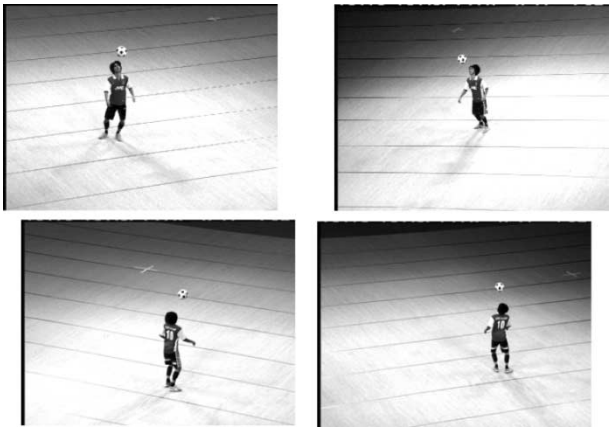

Fig. 12. Examples of real input images.

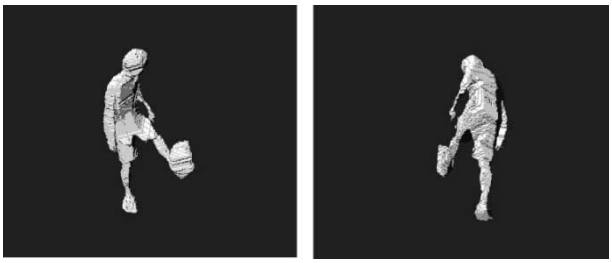

Fig. 13. Reconstructed projective shape in the representation of the orthographic grid space.

B. Pre-Process

The fundamental matrices between the cameras are obtained by putting a checkerboard pattern at various heights, as depicted in Fig. 10, so that the image feature points can be distributed in the objective space. From such images, about 50 image feature points are extracted, and then the same feature points extracted in the other cameras are manually corresponded. Those corresponding feature points between two cameras are used for the estimation of the fundamental matrices. We employ a linear solving method in this experiment.

Note that we do not need the 3-D position of the feature points to estimate the fundamental matrices. If we accurately calibrate the cameras, we need to measure the 3-D position of the feature points, but such measurement requires much effort. We can

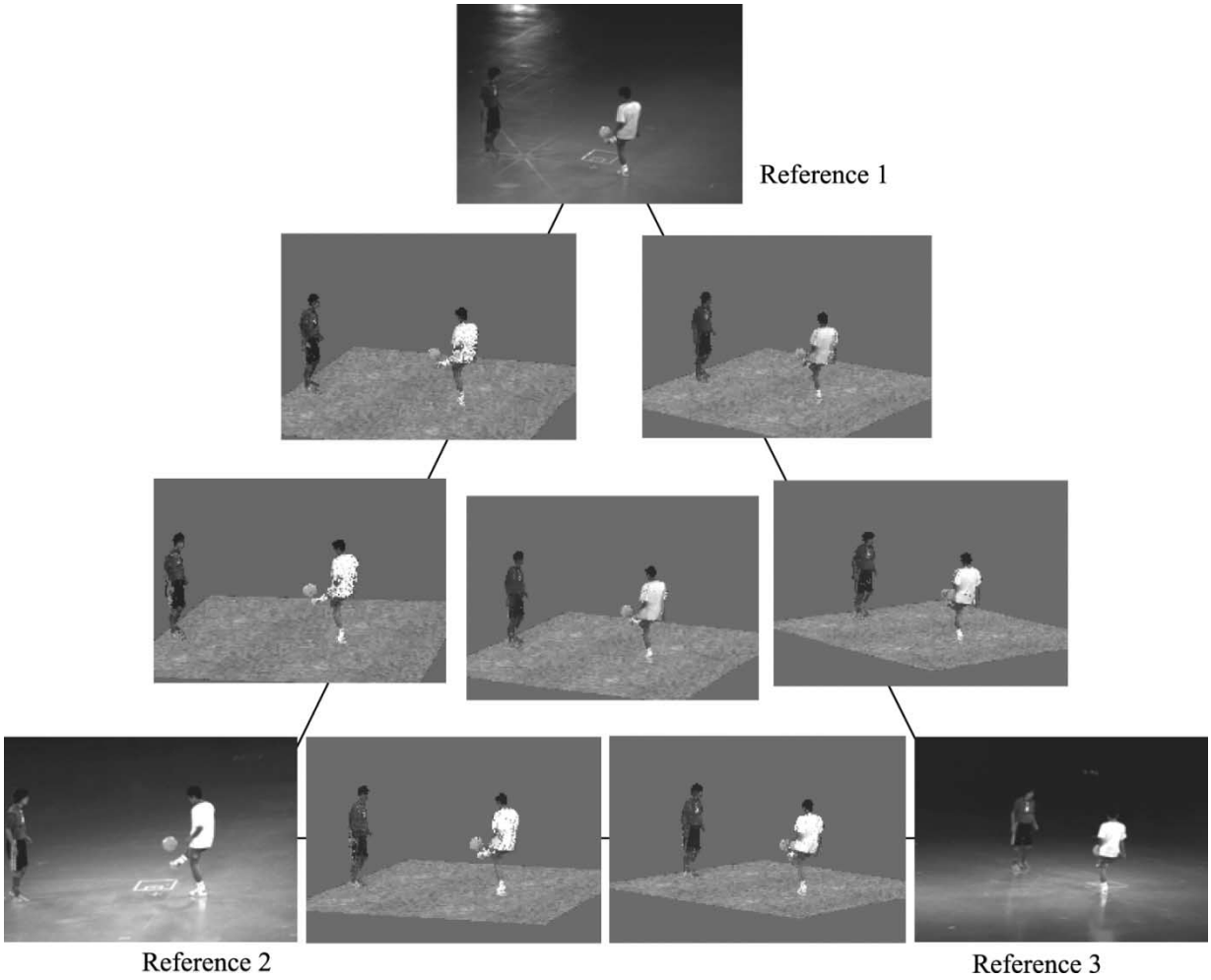

Fig. 14. Synthesized intermediate view images between three reference images.

avoid such effort for measuring the 3-D feature points by applying PGS that is proposed in this paper.

A silhouette image is required for model shape reconstruction in PGS. In this experiment, we perform a simple background subtraction by calculating the simple pixel difference and set a threshold for the difference for determining the silhouette region. Fig. 11 shows an example of this procedure. Since such a simple silhouette extraction strongly depends on the threshold value, some unnecessary regions will also be extracted in some cases, or some holes will be generated in the silhouette regions. In such a case, corrections are done manually in this experiment. However, if we need to make all the procedures run automatically without human interaction, we need to employ an advanced background segmenting methods, such as those described in [13], [20], [21], which remain significant issues to be studied.

VII. Experimental Results

A. Synthesizing Arbitrary View Image From a Single Target Cell

All the cameras were generally pointed at the same target space, where a dynamic object existed. This target space is called a cell. In this section, we show the experimental results obtained for a single target cell. Examples of real captured images are shown in Fig. 12. The silhouette images were generated by background subtraction, and the 3-D shape models were reconstructed in PGS by using the proposed method. The 3-D shape models in the representation of the orthographic grid space, which is seen from the arbitrary view, are shown in Fig. 13.

Fig. 14 shows the synthesized images of a scene in which two men are kicking a ball. Three images of the vertex of the triangle were the selected reference images of the same frame. Arbitrary view images were synthesized from those images by changing the weighting ratio. The images on the line are interpolated from the reference images on either side, and the center of the image is interpolated from the three reference images.

Fig. 15 shows the sequence of a synthesized virtual moving camera image. This is an example of applying our method to an image sequence synthesized by interpolating four reference views.

B. Synthesizing Arbitrary View Image From Several Target Cells

The use of a very large target space, such as a soccer field, requires that we take very high-resolution images to satisfactorily reconstruct precise 3-D shape models. Besides, we can reconstruct only a target object that is observed in all camera images,

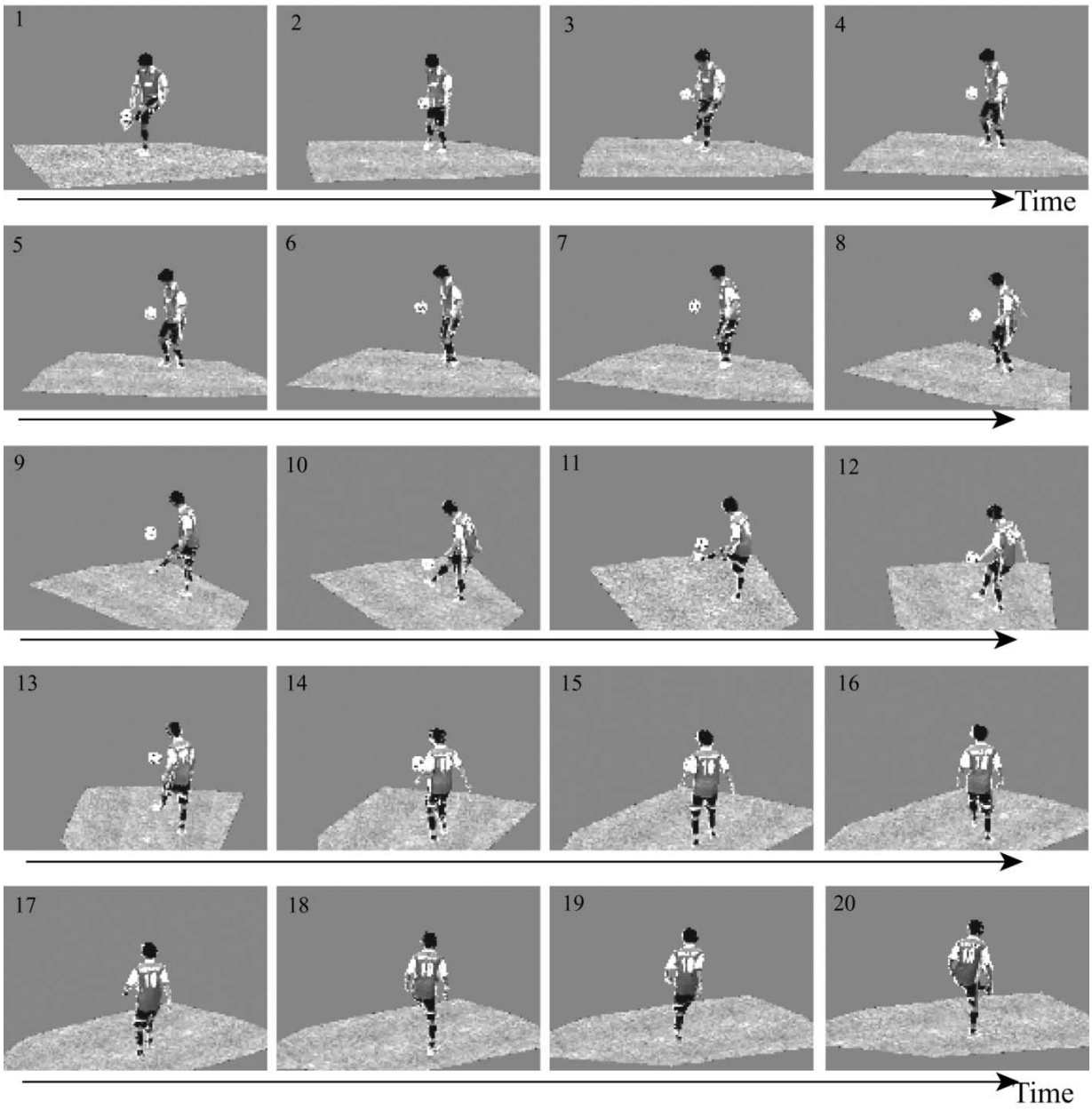

Fig. 15. Image sequence at virtually moving viewpoints for an object with a synthesized floor of grass. Since the shape model is reconstructed in the PGS, the relationship of the occlusion is correctly detected.

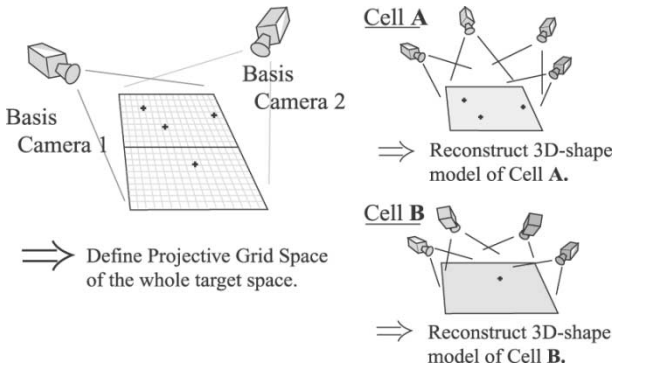

Fig. 16. PGS is defined by two basis cameras that can cover all areas of both cells. Moreover, 3-D shape models are reconstructed in each cell independently.

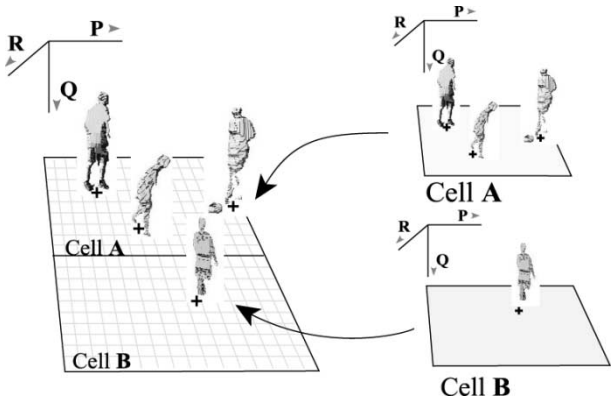

Fig. 17. The 3-D shape models are able to merge, because each cell has common coordinate in the PGS.

because silhouette images from all cameras are required to reconstruct a 3-D shape model. In this section, we propose a method for reconstructing precise 3-D shape models in a large target space, which involves dividing the target space into

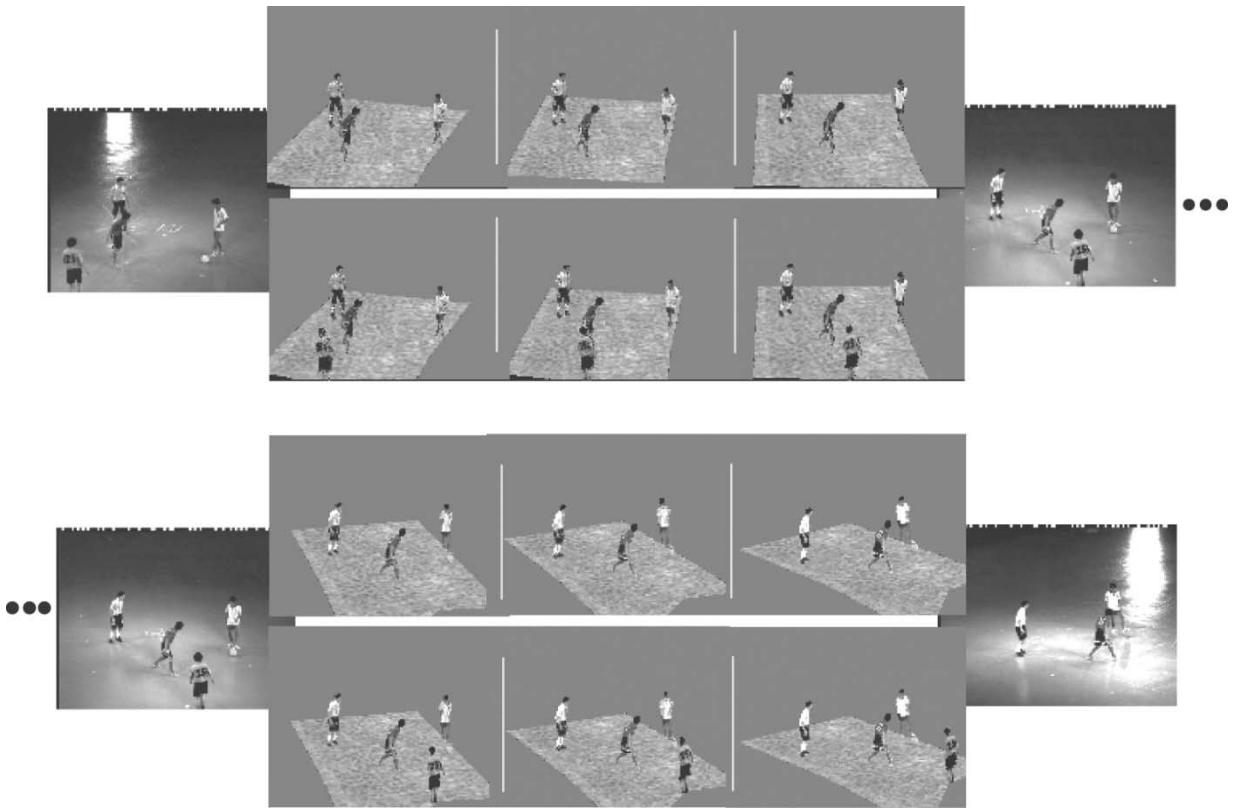

Fig. 18. Synthesized arbitrary view images. Images on either side are reference real images, the middle upper images were synthesized with only the cell $\boldsymbol{A}$ model, and the lower were synthesized with the merged model.

several small subcells and reconstructing a 3-D shape model for every cell.

Here, we consider a target space divided into two cells, cell $\boldsymbol{A}$ and cell $\boldsymbol{B}$. Hence, the PGS is defined as a coordinate common to both cells by two basis cameras, which can cover the entire target space, as shown in Fig. 16. The cameras are divided into two groups, group $\boldsymbol{A}$ and group $\boldsymbol{B}$, to capture images of each cell. All cameras in both groups are related to each basis camera by the fundamental matrices. The 3-D shape model of each cell is reconstructed independently from each camera group by applying the method described in Section IV. The 3-D shape models reconstructed in the cell can be merged into a 3-D shape model of the whole target scene, as shown Fig. 17, because the cells have the same coordinates in PGS. As described in Section IV, every voxel can be projected onto every image that is related to each basis camera by using the fundamental matrix. Therefore, the 3-D model reconstructed in cell $\boldsymbol{B}$ can be projected onto images that have been captured by group-$\boldsymbol{A}$ cameras. Eventually, 3-D shape models reconstructed in PGS that have common coordinates will be projected onto all of the camera images. Thus, the framework of synthesizing arbitrary view images can be applied to the 3-D shape model obtained by merging the different models reconstructed separately in the different cells.

We actually synthesized arbitrary view images by applying our method to a scene captured from two cells. The image capture system was the same used for a single cell, and all cameras were divided into two groups. All wall cameras were mounted at the same height in the hall. Fig. 18 shows synthesized arbitrary view images, where images on either side are reference images and the middle images are images interpolated using the two reference images. The upper images were synthesized from only the 3-D shape model reconstructed in cell A, and the lower images were synthesized from the whole the 3-D shape model by merging the models in the two cells.

In the images synthesized with only the cell-$\boldsymbol{A}$ model, the person wearing the shirt with "25" on the front in the reference image is not synthesized. But, in the images synthesized with the two-cell merged model, that person is synthesized. Because no cameras in group $\boldsymbol{A}$ could "see" the person, the person was removed in the model reconstruction process of the SS method. Therefore, even when the target space is very large, e.g., a soccer field or an American football field, we can synthesize arbitrary view images by dividing the whole target space into several cells and reconstructing a 3-D shape model in each cell separately.

C. Computation Time

We made no effort to minimize the computation times because we were inclined to focus on demonstrating the efficacy of the proposed method in this experiment. However, we do mention the computation time as a reference. Reconstructing the 3-D surface model in PGS took about 30 s. Once the 3-D model is generated, it takes about 0.16 s to render the intermediate images from the reference images, which includes 0.05 s to access the file and 0.04 s to display the image. This was the computation time using a PC that had a 2-GHz Pentium 4 CPU and 2 GB of memory.

VIII. Conclusion

We proposed a method for reconstructing the 3-D shape model in projective grid space (PGS) and synthesizing an arbitrary view image from the multiple image sequences taken with uncalibrated cameras. The PGS can be defined with two basis views, whose relationship is represented by a fundamental matrix. The grid points in the space are related to an arbitrary image by using the fundamental matrices between the image and the two basis views. In the PGS, the shape from the silhouette (SS) method is applied to the reconstruction of the shape model, which provides a dense map of corresponding points between the images for synthesizing intermediate appearance view images. We demonstrated the proposed framework by showing several virtual image sequences generated from multiple-camera image sequences that were corrected in a large space 110-m long $\times$ 50-m wide $\times$ 25-m high.

Acknowledgment

The authors thank the members of the Consortium for the Virtualized Reality Experimental Project in Oita, Japan, with particular thanks to Prof. T. Kanade (Carnegie Mellon University) and Prof. Y. Ohta (University of Tsukuba), for their co-operative efforts in capturing the image sequences.

References

[1] T. Beier and S. Neely, "Feature-based image metamorphosis," in *Proc. of SIGGRAPH '92*, 1992, pp. 35–42.
[2] C. H. Chein and J. K. Aggarwal, "Identification of 3D objects from multiple silhouettes using quadtrees/octrees," *Comput. Vis., Graph., Image Process.*, vol. 36, pp. 100–133, 1986.
[3] S. Chen and L. Williams, "View interpolation for image synthesis," in *Proc. SIGGRAPH '93*, 1993, pp. 279–288.
[4] B. Curless and M. Levoy, "A volumetric method for building complex models from range images," in *Proc. SIGGRAPH '96*, 1996, pp. 303–312.
[5] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The Lumigraph," in *Proc. SIGGRAPH '96*, 1996, pp. 43–54.
[6] A. Hilton, J. Stoddart, J. Illingworth, and T. Windeatt, "Reliable surface reconstruction from multiple range images," in *Proc. ECCV'96*, 1996, pp. 117–126.
[7] A. Katayama, K. Tanaka, T. Oshino, and H. Tamura, "A viewpoint dependent stereoscopic display using interpolation of multi-viewpoint images," *Proc. SPIE*, vol. 2409, pp. 11–20, 1995.
[8] T. Kanade, P. W. Rander, S. Vedula, and H. Saito, "Virtulized reality: digitizing a 3D time-varying event as is and in real time," in *Proc. Int. Symp. Mixed Reality (ISMR99)*, Mar. 1999, pp. 41–57.
[9] A. Laurentini, "The visual hull concept for silhouette based image understanding," *IEEE Trans. Pattern Anal. Machine Intell/*, vol. 16, no. 2, pp. 150–162, 1994.
[10] S. Laveau and O. Faugers, "3-D scene representation as a collection of images," in *Proc. Int. Conf. Pattern Recognition*, 1994.
[11] M. Levoy and P. Hanrahan, "Light field rendering," in *Proc. SIGGRAPH '96*, 1996.
[12] W. Matusik, C. Buehler, R. Raskar, S. Gorlter, and L. McMillan, "Image-based visual hulls," in *Proc. of SIGGRAPH 2000*, 2000, pp. 369–374.
[13] N. Ohta, "A statistical approach to background subtraction for surveillance systems," in *Proc. 8th IEEE Int. Conf. Computer Vision (ICCV 2001)*, vol. 2, 2001, pp. 481–486.
[14] M. Potmesil, "Generating octree models of 3D objects from their silhouettes in a sequense of images," *Comput. Vis., Graph., Image Process.*, vol. 40, pp. 277–283, 1987.
[15] H. Saito and T. Kanade, "Shape reconstruction in projective grid space from large number of images," in *Proc. IEEE Int. Conf. Computer Vision Pattern Recognition*, vol. 2, 1999, pp. 49–54.
[16] H. Saito, S. Baba, M. Kimura, S. Vedula, and T. Kanade, "Appearance-based virtual view generation of temporally-varying events from multi-camera images in 3D room," in *Proc. 2nd Int. Conf. 3D Digital Imaging and Modeling (3DIM '99)*, 1999, pp. 516–526.
[17] D. Scharstein, "View synthesis using stereo vision," *Lecture Notes Comput. Sci.*, vol. 1583, Spring 1999.
[18] S. M. Seitz and C. R. Dyer, "View morphing," in *Proc. SIGGRAPH '96*, 1996, pp. 21–30.
[19] ——, "Photorealistic scene reconstruction by voxel coloring," in *Proc. IEEE Int. Conf. Computer Vision and Pattern Recognition (CVPR97)*, 1997, pp. 1067–1073.
[20] M. Seki, H. Fujiwara, and K. Sumi, "A robust background subtraction method for changing background," in *Proc. 5th IEEE Workshop on Applications of Computer Vision*, 2000, pp. 207–213.
[21] C. Stauffer and E. Grimson, "Similarity templates for detection and recognition," in *Proc. IEEE Int. Conf. Comput. Vision Pattern Recognition (CVPR01)*, vol. 1, 2001, pp. 221–228.
[22] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses," *IEEE J. Robot. Automat.*, pp. 323–344, 1987.
[23] S. Vedula, P. W. Rander, H. Saito, and T. Kanade, "Modeling, combining, and rendering dynamic real-world events from image sequences," in *Proc. 4th Conf. Virtual Systems and Multimedia*, vol. 1, 1998, pp. 326–322.
[24] M. D. Wheeler, Y. Sato, and K. Ikeuchi, "Consensus surfaces for modeling 3D objects from multiple range images," in *Proc. DARPA Image Understanding Workshop*, 1997, pp. 1229–1236.
[25] Z. Zhang, "Determining the Epipolar Geometry and its Uncertainly: A Review," INRIA res. rep. 2927, 1996.

Satoshi Yaguchi received the B.E. degree in Information and Computer Science, and M.E. degree in open and environmental systems from Keio University, Japan, in 2000 and 2002, respectively.

He has been engaged in the research areas of computer vision. Since 2002, he has been with the NTT Comware Corp., Japan.

Hideo Saito (M'92) received the B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University, Japan, in 1987, 1989, and 1992, respectively.

He has been on the faculty of Department of Electrical Engineering, Keio University, since 1992. From 1997 to 1999, he was a Visiting Researcher of the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA. Since 2001, he has been an Associate Professor with Department of Information and Computer Science, Keio University. Since 2000, he has also been a Researcher of PRESTO, JST. He has been engaging in the research areas of computer vision, image processing, and human–computer interaction.

Dr. Saito is a member of IEICE, IPSJ, and SICE, Japan.