Object Extraction for Virtual-viewpoint Video Synthesis

Hiroshi Sankoh

Abstract

This thesis aims to realize virtual-viewpoint video synthesis, that is, aims to create an appearance of the objects in a certain viewpoint, in which real cameras do not exist, from multi-viewpoint video sequences capturing multiple dynamic objects such as humans in real world, and discusses a framework of object extraction to tackle occlusion regions caused by overlapping of multiple objects in every video sequence. Various algorithms to synthesize virtual-viewpoint images have been proposed, and they can be categorized by the types of object representations as follows: 3-dimensional (3D) object model, 2.5-dimensional (2.5D) depth map, and 2-dimensional (2D) billboard. Occlusion handling is an essential and challenging process to acquire all the object representations for multiple objects, and occluded regions have to be interpolated by extracting the texture of an occluded region from other camera at the same frame or from other frames of the same camera. In order to realize such interpolation process, silhouette extraction for object regions and occlusion detections in every frame of each video sequence are required to be appropriately done.

The main contribution of the thesis is to introduce a framework of object extraction methods based on temporal and/or spatial characteristics of the target scene to handle occlusion regions for virtual-viewpoint video synthesis, and propose an approach of object extraction for each object representation: 3D object model, 2.5D depth map, and 2D billboard. A framework of object extraction methods consists of the following three steps. The first one is extracting silhouettes for object regions based on the space characteristics of the coordinate system in which each object representation is formulated. A common approach to refine silhouettes by estimating background regions of the target scene is introduced for all the representations. The second one is detecting occlusion regions based on positional relationships of objects in a 3D space and motion estimation of objects between frames by taking ac-

count of the consistency between the extracted silhouettes and the object representation integrating multiple camera information. And the third one is acquiring visual texture of an object surface based on corresponding regions matching among multiple cameras and/or among consecutive frames by taking the characteristics of a virtual-viewpoint video rendering algorithm for each object representation into consideration. Furthermore, utilizing information for each object representation is organized as follows.

1. 3D object model: correspondence relationship between an arbitrary 3D coordinate in object-centered coordinate system and 2D pixel coordinate of every camera

2. 2.5D depth map: correspondence relationship between an arbitrary 3D coordinate in viewer-centered coordinate system and 2D pixel coordinate of the neighboring camera

3. 2D billboard: correspondence relationship between an arbitrary 2D pixel coordinate in every camera and 2D world coordinate of the specific plane in the target space

The thesis presents an approach of object extraction to solve occlusion problems for each object representation based on the above mentioned framework and utilizing information.

Acknowledgements

This work at Graduate School of Informatics, Kyoto University would not have been possible without grateful helps of many people.

I would like to thank my advisor Professor Michihiko Minoh for supervising this thesis. He taught me the fundamentals of good research, writing, and presentation. I wish to express my gratitude to him for reading the manuscript and making a number of helpful suggestions. I also wish to sincerely thank my thesis committee, Professor Takashi Matsuyama and Professor Katsumi Tanaka for their extensive comments and suggestions.

Thanks to former Associate Professor Masayuki Mukunoki, Associate Professor Masaaki Iiyama, and former Assistant Professor Takuya Funatomi in the laboratory for giving me many constructive suggestions for this thesis.

I appreciate Dr. Yasuyuki Nakajima, President and CEO of KDDI R&D Laboratories, Inc., for his generous support to this research. I would like to show the appreciation to Dr. Yutaka Yasuda, former Chairman of the Board of Directors of KDDI R&D Laboratories, Inc., for his continuous encouragement.

I also deeply appreciate Dr. Hiromasa Yanagihara, Exective Director of Multimedia Division of KDDI R&D Laboratories, Inc. for taking the opportunity of writing this thesis. I would like to express my thanks to Dr. Sei Naito, Group Leader of Ultra-Realistic Communications Laboratory of KDDI R&D Laboratories, Inc. for the continuous guidance since I entered the company. I am also very grateful to Mr. Kansuke Sasaki of Sure Designs for his aggressive support to develop practical software applications based on the proposed approaches of the thesis.

Last but not least, heartfelt thanks go to my family especially my wife Kyoko, my daughter Kureha, and my son Taiga for their understanding and support. I sincerely hope that my children would find inspiration in the thesis someday in the future.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This thesis discusses a framework of object extraction to tackle occlusion regions which is essential for virtual-viewpoint video synthesis.

Virtual-viewpoint video synthesis is defined as to create an appearance of the objects in a certain viewpoint, in which real cameras do not exist, from multiple camera videos capturing dynamic objects such as humans in real world. It is important to create virtual-viewpoint images that precisely reflect the lighting environment of input multiple camera images, as well as to realize a certain view of the object under another lighting condition. As described below, various algorithms have been proposed for virtual-viewpoint video synthesis, and tackling occlusion regions, which are caused by overlapping of multiple objects in every video sequence, is an essential process for realizing virtual-viewpoint video. Occluded regions have to be interpolated by extracting the visual textures from the corresponding regions in other cameras at the same frame or from the correspondent areas in other frames of the same camera.

Virtual-viewpoint image synthesis is important to realize ultra-realistic experiences such as Free-viewpoint Video and Tele-presence. Free-viewpoint Video is a next generation image media, in which audiences can see scenes from anywhere in 3-dimensional (3D) space, and expected to be applied to various digital contents production for real entertainment events such as sports and music concerts. In the media, dynamic scenes including moving objects like humans are captured with multiple cameras, and not only real camera images but also various appearances of the entire scene from arbitrary viewpoints should be reproduced. In this case, it does not necessarily require real-time processing but demands viewer-independent entire scene descrip-

tion. On the other hand, Tele-presence is an immersive video communication system, in which remote participants can feel as if they are sharing the same space with a sense of space continuity. In the system, each participant is captured with a sparsely arranged few cameras, and life-sized video presentation of the object appearance with a sense of space continuity is important to be reproduced. Furthermore, the experiences of eye-contact and motion parallax (natural and continuous appearance changes of the remote participant) dependent on head movement, which is very important for face-to-face communication in the real world, should be provided to users. In this case, the experiences described above should be realized in not arbitrary viewpoints but mainly in viewer-dependent frontal directions including human faces and bodies, and in addition, real-time processing is fully required.

Various algorithms to synthesize virtual-viewpoint images have been proposed, and they can be categorized into two types by the presence and absence of explicit modeling process of geometry information like 3D shape and photometry information such as reflection property of the objects. One is a model-based rendering (MBR) method [1] in which virtual-viewpoint videos are generated by modeling 3D shape and/or reflection property of the objects. The other is an image-based rendering (IBR) method [2][3] in which virtual-viewpoint images are synthesized without modeling process. Both IBR and MBR methods have advantages and disadvantages to synthesize virtual-viewpoint images, and an appropriate selection of the methods is important according to the requirements of target applications.

An MBR method basically describes shapes and spatial organizations of the objects in an object-centered coordinate as 3-dimensinal (3D) object model. The method extracts object regions and visual textures from multiple cameras and reconstructs a 3D shape of each object. Then, the method synthesizes virtual viewpoint images by projecting the 3D model and mapping the visual texture for each object [4][5]. Some MBR methods also estimate reflection property of the objects and models visual texture of the objects to create virtual-viewpoint video under arbitrary lighting environment. Reflection property of the object is basically estimated using the reconstructed 3D shape and the preliminarily estimated lighting environment. Although the accuracy of reconstructed 3D shapes has a large impact on the image quality of virtual-viewpoint images, MBR method does not have restrictions on virtual viewpoint positions in 3D space.

On the other hand, an IBR method basically synthesizes virtual-viewpoint images by switching or combining real multiple camera images, and it is not

necessary to explicitly obtain the geometric and photometric information of objects and scenes. One of the pioneering technology is Eye Vision [6], which was developed by Kanade et al. and was used in Super Bowl XXXV on January 2001, and Eye Vision controls more than 30 pan-tilt-zoom cameras so that an target object is focused and the object size in all the cameras should be same, and shows a key scene from various angles by not reconstructing a complicated shape of the object, but by simply switching camera viewpoints. Actually, Eye Vision had powerful impacts on audiences, even though it did not reproduce continuous appearance changes of the object. In order to smooth appearance changes of objects by switching camera images, some interpolation based methods have been proposed. Ray-space methods [10] [13] synthesizes natural photographed virtual-viewpoint images by cutting the ray-space. However, the rays which pass through the real space have to be densely sampled by multiple camera array systems, in which a huge number of cameras are densely arranged, and virtual-viewpoint positions are limited between real cameras. In ray-space methods, visible surface information in a viewer-centered coordinate system such as surface orientation and distance from camera viewpoints has to be reproduced as 2.5-dimensional (2.5D) depth map.

Furthermore, hybrid approach of an MBR method and an IBR method is proposed and applied to bullet time system which is not only switching real camera images but also extracting silhouettes of object regions from each camera image and representing the object as 2D billboard composed of a set of extracted silhouette images. 2D Billboard methods [7] [8] [9] do not reconstruct a shape but estimates the 3D world coordinate of an object position on the field plane, and represents each object as a set of 2D slice silhouettes and visual textures captured by camera viewpoints. 2D billboard can be overlaid in Computer Graphics (CG) space by adjusting the position and size of billboard, and natural and continuous appearance changes of CG by switching camera images can be realized. The bullet time system was used in the movie The Matrix. 2D Billboard representation does not reproduce continuous appearance changes of the object, but the representation can be easily acquired from at least single camera information, and also prevents visual artifacts of virtual-viewpoint images, since the texture itself extracted from a specific camera is mapped without blending process.

1.1 Object Representations for Virtual-viewpoint Video Synthesis

In the thesis, virtual-viewpoint video synthesis aims to create an appearance of not the entire scenes but the objects, therefore, virtual-viewpoint video synthesis methods can be classified by the types of object representations. As a representative type, 3D object model realizes natural and continuous appearance changes of the object from arbitrary viewpoints, and therefore the representation is suitable to realize Free-viewpoint Video. It is fully required that Free-viewpoint Video applications reproduce natural and continuous appearance changes of each object from viewer-independent arbitrary viewpoints. However, there are mainly two important conditions for 3D object model acquisition as follows; 1) the object surface is captured in precisely with a number of calibrated cameras located in 360-degrees circle, and 2) there exist limited occlusion regions caused by other objects. As a result, 3D object model is not necessarily suitable for Free-viewpoint Video targeting team sport events such as soccer and baseball in which multiple objects are moving in a wide area. In addition, 3D object model is not appropriate for Tele-presence in which multiple cameras are not surrounding the object but they are located in front of the object, and detailed human faces and bodies should be reproduced in viewer-dependent frontal directions.

On the other hand, 2.5D depth map only represents visible surface information such as surface orientation and distance from camera viewpoints in a viewer-centered coordinate system. Therefore, virtual-viewpoints cannot be reproduced from arbitrary viewpoints but a certain limited range of viewpoints. This 2.5D Depth map is suitable for realizing Telepresence, in which the experience of eye-contact and motion parallax should be realized in not arbitrary viewpoints but mainly in viewer-dependent frontal directions including human faces and bodies. For this case, natural continuous appearance changes of an object from only limited range of angles have to be reproduced, and a 2.5D Depth representation in viewer-centered coordinate system is appropriate for this category.

In order to realize Free-viewpoint Video targeting multiple objects such as team sports event held in a large space, 2D billboard representation is appropriate. Actually, the representation does not reproduce natural continuous appearance changes of each object but roughly reproduces 3D positional relationships among multiple objects from arbitrary viewpoints, which is es-

Table 1.1: Feature points of object representations regarding view range and appearance change.

Representation type	View range	Appearance change
3D object model	Arbitrary	Natural and continuous
2.5D depth map	Limited	Detailed and continuous
2D billboard	Arbitrary	Positional relationships

pecially useful to understand and enjoy team sports held in large space. In addition, the importance of natural and continuous appearance changes of the object becomes small in a large space.

On the basis of the discussions above, the feature points of object representations for virtual-viewpoint video synthesis regarding the viewing conditions: viewer independent (arbitrary) or viewer dependent (limited) and the smoothness of appearance changes are summarized as Table 1.1. Basically, 3D object model can be applied to realize both of Free-viewpoint Video and Tele-presence, and is especially suitable for Free-viewpoint Video targeting a single object. When multiple objects have to be captured at the same time, occlusions happens, and the 3D object model has some troubles in terms of image quality for virtual-viewpoint images. A 2D billboard representation does not reproduce natural continuous appearance changes of each object, but the representation is useful for Free-viewpoint Video when multiple objects are moving in a large space and 3D positional relationship among multiple objects is important as in the cases of team sports like soccer games. On the other hand, 2.5D depth map representation is suitable for Tele-presence in which detailed appearances of the target object should be reproduced from not arbitrary but limited viewpoints.

1.2 Object Extraction to Tackle Occlusion Regions for Virtual-viewpoint Video Synthesis

In this section, typical approaches to acquire three types of object representations for virtual-viewpoint video synthesis: 3D object model, 2.5D depth map, and 2D billboard are summarized, and the core issues for all the rep-

resentations are discussed.

A typical approach to get a 3D object model is shape from silhouettes algorithm [11]. Shape from silhouettes algorithm constructs a visual cone by combining an optical center and extracted silhouette region in each camera image with projection matrix, and acquires an intersection region called visual hull as an object shape in an object-centered coordinate. Even for texture-less objects, accurate visual hull can be acquired if silhouettes of object regions are precisely extracted in all the cameras. On the other hand, shape from silhouettes algorithm cannot reconstruct concave regions in principles, i.e., visual hull just circumscribes real shape of an object. Although, shape correction method such as space carving can be applied to visual hull, it is difficult to acquire accurate 3D geometry and photometry for occluded regions.

For getting 2.5D depth map, a representative method is stereo matching. Stereo matching measures a distance between an optical center and each pixel in every camera based on triangulation. For each pixel, corresponding point is searched on epipolar line in neighboring camera image, and detected based on the difference of pixel value. And then, the distance, that is, 3D coordinate in viewer-centered coordinate system is calculated based on the disparity of corresponding points, base line distance, and camera parameters. Although, there are many restrictions for shooting conditions, i.e. the distance and direction of a pair of cameras should be minimized, concave regions can be reconstructed if corresponding points can be correctly detected. However, detecting corresponding points is very difficult task, and therefore 2.5D Depth map cannot be precisely acquired. Recently, RGB-D cameras like Microsoft Kinect are wide spread, and such device is useful for the rough estimation of depth value for each pixel. But, the depth sensing performance of a general purpose of RGB-D camera is not sufficient for extracting object regions, especially in edge regions such as human hair. In addition, virtual-viewpoint image synthesized based on 2.5D depth map usually includes dis-occlusion areas whose textures and depth do not exist in a real camera.

In regard to 2D billboard, each object is represented as a set of 2D slice silhouettes and visual textures extracted from all the cameras, and the object position can be calculated in every frame based on the 2D coordinate of the bottom line of the silhouette region and homography matrix between captured frame and 2-dimensional world coordinate model of the target space. Homography matrix estimation and object extraction for moving cameras are two of the most important processes for 2D billboard. In addition, the

texture of an occluded object in a certain camera cannot be extracted precisely, since some parts of the texture region are not visible in the camera. In order to reconstruct a 2D billboard of an occluded object precisely, the object's texture has to be extracted from another frame of the same camera or another camera of the same frame in which the object is not occluded from the other objects.

As described above, tackling occlusion regions, which are caused by overlapping of multiple objects or some parts of the same object in every video sequence, is an essential process to acquire all the object representations for realizing virtual-viewpoint video. In occluded regions, the information about geometry and photometry of the objects cannot be precisely acquired, and such information has to be interpolated by extracting the visual textures from the corresponding regions in other cameras at the same frame or from the correspondent areas in other frames of the same camera. In order to realize such interpolation process, silhouette extraction for object regions and occlusion detections in every frame of each video sequence are required to be appropriately done.

On the basis of discussions above, the main contribution of the thesis is to introduce a framework of object extraction methods based on temporal and/or spatial characteristics of the target scene to handle occlusion regions for virtual-viewpoint video synthesis, and propose an approach of object extraction for each object representation: 3D object model, 2.5D depth map, and 2D billboard. A framework of object extraction methods consists of the following three steps. The first one is extracting silhouettes for object regions based on the space characteristics of the coordinate system in which each object representation is formulated. A common approach to refine silhouettes by estimating background regions of the target scene is introduced for all the representations. The second one is detecting occlusion regions based on positional relationships of objects in a 3D space and motion estimation of objects between frames by taking account of the consistency between the extracted silhouettes and the object representation integrating multiple camera information. And the third one is acquiring visual texture of an object surface based on corresponding regions matching among multiple cameras and/or among consecutive frames by taking the characteristics of a virtual-viewpoint video rendering algorithm for each object representation into consideration. Furthermore, utilizing information for each object representation is organized as follows.

1. 3D object model: correspondence relationship between an arbitrary 3D coordinate in object-centered coordinate system and 2D pixel coordinate of every camera

2. 2.5D depth map: correspondence relationship between an arbitrary 3D coordinate in viewer-centered coordinate system and 2D pixel coordinate of the neighboring camera

3. 2D billboard: correspondence relationship between an arbitrary 2D pixel coordinate in every camera and 2D world coordinate of the specific plane in the target space

The thesis presents an approach of object extraction to solve occlusion problems for each object representation based on the above mentioned framework and utilizing information.

1.3 Overview of the Thesis

This thesis is composed of five chapters.

In Chapter 1, the author presented the background of this thesis; a framework of object extraction to tackle occlusion regions for virtual-viewpoint video synthesis which is essential for virtual-viewpoint video synthesis.

In the following sections, the key issues for each research topic are briefly summarized, and the original algorithms to overcome those issues are described. And then, in Chapters 2, 3, and 4, the detail about the issues and the algorithm for each research topic is discussed respectively. Finally, Chapter 5 concludes the thesis and mentions about future works.

Chapter 2

Object Extraction for 3D Object Model

A typical approach to get a 3D object model is shape from silhouettes algorithm [11]. Shape from silhouettes algorithm constructs a visual cone by combining an optical center and extracted silhouette region in each camera image with projection matrix, and acquires an intersection region called visual hull as an object shape in an object-centered coordinate. Even for texture-less objects, accurate visual hull can be acquired if silhouette regions are precisely extracted in all the cameras. On the other hand, shape from silhouettes algorithm cannot reconstruct concave regions in principles, i.e., visual hull just circumscribes real shape of an object. Although, shape correction method such as space carving can be applied to visual hull, it is difficult to acquire accurate 3D geometry and photometry for occluded regions.

The chapter presents an approach of object extraction to tackle occlusion problems for 3D object model based on the above mentioned framework and the utilizing information, that is, correspondence relationship between an arbitrary 3D coordinate in object-centered coordinate system and 2D pixel coordinate of every camera. Our method uses an approach for integrating multi-view images in which the background region is determined using voxel information rather than each camera image itself. We introduce a likelihood of background to each pixel of camera images, and derive integrated likelihood in the voxel space. The background region is determined on the basis of minimization of energy functions of the likelihood. Furthermore, the proposed method also applies a robust refinement process for virtual-viewpoint images based on the minimization of difference between synthesized image

and captured image for each viewpoint. Experimental results show the proposed method to be more effective than the existing methods.

2.1 Introduction

A free viewpoint video provides a new visual experience, in which audiences can see scenes from anywhere in 3D space [14]. In the free viewpoint video system, the virtual viewpoint can be moved such as back-and-forth and around as well as up-and-down among objects in a field where cameras cannot be mounted. It gives audiences an immersive feeling, and we call these view-changing experiences "walk-through" and "fly-through".

There are two main categories for generating a free viewpoint video. One is a model-based method [1] and another is an image-based method [15]. In order to realize the visual experiences mentioned above, the former method is more suitable than the latter since the former does not have restrictions on virtual viewpoint positions in 3D space, if a 3D model can be appropriately reconstructed [1]. It should be noted that 3D model accuracy has a large impact on the video quality, and the purpose of our study is to reconstruct the 3D model with high accuracy.

A typical method for acquiring a 3D model of interesting objects (e.g., humans) is a shape from silhouette [11] which reconstructs a visual hull in a 3D voxel space using silhouettes of objects extracted from camera images. Therefore, an accurate silhouette is necessary to generate a high-quality 3D model. There have been the numerous related works on the silhouette extraction. Most of them classify each pixel of a camera image into background region or foreground region using only visual information from a single viewpoint[16][17]. There is a substantial difficulty of the works in the case the foreground area and the background area have the same color, and it is highly probable that the extracted silhouette image includes both of unwanted regions and missed parts which correspond to false positives and false negatives caused by misclassification in extraction process, respectively.

In this chapter, we propose a robust background subtraction method using multi-view images, instead of using only a single camera image. The method applies the likelihood of background to each pixel of a camera image, and derives the integrated likelihood of each voxel in a voxel space, which is considered to be integrated information of multi-view images. The background region is determined based on the likelihood of voxel space with

local adaptation to minimize its energy functions. Furthermore, the proposed method also applies a robust refining process, in which each silhouette is improved based on projections of the 3D model to each viewpoint.

The rest of the paper is organized as follows. Section 2.2 overviews related works, and Section 2.3 details our robust background subtraction method based on 3D model projections with likelihood. Section 2.5 presents experimental results and comparison with conventional methods. Finally, the paper is concluded in Section 2.6.

2.2 Related Works

There has been some research to improve a shape from silhouette method regarding suppression of false positives and false negatives in silhouette extraction. In order to reduce the number of false negatives, papers [18][19] proposed the method to relax a condition for the foreground determination. These methods count the number of viewpoints in which a voxel is projected outside the silhouette, and identifies the voxel included in the foreground object when the number is below the threshold. Especially, the paper [19] calculates the likelihood of misclassification for each voxel based on the ratio of false positives and false negatives in each silhouette image, and decides the threshold used in the paper [18] mentioned above.

However, these methods do not have a removal process for false positives. Therefore, the silhouette extraction process shall exclude unwanted regions precisely. Additionally, the threshold for reducing false negatives is not continuous but a discrete value based on the accepted number of viewpoints outside the silhouette. It might cause false positives in resultant visual hull, since the threshold cannot be set sensitively.

Another approach to refine silhouettes and a visual hull is introducing intersection and projection consistency into 3D space and multi-view images, respectively[20]. This method refines each silhouette and visual hull by cross-referencing both of them. Furthermore, the input data of this method is a rough visual hull that includes all objects. Thus it does not need background subtraction to be applied, and is independent of the accuracy of silhouette extraction. However, in order to remove unwanted regions from each silhouette, this method only uses the edge information of camera images acquired by region segmentation. Therefore, the accuracy of visual hull and silhouette images strongly depends on the performance of region segmentation. In

particular, it is very difficult to extract foreground objects with sufficient precision when similar color features are found in both the foreground and the neighboring background. Consequently, the removal process of unwanted regions does not work properly, and the final visual hull and silhouettes might include false positives and false negatives. Additionally, the refinement process for dealing with false negatives is not considered, and the false negatives remain in the final result.

There is another issue with a shape from silhouette method itself, which may include inaccurate concave surfaces. That is, the reconstructed visual hull is just a convex polyhedron in which the real object is inscribed. Some solutions are proposed in the paper [21], and we do not pursue this issue in this paper.

2.3 Robust Background Subtraction Method

To overcome the problems mentioned in Section 2.2, we propose a robust background subtraction method considering the likelihood of background in each viewpoint instead of extracting binary silhouettes. We derive integrated likelihood, and the background region is determined in the voxel space. Furthermore, we refine visual hull and corresponding silhouette images based on geometric information in 3D space so that robustness for both false negatives and edge features can be improved.

The proposed method has a series of procedures as shown in Figure 2.1, and is summarized into two stages. These are the determination process for the visual hull based on the likelihood and the refinement process reflecting geometric information in voxel space. The proposed scheme takes multi-view synchronized images as input and provides a reconstructed visual hull as output.

2.3.1 Introduction of Likelihood as Background Object

Conventional shape from silhouette methods need a binary silhouette image for every viewpoint; but it is highly probable that the binary silhouette images include false positives and false negatives. Such problems often arise if images include objects whose pixel color values are close to those of the background region. In order to overcome this problem, we analyse the pixel-wise

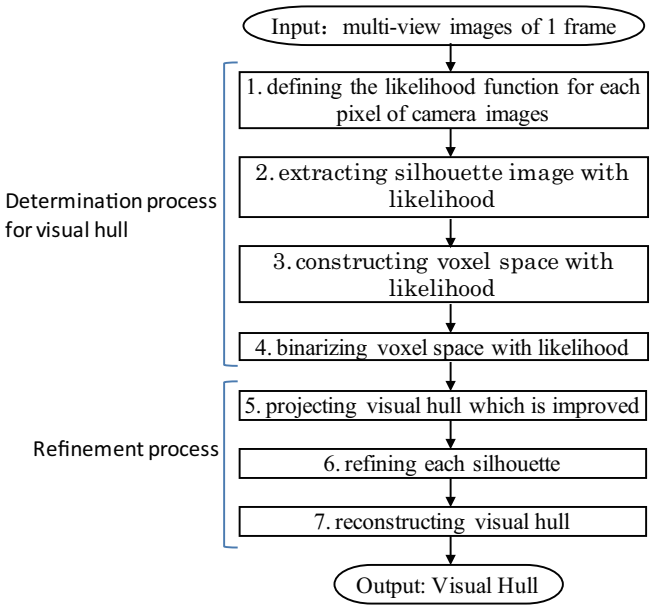


Figure 2.1: Flow chart of proposed method.

likelihood as a background object for input camera images instead of employing the binary segmentation process. Furthermore, we derive integrated likelihood in a voxel space.

First of all, it is assumed that camera images for a certain length of time without any foreground objects are provided in every camera position. We represent each pixel value as multi-dimensional vector $\boldsymbol{x}$ in a particular color space. On the assumption that pixel values are approximated by normal distribution, the likelihood function as a background object is defined as $f(\boldsymbol{x}; \boldsymbol{u}, \boldsymbol{\Sigma})$ by the following equation. In the equation, $\boldsymbol{u}$ and $\boldsymbol{\Sigma}$ are an average vector and a covariance matrix of pixel value $\boldsymbol{x}$ for some frames, respectively.

$$f(\boldsymbol{x}; \boldsymbol{u}, \boldsymbol{\Sigma}) = \exp(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{u})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{u})) \tag{2.1}$$

We acquire a silhouette image with likelihood in each viewpoint based on the function. Then, each voxel v_i in the voxel space is projected to viewpoint n ($n = 0, \ldots, N-1$), and voxel likelihood ρ_i is calculated based on likelihood $f(v_i^{(n)})$ of projected pixel value $v_i^{(n)}$. Here, we construct the voxel

space with likelihood by calculating the average value of $f(v_i^{(n)})$ to achieve a representative feature of all viewpoints using the equation (2.2).

$$\rho_i = \frac{1}{N} \sum_{n=0}^{N-1} f(v_i^{(n)}) \tag{2.2}$$

Conventional works proposed the extended method, which reduces false negatives by ignoring some viewpoints for which voxel v is projected outside the silhouette [18][19]. Compared with these works, our proposed method is able to control false negatives not discretely such as the number of viewpoints but continuously based on the likelihood.

2.3.2 Binarization of Voxel Space with Likelihood

The simplest way to binarize voxel space with likelihood is to employ the thresholding of a single voxel. However, voxels whose likelihood is close to the threshold might be misclassified. To deal with such a problem, we define the energy function considering the adjacency relationship in a 3D space, and binarize the voxel space with likelihood by minimizing the energy function using a graph-cut algorithm.

Energy function $E(\boldsymbol{\alpha}_v)$ is defined by equation (2.3) where $\boldsymbol{\alpha}_v = (\alpha_{v_1}, \ldots, \alpha_{v_i}, \ldots)$ identifies whether each voxel v_i belongs to the foreground or the background region.

$$E(\boldsymbol{\alpha}_v) = U(\boldsymbol{\alpha}_v) + V(\boldsymbol{\alpha}_v) \tag{2.3}$$

$U(\boldsymbol{\alpha}_v)$ is a data term that depends on only likelihood ρ_i of voxel v_i, and gives the energy value as shown in the equations (2.4) and (2.5). Here, b_v and th_ρ are positive constants.

$$U(\boldsymbol{\alpha}_v) = \sum_i U(\alpha_{v_i}) \tag{2.4}$$

$$U(\alpha_{v_i}) = \begin{cases} -\log(\rho_i) & (\alpha_{v_i} = 0) \\ \max[th_\rho + \log(\rho_i), 0] & (\alpha_{v_i} = 1) \end{cases} \tag{2.5}$$

$V(\boldsymbol{\alpha}_v)$ is a smoothing term featuring the difference between ρ_i and ρ_j of a pair of adjacent voxels v_i and v_j. It gives the energy value related to α_{v_i} and α_{v_j} by the equations (2.6) and (2.7) where N_v indicates all the combinations of a pair of adjacent voxels v_i and v_j. In the equation (2.7), $dis_v(\cdot)$ is the Eucidean distance of adjacent voxels, λ_v and κ_v are positive constants, and κ_v is calculated with the expectation operation $< \cdot >$ related to N_v as an equation (2.8).

$$V(\boldsymbol{\alpha}_v) = \sum_{(i,j)\in N_v} V(\alpha_{v_i}, \alpha_{v_j}) \tag{2.6}$$

$$V(\alpha_{v_i}, \alpha_{v_j}) = \begin{cases} \frac{\lambda_v \exp(-\kappa_v(\rho_i - \rho_j)^2)}{dis_v(i,j)} & (\alpha_{v_i} \neq \alpha_{v_j}) \\ 0 & (\alpha_{v_i} = \alpha_{v_j}) \end{cases} \tag{2.7}$$

$$\kappa_v = (2 < (\rho_i - \rho_j)^2 >)^{-1} \tag{2.8}$$

The value of data term U decreases in proportion to the likelihood of a voxel that is classified as foreground. In addition, the value of smoothing term V decreases in proportion to the difference between likelihood ρ_i and rho_j of adjacent voxels beyond the region boundary. The minimization of this energy function is known to be solved based on the graph-cut algorithm [16]. In the proposed scheme, the binarization process for the voxel space is conducted in a similar way while assigning label 0 and 1 to the background and the foreground, respectively.

2.3.3 Refinement based on 3D Model Projections

Removal of unwanted regions for the visual hull

We introduce a removal process of unwanted regions which are misclassified as foreground. Since a voxel size of a real object is usually larger than that of unwanted regions, the voxel size is used to distinguish them. The proposed process eliminates small regions whose voxel size is not ranked in the top R-order of all objects. Prior to rank the regions, the visual hull is divided into the closed regions.

Removal of shadow regions for silhouette images

Shadow regions cannot be extracted as closed regions since the shadow is connected to the foreground regions. Therefore, the removal process mentioned above does not work well. However, it is possible to eliminate shadow regions based on the constraint that they exist on the floor in 3D space. At first, we represent each pixel value of viewpoints as vector $\boldsymbol{I}(p)$ in an appropriate color space, and calculate the difference between foreground image $\boldsymbol{I}_f(p)$ and base image $\boldsymbol{I}_b(p)$ captured without any objects. The pixel that satisfies the condition of equation (2.9) is regarded as an unwanted region candidate. In the equation, I_d indicates the threshold parameter.

$$|\boldsymbol{I}_f - \boldsymbol{I}_b| < I_d \tag{2.9}$$

Considering that the difference of chroma signals between foreground images and base images is small in shadow regions, the color space UV is effective for detecting shadows.

Then, we calculate the intersection point where the light rays of each pixel and the visual hull cross in 3D space. When there is an intersection point whose height from the floor is nearly equal to 0, the pixel can be eliminated as a shadow region.

For a viewpoint n, the projection matrix $\boldsymbol{P}_n$ is defined by 2D pixel coordinate $\boldsymbol{m}_{n,p} = (u_{n,p}, v_{n,p})$, 3D world coordinate $\boldsymbol{M} = (X, Y, Z)$ and scholar s as follows. Here, $\tilde{\boldsymbol{m}}_{n,p} = (u_{n,p}, v_{n,p}, 1)$ and $\tilde{\boldsymbol{M}} = (X, Y, Z, 1)$ are homogeneous coordinates.

$$\boldsymbol{P}_n = \begin{pmatrix} P_{n11} & P_{n12} & P_{n13} & P_{n14} \\ P_{n21} & P_{n22} & P_{n23} & P_{n24} \\ P_{n31} & P_{n32} & P_{n33} & P_{n34} \end{pmatrix} \tag{2.10}$$

$$s\tilde{\boldsymbol{m}}_{n,p}^T = \boldsymbol{P}_n \tilde{\boldsymbol{M}}^T \tag{2.11}$$

The direction of light ray $\boldsymbol{r}_{n,p} = (x_{n,p}, y_{n,p}, z_{n,p})$ of each pixel $\boldsymbol{m}_{n,p} = (u_{n,p}, v_{n,p})$ is defined by sub matrix $\boldsymbol{P}_n^{(f)}$ of the projection matrix $\boldsymbol{P}_n$ as equations (2.12) and (2.13).

$$\boldsymbol{P}_n^{(f)} = \begin{pmatrix} P_{n11} & P_{n12} & P_{n13} \\ P_{n21} & P_{n22} & P_{n23} \\ P_{n31} & P_{n32} & P_{n33} \end{pmatrix} \tag{2.12}$$

$$\boldsymbol{r}_{n,p}^{T} = \frac{(\boldsymbol{P}_n^{(f)})^{-1}\tilde{\boldsymbol{m}}_{n,p}^{T}}{\left|(\boldsymbol{P}_n^{(f)})^{-1}\tilde{\boldsymbol{m}}_{n,p}^{T}\right|} \tag{2.13}$$

Now, therefore, an equation of the light ray of a pixel $\boldsymbol{m}_{n,p}$ in a viewpoint n is expressed by a formula (2.14) with the gradient $\boldsymbol{r}_{n,p}$ and the camera position $M_n = (X_n, Y_n, Z_n)$.

$$(X, Y, Z) = \boldsymbol{M}_n + t\boldsymbol{r}_{n,p} \tag{2.14}$$

For each pixel in an unwanted region candidate, we calculate the intersection point of the visual hull and the corresponding light rays as $\boldsymbol{V}_{n,p} = (X_{V_{n,p}}, Y_{V_{n,p}}, Z_{V_{n,p}})$, and when the condition as shown in equation (2.15) is satisfied, the pixel is eliminated. In the equation, Y_d indicates the threshold parameter which stands for the height in 3D space.

$$Y_{V_{n,p}} < Y_d \tag{2.15}$$

Removal of unwanted regions for silhouette images

We introduce the removal process for unwanted regions that neighbor the contour of the foreground object. Each silhouette image is binarized by minimizing the energy function with graph-cut algorithm, and the pixels regarded as background after minimization are removed. Energy function $E(\boldsymbol{\alpha}_p)$ is defined by equation (2.16) where $\boldsymbol{\alpha}_p = (\alpha_{p_1}, \ldots, \alpha_{p_i}, \ldots)$ identifies whether each pixel p_i belongs to the foreground or the background region.

$$E(\boldsymbol{\alpha}_p) = U(\boldsymbol{\alpha}_p) + V(\boldsymbol{\alpha}_p) \tag{2.16}$$

$U(\boldsymbol{\alpha}_p)$ is a data term that depends on only likelihood $f(\boldsymbol{x}_i)$ of pixel p_i, and gives the energy value as shown in the equations (2.17) and (2.18). Here, b_p is a positive constant.

$$U(\boldsymbol{\alpha}_p) = \sum_i U(\alpha_{p_i}) \tag{2.17}$$

$$U(\alpha_{p_i}) = \begin{cases} -\log(f(\boldsymbol{x}_i)) & (\alpha_{p_i} = 0) \\ -\log(1 - f(\boldsymbol{x}_i)) & (\alpha_{p_i} = 1) \end{cases} \tag{2.18}$$

$V(\boldsymbol{\alpha}_p)$ is a smoothing term featuring the difference between pixel values $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ of a pair of adjacent pixels p_i and p_j. It gives the energy value related to α_{p_i} and α_{p_j} by the equations (2.19) and (2.20) where N_p indicates all the combinations of a pair of adjacent pixels p_i and p_j. In the equation (2.20), $dis_p(\cdot)$ is the Eucidean distance of adjacent pixels, λ_p and κ_p are positive constants, and κ_p is calculated with the expectation operation $< \cdot >$ related to N_p as an equation (2.21).

$$V(\boldsymbol{\alpha}_p) = \sum_{(i,j) \in N_p} V(\alpha_{p_i}, \alpha_{p_j}) \tag{2.19}$$

$$V(\alpha_{p_i}, \alpha_{p_j}) = \begin{cases} \frac{\lambda_p \exp(-\kappa_p (\boldsymbol{x}_i - \boldsymbol{x}_j)^2)}{dis_p(i,j)} & (\alpha_{p_i} \neq \alpha_{p_j}) \\ 0 & (\alpha_{p_i} = \alpha_{p_j}) \end{cases} \tag{2.20}$$

$$\kappa_p = (2 < (\boldsymbol{x}_i - \boldsymbol{x}_j)^2 >)^{-1} \tag{2.21}$$

2.4 Virtual-viewpoint Image Synthesis

In order to synthesize virtual-viewpoint images, visual hull is converted into surface polygon models by applying marching cubes method [22], and the texture of each polygon is acquired by blending the textures extracted from visible multiple cameras with weighting coefficients. As described earlier, shape from silhouettes algorithm cannot reconstruct concave regions in principle, that is, visual hull circumscribes real shape of an object, and therefore, shape correction method such as space carving should be applied to visual hull. Most of the shape correction methods are based on color consistency between multiple cameras and silhouette constraint for each camera, but these methods do not take account the process of rendering virtual-viewpoint images. In this section, the refinement framework of virtual-viewpoint images based on the minimization of difference between synthesized image and captured image for each viewpoint is introduced.

For each pixel p_m^n of a viewpoint n, the intersection point v_m^n of the visual hull and the corresponding light ray is calculated using a formula (2.14). Visible cameras for v_m^n are decided by projecting the 3D coordinate of v_m^n into all the remaining cameras using projection matrix P_n and calculating the intersection point v_m^{cam} of the visual hull and the corresponding light ray in each camera. If the distance between v_m^n and v_m^{cam} is under a certain threshold, the camera cam is considered as visible. The color of p_m^n is acquired by blending the textures extracted from all the visible cameras with weighting coefficients, which are decided according to distance ratios between the voxel and camera positions.

Then the difference image between the synthesized image and original image for object regions in viewpoint n is generated. The regions in which there are great gaps are detected by minimizing energy function for the difference image as an equation (2.16) except that the data term is formulated by not the likelihood but the pixel value of the difference image. In addition, silhouette constraint for each viewpoint, that is, the projection of the visual hull fits closely with the silhouette in each viewpoint, is introduced. For pixels existing on edge regions (sil) of the silhouette in each difference image, hard constraint for energy minimization is installed as follows.

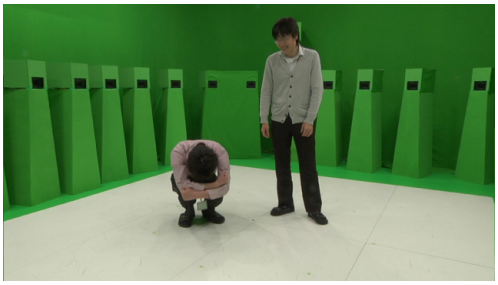
$$U(p_i \in sil, \alpha_{p_i}) = \begin{cases} \infty & (\alpha_{p_i} = 0) \\ 0 & (\alpha_{p_i} = 1) \end{cases} \tag{2.22}$$

The visual hull is refined by detecting the regions in which there are great gaps between synthesized image and original image for each pixel in every viewpoint, and increase or decrease voxels on the visual hull surface along the corresponding light ray in such a way that PSNR between synthesized image and original image is maximized. The refinement process is applied for each pixel in every viewpoint, and each voxel of the visual hull is assigned the flag representing increase or decrease. Finally, each voxel is deleted or added by majority vote of all the viewpoints.

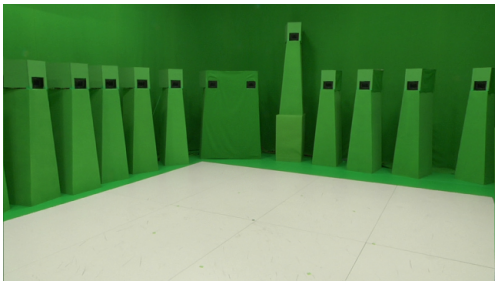
2.5 Experimental Results

In order to evaluate the effectiveness of the proposed method, we conducted three experiments for multi-view video sequences. In experiment 1, we evaluated the reconstruction accuracy of visual hull compared with conventional

works. In experiment 2, we analysed the contributions of each process in the proposed method. Finally, in experiment 3, we generated the free viewpoint video using the visual hull reconstructed by the proposed method. In these evaluations, we used multi-view images which were captured in a 360-degree circle with the spatial resolution of 640×360, and temporally aligned frames in each viewpoint. We prepared three kinds of sequences. Sequence A was captured with 30-cameras in green-back studio, sequence B was captured with 30 cameras in general studio, and Sequence C was captured with 11-cameras in general inddor room. Figures 2.2, 2.3, and 2.4 show examples of foreground and background images.



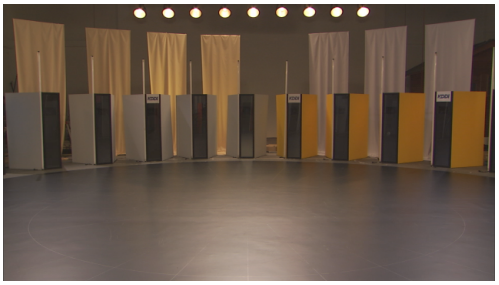
(a) A foreground image



(b) A base image

Figure 2.2: Sequence A

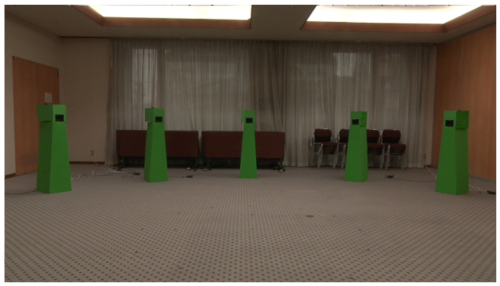
(a) A foreground image



(b) A base image

Figure 2.3: Sequence B

(a) A foreground image



(b) A base image

Figure 2.4: Sequence C

2.5.1 Experiment 1: Reconstruction Accuracy of the Visual Hull

In order to assess the accuracy of the visual hull reconstructed by the proposed method, we conducted an experiment for sequences A, B, and C. The resolution of voxel space was set $2cm^3$ for each sequence, the number of voxels for Sequences A and C were $160 \times 100 \times 100$, and those for Sequence B were $256 \times 128 \times 256$ in $x - y - z$ coordinate system, respectively. The likelihood function of each pixel was defined according to the equation (2.1), and base images of 60-frames in each viewpoint ware used to calculate the likelihood function. The unwanted region removal process for silhouette images was not applied to sequences A and B, and the shadow removal process was not applied to sequence C. Therefore, in equation (2.1) each pixel value is represented as 3-demensional vector in RGB color space in sequences A and B, while it was represented as 2-demensional vector in UV space in sequence C.

We evaluated the accuracy of the finally reconstructed visual hull by comparing the projections of the visual hull and the ground-truth images. The quantitative performance was assessed as follows. First, we prepare ground-truth images which were manually segmented, and then compared them pixel-wise with the projections of the visual hull. Finally, three values Recall, Precision, and F-measure ware calculated by equations (2.23), (2.24), and (2.25) as in the case of related work based on true positives, false positives and false negatives defined as follows.

True Positive (TP)

The pixel correctly extracted as object

True Negative (TN)
The pixel correctly eliminated as background

False Positive (FP)
The pixel incorrectly extracted as object

False Negative (FN)
The pixel incorrectly eliminated as background

$$\text{Recall} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FN}} \tag{2.23}$$

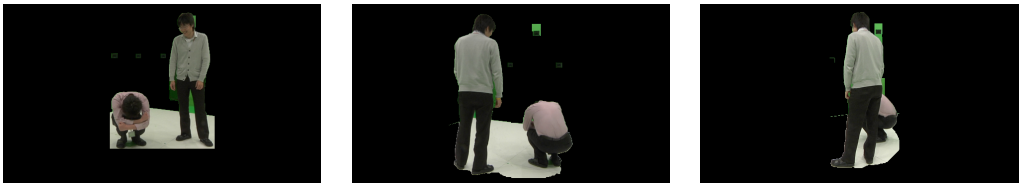
$$\text{Precision} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FP}} \tag{2.24}$$

$$\text{F} - \text{measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{2.25}$$

As conventional schemes, the GrabCut method [17] which uses only single image information, and Zeng's method [20] which uses the information of multi-view images, were also evaluated for comparison. Regarding the GrabCut method, we measured pixel-wise differences between the ground-truth and extracted results in each viewpoint. With regard to Zeng's method, we used projections of the visual hull to evaluate with the same criteria as used in the proposed method.

(a) Viewpoint 00 (b) Viewpoint 15 (c) Viewpoint 20

Figure 2.5: Results of proposed method (Sequence A).



(a) Viewpoint 00 (b) Viewpoint 15 (c) Viewpoint 20

Figure 2.6: Results of GrabCut method (Sequence A).

(a) Viewpoint 00 (b) Viewpoint 15 (c) Viewpoint 20

Figure 2.7: Results of Zeng's method (Sequence A).

For Sequences A and B, the projection images of the visual hull reconstructed by the proposed method are shown in Figures 2.5 and 2.9. The resultant images are shown with textures to help understanding. The results of the GrabCut method are shown in Figures 2.6 and 2.10, and those of Zeng's method are shown in Figures 2.7 and 2.11, respectively. In addition, Ground truth silhouettes manually segmentated are shown in Figures 2.8 and 2.12. As shown in these results, it is obvious that the projection image of the visual hull generated by the proposed method almost corresponds to the object region in each viewpoint, except that there are a little false negatives neighboring the contour of the arms. Such false negatives are assumed to be caused by the estimated error of each projection matrix. The estimated error of the projection matrix directly affects the accuracy, since our proposed method employs projections between voxel space and each viewpoint. On the other hand, the GrabCut and Zeng's method resulted in a lot of false negatives and false positives. The accuracy of Zeng's method depends on the results of region segmentations that are applied first, and the method does not work well especially for objects with similar textures to background.

Additionally, for Sequences A and B, an extracted silhouette image in every camera viewpoint produced by each method was evaluated quantita-

(a) Viewpoint 00 (b) Viewpoint 15 (c) Viewpoint 20

Figure 2.8: Ground truth silhouettes manually segmentated (Sequence A).

(a) Viewpoint 09 (b) Viewpoint 15 (c) Viewpoint 25

Figure 2.9: Results of proposed method (Sequence B).

tively. The comparisons of the quantitative performances in every viewpoint among the proposed method, GrabCut method and Zeng's method are shown in Figures 2.13 and 2.14, respectively. Furthermore, the average scores calculated by the number of cameras for each method are summarized in Tables 2.1 and 2.2. The difference in Recall values among the three methods is not large since all the methods are controlled to avoid false negatives in the early stage of processing to the extent possible. However, the difference in Precision values is significant, which proves that the proposed scheme is more effective than conventional methods.

Table 2.1: Comparison of reconstruction accuracy among methods (Sequence A).

	Recall	Precision	F-measure
Proposed	0.979	0.956	0.967
GrabCut	0.993	0.759	0.858
Zeng	0.985	0.789	0.874

(a) Viewpoint 09 (b) Viewpoint 15 (c) Viewpoint 25

Figure 2.10: Results of GrabCut method (Sequence B).

(a) Viewpoint 09 (b) Viewpoint 15 (c) Viewpoint 25

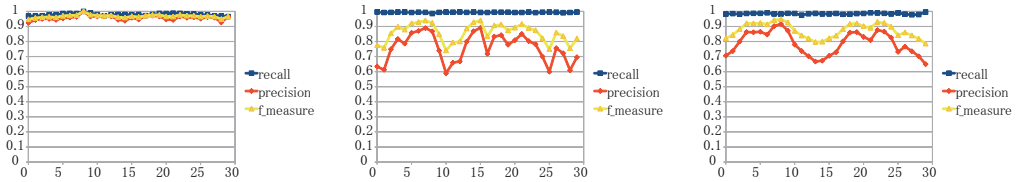
Figure 2.11: Results of Zeng's method (Sequence B).

2.5.2 Experiment 2: Analyzation of Contributions for Refinement Processes

The proposed scheme is assumed to be comprised by two key functions that are the binarization process of the voxel space with likelihood and the refinement process considering the 3D geometry. In order to clarify the contribution of the respective function, we analysed results of each process for proposed method. An example of projection images of the visual hull reconstructed by binarizing the voxel space is shown in Figure 2.15 (a). This result corresponds to the performance without the refinement process. We can confirm that the reconstruction accuracy is comparatively high, though there remain a little false positives in the floor. On the other hand, Figure 2.15 (b) shows the result when refinement process was additionally employed. The difference between Figure 2.15 (a) and Figure 2.15 (b) shows the improvement by refinement process of visual hull. For the specific region in Figure 2.15 (b), the close-up image is shown in Figure 2.15 (c). It shows that there remain shadow regions near objects of legs. We applied shadow removal process to all viewpoints, and reconstructed the visual hull using improved silhouettes. A projection image of the improved visual hull is shown

(a) Viewpoint 09 (b) Viewpoint 15 (c) Viewpoint 25

Figure 2.12: Ground truth silhouettes manually segmentated (Sequence B).



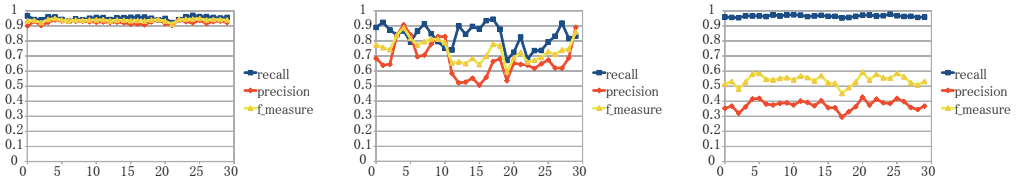
(a) Proposed method (b) GrabCut method (c) Zeng's method

Figure 2.13: Evaluation values in each camera viewpoint (Sequence A).

in Figure 2.15 (d). The difference between Figure 2.15 (c) and Figure 2.15 (d) indicates the improvement by shadow removal process.

To evaluate the removal process for unwanted regions for silhouette images, we analysed the results for sequences B. An example of projection images of the visual hull reconstructed by binarizing the voxel space is shown in Figure 2.16 (a). It is confirmed that false positives are remaining near object contours. The removal process was then applied for every viewpoint, and the visual hull was reconstructed using improved silhouettes. A projection image of the improved visual hull to the corresponding viewpoint is shown in Figure 2.16 (b). For the specific region in Figure 2.16 (a) and Figure 2.16 (b), close-up images are also shown in Figure 2.16 (c) and Figure 2.16 (d), respectively. Figure 2.16 shows that unwanted regions on the floor as well as neighboring contours have been removed appropriately.

Furthermore, we quantitatively evaluated the projection images for both the non-improved visual hull and the improved visual hull in a similar way as in experiment 1, and the results are shown in Table 2.3. The improvement of Precision values shows that false positives were successfully suppressed, and the flatness of Recall values suggests robustness for false negatives.



(a) Proposed method (b) GrabCut method (c) Zeng's method

Figure 2.14: Evaluation values in each camera viewpoint (Sequence B).

Table 2.2: Comparison of reconstruction accuracy among methods (Sequence B).

	Recall	Precision	F-measure
Proposed	0.951	0.924	0.937
GrabCut	0.831	0.674	0.738
Zeng	0.965	0.376	0.541

2.5.3 Experiment 3: Generation of Virtual-viewpoint Images

Virtual-viewpoint images for sequence C was synthesized by the proposed method. In order to evaluate the refinement process of the proposed method, comparative method 1 was implemented based on space carving method [21], and comparative method 2 was implemented based on graph-cut method [22].

A pair images of 3D model and generated virtual viewpoint image acquired by the proposed method is shown in Figure 2.17. Results of comparative method 1, comparative method 2, and original visual hull are shown in Figure 2.18, Figure 2.19, and Figure 2.20 respectively. It was confirmed that the concave region caused by human arms overlap is refined smoothly by the proposed method. The results of comparative method 1 showed that some parts of concave regions were appropriately refined, but some edge regions such as foot and back of the human are incorrectly deleted. From the results of comparative method 2, some parts of concave regions remained as it was, but virtual-viewpoint image synthesized by the refined 3D model did not have the artifacts like green regions.

In order to evaluate the quantitative performances for the proposed method

(a) Projection image of the non-improved visual hull on viewpoint 05

(b) Projection image of the improved visual hull on viewpoint 05

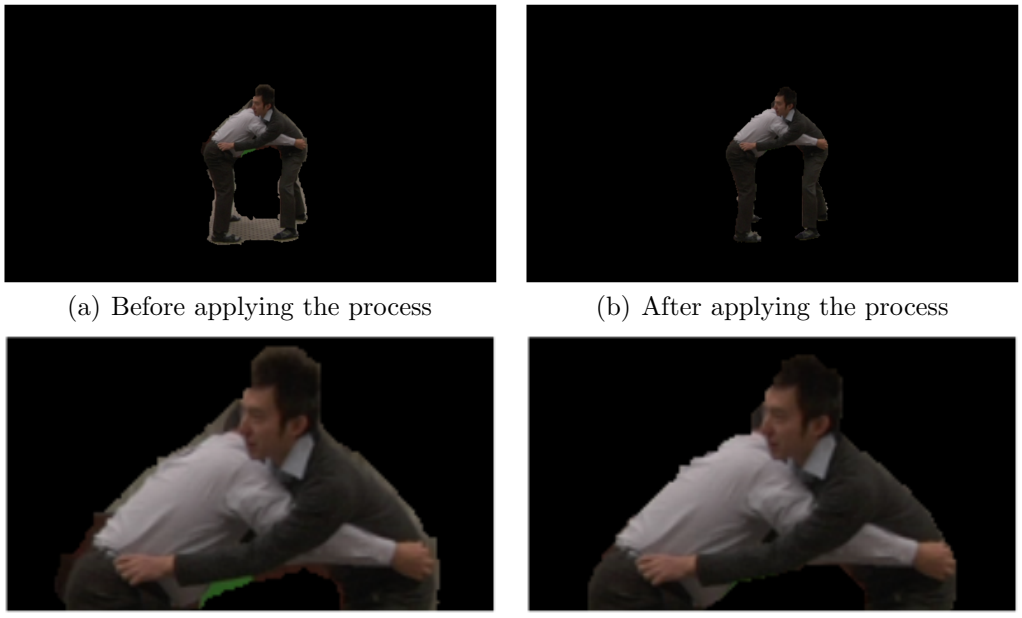
(c) Close-up image of a specific region in (b)

(d) Projection image of the visual hull whose shadow regions were removed on viewpoint 05

Figure 2.15: Results of the refinement process.

and comparative methods, virtual-viewpoint images for each real camera viewpoint was synthesized using 3D model acquired from the remaining 10 camera images, and PSNR for each viewpoint was calculated for each method. The average scores of PSNR for a removal viewpoint, the remaining viewpoints, and all the viewpoints were summarized as Table 2.4. It was confirmed that the scores of PSNR for the proposed method were highest in all the viewpoints, and especially, the score of PSNR for removal viewpoint showed that the refinement process was effective for other viewpoints.

Finally, the scores of PSNR for 30 frames were calculated for each method, and the graph including the average, minimum, and maximum scores were summarized as Figure 2.21 and Table 2.5. The results showed that the proposed method was robust for multiple frames, and improved the scores of PSNR compared with comparative methods.



(a) Before applying the process

(b) After applying the process

(c) Before applying the process (close-up of a specific region)

(d) After applying the process (close-up of a specific region)

Figure 2.16: Results of the removal process for unwanted regions near contours.

2.6 Conclusion

To realize highly precise 3D model reconstruction, we proposed a robust background subtraction method using the integrated information of multi-view images. As an inherent problem of the conventional schemes for 3D model reconstruction, the precision of the visual hull is highly dependent on the background subtraction result for the specific viewpoint. In order to overcome this problem, the proposed scheme employs two main features. One is determination of the background region based on the likelihood in a voxel space, and the other is the refinement of both visual hull and projection images considering 3D space geometry as well as visual information. From experimental results using actual multi-view images, it was confirmed that both key features greatly contributed to significant improvement compared with the conventional methods. Furthermore, it was also confirmed that the virtual viewpoint images were generated precisely while the occluded regions

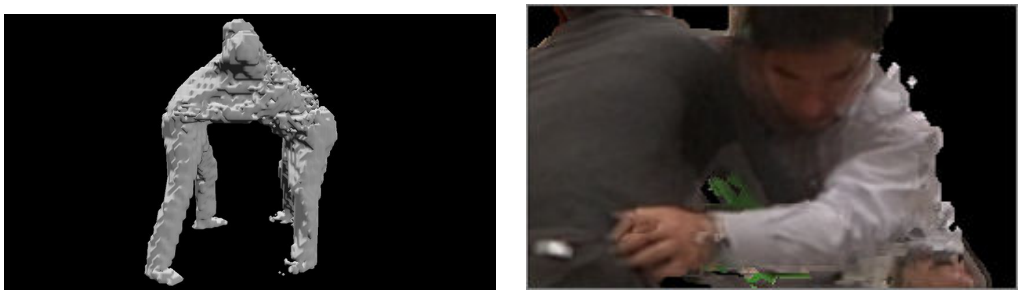
Table 2.3: Effectiveness of modification process.

	Recall	Precision	F-measure
Shadow removal OFF	0.994	0.574	0.717
Shadow removal ON	0.994	0.640	0.770
Refinement OFF	0.994	0.884	0.935
Refinement ON	0.993	0.901	0.945

(a) 3D model (b) Virtual-viewpoint image

Figure 2.17: Results of the proposed method.

were reconstructed successfully. As future works, we need to introduce a process that reduces the influence of estimated error of projection matrixes.



(a) 3D model (b) Virtual-viewpoint image

Figure 2.18: Results of the comparative method 1.

(a) 3D model (b) Virtual-viewpoint image

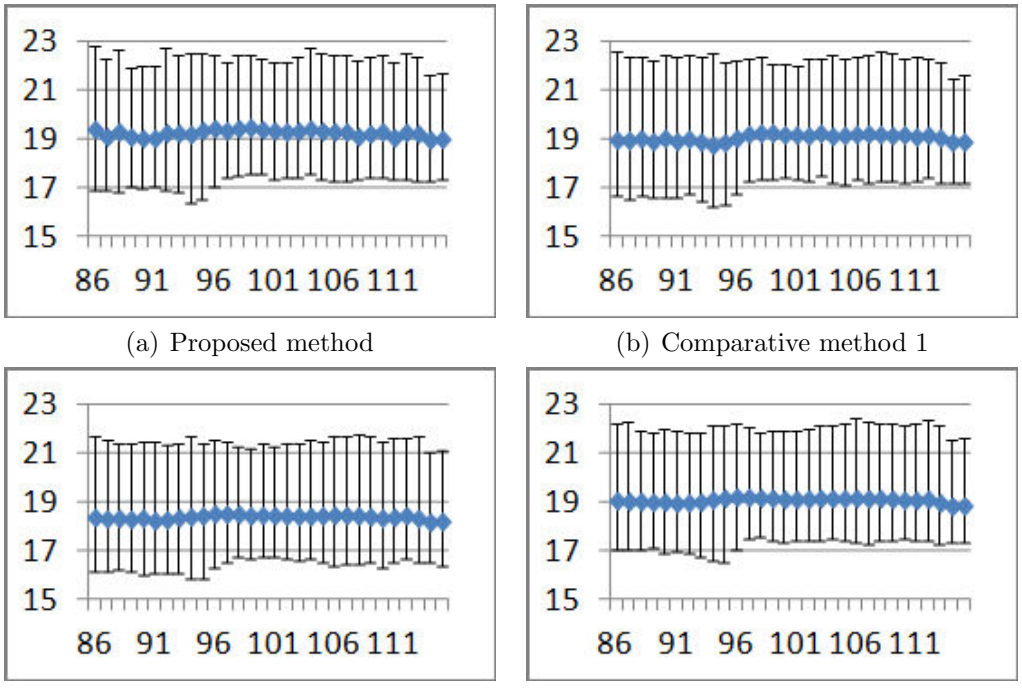
Figure 2.19: Results of the comparative method 2.

(a) 3D model (b) Virtual-viewpoint image

Figure 2.20: Results of the original visual hull.

Table 2.4: Comparisons of PSNR for each method.

	Rmoval	Remaining	All
Proposed method	20.060	20.174	20.333
Comparative method 1	19.610	19.534	19.538
Comparative method 2	19.242	18.733	18.743
Visual Hull	19.969	19.837	19.955



(a) Proposed method (b) Comparative method 1

(c) Comparative method 2 (d) Visual hull

Figure 2.21: Comparisons of Quantitative results for multiple frames.

Table 2.5: Comparisons of PSNR for multiple frames.

	Average	Minimum	Maximum
Proposed method	19.224	17.199	22.324
Comparative method 1	19.042	17.023	22.274
Comparative method 2	18.365	16.399	21.490
Visual Hull	19.051	17.224	22.066

Chapter 3

Object Extraction for 2.5D Depth Map

For 2.5D Depth representation, an image quality in edge regions such as hair may degrade. Furthermore, the virtual viewpoint includes dis-occlusion areas whose textures do not exist in real camera. In this chapter, we propose an an approach of object extraction to solve occlusion problems for 2.5D depth map based on the above mentioned framework and the utilizing information, that is, correspondence relationship between an arbitrary 3D coordinate in viewer-centered coordinate system and 2D pixel coordinate of the neighboring camera. Our proposed method is based on interpolation and extrapolation of depth information in edge regions. In addition, a virtual view synthesis method is also proposed based on tracking 3D regions between consecutive frames. Experimental results showed the effectiveness of the proposed method regarding the image quality of virtual viewpoints. Furthermore, it was confirmed that the experience of depth perception, eye contact, and motion parallax for head movement could be naturally realized.

3.1 Introduction

Immersive video conference systems can reproduce a distant conferee with advanced audio-visual technologies as if he/she is in the same conference room. It allows global companies to reduce the transportation cost for face-to-face meetings. To achieve such an immersive experience at a distance, the current commercial systems such as Cisco's TelePresence [43], HP's HALO,

and Polycom's TPX require building the physically same conference room; that is, the remote sites need to construct the room with exactly the same appearance including the same furnishings such as the conference table, chairs and the wallpaper. Such configurations require considerable cost and thus users are discouraged from introducing the system. In addition, natural view-changing based on one's head movements (motion parallax) is not realized in these systems, and therefore, users do not have the perception that they have had face-to-face communications.

In terms of advanced high-fidelity video conference systems, many related researches have been conducted in this decade. For example, the European FP7 3DPresence project aimed to build a multi-view and multi-user 3D videoconferencing system. In the project, some research activities that cut out the attendees from the real scene and virtually synthesized them into the background of another 3D space were reported [44][45][46]. In other cases, the gaze-corrected (eye contact) view generation method [47][48] and motion parallax realization method [49] were proposed. The papers include several 3D processing techniques such as image acquisition, preprocessing, disparity estimation, view synthesis and image display. The major challenge of these activities was the generation of high quality depth maps or reconstruction of accurate 3D models of human regions using a number of cameras. The main approach of depth map generation is disparity estimation techniques based on stereo block matching [50], while the main approach by 3D model reconstruction is volumetric reconstruction techniques based on the shape from silhouette algorithms [51]. For the stereo block matching method, the depth estimation quality fully depends on camera intervals, and long camera intervals would introduce artifacts. In contrast, the shape from silhouette algorithms could stably reconstruct high quality 3D models using multiple cameras with long intervals, but a number of cameras arranged in a 360-degree view are necessary for 3D model reconstruction.

Our motivation for this study is to realize an immersive video conference system for a general home living space as well as office meeting rooms, based on only a few sparsely arranged "color plus depth" (RGB-D) cameras without dedicated equipment. In the system, the region of an attendee in each conference site is extracted accurately from multi-view video sequences, and remote users can feel as if they are sharing the same space. Then, the segmented texture from a remote camera is naturally synthesized on the display of a local site while maintaining space continuity as illustrated in Figure 3.1. Furthermore, in this paper, we provide the experience of eye-contact

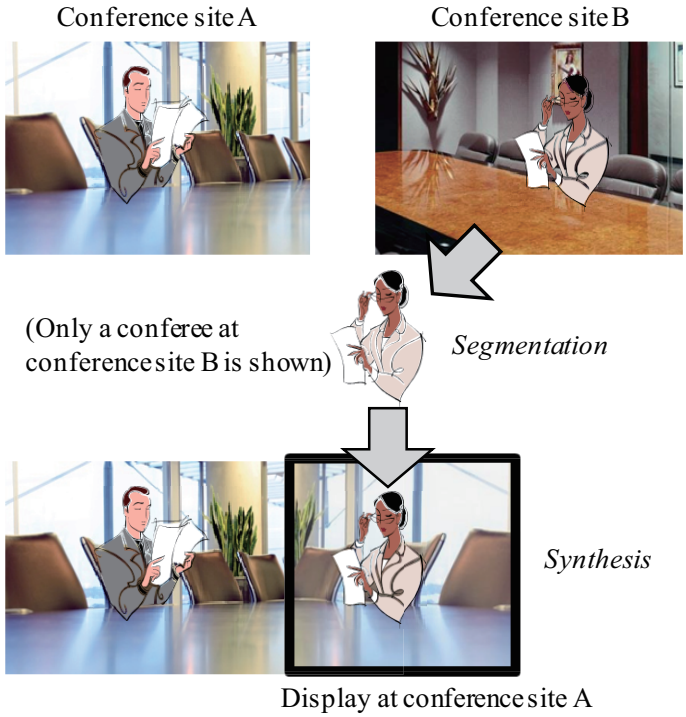


Figure 3.1: Concept of our immersive video conference system.

and motion parallax (view-changing) dependent on head movement, which is very important for face-to-face communication in the real world. In order to provide these experiences in a video conference system, a virtual viewpoint image has to be accurately synthesized by reconstructing a 3D-model of each attendee in real-time, and this is a challenging task as described above.

In this paper, we propose an accurate 3D-model reconstruction method based on interpolating depth information especially in edge regions. In addition, a virtual view synthesis method is also proposed based on tracking 3D regions between consecutive frames. The system targets a home living space with a large screen (as shown in Figure 3.2) as well as regular meeting/office rooms, and enables users to feel as if attendees at a distance are in the same room with a sense of space continuity. The proposed method targets an environment where the cameras are sparsely located in front of the objects and the number is less than or equal to 3. The camera configuration and an example of background images are shown in Figure 3.3.

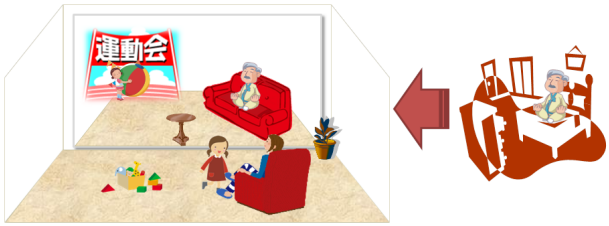


Figure 3.2: A use case for home living with a large screen environment.

(a) camera configuration (b) example of background images

Figure 3.3: Shooting environment of the proposed method.

The remainder of this chapter is organized as follows. Section 3.2 briefly describes the state-of-the-art related works on immersive video conference systems. Section 3.3 introduces the proposed immersive video conference system. Section 3.4 presents experimental results and a comparison with the conventional methods. Finally, the paper is concluded in Section 3.5.

3.2 Related Works

In this section, we focus on the related works based on RGB-D cameras. More recently, state-of-the-art real time immersive video conference systems using RGB-D cameras, such as the Microsoft Kinect [52] have been proposed [53][54]. The paper [53] proposed an eye contact system using a depth camera. Specifically, the system has used several depth map preprocessing techniques: foreground / background separation, joint bilateral filtering, and

discontinuity-adaptive filtering, and finally verified that the system realized eye contact. However, the foreground / background separation process is based on a simple threshold of depth value, and the extraction accuracy in the edge region is insufficient. In addition, the dis-occlusion inpainting algorithm is based on simple hole-filling referring to neighboring regions in the same frame, thus, complicated occluded texture cannot be reconstructed correctly. In contrast, the paper [54] also uses RGB-D cameras, specifically multiple calibrated cameras and infrared dot patterns, and proposed a motion parallax reproduction system. The system reconstructs high-quality 3D models for each user with the help of infrared (IR) dot patterns, and embeds these 3D models into a common virtual environment, even under a real-time process. Furthermore, the system works under transmission of all the color video and information on dot patterns via the IP network. However, an IR mask that represents the foreground is extracted using a simple threshold, and the quality of a specific edge region such as hair and thin parts is insufficient. The system captures only the upper side of the body above the shoulder, and does not consider dis-occluded regions such as the body part behind arms.

The RGB-D camera is useful for the rough extraction of human regions based on depth information. However, the depth sensing performance of a general-purpose RGB-D camera is not high, and the depth measurement does not guarantee sufficient precision. Therefore, the original depth signal captured with an RGB-D camera cannot directly be applied to an immersive video conference system. Actually, there are many research activities related to depth de-noising [55][56], but most of them take much more time compared to real time, and in addition, they are not intended for segmentation / 3D modelling of human regions. To extract human regions, there are some foreground segmentation approaches targeting general spaces whose background has complicated textures. However, most of these schemes employ the process based on edge detection or the simple threshold of each pixel using only a single camera [57][58]. Therefore, they lack robustness for color similarities between the foreground and background. In addition, some research studies exist using multi-view cameras for robust foreground segmentation [59][60]. In a previous work [60], we proposed a background subtraction method using multi-view images, instead of using only a single camera image. The previous method targeted a shooting environment as in a studio where the background is simple, and the luminance change is controlled. Furthermore, more than 10 cameras were set in a 360-degree view. The shooting envi-

ronment for the proposed method in this paper (as shown in Figure 3.3) is completely different from that of the previous method. Therefore, if the previous method were directly applied to the shooting condition of the proposed method, there would have been a concern that segmentation accuracy would become so poor that the natural synthesis necessary for a telepresence system could not be realized.

Regarding dis-occlusion inpainting algorithms, there also exist many related works targeting real time processing [61][62][63]. However, most of them assume that dis-occlusion belongs to the background region, and do not consider the case of self occlusion among human regions. Therefore, conventional methods are inapplicable to virtual viewpoint synthesis based on 3D modelling for an immersive video conference system.

The key issues of related works are summarized as the following two problems. The first is that the accuracy of human segmentation especially for edge regions such as hair is poor due to the lack of depth sensing performance. And the other problem is that the image quality of the synthesized viewpoint according to the remote user's head movement is inadequate, since the image includes visual artefact in edge regions and dis-occluded regions such as body parts covered by arms.

3.3 Segmentation and Virtual-viewpoint Image Synthesis

To overcome the key problems mentioned in Section 3.2, we propose an accurate 3D-model reconstruction method based on interpolating depth information in edge regions. In addition, a virtual view synthesis method is also proposed based on tracking 3D regions between consecutive frames. The proposed method consists of a series of procedures as shown in Figure 3.4, which takes a few RGB-D images and outputs a virtual viewpoint image according to the user's head position captured by the depth sensor. The procedures of the proposed method are summarized into two stages: the first one is foreground segmentation in the RGB image to refine depth information in edge regions, the second stage is virtual view synthesis based on 3D-model reconstruction.

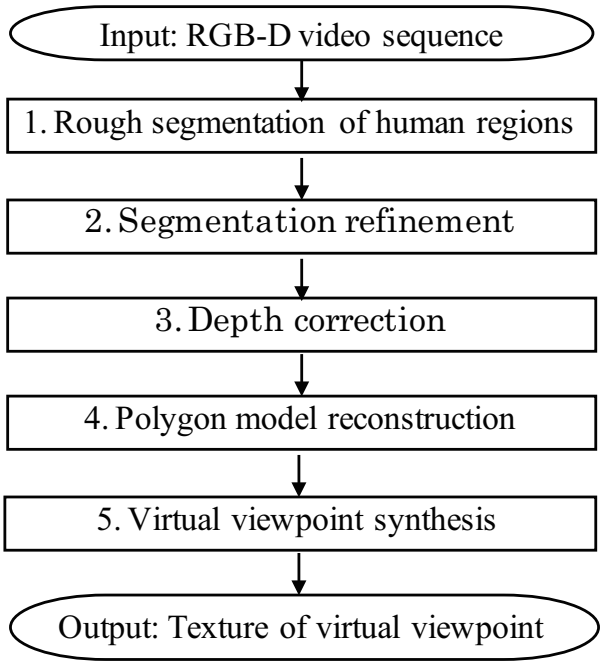


Figure 3.4: Flowchart of the proposed method.

3.3.1 Human Region Segmentation

The proposed method does not determine the background or foreground for each pixel in every single camera but estimates the human regions based on both the object existence probability for an individual camera and color similarities among multiple cameras. In order to calculate the object existence probability for an individual camera, background modeling for each camera pixel is conducted based on the assumption that the pixel value is approximated by the normal probability distribution. For the specific pixel (x, y) of camera cam, the object existence probability $\rho^{(c)}_{cam}(x, y)$ for each color component c in RGB or YUV color space is defined by equation (3.1). Here $I^{(c)}_{cam}(x, y)$ is a component of a multidimensional vector $\boldsymbol{I}_{cam}(x, y)$ in a certain color space and represents an 8-bit value. In addition, $\mu^{(c)}_{cam}(x, y)$ and $\sigma^{(c)}_{cam}(x, y)$ are average and standard deviations of $I^{(c)}_{cam}(x, y)$ for a certain number of consecutive frames in background sequences captured without

foreground objects, respectively.

$$\rho^{(c)}_{cam}(x,y) = 1 - \exp(-\frac{(I^{(c)}_{cam}(x,y) - \mu^{(c)}_{cam}(x,y))^2}{2(\sigma^{(c)}_{cam}(x,y))^2}) \tag{3.1}$$

The foreground region can be roughly segmented based on the simple threshold for equation (3.1), but when using only single camera information, they lack robustness for color similarities between the foreground and background. Therefore, color similarity between multiple cameras is introduced in the proposed method. Color similarity between the pixel (x, y) of camera cam and the pixel (x', y') of camera cam' is measured by equation (2) based on zero-mean normalized cross correlation (ZNCC).

$$R^N(cam^{(x,y)}, cam'^{(x',y')}) = \frac{\sum_{j=0}^{N-1}\sum_{i=0}^{N-1}(\boldsymbol{I}^N_{cam^{(x,y)}}(i,j) - \bar{\boldsymbol{I}}^N_{cam^{(x,y)}})(\boldsymbol{I}^N_{cam'^{(x',y')}}(i,j) - \bar{\boldsymbol{I}}^N_{cam'^{(x',y')}})}{\sqrt{\sum_{j=0}^{N-1}\sum_{i=0}^{N-1}(\boldsymbol{I}^N_{cam^{(x,y)}}(i,j) - \bar{\boldsymbol{I}}^N_{cam^{(x,y)}})^2 \times \sum_{j=0}^{N-1}\sum_{i=0}^{N-1}(\boldsymbol{I}^N_{cam'^{(x',y')}}(i,j) - \bar{\boldsymbol{I}}^N_{cam'^{(x',y')}})^2}} \tag{3.2}$$

In the equation, $R^N(cam^{(x,y)}, cam'^{(x',y')})$ is the matching score calculated over the $N \times N$ windows surrounding the pixel (x, y) of cam and (x', y') of cam'. $\boldsymbol{I}^N_{cam^{(x,y)}}(i,j)$ is a vector in a certain color space, which can be calculated by equation (3.3), and $\bar{\boldsymbol{I}}^N_{cam^{(x,y)}}$ represents the average of $\boldsymbol{I}^N_{cam^{(x,y)}}(i,j)$ in the $N \times N$ window.

$$\boldsymbol{I}^N_{cam^{(x,y)}}(i,j) = \boldsymbol{I}_{cam}(x - \frac{(N-1)}{2} + i)(y - \frac{(N-1)}{2} + j) \tag{3.3}$$

In order to segment foreground objects from multi-view sequences, an individual 3D voxel space, in which each voxel has likelihood, is assigned to every camera. Each voxel is projected into every camera using the camera parameter (projection matrix) estimated prior to the shooting. The relationship between the 3D world coordinates (X, Y, Z) of voxel v and the 2D pixel coordinate (x, y) of the projected pixel $v^{(cam)}$ in camera cam can be represented as equation (3.4) using the projection matrix P_{cam} and scholar s.

$$s(x, y, 1)^T = \boldsymbol{P}_{cam}(X, Y, Z, 1)^T \tag{3.4}$$

Therefore, the likelihood of voxel v is calculated by referring to the corresponding projected pixel (x, y) of camera cam, which is represented as $v^{(cam)} = (x, y)$.

Our previous method [60] calculated the likelihood of each voxel by simply averaging the object existence probability for all the cameras and did not consider the color similarity among multiple cameras. When there exists a number of cameras in a 360-degree view, the previous method worked well, but the accuracy decreased with the decreasing number of cameras. Therefore, in order to improve the robustness for sparsely arranged camera configurations as shown in Figure 3.3, two types of likelihoods are set for each voxel v of the individual 3D voxel space assigned to the camera cam. The first is the object existence probability calculated based on the individual camera as shown in equation (3.5).

$$\rho_{1st}^{(c)}(v^{(cam)}) = \rho_{cam}^{(c)}(x, y) \tag{3.5}$$

The second is the color similarity among multiple cameras defined by equation (6) considering both matching scores between adjacent cameras and object existence probabilities. Here, $cam - 1$ and $cam + 1$ represent the immediate left and the immediate right of a camera cam, respectively. For a far left camera, the first term of the right side of equation (3.6) is calculated as zero, while for a far right camera, the second term is set as zero.

$$\begin{aligned}&\rho_{2nd}^{(c)}(v^{(cam)}) = \\ &R^N(v^{(cam)}, v^{(cam-1)})\rho_{1st}^{(c)}(v^{(cam-1)}) + R^N(v^{(cam)}, v^{(cam+1)})\rho_{1st}^{(c)}(v^{(cam+1)})\end{aligned} \tag{3.6}$$

In order to roughly segment foreground human regions from the individual voxel space, two-staged processes are defined. In the first stage, a simple determination process based on the threshold for each color component is conducted by equation (3.7) where $th_{1st}^{(c)}$ indicates the threshold. If at least one color component satisfies equation (3.7), the voxel is labeled as the foreground region. An example of a roughly segmented human region is shown in Figure 3.5.

$$\rho_{1st}^{(c)}(v^{(cam)}) > th_{1st}^{(c)} \tag{3.7}$$

(a) Original RGB image (b) Rough segmentation

Figure 3.5: Rough segmentation of human region.

Then, as the second stage, energy function E considering the results of the first stage as a hard constraint and 26 $(3 \times 3 \times 3 - 1)$ adjacent voxels in 3D space is defined by equation (3.8) based on the Markov random field model. Here λ, U, and V indicate the weighting parameter, the data term, and the smoothing term, respectively.

$$E(\boldsymbol{\alpha}) = \sum_{k} U(\alpha_k) + \lambda \sum_{(k,l)\in N_v} V(\alpha_k, \alpha_l) \tag{3.8}$$

In the equation, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_k, \ldots)$ indicates binary labels identifying whether each voxel v_k belongs to the foreground $(\alpha_k = 1)$ or the background $(\alpha_k = 0)$ region, and N_v indicates available candidates for the adjacent voxels for (k, l). The data term U depends on only the second likelihood $\rho^{(c)}_{2nd}(v^{(cam)}_k)$, and gives the energy value as shown in equation (3.9) with the threshold $th^{(c)}_{2nd}$.

$$U(\alpha_k) = \begin{cases} \min_c[\rho^{(c)}_{2nd}(v^{(cam)}_k)] & (\alpha_k = 0) \\ \min_c[\max[th^{(c)}_{2nd} - \rho^{(c)}_{2nd}(v^{(cam)}_k), 0]] & (\alpha_k = 1) \end{cases} \tag{3.9}$$

The smoothing term V is defined as the difference between the second likelihoods $\rho^{(c)}_{2nd}(v^{(cam)}_k)$ and $\rho^{(c)}_{2nd}(v^{(cam)}_l)$ of a pair of adjacent voxels v_k and v_l as shown in equation (3.10) based on Gibbs distribution model. In equation

(3.10), $dis(k,l)$ is the Euclidean distance of v_k and v_l in the 3D voxel space, and κ is the positive constant.

$$V(\alpha_k, \alpha_l) = \begin{cases} \frac{\exp(-\kappa \sum_c (\rho_{2nd}^{(c)}(v_k^{(cam)}) - \rho_{2nd}^{(c)}(v_k^{(cam)}))^2)}{dis(i,j)} & (\alpha_k \neq \alpha_l) \\ 0 & (\alpha_k = \alpha_l) \end{cases} \tag{3.10}$$

Here, for the voxel that satisfies condition (3.7), the data term is calculated by equation (3.11) for the hard constraints.

$$U(\alpha_k) = \begin{cases} \infty & (\alpha_k = 0) \\ 0 & (\alpha_k = 1) \end{cases} \tag{3.11}$$

By minimizing the energy function E in equation (3.8) using the graph-cut algorithm, the foreground region in the 3D voxel space for every camera can be extracted. The foreground texture of every camera is extracted by projecting the foreground region into every camera viewpoint. The extracted texture is shown in Figure 3.6.

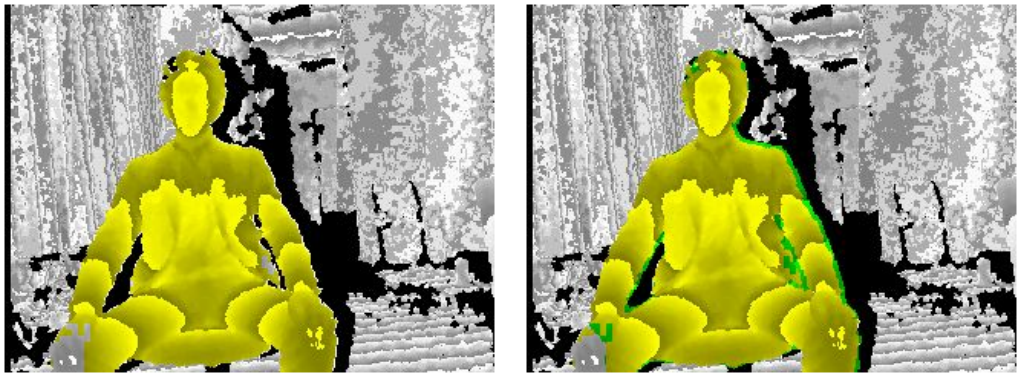
(a) Rough segmentation (b) Accurate segmentation

Figure 3.6: Segmentation refinement.

3.3.2 Virtual-viewpoint Image Synthesis

In order to identify an accurate human region in each depth image, the extracted human texture in each RGB image is projected into the corresponding

depth image based on the camera parameter between the RGB camera and depth sensor. In the depth image, the difference region between the accurate and the rough human areas is identified as shown in Figure 3.7 (green pixel represents the difference). Then, the missing depth value is interpolated by averaging the value in the pixels of surrounding human regions.



(a) Original depth (b) Refined depth

Figure 3.7: Depth correction.

In Step 4, the 3D polygon model for each depth image is reconstructed based on triangulation for the neighboring 4 pixels in the corrected depth image. Two examples of rendering results of the reconstructed polygon model are shown in Figure 3.8.

In Step 5, the virtual viewpoint detected by a user's head tracking is synthesized based on the 3D polygon model with texture. The dis-occluded area is identified dependent on the virtual viewpoint, and the corresponding texture in the previous frame is estimated by tracking the corresponding polygons between consecutive frames as shown in Figure 3.9.

3.4 Experimental Results

In order to evaluate the effectiveness of the proposed method, we conducted two experiments for multi-view video sequences. In the first experiment, our proposed method was applied to two video sequences to evaluate the segmentation accuracy of human regions. In the second experiment, in order

(a) example 1 (b) example 2

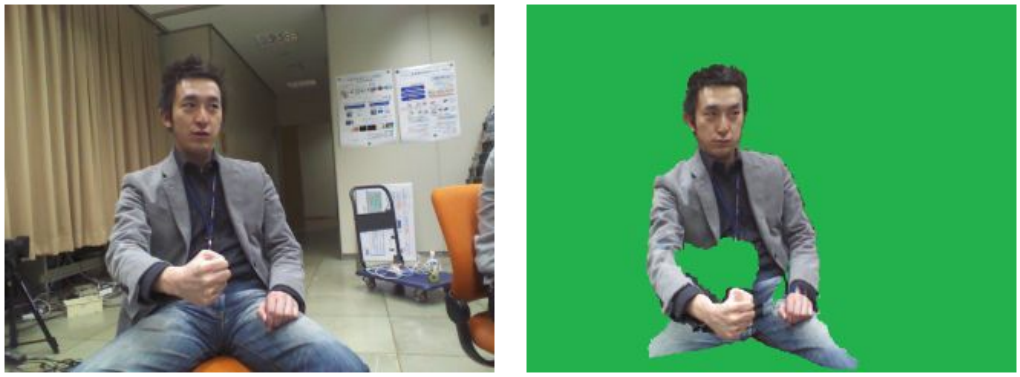
Figure 3.8: Polygon model reconstruction.

to confirm the experience of motion parallax functionality, view synthesis was conducted for specific virtual viewpoints based on the proposed method.

3.4.1 Experiment 1: Comparisons of Segmentation Accuracy

In the experiment, our proposed method was applied to two video sequences captured in an environment where the background texture was complicated and similar to the foreground texture, and includes luminance variations. Then, the segmentation accuracy was compared with three conventional methods: simple threshold segmentation method using only single image information (Single-view method), our conventional method [60] based on the information from multi-view images (Multi-view method), and a depth-based approach (Kinect). In order to compare the accuracy with a depth-based approach, the scene was captured with three Kinect devices arranged at 50-cm and 30-degree angle intervals and at distances of 80-cm from the object as shown in Figure 3.10. The spatial resolution of the sequences was 1280 $\times$ 960, and the frames in each viewpoint were temporally synchronized. Figures 11 and 12 show the test images (Seq. A and Seq. B), which include background images captured without any foreground objects, and these two types of sequences are intended for evaluating the robustness according to the differences of human appearances such as hair style and the color of clothing.

A voxel space was set for every camera, the resolution was set to 1 cm^3,



(a) Original RGB image (b) Virtual-view image

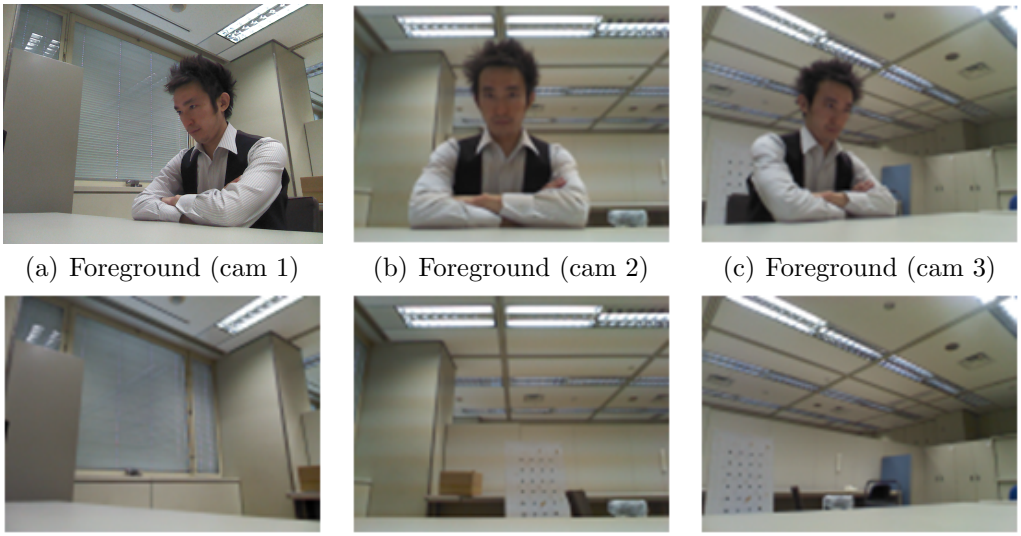
Figure 3.9: Dis-occlusion detection.

(a) Camera configuration (b) Example of the scene

Figure 3.10: Experimental environment.

and the number of voxels was set to $100 \times 100 \times 100$ in the $x-y-z$ coordinate system. Here, each pixel value was represented as YUV color spaces, and 300-frames of background images were used for the background modeling. The thresholds $th_{1st}^{(c)}$ and $th_{2nd}^{(c)}$ were selected considering the result of a preliminary experiment so that the ratio of false positives and false negatives to the ground truth were minimized.

Regarding the single-view method, the object existence probability $\rho_{cam}^{(c)}(x, y)$ was calculated by equation (3.1), and the segmentation was conducted based on equation (3.7). With regard to the multi-view method, a common voxel space was set for all cameras, and the likelihood of each voxel was calculated by averaging the first likelihood $\rho_{1st}^{(c)}(v^{(cam)})$ among cameras.



(a) Foreground (cam 1) (b) Foreground (cam 2) (c) Foreground (cam 3)

(d) Background (cam 1) (e) Background (cam 2) (f) Background (cam 3)

Figure 3.11: Test sequence (Seq. A).

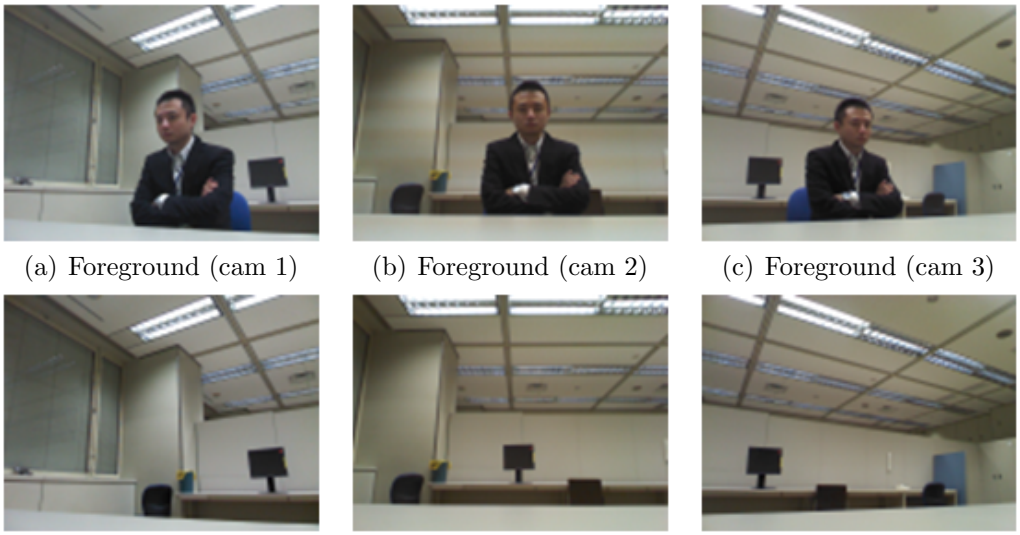
In order to evaluate quantitative performances, the ground truth of the foreground image for every camera was prepared by manual segmentation as shown in Figure 3.13. Finally, three values of Recall, Precision and F-measure were calculated based on the pixel number of true positives, false positives and false negatives as calculated by equation (3.12), (3.13) and (3.14), as in the case of the experiments in Section 2.5 based on true positives, false positives and false negatives defined as follows.

True Positive (TP)
The pixel correctly extracted as object

True Negative (TN)
The pixel correctly eliminated as background

False Positive (FP)
The pixel incorrectly extracted as object

False Negative (FN)
The pixel incorrectly eliminated as background



(a) Foreground (cam 1) (b) Foreground (cam 2) (c) Foreground (cam 3)

(d) Background (cam 1) (e) Background (cam 2) (f) Background (cam 3)

Figure 3.12: Test sequence (Seq. B).

.

$$\text{Recall} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FN}} \tag{3.12}$$

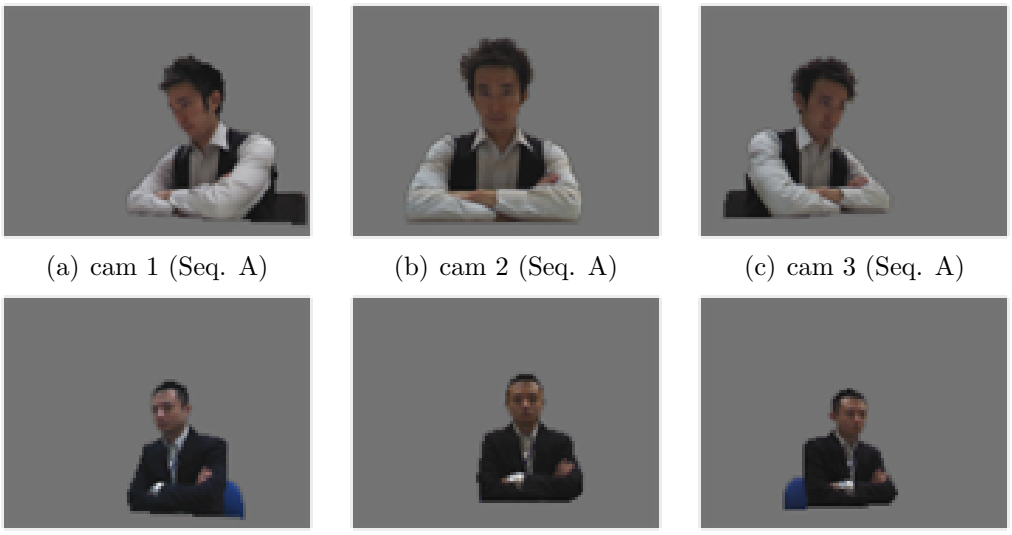
$$\text{Precision} = \frac{\#\text{TP}}{\#\text{TP} + \#\text{FP}} \tag{3.13}$$

$$\text{F} - \text{measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \tag{3.14}$$

Figure 3.14 shows the result of the segmented foreground texture obtained by our proposed method. Results by the single-view method, multi-view method and depth-based method are also shown in Figure 3.15, Figure 3.16 and Figure 3.17, respectively.

As shown in these results, it is obvious that the foreground texture extracted by the proposed method was almost equivalent to the ground truth

image, except that some false negatives could be found especially in the circled region in Figure 3.14. Such false negatives seemed to have been caused by the color similarity between the foreground and the background in all the cameras. In contrast, the single-view and multi-view method suffered from major degradation caused by false negatives or false positives. For the single-view method, false negatives were unavoidable in the region whose background texture was similar to that of the foreground for each camera. The optimization of the threshold value could not solve such problem due to the trade-off between the false positives and false negatives. The multi-view method avoids false negatives based on the visual information obtained with multi-view images. However, it suffers from false positives since the calculated object existence probability in the background region was higher by simply averaging the likelihood among cameras. Therefore, some of the background regions were extracted as false positives. As for the results of Kinect (Figure 3.12), false negatives occurred in the contour regions since the valid depth value could not be obtained in the edge parts.

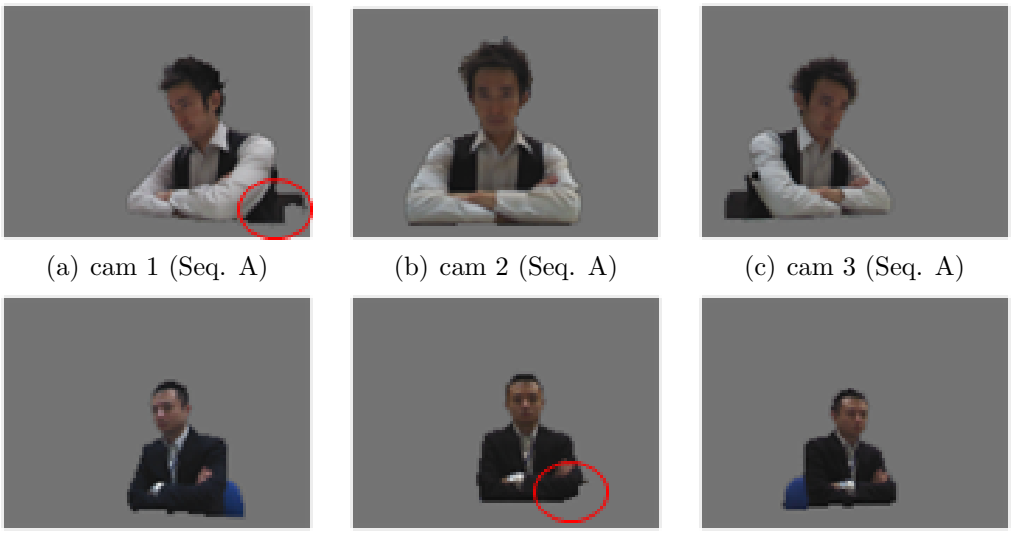


(a) cam 1 (Seq. A) (b) cam 2 (Seq. A) (c) cam 3 (Seq. A)

(d) cam 1 (Seq. B) (e) cam 2 (Seq. B) (f) cam 3 (Seq. B)

Figure 3.13: Ground truth of foreground region for both sequences.

Tables 3.1 and 3.2 show the quantitative performance by the proposed scheme. The results by comparing the schemes described above are also included in those tables. From the results of the single-view method, the



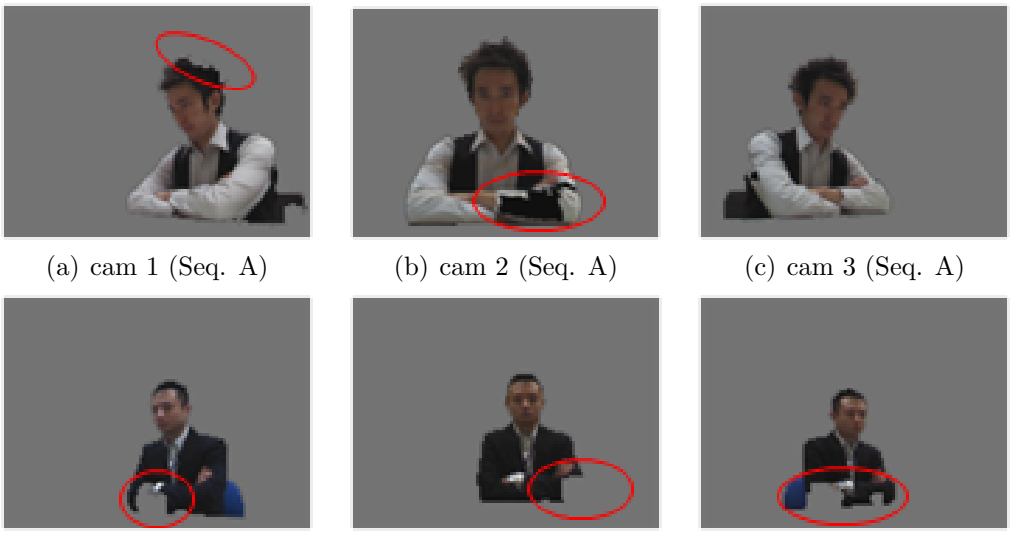
(a) cam 1 (Seq. A) (b) cam 2 (Seq. A) (c) cam 3 (Seq. A)

(d) cam 1 (Seq. B) (e) cam 2 (Seq. B) (f) cam 3 (Seq. B)

Figure 3.14: Foreground of the proposed method for both sequences.

precision values were the highest, but the recall values were smaller than those of the proposed method and multi-view method. This was because both sequences included the background whose texture was similar to that of the foreground object for each frame, and the single-view method was controlled to avoid false positives. The proposed method showed slightly reduced precision values, but there was a considerable increase in recall values compared to the single-view method. This shows that the proposed method is especially useful for reducing false negatives even if the background has a similar texture to the foreground in a certain camera. The multi-view method also showed a considerable rise in the recall values compared to the single-view method and Kinect (depth based) method; however, the precision values substantially decreased in comparison to the single-view method. This shows that the multi-view method was effective in reducing false negatives; however, unwanted background regions were also extracted as foreground.

The experimental results confirmed that F-measure values were maximized by the proposed method for both sequences. This showed that the ratio of false positives and false negatives to the ground truth was minimized by the proposed method. In addition, it was also verified that the gain of the proposed method was mainly attributed to edge regions such as human



(a) cam 1 (Seq. A) (b) cam 2 (Seq. A) (c) cam 3 (Seq. A)

(d) cam 1 (Seq. B) (e) cam 2 (Seq. B) (f) cam 3 (Seq. B)

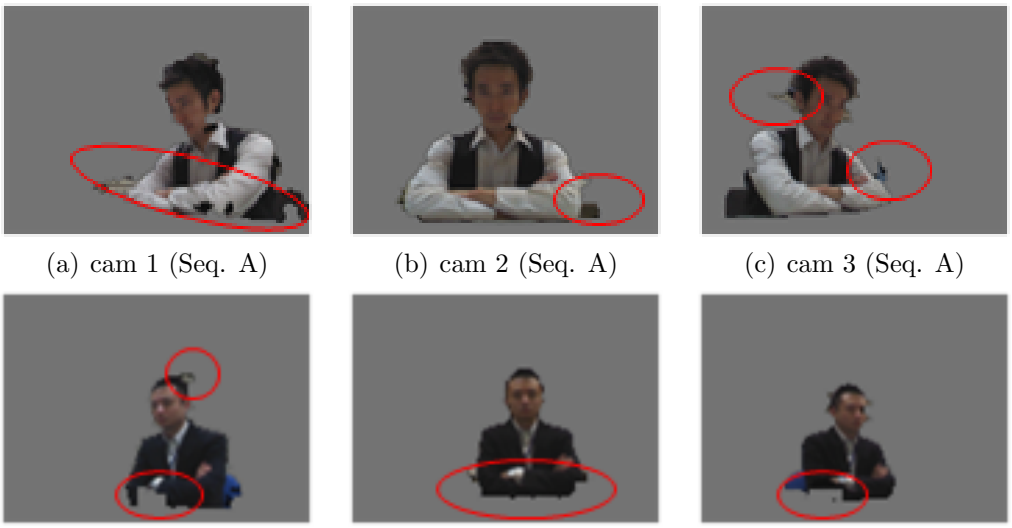
Figure 3.15: Foreground of single-view method for both sequences.

hair, which is especially important for subjective image quality.

3.4.2 Experiment 2: Synthesis for Motion Parallax

In order to confirm the experience of motion parallax functionality achieved by the proposed scheme, some virtual viewpoint images were synthesized from real camera images. For comparison, the virtual viewpoints were synthesized using original depth without the inpainting algorithm. Furthermore, in order to evaluate the effectiveness of the dis-occlusion inpainting algorithm, a simple inpainting method (using Adobe Photoshop) was applied to the virtual viewpoint image.

Figure 3.18 shows the synthesized images for the three virtual viewpoints of v1, v2, and v3 relative to the position of actual cameras as shown in Figure 3.19. In the view synthesis process, the texture of the background was prepared based on the simplified 3D-model of a wall and a table in a meeting room. Although there are some artifacts, which seem to be caused by the error in a 3D model reconstruction, it was confirmed that natural view-change of synthesized images according to a user's head motion was properly realized. A user could obtain such an experience using an auto-stereoscopic



(a) cam 1 (Seq. A) (b) cam 2 (Seq. A) (c) cam 3 (Seq. A)

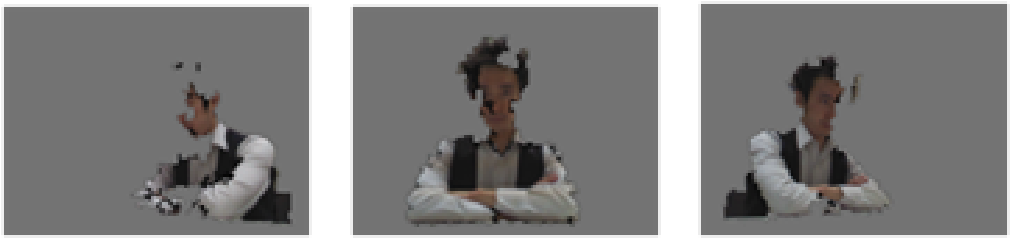
(d) cam 1 (Seq. B) (e) cam 2 (Seq. B) (f) cam 3 (Seq. B)

Figure 3.16: Foreground of multi-view method for both sequences.

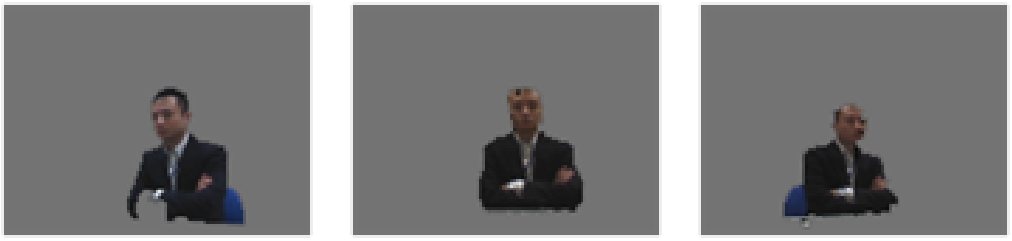
3D display.

Then, in order to evaluate the effectiveness of the proposed view synthesis, the simple view synthesis using original depth without the inpainting algorithm was also evaluated. The comparison results are shown in Figure 3.20. From Figure 3.20, we observe that visual artifacts such as the lack of texture in the edge (hair) and dis-occlusion (name tag) areas were recovered to a certain extent. In addition, image (c) shows that we can have experience of eye-contact according to the position of the gaze direction. Furthermore, in order to evaluate the effectiveness of the dis-occlusion inpainting algorithm, the simple inpainting method (using Adobe Photoshop) was applied to the virtual viewpoint image. The comparison results are shown in Figure 3.21. The dis-occluded area (name tag) was correctly synthesized by the proposed method based on the texture in the previous frame, while the area was erroneously filled with textures in the surrounding regions by the simple inpainting method.

Furthermore, a real time demo system for the local space, where the texture of a remote attendee with CG background is projected on the screen, was developed. In order to realize real time processing, the resolution or RGB image for each Kinect sensor was set to 640×480. The shooting and



(a) cam 1 (Seq. A) (b) cam 2 (Seq. A) (c) cam 3 (Seq. A)



(d) cam 1 (Seq. B) (e) cam 2 (Seq. B) (f) cam 3 (Seq. B)

Figure 3.17: Foreground of Kinect for both sequences.

viewing environments in local space and remote space are shown in Figure 3.22. In Figure 3.22, (a) is an RGB image of a remote space, (b) and (c) are the viewing environments of a local space.

In a demo system, the head position of a local user detected with the Kinect sensor is sent to remote space, and the texture of the human region with CG background in the virtual viewpoint image is rendered in remote space. Then, the texture is encoded to H.264 video format, and sent back to the local space. Finally, the H.264 sequence is decoded and displayed in the local space. The system is composed of a middle-end Desk-top PC equipped with Intel Core i7-3930K CPUs (3.20GHz), 8 Gigabytes of memory, and a NVIDIA Quadro 4000 graphics card. The computational cost for each frame is approximately 33ms; 3D polygon model reconstruction takes about 11ms, virtual view rendering according to a user's head position takes about 12ms, and the transmitted/received process takes about 10ms, respectively.

Pairs of an RGB image of a remote space and the scene of a local user are shown in Figure 3.23. In addition the POV (Point of View) for a local user was captured with a head mounted camera as shown in Figure 3.24. The images in Figure 3.23 and Figure 3.24 show that a local user can experience the motion parallax according to his head movements. Finally, simple pro-

Table 3.1: Comparison of quantitative measurement for Seq. A.

	Recall	Precision	F-measure
Proposed	0.985	0.995	0.990
Single-view	0.899	0.996	0.945
Multi-view	0.969	0.963	0.966
Kinect	0.836	0.987	0.905

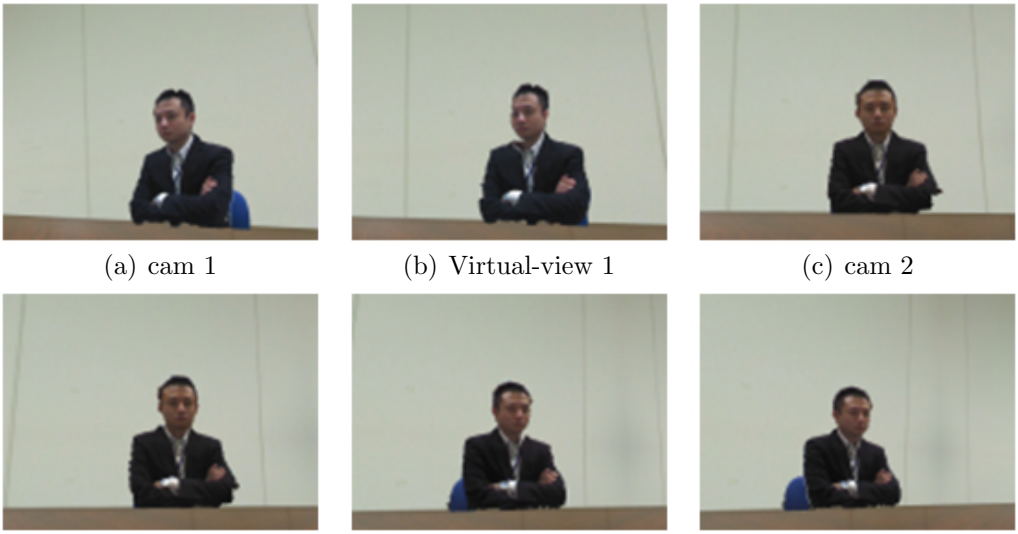
Table 3.2: Comparison of quantitative measurement for Seq. B.

	Recall	Precision	F-measure
Proposed	0.983	0.990	0.987
Single-view	0.940	0.997	0.968
Multi-view	0.975	0.981	0.978
Kinect	0.922	0.977	0.948

jection mapping into the table was applied to enhance the feeling of space continuity, and some images capturing the scene of the local user are shown in Figure 3.25. From the results, it was confirmed that the feeling that users share the same space can be experienced using simple projection mapping for the table area.

3.5 Conclusion

To realize the robust segmentation of foreground objects, such as human regions, from sparsely arranged multi-view cameras, we proposed a method that combined both a labeling process utilizing the object existence probability for an individual camera and energy minimization based on the color similarities among multiple cameras. The experimental results using actual multi-view sequences confirmed the effectiveness of the proposed method regarding foreground segmentation accuracy compared with the conventional work. Furthermore, it was also confirmed that the experience of motion parallax according to head movement could be naturally realized by synthesizing virtual viewpoint images based on the foreground texture and a 3D model of



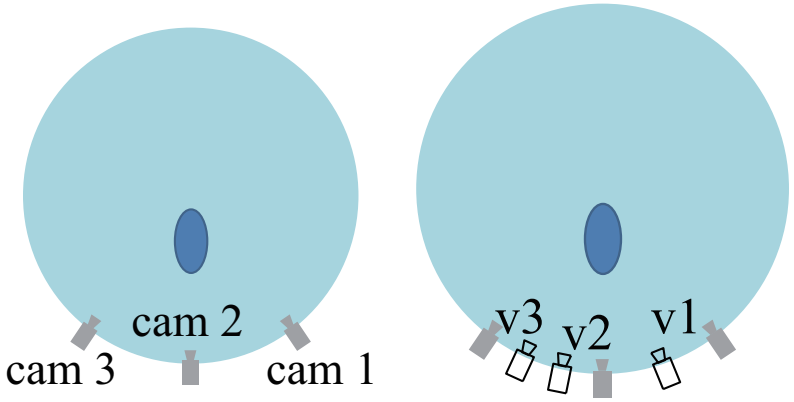
(a) cam 1 (b) Virtual-view 1 (c) cam 2

(d) Virtual-view 2 (e) Virtual-view 3 (f) cam 3

Figure 3.18: Synthesized virtual viewpoint images for Seq. B.

every camera obtained with the proposed method.

As future work, we need to introduce a real-time implementation for high definition resolution (1920×1080) based on GPGPU architectures.



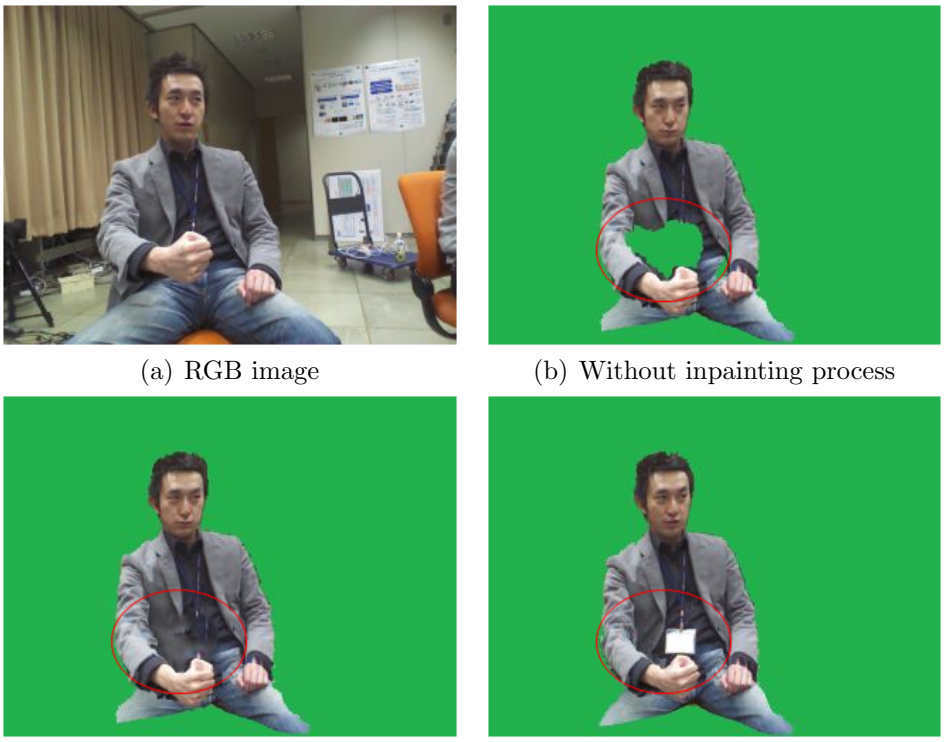
(a) Camera configuration (b) Positions of virtual viewpoints

Figure 3.19: Positions of virtual viewpoints.

(a) RGB Image 1 (b) Comparison 1 (c) Proposed method 1

(d) RGB Image 2 (e) Comparison 2 (f) Proposed method 2

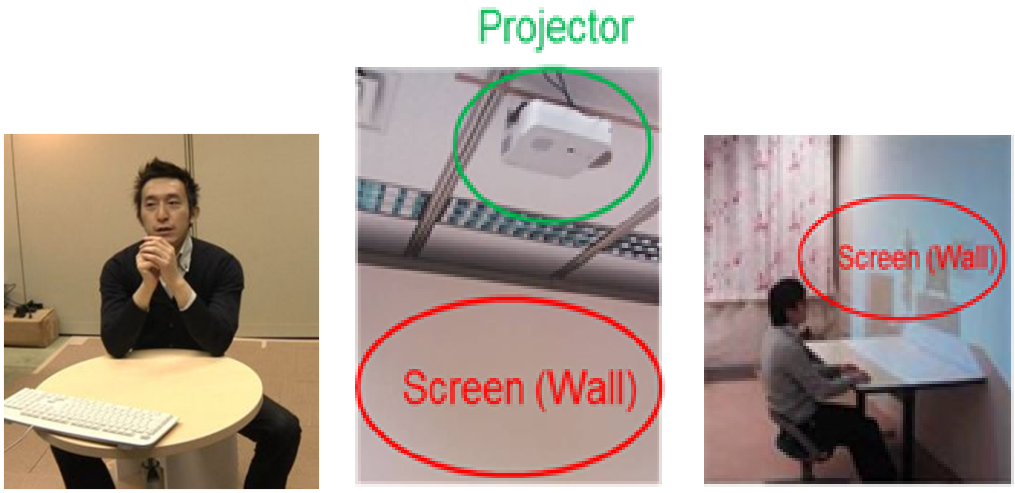
Figure 3.20: Synthesized virtual viewpoint images.



(a) RGB image (b) Without inpainting process

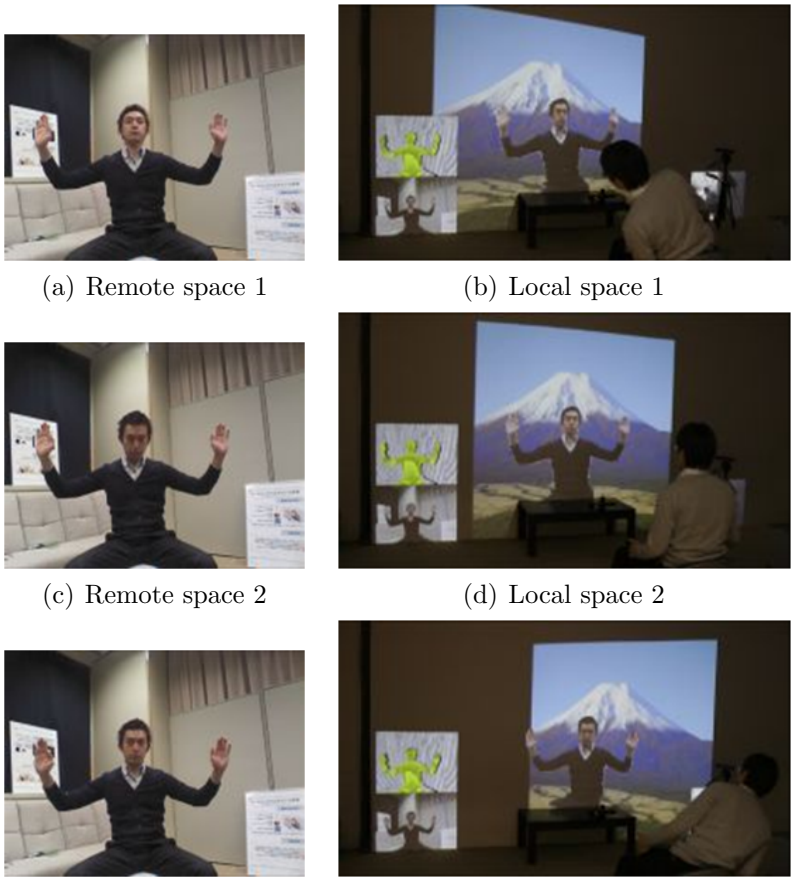
(c) Simple inpainting (d) Proposed method

Figure 3.21: Comparison of dis-occlusion inpainting results.



(a) Remote space (b) Local space 1 (c) Local space 2

Figure 3.22: Experimental environments in local space and remote space.

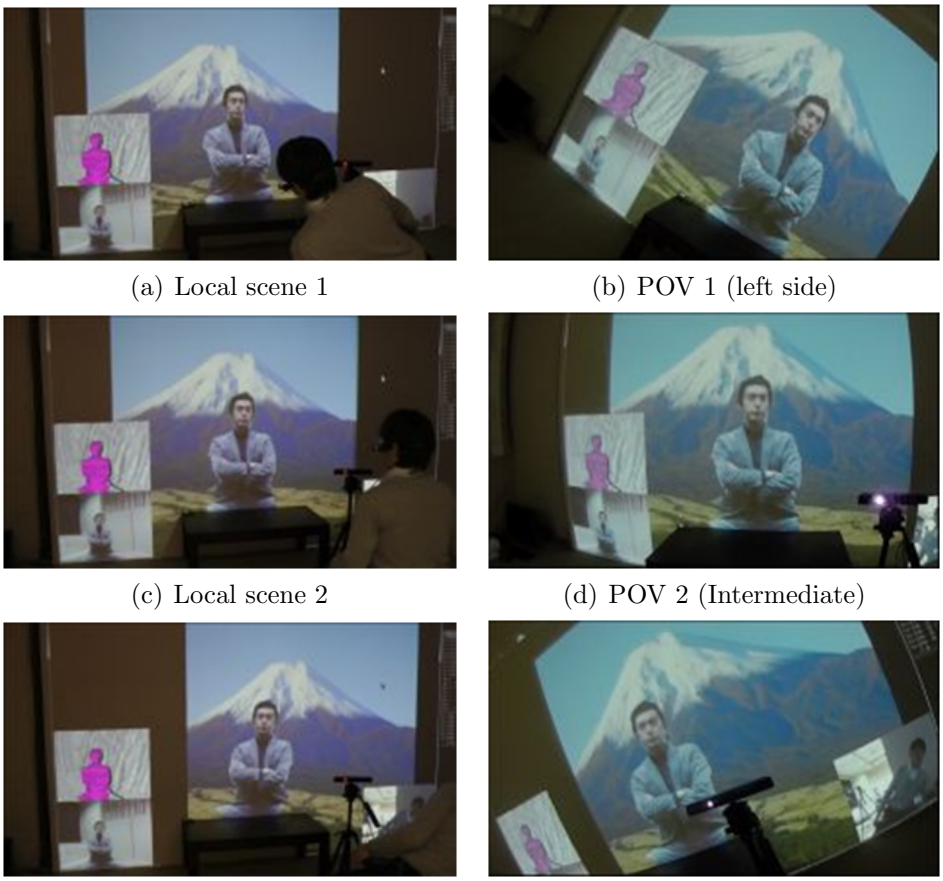


(a) Remote space 1 (b) Local space 1

(c) Remote space 2 (d) Local space 2

(e) Remote space 3 (f) Local space 3

Figure 3.23: Pair images of remote space and local space.



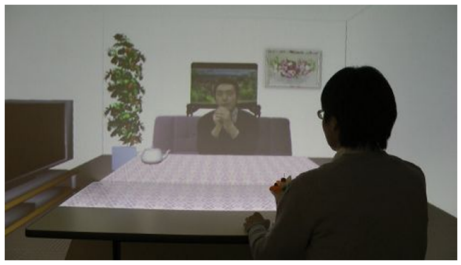
(a) Local scene 1 (b) POV 1 (left side)

(c) Local scene 2 (d) POV 2 (Intermediate)

(e) Local scene 3 (f) POV 1 (right side)

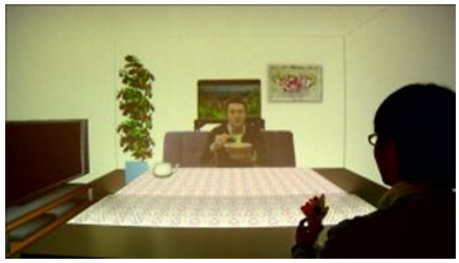
Figure 3.24: POV images for a user in local space.

(a) Remote space 1



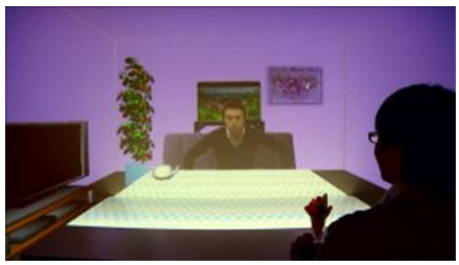
(b) Local space 1

(c) Remote space 2



(d) Local space 2

(e) Remote space 3



(f) Local space 3

Figure 3.25: Simple projection mapping for table.

Chapter 4

Object Extraction for 2D Billboard

In regard to 2D billboard, each object is represented as a set of 2D slice silhouettes and visual textures extracted from all the cameras, and the object position can be calculated in every frame based on the 2D coordinate of the bottom line of the silhouette region and homography matrix between captured frame and 2-dimensional world coordinate model of the target space. Homography matrix estimation and object extraction for moving cameras are two of the most important processes for 2D billboard. In addition, the texture of an occluded object in a certain camera cannot be extracted precisely, since some parts of the texture region are not visible in the camera. In order to reconstruct a 2D billboard of an occluded object precisely, the object's texture has to be extracted from another frame of the same camera or another camera of the same frame in which the object is not occluded from the other objects.

The chapter presents an approach of object extraction to overcome occlusion problems for 2D billboard based on the above mentioned framework and the utilizing information, that is, correspondence relationship between an arbitrary 2D pixel coordinate in every camera and 2D world coordinate of the specific plane in the target space. Our proposed method estimates homography matrices semi-automatically by identifying reliable corresponding feature points between video frames, and also extracts the precise object regions using estimated homograph matrices. Experimental results revealed that the proposed method successfully estimated the precise homography matrices compared to the conventional method. Moreover, it was also con-

firmed that the proposed scheme contributes to further improvement in the experience of free-viewpoint video since the textures of multiple objects were successfully extracted.

Furthermore, we propose a robust object tracking scheme among multi-view cameras and consecutive frames for detecting and interpolating occlusion regions. For a free viewpoint video that provides users with an immersive experience, each object has to be identified consistently among all cameras for every frame in order to share textures of the same objects and replace the textures when an occlusion occurs. In order to satisfy the above requirement, the proposed method extracts objects' silhouette regions and tracks each identified object by associating a closed silhouette region with a tracking ID for every camera. As the frame by frame process, our method confirms whether occlusion occurs for each tracking region, and modifies the texture region by projecting the world coordinate of the object in 3D-space, that can be estimated from the another non-occluded camera image if it is available. Experimental results revealed that the proposed method achieved more robust texture extraction of multiple objects especially for occluded scenes compared to the conventional methods. Furthermore, it was also confirmed that the proposed scheme can improve the experience of free-viewpoint video as the result of precise reconstruction of occluded regions.

4.1 Introduction

A free-viewpoint video provides a beyond-3D experience, in which audiences can see real scenes from anywhere in a 3D space [1][23][24]. In the free-viewpoint video system, the virtual viewpoint can be interactively selected and moved back-and-forth and around as well as up-and-down among objects within a space where cameras cannot be mounted. Such a system gives audiences an immersive feeling, and we call these view-changing experiences "walk-through" and "fly-through" [25][6][26].

There are two main techniques for rendering a free viewpoint video. One is a model-based method [1][23][24] and the other is an image-based method [26]. In order to realize the viewing experiences mentioned above, the former method is more suitable than the latter since the former has no restrictions on virtual viewpoints in a 3D space, if a 3D model of the scene can be appropriately reconstructed. Therefore, in this study, we focus on a model-based method, which can be also divided into two approaches; an accurate model

scheme and a simplified alternative model scheme. Reconstruction of an appropriate 3D model especially in a large space such as a soccer stadium is very difficult, and requires a number of cameras whose projection matrices, which project the points in a 3D world coordinate into the pixels in a 2D image plane, have to be precisely estimated. Instead of reconstructing an accurate 3D model, a method of rendering free-viewpoint video in a soccer stadium using a simplified 3D model called billboard has been proposed [7][8][9][27].

The authors are especially interested in dynamic sports video captured in a large space such as an outdoor soccer stadium, and have been working on the research to synthesize free-viewpoint video for sports scene based on a simplified 3D model called billboard [7][8][9][27]. In these methods, each object such as a soccer player is represented as a billboard, a rectangular plane vertical to the ground, and the visual texture acquired from multi-view cameras is mapped according to the virtual viewpoint indicated by viewers. In order to reconstruct billboard models, the projection matrix is unnecessary, since each object such as a soccer player is represented as a billboard, a rectangular plane vertical to the pitch field, and the visual texture acquired from multi-view cameras is mapped according to the virtual viewpoint selected by viewers. In order to reconstruct billboard models, each object's world coordinate on the field can be estimated using the object's texture region and a homography matrix between the real scale field plane and each camera image. Therefore, a billboard-based method realizes rendering of not only intermediate viewpoints between the real cameras but also any virtual viewpoint in a 3D space, with at least one camera.

Most of the conventional free-viewpoint video rendering methods use only static cameras whose homography matrix and background model do not change little while shooting [7][8][9][27]. Therefore, homography matrix estimation is necessary only for the first frame and object extraction can be achieved based on background subtraction method. However, since static cameras have to be set widely enough to capture the entire scene, the resolution of each object becomes low and is not sufficient for rendering high-quality free-viewpoint video. In addition, broadcasters usually utilize moving pan-tilt-zoom cameras in sporting events, and therefore, those conventional methods cannot be directly applied to the events. Captured images from the moving cameras can improve the image quality in rendering free-viewpoint video, but accurate homography matrix estimation and object extraction with moving pan-tilt-zoom cameras has not been realized so far.

In order to solve the problems mentioned above and to extend the range

of application of free-viewpoint video technology to be used in sport live broadcasting, an alogorithm for homography matrix estimation and object extraction for a moving broadcast camera is proposed in this paper. The proposed method targets 5 to 10-seconds soccer highlights like shot on goal scene captured by a common broadcast camera tracking the movement of ball from a certain level of wide angle. Specifically, our goal is to synthesize free-viewpoint videos of such an highlight scene under 15 minutes (manual operation is only allowed for initial frame) to be used in sport live broadcasting without advance preparations such as intensity camera calibration necessary for estimating projection matrices. Our proposed method estimates not projection matrices but homography matrices by identifying reliable corresponding feature points between consecutive two pair of video frames using only captured video information without intensity camera calibration. Furthermore, by using the estimated homography matrices, the method extracts objects' texture regions in each frame of a moving camera. In this process, objects crossing the boundary between the field and the outside such as the spectator's stand can be identified.

On the other hand, extraction of highly accurate texture regions is a strong requirement for rendering free viewpoint video based on a billboard model; the quality of the user experience for watching free-viewpoint video highly depends on the extraction accuracy of each object's texture region. This is because false positives such as background regions or another object's texture connecting to that of the target object and false negatives such as an imperfect texture lacking legs and/or arms will greatly disrupt the viewing experience. In particular, such false negatives are often caused by occlusion. That is, the texture of an occluded object in a certain camera cannot be extracted precisely, since some parts of the texture region are not visible in the camera. In order to reconstruct a billboard model of an occluded object precisely, the object's texture has to be extracted from another camera in which the object is not occluded from the other objects. Each object is thus required to be identified consistently among all cameras for every frame to share textures of the same objects and replace them when an occlusion occurs. This is categorized into the study of tracking and extracting multiple objects, and is a very challenging task primarily due to difficulties of object extraction and occlusions among objects.

Numerous conventional methods were proposed on this task, and high performance results for tracking accuracy were shown by conducting experiments using various test sequences including soccer contents [28][29]. How-

ever, most of these conventional methods cannot be directly applied to the rendering of free-viewpoint video, since their main purposes for sport events are analysis for understanding strategy or realizing an automatic intelligent robot camera in TV program production. Such applications do not require precisely identified texture regions, and those methods only extract a coarse region which includes many false positives and false negatives.

In order to overcome the problems mentioned above, we propose a robust method of tracking and extracting multiple objects from the fixed multiple cameras for the purpose of rendering an immersive free-viewpoint video of soccer match. The main contribution of our work is to extract precisely identified object regions even if occluded, by detecting occlusion regions of each camera properly for every frame. Furthermore, the method extracts each object's texture region from every camera image and estimates the world coordinate of the object in every frame to render free-viewpoint video. For occlusion regions, the method extracts the texture of occluded objects from another camera or the other frame in which the objects are not occluded.

The remainder of this chapter is organized as follows. Section 4.2 overviews related work, and Section 4.3 details our proposed method based on robustly estimating the homography matrix for every frame. Section 4.4 presents the experimental results and compares performance with a conventional method. Finally, the paper is concluded in Section 4.5.

4.2 Related Works

4.2.1 Homography Matrix Estimation

In order to estimate the homography matrix of a moving camera, the corresponding feature points between the consecutive frames and the pitch field have to be extracted precisely. homography matrix estimation algorithms for sport videos such as tennis and soccer based on detecting court or pitch field lines from the camera image and assigning them to the court model have been proposed [30][31]. In these literatures, the effectiveness was shown for some test sequences. In that scheme, in order to estimate the homography matrix of every frame, image pixels are classified as court line pixels if they meet several criteria including color and local texture constraints. Then, model fitting is conducted to find correspondences between the line of the image and that of the court model. Specifically, homography matrix estimation refinement

based on the algorithm of non-linear gradient descent (Levenberg-Marquardt minimization) and model tracking from previous frames were applied to each frame. However, court lines are not necessarily white objects in the images, and thus the performance might be degraded dependent on the precision of line detection.

Some other camera calibration algorithms for broadcasting sports videos have been proposed [32][33]. The literature [32] was mainly focused on broadcast basketball videos, which have more difficulty accompanied by frequent player region changes compared to soccer videos, and the algorithms in [30] and [31] were modified to be applicable to basketball videos. Similarly, the literature [33] mainly targeted volleyball videos, and camera calibration was conducted based on the algorithm of [30]. These methods have the same problem as that of the literature [30] and [31], that is, camera calibration accuracy depends on the performance of line detections.

4.2.2 Object Extraction

Furthermore, when a camera is moved, object extraction based on static background subtraction cannot be conducted. For this problem, object extraction algorithm for a moving camera was proposed [34][35]. In the literature [34], motion compensated difference keying against the background plate, which can be constructed by piecewise projection of the camera images into a spherical map, was proposed, and the effectiveness for soccer videos was shown. However, in order to construct the background plate, projection matrix has to be estimated precisely for every frame, and this estimation is out of requirements for the target video sequences of this paper described in Section 1. It is desired that the method using only captured video information without prior information should be proposed to be applied to sport live broadcasting. On the other hand, the method in [35] applies Gaussian Mixture Model (GMM) to model the background as composed of two areas: the court area and its surrounding area, which computed directly from the estimated homography matrix. In addition, object regions are extracted based on expectation maximization (EM) procedure of GMM for the purpose of object tracking. From the experimental results for several test sequences such as tennis, badminton, and volleyball, the effectiveness in object tracking was shown. However, the method extracts object regions not as silhouettes but as rectangles, and therefore, presice textures for billboard model cannnot be extracted. Furthermore, those video sequences consisted of only simple

textures in both the court area and its surrounding area. For the video sequences of actual soccer games, which include more complicated background textures, this scheme cannot provide sufficient precision in object region extraction to render high quality free-viewpoint video.

4.2.3 Object Tracking

Multiple object tracking has a long tradition, and is intensively studied in the research area of computer vision. State-of-the-art methods can be divided into single view and multi-view approaches. Single view approaches have advantages of simple and easy developments, but they rely on limited 3D information in a single camera. Multi-view approaches provide precise 3D information about the objects and the space by making use of redundant information from different views. Especially, a Kalman filter and a particle filter offer a framework for representing the tracking uncertainty by only considering information from the past frames, and are suitable for object tracking in live sports events such as a soccer game.

The Kalman filter [36] or particle filter [37] is widely used for both approaches in this field. The Kalman filter is effective for estimating a target state according to consecutive frames when occlusion occurs infrequently. However, each object's state is limited in the Gaussian distribution model, and it is difficult to retain each object's identifier when occlusion occurs. On the other hand, the particle filter addresses some of the limitations of the Kalman filter by exploring multiple hypotheses and has superior performance in complicated environments such as outdoor spaces including illumination variation. [36][37].

The combination of the particle filters and tracking-by-detection approaches is very useful for handling occlusions [38]. These approaches only rely on the final sparse outputs from the object detector which usually include false positives and false negatives, and therefore, the tracking accuracy highly depends on the accuracy of object detections. In order to improve these methods, the paper [28] integrates the object detector itself into the tracking process by monitoring its continuous detection confidence and using it as a graded observation model, and shows the robust tracking performance. These methods have an advantage that they can also be applied to a single moving uncalibrated camera such as a broadcasting camera. However, the object detector needs much training data composed of the positive such as object regions and the negative such as background regions in advance, and the cost of

these preparations is enormous. Furthermore, precise object regions and 3D world coordinates that are necessary for rendering free-viewpoint video cannot be extracted since the detection results include false positives and false negatives.

Homograph-based tracking by particle filtering methods, which extract the principal axes of upright humans tracked in each view and then combine multiple views using a planar homograph, was proposed and the effectiveness was shown in some test sequences [39]. The paper [39] proposed an approach that avoided explicit calibration of cameras and instead utilized constraints on the field of view lines between cameras to track objects across the cameras, while other relevant approaches require calibrated cameras to fuse information in a 3D space. These and similar methods track objects in individual un-calibrated cameras and then create associations across cameras for better localization. For greater robustness, the paper [39] extends the homograph-based concept above to multiple planes parallel to the reference plane. The algorithm neither localizes nor tracks objects from any single viewpoint, rather, evidence is gathered from all the cameras into a unified synergistic framework where occlusion resolution, detection, and tracking are performed simultaneously. These methods are very robust for challenging multi-view sequences including soccer video. However, along with the other approaches, a texture region of each object which is sufficient for rendering high-quality free-viewpoint video cannot be extracted. In order to render high-quality free-viewpoint video, extraction of precisely identified object regions among the multiple cameras, even if occluded, has to be achieved for every frame.

4.3 Proposed Method

In order to overcome the problems mentioned above, we propose a semi-automatic homography estimation method to achieve the robustness for the color similarity between objects and court lines. And then, we also propose an object texture extraction method based on estimating homography matrix for every frame. Furthermore, robust object tracking algorithm is proposed to extract precisely identified object regions even if occluded, by detecting occlusion regions of each camera properly for every frame.

The proposed method has a series of procedures as shown in Figure 4.1, and is summarized into the following tive stages; the first is an initialization

process on parameter settings for the initial frames, the second is a homography matrix estimation process for every frame in an input video sequence, the third is a texture extraction method based on the results of the second process, the fourth is a obcect tracking method based on occlusion detections, and the fifth is a free-viewpoint video rendering method. These processes are described in more detail in the following.

4.3.1 Initialization Process

A homography matrix is a planar transformation, and for a point on the field plane, the relationship between the 2D pixel coordinate (u, v) in a video frame and the 2D world coordinate (X, Y) on the field plane is represented by equation (4.1) using the homography matrix H_{frame} and a scalar parameter s.

$$(X, Y, 1)^T = s\boldsymbol{H}_{frame}(u, v, 1)^T \tag{4.1}$$

H_{frame} can be computed based on the least-square method by obtaining at least four correspondences between a video frame and the field plane. In the soccer scene, the lengths between lines on the field plane are known, and thus the 2D world XY-coordinate of some feature points can be determined in advance. Therefore, the homography matrix between the initial frame and the field plane can be estimated by manually designating the corresponding feature points. Figure 4.2 shows the correspondence relationship of feature points between the first frame and the field plane, and each feature point (indicated by an red cross) is projected with the homography matrix H_{frame} estimated by equation (4.1).

In Step 2, the object region and the 2D XY-coordinate of each object on every camera for the initial frame are calculated based on the following chrome-key based segmentation. In a soccer game, a chrome-key based operation is useful for extracting the textures of objects such as soccer players because a playing field consists of a green grassy region with white lines. In the first frame, a rectangular region composed of only green pixels are indicated by manual operations, and an average μ_c and a variance σ_c^2 of all the pixels' values I_c^k (k represents a pixel index) in the region are calculated for each color component c of the color space such as RGB or YUV. For every frame, green pixel removal process based on the threshold for each color component is done by equation (4.2) where th_c indicates the threshold.

If all the componets satisfy the equation (4.2), the pixel k is removed. By applying green pixel removal process, object regions in the green field plane can be precisely extracted, however, unwanted pixel noises in spectator stand area still remain. For the initial frame, those remained unwanted regions are manually eliminated, and a binary image is generated for every camera as shown in Figure 4.3. Here, object regions are extracted as a set of pixels that do not satisfy inequation (4.2) for at least one color component, and are represented in white, while background regions are represented in black.

$$\mu_c - \sqrt{\sigma_c^2} - th_c < I_c^{(k)} < \mu_c + \sqrt{\sigma_c^2} + th_c \tag{4.2}$$

In Step 3, the consistent tracking identifier tr among multiple cameras are assigned to each object. The white pixels in every camera image of the step 2 are divided into the closed regions based on the eight neighboring pixel connections, and each closed region is represented as a rectangle as shown in Figure 4.4 (a). However, the rectangle including occlusions does not correspond to an individual object region, so for the initial frame, we manually segment the closed region into the proper number of rectangles, and the tracking identifier tr is set for each object region in a certain camera as shown in Figure 4.4 (b). Then, the pixel coordinate of the center point of each rectangle's base is acquired as the object's foot position, and the 2D XY-coordinate on the ground plane is calculated by translating the pixel coordinate above with the homograph matrix H_{cam} as shown in Figure 4.5. In order to assign the tracking identifier to each object consistently among the multiple cameras, we project the 2D XY-coordinate of each object into the other cameras using the homograph matrices, and calculate the distance between the projected point and the object's foot position in every camera. Then the tracking identifier is assigned to the nearest object according to the distance above, and for each object, we check whether or not the same identifier is assigned among all the cameras. If there exist objects whose identifiers among the multiple cameras are not consistent, we manually correct the identifier. Some manual operations described above are only necessary for a portion of regions in each camera such as occlusions of the initial frame.

In Step 4, the particle filter is set for each object region with the consistent tracker number tr among the multiple cameras for the initial frame. For each object's filter, N particles are sampled in the bounding box set as a rectangle of each object. Additionally, appearance information is set for each

object's filter according to the object's uniform color. In a soccer game, the uniform color is divided into five classes such as referees, two goal keepers, and two teams of other players. The state of each particle is defined as equations (4.3), (4.4), and (4.5). Here, $(u(t), v(t))$ represents a pixel coordinate, and $(\Delta u(t), \Delta v(t))$ represents a velocity component in every camera image. In order to propagate the particles, we assume a constant velocity motion model as follows using a process noise $\omega(t)$ drawn from zero-mean Normal distribution; Σ_4 represents a covariance matrix which consists the covariance of $(u(t), v(t))$ and that of $(\Delta u(t), \Delta v(t))$.

$$\boldsymbol{c}(t) = (u(t), v(t), \Delta u(t), \Delta v(t))^T \tag{4.3}$$

$$\boldsymbol{c}(t) = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \boldsymbol{c}(t-1)^T + \boldsymbol{\omega}(t) \tag{4.4}$$

$$\boldsymbol{\omega}(t) \sim N(\mathbf{0}, \boldsymbol{\Sigma}_4) \tag{4.5}$$

4.3.2 Homography Matrix Estimation

In Step 5, the corresponding feature points between the frame t and the frame $t+1$ are extracted based on SURF (Speeded Up Robust Feature) descriptors [17][18] as shown in Figure 4.6, since the descriptors are known to be robust for image translations like zooming in/out and rotations that accompany usual broadcasting camera operations (e.g. pan, tilt, and zoom).

Then, in Step 6, each feature point in the frame t is projected on the field plane using H_t, and the projected points existing outside the field plane are eliminated as unwanted points. The remaining feature points are shown in Figure 4.7, in which the red box indicates outside areas of the field plane. The world XY-coordinate of each remaining feature point (u_{t+1}, v_{t+1}) in the frame $t+1$ is given by the relationship between the corresponding point (u_t, v_t) and its corresponding XY-coordinate. As a result, H_{t+1} can be estimated based on the least-square method with the remaining feature points. The resultant matrix H_{t+1} is robust against the miss-matching of the feature points when many feature points exist.

4.3.3 Object Extraction

By applying green pixel removal process based on inequation (4.2), object regions in the green field plane can be precisely extracted, however, unwanted pixel noises in spectator stand area still remain as shown in Figure 4.8 (b). In order to remove the unwanted pixel noises, the objects existing outside the green field has to be identified by projecting the outer line between the field plane and spectator stand area with the estimated homography matrix. At first, object region pixels only in the field plane are extracted, and the pixels existing on the outer line are divided into closed segments based on adjacent connectivity. Then, for each segment on the outer line, rectangle region, whose width and height are set based on the length of segment enough to include the object existing outside the green field, are extracted as an object candidate region as shown by the blue rectangles in Figure 4.8 (c).

In order to extract an object texture from each object candidate region, grab-cut based segmentation [19] is known to be effective. However, the grab-cut algorithm needs some manual operations to indicate the pixels in edge regions between background and foreground for each candidate region, and it takes much time for all the frames. To overcome the problems, automatic segmentation process is introduced as follows. At first, pixels around the corner of a bounding box are extracted as examples of unwanted regions. The average and the variance of those pixel values are calculated for each color component, and the object existence likelihood $\rho_c(k)$ of pixel index k is defined as equation (4.6) based on normal distributions of all the pixels in the indicated region above.

$$\rho_c^{(k)} = 1 - \exp(-\frac{(I_c^{(k)} - \mu_c)^2}{2\sigma_c^2}) \tag{4.6}$$

Then, an energy function considering the relation among adjacent pixels is introduced with the likelihood $\rho_c(k)$. The frame is binarized by minimizing the energy function using a graph-cut algorithm. Energy function $E(\boldsymbol{\alpha})$ is defined with the data term $U(\boldsymbol{\alpha})$ and the smoothing term $V(\boldsymbol{\alpha})$ by equation (4.7), where $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_k, \ldots)$ indicates the labels identifying whether the corresponding pixel belongs to the object textures or not, respectively. The data term $U(\boldsymbol{\alpha})$ depends on only likelihood $\rho_c^{(k)}$ and gives the energy value related to equations (4.8) and (4.9) with the threshold parameter th_c. The smoothing term $V(\boldsymbol{\alpha})$ is defined by equations (4.10) and (4.11) where $\boldsymbol{I}^{(k)}$ and $\boldsymbol{I}^{(l)}$ corresponds to adjacent pixel values with the index k and l ($(k, l) \in N_p$),

respectively. In the equation (4.11), $dis(-)$ is the Euclidean distance, and N_p indicates available candidates of the adjacent pixels for (k, l). Variables λ and κ are the positive constants.

$$E(\boldsymbol{\alpha}) = U(\boldsymbol{\alpha}) + \lambda V(\boldsymbol{\alpha}) \tag{4.7}$$

$$U(\boldsymbol{\alpha}) = \sum_{k} U(\alpha_k) \tag{4.8}$$

$$U(\alpha_k) = \begin{cases} \max_c[\rho_c^{(k)}] & (\alpha_k = 0) \\ \max[0, th_c - \max_c[\rho_c^{(k)}]] & (\alpha_k = 1) \end{cases} \tag{4.9}$$

$$V(\boldsymbol{\alpha}) = \sum_{(k,l)\in N_p} V(\alpha_k, \alpha_l) \tag{4.10}$$

$$V(\alpha_k, \alpha_l) = \begin{cases} \frac{\exp(-\kappa|\boldsymbol{I}^{(k)} - \boldsymbol{I}^{(l)}|^2)}{dis(k,l)} & (\alpha_k \neq \alpha_l) \\ 0 & (\alpha_k = \alpha_l) \end{cases} \tag{4.11}$$

After minimizing the energy function, the objects' textures can be extracted precisely as illustrated in Figure 4.8 (d).

4.3.4 Object Tracking

In Step 9, the likelihood $w_{tr,p}$ of each particle p of the tracker tr on every camera is defined by object existence probability $\rho_{c,exi}^{(k_p)}$ based on a background model and appearance probability $\rho_{c,uni}^{(tr,k_p)}$ based on an uniform color model, as in equation (4.12) with the weighting parameter λ.

$$w_{tr,p} = \lambda \rho_{c,exi}^{(k_p)} + (1 - \lambda)\rho_{c,uni}^{(tr,k_p)} \tag{4.12}$$

For each pixel in a fixed camera image, pixel values for consecutive frames in which each object region is respectively small and each object does not

stop for a long time are approximated by a normal distribution. Therefore, object existence probability $\rho_{c,exi}^{(k_p)}$ is defined by the following equation (4.13). In the equation, $I_c^{(k_p)}$ and $\sigma_c^{(k_p)}$ are the average and the variance of the pixel values for the tracking target frames.

$$\rho_{c,exi}^{(k_p)} = 1 - \exp(-\frac{(I_c^{(k_p)} - u_c^{(k_p)})^2}{2\sigma_c^{(k_p)}}) \tag{4.13}$$

The tracker's uniform color is divided into at most the five classes mentioned above. Object regions in each class are extracted from every camera for the initial frame, and the tracker's appearance model $\rho_{c,uni}^{(tr,k_p)}$ in each class is defined as equation (4.14) based on normal distributions of all the pixel values in the class. In the equation, $\bar{I}_c^{(tr)}$ and $\sigma_c^{(tr)}$ are the average and the variance of pixel values for each uniform color class.

$$\rho_{c,uni}^{(tr,k_p)} = \sum_c \exp(-\frac{(I_c^{(k_p)} - \bar{I}_c^{(tr)})^2}{2\sigma_c^{(tr)}}) \tag{4.14}$$

For every frame t, re-sampling of particles is performed according to the likelihood $w_{tr,p}$ and the threshold th.

In Step 10, we assign each particle of the tracker to the temporal object identifier *label* based on the binary image of object regions which are acquired based on equation (4.2). Association between the tracker number tr and the identifier *label* is achieved by selecting the identifier *label* to which the most number of particles for the tracker are assigned. If the same object identifier *label* is shared by more than two trackers, these trackers are considered as occlusions. For example, in Figure 4.9, the object identifier $label = 3$ is shared by tracker identifier $tr = 1$ and $tr = 2$, and these trackers are determined as occlusions.

The XY-coordinate of each tracker is calculated by translating the associated object's foot position as mentioned in Figure 4.5. Therefore, for a certain occlusion region, the XY-coordinate of each object that belongs to the region will be the same. Figures 4.10 and 4.11 show that the XY-coordinates of tracker $tr = 1$ and tracker $tr = 2$ are estimated as the same position in Camera 1 because of the occlusions, while their coordinates are calculated precisely in Camera 2 because tracking of these objects is successful. For the occluded trackers, 2D XY-coordinates are estimated by calculating the average of those in other cameras in which occlusion does not occur.

In Step 11, the tracker region is initialized using the tracker's XY-coordinate estimated from the other cameras and the associated object region in the previous frame in which the tracker is not considered as an occlusion. Figure 4.12 shows that the tracker region of tracker $tr = 1$ and tracker $tr = 2$ is initialized based on the XY-coordinate of each tracker in Camera 2 and the rectangular region in the past frame.

4.3.5 Free-viewpoint Video Rendering

In Step 12, the bounding box of each object texture is extracted, and the pixel coordinate of the center point of each rectangle's base is acquired as the foot position of each object. Then, the 2D world coordinate on the field plane of each object is calculated by projecting the pixel coordinate above with the estimated homography matrix H_t for every frame. Finally, the billboard model of each object is reconstructed by integrating the texture and the 2D world coordinate for every frame, and the CG virtual space including background model such as soccer stadium for all the frames are produced.

For occluded regions, each extracted bounding box includes more than two objects, and the texture and the 2D world coordinate of an occluded object cannot be precisely calculated. In order to reconstruct a billboard model of an occluded object, the texture has to be extracted from the corresponding object in another frame. Each object is identified consistently among all the frames based on object tracking technology, therefore, both the last minute frame and the immediate frame, in which the target object is non-occluded, can be picked up. For each non-occluded frame, the texture is extracted and 2D world coordinate is calculated, and then, the texture of a target object in the occluded frame is extracted by blending the textures in the last and the immediate frames according to the ratio of the number of frames. The 2D world coordinate is also estimated by interpolating that in both of non-occluded frames, and thus, the billboard model of an occluded object can be reconstructed based on object tracking technology.

Free-viewpoint video of arbitral position and direction can be rendered by texture mapping of each billboard model, and the shadow of each object can be also synthesized by projecting the silhouette region on the field plane based on virtual light source whose position is randomly selected.

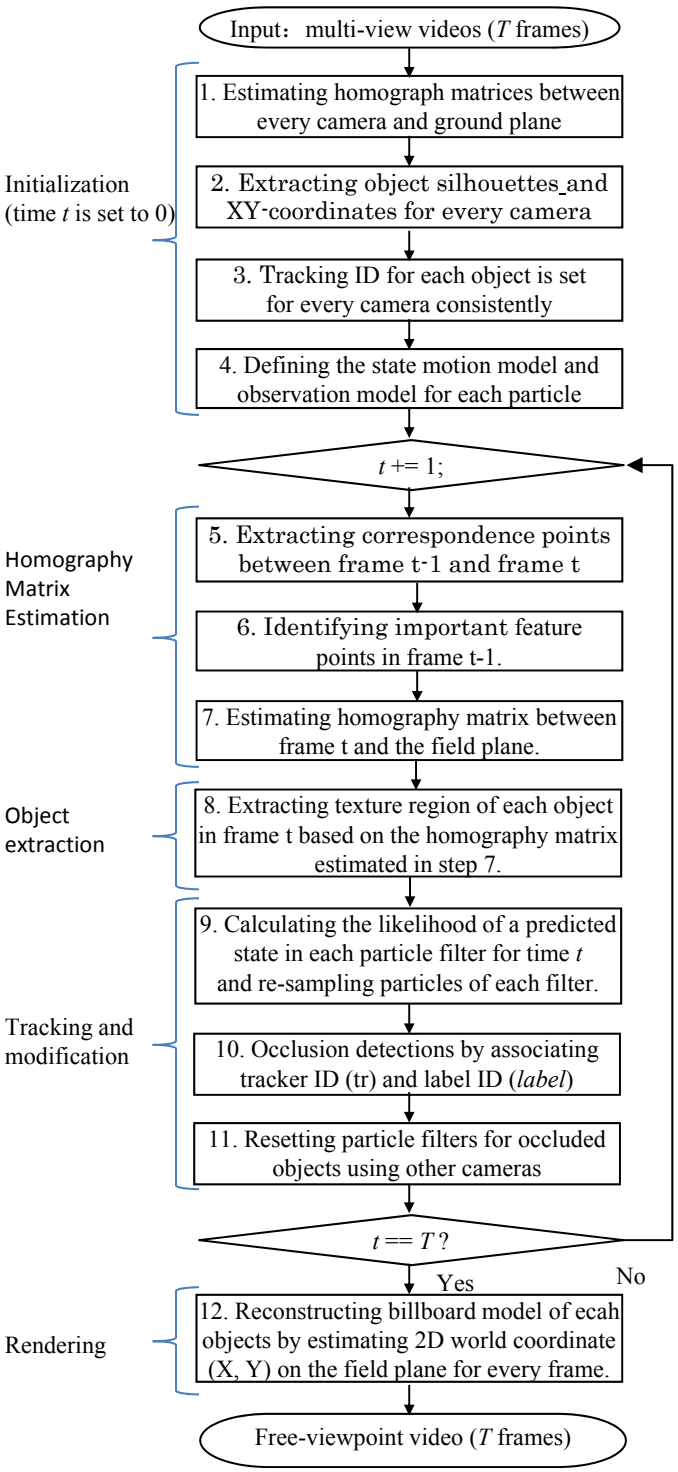
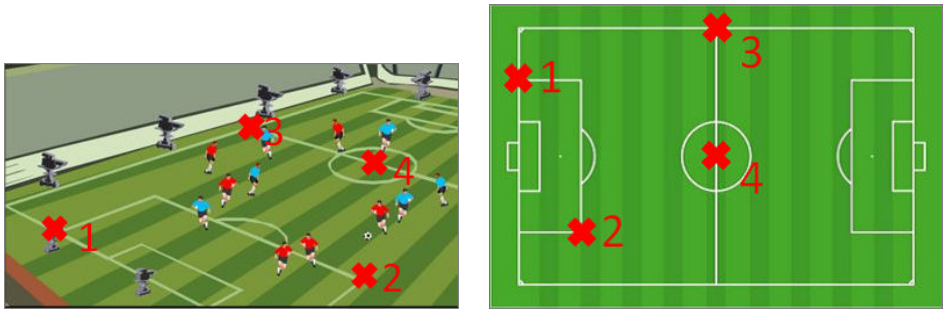


Figure 4.1: Flowchart of the proposed method.

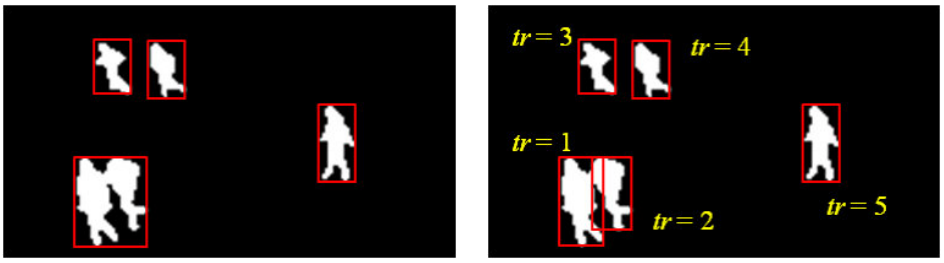


(a) Projected points and lines (b) Feature points on the field plane

Figure 4.2: Examples of projected feature points on the field

(a) Original image (b) Binary silhouette image

Figure 4.3: Extractions of object regions.



(a) Bounding rectangle of objects (b) Object IDs

Figure 4.4: Setting of object IDs

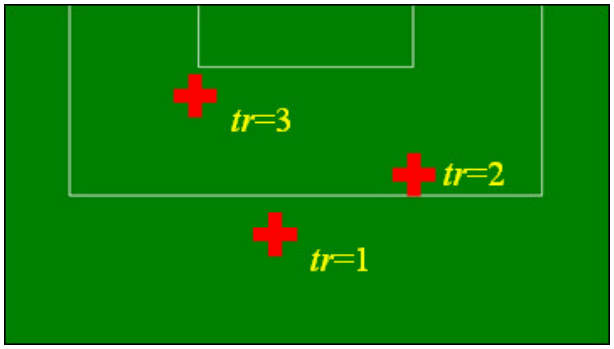
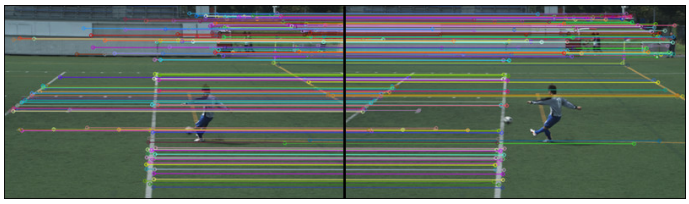
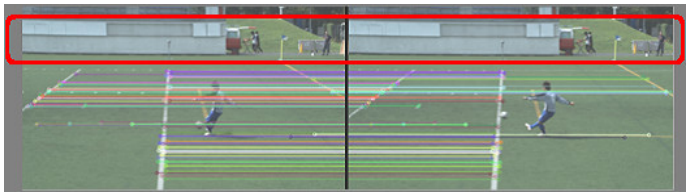


Figure 4.5: 2D XY-coordinate of each object on the ground plane



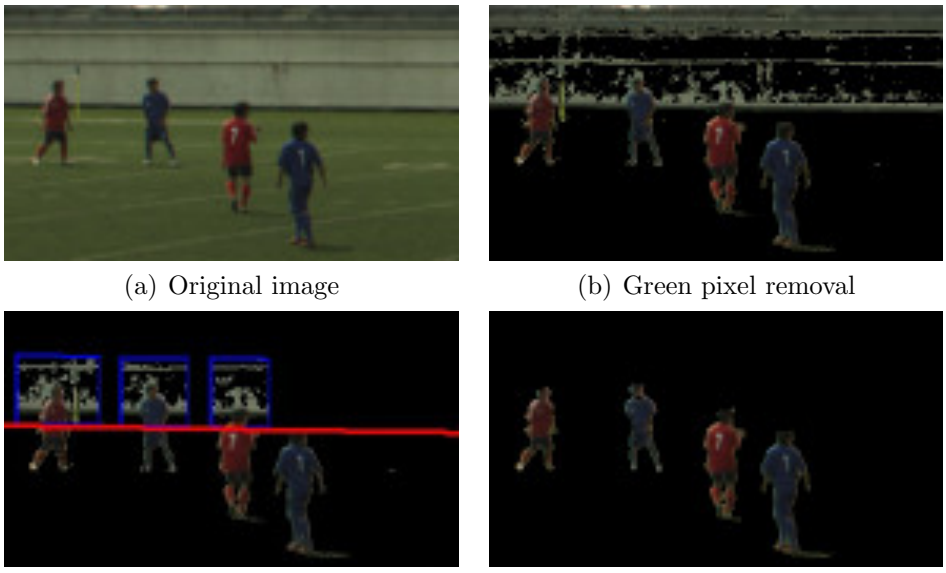
(a) Frame t and Frame $t+1$

Figure 4.6: Corresponding feature points.



(a) Frame t and Frame $t+1$

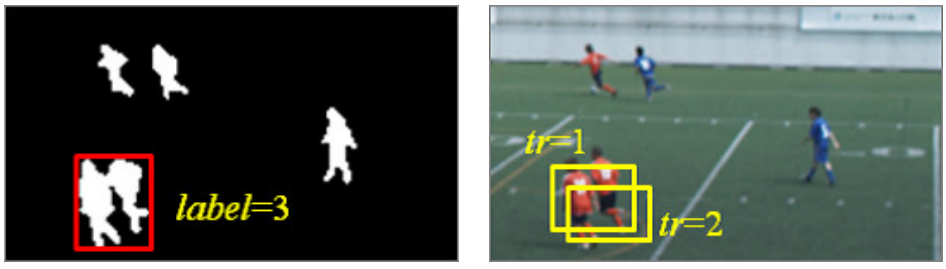
Figure 4.7: Remaining corresponding feature points.



(a) Original image (b) Green pixel removal

(c) Object candidate regions (d) Object textures

Figure 4.8: Examples of extracted object textures.

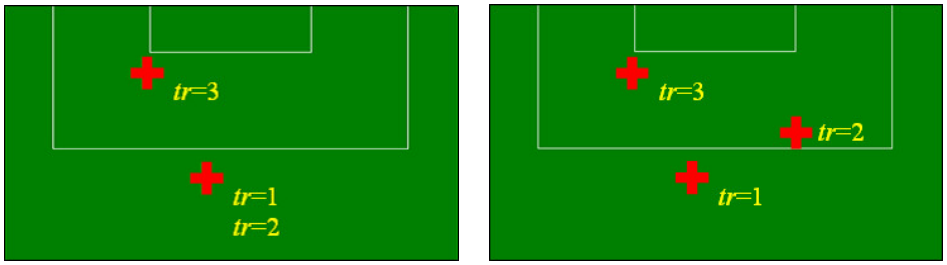


(a) label ID (b) tracker ID

Figure 4.9: Occlusion detections (Camera 1)

(a) label ID (b) tracker ID

Figure 4.10: Non-Occlusion detections (Camera 2)



(a) cam1 (b) cam2

Figure 4.11: XY World coordinate of each object.

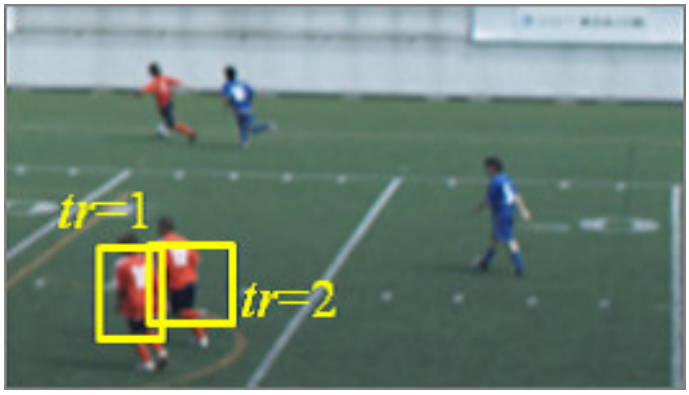


Figure 4.12: Modifiactions of tracker regions

4.4 Experimental Results

In order to evaluate the effectiveness of the proposed method, we conducted four experiments for some kind of video sequences. In the first experiment, our proposed homography matrix estimation method was applied to each video sequence, and the estimation accuracy of the homography matrices was assessed qualitatively and quantitatively. In the second experiment, our proposed object extraction method was applied, and the free-viewpoint video was generated using the texture and the XY-coordinate of each object for every frame. Then, the accuracies of textures and the quality of free-viewpoint video were evaluated in a subjective way. In the third experiment, we evaluated the tracking accuracy compared with the conventional work. In the fourth experiment, we rendered the free viewpoint video using an object region and the XY-coordinate of each object on every camera, which are acquired from the proposed method.

For the first and second experiments, three video sequences (Seq. A, Seq. B, and Seq. C) were prepared. Every sequence was shot by a single moving camera for the scene of soccer game in an outdoor field. The spatial resolution of Seq. A is 4096×2304, and those of Seq. B and Seq. C are 3840×2160. The frame rate of each video sequence is 30 frames / sec, and Seq. A and Seq. B contains 180 frames, while Seq. C contains 90 frames. For every sequence, the initial frame and the last frame are shown in Figure 4.13, Figure 4.14, and Figure 4.15, respectively. For the third and fourth experiments, we prepared two types of sequences (Seq. D and Seq. E) acquired in the outdoor soccer stadium for assessing the differences of tracking accuracy caused by camera arrangements. Seq. D were captured with two fixed cameras that are located at low-level height on the ground as shown in Figure 4.16 (a), and Seq. E were captured with four fixed cameras that are located at high-level height in the stadium stand as shown in Figure 4.16 (b). The spatial resolution and the frame rate of each sequence are 4096×2304 and 30fps respectively, and all the sequences are temporally synchronized. Figures 4.17 and 4.18 show the initial frames in each viewpoint of Seq. D and Seq. E.

4.4.1 Estimation of Homography Matrix

For each video sequence, the homography matrix between the initial frame and the field plane was estimated by manually designating the corresponding feature points. Then, in order to observe the estimation accuracy of the

(a) Frame # 1

(b) Frame # 180

Figure 4.13: Initial and the last frames of Seq. A.

(a) Frame # 1

(b) Frame # 180

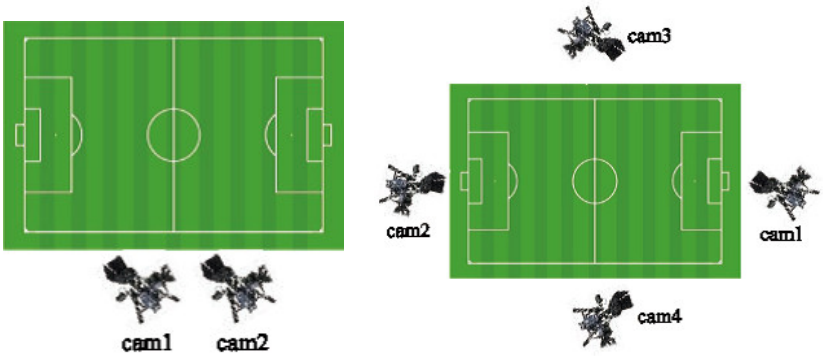
Figure 4.14: Initial and the last frames of Seq. B.

homography matrix in every consecutive frame, the proposed method and the conventional method were applied to each video sequence. For comparison, the conventional homography matrix estimation method using RANSAC-based line detections was implemented by referring to the papers [32] and [35]. In order to evaluate both the proposed method and the conventional method under the same conditions, homography matrices of the initial frames for both methods were manually calculated using feature points in a field plane. The parameters th_c, λ and κ in the energy function of the proposed method were given based on the results of the preliminary experiments. The parameters for the conventional method such as the threshold for detecting court lines were decided in the same manner.

The estimation results by the proposed method are shown in Figure 4.19, where the specific two frames were randomly selected for every sequence to evaluate the variability of accuracy over time. The similar results by the conventional method are also shown in Figure 4.20. The feature lines of the

(a) Frame # 1 (b) Frame # 90

Figure 4.15: Initial and the last frames of Seq. C.



(a) Seq. D (b) Seq. E

Figure 4.16: Camera configurations

field plane were projected onto each frame using the estimated homography matrix for both methods, and each line is represented in red. From these results, in the proposed method, each projected line is fitted to the field line in every frame, while in the conventional method, the differences between the projected line and the field line are significant. In Seq. A, there are two kinds of court lines; the one is white lines for American football game, and the other is yellow lines for soccer game. In the conventional method, both color information was used for line detections, since the number of yellow lines is not enough for homography matrix estimation. On the other hand, court field contains a number of white pixels such as the scale marks and the line-number of American football. As a result, so many false detections degraded the accuracy of estimated homography matrix. On the other hand,

(a) camera 1 (b) camera 2

Figure 4.17: Initial frames of Seq. D

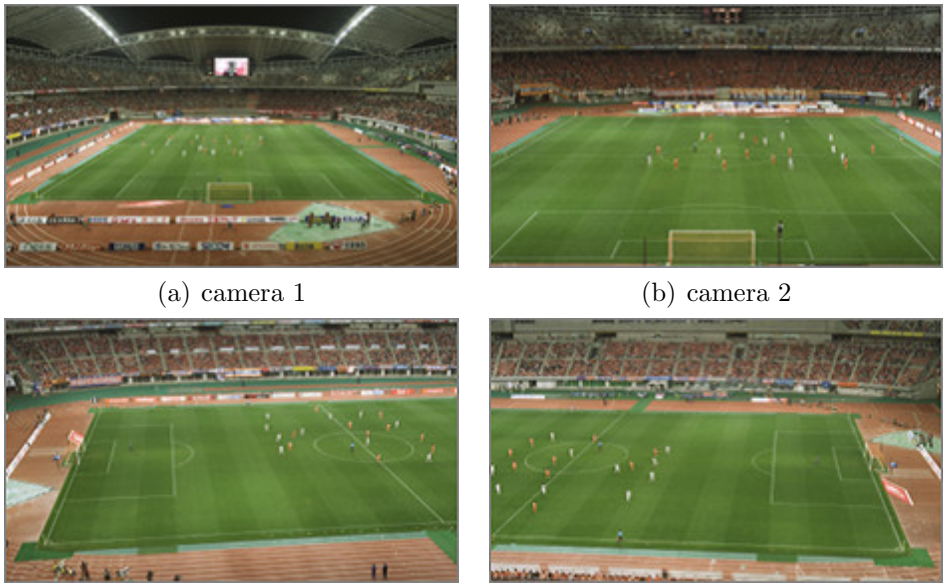
Table 4.1: Comparison of quantitative measurement.

	Sec. A	Sec. B	Sec. C
Proposed method	0.357	0.268	0.294
Conventional method	0.425	0.304	1.149

in Seq. B and Seq. C, white uniform players are detected as false positives, and they degrade the estimation accuracy of homography matrix.

In addition, the quantitative results are evaluated. First, the feature points of specific frames were indicated by manual operations, and each of them was projected with the estimated homography matrix onto the field plane. Then, the distance between the projected point and the real feature point in the world coordinate (scale unit is a meter) was calculated. Finally, the average distance of all the feature points was compared between the proposed method and the conventional method as shown in Table 4.1.

The results demonstrated that the proposed method can estimate the homography matrix more accurately than the conventional method for all the sequences. The estimation accuracy of the homography matrix has a huge effect on the quality of object extraction, since the textures of unwanted regions such as white lines and spectator stands has to be eliminated by projecting each line to the current frame. Experientially, it is desired that the estimation error should be limited to 1.0 meter or less to extract object textures with sufficient quality for the soccer videos, and Table 1 shows that all the results of the proposed method satisfy the criteria.



(a) camera 1 (b) camera 2

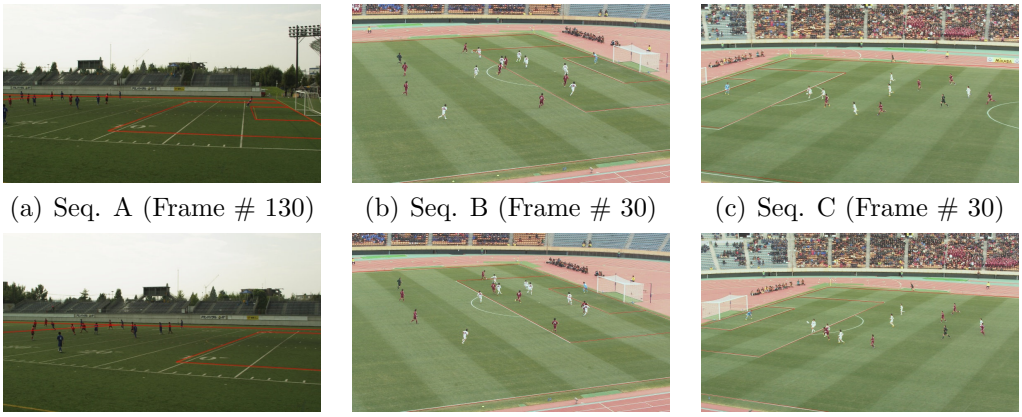
(c) camera 3 (d) camera 4

Figure 4.18: Initial frames of Seq. E

4.4.2 Object Extraction and Synthesis of Free-viewpoint Video

For each video sequence, objects' textures of every frame were extracted by applying the proposed method. The object extraction results by the proposed method are shown in Figure 4.21. In order to evaluate the effectiveness of energy minimization process in the proposed method, the comparative method was implemented. In the comparative method, objects' textures were extracted as the candidate regions like Figure 4.8 (c), based on the homography matrix estimated by the conventional method in Section 4.4.1. The results by the conventional scheme were also shown in Figure 4.22.

From these results, it is confirmed that each object texture is extracted precisely for every frame in the proposed method. On the other hand, the results by the comparative method include some unwanted regions such as lines and spectators' regions. Specifically, in Seq. A, textures of unwanted regions such as the spectator stand and lines are different from those of players' uniforms, and therefore the segmentation of unwanted textures from candidate regions work well. In Seq. B and Seq. C, segmentation of white lines from



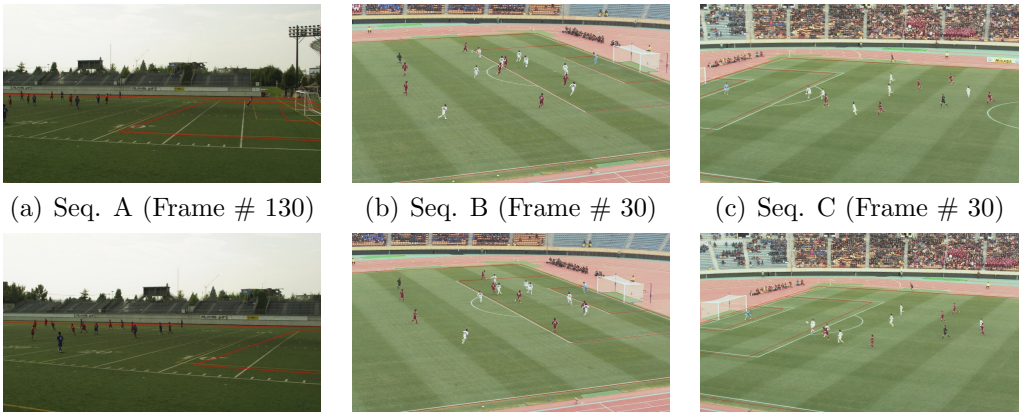
(a) Seq. A (Frame # 130) (b) Seq. B (Frame # 30) (c) Seq. C (Frame # 30)

(d) Seq. A (Frame # 160) (e) Seq. B (Frame # 60) (f) Seq. C (Frame # 60)

Figure 4.19: Estimation results of the proposed method.

the candidate region is very difficult, since the color difference between white lines and players with white uniform is small. In the proposed method, the candidate rectangle region including some players with white uniform and a part of white lines can be detected, and the unwanted areas composed of white lines could be removed precisely by considering the relation among adjacent pixels.

Then, the free-viewpoint video of each video sequence was rendered based on billboard model constructed using both a homography matrix and an object texture acquired by applying the proposed method for every frame. As described in Section 4.3.5, each extracted bounding box includes more than two objects, and the texture and the 2D world coordinate of an occluded object cannot be precisely calculated for occluded regions. However, when occlusions do not continue over a long time frame, artifacts in synthesized video are not so serious for subjective image quality. Therefore, in this Section, the bounding box itself for an occluded region is used to reconstruct the billboard model. For the selected two virtual viewpoints, the free-viewpoint video was generated, and the resultant images of a certain frame are shown in Figure 4.23. Comparative results synthesized using both a homography matrix and an object texture got from the conventional method are also shown in Figure 4.24. From these results, it was confirmed that the positional relationships among objects in a 3D space and appearances were correctly reproduced using the proposed method. On the other hand, from the re-



(a) Seq. A (Frame # 130) (b) Seq. B (Frame # 30) (c) Seq. C (Frame # 30)

(d) Seq. A (Frame # 160) (e) Seq. B (Frame # 60) (f) Seq. C (Frame # 60)

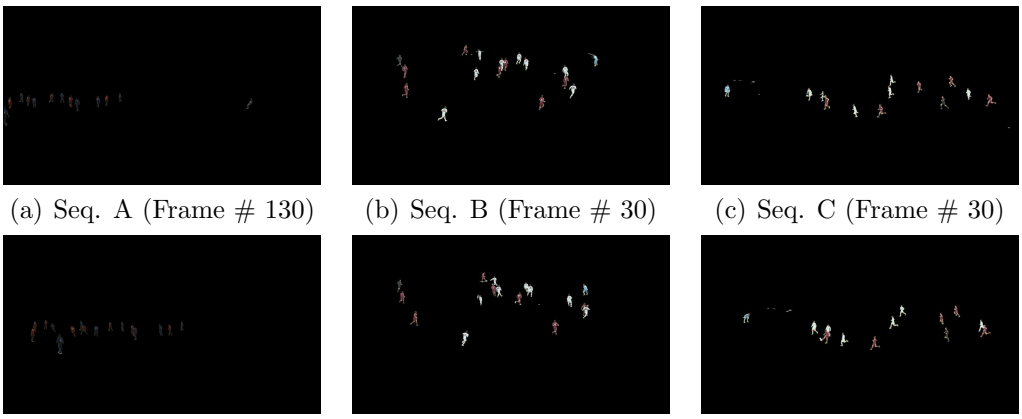
Figure 4.20: Estimation results of the conventional method.

sult by the comparative method, objects including the outlier texture (i.e. white lines and a spectator's stand) were represented as one closed region, and positional relationships and appearances were not precisely reproduced. Furthermore, Figure 4.23(e) shows that extracted player textures do not include unwanted regions, and are so high-definition that the uniform numbers can be recognized clearly if the virtual viewpoint was set close to the players on the field plane. These results show that the proposed algorithm for homography matrix estimation and object extraction for a moving camera improved the subjective image quality in rendering free-viewpoint video.

4.4.3 Object Tracking

In order to assess the tracking accuracy of the proposed method, we used 70 consecutive frames of Seq. D and 630 frames of Seq. E. In Seq. D, tracking targets were divided into two classes such as the red uniform team and the blue uniform team (one goal keeper is treated as having the blue uniform), and in Seq. E, they were divided into five classes such as the orange uniform team, the green uniform keeper, the white uniform team, the black uniform keeper, and the light blue uniform referees. As a conventional scheme, the simple particle filtering method that uses only single image information and does not use occlusion information was also evaluated for comparison.

For the quantitative evaluation, we also introduce a new term *gmme* for



(a) Seq. A (Frame # 130) (b) Seq. B (Frame # 30) (c) Seq. C (Frame # 30)

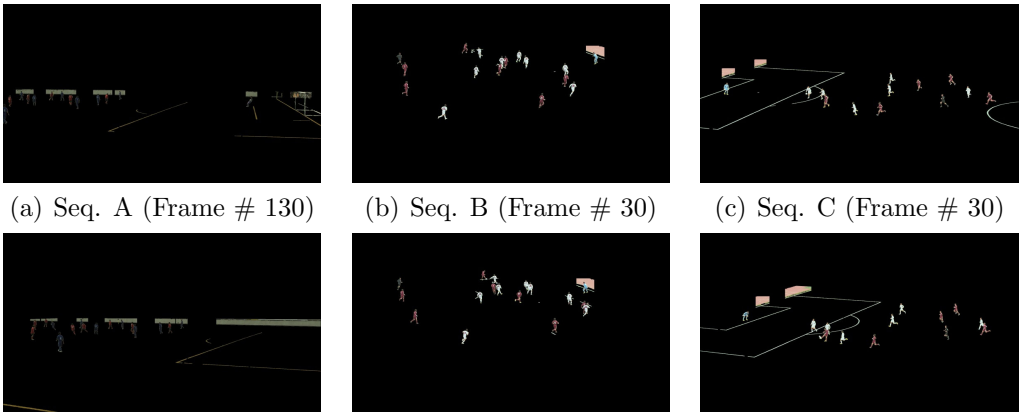
(d) Seq. A (Frame # 160) (e) Seq. B (Frame # 60) (f) Seq. C (Frame # 55)

Figure 4.21: Extracted textures of the proposed method.

measuring the proportion of identity switches in a global manner as well as the paper [29]. For every detection at every frame, the term gmme is incremented if the tracking number does not correspond to the ground truth identifier gt_t as shown in equation (4.15).

$$gmme = \frac{\sum_{t=1}^{T} gmme_t}{\sum_{t=1}^{T} gt_t} \times 100 \tag{4.15}$$

The tracking results in some frames of each method for Seq. D and Seq. E are shown in Figures from 4.25 to 4.30. Figures 4.25 and 4.28 show the results of the proposed method, Figures 4.26 and 4.29 show those of the conventional method 1, and Figures 4.27 and 4.30 show those of the conventional method 2. As shown in these figures, the proposed method tracks each object precisely for both sequences, while the conventional methods fail to track some objects. Figures from 4.28 to 4.30 confirm that even in the case of occlusions caused from more than two objects at long intervals, the proposed method continues to track each object consistently among multiple cameras. If the proposed method fails to track some objects for some frames, it can recover based on the other camera. On the other hand, it is difficult for the conventional method to recover from miss-detections or tracker number switches, and therefore, it often fails to track each object after occlusion occurs.



(a) Seq. A (Frame # 130) (b) Seq. B (Frame # 30) (c) Seq. C (Frame # 30)

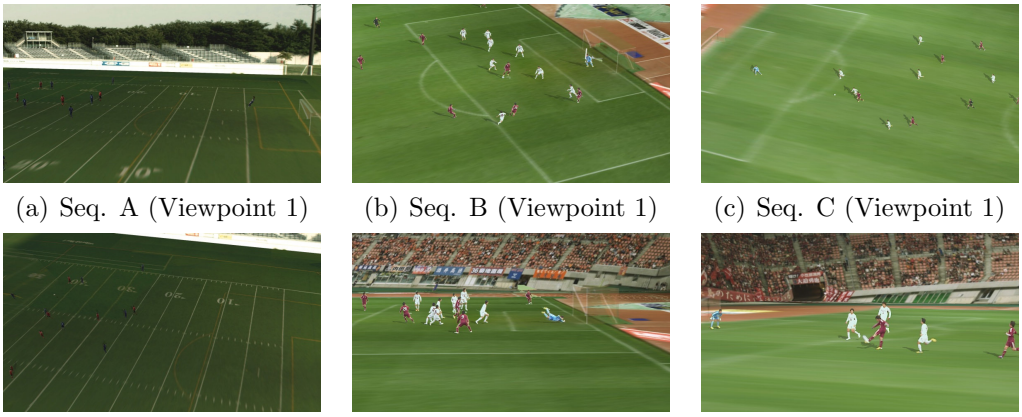
(d) Seq. A (Frame # 160) (e) Seq. B (Frame # 60) (f) Seq. C (Frame # 55)

Figure 4.22: Extracted textures of the conventional method.

Table 4.2: Comparison of quantitative measurement.

	Seq. D	Seq. E
Proposed	0	1.617
Conventional 1	6.875	9.265
Conventional 2	0	16.548

Furthermore, the term *gmme* for Seq. D and Seq. E are shown in Table 4.2. The results show the similarity of instantaneous identity switches, while the differences of false negatives and false positives are relatively higher. By contrast, for the new metric gmme, the performance difference between the proposed method and the conventional method is larger than any other metrics mentioned above. In the proposed method, there were no identity switches and therefore, the values of fn and gmme are the same; while in the conventional method, such cases were observed, and once it occurred, tracking recovery could not be realized. This means that the proposed method preserves the identifier of each object for long intervals, and is robust for occlusions and recovery after miss-tracking can be achieved correctly using the other camera. On the other hand, the performance differences between both methods for Seq. E are comparatively higher than those of Seq. D, and this indicates that the performance of the proposed method depend on the



(a) Seq. A (Viewpoint 1) (b) Seq. B (Viewpoint 1) (c) Seq. C (Viewpoint 1)

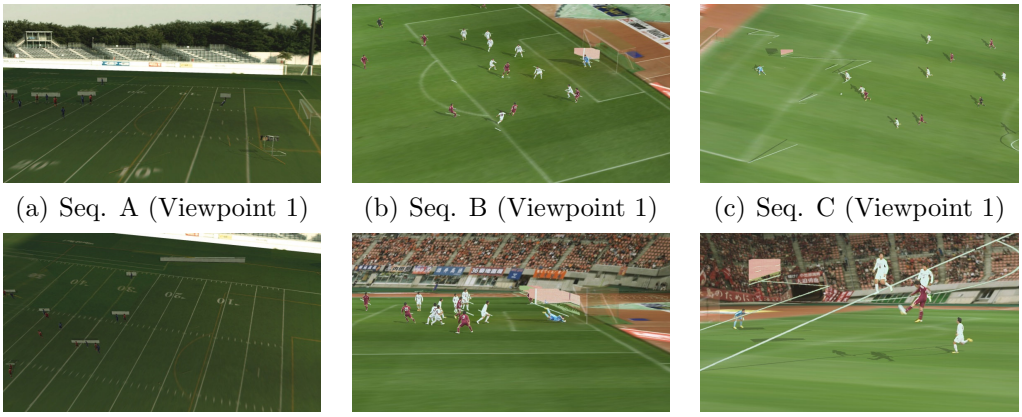
(d) Seq. A (Viewpoint 2) (e) Seq. B (Viewpoint 2) (f) Seq. C (Viewpoint 2)

Figure 4.23: Free-viewpoint video of the proposed method.

camera arrangements such as the number of viewpoints. These results are expected to improve the image quality of the free-viewpoint video, especially in the case of frequent occlusion occurrence.

4.4.4 Free-viewpoint Video Rendering based on Object Tracking

The free-viewpoint video of Seq. E was rendered using the texture and the XY-coordinate of each object among the multiple cameras of the proposed method. For every frame, we reconstruct the billboard model of each object based on the XY-coordinate, and the texture size acquired from the tracking results. In the case of occlusions, the XY-coordinates and the textures of objects are interpolated between a non-occluded texture of the same tracking ID in the last-minutes frame and the texture in the immediate frame. As a comparative method 1 and comparative method 2, the texture was interpolated based on the tracking results of conventional method 1 and conventional method 2 in Section 4.4.3. As a comparative method 3, we also render free-viewpoint video using texture regions that are obtained from simple background subtraction based on equation (4.2) and XY-coordinates estimated from the foot positions of these texture regions. In this method, each object cannot be separated in occlusion regions, and then occluded objects are considered as one object whose foot position is treated as that of



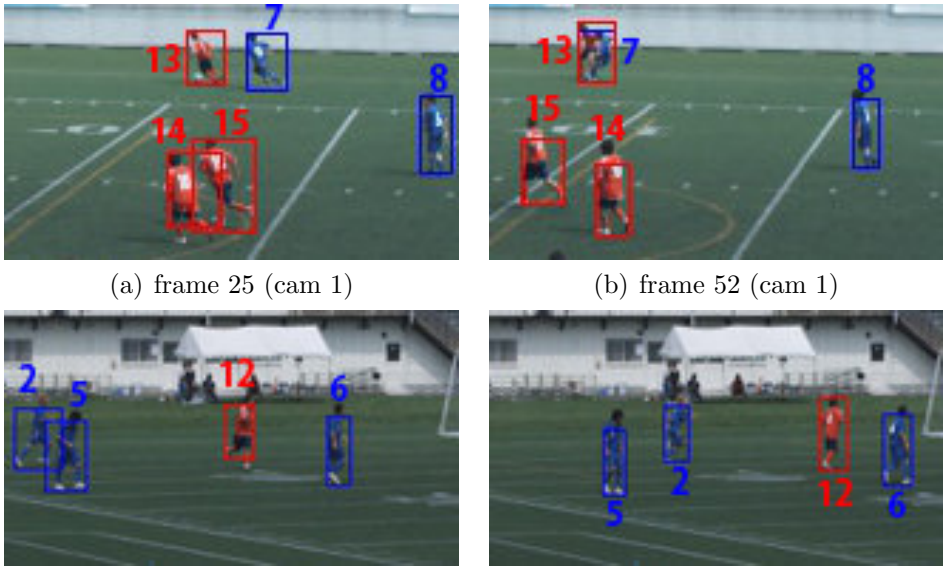
(a) Seq. A (Viewpoint 1) (b) Seq. B (Viewpoint 1) (c) Seq. C (Viewpoint 1)

(d) Seq. A (Viewpoint 2) (e) Seq. B (Viewpoint 2) (f) Seq. C (Viewpoint 2)

Figure 4.24: Free-viewpoint video of the conventional method.

the former object.

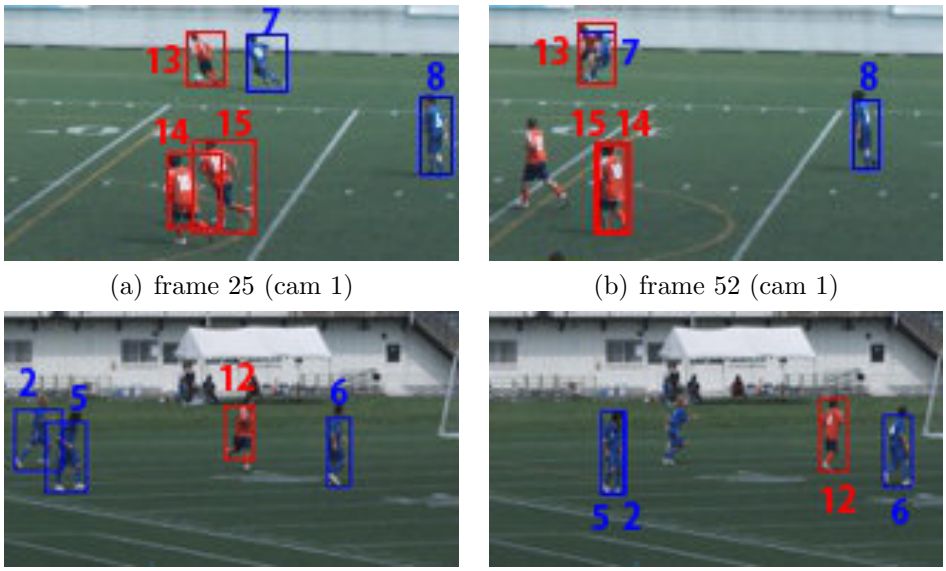
The examples of the virtual viewpoints in a certain frame rendered by each method are shown in Figures from 4.31 to 4.34. In order to compare the synthesized results with real camera images, closeup of each camera for the same scene is shown in Figure 4.35. From these results, in the comparative methods, the occlusion regions that consist of two objects are rendered as one object respectively, and the positional relationships between these objects are not reconstructed correctly. On the other hand, in the proposed method, the positional relationships and the appearances including shadows are correctly reproduced, and it improves the subjective image quality of the rendered free-viewpoint video especially in occlusion regions.



(a) frame 25 (cam 1) (b) frame 52 (cam 1)

(c) frame 8 (cam 2) (d) frame 47 (cam 2)

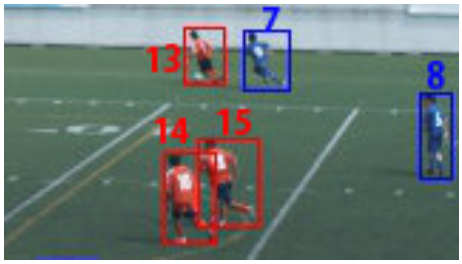
Figure 4.25: Results of the proposed method for Seq. D.



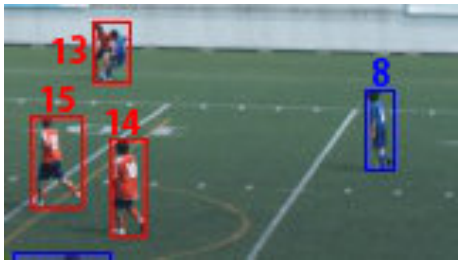
(a) frame 25 (cam 1) (b) frame 52 (cam 1)

(c) frame 8 (cam 2) (d) frame 47 (cam 2)

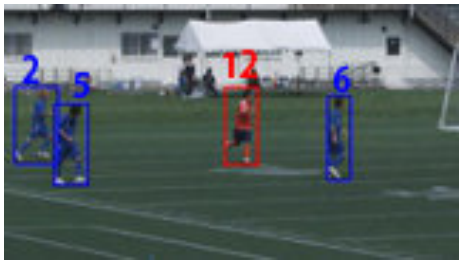
Figure 4.26: Results of the conventional method 1 for Seq. D.



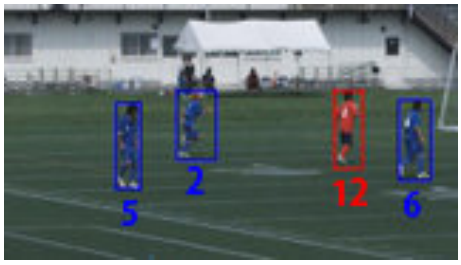
(a) frame 25 (cam 1)



(b) frame 52 (cam 1)

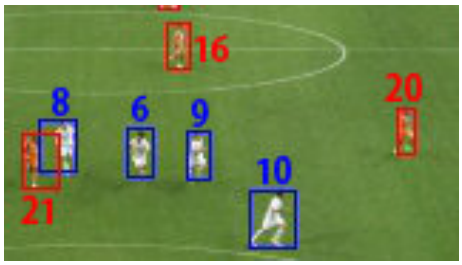


(c) frame 8 (cam 2)



(d) frame 47 (cam 2)

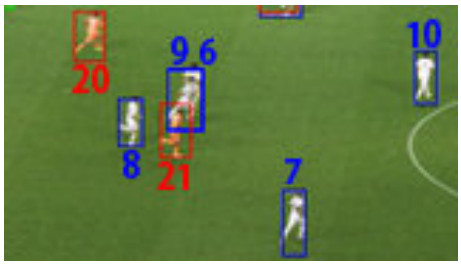
Figure 4.27: Results of the conventional method 2 for Seq. D.



(a) frame 230 (cam 1)

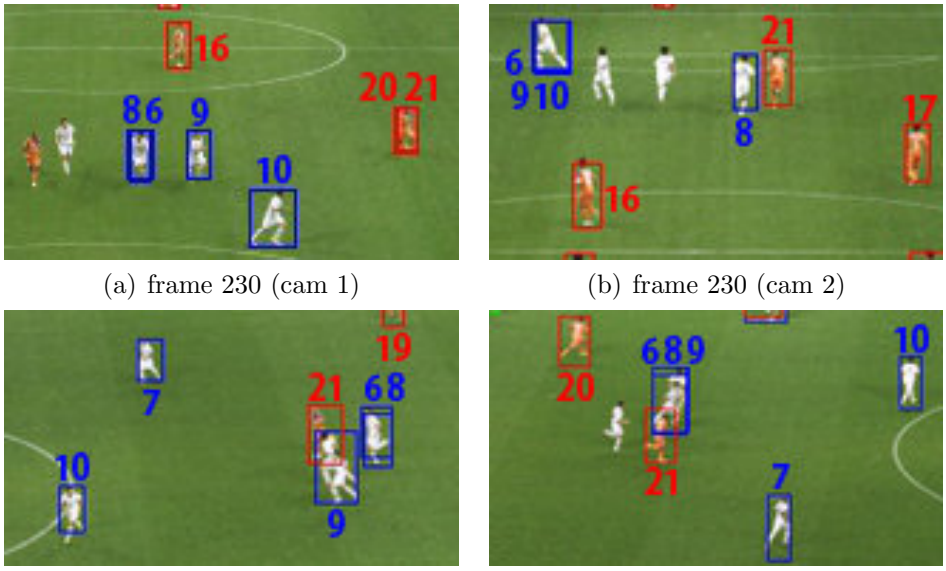
(b) frame 230 (cam 2)

(c) frame 230 (cam 3)



(d) frame 230 (cam 4)

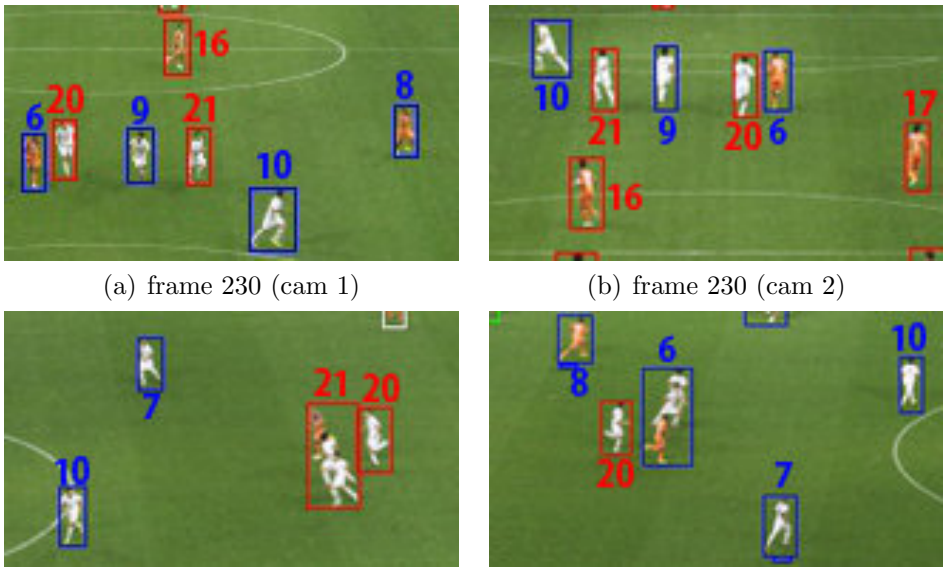
Figure 4.28: Results of the proposed method for Seq. D.



(a) frame 230 (cam 1) (b) frame 230 (cam 2)

(c) frame 230 (cam 3) (d) frame 230 (cam 4)

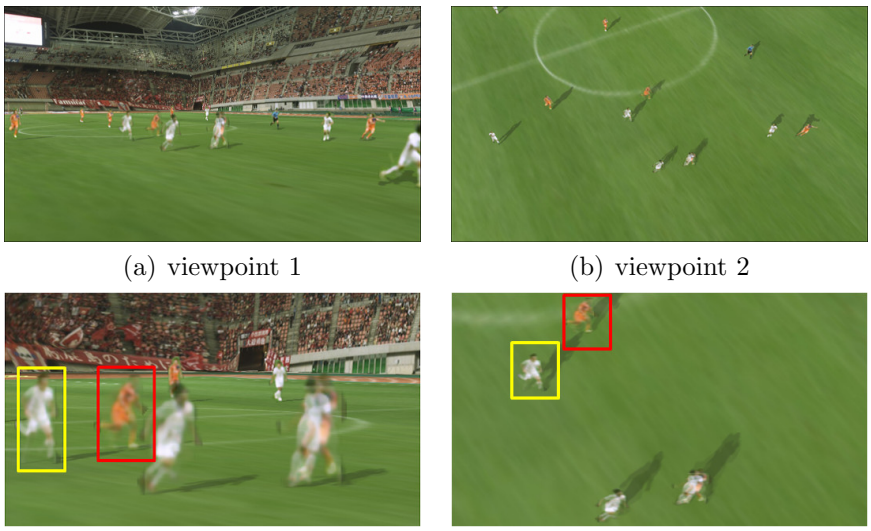
Figure 4.29: Results of the conventional method 1 for Seq. E.



(a) frame 230 (cam 1) (b) frame 230 (cam 2)

(c) frame 230 (cam 3) (d) frame 230 (cam 4)

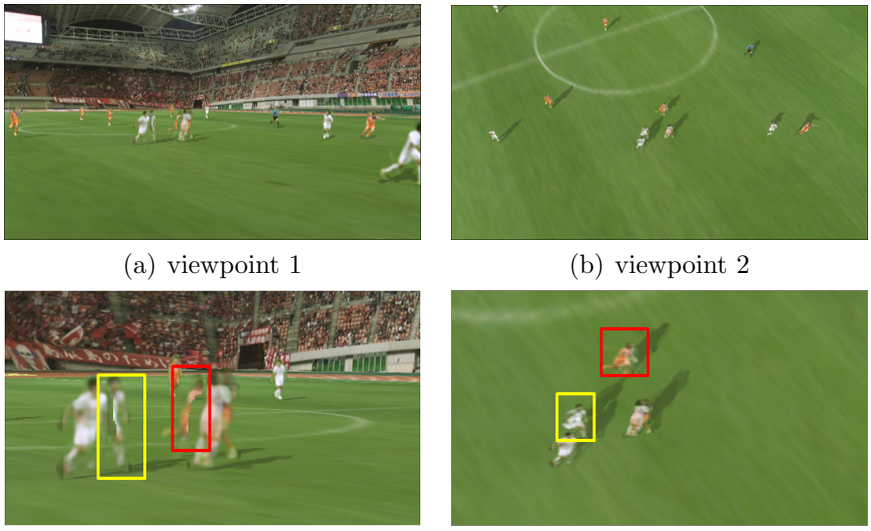
Figure 4.30: Results of the conventional method 2 for Seq. E.



(a) viewpoint 1 (b) viewpoint 2

(c) zooming up of viewpoint 1 (d) zooming up of viewpoint 2

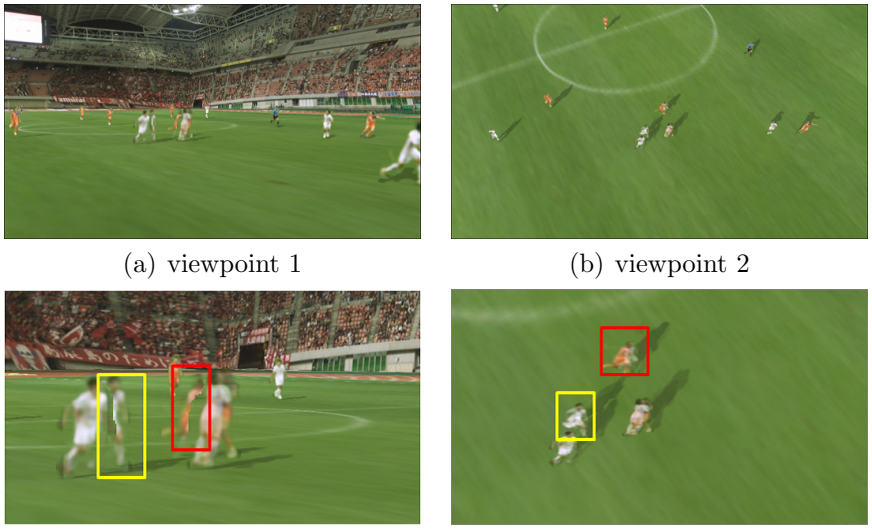
Figure 4.31: Free-viewpoint video of Seq. E rendered by the proposed method



(a) viewpoint 1 (b) viewpoint 2

(c) zooming up of viewpoint 1 (d) zooming up of viewpoint 2

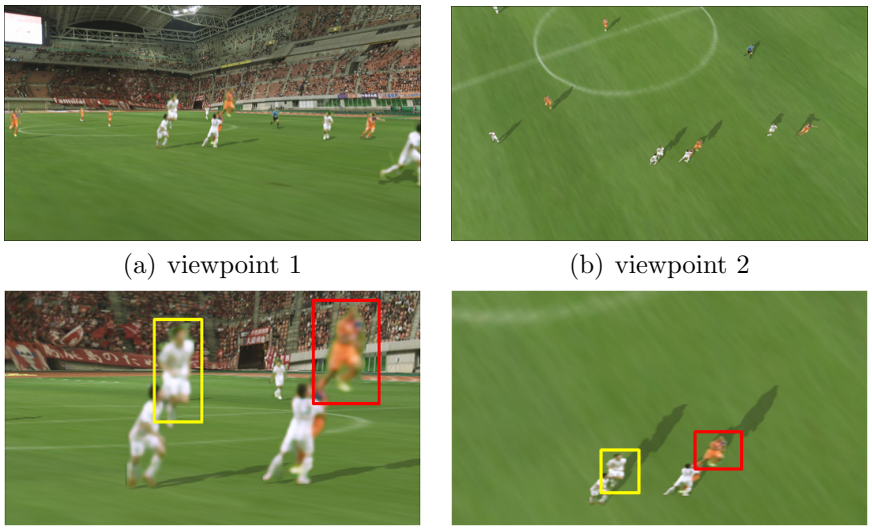
Figure 4.32: Free-viewpoint video of Seq. E rendered by the comparative method 1.



(a) viewpoint 1 (b) viewpoint 2

(c) zooming up of viewpoint 1 (d) zooming up of viewpoint 2

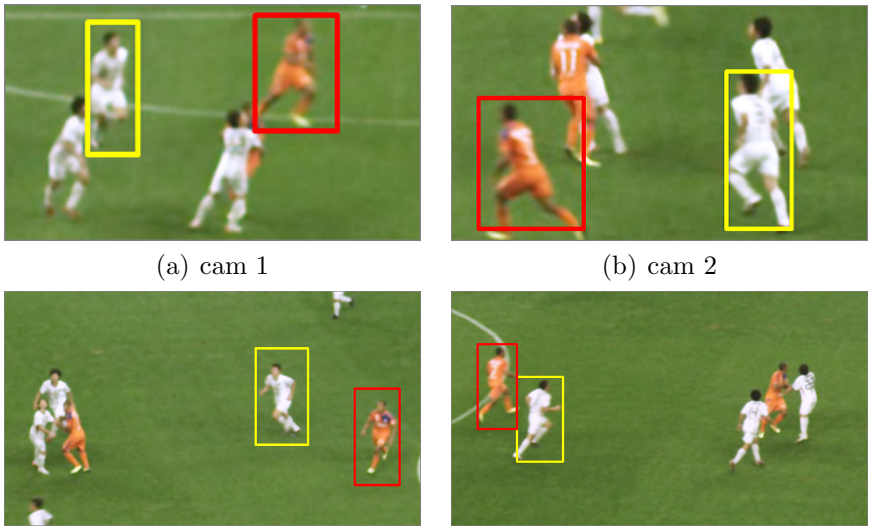
Figure 4.33: Free-viewpoint video of Seq. E rendered by the comparative method 2.



(a) viewpoint 1 (b) viewpoint 2

(c) zooming up of viewpoint 1 (d) zooming up of viewpoint 2

Figure 4.34: Free-viewpoint video of Seq. E rendered by the comparative method 3.



(a) cam 1 (b) cam 2

(c) cam 3 (d) cam 4

Figure 4.35: Closeup of each camera image for Seq. E

4.5 Conclusions

In this chapter, we propose a homography matrix estimation method and an object extraction method using a moving camera for realizing an immersive free-viewpoint experience in sport videos. Our proposed method estimates homography matrices based on the feature point matching between video frames. Furthermore, the method extracts objects' texture regions based on the estimated homography matrices. Experimental results revealed that the proposed method can achieve more accurate estimation of the homography matrices compared to the conventional method. Moreover, it was also confirmed that the proposed scheme contributes to further improvement in the experience of free-viewpoint video since the textures of multiple objects were successfully extracted.

Furthermore, to realize highly precise free-viewpoint video in wide outdoor spaces such as soccer stadiums, we proposed a robust multiple object tracking and extracting method using the integrated information of multi-view video sequences. In order to extract the texture of an occluded object in a certain camera which cannot be extracted precisely because of non-visibility, the object's texture has to be extracted from another frame in which the object is not occluded from the other objects. Each object is thus required to be identified consistently among all cameras for every frame to share textures of the same objects and replace them in the event that occlusion occurs. As an inherent problem, the conventional schemes for multiple object tracking and extracting methods do not require precisely identified texture regions, and those methods only extract a rough region such as a rectangular area of each object which includes many false positives and false negatives. In order to overcome these problems, the proposed scheme tracks and extracts objects in each viewpoint by keeping the object identifier among multiple cameras, and checks whether or not object occlusion occurs for every frame. When the occlusion is caused by some objects in a certain viewpoint, the method detects the other camera in which those objects are not occluding and occluded, and modifies the object regions and XY-coordinates based on those in the camera. From the experimental results using the actual two types of multi-view video sequences, it was confirmed that the proposed method contributed to significant improvement for tracking and extracting accuracy compared with the conventional method. Furthermore, it was also confirmed that the free-viewpoint video was rendered precisely while the occluded regions were reconstructed successfully.

As future works, the introduction of a process that reduces the influence of estimated error of projection matrices is required.

Chapter 5

Conclusions and Future Works

This thesis discusses a framework of object extraction to tackle occlusion regions for virtual-viewpoint video synthesis, which is important to realize ultra-realistic experiences such as Free-viewpoint video and Tele-presence. The main contribution of the thesis is to introduce a framework of object extraction methods based on temporal and/or spatial characteristics of the target scene to handle occlusion regions for virtual-viewpoint video synthesis, and propose an approach of object extraction for each object representation: 3D object model, 2.5D depth map, and 2D billboard. A framework of object extraction methods consists of the following three steps. The first one is extracting silhouettes for object regions based on the space characteristics of the coordinate system in which each object representation is formulated. A common approach to refine silhouettes by estimating background regions of the target scene is introduced for all the representations. The second one is detecting occlusion regions based on positional relationships of objects in a 3D space and motion estimation of objects between frames by taking account of the consistency between the extracted silhouettes and the object representation integrating multiple camera information. And the third one is acquiring visual texture of an object surface based on corresponding regions matching among multiple cameras and/or among consecutive frames by taking the characteristics of a virtual-viewpoint video rendering algorithm for each object representation into consideration.

A typical approach to get a 3D object model is shape from silhouettes algorithm. The chapter 2 presents an approach of object extraction to tackle occlusion problems for 3D object model based on the above mentioned framework and the utilizing information, that is, correspondence relationship be-

tween an arbitrary 3D coordinate in object-centered coordinate system and 2D pixel coordinate of every camera. Our method uses an approach for integrating multi-view images in which the background region is determined using voxel information rather than each camera image itself. We introduce a likelihood of background to each pixel of camera images, and derive integrated likelihood in the voxel space. The background region is determined on the basis of minimization of energy functions of the likelihood. Furthermore, the proposed method also applies a robust refining process, in which each silhouette is modified on the basis of projections of a 3D-model to each viewpoint and a 3D-model is reconstructed using modified silhouettes. Experimental results show the proposed method to be more effective than the existing methods.

For 2.5D Depth representation, an image quality in edge regions such as hair may degrade. Furthermore, the virtual viewpoint includes dis-occlusion areas whose textures do not exist in real camera. In the chapter 3, we propose an an approach of object extraction to overcome occlusion problems for 2.5D depth map based on the above mentioned framework and the utilizing information, that is, correspondence relationship between an arbitrary 3D coordinate in viewer-centered coordinate system and 2D pixel coordinate of the neighboring camera. Our proposed method is based on interpolation and extrapolation of depth information in edge regions. In addition, a virtual view synthesis method is also proposed based on tracking 3D regions between consecutive frames. Experimental results showed the effectiveness of the proposed method regarding the image quality of virtual viewpoints. Furthermore, it was confirmed that the experience of depth perception, eye contact, and motion parallax for head movement could be naturally realized.

In regard to 2D billboard, each object is represented as a set of 2D slice silhouettes and visual textures extracted from all the cameras, and the object position can be calculated in every frame based on the 2D coordinate of the bottom line of the silhouette region and homography matrix between captured frame and 2-dimensional world coordinate model of the target space. The chapter 4 presents an approach of object extraction to solve occlusion problems for 2D billboard based on the above mentioned framework and the utilizing information, that is, correspondence relationship between an arbitrary 2D pixel coordinate in every camera and 2D world coordinate of the specific plane in the target space. Our proposed method estimates homography matrices semi-automatically by identifying reliable corresponding feature points between video frames, and also extracts the precise object

regions using estimated homograph matrices. Furthermore, we propose a robust object tracking scheme among multi-view cameras and consecutive frames for rendering an immersive free-viewpoint video in a large outdoor space such as a soccer stadium. Experimental results revealed that the proposed method achieved more robust texture extraction of multiple objects especially for occluded scenes compared to the conventional methods. Furthermore, it was also confirmed that the proposed scheme can improve the experience of free-viewpoint video as the result of precise reconstruction of occluded regions.

As future works, we need to introduce a process that reduces the influence of estimated error of projection matrixes, which is intimately related all the proposed modeling method in the thesis. For 3D object model, the accuracy of the estimated projection matrix for each camera has huge impact on the quality of reconstructed 3D shape, and some refinement processes to absorb the error of projection matrix for each camera is very important. For 2.5D depth map, the algorithm that reduces correspondence relationship error between color image and depth image is essential for synthesizing high-quality virtual-viewpoint images. Finally, for 2D billboard, the smoothness of world 2D coordinates of each object for consecutive frames has to be refined, since the foot position estimation is not accurate due to the pixel wise errors of background subtraction and homography matrix estimation.

Bibliography

[1] T. Kanade, P. W. Rander, and P. J. Narayanan, "Virtualized Reality: Constructing Virtual Worlds from Real Scenes," IEEE Multimedia, vol. 4, no. 1, pp. 34-47, 1997.

[2] C. Zhang and T. Chen, "A Survey on Image-based Rendering - Representation, Sampling and Compression," Signal Processing: Image Communication, vol. 19, no. 1, pp. 1-28, 2004.

[3] H. Y. Shum, S. C. Chan, and S. B. Kang, "Image-Based Rendering," Springer, 2008.

[4] T. Matsuyama, X. Wu, T. Takai, and T. Wada, "Real-Time Dynamic 3D Object Shape Reconstruction and High-FidelityTexture Mapping for 3D Video," IEEE Trans. on Circuits and Systems for Video Technology, Vol. CSVT-14, No. 3, pp. 357-369, 2004.

[5] J. Starck, and A. Hilton, "Surface Capture for Performance-Based Animation," IEEE Computer Graphics and Applications, Vol. 27, No. 3, pp. 21-31, 2007.

[6] T. Kanade, "Eye Vision," `http://www.ri.cmu.edu/events/sb35/tksuperbowl.html`

[7] Y. Ohta, I. Kitahara, Y. Kameda, H. Ishikawa, and T. Koyama, "Live 3D Video in Soccer Stadium," International Journal of Computer Vision (IJCV), vol. 75, no. 1, pp. 173-187, 2007.

[8] K. Hayashi, and H. Saito, "Synthesizing Free-viewpoint Images from Multiple View Videos in Soccer Stadium," In Proc. IEEE Conference Computer Graphics, Imaging and Visualization (CGIV 2006), pp. 220-225, 2006.

[9] M. Germann, A. Hornung, R. Keiser, R. Ziegler, S. Wurmlin, and M. Gross, "Articulated Billboards for Video-based Rendering," In Proc. EUROGRAPHICS, pp. 585-594, 2010.

[10] T. Fujii and M. Tanimoto, "Free Viewpoint TV System based on Ray-space Representation," in Proc. ITCom 2002: The Convergence of Information Technologies and Communications, pp. 175-189, 2002.

[11] W. N. Martin and J. K. Aggarwal, "Volumetric Description of Objects from Multiple Views," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 5, no. 2, pp. 150-158, 1983.

[12] W. E. Lorensen and H. E. Cline, "Marching Cubes: A High Resolution 3d Surface Construction Algorithm," In Proc of ACM SIGGRAPH, vol.21, no.4, pp.163-169, 1987.

[13] M. Tanimoto, M. P. Tehrani, T. Fujii, T. Yendo, "Free-viewpoint TV," IEEE Signal Processign Magazine, vol. 28, no. 1, pp. 67-76, 2011.

[14] A. Ishikawa, M. P. Tehrani, S. Naito, S. Sakazawa, and A. Koike, "Free Viewpoint Video Generation for Walk-through Experience using Image-based Rendering," In Proc of the 16th ACM international conference on Multimedia, pp.1007-1008, 2008.

[15] N. Inamoto and H. Saito, "Virtual Viewpoint Replay for a Soccer Match by View Interpolation from Multiple Cameras," IEEE trans. Multimedia, vol.9, no.6, pp.1155-1166, 2007.

[16] Y. Y. Boykov, M. P. Jolly, "Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images," IEEE conference on Computer Vision, CD-ROM.

[17] C. Rother, V. Kolmogorov, A. Blake, "GrabCut: Interactive Foreground Extraction using Iterated Graph Cuts," In Proc of ACM SIGGRAPH, vol. 23, pp. 309-314, 2004.

[18] G. K. M. Cheung, T. Kanade, J. Y. Bouguet, and M. Holler, "A Real Time System for Robust 3d Voxel Reconstruction of Human Motions," IEEE conference on Computer Vision and Pattern Recognition, vol. 2, pp. 714-720, 2000.

[19] J. L. Landabaso, M. Pardas, and J. R. Casas, "Shape from Inconsistent Silhouette," Int. Journal of Computer Vision and Image Understanding, vol. 112, no. 2, pp. 210-224, 2008.

[20] G. Zeng and L. Quan, "Silhouette Extraction from Multiple Images of an Unknown Background," Asian Conference on Computer Vision, pp.628-633, 2004.

[21] K. N. Kutulakos and S. M. Seitz, "A Theory of Shape by Space Carving," Int. J. Comput. Vis., vol. 38, no. 3, pp. 192-218, 2000.

[22] K. Hisatomi, K. Tomiyama, M. Katayama, and Y. Iwadate, "Method of 3D reconstruction using graph cuts and its application to preserving intangible cultural heritage," Proc. IEEE Conference on Computer Vision, pp. 923 - 930, 2009.

[23] S. M. Seitz, and C. R. Dyer, "View Morphing," In Proc. ACM SIGGRAPH, pp. 21-30, 1996.

[24] K. N. Kutulakos, and S. M. Seitz, "A Theory of Shape by Space Carving," International Journal of Computer Vision (IJCV), Vol. 38, No. 3, pp. 192-218, 2000.

[25] H. Sankoh, A. Ishikawa, S. Naito, and S. Sakazawa, "Robust Background Subtraction Method based on 3d Model Projections with Likelihood," In Proc. IEEE International workshop on Multimedia Signal Processing (MMSP 2010), pp. 171-176, 2010.

[26] N. Inamoto, and H. Saito, "Virtual Viewpoint Replay for a Soccer Match by View Interpolation from Multiple Cameras," IEEE trans. Multimedia, Vol. 9, No. 6, pp. 1155-1166, 2007.

[27] H. Sankoh and S. Naito,"Free-viewpoint Video Rendering in Large Outdoor Space such as Soccer Stadium based on Object Extraction and Tracking Technology," The Journal of The Institute of Image Information and Television Engineers (ITE), Vol. 68, No. 3, pp. J125-J134, 2014.

[28] M. Breitenstein, F. Reichin, B. Leibe, E. Koller-Meier, and L. V. Gool: "Robust Tracking-by-detection using a Detector Confidence Particle Filter", IN Proc of IEEE Conference on ICCV, pp. 1515-1522 (2009).

[29] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua: "Tracking Multiple People under Global Appearance Constraints", In Proc of IEEE Conference on ICCV (2011).

[30] D. Farin, S. Krabbe, P. H. N. de With, and W. Effelsberg, "Robust Camera Calibration for Sport Videos using Court Models," In Proc. Storage and Retrieval Methods and Applications for Multimedia (SPIE), Vol. 5307, pp. 80-91, 2004.

[31] D. Farin, J. Han, P. H. N. de With, "Fast camera calibration for the analysis of sport sequences," In Proc. IEEE International Conference on Multimedia and Expo (ICME 2005), 2005.

[32] M-C. Hu, M-H. Chang, J-L. Wu, and L. Chi, "Robust Camera Calibration and Player Tracking in Broadcast Basketball Video," IEEE Trans. Multimedia, Vol. 13, No. 2, pp. 266-279, 2011.

[33] C-C. Hsu, H-T. Chen, C-L. Chou, and S-Y. Lee, "Spiking and Blocking Events Detection and Analysis in Volleyball Videos," In Proc. IEEE International Conference on Multimedia and Expo (ICME 2012), pp. 19-24, 2012.

[34] A. Hilton, J-Y. Guillemaut, J. Kilner, O. Grau, and T. Graham, " 3D-TV Production from Conventional Cameras for Sports Broadcast," IEEE trans. Broadcasting, Vol. 57, No. 2, pp. 462-476, 2011.

[35] J. Han, D. Farin, and P. H. N. de With, "Broadcast Court-Net Sports Video Analysis Using Fast 3-D Camera Modeling," IEEE trans. Circuits and Systems for Video Technology (TCSVT), Vol. 18, No. 11, pp. 1628-1638, 2008.

[36] A. Mittal and L. Davis: "M2tracker: A Multi-view Approach to Segmenting and Tracking People in a Cluttered Scene", IJCV, **51**, 3, pp. 189-203 (2003).

[37] C. Yang, R. Duraiswami, and L. Davis: "Fast Multiple Object Tracking via a Hierarchical Particle Filter", In Proc of IEEE Conference on ICCV (2005).

[38] K. Okuma, A. Taleghani, N. D. Freitas, J. J. Littele, and D. G. Lowe: "A boosted Particle Filter: Multi Target Detection and Tracking", In Proc of ECCV, pp. 28-39 (2004)

[39] S. Iwase and S. Saito: "Parallel Tracking of All Soccer Players by Integrating Detected Positions in Multiple View Images", In Proc of IEEE Conference on ICPR, pp. 751-754 (2004).

[40] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speeded Up Robust Features," In Proc. European Conference on Computer Vision (ECCV 2006), pp. 404-417, 2006.

[41] H. Bay, A. ESS, T. Tuytelaars, and L. V. Gool, "Speeded Up Robust Features (SURF)," Journal of Computer Vision and Image Understanding, Vol. 110, No. 3, pp. 346-359, 2008.

[42] C. Rother, V. Kolmogorov, A. Blake, "GrabCut: interactive foreground extraction using iterated graph cuts," ACM Transactions on Graphics (TOG) , **23**, pp. 309-314 (2004)

[43] T. Szigeti, K. McMenamy and R. Saville, "Cisco TelePresence Fundamentals," Cisco Press, May 2009.

[44] O. Schreer, I. Feldmann, W. Waizenegger, N. Atzpadin, P. Eisert, and H. Belt, "3Dpresence: A System Concept for Multi-user and Multi-party Immersive 3D Videoconferencing," in 5th European Conf. on Visual Media Production (CVMP 2008), pp. 321-334 (2008).

[45] I. Feldmann, N. Atzpadin, O. Schreer, J.-C. Pujol-Acolado, J.L. Landabaso, and O.D. Escoda, "Multi-view Depth Estimation based on Visual-hull Enhanced Hybrid Recursive Matching for 3D video Conference Systems," in Proc. 16th IEEE International Conference on Image Processing (ICIP), pp. 745-748, 2009.

[46] J. Civit, O. D. Escoda, "Robust Foreground Segmentation for GPU Architecture in an Immersive 3D Video-Conferencing Systems," IEEE Int. Workshop on MMSP, 2010.

[47] R. Yang and Z. Zhang, "Eye Gaze Correction with Stereovision for Video-Teleconferencing," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 26, no. 7, pp. 956-960, July 2004.

[48] S. Lee, I. Shin, and Y. Ho, "Gaze-corrected View Generation Using Stereo Camera System for Immersive Videoconferencing," IEEE Transactions on Consumer Electronics, vol. 57, no. 3, pp. 1033-1040, Aug. 2011.

[49] C. Zhang, D. Florencio, and Z. Zhang, "Improving Immersive Experiences in Telecommunication with Motion Parallax," IEEE Signal Processing Magazine, vol. 28, no. 1, pp.139-144, Jan. 2011.

[50] D. Scharstein and R. Szeliski, " A Taxonomy and Evaluation of Dense Two-frame Stereo Correspondence Algorithms, " Int. J. Comput. Vision (IJCV), vol. 47, no. 1-3, pp. 7-42 (2002).

[51] K. N. Kutulakos and S. M. Seitz, "A Theory of Shape by Space Carving, " Int. J. Comput. Vision, vol. 38, no. 3, pp. 192-218 (2000).

[52] D. Catuhe, " Programming with the Kinect for Windows Software Development Kit, " Microsoft Press, September 2012.

[53] S. Lee, I. Shin, and Y. Ho, "Generation of Eye Contact Image using Depth Camera for Realistic Telepresence," Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA), pp. 1-4, 2013.

[54] C. Zhang, Q. Cai, P.A. Chou, and Z. Zhang, "Viewport: A Distributed, Immersive Teleconferencing System with Infrared Dot Pattern," IEEE Multimedia, vol. 20, no. 1, pp.17-27, Jan.-Mar. 2013.

[55] H. Wei, L. Xin, G. Cheung, and O. Au, "Depth Map Denoising using Graph-based Transform and Group Sparsity," IEEE 15th International Workshop on Multimedia Signal Processing (MMSP), pp. 1-6, 2013.

[56] S. Lee, I. Shin, and Y. Ho, "Depth Image Filter for Mixed and Noisy Pixel Removal in RGB-D Camera Systems," IEEE Transactions on Consumer Electronics, vol. 59, no. 3, pp. 681-689, Aug. 2013.

[57] C. Rother, V. Kolmogorov, A. Blake, "GrabCut: Interactive Foreground Extraction using Iterated Graph Cuts, " In Proc of ACM SIGGRAPH, vol. 23, pp. 309-314 (2004).

[58] S. Morita, K. Yamazawa, and N. Yokoya, "Internet Telepresence by RealTime View Dependent Image Generation with Omnidirectional Video Camera," In Proc. SPIE Electronic Imaging, Vol.5018, pp.51-60 (2003).

[59] J. L. Landabaso, M. Pardas, and J. R. Casas, "Shape from Inconsistent Silhouette," Computer Vision and Image Understanding, vol. 112, no. 2, pp. 210-224 (2008).

[60] H. Sankoh, A. Ishikawa, S. Naito, and S. Sakazawa, "Robust Background Subtraction Method based on 3d Model Projections with Likelihood," In 12th International Workshop on Multimedia Signal Processing (MMSP 2010), pp. 171-176 (2010).

[61] C. Guillemot and O. L. Meur, "Image Inpainting: Overview and Recent Advances," IEEE Signal Processing Magazine, vol. 31, no. 1, pp.127-144, Jan. 2014.

[62] S. Liu, P. A. Chou, C. Zhang, Z. Zhang, and C. W. Chen, "Virtual View Reconstruction Using Temporal Information," IEEE International Conference on Multimedia and Expo (ICME), pp. 115-120, July 2012.

[63] W. Sun, O. C. Au, L. Xu, Y. Li, and W. Hu, '"Novel Temporal Domain Hole Filling based on Background Modeling for View Synthesis," in Proc. 19th IEEE International Conference on Image Processing (ICIP), pp. 2721-2724, 2012.

List of Publications

Award

1. Academic Encouragement Award 2011 in IEICE (Information and Systems Society), "Highly Precise Free-viewpoint Video Generation Technology Enhanced by Utilizing Zooming-in Camera Images," IEICE General Conference, D-11-51, 2011. (In Japanese)

2. Suzuki Memorial Achievement Award 2011 in The Institute of Image Information and Television Engineers (ITE), "Dynamic Camera Calibration Method for Zoom-camera Based on Feature Point Matching between Video Frames," ITE annual conference, 4-5, 2011. (In Japanese)

3. Fujio Frontier Award 2011in ITE, "Free-viewpoint Video Production and Distribution Technology for Meta Space Realization." (In Japanese)

4. Best Technical Demonstration Runner-ups in The 20th ACM International Conference on Multimedia (ACM Multimedia 2012), "Interactive Music Video Application for Smartphones Based on Free-viewpoint Video and Audio Rendering."

5. Young Researcher Award 2013 in FIT, "Foreground Object Segmentation Method from Sparsely Arranged Multi-view Cameras," Forum on Information Technology (FIT 2013), I-016, 2013. (In Japanese)

6. Technical Progress Award 2013 in ITE, "Sports Video Production Technology "Free Vision" based on Free-viewpoint Video Synthesis." (In Japanese)

Journal Articles

1. Hiroshi Sankoh, Akio Ishikawa, Sei Naito, and Shigeyuki Sakazawa, "Robust Background Subtraction Method Based on 3D-Model Projection with Likelihoods," The Journal of The Institute of Image Information and Television Engineers (ITE), Vol. 64, No. 11, pp. 1685-1697, 2010. (In Japanese)

2. Hiroshi Sankoh and Sei Naito, "Precise 3D Model Reconstruction Scheme Based on Both of Geometric Consistency in Voxel Space and Quality Analysis of Free-viewpoint Image," IEICE Transactions on Information and Systems, Vol. J95-D, No. 9, pp. 1769-1782, 2012. (In Japanese)

3. Hiroshi Sankoh and Sei Naito, "Free-viewpoint Video Rendering in Large Outdoor Space such as Soccer Stadium based on Object Extraction and Tracking Technology," The Journal of The Institute of Image Information and Television Engineers (ITE), Vol. 68, No. 3, pp. J125-J134, 2014. (In Japanese)

4. Hiroshi Sankoh, Sei Naito, Toshiharu Sakata, Mitsuru Harada, and Michihiko Minoh, "Free-viewpoint Video Synthesis for Sport Scenes Captured with a Single Moving Camera," ITE Transactions on Media Technology and Applications (MTA), Vol. 3, No.1, pp. 48-57, 2015.

Refereed Conference Presentations

1. H. Sankoh, A. Ishikawa, S. Naito, and S. Sakazawa, "Robust Background Subtraction Method Based on 3D Model Projections with Likelihood," In Proc. IEEE 12th International Workshop on Multimedia Signal Processing (IEEE MMSP 2010), pp. 171-176, 2010.

2. H. Sankoh, M. Sugano, and S. Naito, "Dynamic Camera Calibration Method for Free-viewpoint Experience in Sport Videos," In Proc. 20th ACM International conference on Multimedia (ACM MM' 12), pp. 1125-1128, 2012.

3. H. Sankoh, M. Sugano, and S. Naito, "Robust Foreground Segmentation from Sparsely Arranged Multi-view Cameras," In Proc. 2013

IEEE 15th International Workshop on Multimedia Signal Processing (IEEE MMSP 2013), pp. 019-024, 2013.

4. H. Sankoh and S. Naito, "Free-viewpoint Video Synthesis for Sport Scenes Captured with a Single RGB-D Camera," In Proc. 2014 IEEE 16th International Workshop on Multimedia Signal Processing (IEEE MMSP 2014), Demo Session, 2014.

Rights for Publications

Journal Articles

Refereed Conference Presentations