# Appearance-Based Virtual View Generation of Temporally-Varying Events from Multi-Camera Images in the 3D Room

Hideo Saito, Shigeyuki Baba
Makoto Kimura, Sundar Vedula, Takeo Kanade

April 29, 1999
CMU-CS-99-127

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

**Abstract**

In this paper, we present an "appearance-based" virtual view generation method for temporally-varying events taken by multiple cameras of the "3D Room", developed by our group. With this method, we can generate images from any virtual view point between two selected real views. The virtual appearance view generation method is based on simple interpolation between two selected views. The correspondence between the views are automatically generated from the multiple images by use of the volumetric model shape reconstruction framework. Since the correspondences are obtained by the recovered volumetric model, even occluded regions in the views can be correctly interpolated in the virtual view images. The virtual view image sequences are presented for demonstrating the performance of the virtual view image generation in the 3D Room.

This research was supported by Robotics Institute internal funds. Also, partial support was provided by Intel Corporation, Matsushita Electric Ltd., Sony Corporation, and K2T, Inc.

**Keywords:** Virtualized reality, 3D scene modeling, Multi-camera images, Image based rendering, View interpolation

## 1 Introduction

The technology of 3D shape reconstruction from multiple view images has recently been intensely researched, mainly because of advances of computation power and capacity of data handling. Research in 3D shape reconstruction from multiple view images has conventionally been applied in robot vision and machine vision systems, in which the reconstructed 3D shape is used for recognizing the real scene structure and object shape. For those kinds of applications, the 3D shape itself is the target of the reconstruction.

New applications of 3D shape reconstruction have recently been introduced, one of which is arbitrary view generation from multiple view images. The new view images are generated by rendering pixel colors of input images in accordance with the geometry of the new view and the 3D structure model of the scene. The 3D shape reconstruction techniques can be applied to recover the 3D model that is used for generating new views. Such a framework for generating new views via recovery of a 3D model is generally called as "model-based rendering".

On the contrary, image-based rendering (IBR) has recently been developed for generating new view images from multiple view images without recovering the 3D shape of the object. Because IBR is essentially based on 2D image processing (cut, warp, paste, etc.), the errors in 3D shape reconstruction do not affect the quality of the generated new images as much as for model-based rendering. This implies that the quality of the input images can be well preserved in the generated new view images.

In this paper, we present an a virtual view generation method based on the IBR framework for temporally-varying events taken by multiple camera images of the "3D Room"[10], which we have developed for digitizing dynamic events, as is and in their entirety. With this method, we can generate images from any virtual appearance viewpoint which is specified by the relative position of the view points of input images. This way of specifying the virtual view point is not controlled by the explicit 3D position in the object space, but rather by location relative to the input cameras.

The virtual appearance views are generated in accordance with the correspondence between images. Even though the image generation procedure is based on a simple 2D image morphing process, the generated virtual view images reasonably represent 3D structure of the scene because of the 3D structure information included in the correspondence between the images. The correspondence between images are automatically detected by the 3D reconstruction algorithm [14] which we developed for our previous multiple camera systems. The 3D structure recovery helps to avoid the occlusion problem between the images used to generate virtual view images.

We demonstrate the performance of the proposed framework for virtual view generation from multiple cameras by showing several virtual image sequences of a dynamic event.

## 2 Related Work

Recent research in both computer vision and graphics has made important steps toward generating new view images. This work can be broken down into two basic groups: generating new view images from 3D structure models that are reconstructed from range images (so called "model-based rendering"), and generating new view images directly from multiple images (so called "image-based rendering"). 3D structure reconstruction using volumetric integration of range images, such as Hilton et. al. [7], Curless and Levoy [3], and Wheeler et. al. [21], led to several robust approaches to recovering global 3D geometry. Most of this work relies on direct range-scanning hardware, which is too slow and costly for a dynamic multiple sensor modeling system. Our method does not use a range-scanner but applies image-based stereo for generation of range images [14]. Debevec et. al. [5] use a human editing system with automatic model refinement to recover 3D geometry and a view-dependent texture mapping scheme to the model.

This structure recovery method does not map well to our objectives because of the human modeling time.

Image-based rendering has also seen significant development. Katayama et. al. demonstrated that images from a dense set of viewing positions on a plane can be directly used to generate images for arbitrary viewing positions [11]. Levoy and Hanrahan [12] and Gortler et al. [6] extend this concept to construct a four-dimensional field representing all light rays passing through a 3D surface. New view generation is posed as computing the correct 2D cross section of the field of light rays. A major problem with these approaches is that thousands of real images may be required to generate new views realistically, therefore making the extension to dynamic scene modeling impractical. View interpolation [2, 20] is one of the first approaches that exploited correspondences between images to project pixels in real images into a virtual image plane. This approach linearly interpolates the correspondences, or flow vectors, to predict intermediate viewpoints. View morphing [16] is an extension of image morphing [1], that correctly handles the 3D geometry of multiple views. The method presented in our paper is based on this view interpolation framework.

There are other bodies of work involving multiple camera systems. Our group has developed a system using a number of cameras for digitizing whole real world events including 3D shape information of dynamic scenes [9, 14, 19]. Davis et. al. [4] have developed a multiple camera system for human motion capturing without any sensors on the human body. Jain et. al. [8] proposed Multiple Perspective Interactive (MPI) Video, which attempts to give viewers control of what they see, by computing 3D environments for view generation by combining a priori environment models and the dynamic pre-determined motion models. Even in the single camera case, 3D structure recovery from a moving camera involves using a number of images around the object [15, 17].

## 3 3D Room

The "3D room" is a facility for "4D" digitization - capturing and modeling a real time-varying event into computers as 3D representations which depend on time (1D). On the walls and ceiling of the room, a large number of cameras are mounted, all of which are synchronized with a common signal. Our 3D Room [10] is 20 feet (L) $\times$ 20 feet (W) $\times$ 9 feet (H). As shown in figures 1 and 2, 49 cameras are currently distributed inside the room : 10 cameras are mounted on each of the four walls, and 9 cameras on the ceiling. A PC cluster computer system (currently 17 PCs) can digitize all the video signals from the cameras simultaneously in real time as uncompressed and lossless color images at full video rate ($640 \times 480 \times 2 \times 30$ bytes per seconds). Figure 3 shows the diagram of the digitization system. The images thus captured are used for generating the virtual view image sequences in this paper.

## 4 Appearance-based virtual view generation from multiple camera images

Figure 4 shows the overview of the procedure for generating virtual view images from multiple image sequences collected in the 3D Room.

The input image sequences provide depth image sequences by applying multiple baseline stereo frame by frame. The depth images of all cameras are merged into a sequence of 3D shape models, using a volumetric merging algorithm.

For controlling the appearance-based virtual view point in the 3D Room, two interpolating cameras are selected. The intermediate images between the selected two images are generated by interpolation of the selected images from the correspondence between the images. The corresponding points are computed by using the 3D shape model. The weighting value between the images controls the appearance of the virtual

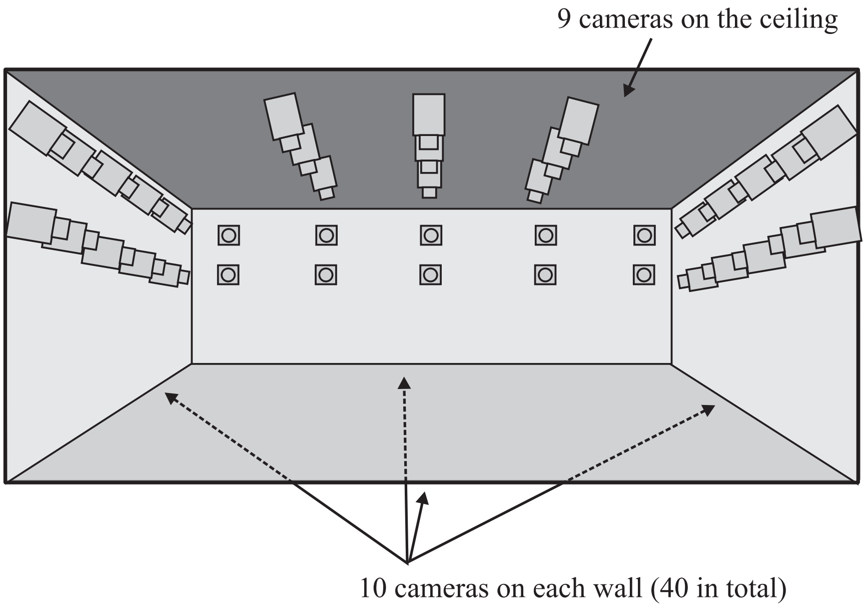


Figure 1: Camera placement in the 3D Room.

Figure 2: Panoramic view of the 3D Room.

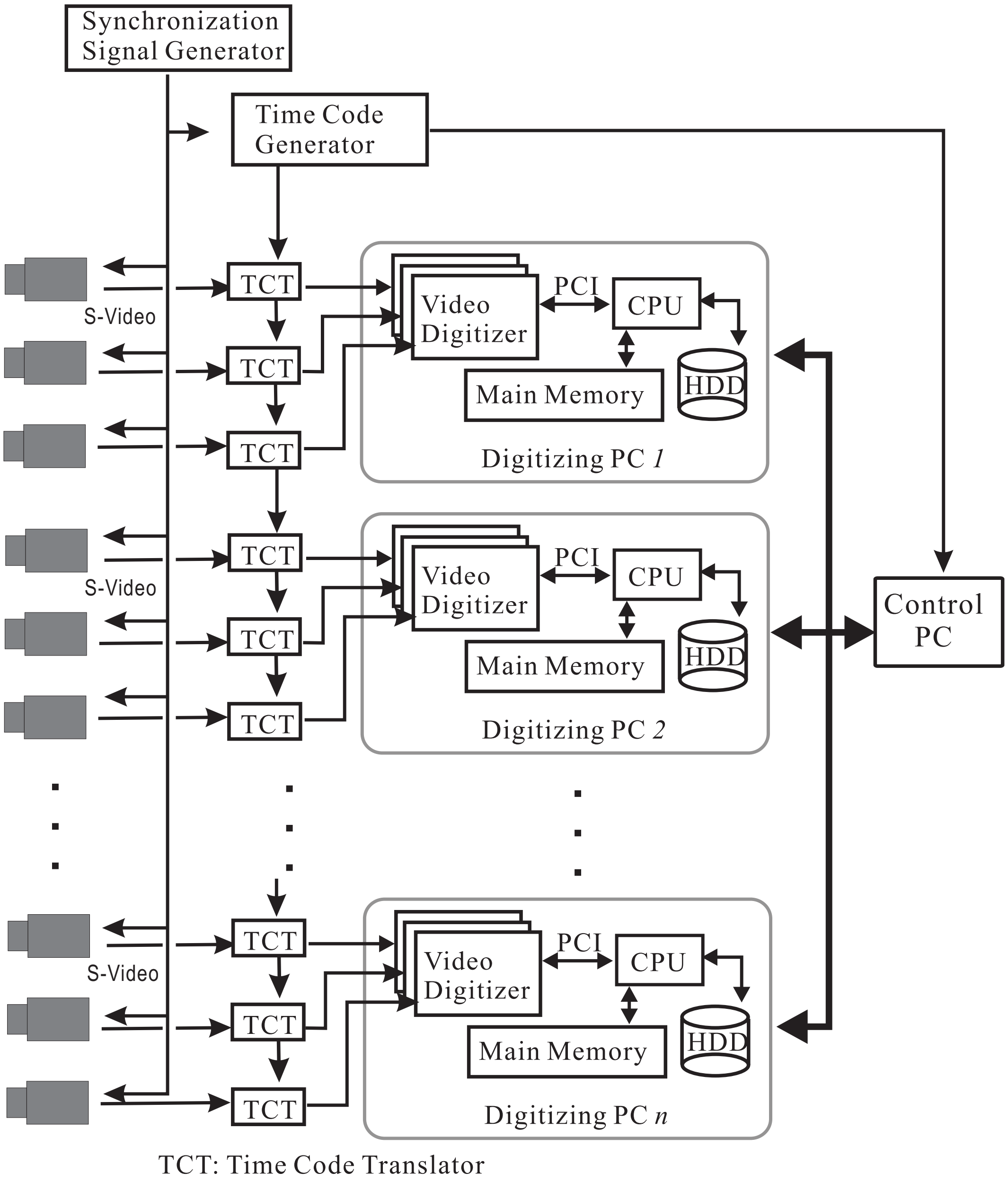


Figure 3: The digitization system of the 3D Room consists of 49 synchronized cameras, one time code generator, 49 time code translators, 17 Digitizing PCs and one Control PC.

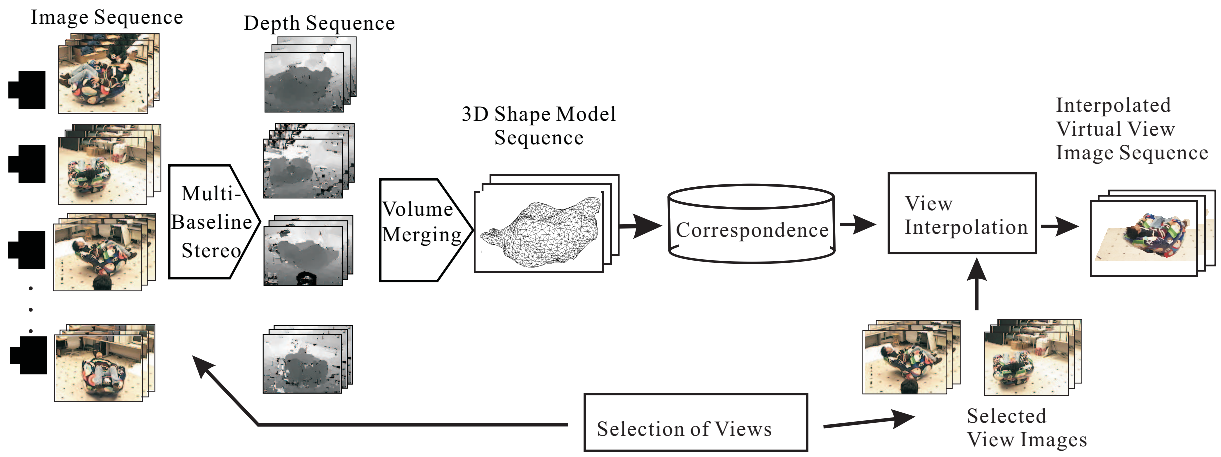


Figure 4: Overview of the procedure for generating virtual view images from multiple camera in the 3D Room.

view point.

In this way, virtual view image sequences can be generated. In the following subsections, we explain the details of the procedure.

### 4.1 3D shape model computation

Multiple baseline stereo (MBS) [13] is employed for obtaining a depth image for every camera in the 3D Room. Some (2-4) neighboring cameras are selected for computing the MBS of every camera. All the depth images for all cameras are merged to generate the volumetric model.

This volumetric merging generates a 3D shape model of the object that is represented by a triangle mesh. The volume of interest is specified during the volumetric merging so that only the objects of interest can be extracted. An example of the reconstructed 3D shape model in, as a triangle mesh representation is shown in figure 5.

For reconstructing the 3D shape models from multiple images, each camera has to be fully calibrated prior to the reconstruction procedure. We use Tsai's camera calibration method [18], which calculates six degrees of freedom of rotation and translation for extrinsic parameters, and five intrinsic parameters which are focal length, aspect ratio of pixel, optical center position, and first order radial lens distortion.

To estimate the camera parameters of all cameras, we put point light sources (LEDs) in the volume of interest, and capture images from all cameras. The placement of the point light sources in the volume of interest is shown in figure 6, where the plate that has $8 \times 8$ LEDs at an interval of 300mm is placed at 5 vertical positions, displaced 300mm from each other. The images of these point light sources provide the relationship of the 3D world coordinates to the 2D image coordinates for every camera. The camera parameters are estimated from this relationship by a non-linear optimization [18].

### 4.2 Correspondence from 3D model

The 3D shape model of the object is used to compute correspondences between any pair of views. Figure 7 shows the scheme for making correspondences in accordance with the 3D shape model. For a point in view 1, the intersection of the pixel ray with the surface of the 3D model is computed. The 3D position of

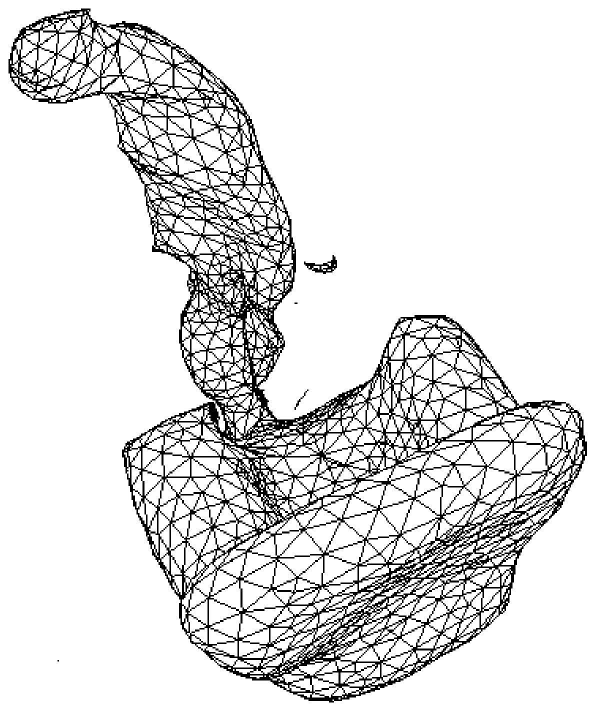

Figure 5: An example of the reconstructed 3D shape model, using a triangle mesh representation. The number of triangles in the mesh is 10,000.

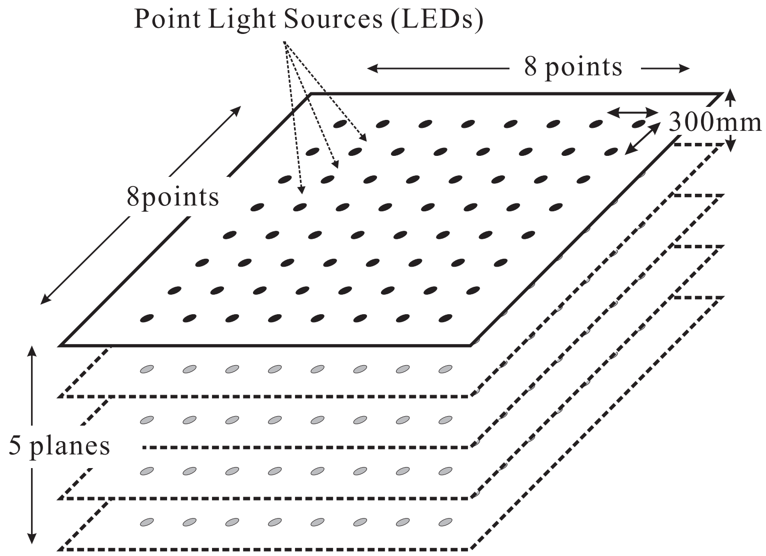


Figure 6: Placement of point light sources for calibration of every camera in the 3D Room.

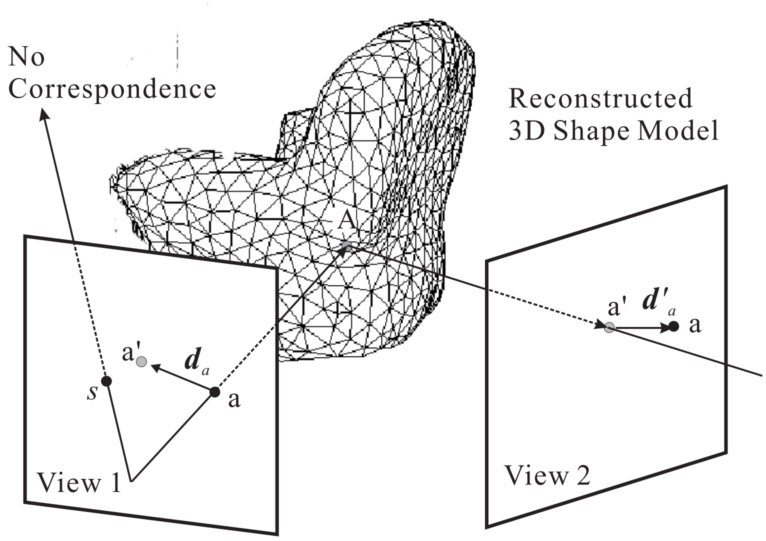


Figure 7: The scheme for making correspondences in accordance with a 3D shape model.

the intersecting point is projected onto the other image, view 2. The projected point is the corresponding point of the pixel in view 1.

In figure 7, the ray of the point $a$ intersects the surface at $A$, and is then projected onto the point $a'$. In this case, the point $a'$ in view 2 is the corresponding point for the point $a$ in view 1.

If there is no intersection on the surface (like the point $s$ in view 1), the pixel does not have any corresponding point.

For each point that has corresponding point, the disparity vector of correspondence is defined. The disparity vector $\mathbf{d}_a$ for the point $a$ is the flow vector from $a$ to $a'$. The disparity vector $\mathbf{d}'_a$ for the point $a'$ is the flow vector from $a'$ to $a$.

### 4.3 Virtual view generation

For controlling the appearance-based virtual view point in the 3D Room, two cameras are selected initially. The intermediate images between the selected two images are generated by interpolation of the selected images from the correspondence between them. The correspondence is computed by using the 3D shape model as described above. The weighting value between the images controls the virtual view point in the sense of the appearance.

The interpolation is based on the related concepts of "view interpolation" [2] and "view morphing" [1], in which the position and color of every pixel are interpolated from the corresponding points in two images. The following equations are applied to the interpolation:

$$\mathbf{P}_i = w_1 \mathbf{P} + w_2 \mathbf{P}', \tag{1}$$

$$I_i(\mathbf{P}_i) = w_1 I(\mathbf{P}) + w_2 I'(\mathbf{P}'), \tag{2}$$

where

$$w_1 + w_2 = 1,$$

$\mathbf{P}$ and $\mathbf{P}'$ are the position of the corresponding points in the two views (view 1 and view 2), $I(\mathbf{P})$ and $I'(\mathbf{P}')$ are the colors of the corresponding points, and $\mathbf{P}_i$ and $I(\mathbf{P}_i)$ are the interpolated position and color. The interpolation weighting factors are represented by $w_1$ and $w_2$ ($w_1 + w_2 = 1$).

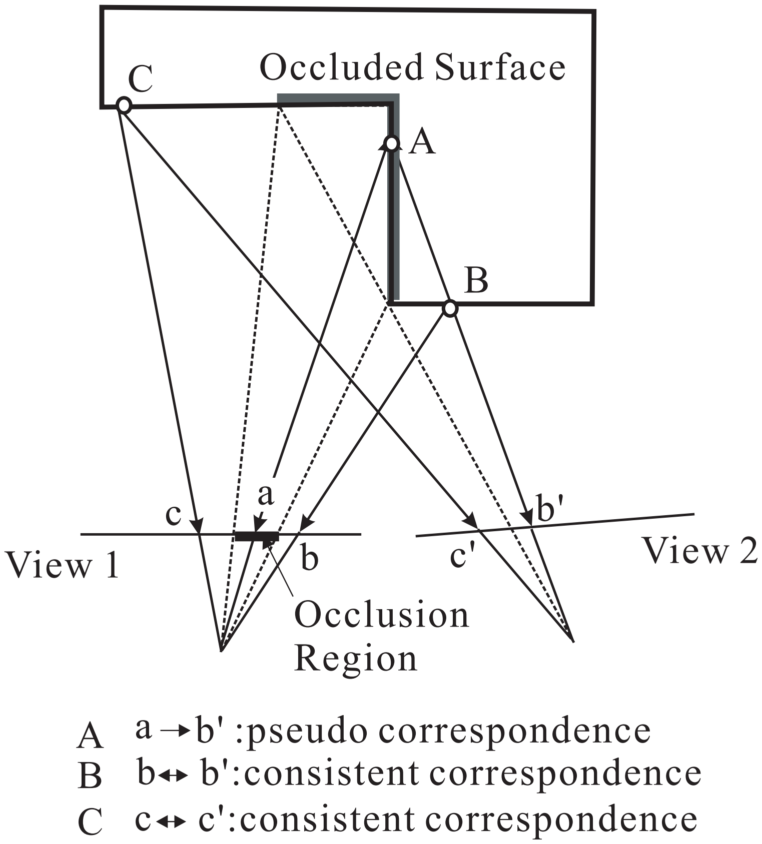


Figure 8: Consistent correspondence and pseudo correspondence. As the point a is in an occluded region, there is no corresponding point in view B. The pseudo correspondence from the point a is provided by the 3D shape of the object, by virtually projecting the surface point onto the view B (represented as b'). The point b' is not only the pseudo corresponding point from a, but also the real corresponding point from b in the view A.

This interpolation method requires consistent correspondence between two images. However, there is the case that some regions in one image cannot be seen in another image, as shown in figure 8. In this case, interpolation of the color between two views by the method described by equations (1) and (2) is impossible. As a result, there is no description of the color of the occlusion region in the generated interpolated images.

To avoid such problems in view interpolation, we introduce the concept of a "pseudo corresponding point" which can be computed for the 3D shape of the scene. In figure 8, a point $a$ in view 1 is re-projected onto $b'$ in view 2 by using the reconstructed shape, even though the point on the object surface cannot be seen in view 2. The point $b'$ is the pseudo corresponding point for $a$, that corresponds to $a$ only in the geometrical sense. The pseudo corresponding point enables the interpolation of the position by applying the equation (1) for occluded points. The interpolation of the color is still impossible because the color of the pseudo corresponding point is not actually corresponding in terms of the color of the image. Accordingly, the color is not interpolated for the pseudo correspondence, but just selected to be the color of the occluded point. This is expressed by the following equation.

$$I_i(\mathbf{P}_i) = \begin{cases} I(\mathbf{P}) & \text{if } \mathbf{P} \text{ is not seen in view 2} \\ I'(\mathbf{P}') & \text{if } \mathbf{P}' \text{ is not seen in view 1} \end{cases} \tag{3}$$

By using the pseudo corresponding point, we can generate intermediate view images without missing parts of occlusion regions.

Pseudo corresponding points can be detected only if the 3D structure of the scene is available. This suggests that 3D shape reconstruction plays an important role in view interpolation between two views, even though the interpolation procedure involves only 2D image processing without any concept of a 3D structure.

In figure 9, the effect of the pseudo correspondence is presented. If only the two input images are given without any 3D shape information, the points in the occlusion regions in view 1 (circled areas) cannot have any corresponding points, because no information is available for making the points in the occlusion regions that correspond to the image of the view 2. Therefore, the colors in the occlusion region completely vanish in the interpolated view images as shown in figure 9(a). On the contrary, the complete 3D shape model, which is reconstructed by the volumetric merging of the depth images at all cameras, enables us to compute the pseudo correspondence even for the occlusion region. Because of the pseudo correspondences, the occlusion region can be successfully interpolated in the virtual view as shown in figure 9(b).

### 4.4 View interpolation algorithm

For implementing the interpolation by the pseudo correspondence, we take the two-step algorithm, where two warped images of two interpolating real images are first generated in accordance with the disparity of correspondence, and then the two warped images are blended. Figure 10 shows this algorithm.

As described in section 4.2, disparity vector images $\mathbf{d}(u,v)$ and $\mathbf{d}'(u,v)$ are computed for two interpolating images $I(u,v)$ and $I'(u,v)$ of view 1 and view 2, respectively. For each interpolating image, the warped image is generated by shifting the pixel in the weighted disparity vector. The relation between the warped images $I_w(u,v)$ and $I'_w(u,v)$ and input images$I(u,v)$ and $I'(u,v)$ is

$$\begin{aligned} I_w(u + w_1 d_u(u,v), v + w_1 d_v(u,v)) &= I(u,v), \\ I'_w(u + w_2 d'_u(u,v), v + w_2 d'_v(u,v)) &= I'(u,v). \end{aligned} \tag{4}$$

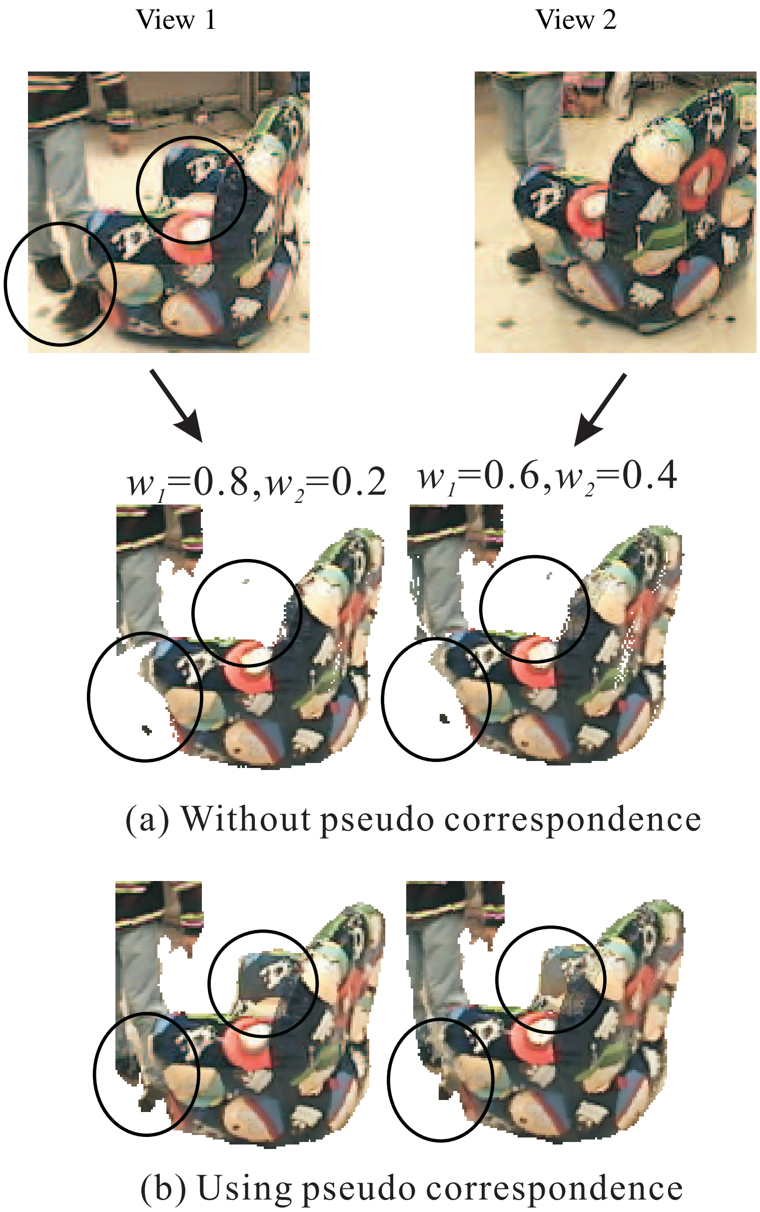


Figure 9: Effect of the pseudo correspondence in view interpolation. If we only have two views, there is no way to compute the correspondences correctly for the occlusion regions in view 1 (circled areas). Therefore the colors in this regions do not exist in the interpolated view images (a). On the other hand, the 3D shape model provides the pseudo corresponding points for the occlusion regions in view 1. Therefore the colors in those regions appear in the interpolated view images (b).

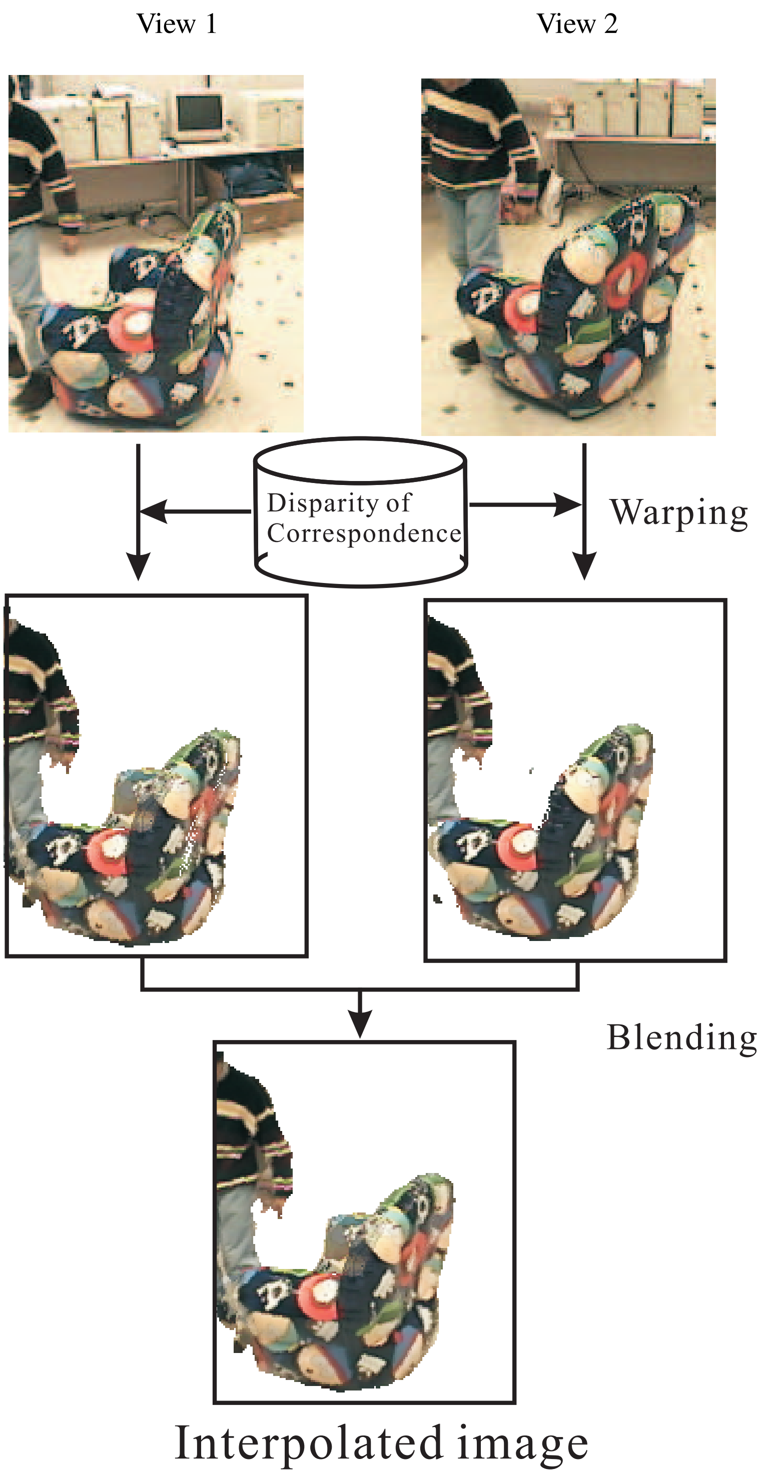


Figure 10: Interpolation between two views. Each image is warped by the weighted disparity of correspondences. The warped images are then blended for generating the interpolated image.

Since the disparity value is not limited to an integer but a floating point value, the shifted pixel can be placed on any point that is not coincident with the pixel sampling point. The color value on the pixel point is computed by bilinearly interpolating from the neighboring shifted color values.

The two warped images are blended into the interpolated image of the two input images according to the following equation.

$$I_i(u,v) = \begin{cases} w_1 I(u,v), \\ \quad \text{if } I(u,v) \neq 0 \text{ and } I'(u,v) = 0, \\ w_2 I'(u,v), \\ \quad \text{if } I(u,v) = 0 \text{ and } I'(u,v) \neq 0, \\ w_1 I(u,v) + w_2 I'(u,v), \text{otherwise}, \end{cases} \tag{5}$$

If a warped pixel color is shifted by the pseudo corresponding point, the color value is computed in only one warped image. This corresponds to the first two cases in this equation.

## 5 Examples of the virtual view

The interpolated view images using various weighting factors for the fixed instance are shown in figure 11. This demonstrates that the virtual view smoothly moves as the weight factor is changed. As can be seen, the occlusion regions have successfully been interpolated in the virtual view images.

Figure 12 shows the virtual view images for a dynamic event. The real view images used for interpolation are also shown with the virtual view images. This figure also demonstrates that the occlusion region has correctly been interpolated in the image sequence.

Figure 13 shows an example of virtual view image sequence for a dynamic event modeled by multiple cameras. As seen in this figure, the virtual viewpoint can move between multiple pairs of cameras for generating the long trajectory of the virtual view.

## 6 Conclusions and Future Work

We have presented a method for virtual view image generation from multiple image sequences collected in the 3D Room. The virtual view generation method is based on view interpolation of two views from correspondences that are provided by a volumetric model recovered from input multiple views. Occlusion regions can be interpolated using the pseudo correspondence information that represents only geometrical correspondence.

This framework for virtual view generation falls into the IBR framework because we do not use 3D structure information directly at the image generation stage. However, the correspondence used for the generation cannot be obtained without the 3D structure information which is recovered by volumetric merging of the depth images provided by a Multiple Baseline Stereo algorithm. In this sense, 3D recovery is a significant component in generating new view images even if the generation procedure is based on 2D image processing.

In the present method, we do not take into account the geometrical correctness of the interpolated virtual view because we currently only use simple correspondences between images. However, as Seitz et .al. [16] pointed out in view morphing, such simple correspondence interpolation can not correctly interpolate the geometry of the views. For more realistic new view generation, such correctness of the geometry has to be considered also.

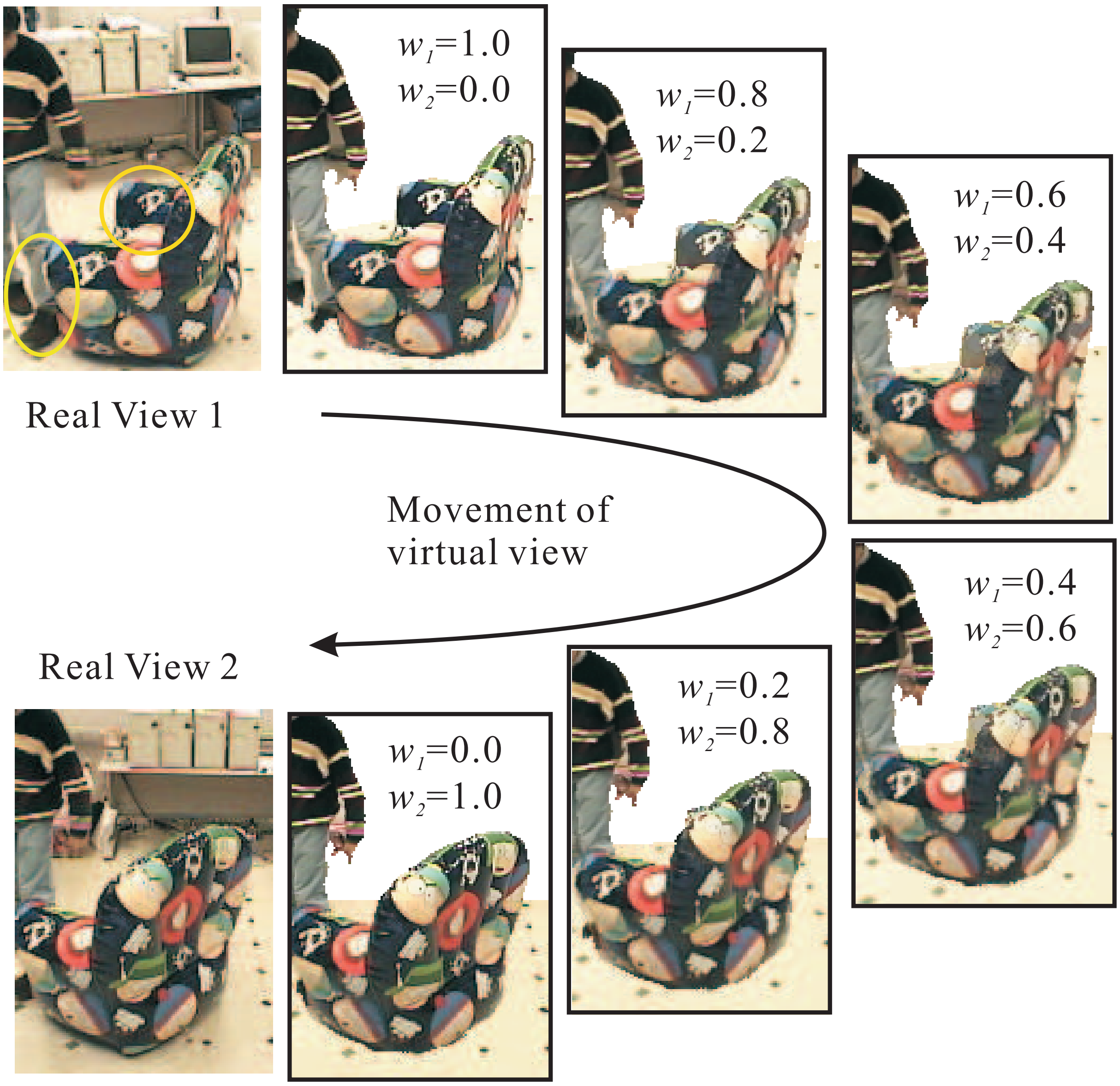


Figure 11: Generated virtual view images by interpolation of two real view images using various weighting factors for the fixed instance. The virtual view smoothly moves as the weight factor is changed. The occlusion regions (circled areas) have successfully been interpolated in the virtual view images.

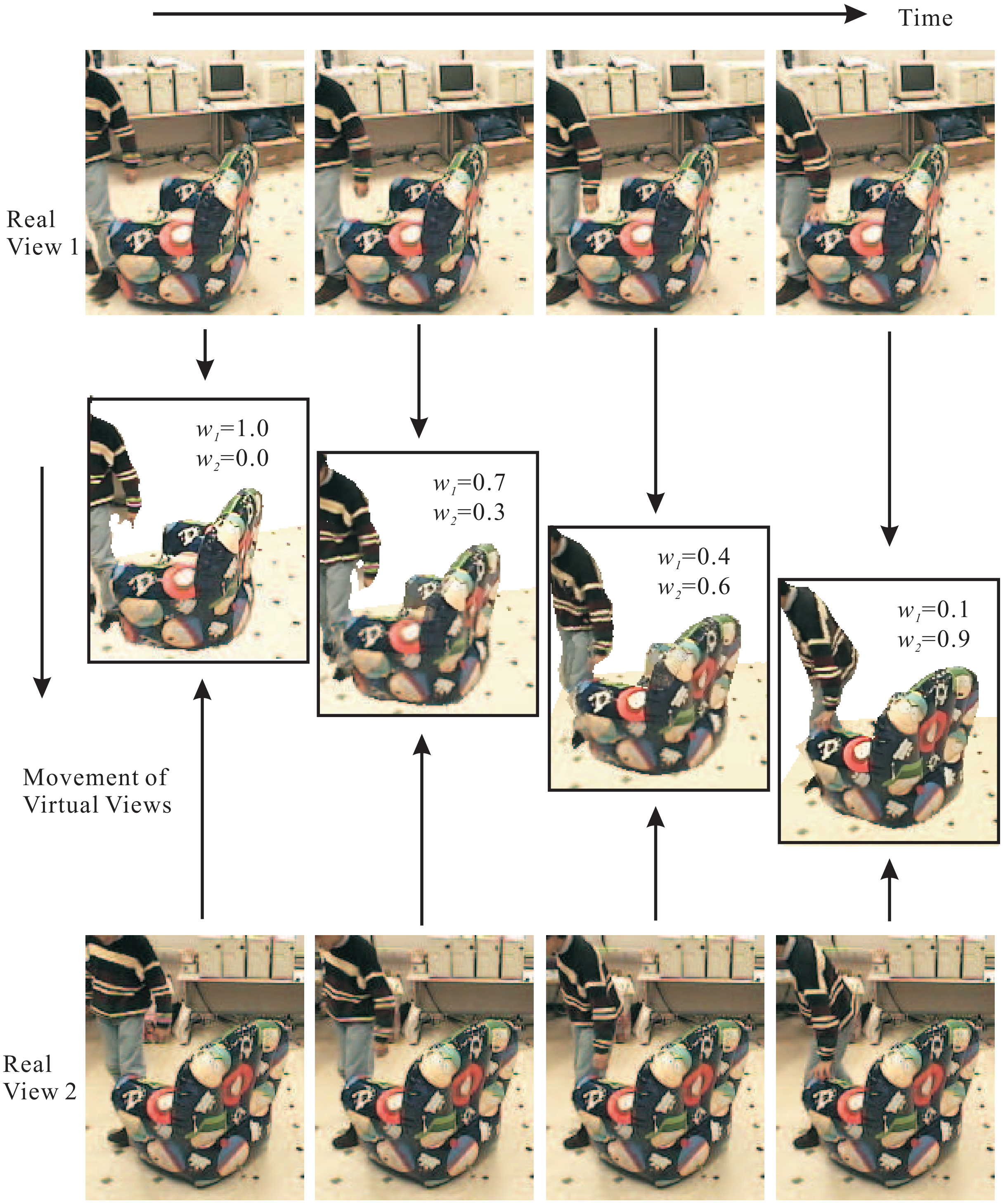


Figure 12: Generated virtual view images for a dynamic event. The real view images used for interpolation are also shown with the virtual view images. The occlusion region has correctly been interpolated in the image sequence.

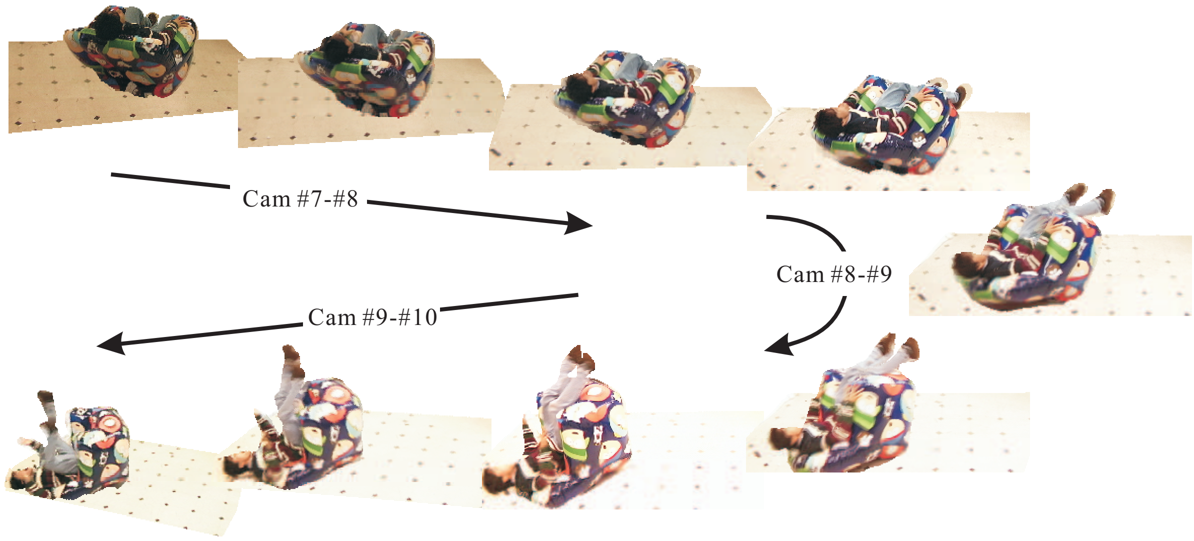


Figure 13: Generated virtual view images for a dynamic event for four cameras (cameras #7, #8, #9, #10).

We currently interpolate new views from two views. This means that the virtual camera can only move on the line between the views. We plan to extend our framework to the interpolation of three camera views to make the virtual view move on the plane of these three cameras.

## References


[1] T. Beier, S.Neely, "Feature-Based Image Metamorphosis", *Proc. of SIGGRAPH'92*, pp.35-42, 1992.

[2] S. Chen, and L.Williams, "View Interpolation for Image Synthesis", *Proc. of SIGGRAPH'93*, pp.279-288, 1993.

[3] B. Curless and M. Levoy, "A Volumetric Method for Building Complex Models from Range Images", *Proc.of SIGGRAPH '96*, 1996.

[4] D.M.Gavrila and L.S.Davis, "3-D Model Based Tracking of Humans in Action : Multi-View Approach",*Proc. Computer Vision and Pattern Recognition 96*, pp. 73-80, 1996.

[5] P. Debevec, C. Taylor, and J. Malik, "Modeling and Rendering Architecture from Photographs: A Hybrid Geometry- and Image-based Approach", *Proc. of SIGGRAPH'96*, 1996.

[6] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen, "The Lumigraph", *Proc. of SIGGRAPH'96*, 1996.

[7] A. Hilton, J. Stoddart, J. Illingworth, and T. Windeatt, "Reliable Surface Reconstruction From Multiple Range Images", *Proceedings of ECCV'96* pp.117-126, 1996.

[8] R. Jain and K. Wakimoto, "Multiple Perspective Interactive Video", *Proceedings of IEEE Conference on Multimedia Systems*, 1995.

[9] T. Kanade, P. W. Rander, and P. J. Narayanan, “Virtualized Reality: Constructing Virtual Worlds from Real Scenes”, *IEEE MultiMedia*, Vol.4, No.1, 1997.

[10] Takeo Kanade, Hideo Saito, and Sundar Vedula, “The 3D Room: Digitizing Time-Varying 3D Events by Synchronized Multiple Video Streams”, *CMU-RI-TR-98-34*, 1998.

[11] A. Katayama, K. Tanaka, T. Oshino, and H. Tamura, “A view point dependent stereoscopic display using interpolation of multi-viewpoint images”, *SPIE Proc.* Vol. 2409: Stereoscopic Dis plays and Virtual Reality Systems II, pp.11-20, 1995.

[12] M. Levoy and P. Hanrahan, “Light Field Rendering”, *Proc. of SIGGRAPH’96*, 1996.

[13] M. Okutomi and T.Kanade, “A Multiple-Baseline Stereo”, *IEEE Trans. on PAMI*, Vol.15, No.4, pp.353-363, 1993.

[14] P. J. Narayanan, P.W.Rander, and T.Kanade, “Constructing Virtual Worlds using Dense Stereo”, Proc. ICCV ‘98, 1998.

[15] M.Pollefeys, R.Koch, and L.V. Gool, “Self-Calibration and Metric Reconstruction in spite of Varying and Unknown Internal Camera Parameters”, Proc. ICCV98, pp.90-95, 1998.

[16] S.M.Seitz and C.R.Dyer, “View Morphing”, *Proc. of SIGGRAPH’96*, pp.21-30, 1996.

[17] C.Tomasi and T.Kanade, “Shape and Motion from Image Streams Under Orthography: A Factrization Method”, *Int’l J. Computer Vision*, Vol.9, No.2, pp.137-154, 1992.

[18] R. Tsai. “A Versatile Camera Calibration Technique for High- Accuracy 3D Machine Vision Metrology Using Off-the-Shelf Tv Cameras and Lenses”,*IEEE Journal of Robotics and Auto mation* RA-3, 4, pp.323-344, 1987.

[19] S.Vedula, P.W.Rander, H.Saito, and T.Kanade, “Modeling, Combining, and Rendering Dynamic Real-World Events From Image Sequences”, *Proc. 4th Conf. Virtual Systems and MultiMedia*, Vol.1, pp.326-332, 1998.

[20] T. Werner, R. D. Hersch and V. Hlavac, “Rendering Real-World Objects Using View Interpolation”, In IEEE International Conference on Computer Vision:ICCV95, pp.957-962, 1995.

[21] M.D. Wheeler, Y. Sato, and K. Ikeuchi, “Consensus surfaces for modeling 3D objects from multiple range images”, *DARPA Image Understanding Workshop*, 1997.