Dissertation for Doctor of Philosophy

Depth Map Creation and Mesh-based Hierarchical 3-D Scene Representation in Hybrid Camera System

Sung-Yeol Kim

Department of Information and Communications
School of Information and Mechatronics

Gwangju Institute of Science and Technology

2008
Abstract

As immersive multimedia services are expected to be available anytime and anywhere through a high-speed optical network in the near future, the importance of three-dimensional (3-D) video is increasingly recognized as an essential part of high-quality visual media. As a 3-D video representation, it is widely accepted that a monoscopic color video enriched with per-pixel depth information, which is often called as video-plus-depth, provides the groundwork for the envisaging 3-D applications. In general, we utilize depth-image-based rendering (DIBR) techniques to synthesize 3-D views of a scene from video-plus-depth.

Therefore, with respect to the future 3-D applications, such as 3-D TV, it is very important for us to estimate accurate depth information from real scenes. In order to generate reliable depth maps, a variety of depth estimation algorithms based on stereo matching have been proposed in the fields of computer vision and image processing for decades. However, even though state-of-the-art depth map generation algorithms have been developed, accurate measurement of depth information from natural scenes still remains problematic, because it is difficult to estimate depth information on the textureless or occluded region.

In this thesis, we present a new scheme to generate high-quality and high-resolution depth maps using a hybrid camera system. Basically, the hybrid camera system is constructed by combining stereoscopic RGB cameras and a single-view depth camera, which is used as a supplement to estimate depth information on the region of interest (ROI). With the assumption that possible viewers of 3-D scenes generated by video-
plus-depth tend to focus more on the foreground objects, the ROI, than the background in a natural color image, we enhance the accuracy of ROI depths in the depth map with the depth camera data to provide them natural 3-D views.

In order to generate ROI enhanced depth maps using the hybrid camera set, we first estimate the initial depth information of the left image by applying a color segmentation-based stereo matching algorithm on the left and right images captured by the stereoscopic RGB cameras. Thereafter, the depth information obtained by the depth camera is projected onto the ROI of the left image utilizing 3-D image warping, and then linearly interpolated to fill depth holes occurred in the ROI. Finally, we merge the warped ROI depths with the background ones extracted from the initial depth map to generate the final depth map.

In addition, we propose a new scheme to render dynamic 3-D scenes represented by video-plus-depth fast using meshes, called as a hierarchical decomposition of depth maps. In the hierarchical decomposition, we create three disjoint layers from a depth map according to the existence of edge information in each frame: regular mesh, boundary, and feature point layers. Especially, number-of-layer (NOL) information is used to manage the three disjoint layers and the shape of 3-D surfaces for 3-D scene rendering.

In this decomposition, feature points extracted from a depth map are represented by the three decomposed layers as 3-D geometry information, and then used to construct a 3-D surface in each frame. The corresponding color image of the depth map is used as a texture to cover the 3-D surface. Our contribution is that we find out a method to maintain data regularity of feature points for dynamic 3-D scene rendering, even although the proposed method is a kind of the mesh-based approach. Therefore, we can realize a fast rendering system for the 3-D video contents service based on a video-plus-depth structure. Furthermore, the proposed hierarchical decomposition supports the functionality on multi-layer representation of 3-D scenes by controlling the amount of feature point layers.
## Contents

Abstract i  
Acknowledgements iii  
List of Contents vi  
List of Tables viii  
List of Figures ix  

1 Introduction 1  
1.1 Necessity of Video-plus-depth .......................... 1  
1.2 Problem Statements .................................. 6  
1.3 Contributions ........................................ 11  
1.4 Organization of the Thesis .............................. 14  

2 Review of Previous Works 15  
2.1 Acquisition of Depth Information ....................... 15  
2.1.1 Stereo matching .................................. 16  
2.1.2 Depth Camera System ............................... 21  
2.1.3 Hybrid Depth and Stereo Camera System ............ 23  
2.2 3-D Scene Representation ............................... 25  
2.2.1 Image-based Rendering ............................. 25  
2.2.2 Depth Image-based Representation ................... 27  
2.2.3 3-D Mesh Representation .......................... 28  

3 ROI Enhanced Depth Map Generation 31  
3.1 The Proposed Hybrid System System .................... 31  
3.2 Generation of Depth Map Using Hybrid System ........... 35  
3.2.1 Relative Camera Calibration ........................ 37  
3.2.2 Generation of Initial Disparity Map .................. 39  
3.2.3 Depth Data Correction ............................. 43  
3.2.4 Generation of ROI disparity map ..................... 46  
3.2.5 Generation of ROI Enhanced Depth Map ............... 50  
3.3 Results and Evaluations ............................... 52  

– vi –
4 Hierarchical Decomposition of Depth Maps
   4.1 Regular Mesh Layer .................................................. 66
   4.2 Boundary Layer .......................................................... 70
   4.3 Feature Point Layer ..................................................... 72
   4.4 Number of Layer for 3-D Scene Rendering .......................... 76
   4.5 Results and Evaluations ................................................. 79

5 Applications ................................................................. 92
   5.1 Generation of ROI Enhanced Multi-view Depth Map ................. 92
   5.2 Quality Scalable 3-D Video ............................................. 96
   5.3 3-D Video Contents Generation for Realistic Broadcasting Services .. 103

6 Conclusions ................................................................. 117

References ................................................................. 118
# List of Tables

2.1 Stereo matching algorithms ............................................. 19

3.1 Specifications of proposed hybrid system ........................ 34

4.1 Layer and mode information induced by NOL .................... 77

4.2 3-D shape pattern assigned into grid cells ....................... 78

4.3 Rendering speed comparison ............................................. 90

4.4 Depth quality evaluation according to the size of grid cell .... 90
List of Figures

1.1 Trends in digital broadcasting .......................... 2
1.2 Limitations of current depth cameras ...................... 8
1.3 Depth map representation using meshes .................. 10
1.4 ROI enhanced depth map generation using a hybrid camera set 12
1.5 3-D scene reconstruction using hierarchical decomposition .... 13
2.1 Principle of stereo matching ............................. 17
2.2 Schematic diagram of the depth camera .................. 22
2.3 Color and depth maps obtained from a depth camera ....... 23
2.4 3-D depth acquisition using multi-view images and depth camera .................. 24
2.5 Categories of image-based rendering .......................... 26
2.6 Depth image-based modeling .............................. 27
2.7 3-D mesh model ........................................ 29
3.1 Possible hybrid camera system ............................ 32
3.2 Hybrid system using stereo and depth camera ............... 33
3.3 Four images from hybrid system .......................... 34
3.4 The frame work for depth map generation .................. 35
3.5 Image rectification ....................................... 40
3.6 Color segmentation ....................................... 41
3.7 Initial depth map generated by stereo matching ............. 42
3.8 Depth data range correction ................................ 43
3.9 Inherent problems of depth camera data
3.10 Optical noise reduction.
3.11 Boundary depth cleaning.
3.12 Initial ROI disparity map generated by 3-D image warping.
3.13 Color segment set for ROI.
3.14 Color segment-based hole-filling algorithm.
3.15 ROI disparity map.
3.16 ROI enhanced disparity map.
3.17 ROI enhanced depth map.
3.18 BEAR images.
3.19 The 1st frame of ACTOR image sequence.
3.20 The 2nd frame of ACTOR image sequence.
3.21 The 3rd frame of ACTOR image sequence.
3.22 Generated depth map for BEAR image.
3.23 Generated depth map for the 1st frame of ACTOR image sequence.
3.24 Generated depth map for the 2nd frame of ACTOR image sequence.
3.25 Generated depth map for the 3rd frame of ACTOR image sequence.
3.26 3-D scene construction of ROI for BEAR image.
3.27 3-D scene construction of ROI for ACTOR image sequence.
4.1 Overall outlines of hierarchical decomposition of depth maps.
4.2 Division of depth map using grid cells.
4.3 Generation of regular mesh images.
4.4 Types of a boundary layer ........................................ 70
4.5 Generation of a boundary layer ................................. 71
4.6 Generation of feature point images .............................. 73
4.7 Extracting influent feature points using maximum distance algorithm . 74
4.8 Representation of a 3-D scene with decomposed depth data ........ 75
4.9 NOL information .................................................. 77
4.10 Filling holes using additional triangles ........................ 79
4.11 Test sequences .................................................... 80
4.12 the 12th frame of Homeshopping ............................ 82
4.13 the 23rd frame of Homeshopping ............................. 83
4.14 the 60th frame of Breakdancers ............................... 84
4.15 the 87th frame of Breakdancers ............................... 85
4.16 the 1st frame of ACTOR ...................................... 86
4.17 the 2nd frame of ACTOR ...................................... 87
4.18 the 3rd frame of ACTOR ...................................... 88
4.19 The rendering result of dynamic 3-D scenes ................. 89
4.20 Visual quality estimation ...................................... 91
4.21 Visual quality estimation (grid cell:16×16) ................. 91

5.1 Hybrid camera set combining multi-view cameras with a depth camera. 93
5.2 Overall framework for multi-view ROI enhanced depth map generation using a hybrid camera system. ............................. 93
5.3 Results for ROI depth estimation ............................... 97
5.4 Intermediate view generation. ........................................... 98
5.5 System architecture for quality scalable services of 3-D video .... 100
5.6 Representation of multi-layer structure ................................. 101
5.7 Results of base layers .................................................. 102
5.8 Results of base layers and one enhancement layer ............... 102
5.9 Results of base layers and two enhancement layers ............. 103
5.10 Proposed 3-D video content generation system ....................... 105
5.11 Hybrid camera set and its output images ......................... 107
5.12 Depth map generation of a dynamic object ......................... 108
5.13 Dynamic 3-D scene generation using hierarchical decomposition. 109
5.14 Node design for video-plus-depth .................................. 111
5.15 Construction of a hybrid camera set ............................... 113
5.16 3-D home-shopping content ....................................... 114
5.17 Free view-point changing .......................................... 115
5.18 Other interactions .................................................. 115
Chapter 1

Introduction

1.1 Necessity of Video-plus-depth

As the rapid development in the field of digital communications, computer powers, and high-speed networks, it is not too much to say that we are living in the age of information revolution and digital epoch. We can not only feel deep impressions by a high definition (HD) television equipped with a large screen and a high power speaker, but also call to someone with a cellular phone supporting the functionality for playing moving pictures. In addition, banking services and product purchases are possible at home through Internet. Furthermore, we will be able to touch and manipulate provided contents directly with haptic devices in the near future. In this wise, we are endeavoring to create reproductions, which are imitated with the real world, to feel the impression of ‘being there’, or presence, from them as much as possible [1].

We can say that the essence of the digital age is the advance of multimedia technology for digital broadcasting. Since a digital broadcasting system converts analog signals to digital ones to minimize interference among transmitted data, it is possible to serve high-quality audio-visual multimedia data to users. In addition, since the digital broadcasting system uses the assigned frequency bandwidth effectively, we can transmit various multimedia data through a limited communication channel.
In virtue of the rapid growth of digital technology, the tendency of broadcasting services [2] has been changed from one-directional services to bi-directional services or interactive services, such as stereoscopic TV [3], three-dimensional (3-D) TV [4], and realistic broadcasting [5]. As shown in Fig. 1.1, the next-generation broadcasting system is supposed to provide a variety of user-friendly interactive information, as well as high-quality audio-visual broadcasting contents.

Especially, 3-D TV is considered a main theme as a future broadcasting system supporting a natural viewing experience in the true dimension. In general, 3-D natural views are usually created from the two 3-D video representations: a multi-view video [6] and a video-plus-depth [7]. A multi-view video represents a 3-D video with the collection
of multiple videos generated by capturing a scene at different camera locations. Since
the multi-view video produces natural 3-D views with a number of images at the viewing
position, we can be easily immersed in the 3-D content. However, we need to much
effort to control a huge number of camera at the same time. Moreover, since a multi-
view camera system usually requires complicated coding and transmission schemes [8]
in proportional to the number of cameras, it is hard to sent its data to the receiver
side within a limited bandwidth channel environment.

As the alternative for the 3-D video representation, it is widely accepted a mono-
scopic color video enriched with per-pixel depth information, which is often referred as
a video-plus-depth. Since the video-plus-depth includes depth information as geometry
data in a scene, we can generate free-viewpoint images using depth image-based
rendering (DIBR) techniques [9] for the 3-D video contents service. Although the video-
plus-depth can support narrow-viewing angle views in comparison with a multi-view
video, they are considered as a suitable 3-D video representation for 3-D TV, because
it can support both backwards-compatibility to current 2-D digital systems and easy
adaptability to a wide range of different 2-D and 3-D displays. Recently, the ISO/ICE
JTC/SC29/WG11 Moving Picture Experts Group (MPEG) has also been interested in
multi-view video with depth (MVD), which is the combination of the multi-view video
and the video-plus-depth, for free-viewpoint TV (FTV) and 3-D TV [10], [11].

With respect to the current 3-D TV and FTV research activities, it is very important
for us to estimate accurate depth information from natural scenes. In the field of
computer vision and image processing, a number of depth estimation algorithms have
been proposed to generate accurate depth maps [12]. However, accurate measurement of depth information from natural scenes still remains problematic.

In general, we can classify 3-D depth estimation methods into two categories: active depth sensing and passive depth sensing. Passive depth sensing methods calculate depth information indirectly from 2-D images captured by two or more video cameras. Examples of passive depth sensing include shape from stereo [13], [14], [15] shape from silhouette [16], and shape from focus [17]. Since the passive depth sensing needs simpler and cheaper depth acquisition system than active depth sensing, we can create depth maps inexpensively. However, accuracy of the depth information obtained by passive depth sensing methods is relatively lower than one captured by active depth sensing methods directly.

On the other hand, active depth sensing methods usually employ physical sensors as a supplement for depth acquisition, such as laser sensors, infrared ray (IR) sensors, or light pattern sensors. In other words, active depth sensing methods obtains depth maps of natural scenes directly using these kinds of sensors. Structured light patterns [18] and depth cameras [19] are representative examples of the active depth sensing methods. Although currently available direct depth estimation tools are quite expensive and produce only low-resolution depth maps, they can create more accurate depth maps in a shorter time than passive depth sensing methods.

Especially, we can obtain depth maps from natural scenes in real time using a time-of-flight (TOF) depth cameras, such as Z-Cam, developed by 3DV Systems, Ltd. [20] or NHK Axi-vision HDTV camera [21]. These depth cameras simultaneously capture
color images and their associated per-pixel depth information by integrating a high-speed pulsed IR light source with a conventional broadcast TV camera. The ATTEST project [22] has already shown us a possibility of realizing a 3-D TV system using a depth camera, Z-Cam.

On one hand, there are two approaches to render consecutive 3-D scenes represented by a video-plus-depth: mesh-based rendering [23], [24] and image-based rendering [25]. In the mesh-based rendering approach, we extract feature points from a depth map, and then generate a 3-D surface with the feature points utilizing a mesh triangulation technique in each frame. The main advantage of the mesh-based representation is a high rendering speed, because we usually employ a graphic accelerator to render dynamic 3-D scenes.

On the other hand, the image-based rendering method uses all depth information in a depth map to create 3-D views. In order to synthesize virtual views from the video-plus-depth in the image-based representation, we project color and depth information onto the virtual camera using 3-D image warping, and then remove holes occurred by the warping process using a hole-filling algorithm.

In this thesis, we focus on the generation and processing of the video-plus-depth in a hybrid camera system. In the hybrid camera environment, we capitalize on the advantage of both passive depth sensing and active depth sensing approaches at the same time to estimate accurate depth information. In addition, we present a fast 3-D scene rendering algorithm using a mesh-based rendering approach.
1.2 Problem Statements

The first question to handle in this Thesis is,

- “Can we obtain high-resolution and high-quality video-plus-depth using a current low-resolution depth camera?”

Stereo matching is well-known as the passive depth sensing method. In stereo matching, we determine pairs of points that correspond to same scene point in stereoscopic images, and the distance of two points in a pair, called as disparity, is used to estimate the depth information. However, since textureless or occluded regions usually cause the matching failure in the passive depth estimation, it is very difficult to obtain accurate depths on the regions. In order to get more accurate depth information, we can utilize a depth camera technology. However, there are still some inherent problems in the current depth camera system unfortunately.

The first problem of the current depth camera is that the captured depth map includes optical noise. Optical noise usually occurs as a result of differences in reflectivity of IR sensors according to color variation in objects. In addition, the generated depth map may have unmatched depth data with the corresponding color data on the region of object boundaries. Kim et al. have developed a novel modeling algorithm that uses a series of processing steps including median filtering, adaptive sampling, mesh triangulation, and Gaussian smoothing [19] to minimize optical noise in the depth map captured by a depth camera. In a row, Cho et al. have also analyzed the inherent problems of the depth map and proposed some practical solutions to deal with these
problems [26]. Especially, they dealt with the region that had failed to obtain depth information using a depth camera due to the characteristic of IR sensors, such as the region of a black human hair.

The second problem of the current depth camera is that the measuring distance of the depth camera to capture depth information is limited. In practice, the depth capturing distance of current depth cameras is approximately from 1m to 4m. Hence, we cannot obtain depth information from objects out of the measuring distance. Furthermore, when we increase the measuring distance, the precision of depth information in the depth map is decreased, because the measuring distance is quantized in 256 levels to represent depth values. In 2005, Um et al. proposed a depth map fusion method that combines trifocal cameras and a depth camera [27] to increase the measuring distance of a depth camera. This hybrid camera system has generated enhanced depth maps estimated by applying a stereo matching algorithm to three-view images with depth information captured by the depth camera. However, this system cannot produce high-resolution depth maps, because it completely depends on the low-resolution depth camera.

The last problem of the current depth camera is that the current depth camera system can only produce low-resolution depth maps. The resolution of depth maps acquired by Z-Cam is 720×486. Moreover, the Axi-vision HDTV camera, which was aimed at the acquisition of high definition (HD) depth maps, maximally produces depth maps with the image resolution 1280×720, as shown in Fig. 1.2.
Figure 1.2: Limitations of current depth cameras.

One possible solution for generating spatially high-resolution depth maps is to upgrade the current depth camera or develop a new depth camera. However, due to many challenges in real-time distance measuring systems, upgrading and improvements of depth cameras are very costly and time-consuming. There were some hybrid approaches to extend the depth map resolution spatially using a current depth camera. Diebel et al. constructed a hybrid camera set composed of a high-resolution color camera and a low-resolution depth camera, and generated high-resolution range images using a concept of Markov Random Fields [28]. In addition, Hahle et al. also integrated a 3-D time-of-flight depth camera and a standard color camera to make use of dependencies and characteristics of different modalities recently [29]. However, since these works have mainly focused on a static 3-D scene reconstruction with the refined depth information acquired from the two-camera setup, they could not provide depth
information of background, the region out of the measuring range of a depth camera.

The second question to handle in this Thesis is

- “Can we render consecutive 3-D scenes generated by spatially high-resolution video-plus-depth using mesh-based representation?”

It is hard to render consecutive 3-D scenes generated by spatially high-resolution video-plus-depth in real time. A method to directly render 3-D models and surfaces represented by a mesh structure using programmable graphics processing units (GPUs) [30] has been issued these days to increase the rendering speed. However, it is still hard to create 3-D scenes represented by a tremendous number of triangles in real time. For example, when the resolution of depth maps is 1024×768 and each pixel is triangulated by the simple precedent of adding horizontal, vertical and diagonal lines between each pixel, the number of needed triangles to render for each frame is 1,565,190. Furthermore, if we are willing to provide high definition (HD) level contents, it is almost impossible to render 3-D scenes constantly in real time.

In order to reduce the number of triangles during the mesh-based rendering, we create a 3-D surface with a small number of feature points frame by frame [31], as shown in Fig. 1.3, and then cover the surface with the color image using a texture mapping technique. In this thesis, feature points mean depth pixels in a depth map that influence critically on the shape of 3-D scenes. In this wise, we extract feature points from depth maps, and then generate consecutive 3-D surfaces by applying a mesh triangulation technique to render them in real time. Grewatsch et al. presented a mesh-based representation using feature points [23]. Chai et al. also introduced a depth map
representation using an adaptation of the triangular mesh generation [24]. For real-time rendering, they matched the tree traversal information into the mesh rendering order. Recently, Morvan et al. proposed another scheme to represent depth map using meshes, which has employed quad-tree decomposition and plane approximation [32].

These previous depth map representation using meshes selected feature points irregularly. The irregularity caused to increase rendering time of consecutive 3-D scenes generated by video-plus-depth data. One solution to remove the irregularity is that we regard all depth information in a depth map as feature points. However, since this approach generates a huge number of triangles as we mentioned before, it is almost impossible to render consecutive 3-D scenes in real time. Hence, it is necessary to develop efficient scheme to extract a small number of feature points regularly without serious visual quality degradation.

![Depth map and 3D surface using feature points](image1.png)

Figure 1.3: Depth map representation using meshes.
1.3 Contributions

There are two main contributions in this thesis. The first contribution [33] is that

- We proposed a new scheme to generate depth maps enhanced the depth information on the region of interest (ROI) using a hybrid camera set.

In general, people tend to focus more on interesting foreground objects, ROI, than background in a natural color image. ROI image processing considering the tendency of human is widely used to provide an optimized video service in the limited channel environment [34]. Recently, as the importance of video-plus-depth has increased for the future 3-D application, it is necessary to get high-quality information for ROI in the depth maps, as well as in the color images.

In this thesis, we focus on the two limitation of currently available depth camera: limitation of measuring distance for depth acquisition and generation of spatially low-resolution depth maps. Due to the first limitation, it is almost impossible to obtain all depth information corresponding color information captured by a RGB camera in a depth camera system. As considering the limitation of measuring distance in a depth camera system, we use a depth camera as a supplement to obtain accurate depth information for ROI in a scene. In addition, in order to generate spatially high-resolution depth maps, we construct a hybrid camera set which is composed of high-resolution stereoscopic or multi-view cameras and a low-resolution depth camera.

In the proposed hybrid camera set, depth information of low-resolution depth camera are projected onto the high-resolution RGB cameras to generate high-resolution
Figure 1.4: ROI enhanced depth map generation using a hybrid camera set.

ROI depth maps in each frame. The depth information of background, no ROI, are estimated by a conventional depth estimation based on stereo matching. Finally, the estimated ROI and background depth maps are merged into one to generate an ROI enhanced depth map in each frame. As a result, we can produce a high-resolution and high-quality video-plus-depth by the hybrid camera set. Figure 1.4 shows the proposed system to acquire high-quality depth video.

The second contribution [35] is that

- We developed a new scheme to render dynamic 3-D scenes represented by video-plus-depth in real time using a hierarchical decomposition of depth maps.
In order to render consecutive 3-D scenes with DIBR using meshes in real time, the proposed hierarchical decomposition separates depth maps into three disjoint layers according to the existence of edges: regular mesh, boundary, and feature point layers. In the hierarchical decomposition of depth maps, the three layers represent regularly-selected feature points from depth maps. In order to render 3-D scenes represented by video-plus-depth in real time, we can generate a 3-D surface using pre-defined 3-D shape patterns according to the layer and the selected feature point in each frame. The generated 3-D surface is covered by the corresponding color image using texture mapping. As a result, we can realize a fast rendering system to support 3-D contents generated by video-plus-depth. Especially, the proposed hierarchical decomposition supports multi-layer structure by controlling the amount of feature points in the feature points layer. Figure 1.5 shows the 3-D scene reconstruction based on the hierarchical decomposition of depth maps.
1.4 Organization of the Thesis

The thesis is organized as it follows. In Chapter 2, we will briefly describe related works about acquisition of depth information and 3-D scene representation. It contains the overview of stereo matching, depth camera system and hybrid depth and stereo camera system. In addition, we will explore the concept of depth image-based rendering and 3-D mesh representation.

In Chapter 3, we will describe our proposed scheme to generate high-resolution and high-quality depth maps using a hybrid camera system. It consists of the description part of proposed hybrid camera set and the generation part of depth maps.

In Chapter 4, we will introduce proposed dynamic 3-D scene rendering scheme based on hierarchical decomposition of depth maps. We also explain how to represent and render 3-D scenes with video-plus-depth.

In Chapter 5, three examples of applications using the proposed video-plus-depth techniques are presented. One is the generation of multi-view depth maps for 3-D TV. Another is a 3-D video player system supporting quality scalability based on the framework of multi-layer representation. The other is an interactive 3-D contents generation supporting various user-friendly interaction, such as free viewpoint changing, free composition with computer graphics, and even haptic interaction.

In Chapter 6, we will summarize the main issues and recommends future works.
Chapter 2

Review of Previous Works

2.1 Acquisition of Depth Information

Recently, many works have been carried out for the acquisition of 3-D depth information in the field of computer vision and image processing. 3-D depth sensors can be classified into two categories: active and passive depth sensors [36] [37]. The former method uses an active illumination and provides highly accurate depth information. The 3-D depth acquisition methods using lasers or structured light [38] [18] belong to the active sensor category, and are applied to obtain 3-D depth information of static scenes. However, it is very difficult to get the depths from dynamic real scenes at video frame rates using these kinds of active sensor methods. Recently, depth cameras equipped with an infrared rays (IR) source have released [39] so that they can capture depth information of natural scenes in real time based on the principle of time of flight (TOF) [20]. However, these active depth sensors are usually expensive unfortunately. Moreover, the depth camera is only available in a limited indoor environment, such as a virtual studio.

On the other hand, passive sensors acquire 3-D depth information from the images obtained by multi-view cameras. Although the passive approaches capture texture information along with depth information, they have many errors in the regions of
occlusion, illumination changes, and low texture. Besides, we need long computation time to get depth information. Examples of passive 3-D depth sensors are shape from stereo [40], shape from silhouette [41], and shape from motion [16].

2.1.1 Stereo matching

Stereo matching is best known as one of passive 3-D depth sensing methods. Extraction of depth information from stereoscopic images has been studied in the field of computer vision for decades. Recently, multi-view images has been used to obtain more accurate depth information instead of binocular images. The task of stereo matching is the computation of 3-D data from 2-D input images. It is exactly what the human visual system is doing when we perceive depths. Since two images captured by our eyes are obtained from slightly different perspectives, the position of a scene point in one view is horizontally displaced in the other view. The amount of the displacement allows reasoning about the depth of the scene point.

As shown in Fig. 2.1, we determine pairs of points that correspond to same scene point in binocular images in stereo matching. The length of the horizontal displacement vector is commonly called as disparity. Basically, the disparity of a pixel is inversely proportional to the distance of the pixel from the cameras. The human brain converts the disparity information into a 3-D impression of the world by recognizing the principle. Although the process seems to be simple in human visual system without our any efforts, the stereo correspondence matching task turns out to be difficult when we try to solve the task with a computer.
Figure 2.1: Principle of stereo matching.

Scharstein et al. has given us an extensive review on stereo matching algorithms [12]. His work mentioned four steps performed during stereo matching: matching cost computation, cost aggregation, optimization and disparity refinement. According to the way to find out correspondence points, we can classify stereo matching algorithms into two categories: local and global correspondence methods. The local correspondence method relies on the local information to determine the corresponding point. On the other hand, the global correspondence method depends on the information of a whole scanline or the entire image to compute disparities. While local correspondence methods are sensitive to ambiguous regions, such as uniform zones or occlusion regions, global methods are less sensitive to the regions. However, we need more computational power to estimate disparities in the global approaches.

In general, global correspondence methods consider smoothness constraint to gen-
erate depth maps. Most global approaches are formulated by energy minimization. When we define the energy $E(\varepsilon)$ at disparity $\varepsilon$, $E(\varepsilon)$ is calculated by Eq. 2.1.

$$E(\varepsilon) = E_{data}(\varepsilon) + \lambda E_{reg}(\varepsilon)$$  \hspace{1cm} (2.1)$$

where $E_{data}(\varepsilon)$ is the cost of matching left and right images at a disparity $\varepsilon$, and $E_{reg}(\varepsilon)$ is a regularization term to preserve the discontinuity of disparities at a disparity $\varepsilon$.

In local correspondence methods, we assume that tiny image patches have similar intensity patterns across views. In order to find out the point of maximum correspondence, the local approaches usually make a window move on the scanline on the other view image. The final disparity is obtained by selecting the point of the highest matching score. Since the matching score of a pixel in the local methods is not influenced on disparities of neighboring pixels, we have only to concentrate on the pixel locally. In contrast to the local approaches, global correspondence methods generate their smoothness models using neighboring pixels. Table 2.1 shows the outline for the local and global correspondence methods of state-of-the-art stereo matching algorithms [42].

In order to reduce erroneous results around disparity discontinuities, segmentation-based stereo matching has been proposed [40]. There are three assumptions in the segmentation-based stereo matching. One is that disparity discontinuities align the region of edges. Another is that disparity variation within a segment is small. The other is that approximation of a scene with piece-wise planar surfaces is close to its real surface.

Segmentation-based stereo matching algorithms divide images into segments having
Table 2.1: Stereo matching algorithms

<table>
<thead>
<tr>
<th>Method</th>
<th>Algorithm</th>
<th>Weakness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic Programming</td>
<td>Algorithm</td>
<td>Determine disparity surface for a scene as the best path over sequence of ordered primitives. Typically, order is defined by the epipolar ordering constraint.</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>Suffer from horizontal streaks, especially in regions of low texture. Depend on the validity of the ordering constraint, so disparity computation fails in the region of thin objects.</td>
</tr>
<tr>
<td>Graph Cuts</td>
<td>Algorithm</td>
<td>Determine the disparity surface as the minimum cut of the maximum flow in a graph</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>Disparity values which lie between the disparities of fore- and background are computed in the proximity of depth discontinuities</td>
</tr>
<tr>
<td>Intrinsic Cues</td>
<td>Algorithm</td>
<td>Map epipolar search to intrinsic curve space to convert the search problem to a nearest-neighbor look-up problem. Ambiguities are resolved using dynamic programming.</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>Suffer from occlusion and uniform or repetitive texture in the image. No computational algorithm to measure occlusions.</td>
</tr>
<tr>
<td>Belief Propagation</td>
<td>Algorithm</td>
<td>Solve for disparities via message passing in a belief network</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>Unsuitable for real-time applications.</td>
</tr>
<tr>
<td>Cooperative Method</td>
<td>Algorithm</td>
<td>Initial matching scores using a function of image intensities are refined by an iterative update function that applies uniqueness and consistency assumptions.</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>Higher computational effort, and depth boundaries tend to be blurred. Depend on good initialization.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Local Methods</th>
<th>Algorithm</th>
<th>Weakness</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block Matching</td>
<td>Algorithm</td>
<td>Search corresponding points base on minimum error over small region, typically using normalized cross-correlation or rank metrics.</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>Disparity within the window must be constant. Blurs across depth discontinuities. Fills the results towards frontal-parallel surfaces. Poor performance in occluded regions.</td>
</tr>
<tr>
<td>Gradient-based Optimization</td>
<td>Algorithm</td>
<td>Minimize typically the sum of squared differences, over a small region.</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>Relies on the hypothesis that object brightness remains constant between images, so sensitive to the noise of intensities.</td>
</tr>
<tr>
<td>Feature Matching</td>
<td>Algorithm</td>
<td>Match only for significant geometric features like edges, curves, corners.</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>Produces dense maps, but sparse maps.</td>
</tr>
<tr>
<td>Phase-based Method</td>
<td>Algorithm</td>
<td>Globally align the images corresponding to a phase modulation of image's Fourier transform</td>
</tr>
<tr>
<td></td>
<td>Weakness</td>
<td>A bad choice of scale parameter increases the disparity error.</td>
</tr>
</tbody>
</table>
non-overlapping homogeneous color. A disparity is assigned to a segment based on initial disparity estimation. The initial disparities of each segment will be improved by iterative optimization or refinement stages using the information of neighboring segments. So far, the segmentation-based approaches have shown better performances than any other stereo matching algorithm. Most top-ranked stereo matching algorithms in the Middlebury database are segmentation-based methods. Gelautz [43] summarize the advantages and disadvantages of segmentation-based stereo matching.

Advantages of segmentation-based stereo matching are

- Since segmentation-based stereo matching algorithms constrain the disparity inside a region to follow a single disparity model, the disparity inside regions of poor texture can be assigned as smooth disparity values.
- Disparity boundaries can be more accurately identified by the use of monocular cues than use of disparities only in stereo matching.
- The robustness in occlusion areas is improved. In theory, matching might even succeed for a segment that is partially occluded, since it it still possible to match the segment’s non-occluded pixels.
- The number of segments is usually significantly smaller than the number of pixels.

This give rise to potentially much faster stereo matching algorithms.

Disadvantages of segmentation-based stereo matching are

- The success of such methods depends on the ability of the segmentation algorithm to accurately delineate the objects outlines. Oversegmentation is therefore safer.
• Choosing an appropriate disparity model is a difficult task to represent the real disparity. While simple models may oversimplify the real disparity, complex models may overestimate the data and show undesired effects due to image noise.

• Modeling the stereo correspondence problem on the segment level exclusively is still insufficient to handle occlusions.

2.1.2 Depth Camera System

3-D information of animated computer graphic models can be represented by a set of synchronized color and depth streams. In general, a simulated video-plus-depth is easily obtained by rendering the computer graphic models in off-line rendering packages, where depth buffer values are saved as depth maps. However, it is difficult to acquire the depth map sequence from dynamic real objects, because current capturing devices or related algorithms lack of accuracy and reliability to capture them in real time. Recently, the depth camera that equipped with the active depth sensor has been developed to produce relatively better quality depths. The depth camera captures the depth values in a scene and generates a depth map of the scene by gray-level scaling of the real distances. The active depth camera makes it possible to record dynamic real objects at video frame rate. Figure 2.2 shows the schematic diagram of a depth camera.

A depth camera consists of a conventional video camera and a D camera that produce color and depth signals respectively. The D camera captures the depth value of each pixel for dynamic objects in the FOV. The depth value implies the scaled
distance from the camera to the object by 8-bit gray level. The principle of the depth camera is based on an TOF technique. The IR laser emerges from the ring illuminator to form the field of illumination (FOI), and the taking lens gathers the reflected light from the objects in the FOV. The D-module device splits the incoming light into two parts. The visible light is sent to the RGB camera, while the IR light is reflected toward the D camera, which delivers the depth information. The capturing range defined as a light wall is quantized into the level of 256, and depth information is saved in gray-level images. Figure 2.3 shows color images and depth maps captured from a depth camera.

The depth map at each frame represents 3-D information of the objects or scenes at the specified single viewpoint. The accuracy of the depth map depends on the user-defined capturing range, which has a typically sub-centimeter resolution in narrow capturing range. However, the quality of the depth map is also affected by environ-
mental conditions and operational parameters. The depth camera can capture dynamic objects and scenes more than 60 fps, but recording does not keep up with the capturing time. Therefore, the capturing frame rate is usually set to 30 fps.

2.1.3 Hybrid Depth and Stereo Camera System

In order to acquire more accurate 3-D information, we can utilize both passive and active 3-D sensor methods together. In 2005, Electronics and Telecommunications Research Institute (ETRI) in Korea first developed a system enabling 3-D scene reconstruction combining multi-view images with a depth camera [27]. In order to generate high-quality depth maps, a depth camera first measures the distance from the camera to the scene at each pixel together with its color image as an active depth sensing method. Then, the ETRI work uses a passive depth sensing method using a multi-view image-based stereo matching. For obtaining enhanced video-plus-depth, a three view camera is used as shown in Fig. 2.4.

The center-view image is used as a reference image. Two disparity maps are ob-
Figure 2.4: 3-D depth acquisition using multi-view images and depth camera.

Obtained from stereo matching between the center-left image pair and the center-right image pair. After the rectification, we obtain disparity maps for the two stereo image pairs: center-Left, and center-Right. In the ETRI work, the method developed by Zitnick et al. [40] and Jeong et al. [44] were used together as the stereo matching algorithm. Any stereo matching algorithm can be employed, because the fusion method is independent of stereo matching algorithms. Finally, the two disparity maps obtained from stereo matching are used as inputs of the propose fusion method.

In general, although the disparity map obtained by stereo matching provides a wide depth range, it has errors in the occluded or low texture area of the image. On the
other hand, the depth map obtained from the depth camera is usually more accurate than disparity map from stereo matching. However, the depth camera approach gives an erroneous depth value when the object has a shiny surface or includes black color in a scene. Moreover, the depth camera can be used only in the indoor environment. In order to compensate the weakness of each method and to get a more accurate depth map for the reference view, depth map fusion method uses the reliability of each pixel depth value from one depth maps and two disparity maps. As we reviewed the ETRI work, the output of the depth maps was a standard definition (SD) level video-plus-depth that the current depth camera only supports, because it was dependent on the depth camera, not multi-view cameras.

2.2 3-D Scene Representation

2.2.1 Image-based Rendering

In recent years, computer graphics and computer vision have become interdisciplinary and have attained concentrative interests in the graphics community with set of techniques known as image-based modeling and rendering (IBMR) [45]. One of the main advantages of IBMR techniques is the ability of producing photo-realistically rendered images from the real-world images. Thereby, image-based approaches reduce tedious and labor-intensive modeling and rendering tasks that are required to generate complex 3-D scenes using conventional techniques.

Since there have been researches on geometry-based rendering methods, lots of useful modeling and rendering techniques have been developed. However, geometry-
based rendering requires elaborate modeling and long processing time. As an attractive alternative to overcome these problems, image-based rendering (IBR) techniques [46] have received much attention. They use 2-D images as primitives to generate an arbitrary view of the 3-D scene. IBR techniques require proper computational resources and do not bother from the complexity of 3-D objects in the scene. In addition, it is much easier to acquire a photo or a picture than complex 3-D models of the scene. In spite of these benefits, the amount of data generated from IBR is very huge. Therefore, coding of IBR data is one of the main requirements of IBR techniques.

![Categories of image-based rendering](image)

**Figure 2.5: Categories of image-based rendering**

Various IBR techniques can be classified into three categories based on how much geometry information is used: rendering with no geometry, rendering with implicit geometry, and rendering with explicit geometry, as shown in Fig. 2.5. Among the variety of methods, a layered depth image (LDI) [47] is one of the efficient rendering methods for 3-D objects with complex geometries. LDI is contained in rendering with explicit geometry. It represents the current scene using an array of pixels viewed from a single camera location.
2.2.2 Depth Image-based Representation

As an input of image-based modeling, depth map representation can be a good solution, which consists of color and depth map sequence. With the development of active depth sensor, the depth video can be captured directly at a single camera view using the time-of-flight (TOF) technique. Since there are no angular differences between the color camera and the depth sensor, depth values are assigned to corresponding pixels of the color camera. One of the major advantages of the depth camera is relatively independent from the constant texture and the shape visibility. Therefore, a depth image-based representation and modeling can be a suitable method to reconstruct accurate geometric models for the broadcasting applications [9].

![Feature Points, 3-D surface, 3-D scene](image)

Figure 2.6: Depth image-based modeling

Especially, we can generate 3-D scenes using depth map representation using meshes. In the mesh-based rendering, polygons are used as rendering primitives that are optimized to conventional rendering pipelines. As shown in Fig. 2.6(a), we select a number of feature points in a depth map. The feature point plays a role as geometry infor-
mation in 3-D space. Therefore, a 3-D surface can be generated by the feature point data using a mesh triangulation algorithm, such as Delaunay triangulation [49], as shown in Fig. 2.6(b). Then, the generated 3-D surface is covered by the corresponding color image to create 3-D scenes, as shown in Fig. 2.6(c). Although 3-D geometric reconstruction of real scenes is quite difficult when conventional geometric techniques are used, the depth map representation using meshes enables us to render 3-D scenes constantly in real time, so that we can enjoy 3-D views. Furthermore, the mesh-based representation has the ability of supporting multi-modal interactions.

2.2.3 3-D Mesh Representation

In the section, we review briefly about the 3-D meshes that is the basic concept for the depth map representation using a mesh structure [48]. In general, a mesh model $M$ is simply a set of planar polygons in the three-dimensional Euclidean space $R^3$. To represent a mesh surface, we can assume that the model consists of triangular faces entirely. Any non-triangular polygons can be triangulated by a pre-processing. Fundamentally, there are three types of information to describe the mesh surfaces. These are geometry information, connectivity information and photometry information.

The geometry information describes vertex locations which are represented by three-dimensional floating vectors $\{v_x, v_y, v_z\}$. The connectivity information describes the incidence relations among mesh elements which are vertices, edges and faces. A vertex is represented as an index in vertex incidence table. An edge $\{e_{vid1}, e_{vid2}\}$ is represented as a set of two vertex indices. A face $\{f_{vid1}, f_{vid2}, f_{vid3}\}$ is also represented as a set of
three vertex indices. The photometry information includes surface normal vectors, colors, texture coordinates and other mesh properties attached to mesh elements. Fig. 2.7 shows the information of mesh representation.

![3-D mesh model](image)

*Figure 2.7: 3-D mesh model.*

We will use the following definition: a mesh model \( M = (V, F) \) contains a list of vertices \( V \) and a list of triangular face \( F \). The vertex list \( V = (v_1, v_2, \ldots, v_r) \) is an ordered sequence which has the total number of vertex \( r \). The face list \( F = (f_1, f_2, \ldots, f_n) \) is also ordered sequence which has the total number of triangular face \( n \).

In addition, photometry information includes surface normal vectors, colors, and texture coordinates, which are the attributes of vertices needed to render the 3-D mesh model. In general, mesh model can be divided into two categories according to the type of photometry information: meshes with color and mesh with texture. While meshes with color have color information at each vertex directly, meshes with texture
have texture coordinate information at each vertex to indicate the location of texture
segment to map to the corresponding face.

In general, when we acquire a 3-D model data from a 3-D data acquisition device,
we usually employ a texture mapping technique with texture coordinates and texture
images instead of colors. In the depth map representation using meshes, we can regard
the x- and y-coordinate of each pixel as the texture coordinate information. Therefore,
we generate 3-D scenes using meshes with texture.
Chapter 3

ROI Enhanced Depth Map Generation

In this thesis, we propose a new scheme to generate region-of-interest (ROI) enhanced depth maps combining high-resolution stereoscopic cameras with a low-resolution active range depth camera [33]. Basically, the hybrid camera set produces four synchronized images for each frame: left and right images from the stereoscopic cameras, a color image and its associated depth map from the depth camera. In order to generate ROI enhanced depth maps using the hybrid camera system, we first estimate the initial depth information for the left image by applying a color segmentation-based stereo matching algorithm on the left and right images. Thereafter, the depth information obtained by the depth camera is projected onto ROI of the left image using a 3-D image warping technique, and then linearly interpolated to fill depth holes in ROI as a unit of color segment. Finally, we merge the warped depth information of ROI with the background one of the initial depth map to generate the final ROI enhanced depth map.

3.1 The Proposed Hybrid System System

In order to overcome the limitation of the current depth camera and to use them as the supplement for depth estimation, we propose a camera set combining an RGB camera set and a depth camera. The RGB camera set is composed of single camera
or multiple cameras. When the RGB camera set is constructed by multiple cameras, i.e. multi-view cameras, the characteristic of each camera lens is equal to ones of other cameras. In this thesis, we call the camera system as a hybrid camera set or a hybrid camera system. Figure 3.1 shows the possible hybrid camera set using RGB cameras and a depth camera.

![Possible hybrid camera system](image)

Figure 3.1: Possible hybrid camera system

The construction of the hybrid camera system can be divided according to the property of the color lens in a depth camera. When the property and resolution of the color lens in the depth camera system is different with ones of RGB cameras, the position of a depth camera is under the RGB camera. On the other hand, when the property and resolution of the color lens in the depth camera system is equal to ones of RGB cameras, the position of a depth camera is beside the RGB cameras. Since the
depth camera plays a role as a supplement to estimate depth information at the RGB cameras, outputs of the hybrid camera system are as many video-plus-depths as the number of RGB cameras.

In this thesis, we generate ROI enhanced depth maps with a possible hybrid camera set combining two high-definition (HD) RGB cameras and a standard-definition (SD) depth camera, Z-Cam, as shown in Fig. 3.2. Each camera in the hybrid camera system is connected to a personal computer equipped with a video capturing board. In addition, a clock generator is linked to the camera set to provide synchronization signals constantly. Table 3.1 shows the general specification of the hybrid system. In this chapter, we are willing to generate the ROI enhanced depth maps at the left camera using the hybrid camera set.

![Figure 3.2: Hybrid system using stereo and depth camera](image)

With the proposed hybrid camera system, we capture four synchronized 2-D images
Table 3.1: Specifications of proposed hybrid system

<table>
<thead>
<tr>
<th>Types of Device</th>
<th>Specifications</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stereo camera</td>
<td>Output Format</td>
<td>NTSC or PAL (16:9 ratio, High Definition)</td>
</tr>
<tr>
<td>Depth Camera</td>
<td>Depth Range</td>
<td>0.5 to 7.0m</td>
</tr>
<tr>
<td></td>
<td>Field of View</td>
<td>40 degrees</td>
</tr>
<tr>
<td></td>
<td>Output Format</td>
<td>NTSC or PAL (4:3 ratio, Standard Definition)</td>
</tr>
<tr>
<td>Sync. Generator</td>
<td>Output Signal</td>
<td>SD/HD Video Generation</td>
</tr>
</tbody>
</table>

for each frame: left and right images from the stereoscopic cameras, and a color image and a depth map from the depth camera. Figure 3.3 shows the four images acquired by the hybrid system. In the hybrid system, the resolution of captured stereoscopic images is 1920×1080, and the resolution of captured depth and color images from the depth camera is 720×486.

![Four images from hybrid system](image)

Figure 3.3: Four images from hybrid system.
3.2 Generation of Depth Map Using Hybrid System

Figure 3.4 describes the overall framework of the proposed ROI enhanced depth map generation. In order to clearly explain the methodology of the proposed scheme, we define image terminologies used in this paper in advance. In the defined image terminologies, a color image and a depth map naturally have the same resolution as the depth camera. On the other hand, the other images have the same resolution as the stereoscopic cameras.

- Left image: an HD image captured by the left HD camera
- Right image: an HD image captured by the right HD camera
• Depth image: an SD depth map captured by the depth camera

• Color image: an SD color image captured by the depth camera

• Initial disparity map: a disparity map obtained by stereo matching with left and right images

• Initial ROI disparity map: a disparity map obtained by stereo matching with left and right images

• ROI disparity map: a disparity map interpolated by a hole-filling algorithm with an initial ROI disparity map

• ROI enhanced disparity map: a disparity map generated by merging the initial disparity map with an ROI disparity map

• ROI enhanced depth map: the final depth map generated by a disparity to depth conversion with an ROI enhanced disparity map

Before capturing four synchronized images using the hybrid camera system, we carry out camera calibration for the three cameras independently, and then calculate the relative camera information on the basis of the position of the depth camera. At the side of the stereoscopic cameras, the left and right images are rectified and color-segmented. To generate an initial disparity map of the left image, we apply a stereo matching algorithm on the color-segmented left and right images.

At the side of the depth camera, we first reduce optical noise in the depth map using a depth data enhancing technique. Thereafter, we carry out 3-D image warping
to move the depth information obtained by the depth camera into the world coordinate, and then reproject the warped depth information onto the left camera to generate the initial ROI disparity map. Depth holes in the initial ROI disparity map are removed by a hole-filling algorithm to generate an ROI disparity map.

Next, we create the ROI enhanced disparity map by combining the initial disparity map generated by stereo matching with the ROI disparity map generated by 3-D image warping. Finally, an ROI enhanced depth map is generated with the final disparity map using a disparity to depth conversion method.

3.2.1 Relative Camera Calibration

Since we merge three cameras to construct the hybrid camera system, the intrinsic camera parameters of each camera are different as well as extrinsic camera parameters. Therefore, it is necessary to calculate the relative camera information for the hybrid camera set using a camera calibration algorithm.

Basically, camera calibration is executed with pattern images captured by the hybrid camera system. We use a camera calibration algorithm [50] to measure the camera parameters. Since the hybrid camera is composed of three cameras, we independently carry out the camera calibration algorithm three times. As a result, we can get three projection matrices for each camera as Eq. 3.1

\[
\begin{align*}
P_s &= K_s[R_s|t_s] \\
P_l &= K_l[R_l|t_l] \\
P_r &= K_r[R_r|t_r]
\end{align*}
\]  (3.1)
where $P_s$ is the projection matrix of the depth camera generated by its camera intrinsic matrix $K_s$, rotation matrix $R_s$, and transition matrix $t_s$. The term $P_l$ and $P_r$ indicate the projection matrices of the left and right cameras generated by their camera intrinsic matrices $K_l$ and $K_r$, rotation matrices $R_l$ and $R_r$, and transition matrices $t_l$ and $t_r$, respectively.

After estimating the camera information, the left and right images are rectified. Hence, the projection matrices $P_l$ and $P_r$ of the left and right images are changed as

\begin{align}
\tilde{P}_l &= K'_l[R'_l|t_l] \\
\tilde{P}_r &= K'_r[R'_r|t_r]
\end{align}

(3.2)

where $K'_l$ and $K'_r$ are the changed camera intrinsic matrices for the left and right cameras by image rectification, respectively. The term $R'_l$ and $R'_r$ are the changed rotation matrices for the left and right cameras, respectively.

In order to calculate the relative camera information of the three cameras, we convert the rotation matrix $R_s$ of the depth camera into the identity matrix $I$ by multiplying inverse rotation matrix $R_s^{-1}$. Thereafter, we convert the transition matrix $t_s$ of the depth camera into the zero matrix $O$ by subtracting the transition matrix $t_s$. Hence, we can define the new relative rotation and transition matrices for the left and right cameras on the depth camera as

\begin{align}
\tilde{R}_l' &= R'_lR_s^{-1}, \tilde{\tilde{P}}_l = t_l - t_s \\
\tilde{R}_r' &= R'_rR_s^{-1}, \tilde{\tilde{P}}_r = t_r - t_s
\end{align}

(3.3)

where $R'_l$ and $R'_r$ are the relative rotation matrices for the left and right cameras, respectively. $t'_l$ and $t'_r$ are the relative transition matrices for the left and right cameras.
cameras, respectively.

As a result, the relative projection matrices of the three cameras are represented as

\[ P'_s = K'_s [I | O] \]
\[ \tilde{P}'_l = K'_l [\tilde{R}'_l | \tilde{t}'_l] \]
\[ \tilde{P}'_r = K'_r [\tilde{R}'_r | \tilde{t}'_r] \]  

(3.4)

where \( P'_s \), \( \tilde{P}'_l \) and \( \tilde{P}'_r \) indicate the modified projection matrices of the depth camera, the left camera, and the right camera, respectively.

### 3.2.2 Generation of Initial Disparity Map

After acquiring left and right images from the stereo cameras in the hybrid system, we need to do image rectification to make the procedure of stereo matching simple. The left and right images are applied by an image rectification technique to generate their rectified images. Since image rectification makes each epipolar line of the binocular images parallel, we have only to search corresponding points of pixels horizontally during stereo matching. Figure 3.5 shows the result of image rectification.

After image rectification, the rectified left and right images are color-segmented using a graph-based image segmentation algorithm [51]. Before doing color segmentation for each view, we smooth the binocular images using bilateral filtering [52]. The purpose of smoothing prior to doing color segmentation is to remove image noises as much as possible to create more consistent color segments. After image smoothing, each pixel is assigned its own segment generated by graph-based image segmentation. Figure 3.6 shows the result of color segmentation for the rectified left image.
Figure 3.5: Image rectification.
Figure 3.6: Color segmentation.
After color segmentation, we determine the disparity of each segment in the left image by computing the sum of absolute difference (SAD) with its corresponding region in the right image. Then, we relax the assumption that each segment has a single disparity during a disparity smoothing stage by considering disparities of its neighboring segments. Figure 3.7 shows the initial disparity map of the left image.

In this paper, since we mainly focus on enhancing the depth information of ROI in the initial disparity map with the proposed hybrid camera system, any kind of high-performance stereo matching algorithm is acceptable to obtain the initial disparity map.
3.2.3 Depth Data Correction

In practice, since depth information captured by a depth camera is not equal to real one, it is essential to calibrate the captured depths. As shown in Fig. 3.8, we can construct a hybrid camera set combining two RGB cameras with a depth camera. The real depth $Z(i, j)$ of the captured depth $D(i, j)$, the intensity of pixel position $(i, j)$ in a depth map $D$ acquired by the depth camera, is calculated by Eq. 3.5

$$D_{\text{Max}} = \frac{f \times B}{\text{DisparityMin}} - \text{Diff},$$

$$D_{\text{Min}} = \frac{f \times B}{\text{DisparityMax}} - \text{Diff},$$

$$Z(i, j) = D_{\text{Min}} + ((255 - D(i, j) - H_{\text{Min}}) \times \frac{D_{\text{Max}} - D_{\text{Min}}}{H_{\text{Max}} - H_{\text{Min}}}$$

where $D_{\text{Max}}, D_{\text{Min}}, H_{\text{Min}}, H_{\text{Max}}, \text{DisparityMin}$ and $\text{DisparityMax}$ are the min and max distances from the depth camera, the min and max intensities value minus 255.
in the depth map $D$, the min and max disparities on ROI estimated by the left- and right-view images, respectively. $\text{Diff}$ is the distance difference between the multi-view cameras and the depth camera. The terms of $f$ and $B$ indicate the focal length of the left-view camera and baseline distance between left- and right-view cameras or between right- and left-view cameras, respectively.

A depth camera simultaneously captures the photometric and geometric information of target scenes. However, the depth map usually contains discrete and rugged noises, as shown in Fig. 3.9. In addition, the region of boundary occurs mismatches between color and depth information at times.

In order to solve these problems of the depth map, an additional enhancement processing is required after depth range correction [26]. Depth maps are enhanced after performing a series of operations including bilateral filtering, down sampling, linear interpolation and boundary recovery. Since bilateral filtering makes the surface of depth
Figure 3.10: Optical noise reduction.

Figure 3.11: Boundary depth cleaning.
information smooth as maintaining the region of boundary, it is effective to remove optical noises in depth maps. However, it is not enough that the smoothing filtering generates reliable depth information, because some noises are still left. To handle remained optical noises, we down-sample the data of depth maps, and then interpolate depth information with sampled data linearly. Figure 3.10 shows the improved depth map after depth data enhancements.

Especially, critical noises are occurred in the region of object boundary. In order to get rid of the boundary noise, we first find boundary as a unit of 4x4 block. Then, the average value of valid depth data in a block on boundary is calculated. Finally, we remove the noise or replace it with the average value. Figure 3.11 shows that the boundary noise are reduced after boundary depth cleaning.

3.2.4 Generation of ROI disparity map

We carry out 3-D image warping to create the ROI disparity map. In the step of 3-D image warping, we move the depth data acquired by the depth camera to the world coordinate, and then reproject the warped 3-D data into the left camera to get an initial ROI disparity map. When $D_s(p_{sx}, p_{sy})$ is the depth information at the pixel position $(p_{sx}, p_{sy})$ in the depth map, we can regard the pixel $p_s$ $(p_{sx}, p_{sy}, D_s(p_{sx}, p_{sy}))$ as a 3-D point. The corresponding point $p_l$ of the left image is calculated by Eq. 3.6

$$p_l = \tilde{P}_l \cdot P_s^{-1} \cdot p_s$$

(3.6)

where $\tilde{P}_l$ and $P_s^{-1}$ are the relative projection matrix of the left camera and the inverse relative projection matrix of the depth camera, respectively. Here, $p_l$ $(p_{lx}, p_{ly}, 1)$ has
the corresponding pixel position \((p_{lx}, p_{ly})\) of the pixel \(p_s\) in the left image.

In addition, the depth information \(D_l(p_{lx}, p_{ly})\) of \(p_l\) is calculated by Eq. 3.7

\[
D_l(p_{lx}, p_{ly}) = \tilde{t}_l^{\prime} + D_s(p_{sx}, p_{sy})
\]

(3.7)

where \(\tilde{t}_l^{\prime}\) indicates the third value of the relative transition matrix \(\tilde{t}_l\) of the left camera.

Figure 3.12 shows the initial ROI disparity map generated by 3-D image warping. When we compare the depth map in Fig. 3.7 with the initial ROI disparity map, we can notice that the body region of the bear doll is extended to fit with the high-resolution left image by 3-D image warping. We can also notice that some holes occur in the initial ROI disparity map.

ROI of the left image and the initial ROI disparity map do not match correctly on the region of ROI boundaries. The main reason of the mismatch is the differences in reflectivity of IR sensors in the depth camera according to color values. In addition, the incorrectness of the camera calibration result can be the cause of the mismatch. In this thesis, we solve the mismatch problem with the color segmented left image and
the initial ROI disparity map.

In order to correctly detect ROI of the left image, we match the color segmented left image with the initial ROI disparity map. Then, we construct the color segment set for ROI from color segments of the left image by Eq. 3.8

\[
R(s_i) = \begin{cases} 
1, & \text{if } \frac{n(A(s_i))}{n(s_i)} \geq 0.5 \\
0, & \text{otherwise}
\end{cases}
\]  

(3.8)

where \( R(s_i) \) indicates whether the \( i \)th color segment \( s_i \) of the color segmented left image is included in ROI of the left image or not. When \( R(s_i) \) is 1, the corresponding color segment is included in the color segment set for ROI. The term of \( n(s_i) \) is the total count of pixels in \( s_i \), and \( n(A(s_i)) \) is the total count of pixels on the region of initial ROI disparity map \( A(s_i) \) that is matched with the region of \( s_i \). Figure 3.13 shows the color segment set for ROI.

After ROI detection, we refine the initial ROI disparity map from the color segment set. We get rid of outside pixels of ROI from the initial ROI disparity map. As a result,
we can get the refined initial ROI disparity map.

In order to generate an ROI disparity map, we fill holes in the refined ROI disparity map with the pixels generated by linearly interpolating with their neighboring pixels [11]. The hole-filling algorithm is performed as a unit of the color segment by Eq. 3.9

\[
I(R(x, y))_k = \frac{1}{n} \cdot \sum_{i=0}^{W} \sum_{j=0}^{W} I(R(i, j))_k 
\]

where \(I(R(x, y))_k\) is the interpolated pixel value at the \((x, y)\) position of the \(k^{th}\) color segment in the refined initial ROI disparity map \(R\) using the valid neighboring pixel value \(I(R(i, j))_k\) in the \(k^{th}\) color segment. The term \(n\) is the valid number of pixels within a \(W\times W\) window. Since the hole-filling algorithm is performed as a unit of a color segment, the valid pixels in the other color segments do not affect to fill the depth holes in the target color segment. Hence, we can maintain the property of depth smoothness and discontinuity at the same time. Figure 3.14 illustrates the color segment-based hole-filling algorithm. Figure 3.15 shows the generated ROI disparity map.
3.2.5 Generation of ROI Enhanced Depth Map

We combine the initial disparity map with the ROI disparity map to generate an ROI enhanced disparity map. The ROI enhanced disparity map $F$ is created by replacing the depth information of ROI in the initial disparity map $H$ with the depth information of the ROI disparity map $R$ by Eq. 3.10

$$I(F(i,j)) = \begin{cases} I(H(i,j)), & \text{if } I(R(i,j)) = 0 \\ I(R(i,j)), & \text{otherwise} \end{cases}$$  \hspace{1cm} (3.10)

where $I(F(i,j))$, $I(H(i,j))$, and $I(R(i,j))$ are the depth values at the $(i, j)$ position in $F$, $H$, and $R$, respectively. Figure 3.16 shows an ROI enhanced disparity map generated by the hybrid camera system.

Disparity values in the ROI enhanced disparity map are converted into their depth values to generate the final ROI enhanced depth map by Eq. 3.11

$$D(i,j) = 255 - (\text{focal} \times \text{base} \frac{\text{Disp}(i,j) - \text{near}}{\text{far} - \text{near}}) \times \frac{256}{\text{far} - \text{near}}$$  \hspace{1cm} (3.11)

where $D(i, j)$ is the corresponding depth value for the disparity $Disp(i,j)$ at the pixel
position \((i, j)\) in the ROI enhanced disparity map. The term focal and base indicate
the focal length of the left camera and the baseline distance between left and right
cameras, respectively. The term far and near indicate the minimum and maximum
distance in the scene. Figure 3.17 shows the ROI enhanced depth map generated by
the hybrid camera system. The resolution of the final depth map is equal to the image
resolution of the left camera.
3.3 Results and Evaluations

In order to evaluate the proposed scheme, we have constructed a hybrid camera system with two HD cameras and one Z-Cam as the depth camera, as shown in Fig. 3.2. The distance that we can measure as depth information by Z-Cam is from 1.75m to 2.15m. The baseline distance between HD left and right cameras is 20cm.

In our experiment, we capture four synchronized BEAR images using the hybrid camera system. In addition, we captured ACTOR image sequence. Figure 3.18 shows the four synchronized image set of the BEAR images. Figure 3.19, Figure 3.20, and Figure 3.21 show the frames of ACTOR image sequence. Once we have test images, we have generated depth maps at the HD left camera by applying the stereo matching...
Figure 3.19: The 1st frame of ACTOR image sequence.
Figure 3.20: The 2\textsuperscript{nd} frame of ACTOR image sequence.
Figure 3.21: The 3rd frame of ACTOR image sequence.
method using SAD [53] and the belief propagation [51] on the HD left and right images.

Figure 3.22 shows the generated depth map for the BEAR image. In order to evaluate quality of the generated depth map, we make a ground truth depth map of the BEAR image by projecting the depth data acquired by a 3-D scanning device [54] at the HD left camera. As shown in Fig. 3.22(b), we only capture the true depth data for ROI due to the limited capability of the 3-D scanning device. Figure 3.22(c) and Figure 3.22(d) show the depth maps generated by the stereo matching method using SAD and the belief propagation. Figure 3.22(e) and Figure 3.22(f) shows the ROI disparity map and the proposed ROI enhanced depth map for the BEAR image.

In addition, Figure 3.23, Figure 3.24, and Figure 3.25 show the generated depth maps for the 1st frame of the ACTOR image sequence. As shown in the fourth figure, each figure also shows the initial disparity map generated by 3-D image warping with the depth data obtained by the Z-Cam.

In order to measure the performance of the proposed scheme objectively, a quantitative analysis based on the ground truth comparison is used. Here, bad matching means that the depth value is different from the corresponding ground truth depth value by more than one pixel value. The bad matching rate $B_A$ is defined as the percentage of bad matching pixels.

The generated depth map for the BEAR image by stereo matching method using SAD and belief propagation had 85.3% and 50.1% for $B_A$, respectively. On the other hand, the depth map generated by the proposed scheme had 38.9% for $B_A$. As a result, the depth map produced by our hybrid camera system was more accurate by approx-
Figure 3.22: Generated depth map for BEAR image.
Figure 3.23: Generated depth map for the 1st frame of ACTOR image sequence.
Figure 3.24: Generated depth map for the 2nd frame of ACTOR image sequence.
Figure 3.25: Generated depth map for the 3rd frame of ACTOR image sequence.
Figure 3.26: 3-D scene construction of ROI for BEAR image.

As shown in Fig. 3.26(a), when we compare 3-D scenes subjectively with the one approximately 46.4% for $B_A$ than stereo matching method using SAD and by approximately 11.2% for $B_A$ than belief propagation for the BEAR image.

Figure 3.26 and Figure 3.27 show the results of 3-D scene reconstruction for ROI of the BEAR image and ACTOR image sequence. After extracting ROI from each depth map, we have made 3-D scenes for ROI by presenting the pixel position of depth maps and its corresponding depth data as 3-D geometry data. We used hierarchical decomposition of depth maps [35] for the 3-D scene reconstruction.

As shown in Fig. 3.26(a), when we compare 3-D scenes subjectively with the one
generated with the ground truth for the BEAR image, the result of 3-D modeling for ROI of the BEAR image by the proposed scheme more closely resembled the original model than the belief propagation method. Hence, we subjectively notice that the depth map obtained by the proposed scheme has more reliable depth data than the full stereo matching method using SAD and belief propagation.

In addition, as shown in Fig. 3.27, the 3-D surfaces for ROI of ACTOR image sequence were still smoother than ones generated by the other methods. As a result, we can see that the mismatched disparity information occurred during stereo matching were notably reduced by the proposed scheme.
Furthermore, although the image resolution of input depth maps captured by Z-Cam was 720×486, the image resolution of the output depth maps generated by proposed method was 1920×1080. Since we have projected the depth camera data into the HD left camera, the image resolution of the ROI enhanced depth map was equal to the image resolution of the HD left camera. As a result, we could successfully generate high-resolution depth maps using the current low-resolution depth camera.
Chapter 4

Hierarchical Decomposition of Depth Maps

In order to generate dynamic 3-D scenes with a video-plus-depth, we can use the depth map representation using meshes. The mesh-based depth video is a collection of 3-D scenes represented by color images and feature points extracted from depth maps. In this thesis, feature points are depth pixels in a depth map that influence critically on the shape of 3-D scenes. After extracting a small number of feature points from a depth map, we create a 3-D surface with the feature points using mesh triangulation, and then cover the 3-D surface with the color image using texture mapping.

The advantage of the mesh-based representation is the high rendering speed that allows us to enjoy 3-D scenes in real time. However, it is hard to handle the feature point data in the mesh-based depth video due to their irregularity. In this thesis, we propose a new scheme to represent 3-D dynamic scenes using an hierarchical decomposition of depth maps. The hierarchical decomposition not only maintains the regularity of feature point data, but also supports the functionality on quality scalability for dynamic 3-D scene services.

In the hierarchical decomposition, we convert a depth map into three disjoint layers according to the region of edges: regular mesh, boundary, and feature point layers. A regular mesh layer is obtained by down-sampling the depth map. A boundary layer is generated by gathering pixels of the depth map on the region of edges. In order
to generate a feature point layer, we select pixels of the depth map on the region of no edges according to their influence on the shape of a 3-D surface. In addition, we maintain number-of-layer (NOL) information to manage the three disjoint layers structurally. For rendering a frame of 3-D dynamic scenes, we first generate an initial surface utilizing the information of regular mesh and boundary layers according to NOL information. Then, we improve the visual quality of the initial surface by adding the depth information from feature point layer according to NOL information.
Figure 4.1 shows the overall outlines of the proposed hierarchical decomposition of depth maps. In the hierarchical decomposition, we first extract edge from a depth map, and then divide the region of the depth map uniformly. Next, the three layers are extracted from the depth map using down-sampling, quad-tree decomposition, full modeling, and maximum distance algorithms. According to the existence of edge and the number of feature points in a grid cell, which is the unit of this decomposition, we generate NOL information in each grid cell. A graphic render is charge of generating 3-D surfaces with decomposed layer information. A 3-D shape pattern represented by a mesh structure is assigned on each grid cell according to its NOL information. In addition, the constructed 3-D surface in each frame are covered by its corresponding color image using texture mapping.

4.1 Regular Mesh Layer

In a 3-D mesh model, we define the valence of a vertex as the total number of its neighbor vertices or faces. When valences of all vertices in a 3-D mesh model are equal, we call the model as a regular mesh. We apply the concept to obtain regular mesh image from depth maps. First, we can define depth maps $\psi_n$ by Eq.4.1.

$$\psi_n(i,j) = D_n(i,j)$$  \hspace{1cm} (4.1)

where the term of $i$ and $j$ indicates the horizontal and vertical position in a depth map, respectively, and $D(i, j)$ means the intensity at $(i, j)$. The term of $n$ is the frame number of a depth map, and $N$ is the total number of depth maps. $W$ and $H$ are the
horizontal and vertical resolution of depth maps, respectively.

In the hierarchical decomposition, we first extract edges by applying the Sobel filter [55] into the depth map vertically and horizontally. Since depth maps are not disturbed by lights or surroundings, we use them for edge extraction instead of color images. Thereafter, we divide the region of the depth map uniformly according to the size of the grid cell. Figure 4.2 shows region division of the depth map by grid cells. Figure 4.2(b) shows the result of edge extraction from the depth map in Fig. 4.2(a). According to the existence of edges in a grid cell, we divide the depth map into regions of edges and no edges, as shown in Fig. 4.2(c). The region of edges is the set of grid
cells that includes edge information, referred to as edge-grid cells; similarly, the region of no edges is the set of grid cells excluding edge information, referred to as no-edge-grid cells.

We define the size of a grid cell as $2^m2^n$ resolution, such as $16 \times 16$, $8 \times 8$, or $16 \times 8$. Once we choose the size of a grid cell, we should maintain it for each depth map during the hierarchical decomposition of depth maps. When we set the size of grid cell, we should be careful to select the size of a grid cell, because it is inversely proportional to the amount of distortion of generated 3-D scenes. We commonly use the size of a grid cell as $4 \times 4$ or $8 \times 8$.

A regular mesh layer can then be obtained by downsampling the corresponding depth map. When the size of a grid cell is $p \times q$, the regular mesh layer is generated by downsample its depth map with the horizontal sampling rate $p$ and the vertical sampling rate $q$. In other words, we gather the four pixels at the corner of each grid cell to make regular mesh layers in each frame.

Figure 4.3(a) illustrates how to generate a regular mesh layer from a depth map. Figure 4.3(b) shows the rendering result of the wire frame mode for a 3-D surface with a regular mesh layer and its rendering result for a 3-D surface. As shown in Fig. 4.3(b), all valances of the 3-D scene are six, i.e. a regular mesh. However, it is not enough for the initial surface of a 3-D scene because there are serious distortions in the region of edges as shown in the rendering result. When the image resolution of a depth map is $W \times H$ and the size of grid cell is $p \times q$ pixels, the total number of grid cells will be $(W/p+H/q)$, and the total number of triangles will be $2 \times (W/p+H/q)$. 

– 68 –
Figure 4.3: Generation of regular mesh images
4.2 Boundary Layer

A boundary layer includes feature points in the edge-grid cell. In order to regularly extract feature points, we employed four quad-tree modes and a full modeling mode. After uniformly dividing each edge-grid cell into four sub-grid cells, we use the full modeling mode when more than one sub-grid cell includes edges. Otherwise, one of the quad-tree modes is selected according to the location of the sub-grid cell that includes edges. Figure 4.4 shows the full modeling mode and four quad-tree modes: up-left, up-right, down-left, and down-right quad-tree modes. Dark pixels are assigned to the boundary image along the raster scanning order as feature points. Here, we extracted 10 feature points in the quad-tree mode and 21 feature points in the full modeling mode.

A boundary layer is used to refine a 3-D surface generated by a regular mesh layer for the region of edges. Since most of the serious distortions are mainly occurred in
Figure 4.5: Generation of a boundary layer

(a) Boundary layer generation

(b) 3-D scene generated by regular mesh and boundary layer
their areas, we should deal with the region of edges carefully. Since the region of depth information is different between a regular mesh and boundary layers, the boundary layer is independent with its regular mesh layer.

Figure 4.5(a) explains how to obtain a boundary layer from a depth map in case of full modeling mode. We choose 21 feature points to describe a grid cell in a full modeling mode and assign the extracted feature points into the boundary layer along a raster scanning order. Figure 4.5(b) shows the rendering result of the wire frame mode for a 3-D surface with a regular mesh and boundary layers and its rendering result for a 3-D surface.

Since we add the boundary surface generated by the boundary layer into the regular mesh surface generated by the regular mesh layer, we can reduce the distortions occurred in the previous 3-D surface. In addition, we note that we can regenerate quad-tree and a full modeling mode only if we recognize the depth pixels and the type information. In other words, the vertical and horizontal positions for depth pixels can be predicted by the size of a grid cell and the resolution of depth maps, and then we can generate the two modes with the type information and a boundary layer.

4.3 Feature Point Layer

After obtaining regular mesh and boundary layers, we generate feature point layers from a depth map in each frame. A feature point layer includes feature points in the no-edge-grid cell. As we deal with the region of edges to generate a boundary layer in each frame, we handle the region of no edges with feature point layers. Basically,
feature point layers are used to enhance the visual quality of the region of no edges in the 3-D scene.

In order to determine the influential feature points in the no-edge-grid cells, scores of all pixels in the grid cell except for the pixels on the boundary were estimated using a maximum distance algorithm. The most influential feature points of each cell were then gathered into the 1st feature point layer. Likewise, the second influential feature points were also gathered into the 2nd feature point layer; this process was repeated for all subsequent points. Figure 4.6 shows the generation of feature point images.

Unlike regular mesh and boundary layers, the x- and y-coordinate information should be considered to accurately represent feature point positions and depth values.
In this Thesis, in order to omit the coordinate data, we defined four representatives to assign extracted feature points into the feature point layer. When we divided the no-edge-grid cell into four sub-grid cells, the center position of each sub-grid cell becomes the position of each of the four representatives. The feature points extracted from the no-edge-grid cells are then mapped into the closest representative. Since the 3-D surface of the no-edge-grid cell is smooth, we can successfully describe the surface with one or two feature points in the grid cell [56].

When we extract feature points, we employ a maximum distance algorithm. Figure 4.7 shows the flow chart to extract feature points constantly. The proposed feature point extraction method will be strongly affected by the quality of depth maps. For
example, if there are noises in a depth map, the feature points tend to be converged into the noises. In general, dominant stereo algorithms or depth acquisition techniques using a depth camera utilized smoothness and consistency constraints to obtain geometrically continuous depth information on the region of no edges. Therefore, the algorithm was operated with geometrically continuous depth data on the region of no edges.

After generating initial surfaces with regular mesh and boundary layers, we improve visual qualities of the 3-D initial surfaces by adding the depth values in feature point layers. With depth values in feature point layers, we generate more complex surfaces in the region of no edges using mesh triangulation as a unit of a grid cell. Figure 4.8(a)
shows the wire frame for the final surface generated by regular mesh, boundary, and feature point layers. Figure 4.8(b) shows the 3-D scene generated by covering the 3-D surface with a color image using texture mapping. As shown in Fig. 4.8(c), the 3-D surface for the body of a man is enhanced comparing with the surface in Fig. 4.5.

4.4 Number of Layer for 3-D Scene Rendering

We utilize NOL to manage the structural data of dynamic 3-D scenes represented by video-plus-depth. A NOL is composed of structural data, which are the number of feature points in a grid cell and the type information in the boundary layer. By doing that, we can recognize the elements of a 3-D surface in each frame, such as the number of feature points and the region of edges from a NOL information. Figure 4.9 show an example of NOL information.

We assign the number from 0 to 21 in the NOL images to indicate mode information and needed decomposed images. As shown in Table 4.1, the NOL from 1 to 4 indicate four quad-tree modes for edge-grid cells. The NOL information for the full modeling mode is 5. In addition, the NOL from 6 to 21 indicate the number of feature point images and the location of representatives of feature points to describe no-edge-grid cells. Especially, when the NOL is 0, we only use 4 feature points extracted from a regular mesh image.

Table 4.2 shows the number of needed feature points and triangles and the shape of the reconstructed 3-D surface for grid cells according to NOL information. When there are no feature points in a no-edge-grid cell, we create the 3-D surfaces with only 2
Figure 4.9: NOL information

Table 4.1: Layer and mode information induced by NOL

<table>
<thead>
<tr>
<th>NOL</th>
<th>Required Layers</th>
<th>Mode</th>
<th># of Feature Point</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>RML</td>
<td>-</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>RML + BL</td>
<td>Up-left</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>RML + BL</td>
<td>Down-left</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>RML + BL</td>
<td>Up-right</td>
<td>14</td>
</tr>
<tr>
<td>4</td>
<td>RML + BL</td>
<td>Down-right</td>
<td>14</td>
</tr>
<tr>
<td>5</td>
<td>RMI + BL</td>
<td>Full modelling</td>
<td>25</td>
</tr>
<tr>
<td>6-21</td>
<td>RMI + FPL</td>
<td>-</td>
<td>5-8</td>
</tr>
</tbody>
</table>
triangles. For the other no-edge-grid cells, of which NOL are from 6 to 21, we generate the 3-D surface using from 4 to 10 triangles with from 5 to 8 feature points extracted from a regular mesh image and feature point layers. For edge-grid cells represented by quad-tree modes, we generate the 3-D surface using 20 triangles with 14 feature points extracted from a regular mesh and boundary layers. For edge-grid cells represented by the full modeling mode, we generate the 3-D surface using 44 triangles with 25 feature points.

Table 4.2: 3-D shape pattern assigned into grid cells

<table>
<thead>
<tr>
<th>NOL</th>
<th># of feature points</th>
<th># of triangles</th>
<th>Shape of surface</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>14</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>14</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>14</td>
<td>20</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>25</td>
<td>44</td>
<td></td>
</tr>
<tr>
<td>6~21</td>
<td>5</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>10~13</td>
<td>6</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>14~17</td>
<td>7</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>18~21</td>
<td>8</td>
<td>10</td>
<td></td>
</tr>
</tbody>
</table>

In order to render 3-D dynamic scenes, we define their initial surfaces with regular mesh and boundary layers according to NOL information. After constructing initial 3-D surfaces, we will meet serious distortions, holes, close to the region of edges owing to the difference of depth values in between regular mesh and boundary layers. For
preventing holes, exceptional processing is required to fill out them, as shown the shape of surface for NOL from 1 to 5 in Table 4.2. When depth values on boundaries in a grid cell are different with depth values in the regular mesh layer, we carry out a mesh triangulation to fill the regions. With such an exceptional process, we prevent serious distortions. Figure 4.10 shows the result of filling holes. After generating initial surfaces, we improve visual qualities of the 3-D initial surfaces by adding the depth values in feature point layers.

4.5 Results and Evaluations

We have tested the performance of our proposed scheme with two test sequences. Test data were Home-shopping, Breakdancers, and ACTOR sequences. Home-shopping sequences have 100 frames with 720×480 resolutions, while Breakdancer sequences have 100 frames with 1024×768 resolutions. Depth maps for Home-shopping are captured by
a depth camera, Z-Cam. On the other hand, depth maps for Breakdancer sequences are obtained by a stereo matching algorithm. ACTOR sequences are generated by the hybrid camera system introduced in Chapter 3. The image resolution of ACTOR sequence is 1920×1080.

![Test sequences](image)

(a) Homeshopping  
(b) Breakdancers  
(c) ACTOR

Figure 4.11: Test sequences

The experiment was implemented on an Intel based PC (Dual 3.2GHz Pentium 4 Xeon, 2GB DDRRAM) under Microsoft Windows XP. Figure 4.11 shows the test
sequences. In this experiment, we set the size of grid cell as 16×16. In Homeshopping and Breakdancers, we generated regular mesh, boundary, and the 1st feature point layers with NOL in each frame.

Figure 4.12 and Figure 4.13 show the result of 3-D scenes for the 12th, and 23rd frame of Homeshopping generated by the hierarchical decomposition, respectively. The upper images in each figure are the wire frame images of the corresponding 3-D scenes in Homeshopping, and the lower images are the result of texture mapping. Figure 4.14 and Figure 4.15 also show the result of 3-D scenes for the 60th and 87th frame of Breakdancers generated by the hierarchical decomposition, respectively. Figure 4.16, Figure 4.17, and Figure 4.18 also show the result of 3-D scenes for the 1st, 2nd and 3rd frame of ACTOR sequence, respectively.

Since we handled the region of edges carefully to maintain the visual quality of 3-D scenes, the more number of triangles were assigned into the edge-grid cells than no-edge-grid cells. In addition, we could notice that the region of no edges were also improved by feature point images. Figure 4.19 shows the rendering result of dynamic 3-D scenes for the two test sequences.

We compared our proposed scheme with 3-D full modeling. In general, the 3-D full modeling generates a 3-D scene with all depth pixels in a depth map. In this experiment, we used 25 pixels to represent each grid cell, of which the size is 4×4, as full modeling model in the boundary layer do. As shown in Table 4.3, we could increase rendering speed for consecutive 3-D scenes more than about 20 times comparing with the 3-D full modeling, because our scheme rendered the region of edges and no edges
Figure 4.12: the 12th frame of Homeshopping
Figure 4.13: the 23rd frame of Homeshopping
Figure 4.14: the 60\textsuperscript{th} frame of Breakdancers
Figure 4.15: the $87^{th}$ frame of Breakdancers
Figure 4.16: the 1st frame of ACTOR
Figure 4.17: the 2\textsuperscript{nd} frame of ACTOR
Figure 4.18: the 3rd frame of ACTOR
In the 3-D full modeling, since there were tremendous amount of triangles to render, it needed a more time to generate 3-D surfaces than the proposed scheme. However, the advantage of the 3-D full modeling is much simpler in the aspect of the design of rendering device than the proposed method. Although the 3-D full modeling only used one pattern to generate 3-D scene, our scheme used ten patterns according to NOL information. Even so, since the proposed hierarchical decomposition reduce the number of triangles of 3-D scene efficiently without serious visual quality degradation, the rendering speed was higher than the 3-D full modeling method.

In addition, we evaluated the quality of depth maps generated by decomposed layers. Table 4.4 presents the variations of the visual quality for the interpolated depth of a 3-D scene adaptively.

![Figure 4.19: The rendering result of dynamic 3-D scenes](image)
Table 4.3: Rendering speed comparison

<table>
<thead>
<tr>
<th>Test Data</th>
<th>3-D Full Modeling</th>
<th>Proposed Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td># of Tri.</td>
<td>Frame rates</td>
</tr>
<tr>
<td>Homeshopping</td>
<td>172,800</td>
<td>0.65 fps</td>
</tr>
<tr>
<td>Breakdancer</td>
<td>402,432</td>
<td>0.51 fps</td>
</tr>
</tbody>
</table>

Table 4.4: Depth quality evaluation according to the size of grid cell

<table>
<thead>
<tr>
<th>Test Data</th>
<th>PSNR (dB), when QP in H.264/AVC is 30</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Original</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Homeshopping</td>
<td>45.02</td>
</tr>
<tr>
<td>Breakdancer</td>
<td>42.47</td>
</tr>
</tbody>
</table>

maps according to the grid cell size. Figure 4.20 shows the depth map generated by 3-D full modeling and the proposed method using linear interpolation, when the size of grid cell is 4×4 and 8×8. Figure 4.21 shows the depth map when the size of grid cells is 16×16. As can be seen in Fig. 4.21, the larger the size of the grid cell, the greater the visual degradation of the interpolated depth map. Hence, this implies that we need to carefully select the grid cell size, according to network situations and target applications.
Figure 4.20: Visual quality estimation

Figure 4.21: Visual quality estimation (grid cell: 16x16)
Chapter 5

Applications

5.1 Generation of ROI Enhanced Multi-view Depth Map

We can extend the idea of ROI enhanced depth map generation introduced in Chapter 3 to create multi-view video-plus-depths by combining a multi-view video camera set and a single-view depth camera [57]. In order to obtain the multi-view depth map, we perform 3-D image warping to project depth information obtained by the depth camera onto each camera in the multi-view camera set. Then, the segment-based hole-filling algorithm is repeated as many as the number of video cameras to generate the multi-view ROI depth map using each warped depth information. Figure 5.1 shows the constructed hybrid camera set combining three video cameras, as the multi-view camera, and a depth camera.

Figure 5.2 describes the overall framework for multi-view ROI enhanced depth map generation using the hybrid camera system. First, we carry out camera calibration independently for all cameras in the hybrid camera set. Then, we calculate relative camera information for the multi-view camera on the basis of the camera information of the depth camera. Thereafter, we rectify the multi-view image acquired by the multi-view camera set using the estimated camera information in each frame [58], and the rectified multi-view images are color-segmented. Next, we carry out 3-D image
Figure 5.1: Hybrid camera set combining multi-view cameras with a depth camera.

Figure 5.2: Overall framework for multi-view ROI enhanced depth map generation using a hybrid camera system.
warping to move the depth camera data to the world coordinate and to reproject the warped depth data into each camera in the multi-view camera set. The projected depth data and their corresponding regions in the multi-view image are used to detect each-view ROI and to generate the multi-view ROI depth map via depth hole-filling. On one hand, we measure the depths of each-view background of the multi-view image using a stereo matching algorithm based on color segmentation. Finally, the ROI enhanced multi-view depth maps are created by merging the multi-view ROI depths and background depths in each frame.

There are three steps to obtain ROI depths using depth camera data: 3-D image warping, ROI detection, and depth hole filling. In the step of 3-D image warping, the depth camera data are moved onto the world coordinates, and then the warped depths are projected onto each camera in the multi-view camera set. When we generate a depth map at the $k^{th}$ camera of the multi-view camera, the pixel position $(u, v)$ in the $k^{th}$-view depth map corresponding the pixel position $(i, j)$ in the depth map $D$ acquired by the depth camera is calculated by Eq. 5.1

$$p_k = P_k \cdot P_d^{-1} \cdot p_d$$  

(5.1)

where $p_k$ and $p_d$ are 3-D points including the pixel position $(u, v)$ in the $k^{th}$-view depth map and the corresponding pixel position $(i, j)$ in the depth map $D$, respectively. $P_k$, and $P_d^{-1}$ are the projection matrix of the $k^{th}$ camera and the inverse projection matrix of the depth camera. Warped depth information $D(u, v)_k$, the intensity at the pixel position $(u, v)$ in the $k^{th}$-view depth map corresponding the depth information $D(i, j)$
at the pixel position \((i, j)\) in the depth map \(D\), is calculated by Eq. 5.2

\[
D(u, v) = \text{Diff}_k + D(i, j)
\]  

(5.2)

where \(\text{Diff}_k\) is the distance difference between the \(k^{th}\) camera and the depth camera in the Z-axis direction.

The warped depth map does not correctly match with its corresponding color image on ROI due to the physical characteristic of depth sensors and the slight incorrectness of camera calibration. To solve the mismatch, we color-segment the each-view image, and then overlap the warped depth map onto its color segmented image. By doing that, we can extract a color segment set for ROI in the each-view color segmented image by Eq. 5.3

\[
R(s_i)_k = \begin{cases} 
1, & \text{if } \frac{n(A(s_i))_k}{n(s_i)_k} \geq 0.5 \\
0, & \text{otherwise}
\end{cases}
\]  

(5.3)

where \(R(s_i)_k\) indicates whether the \(i^{th}\) color segment \(s_i\) in the \(k^{th}\)-view color segmented image is included on ROI or not. When \(R(s_i)_k\) is 1, the color segment is included on \(k^{th}\)-view ROI. \(n(s_i)_k\) is the total count of pixels in \(s_i\) in the \(k^{th}\)-view color segmented image, and \(n(A(s_i))_k\) is the total count of pixels on the region of the warped depth map \(A(s_i)\) that is matched with the color segment \(s_i\) in the \(k^{th}\)-view color segmented image.

After ROI detection, we fill depth holes in the warped depth map with the pixels generated by linearly interpolating their neighboring pixels in each view to generate multi-view ROI depth maps. In order to obtain depth information of background, a segment-based stereo matching method is applied onto multi-view images. Each
color segment in the $k^{th}$-view image is matched with the corresponding region in the $(k-1)^{th}$- and $(k+1)^{th}$-view images to calculate its initial disparity. The initial disparity of each color segment is then refined by disparities of its neighboring color segments. The refined disparities are then converted into depths by disparity-to-depth conversion.

After acquisition of ROI and background depth maps, we merge the two depth maps into one to generate an ROI enhanced depth map in each view.

In our experiment, we captured a multi-view test image, BEAR, using the hybrid camera set in Fig. 5.1. The BEAR image consists of three-view color images acquired by three video cameras with image resolution 1920×1080 and a depth map acquired by a depth camera with image resolution 720×486. Figure 5.3(a) shows the three-view color images and the depth map of the BEAR image. Figure 5.3(b) and Figure 5.3(c) show the result of multi-view depth maps generated by belief propagation and the proposed method, respectively.

In addition, we generate intermediate views using the multi-view depth maps generated by the proposed scheme. As shown in Fig. 5.4, boundary regions in our depth maps are clearer, especially regions of the background curtain in the BEAR image have more accurate depth data; therefore, we can generate intermediate views more faithfully.

5.2 Quality Scalable 3-D Video

In order to serve the 3-D video to consumers successfully in the transmission and display environment, it is necessary to develop the multi-layer framework to support
Figure 5.3: Results for ROI depth estimation.
Figure 5.4: Intermediate view generation.
quality scalability. In other words, we should be able to control the amount of required data for the dynamic 3-D scene within the framework of multi-layer representation according to target applications. For example, we have only to provide the base layer of 3-D contents to consumers in mobile applications. Progressive meshes [59] give us an efficient solution on how to represent hierarchical layers for 3-D mesh models. However, we need much effort to control the mechanism for the scalable dynamic 3-D scene service. What’s more, we suffer from dealing with the irregular 3-D information at times.

We also proposed a system to support the functionality on quality scalability for 3-D video using a hierarchical decomposition [60]. Figure 5.5 shows the entire architecture for providing quality scalable services for 3-D video. The proposed system can be divided into two parts; a sender and a receiver. At a sender side, color images and their depth maps are obtained from either a depth camera system or a multi-view camera system, as well as from the hybrid system introduced in Chapter 3. Depth maps are decomposed into four types of special images. Then, we merge the disjoint images of each depth map into an image and the merged images are coded by H.264/AVC. Finally, bit streams of 3-D videos are transmitted to a receiver through a channel.

At a receiver side, transmitted bit streams will be decoded so as to recover four types of images of depth maps and color images. In order to construct multi-layer representation for 3-D videos, we regard NOL, regular mesh and boundary images as a base layer, and feature point images as enhancement layers. According to consumers’ capabilities and target applications, a layer selector determines the amount of required
Figure 5.5: System architecture for quality scalable services of 3-D video

Data to represent 3-D natural videos. Finally, we can provide consumers with 3-D surfaces constantly with multi-view images or stereoscopic images.

Basically, multi-layer representation of 3-D dynamic scenes is divided into two layers: a base layer and enhancement layers. A base layer means the minimum required data to generate 3-D dynamic scenes and enhancement layers are additional data to improve the visual quality of the 3-D surfaces generated by the base layer. In the hierarchical decomposition of depth maps, we regard initial surfaces generated by regular mesh, boundary, and NOL images as a base layer, and the depth information in feature point image as enhancement layers. Figure 5.6 explains the multi-layer representation of 3-D dynamic scenes using the hierarchical decomposition.
We tested the performance of our system with several depth maps and texture image sequences. Tested sequences were Home-shopping, Break-dance and Ballet. In this experiment, we used $16 \times 16$ pixels as the size of grid cell. For all test sequences, we generated a base layer and two enhancement layers. Figure 5.7 shows the result of base layers of the $60^{th}$, $70^{th}$, and $61^{st}$ frame of test sequences in an order, respectively.

As we mentioned, we constructed minimum required surfaces of 3-D natural videos with NOL, regular mesh, and boundary images. As shown in Fig. 5.7, we could maintain the reliable visual qualities with a base layer when we compared with a full modeling using all depth data in a depth map. The main reason that we could maintain visual qualities was that we processed the region of boundaries carefully using a quad-tree structure and a full modeling technique adaptively.

Figure 5.8 and Figure 5.9 show the result of enhancement layers for the $60^{th}$, $70^{th}$, and $61^{st}$ frames of test sequences. We added enhancement layers to improve visual
Figure 5.7: Results of base layers

Figure 5.8: Results of base layers and one enhancement layer
qualities from a base layer. In Fig. 5.8, we inserted one enhancement layer, i.e. a feature point per a grid cell. We could notice that the visual qualities of 3-D natural videos improved better than ones with a base layer in Fig. 5.7. Furthermore, we could notice that the surfaces reconstruct by a base layer and two enhancement layers in Fig. 5.9 were more complex and better visual quality than ones reconstructed by a base layer and an enhancement layer in Fig. 5.8.

5.3 3-D Video Contents Generation for Realistic Broadcasting Services

Recently, the more advanced broadcasting approach has been introduced, so-called as realistic broadcasting [61]. Realistic broadcasting is expected to provide immersive 3-D broadcasting contents for users to experience realism from them through our five senses directly, not only acoustic and visual sensation. Fundamentally, a realistic broadcasting system will support various user-friendly interactions, such as free view-
point changing and free composition with computer graphic data. Especially, we can touch and manipulate the immersive contents with a haptic device [62].

As one application of the proposed method in this thesis, we introduce a new scheme to generate dynamic 3-D contents for the realistic broadcasting service using a high-resolution color camera and a current low-resolution depth camera. The proposed hybrid camera set produces high-resolution video-plus-depth to represent a photo-realistic dynamic 3-D object. In addition, we introduce a 3-D video generation system based on the MPEG-4 multimedia framework [63] to support a variety of direct interactions in realistic broadcasting. Our main contribution is to develop an explicit solution to generate a high-quality and high-resolution dynamic 3-D contents using our hybrid camera set. In addition, we design and insert a new node for the depth video in the MPEG-4 system to provide a practical solution to stream 3-D broadcasting contents while supporting free view-point changing, free composition with computer graphic models, and even haptic displays.

Figure 5.10 illustrates the overall system architecture of the proposed 3-D video contents generation [?]. At the sender side, we combine a high-resolution color camera and a low-resolution depth camera to construct a hybrid camera set. Before acquiring output images from the hybrid camera set, we carry out camera calibration for the two cameras, and then calculate the relative camera information on the basis of the position of the depth camera. After camera calibration, we carry out a series of image processing steps including depth data enhancing, 3-D image warping, boundary cleaning, and depth hole filling to generate a high-resolution depth map sequence, a depth video,
using the hybrid camera set. A high-resolution color image sequence, a color video, is
directly captured by the high-resolution color camera.

These color and depth videos are then spatio-temporally combined with other multi-
media, such as computer graphics models, using the MPEG-4 Binary Format for Scene
(BIFS) to create dynamic 3-D scenes. The MPEG-4 BIFS is a scene descriptor that
contains the spatio-temporal relationship between each multimedia object and some
interactivity information [65]. On one hand, the color and depth videos are compressed
by H.264/AVC coders independently, and other multimedia data and the scene descrip-
tion information are compressed by their coders. Finally, these compressed bitstreams

Figure 5.10: Proposed 3-D video content generation system
are multiplexed into one bitstream in the MPEG-4 system.

At the client side, we extract the color video, depth video, scene description information, and other multimedia data from the transmitted bitstream. Thereafter, we construct 3-D surfaces by applying a mesh triangulation algorithm on depth video data, and the constructed 3-D surfaces are then covered by the color video data to represent a dynamic 3-D object. In addition, other multi-media data are combined with the dynamic 3-D object by referring the transmitted scene description information, MPEG-4 BIFS. As a result, we can experience various interactions with the immersive content at home through a 3-D display device and a haptic device.

In order to model a dynamic 3-D object, such as a home-shopping host, we acquire high-resolution video-plus-depth data using a hybrid camera set. Figure 5.11 shows the main components of the proposed hybrid camera set.

Basically, the hybrid camera set consists of a high-resolution color camera and a low-resolution depth camera, and each camera is connected to a personal computer equipped with a video capturing board. In addition, a clock generator is linked to the camera set to provide synchronization signals constantly. The hybrid camera set produces three synchronized 2-D images in each frame: a high-resolution color image from the color camera, and a low-resolution color image and its depth map from the depth camera. In this thesis, we construct the video-plus-depth structure with the color images captured by the color camera and their corresponding depth maps generated by the hybrid camera set.

The procedure is equal to the method in chapter 3 in that we use 3-D image warping
to generate high-resolution depth maps. However, since the hybrid camera system includes only a RGB camera, we can only create ROI depth maps. In other words, there is no way to get background depths. Figure 5.12 shows the generation of ROI depth map for representing a dynamic object, such as a home-shopping host.

After obtaining spatially high-resolution video-plus-depth, we construct dynamic 3-D scenes using hierarchical decomposition of depth maps. Figure 5.13 shows the result of 3-D scene reconstruction represented by video-plus-depth.

The high-resolution depth video generated by the hybrid camera set is coded by a H.264/AVC coder [66], though other conventional video coders, such as MPEG-2, can be used. The color video is also coded by another H.264/AVC coder. As a result, we generate two video bit-streams to represent the video-plus-depth structure.
Figure 5.12: Depth map generation of a dynamic object.
At the client side, the video streams are decoded and used to represent a dynamic 3-D object. By regarding the position of pixel and its depth value in the depth map as 3-D geometry information, we construct the 3-D surface using a mesh triangulation algorithm for each frame. Thereafter, the constructed 3-D surface is covered by the corresponding color image, which is regarded as a texture. When the depth value is equal to zero, we ignore the region during 3-D model rendering.

In order to deliver 3-D contents, a multimedia framework is needed. However, traditional multimedia frameworks such as MPEG-1 and MPEG-2 merely deal with efficient coding issues and synchronization between the video and audio, not provide interactive functionalities. Hence, we direct attention to and focus on an MPEG-4 multimedia framework that supports streaming data for various media objects and
provides flexible interactivity in broadcast applications.

A major difference in MPEG-4, with respect to previous audio-visual standards, is the object-based audio-visual representation model that underpins the MPEG-4 multimedia framework. In the MPEG-4 multimedia framework, an object-based scene is built using individual objects that have relationships in space and time. Based on this relationship, the MPEG-4 framework allows us to combine a variety of media objects, such as video-plus-depth and computer graphics models, into a scene to provide interactivity to users.

The MPEG-4 system defines the scene description, referred to as BIFS. The MPEG-4 BIFS defines how the objects are spatio-temporally combined for presentation. All visible objects in the 3-D scene are described within the Shape node in MPEG-4 BIFS.

The Shape node should have both appearance and geometry information; the appearance is expressed by the color video through a *MovieTexture* node. However, although the geometry should be expressed by the depth video, MPEG-4 BIFS does not support this geometry. Therefore, we designed a new node representing the depth video, referred to here as a *DepthMovie* node. In this paper, a new *DepthMovie* node that can be stored in the geometry field is designed as described in Fig. 5.14(a).

The upper four fields of the *DepthMovie* node are the same as the fields of the *DepthImage* node [67] that indicates the camera parameters. The texture field can store depth video as geometry through a *MovieTexture* node that usually indicates the 2-D rectangular video. Then, the corresponding color video is stored on the texture field of the Appearance node. In this way, these two nodes can describe a 3-D surface.
Figure 5.14: Node design for video-plus-depth.

(a) Node specification

```
DepthMovie
{
    field SFVec2f  fieldOfView  0.785398 0.785398
    field SFFloat  nearPlane   10
    field SFFloat  farPlane    100
    field SFBool   orthographic TRUE
    field SFTextureNode texture NULL
}
```

(b) An example of description of video-plus-depth

```
Shape {
    appearance Appearance {
        texture MovieTexture { url "colorVideo.h264" } }
    geometry DepthMovie {
        texture MovieTexture { url "depthVideo.h264" } }
}
```
Figure 5.14(b) shows an example describing video-plus-depth using the *DepthMovie* node.

In general, computer graphic models are represented by the 3-D mesh structure. The 3-D mesh models can be described using predefined nodes in MPEG-4 BIFS. The MPEG-4 BIFS data including scene description information and computer graphic model data are coded by the BIFS encoder provided by the MPEG-4 system.

Thereafter, the compressed color video, depth video, and MPEG-4 BIFS streams are multiplexed into an MP4 file that is designed to contain the media data of an MPEG-4 presentation. The MP4 file can be played from a local hard disk and over existing IP networks. In addition, a streaming server, which can stream MPEG-4 content over the Internet in real time or on demand, is used to stream the MP4 file. Hence, viewers can enjoy the media contents in the context of a video-on-demand concept.

In order to evaluate the proposed scheme and to create a practical example of realistic broadcasting, we focused on a home-shopping scenario. Since the home-shopping channel shows a special product for consumers and aims to sell the advertised product, it will be fit to test its broadcasting contents as an example of interactive and immersive contents. To make interactive 3-D home-shopping contents, we have constructed a hybrid camera set with one high definition (HD) camera, Canon XL-H1, as the high-resolution color camera, and a depth camera, Z-Cam, as the current low-resolution depth camera. Figure 5.15 shows the shot using the hybrid camera set.

The synchronized output images captured by our hybrid camera set were a color image with image resolution 1920×1080 from the HD camera, a color image with image
resolution 720×486 from Z-Cam, and a depth map with image resolution 720×486 from Z-Cam for each frame. We can see two HD cameras in the hybrid camera set in Fig. 5.15. In this work, we only used the left camera of the two as a high-resolution color camera. The right camera was used to evaluate the depth quality later. In our experiment, we have generated 3-D contents with image resolution 1920×1080.

In order to import HD video-plus-depth and computer graphic models in the MPEG-4 multimedia framework, we employed GPAC packages [68] that were open sources for the MPEG-4 multimedia framework to use research and academic purposes. In addition, the GPAC packages provided an MPEG-4 player called as Osmo. Therefore, after converting 3-D home-shopping contents into an MP4 file, we enjoyed it in the Osmo player.

Figure 5.16 shows the created 3-D home-shopping content played by the Osmo player. As shown in Fig. 5.16(a), a home-shopping host introduced a sofa as an advertised product, which was represented by a 3-D mesh model composed of 4,774 vertices and 2,596 triangular faces. Thereafter, the sofa was appeared at the left
position of the home-shopping host, as shown in Fig. 5.16(b). Finally, the sofa moved to the center of the scene for users to watch its shape in detail and interact with it, as shown in Fig. 5.16(c).

In the comparison with the first home-shopping content in 2005, since the home-shopping host was modelled with HD video-plus-depth generated by the proposed hybrid camera set and the background was captured by the HD color camera, the second home-shopping content was high-resolution broadcasting contents. In addition, it was possible for us to stream the home-shopping contents through a network after setting up a streaming server, Darwin Streaming Server (DSS) [69], because we had described dynamic 3-D scenes with MPEG-4 BIFS and generated the 3-D contents in the MPEG-4 multimedia framework.

The second home-shopping content could provide a variety of direct interactions, such as free-viewpoint changing and haptic interaction as the first one had done. Since we modelled the home-shopping host, background, and even subtitles using mesh trian-
Figure 5.17: Free view-point changing.

Figure 5.18: Other interactions.

gulation, users could change the viewing position of the content freely by real-locating
the position of a virtual camera. Figure 5.17 shows the result of free view-point chang-
ing.

In addition, since the 3-D scene was described using MPEG-4 BIFS, the dynamic
3-D home-shopping host could be easily combined with computer graphic models, a
soft and subtitles, as shown in Fig. 5.18(a). Furthermore, users could touch and
manipulate the sofa in the 3-D con-tent with a haptic device using a haptic rendering
algorithm [62], as shown in Fig. 5.18(b).

– 115 –
In June 2007, the 3-D home-shopping content was demonstrated at the ITRC Forum held at an exhibition center in Korea. The exhibition event attracted IT experts and academics, as well as the general public. Although we needed to guide for inexperienced visitors to enjoy the 3-D content, most attendees in the event showed their interests in watching the 3-D home-shopping host in the 3-D scene by changing their view points and in touching the advertised product through a haptic device.

However, some visitors had troubles to manipulate the 3-D home-shopping content and to understand the concept of realistic broadcasting services at times. Especially, there were some comments from visitors about the limited range of view points and the unnatural haptic information. We are planning on receiving and considering their comments for the more advanced realistic broadcasting system.
Chapter 6

Conclusions

In this thesis, we have proposed a new scheme to generate high-quality and high-resolution depth maps using a hybrid camera system. With the hybrid camera system, we could solve inherent technical problems in the currently available depth camera system. Especially, we have presented a 3-D image warping technique and a disparity map merging scheme to generate ROI enhanced depth maps. We have noted it by the objective and subjective evaluations that the proposed scheme could produce more accurate depth maps than conventional stereo matching algorithms. The proposed scheme also generated higher resolution depth maps than what current depth cameras could produce.

We also proposed a new scheme to represent 3-D dynamic scene using a hierarchical decomposition of depth maps. With the proposed scheme, we could represent and render consecutive 3-D scenes successfully while supporting the framework of a multi-layer representation, and we also reduced the amount of bits needed to code depth maps. In addition, we increased rendering speed for 3-D dynamic scenes more than about 20 times in comparison with a 3-D full modelling. Furthermore, we could control visual qualities of 3-D dynamic scenes according to target applications. We hope that the proposed hybrid camera system can present new directions for further research related to depth estimation and will be used in future 3-D multimedia applications.
References


– 122 –
36. C.H. Esteban and F. Schmitt, “Silhouette and Stereo Fusion for 3D Object Mod-

Accurate Virtual Environments,” ISPRS Journal for Photogrammetry and Remote

38. S. Zhang and P. S. Huang, “High-resolution, Real-time Three-dimensional Shape


Video View Interpolation Using a Layered Representation,” SIGGRAPH, pp. 600-

41. W. Niem, “Robust and Fast Modeling of 3D Natural Objects from Multiple

42. Y.K. Kim, “Stereo Matching Using Global and Local Color Segmentation based
on Quad-tree Decomposition,” M.S. Thesis, Gwangju Institute of Science and

43. M. Gelautz, “Segmentation-based Stereo and Motion with Occulusions,” Ph.D


