A Study on Efficient Light Field Coding

by **Kohei ISECHI**

Department of Information and Communication Engineering

Graduate School of Engineering

Nagoya University

Abstract

A light field is one of fundamental data formats for 3D image processing. A light field is represented as a 4D signal and has rich visual information in the 3D space. In practical use, a light field can be interpreted as a set of dense multi-view images which have 2D spatial and 2D view coordinates. Thanks to the rich 3D visual information many kinds of applications for 3D image processing can be achieved by using a light field such as depth estimation, free-viewpoint image synthesis, and 3D display. Such applications of a light field allow us to realize a practical 3D image/video system in many fields such as: industry, education, entertainment, and medicine. The light field used for the practical 3D image processing consists of tens-to-hundreds viewpoint images. Additionally, light field images having higher resolution are also required to provide high-quality 3D images. Thus, the amount of data included in the light field becomes quite huge.

One of important tasks in light field processing is efficient light field coding. The huge data size of a light field is inconvenient in many cases such as 3D image/video streaming, storing, and broadcasting. A light field basically has huge redundancy because the distance between viewpoints of a light field is quite small and the light field images are similar to each other. By extracting the redundancy, the amount of light field data can be dramatically reduced. Efficient light field coding methods are essential to realize 3D image/video applications. In this thesis, I explore an efficient coding method for light field compression and consider a practical light field transmission scenario where a light field is reproduced by using a 3D display.

For an efficient light field coding, I consider a novel light field coding scheme. Fujii Laboratory, to which I belong, has proposed a novel light field coding method which represents a light field using the sum of weighted binary patterns. The binary patterns and corresponding weight values are computationally obtained so as to optimally approximate the target light field. This scheme is definitely different from those of modern image/video coding standards such as H.265/HEVC which include complex procedures, e.g. intra/inter-frame pre-

diction and transformations. We have demonstrated that this scheme achieves comparable performance to the modern video coding standards and enables a dramatically simple decoding process. However, its computational complexity for encoding is quite high. To reduce the computational complexity, we have also proposed a progressive coding scheme that progressively encodes the target light field with a small number of weighted binary patterns in a step-by-step manner. The progressive scheme remarkably mitigates the computational complexity, but its rate-distortion performance is slightly degraded. There is a trade-off between rate-distortion performance and computational complexity. To address the above trade-off problem, I design a disparity compensation framework, which can be applied to the progressive coding, to improve the rate-distortion performance while keeping feasible computational complexity. Experimental results demonstrate that the method with disparity compensation improves the rate-distortion performance of the progressive coding without a computational complexity explosion.

For a practical scenario where a light field is transmitted and reproduced by a 3D display, I investigate coding efficiency of a light field. I assume that the light field is reproduced by using a compressive 3D display, which reproduces the light field with several light-attenuating LCD panels stacked in front of a backlight. Viewers can observe different images from different viewpoints because the light rays emitted from the backlight pass through different pixels at each light-attenuating layer depending on the outgoing direction. The transmittance patterns for the layer panels (layer patterns) are calculated so that the display can reproduce a given light field as accurately as possible. Under the scenario, either light fields or pre-calculated layer patterns should be transmitted from the sender to the receiver. It should be noted that the layer patterns by themselves are compressive representations, because the entire light field is reduced into only a few layer patterns. However, it is unclear how much the quality of the reproduced light field is degraded due to the encoding errors of the layer patterns. To clarify this point, I compared the coding efficiencies of the light fields and layer patterns under this communication scenario. Experimental results show that the layer patterns have advantages over the light fields in terms of the rate-distortion performance in a lower bit range.

In short, this thesis proposed a novel light field coding method using weighted

binary patterns and investigated coding efficiency of a light field in the scenario where a compressive 3D display is the receiving terminal. The important contributions in this thesis are as follows. First one is improving the light field coding method which approximates a light field using weighted binary patterns by using disparity compensation framework. Second one is demonstrating coding efficiency of a light field and layer patterns in the practical light field transmission scenario. I believe that these contributions would help to realize a practical 3D image processing system using a light field.

Contents

List of Figures

ix

List of Tables

Chapter 1

Introduction

This chapter describes general research background and a purpose of this study. The overview of a light field is firstly introduced; and then, three-dimensional (3D) display is introduced as one of attractive applications of a light field, which this thesis focuses on. Then, this chapter expounds main issues for handling a light field, i.e. light field coding problems which this thesis tackles. After that, a research purpose and main contributions of this thesis are described. At the end of this chapter, the whole structure of this thesis is summarized.

1.1 Research Background

This world is filled with light rays from various light sources, e.g. the sun, the moon, and street lights. The light rays pass through any point in the 3D space with any direction and intensity. The light rays travel and are reflected by any object in the 3D space; and then, a part of the light rays reaches our eyes. The appearance of the world can be visually observed by receiving the lights rays which reach our eyes. Therefore, visual information can be described by formulating the light rays.

The intensity of light rays reaching our eyes is practically defined as a four-dimensional (4D) signal by introducing three essential assumptions [1–3]. Although the name of 4D signal is differently called "light field" in [1], "lumigraph" in [2], and "ray space" in [3], "light field" is employed in the rest of this thesis. The 4D light field signal can be in-

terpreted as a set of dense multi-view images of which intervals between viewpoints are quite small. The dense multi-view images can be captured by using special camera systems such as multi-camera array systems [4,5] and a light field camera [6–10]. Since the light field signal includes rich visual information in the 3D space, the light field data can be utilized for various kinds of applications such as depth estimation [11–14], free-viewpoint image synthesis [1, 15, 16], and stereoscopic displaying [17–22]. Figure 1.1 shows examples of still light field images with different number of viewpoints. The light fields (a) "Dragon and Bunnies" (b) "friends", and (c) "truck" are provided from [23], [24], and [25], respectively. Nowadays, the light field becomes one of fundamental data formats for 3D image processing.

As one of attractive light field applications, a 3D display is investigated. A 3D display provides 3D scene perception to us based on a mechanism of human vision such as binocular parallax and motion parallax. Thanks to the capability of displaying a 3D scene, a 3D display allows us to have natural spatial perception and is expected to be applied for many fields such as medicine, education, and entertainment. Several architectures have been investigated for implementing a 3D display [19, 26–36]. These architectures basically give motion parallax to viewers with naked eyes by providing different images according to viewpoints. As one of promising 3D-display architectures, a compressive 3D display has been investigated [19, 35–38], which reproduces the light field with several light-attenuating LCD panels stacked in front of a backlight. Figures 1.2 and 1.3 show a structure of the compressive 3D display and its hardware prototype [39]. The transmittance of each layer panel can be controlled pixel by pixel individually. The light rays emitted from the backlight pass through different pixels at each light-attenuating layer depending on the outgoing direction, so that viewers can observe different images from different viewpoints. This means that the display can reproduce a light field, allowing the viewers to perceive motion parallax and to have auto-stereoscopic experiences. The transmittance patterns (layer patterns) displayed on the layer panels are calculated so that the display can reproduce a given light field as accurately as possible. The requirements for displaying a high-

(a) 5×5 light field (Dragon and Bunnies)
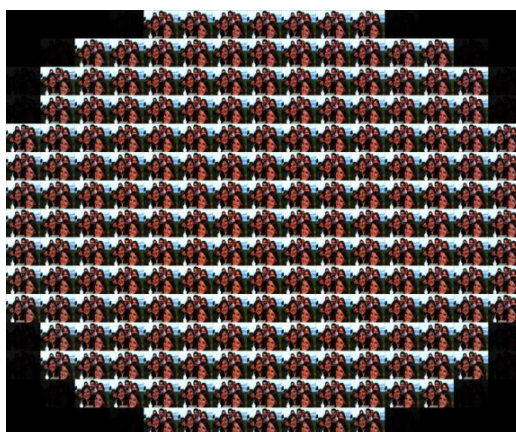
(b) 15×15 light field (friends)
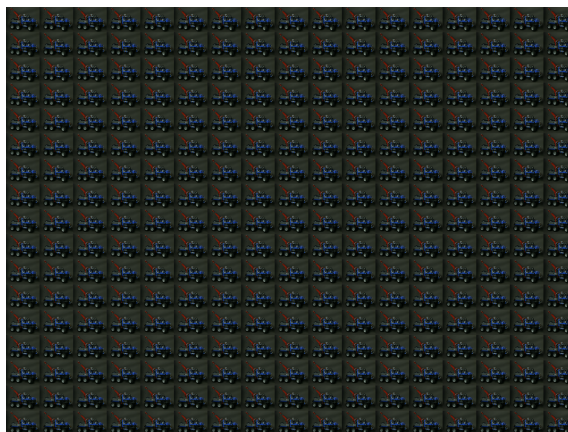
(c) 17×17 light field (truck)

Figure 1.1: Light fields with different number of viewpoints
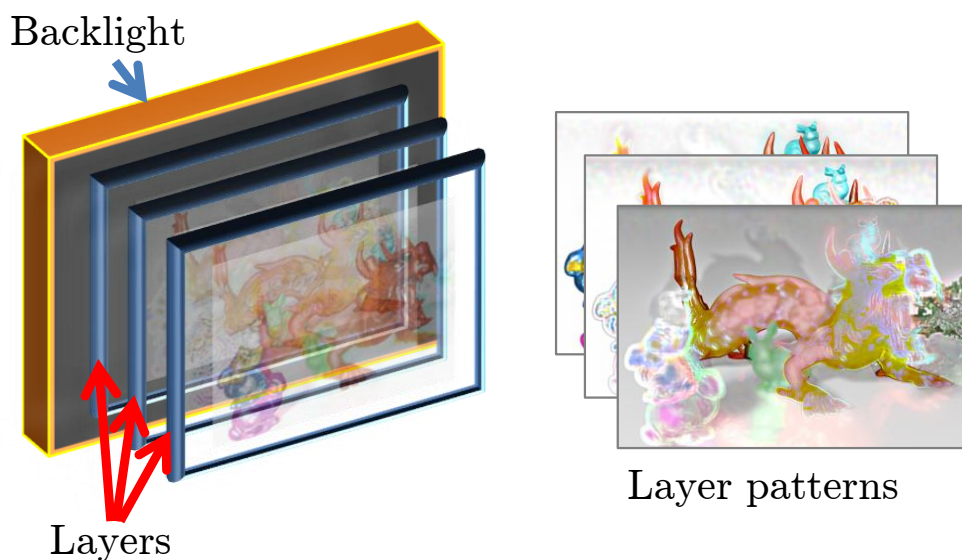
Figure 1.2: Structure of compressive 3D display

Figure 1.3: Hardware prototype of compressive 3D display

quality light field with the compressive display have been analyzed in [21], which revealed that the larger number of the given light field is desirable in order to reproduce a high-quality light field.

One of important issues for handling the light field is huge data size. Light fields used in the practical applications consist of tends-to-hundred multi-view images as shown in Figure 1.1. Light fields with a larger number of viewpoints are desirable to provide high-quality 3D experiences in

the practical applications of a light field described the above. A Light field with 5×5 viewpoints such as the Dragon and Bunnies dataset is slightly insufficient in many practical cases. Increasing the number of viewpoints leads to an increase of the data amount of a light field. In the case of Figure 1.1, the truck dataset has $17 \times 17 = 289$ viewpoints while the Dragon and Bunnies dataset has $5 \times 5 = 25$ viewpoints; thus, the data amount of truck dataset becomes about 8.5 times compared with that of Dragon and Bunnies. Compared to a conventional two-dimensional (2D) still image, the data amount of a 17×17 light field is 289 times. In the case of light field videos, multi-view images are accumulated in time sequence; consequently, its data amount becomes huge soon. When a 17×17 light field video with 10 seconds and 30 frames per second, namely a light field video with 300 frames, is considered, its data amount is 86,700 times compared to a typical 2D video sequence with 300 frames.

Due to the huge data size of a light field, efficient coding of a light field is one of the important research topics for handling a light field. The huge data size of a light field causes difficulty in light field applications. For example, assuming a light field streaming system, it would occupy much larger network bandwidth than a general single-viewpoint video streaming system; thus, efficient light field coding scheme is required. As shown in Figure 1.1, the light field has large redundancy which is different from typical single-viewpoint video because the light field images are similar to each other due to quite small intervals between viewpoints. By extracting the redundancy, the amount of light field data can be dramatically reduced. There are many researches which focus on efficiently removing the redundancy of a light field.

In order to remove the redundancy of light fields, most research uses modern video coding standards such as H.265/HEVC [40] with small modifications [41–46]. The modern video coding standards have been developed for compressing typical 2D videos where various kinds of techniques such as intra/inter-frame prediction, discrete cosine transform (DCT), and arithmetic coding, are employed to eliminate the redundancy of the 2D video sequences. A light field can be regarded as a 2D video sequence by aligning viewpoint images in some order, and the modern video coding

standards can be applied to the light field. In particular, motion compensation techniques, which predict motion of captured objects between sequential frames in the target video, are useful for removing redundancy among the images at different viewpoints in a dense light field. The modern coding standards with small modifications have achieved good coding performance for a light field.

1.2 Research Purpose

This thesis tackles light field coding problems in two directions. First one is addressing general light field coding problems, namely how to represent a light field with small amount of data by removing its redundancy. Second one is considering a light field coding problem in a practical scenario where the compressive 3D display is a receiving terminal.

1.2.1 Light Field Coding Using Weighted Binary Patterns

For the first direction, this thesis proposes a novel light field coding scheme using weighted binary patterns. Fujii Laboratory, to which I belong, has proposed the coding scheme with weighted binary patterns in [47]. In contrast to the coding methods using the modern video cording standards with small modifications, we have developed this scheme based on an idea that the standard video coding techniques are not necessarily the most suitable for a dense light field. The key concept of this scheme is that a light field is approximated by using a linear combination of only several binary patterns. This scheme is named "baseline" as a counterpart of the progressive scheme mentioned later. The binary patterns and weights are obtained by solving an optimization problem, of which an objective function is the difference between a target light field and a light field that is reconstructed as the sum of weighted binary patterns. This concept was originally proposed for generating temporal binary sequences used for multi-view displays with active shutter grasses [48]. However, to the best of our knowledge, there is no research for applying this concept to light

field coding and validating the suitability.

The proposed coding scheme is completely different from these of modern video coding standards, and its decoding process, i.e., calculating the linear combination of binary patterns, is dramatically simpler than that of the standard codecs. The simplicity of decoding process leads to faster and less power-hungry decoder than those of the standard codecs. However, the encoding process, which involves solving the optimization problem, takes much longer than the video coding standards. As the number of binary patterns increases, the encoding time exponentially increases; then, the encoding time soon becomes infeasible. The encoding process can be made computationally lighter by dividing binary patterns and weights into several groups on the basis of the divide-and-conquer strategy. This framework progressively approximates a light field with a small number of weighted binary patterns at each group and can greatly reduce the computational complexity for obtaining optimal binary patterns. Although the approximation accuracy is slightly degraded, the progressive scheme is expected to achieve reasonable rate-distortion performance.

The baseline method and its progressive extension still have a problem of the trade-off between computational complexity and rate-distortion performance. One of the methods for dealing with the problem is the use of disparity compensation, which is to shift the pixels in the images according to a specified disparity value and the viewpoint positions. In the coding schemes using weighted binary patterns, the binary patterns represent only the common components among the images at different viewpoints, while the weights represent viewpoint-dependent components of the light field. If binary patterns can also represent viewpoint-dependent components, the approximation accuracy of the coding scheme would be improved. The disparity compensation can provide the binary patterns with the capability to represent viewpoint-dependent components. The effectiveness of applying disparity compensation to the coding scheme has been preliminarily investigated in [49]. Experimental results show that the disparity compensation improves the approximation accuracy, but the disparity values were empirically determined beforehand. The appropriate disparity values depend on a captured scene, so a method of finding the

appropriate disparity values should be provided in some way.

In this thesis, I propose a disparity compensation framework for the light field coding, which uses weighted binary patterns. I aim at improving rate-distortion performance of the progressive light field coding by adaptively applying disparity compensation while avoiding infeasible computational complexity. The proposed method applies disparity compensation to the binary patterns. An optimization algorithm to adaptively search the best disparity depending on captured scenes is also introduced. Experimental results show that the proposed method improves rate-distortion performance while avoiding infeasible computational complexity if the number of binary patterns increases.

1.2.2 Light Field Coding for Compressive 3D Display

For the second direction, this thesis considers a light field coding scenario where the compressive 3D display is adopted as a receiving terminal. As mentioned previously, the requirements for providing a high-quality light field with the compressive display have been analyzed [21]. Based on the requirements, an end-to-end system for displaying a real 3D scene using a compressive display with three LCD panels has been developed [50]. This system captures a light field with multi-view camera or light field camera and displays it with a compressive 3D display. Although a practical system for a compressive 3D display is investigated, coding efficiency of a light field has never been discussed in such a practical system. Additionally, there are many studies on efficient light field coding as also mentioned previously, but they also do not carefully consider a practical scenario for light field transmission.

In this thesis, I investigate coding efficiency of a light field in a light field transmission system where a compressive 3D display is adopted as the receiving terminals. In this scenario, two light field transmission frameworks are supposed. In the first framework, target light field is directly transmitted to the receiver side and layer patterns are calculated using the received light field and it is reproduced by a compressive 3D display.

In the second framework, layer patterns are calculated in advance at the sender side and transmitted, and a compressive 3D display in the receiver side reproduces the light file using the transmitted layer patterns directly. It should be noted that the layer patterns by themselves are compressive representations, because the entire light field (tens to hundreds of images) are reduced into only a few layer patterns. However, the layer patterns are different from natural images, and thus, it is unclear how much the encoding errors of the layer patterns affect the quality of reproduced light fields. This thesis reveals the coding efficiency of the light field and the layer patterns in the practical scenario through simulative experiments. Experimental results show that the second framework achieves better rate-distortion performance in lower bit ranges.

1.3   Contributions

There are two main contributions in this thesis. The first contribution is to improve a light field coding scheme, which uses weighted binary patterns, by applying disparity compensation framework. I propose an optimization algorithm for the method of applying disparity compensation. When disparity compensation is applied to each of the binary patterns in a straightforward manner in the optimization problem, it is quite difficult to solve the problem because it includes three sets of unknown values: binary patterns, its corresponding weights, and disparity values applied to the binary patterns. By combining the progressive scheme with disparity compensation, the optimization problem at each group can be solved with brute-force search on a set of candidate disparity values because the number of binary patterns in each group is small. The proposed method finds the best disparity value at each group so that the encoding result adaptively uses good disparity values depending on target light fields. Furthermore, disparity compensation is not considered in the baseline method using weighted binary patterns; thus, the proposed method outperforms not only the progressive framework but also the baseline one in some datasets.

The second contribution is to examine the compression efficiency of a

light field in the practical light field transmission scenario where a compressive 3D display is adapted as the receiving terminal. As mentioned previously, coding efficiency of a light field in practical scenarios has never been discussed. Although several 3D display architectures have been investigated, I focus on the compressive 3D display as a promising 3D display device and suppose a practical scenario using the compressive 3D display; then, I reveal that the coding efficiency of a light field and layer patterns through simulative experiments.

1.4 Outline of Thesis

The rest of this thesis is organized as follows.

Chapter 2 Background Knowledge and Related Works. This chapter describes background knowledge and related works of this study to get a better understanding of this thesis. First, definition of a light field is introduced, and then, 3D display technologies are reviewed as an attractive light field application. Then, video coding standards, which can be applied to a light field, and their applications for a light field are briefly described.

Chapter 3 Light Field Coding with Weighted Binary Patterns. This chapter describes a light field coding scheme using weighted binary patterns and its extension using disparity compensation. First, basic concept and algorithm of the coding scheme are introduced; and then, acceleration method for the scheme is described. Then, I expound a proposed method of applying disparity compensation to binary patterns in the proposed scheme. Finally, the effectiveness of the proposed method is validated through experiments.

Chapter 4 Light Field Coding for Compressive 3D Display. This chapter considers two light field transmission frameworks under a scenario where light field is finally reproduced by using a compressive 3D display. Coding efficiency of the two frameworks is investigated by simulative experiments. To evaluate the coding efficiency from the various viewpoints, the simulative experiments are conducted in several conditions and datasets.

Chapter 5 Relationship to Real-World Data Circulation. This chapter discusses the relationship between real-world data circulation and this study. This study is supported by a special curriculum for Ph.D. students called real-world data circulation leaders, program for leading graduate schools, Nagoya University [51]. Real-world data circulation (RWDC) is a novel academic field proposed by the curriculum. I first introduce the basic concept of RWDC; and then, I describe the relationship between RWDC and light field processing.

Chapter 6 Conclusion and Future Works. This chapter concludes the whole work in this thesis, and describes the remaining issues of the current work. Based on these issues, I describe several future perspectives for improving the current work.

The organization diagram of this thesis and corresponding publications are illustrated in Figure 1.4.
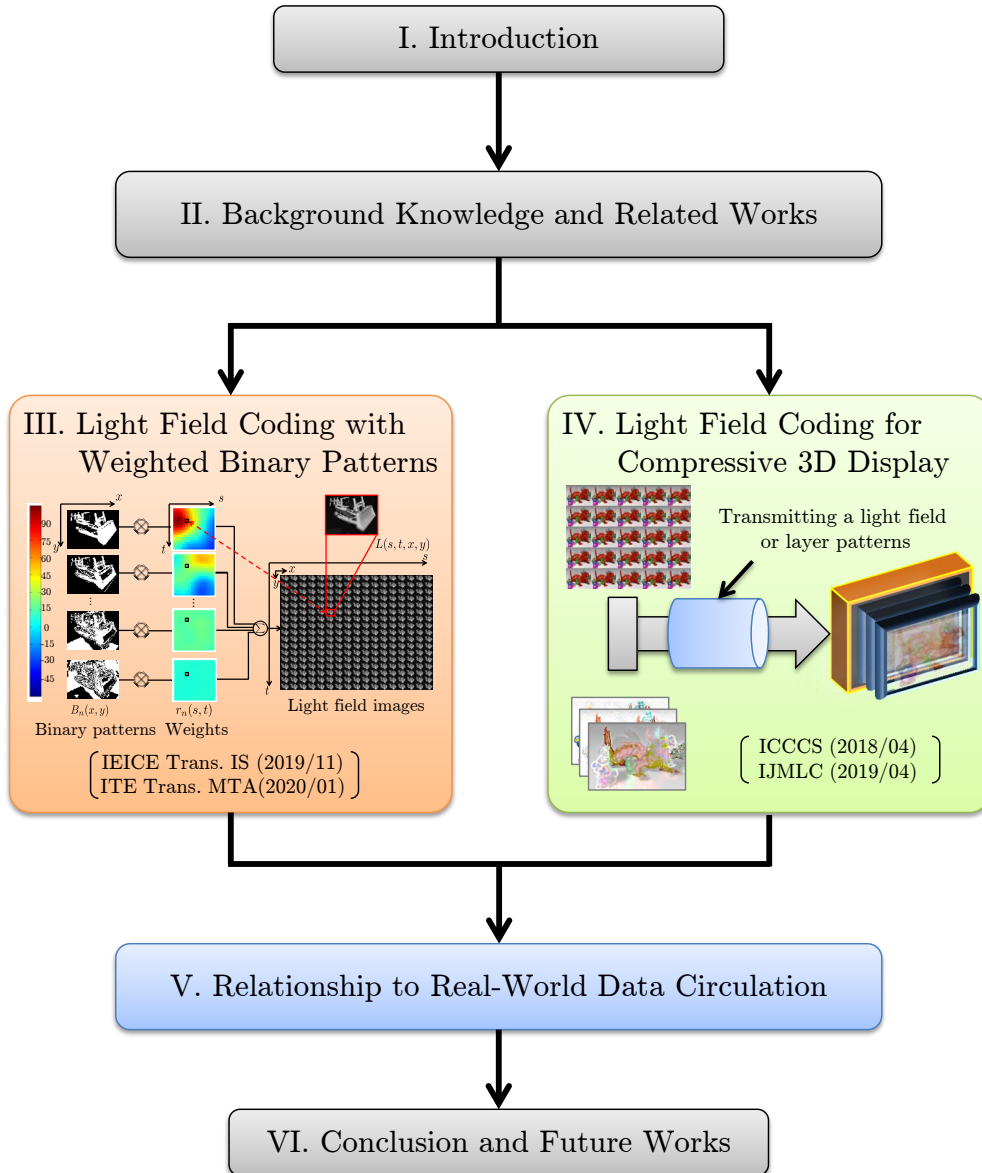
Figure 1.4: Organization diagram of this thesis and corresponding publications

Chapter 2

Background Knowledge and Related Works

This chapter reviews background knowledge and related works of this study in order to get a better understanding of light field coding problems, which this thesis addresses. As the background knowledge, definition of a light field is first introduced. As one of practical applications of a light field, this chapter introduces 3D display technologies because this thesis treats a practical light field coding scenario where a 3D display is a receiving terminal. Then, video coding standard technologies, which is widely used for practically compressing a light field, are briefly reviewed. As related works of this study, several researches applying the video coding standards to light field compression are introduced.

2.1   Definition of Light Field

This section introduces the definition of a light field signal. Adelson *et al.* proposed the plenoptic function [52] which defines intensity of light rays entering an eye as a function P with seven parameters as follows:

$$\mathcal{P} = P(\theta, \phi, \lambda, t, V_x, V_y, V_z), \tag{2.1}$$

where, λ, t, and (V_x, V_y, V_z) indicate the wavelength of light rays, time, and the viewpoint of pupil, respectively. The parameter (θ, ϕ) is equal to the angle with which the light rays pass through the center of pupil.

(a) One-plane representation   (b) Two-plane representation

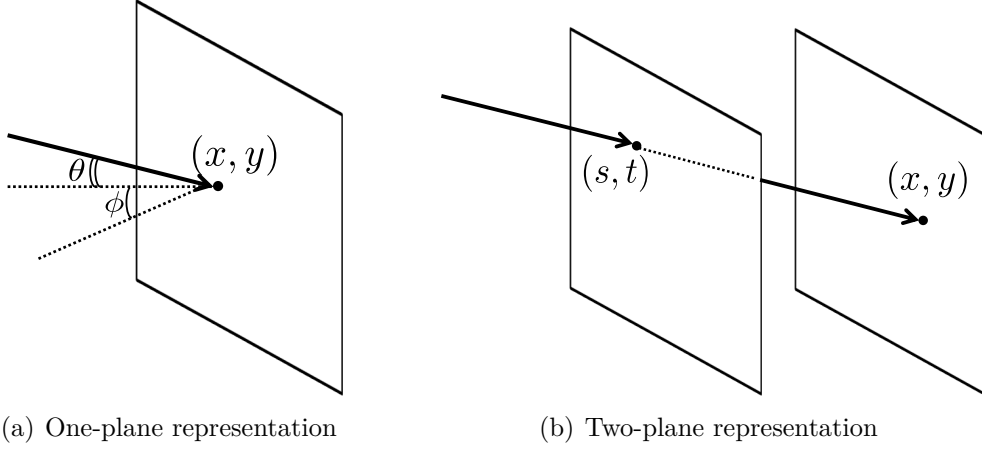Figure 2.1: Parameterization of 4D light filed

Equation (2.1) completely formulates the light rays and $\mathcal{P}$ includes whole of light-rays information, but it is actually overabundant for 3D image processing. For practical usage, static scene is considered. In the static scene, time is unnecessary to be considered, so that t becomes constant. If the color information is handled with RGB three primary colors, the wavelength λ becomes constant. For further assumption, all light rays are not attenuated and interfered. Under this assumption, the parameter V_z can be fixed. Finally, the plenoptic function with only four parameters can be defined under the above simplification. By rewriting V_x and V_y as x and y, respectively, a 4D light-rays signal $\mathcal{L}$ is defined as follows:

$$\mathcal{L} = L(\theta, \phi, x, y). \tag{2.2}$$

I call $\mathcal{L}$ "light field". This light field $\mathcal{L}$ indicates a grayscale signal. To deal with a color signal, several 4D light field signals should be handled simultaneously. For instance, if color information is described with RGB color space, three signals $\mathcal{L}_R$, $\mathcal{L}_G$, and $\mathcal{L}_B$, which denote intensities of R, G, and B components respectively, should be considered. This thesis basically treats a grayscale light field signal, but color light field is considered in Chapter 4.

A 4D light field signal in the 3D space can be defined using one or two reference planes as shown in Figure 2.1. In the one-plane representation
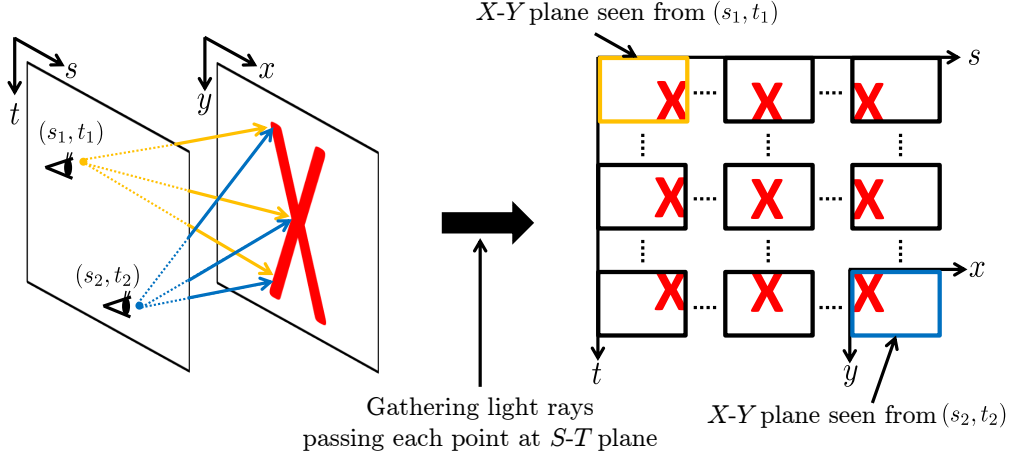
Figure 2.2: Relationship between 4D light field and multi-view images

(Figure 2.1(a)), $L(\theta, \phi, x, y)$ indicates a light ray which passes a point (x, y) on a reference plane with an angle (θ, ϕ). On the other hand, the two-plane representation parameterizes a light ray which passes through a point (s, t) on the left-side plane and a point (x, y) on the right-side plane as shown in Figure 2.1(b). A 4D light field signal with the two-plane parameterization can be rewritten as follows:

$$\mathcal{L} = L(s, t, x, y). \tag{2.3}$$

The two-plane representation can formulate all light rays filling a cube of which the top and bottom sides are the left-side (S–T) and the right-side (X–Y) reference planes, respectively. In the two-plane representation, the 4D light field $L(s, t, x, y)$ can be interpreted as a set of dense multi-view images of which intervals between viewpoints are quite small. Figure 2.2 illustrate the mechanism of this interpretation. When all light rays passing a point (s, t) at the S–T plane are gathered, they constitute scenery of the X–Y plane which human can see from the point (s, t) because they can be interpreted as passing the X–Y plane and reaching human eyes at (s, t). This scenery can be interpreted as an image signal seen from the viewpoint (s, t). A 4D light field can be composed of the set of image signals seen from each point at the S–T plane. The set of image signals is equal to the multi-view images of which intervals between viewpoints are

Display panel

Parallax barrier
(slits)

Viewpoints
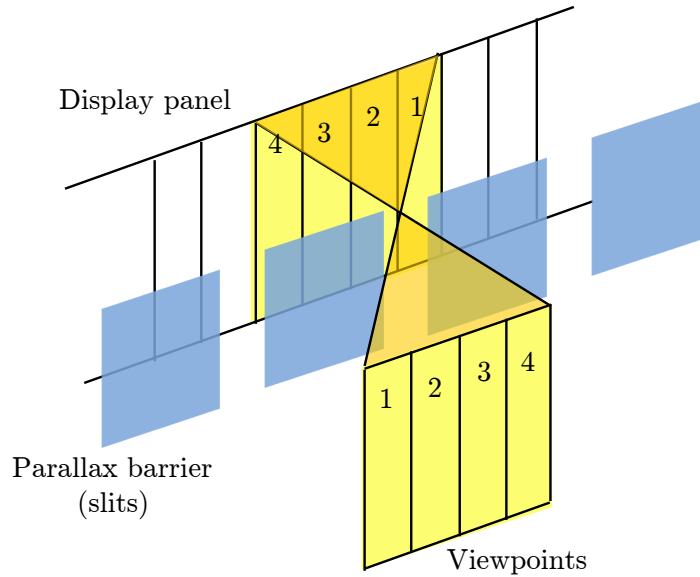
Figure 2.3: Parallax-barrier-based 3D display

quite small. These two parameterizations are fundamentally equivalent and convertible to each other, but the two-plane parameterization is more useful for image processing than the one-plane parameterization because a light field signal can be handled as image data. A 4D light field is basically handled with the two-plane parameterization in this thesis.

2.2 3D Display Technologies

A 3D display is one of light field applications and has attracted much research interests. A 3D display provides 3D scene perception to us based on a mechanism of human vision such as binocular parallax and motion parallax. Thanks to the capability of displaying a 3D scene, a 3D display allows us to have natural spatial perception and is expected to be applied for many fields such as medicine, education, and entertainment. Note that, in this section, a light field is formulated using one-plane representation described in Section 2.1 because it is useful for considering the light rays reproduced by a 3D display.

Several architectures of 3D displays have been developed [19, 26–36].

Figure 2.4: Lenslet-array-based 3D display

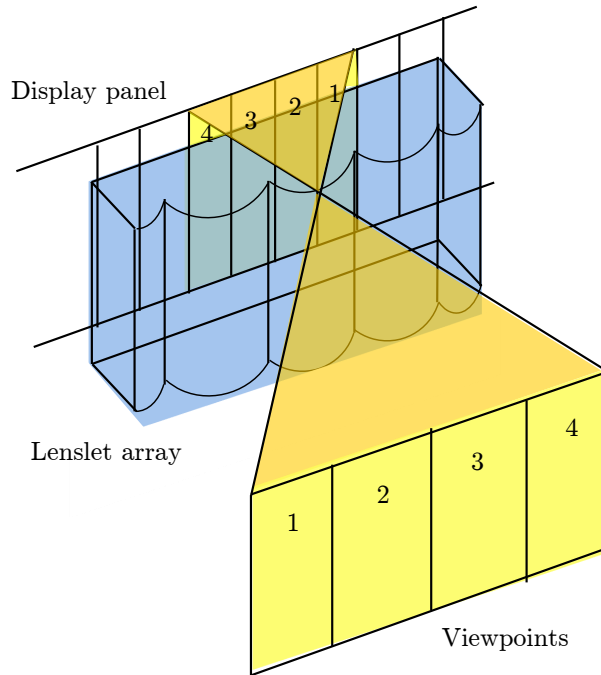These architectures basically give motion parallax to viewers with naked eyes by providing different images according to viewpoints. Figure 2.3 shows the structure of parallax-barrier based 3D display [26–29], which is one of typical glass-free 3D displays. The parallax-barrier-based display has an array of slits in front of a display panel. The array of slits covers a part of display panel depending on viewpoints; namely, viewable pixels on the display panel differs according to the viewpoint. By reproducing different images on each viewable area which depends on viewpoints, viewers can observe different images depending on the viewpoints. As another architecture of glass-free 3D display, a lenslet-based display has been developed [30–34]. Figure 2.4 illustrates the structure of lenslet-based display. The lenslet-based display has an array of micro lenses index instead of the slits used in the parallax-barrier-based display. The micro lenses have special refractive index and viewers at a specific viewpoint can observe only a part of images on the display panel; therefore, the lenslet-based display can reproduce different images depending on the viewpoints.

(a)

Backlight

Layers

(b)

(c)

Front layer   Middle layer   Rear layer
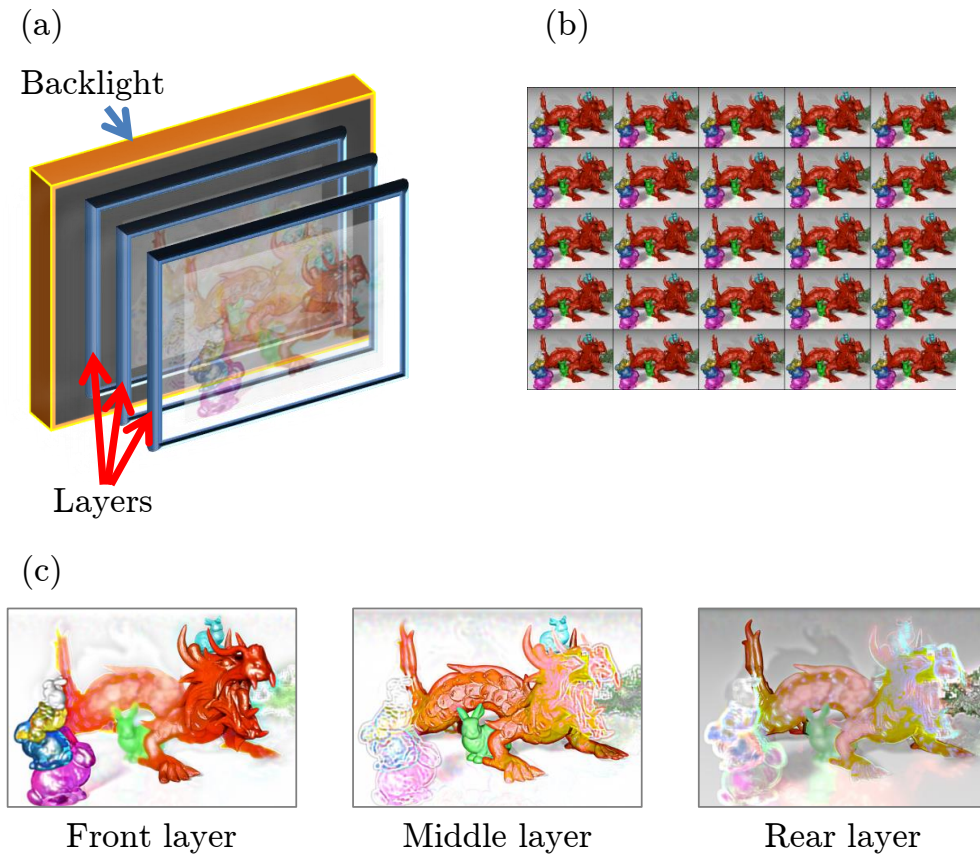
Figure 2.5: (a) Compressive 3D display, (b) Light field images "Dragon and Bunnies", (c) Layer patterns for Dragon and Bunnies

In these architectures, there is a trade-off problem between the number of viewpoints and spatial resolution due to the slits and micro-lens array. Additionally, the intensities of output light rays of parallax-barrier-based display tends to be low because a part of light rays from the display panel are physically obstructed by the slits. As an architecture which overcomes or alleviates the above problems, a compressive 3D display has been investigated [19, 35–38]. This thesis focuses on the compressive 3D display as a reproduction device of a light field signal.

The compressive display consists of a few light-attenuating layers (e.g. LCD panels) stacked on a backlight as shown in Figure 2.5. The transmittance of each layer panel can be controlled pixel by pixel individually. The

$$L(\boldsymbol{\theta}, \boldsymbol{x})$$

Front layer

Middle layer

Rear layer

θ

Backlight

Figure 2.6: Reproduced light rays from compressive display

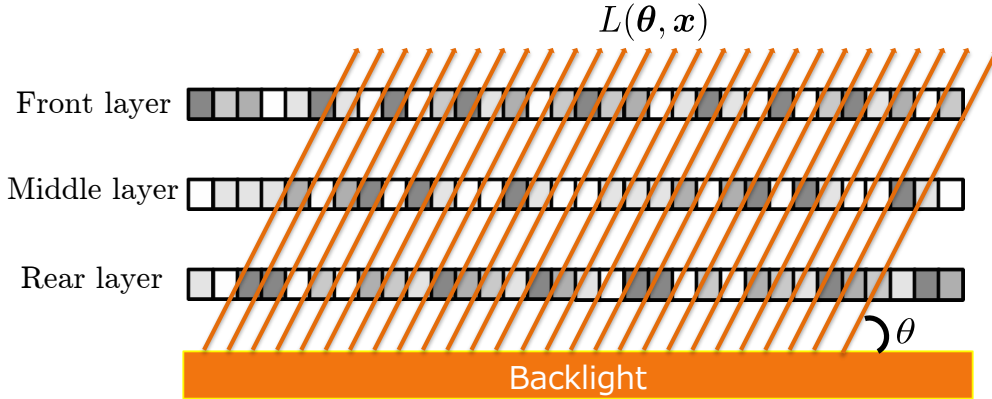light rays emitted from the backlight pass through different pixels in each layer depending on the outgoing directions θ as shown in Figure 2.6; thus, the display can reproduce different images according to the viewpoints. This means that the display can reproduce a light field (dense multi-view images), providing the viewers with auto-stereoscopic images and motion parallax. The transmittance patterns displayed on the layers (layer patterns) are optimized so that the display can reproduce a given light field as accurately as possible. This optimization problem for obtaining the layer patterns is solved by using non-negative tensor factorization. The requirements for displaying a high-quality light field with the compressive display have been analyzed [21], which revealed that the disparity range in the given light field should be limited in order to reproduce the high-quality 3D objects with the compressive display. Based on these requirements, an end-to-end system for displaying a real 3D scene using a compressive display with three LCD panels has been developed [50]. In this system, a real 3D scene is captured by using a multi-view camera or a light field camera such as Lytro Illum [7], and then layer patterns are calculated using the captured light field. In the case of using a multi-view camera, view interpolation using image-based rendering is employed to satisfy the requirement on the disparity range.

The principle of compressive 3D display can be formulated as follows. The following formulation assumes that a light field is a grayscale signal.

A light rays L emitted from the backlight is represented as follows:

$$L(\boldsymbol{x}, \boldsymbol{\theta}) = B_0 \prod_{k \in \mathcal{K}} a_k(\boldsymbol{x} + k\boldsymbol{\theta}), \tag{2.4}$$

here $\boldsymbol{x} \in \mathbb{R}^2$ denotes the image coordinate and $\boldsymbol{\theta} \in \mathbb{R}^2$ is the outgoing direction of the light ray. Symbol $\mathcal{K}$ means the set of indices of layers, $a_k(\boldsymbol{x})$ indicates the transmittance patterns of the k-th layer, and B_0 is the intensity of the backlight. With a fixed direction $\boldsymbol{\theta}$, the ensemble of light rays $L(\boldsymbol{x}, \boldsymbol{\theta})$ corresponds to an image observed from the direction $\boldsymbol{\theta}$, which is denoted as $I_{\boldsymbol{\theta}}(\boldsymbol{x})$. To obtain the transmittance pattern $a_k(\boldsymbol{x})$, an optimization problem shown below is solved with a given set of $I_{\boldsymbol{\theta}}(\boldsymbol{x})$.

$$a_k(\boldsymbol{x}) = \arg \min_{a_k} \sum_{\boldsymbol{x}, \boldsymbol{\theta}} \|I_{\boldsymbol{\theta}}(\boldsymbol{x}) - L(\boldsymbol{x}, \boldsymbol{\theta})\|^2. \tag{2.5}$$

Equation (2.5) can be solved using non-negative tensor factorization (NTF) [19].

To improve the reproduction quality of light fields, time-division multiplexing (TDM) has also been investigated. In TDM framework, the layer patterns are rapidly alternated, and thus, the viewers perceive their average over time. In this case, Equation (2.4) is rewritten as:

$$L(\boldsymbol{x}, \boldsymbol{\theta}) = \frac{1}{T} \sum_{t=1}^{T} \left\{ B_0 \prod_{k \in \mathcal{K}} a_k^{(t)}(\boldsymbol{x} + k\boldsymbol{\theta}) \right\}, \tag{2.6}$$

where T is the number of time division, and $a_k^{(t)}(\boldsymbol{x})$ indicates the transmittance pattern of the k-th layer at time t. These transmittance patterns can be obtained by solving below equation in the same manner as Equation (2.5).

$$a_k^{(t)}(\boldsymbol{x}) = \arg \min_{a_k} \sum_{\boldsymbol{x}, \boldsymbol{\theta}} \left\| I_{\boldsymbol{\theta}}(\boldsymbol{x}) - \frac{1}{T} \sum_{t=1}^{T} \left\{ B_0 \prod_{k \in \mathcal{K}} a_k^{(t)}(\boldsymbol{x} + k\boldsymbol{\theta}) \right\} \right\|^2. \tag{2.7}$$

Although the above formulation assumes that a light field is grayscale, a color light field can be handled by individually applying the above formulation to R, G, and B signals.

2.3 Video Coding Standards and Their Extensions for Light Field

Multi-view images consisting a light field can be regarded as a single-viewpoint 2D video by aligning viewpoint images; thus, video coding standards, which encode a 2D video, can be applied to a light field for removing its redundancy. This section briefly introduces basic knowledge of video coding standards and their applications for light field coding. Along with the development of higher-resolution image/video formats, video coding technology has also been developed over many years. From the 1980s, ITU-T (International Telecommunication Union Telecommunication Standardization Sector) and MPEG (Moving Picture Experts Group) have standardized video coding technologies. Video coding standards started from H.261 established in 1990, and new standards have been continuously established such as H.262/MPEG-2 in 1995 and H.264/MPEG-4 AVC in 2003. The latest version of video coding standards is H.265/HEVC [53] standardized in 2013. HEVC aims to handle 4K/8K videos as the target of coding in order to prepare for emerging applications that employ higher-resolution image/video and consume larger storage capacity and network traffic. HEVC inherits a hybrid coding framework from the previous standards and improves its coding efficiency. The hybrid coding framework includes prediction coding and transform coding that remove redundancy of an input video in time direction and spatial direction. The hybrid coding was initially employed in H.261 and has been continuously improved during the updates of standards without dramatical changes of the framework.

The basic structure of HEVC encoder is illustrated in Figure 2.7. Input video is split into frame images, and a frame is evenly split into blocks called coding tree units (CTUs) as shown in Figure 2.8. The CTUs consist of coding units (CUs), which is further split based on quadtree structure. White blocks are the largest CUs (LCUs) and yellow blocks are partitioned CUs in Figure 2.8. HEVC employs various CU partition patterns and adaptively apply prediction and transform to partitioned CUs. Intra prediction predicts a target block using neighbouring previously encoded

Input: Video

Block
Partition

+

−

Motion
Detection

Motion
compensation

Intra
prediction

Encode
controller

Frame
memory

Quantization
parameter

In-loop
filter

+

+

Transform,
Quantization

Inverse transform,
Dequantization

Intra-prediction
mode

SAO
parameters

Entropy
coding

Motion
parameters

Output:
Bitstream

Figure 2.7: Encoding process of HEVC

Frame partition in HEVC

LCU partitioned based on
quadtree structure
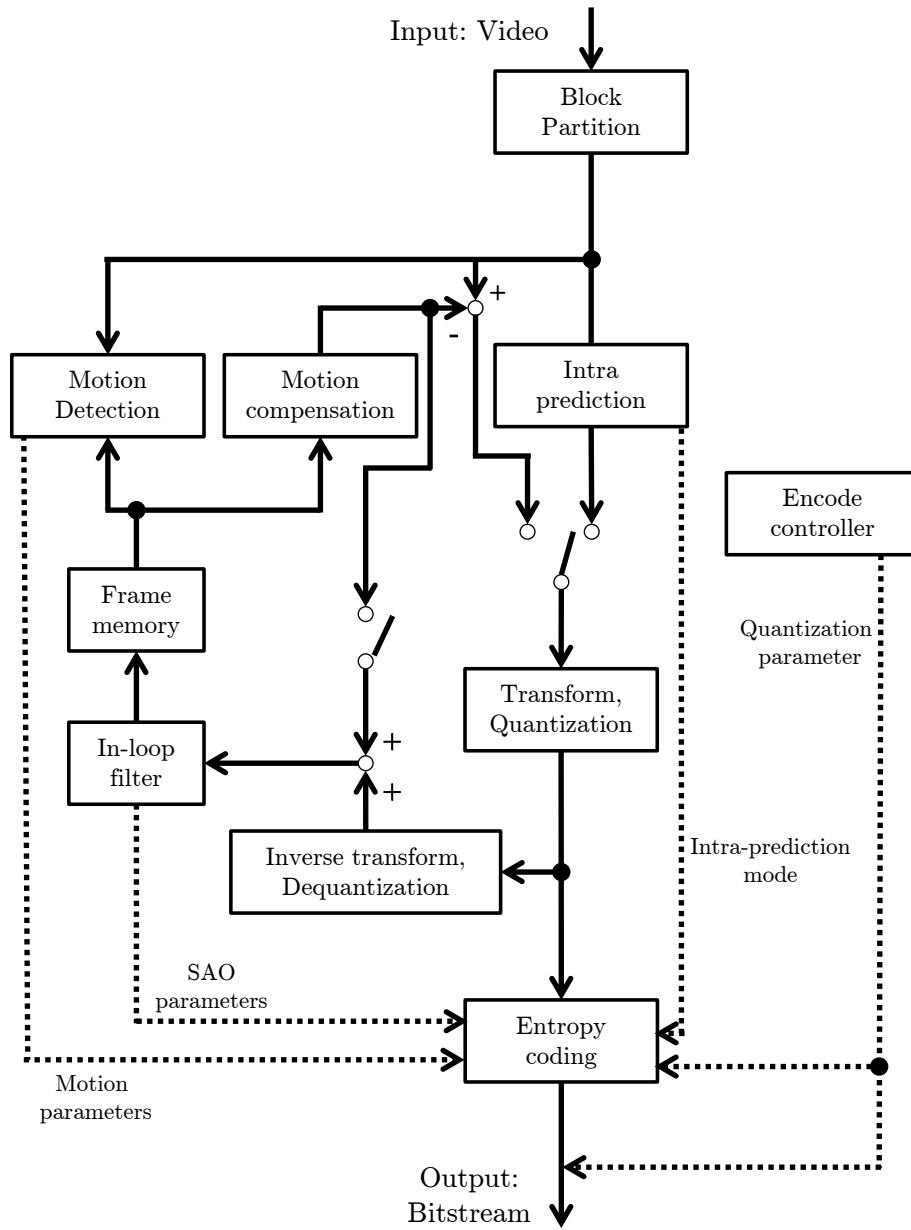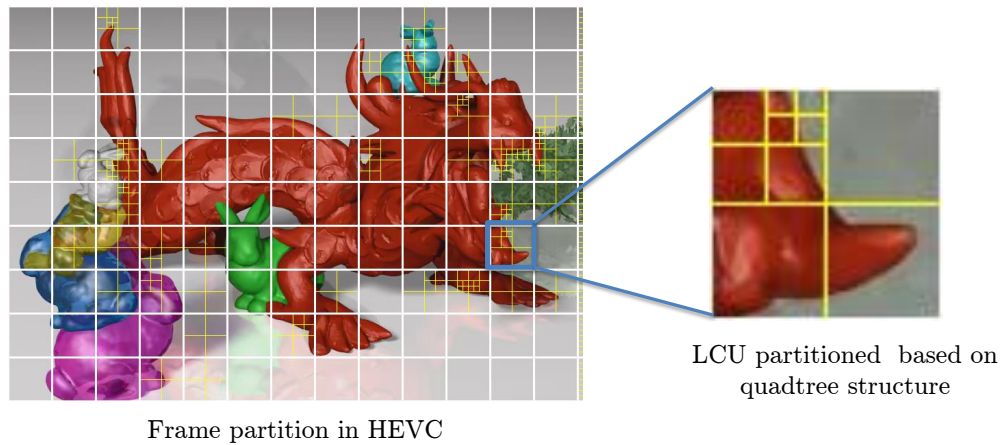
Figure 2.8: Example of frame partition

blocks. Figure 2.9 illustrates examples of intra prediction methods. The
Vertical prediction (a) and horizontal prediction (b) predict the current
block using the pixels from the adjacent macro blocks above and to the
left, respectively. The DC prediction (c) computes the mean pixel value of
the 32 pixels of the above and left macro blocks and assigns the mean pixel
value to all the pixels in the current macro block. Motion compensation
predicts a target block using blocks in other previously-encoded frames.
Figure 2.10 illustrates an example of motion compensation methods. In
this example, the current block is predicted by using a linear combination
of reference blocks at positions different from that of the current block.
Reference blocks are searched to be most similar to the current block from
the reference frames. The difference of block positions is equal to the mo-
tion of objects on the current block; thus, motion compensation predicts
motion of objects between frames. Since motion compensation utilizes the
other frames that do not include the target block, motion compensation is
called inter-frame prediction. These prediction methods are classified as
prediction coding scheme and play an important role in the video coding
standards. The prediction coding contributes to a dramatical reduction of
temporal redundancy of an input video. After the prediction process, an
orthogonal transform such as discrete cosine/sine transform (DCT/DST)
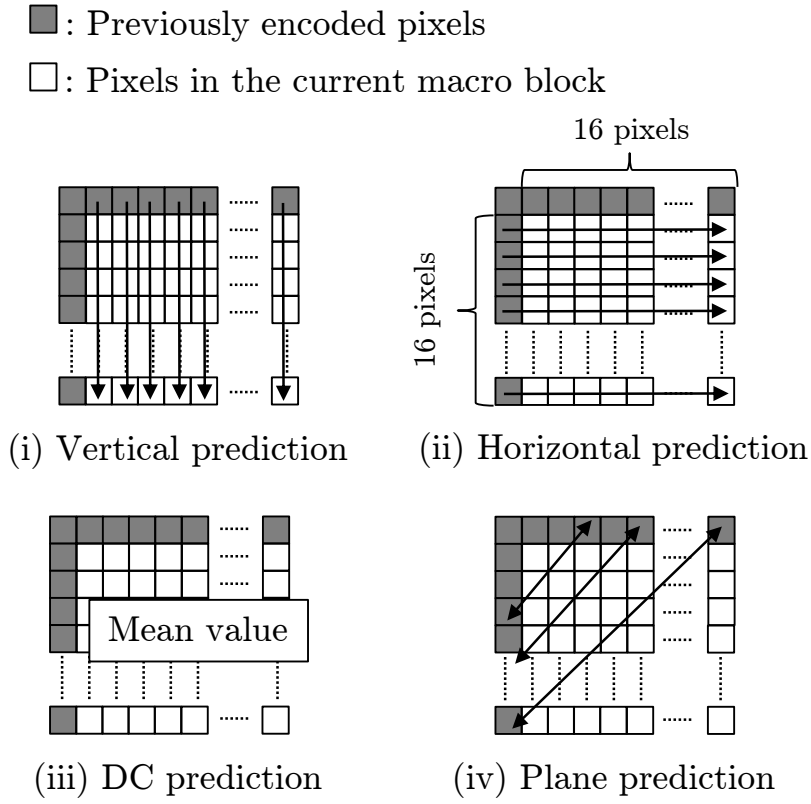is applied to prediction error, and the DCT/DST coefficients are quan-

■ : Previously encoded pixels

□ : Pixels in the current macro block

16 pixels

16 pixels

(i) Vertical prediction

(ii) Horizontal prediction

Mean value

(iii) DC prediction

(iv) Plane prediction

Figure 2.9: Examples of intra prediction method

Block A

+offset o_1

×weight w_{10}

Block B

×weight w_{11}

+offset o_2

×weight w_{20}
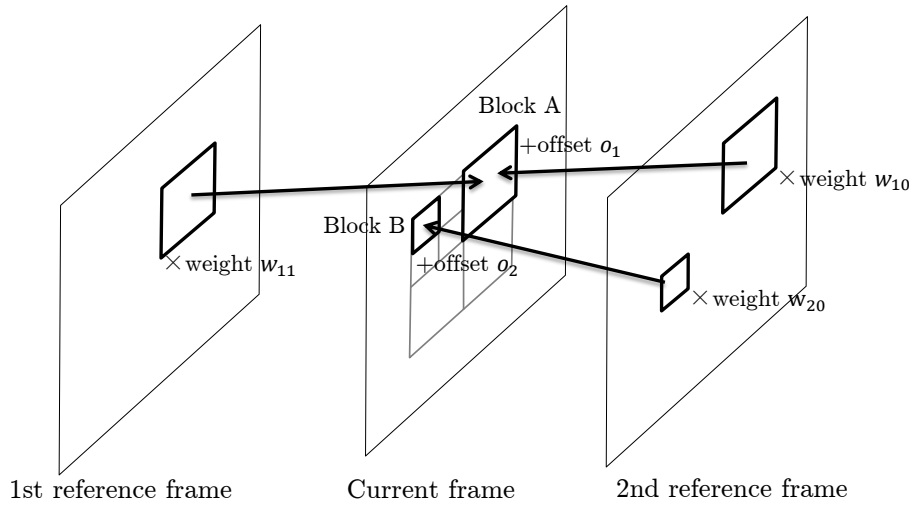
1st reference frame

Current frame

2nd reference frame

Figure 2.10: Example of motion compensation method

tized. Finally, the quantized coefficients and supplementary information (e.g. motion parameters, prediction modes, etc.) are encoded by using entropy coding such as arithmetic coding. The encoded frame is decoded locally in the encoder with smoothing filter called deblocking filter and decoded frame is utilized for performing motion compensation to next or following frames. The encoder repeats the above process frame-by-frame until the all frames of the input video is encoded.

Figure 2.11 illustrates the basic structure of HEVC decoder. HEVC decoder restores a video from a bitstream which is the output of HEVC encoder. The decoding process basically follows encoder process in reverse order: entropy decoding, dequantization and inverse transform, prediction, and smoothing filter. The encoding/decoding processes in HEVC are briefly introduced in this paragraph, but each technique used in the processes is more complicated because they are continuously sophisticated at every establishment of new standard. The basic approach of each video coding standard stays unchanged while components of coding techniques have been optimized with enormous time and labor. For more detailed explanation of HEVC, I refer the interested readers to an overview article [53].

In the field of video coding, coding efficiency is basically evaluated by using rate-distortion characteristics. The small bitrate with high-quality reconstructed video is desirable for video coding. The bitrate means the size of bitstream per second in a video. The rate-distortion characteristics denote how much video quality is maintained with a specific bitrate. This characteristic can be defined for every coding methods and be compared based on rate-distortion curve as shown in Figure 2.12, where horizontal and vertical axes are bitrate and video quality, respectively. As the criteria for quality of an encoded video, peak signal-to-noise ratio (PSNR) is typically used. PSNR indicates the extent of signal degradation based on ratio of noise caused by encoding to the maximum allowable pixel values in a signal. Under the assumption that a pixel value is represented with

Input:
Bitstream

Entropy
decoding

Conversion coefficient,
Quantization parameters

Dequantization,
Inverse transform

Prediction
image

Prediction
error

Prediction
process

Filter
information

Reference
image

Reference
image memory

Block-based
adaptive
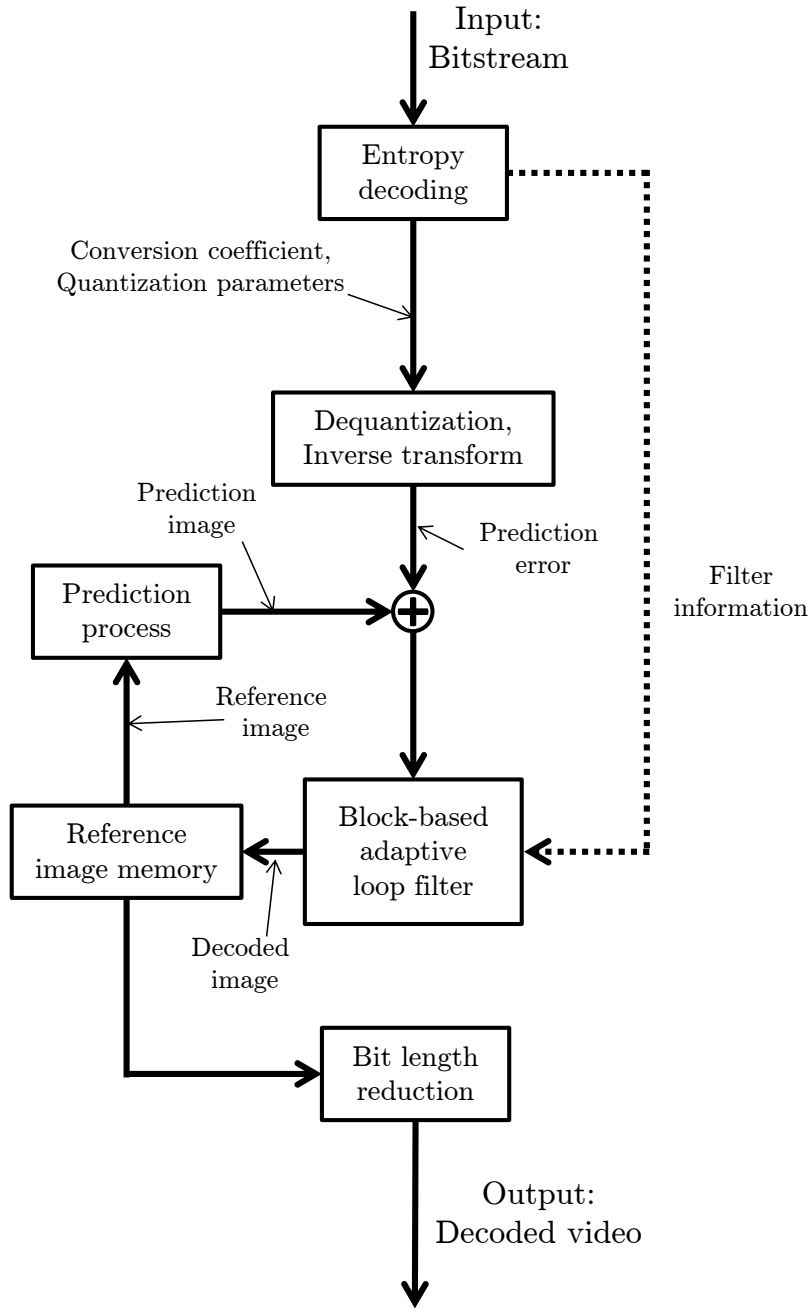loop filter

Decoded
image

Bit length
reduction

Output:
Decoded video

Figure 2.11: Decoding process of HEVC
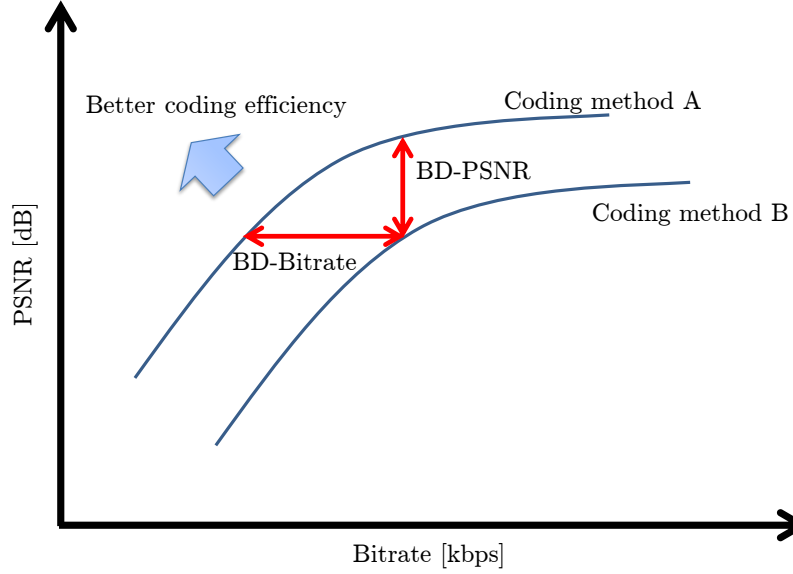
Figure 2.12: Example of rate-distortion curves

eight bits, PSNR is calculated via following equation:

$$\text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}}, \tag{2.8}$$

$$\text{MSE} = \frac{1}{XY} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \left(A(x,y) - B(x,y) \right)^2, \tag{2.9}$$

where X and Y mean the width and height of an image signal, $A(x,y)$ and $B(x,y)$ denote an original image signal and an image signal obtained by decoding the bitstream, respectively, and MSE indicates the mean square error. The decoded image signal $B(x,y)$ includes encoding noises occurred by the lossy coding. High PSNR indicates that less noise is included in the decoded signal; thus, higher PSNR value means higher quality of the decoded signal. In the case of video signal, PSNR is measured for each frame and the average over all frames is typically employed for a quality evaluation. A rate-distortion curve of a method achieving better coding efficiency appears on the top-left in a graph because top-left area means high quality is achieved with a small bitrate. Therefore, coding method A is superior to coding method B in the case of Figure 2.12. To quantitatively compare rate-distortion curves, Bjøntegaard delta rate

(BD-rate) and Bjøntegaard delta PSNR (BD-PSNR) is proposed [54]. BD-rate evaluates the difference of bitrate at the same PSNR, and BD-PSNR evaluates the difference of PSNR at the same bitrate. In the case of a light field, $A(x, y)$ and $B(x, y)$ become 4D signals as follows:

$$\text{MSE} = \frac{1}{STXY} \sum_{s=0}^{S-1} \sum_{t=0}^{T-1} \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} \left(A(s, t, x, y) - B(s, t, x, y) \right)^2, \quad (2.10)$$

where S and T denotes the number of horizontal and vertical viewpoints, respectively. To deal with color image signals in RGB color space, Equations (2.9) and (2.10) are rewritten as

$$\text{MSE} = \frac{1}{XY} \sum_{c=0}^{3} \sum_{x,y} \left(A_c(x, y) - B_c(x, y) \right)^2, \quad (2.11)$$

$$\text{MSE} = \frac{1}{STXY} \sum_{c=0}^{3} \sum_{s,t,x,y} \left(A_c(s, t, x, y) - B_c(s, t, x, y) \right)^2, \quad (2.12)$$

where the subscript c denotes a color component. For example, values of subscript $c = 0$, 1, and 2 indicate R, G, and B components, respectively.

Along with the development of 3D image processing technologies, demand for efficient light field compression has been growing. Video coding techniques in the video coding standards can be applied to light field signal for removing its redundancy; then, many studies on light field coding based on the video coding standards have been investigated [41–46, 55, 56]. Extensions of HEVC for multi-view images has been standardized as Multi-view HEVC (MV-HEVC) [41]. The extensions aim at encoding multi-view images with motion-compensation-based method. Y. Li et al. proposed an intra prediction scheme for light field images based on the inter-frame prediction of HEVC [42]. This method integrates the inter-prediction scheme into the intra prediction and applies it to light field data. Another approach for applying the video coding technologies is handling light field images as a single-viewpoint video by aligning viewpoint images in some order. When the single-viewpoint video from light field images is encoded by using HEVC, redundancy between viewpoint images are removed by motion compensation process. Influence of frame order of light field images is investigated in [43, 55–58]. The most straightforward order is raster

scan order, but zigzag order and circular order from the center of view-
point can be considered as shown in Figure 2.13, where the small squares
indicate viewpoint images and its number and color indicate the frame or-
der. It has been revealed that coding efficiency is actually different from
each frame order. In [57, 58], several approaches of finding an adaptive
frame order depending on contents of light fields have been investigated.
These studies try to encode light field images using HEVC with small
modifications.

(a) Raster

(b) Zigzag

(c) Circular

Figure 2.13: Example of frame orders for applying modern video codecs to light fields

Chapter 3

Light Field Coding with Weighted Binary Patterns

This chapter describes a novel light field coding scheme using weighted binary patterns. This chapter aims at addressing general light field coding problems. A general introduction of the proposed schemes is first explained. Then, details of "baseline" and "progressive" schemes are expounded, and their performance are evaluated. A method using a disparity compensation technique for further improving the proposed schemes is introduced, and its performance is validated by experiments. Finally, the whole of contents in this chapter is summarized. The contents of this chapter have first been published in journal papers [2] and [3] in the publication list enumerated in the end of this thesis.

3.1   Introduction

This chapter proposes a novel light field coding scheme using weighted binary patterns. Our laboratory has proposed the coding scheme with weighted binary patterns in [47]. In contrast to the coding methods using the modern video cording standards with small modifications, we believe that the standard video coding techniques are not necessarily the most suitable for a dense light field. This scheme has been developed based on this mind. The key idea of this scheme is that a light field is approximated by using only several weighted binary patterns. The binary

patterns and corresponding weight values are computationally obtained so as to optimally approximate the target light field.

This idea was originally proposed for generating temporal binary sequences used for multi-view displays with active shutter grasses developed by Koutaki [48]. He implemented a multi-view display system with a digital light processing (DLP) projector and active shutter glasses. The DLP can display binary patterns while rapidly switching them, the rapidly switched binary patterns can represent multi-values signals such as grayscale and RGB color signals because human eyes perceive the average of rapidly switched patterns. His display system utilized the active shutter glasses, which control the transmittance of light rays reaching human eyes, to display different images according to viewpoint positions. By varying the transmittance according to viewpoint positions, viewers at different positions can perceive different images, which are generated with the linear combination of sum of binary patterns using different weights according to the positions. His system demonstrated that weighted sum of small-number binary patterns can accurately represent multi-view images. We have believed that his idea can be utilized as a compressive representation of a light field and be applied to light field coding. To the best of our knowledge, there is no research for applying this idea to light field coding and validating the suitability.

The proposed coding scheme is completely different from these of modern video coding standards, and its decoding process is dramatically simpler than that of the standard codecs. The simplicity of decoding process allows us to implement faster and less power-hungry decoder than those of the standard codecs. However, the encoding process, i.e. solving the optimization problem, takes much longer than the video coding standards. As the number of binary patterns increases, the encoding time exponentially increases; then, the encoding time soon becomes infeasible. To reduce the computational complexity of the encoding process, a progressive coding scheme has been proposed. The progressive scheme progressively approximates a light field with a small number of weighted binary patterns at each group and can greatly reduce the computational complexity for obtaining optimal binary patterns. This approach is based on the

divide-and-conquer strategy, so that the progressive scheme is expected to achieve reasonable rate-distortion performance although the approximation accuracy is slightly degraded. Additionally, the progressive scheme gives scalability to encoded data. Using only a part of groups of weighted binary patterns, rough light field images can be obtained with lower computational complexity than using all weighted binary patterns. By selecting the number of groups used for decoding, the quality of decoded light field can be controlled. However, there still remains a problem that the approximation accuracy is slightly degraded in exchange for reduction of computational complexity. Therefore, the baseline method and its progressive extension have a problem of the trade-off between computational complexity and rate-distortion performance.

One of the methods for dealing with the problem is the use of disparity compensation, which is to shift the pixels in the images according to a specified disparity value and the viewpoint positions. In the coding schemes using weighted binary patterns, the binary patterns represent only the common components among the images at different viewpoints, while the weights represent viewpoint-dependent components of the light field. If binary patterns can also represent viewpoint-dependent components, the approximation accuracy of the coding scheme would be improved. The disparity compensation can provide the binary patterns with the capability to represent viewpoint-dependent components. The effectiveness of applying disparity compensation to the coding scheme has been preliminarily investigated in [49]. Experimental results show that the disparity compensation improves the approximation accuracy, but the disparity values were empirically determined beforehand. The appropriate disparity values depend on a captured scene, so a method of finding the appropriate disparity values should be provided in some way.

In this chapter, I propose a method of applying disparity compensation to the progressive light field coding with weighted binary patterns. I aim at improving rate-distortion performance of the progressive light field coding by adaptively applying disparity compensation while avoiding infeasible computational complexity. When disparity compensation is applied to each of the binary patterns in a straightforward manner in the

optimization problem, it is quite difficult to solve the problem because it includes three sets of unknown values: binary patterns, its corresponding weights, and disparity values applied to the binary patterns. By combining the progressive scheme with disparity compensation, the optimization problem at each group can be solved with brute-force search on a set of candidate disparity values because the number of binary patterns in each group is small. The proposed method finds the best disparity value at each group so that the encoding result adaptively uses good disparity values depending on target light fields. Furthermore, disparity compensation is not considered in the baseline method using weighted binary patterns; thus, the proposed method might outperform not only the progressive framework but also the baseline one.

3.2 Light Field Coding Using Binary Patterns and Weights

This section introduces the baseline and progressive coding schemes using weighted binary patterns and evaluates their performances. The contents of this section have been published in a journal paper [2] in the publication list.

3.2.1 Baseline Method

First, a framework of "baseline" light field coding is introduced. A light field, which is equivalent to multi-view images, to be compressed is given as $L(s, t, x, y)$. The set of symbols $(s, t), (s = 1, \cdots, S, t = 1, \cdots, T)$ and $(x, y), (x = 1, \cdots, X, y = 1, \cdots, Y)$ indicate viewpoint coordinates in a 2D grid and pixel coordinates in a viewpoint image, respectively. A light field $L(s, t, x, y)$ can be exactly represented using a linear combination of binary patterns as follows:

$$L(s, t, x, y) = \sum_{n=1}^{STb_L} B_n(x, y) r_n(s, t), \tag{3.1}$$

where $B_n(x, y) \in \{0, 1\}, (n = 1, 2, \cdots, N), r_n(s, t) \in \mathbb{R}$, and b_L indicate binary patterns, corresponding weights, and a bit depth used for light-

field pixels respectively. A pixel value represented with b_L bits is equal to a b_L-digits binary number. In the case of $S = T = 1$ (namely a conventional single-viewpoint image), each pixel value of the n-th binary pattern corresponds to n-th digit of b_L-digit binary number of each pixel at the same position in an image, so that a linear combination of b_L binary patterns can represent an image. Therefore, a $S \times T$ light field can be exactly represented by using a linear combination of $S \times T \times b_L$ binary patterns. This image-signal representation using weighted sum of binary patterns is utilized in the DLP projector [59, 60] which is widely used in projector systems that enlarge and display an image. In practical usage, a large number of viewpoints such as $S = T = 17$ is desirable; thus, a quite large number of binary patterns is required to represent a light field. However, from the viewpoint of compression, the number of binary patterns is desirable to be small. Koutaki demonstrated that weighted sum of small-number binary patterns can accurately represent multi-view images in his multi-view display system [48]. We employed his idea as compressive representation of a light field.

The baseline method approximates a light field $L(s, t, x, y)$ by using the sum of N weighted binary patterns as follows:

$$L(s, t, x, y) \simeq \sum_{n=1}^{N} B_n(x, y) r_n(s, t). \tag{3.2}$$

Figure 3.1 illustrates the framework of light field approximation using weighted binary patterns. All the images at different viewpoints are represented with the same binary patterns, but different weights are used depending on the viewpoints; therefore, the binary patterns and the weights have the common and viewpoint-dependent components of the multi-view images, respectively.

To obtain binary patterns and weights that can accurately approximate the given light field, we solve an optimization problem defined as:

$$\underset{\substack{B_n(x,y) \\ r_n(s,t)}}{\arg \min} \sum_{s,t,x,y} \left\| L(s, t, x, y) - \sum_{n=1}^{N} B_n(x, y) r_n(s, t) \right\|^2. \tag{3.3}$$

The optimal solution to Equation (3.3) cannot be easily obtained because Equation (3.3) includes two sets of unknowns, $B_n(x, y)$ and $r_n(s, t)$. We
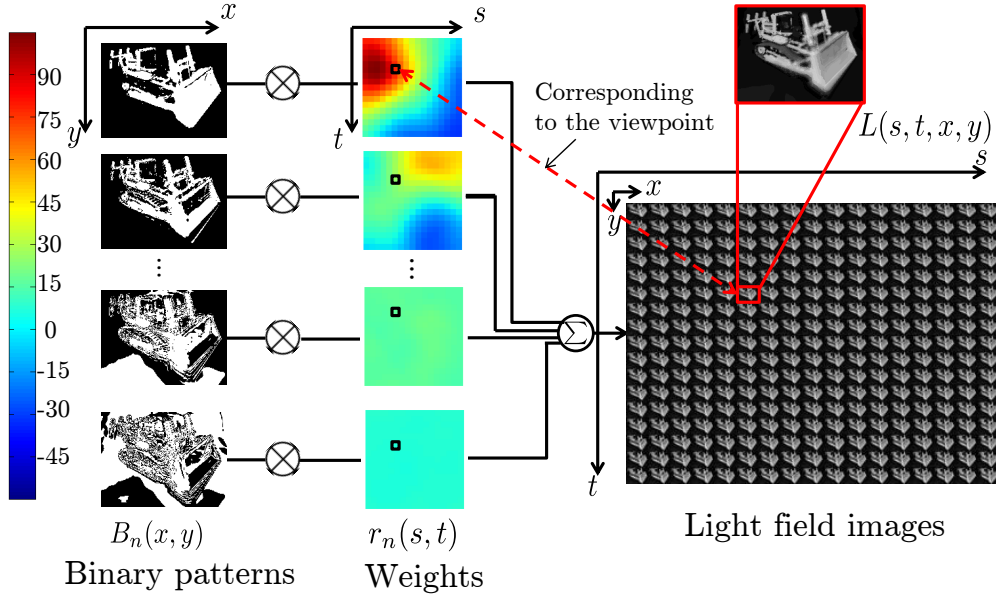
Figure 3.1: Light field approximation using binary patterns $B_n(x, y)$ and corresponding weights $r_n(s, t)$

use an alternating optimization to obtain the solution, where the binary patterns $B_n(x, y)$ are initialized at first, and we repeat the two following steps until convergence.

(i) The binary patterns are fixed, and the weights are optimized.

(ii) The weights are fixed, and the binary patterns are optimized.

The optimization of binary patterns (i) can be regarded as a standard least squares minimization problem that can be easily solved by well-known methods. For instance, "solve" function in OpenCV can be employed as a numerical solver for this least square minimization. The solution to the optimization of weights (ii) can be obtained individually for each pixel (x, y) because this problem is pixel-independent. The optimization (ii) is regarded as a binary combinational optimization known as a NP-hard problem. We obtain its solution by using simple brute-force search although the computational complexity of brute-force search is heavy and increases exponentially as the number of binary patterns is increase. After convergence, a light field $L(s, t, x, y)$ can be reconstructed using Equa-
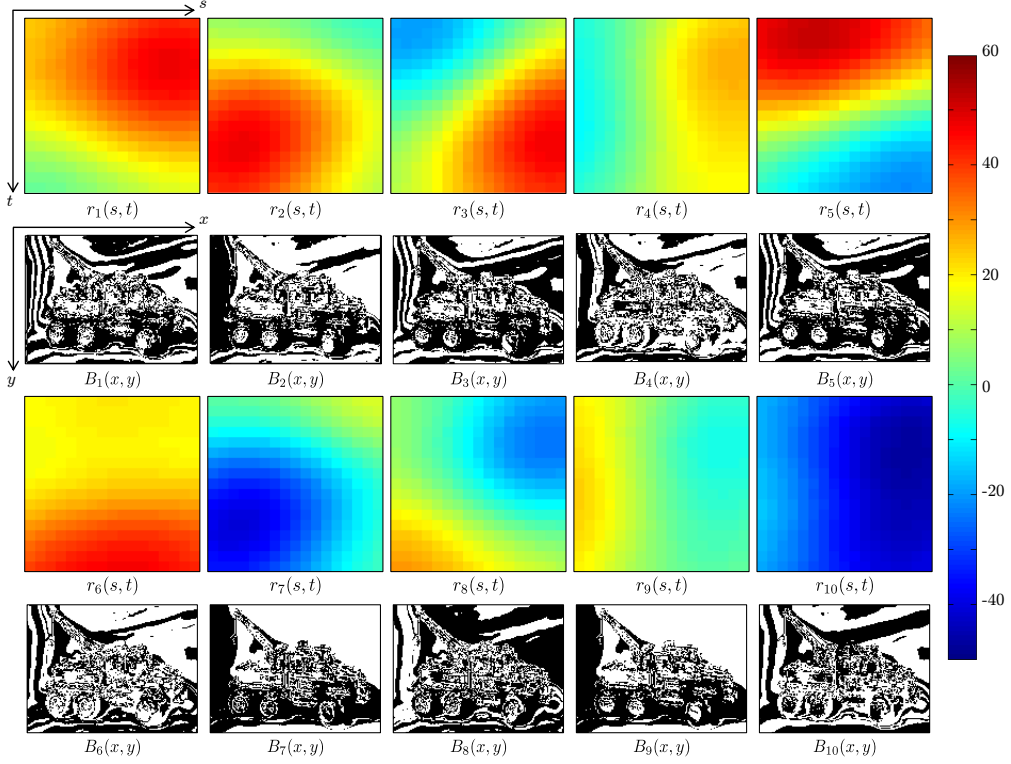
Figure 3.2: Binary patterns $B_n(x, y)$ and weights $r_n(s, t)$ with $N = 10$ for truck dataset

tion (3.2) with the obtained solution to Equation (3.3). In our current implementation, all $B_n(x, y)$ are initialized to the same image obtained by applying binary thresholding to the top-leftmost image of the target light field. Since all the binary patterns are identical at this point, the weights $r_n(s, t)$ cannot be determined uniquely in the optimization (i) of the first iteration. We empirically adapted solve function in OpenCV to obtain one of the possible solutions, namely a set of weights, and move on to the next step.

An example of binary patterns and weights for truck dataset [25] is shown in Figure 3.2 [47]. The dataset has 17×17 viewpoints and 160×120 pixels in grayscale. Binary patterns and weights in Figure 3.2 are obtained by solving Equation (3.3) with $N = 10$. Figure 3.3 [47] indicates the error maps with absolute difference between original and decoded light field

(a) Top left image

(b) Central image

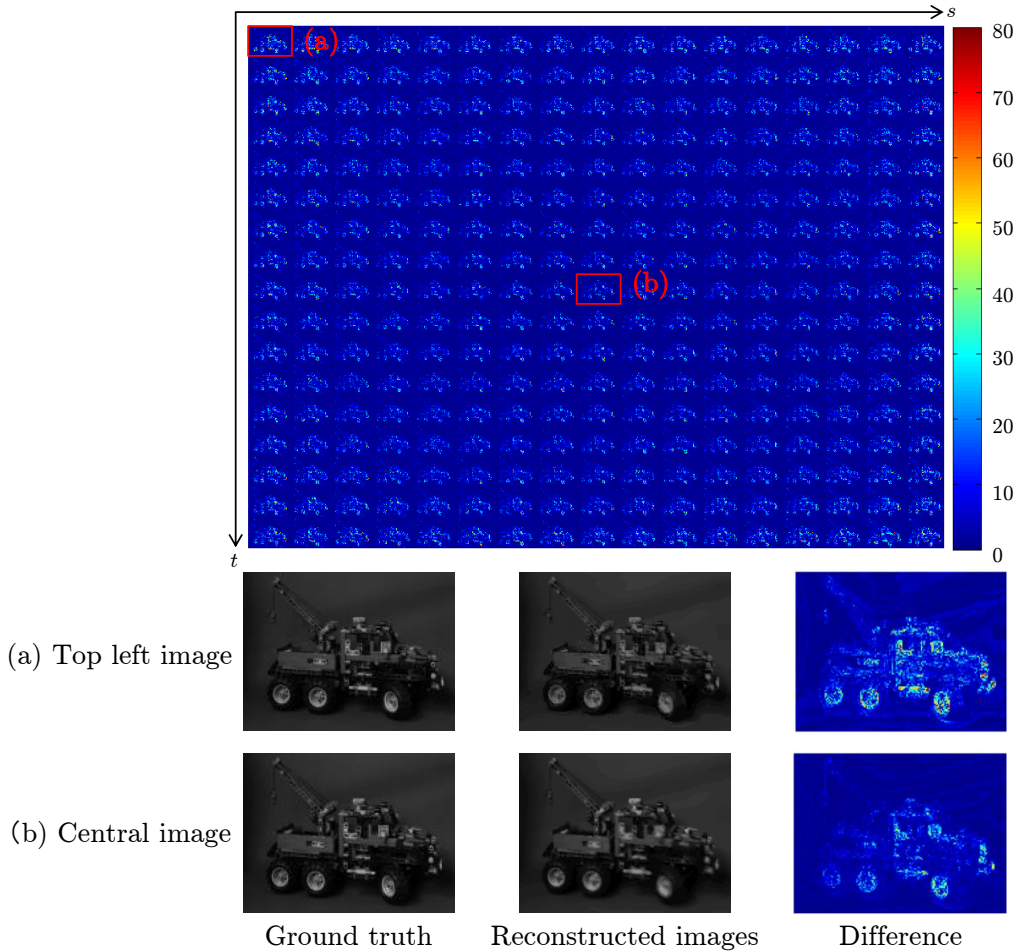Ground truth   Reconstructed images   Difference

Figure 3.3: Error map between original and decoded light field images with $N = 10$

images. The decoded images can be obtained using Equation (3.2) with the binary patterns and weights as shown in Figure 3.2. According to the Figure 3.3, the error tends to be larger for the viewpoints located farther from the center and for the regions having larger disparities. For instance, errors in the front wheel of the truck are larger than that in the other pixel. This tendency is caused by the fact that the common binary patterns are used for all viewpoint images, which imposes a limitation on the representation capability for the original 4D signal. The errors can be reduced by increasing the number of binary patterns N.

Similarly to other coding schemes, the proposed scheme has a trade-off between rate (number of bits) and distortion (accuracy of decoded data). This trade-off can be controlled by simply changing the number of binary patterns N. When a target light field is represented by using N binary patterns and weights, the compression ratio of the scheme is calculated as follows:

$$\text{compression ratio} = \frac{N(XY + STb_r)}{STXYb_L}, \tag{3.4}$$

where b_r indicate bit depth used for weights $r_n(s, t)$. For instance, a compression ratio for light field images with $X = 160$, $Y = 120$, $S = 17$, $T = 17$, and $b_L = 8$ equals 0.54% in the case of $N = 10$ and $b_r = 16$. This compression ratio is calculated based on the binary patterns and weights which are not compressed. To further reduce the data amount, a lossless compression algorithm, e.g., gzip, can be applied to the binary patterns and weights.

An important advantage of the proposed scheme is extreme simplicity of decoding process. As shown in Figure 3.1, the light field can be reconstructed by using only product-sum operations. On the other hand, the decoding process of modern video coding standards includes complicated operations such as intra/inter-frame prediction, inverse DCT/DST transformation, and dequantization as mentioned in Section 2.3. Thus, the decoding process of the proposed scheme is dramatically simpler than that of modern video coding standards. This simplicity allows us to implement a faster and less power-hungry decoder using field programmable gate array (FPGA) and an application specific integrated circuit (ASIC).

3.2.2 Progressive Method

Next, a framework of "progressive" light field coding is expounded, which accelerates the encoding process of baseline method. To accelerate the encoding process, the coding scheme is made progressive on the basis of the divide-and-conquer strategy as shown in Figure 3.4. Assuming that an original light field $L(s, t, x, y)$ is approximated by using totally N binary patterns and weights, the N binary patterns are divided into M groups
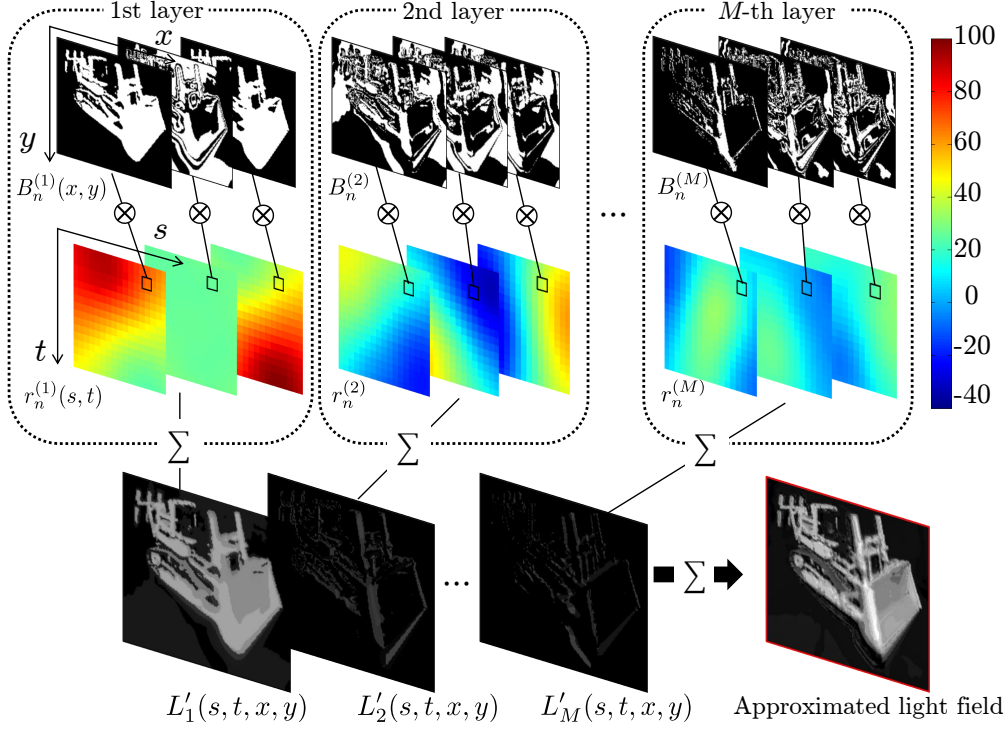
Figure 3.4: Progressive coding framework

(layers). Each layer has $\mathcal{N}$ binary patterns so that $\mathcal{N}M = N$ is satisfied. At the first layer, the target light field $L_1(s,t,x,y)$ is set as an original light field $L(s,t,x,y)$ and is approximated by using $\mathcal{N}$ binary patterns and weights based on Equation (3.3). The light field obtained by the approximation is denoted as $L'_1(s,t,x,y)$. At the next layer, the target light field $L_2(s,t,x,y)$ is defined as the difference between $L(s,t,x,y)$ and $L'_1(s,t,x,y)$. The target light field $L_2(s,t,x,y)$, i.e., the residual light field, is also approximated with $\mathcal{N}$ binary patterns and weights in the same manner as the first layer. At the following layers, we progressively repeat the process where $\mathcal{N}$ binary patterns and weights are calculated to

---

Algorithm 1 Progressive light field coding using weighted binary patterns

Input: $L(s, t, x, y)$
Output: $B_n^{(m)}(x, y), r_n^{(m)}(s, t)$ $(n = 1, 2, \cdots, \mathcal{N}, m = 1, 2, \cdots, M)$
Initialize $L_1(s, t, x, y) \Leftarrow L(s, t, x, y)$
for $m = 1$ to M **do**
    Obtain $B_n^{(m)}(x, y), r_n^{(m)}(s, t)$ $(n = 1, 2, \cdots, \mathcal{N})$ using Eq. (3.5)
    Carry over the residual using Eq. (3.6)
    $m \Leftarrow m + 1$
end for

---

approximate the target light field $L_m(s, t, x, y)$ as follows:

$$\underset{\substack{B_n^{(m)}(x,y) \\ r_n^{(m)}(s,t)}}{\arg \min} \sum_{s,t,x,y} \|L_m(s, t, x, y) - L_m'(s, t, x, y)\|^2, \tag{3.5}$$

$$L_m(s, t, x, y) = L(s, t, x, y) - \sum_{i=1}^{m-1} L_i'(s, t, x, y), \tag{3.6}$$

$$L_i'(s, t, x, y) = \sum_{n=1}^{\mathcal{N}} B_n^{(i)}(x, y) r_n^{(i)}(s, t), \tag{3.7}$$

where $B_n^{(i)}(x, y)$, $r_n^{(i)}(s, t)$, and $L_i'(s, t, x, y)$ indicate the n-th binary pattern, the corresponding weight, and the approximated light field at the i-th layer, respectively. Finally, an original light field $L(s, t, x, y)$ is approximated as follows:

$$L(s, t, x, y) \simeq \sum_{m=1}^{M} L_m'(s, t, x, y). \tag{3.8}$$

Specific algorithm for obtaining a solution of the progressive scheme is shown in Algorithm 1.
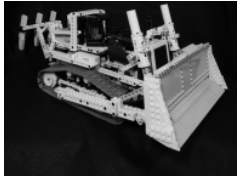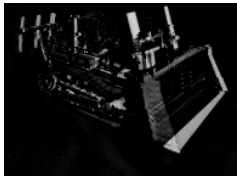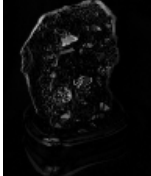
This progressive extension achieves remarkable reduction of the computational complexity by optimizing a small number of binary patterns and weights per layer on the basis of the divide-and-conquer strategy. Although the strategy generally cannot bring about the global-optimum solution, a feasible solution can be obtained with less computational complexity since the search space of the problem is reduced. In the progressive coding, dividing the binary patterns and weights into M groups is equal to

the reduction of search space. The main bottleneck of the baseline scheme is optimizing binary patterns; thus, reducing the number of binary patterns in the optimization can significantly accelerate the encoding process. The computational complexity for binary combinational optimization in the baseline scheme with N binary patterns is $\mathcal{O}(2^N)$, but the computational complexity for that in the progressive scheme is reduced to $\mathcal{O}(2^{\mathcal{N}})$ by dividing N binary patterns into M layers. Total computational complexity for binary combinational optimization in the progressive scheme equals $\mathcal{O}(M2^{\mathcal{N}})$ since the progressive scheme repeats the optimization for each layer. For instance, the computational cost of the progressive coding is reduced to $1/128$ compared with that of the baseline one in the case with $N = 12$, $M = 4$, and $\mathcal{N} = 3$. As described above, the progressive extension finds a feasible solution with less computational complexity, but the approximation accuracy cannot help being degraded since the divide-and-conquer strategy cannot find the global-optimum solution. Consequently, there is a trade-off between computational complexity and rate-distortion performance.

3.2.3 Performance Evaluation

To evaluate the performance of the coding schemes using weighted binary patterns, we implemented the baseline and progressive methods using the software made available from our website [61]. For optimizing Equations (3.3) and (3.5), the number of iterations was set to 20. To compress the binary patterns and weight values, gzip ver.1.6 was adopted as an optional post-processing. We adopted two implementation for H.265/HEVC: FFmpeg ver. 4.1 with default parameters and the HEVC Test Model [62] with a random access configuration [63]. As the difference between FFmpeg and HEVC Test Model, FFmpeg makes use of the x265 library for HEVC encoding and its default option is chosen to provide moderate coding efficiency with a reasonable complexity, while the configuration of the HEVC Test Model is basically determined to bring out the potential performance in the standardization process. To ascertain the effect of inter-frame prediction, we also tested all intra mode with the HEVC Test Model. When

Datasets	Truck	Bulldozer	Amethyst
# of views	17×17		
size	160×120	192×144	96×128
Top-left image			
Difference between top-left and top-right images			

Datasets	Bunny	Crystal	Knight
# of views	17×17		
size	128×128	128×128	128×128
Top-left image			
Difference between top-left and top-right images			

Figure 3.5: Datasets

we applied these video codecs to a light field dataset, we aligned images in the dataset in the row-major order and were regarded them as a single-viewpoint video sequence.

We first demonstrate performance of the baseline method in different configurations with truck dataset [25]. Figure 3.6 [47] shows the PSNR of reconstructed images using Equation (3.3) with different numbers of binary images ($N = 8$, 10, and 12). Each square corresponds to each viewpoint. As N increases, the entire reconstruction accuracy increases accordingly. The PSNR values are lower for the viewpoints located at
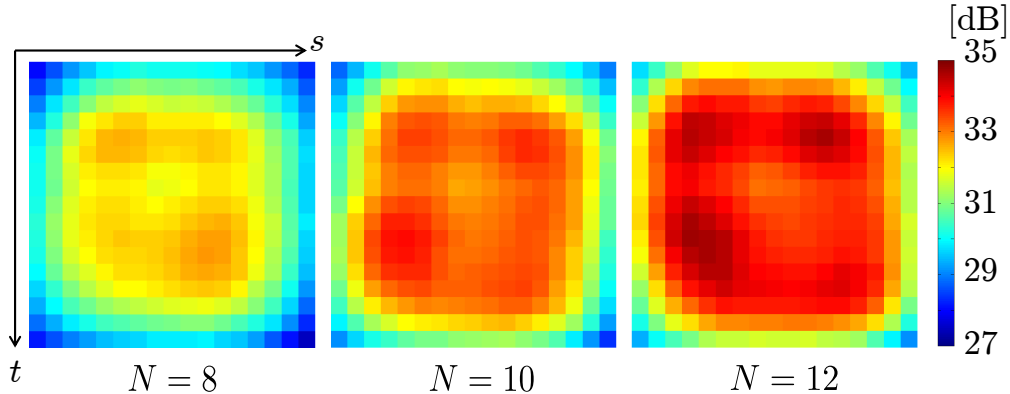
Figure 3.6: PSNR of reconstructed images using Eq. (3.3) with different number of binary patterns

peripheral positions than those located near to the center. This tendency is consistent with the error maps shown in Figure 3.3.

Next, we evaluated the rate-distortion performance and encoding time with and without the progressive framework. The PSNR values were calculated from the mean squared errors over all the viewpoints and pixels. The bitrate values were calculated from the sizes of files that were compressed with gzip. Computational times were measured on a Desktop PC running Windows 10 pro equipped with Intel (R) Core (TM) i5-4590 3.3-GHz CPU and 8.0 GB main memory. Figure 3.7 [47] shows that rate-distortion curves of the baseline and the progressive method. As shown in Figure 3.7 (a), using the progressive framework slightly decreases the coding efficiency compared to the baseline one. Figure 3.7 (b) shows the encoding time with and without the progressive framework, where the total number of binary images N was fixed to 12. We can see that an approximately 100-fold increase in speed is achieved with the progressive framework. In terms of the balance between the rate-distortion performance and the encoding time, $\mathcal{N} = 3$ is likely to be the best parameter in the progressive framework. Therefore, we employ this configuration from the following evaluation.

As shown in Figures 3.9–3.14 [47], we compared the baseline and progressive methods with H.265/HEVC in regard to the rate-distortion per-
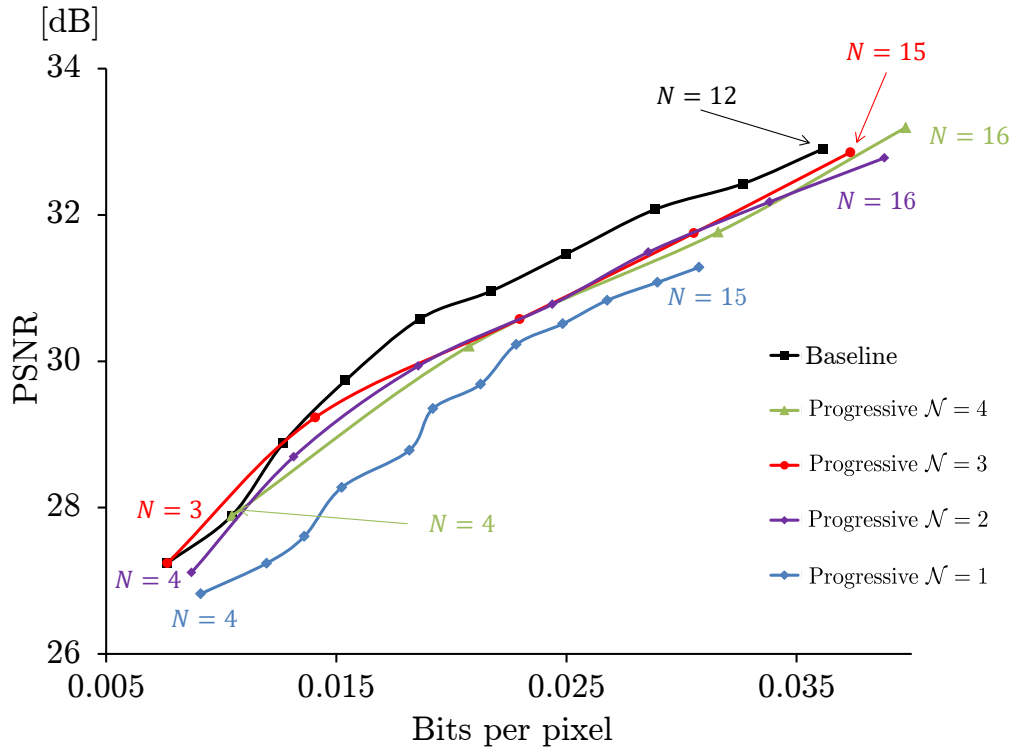
(a) PSNR
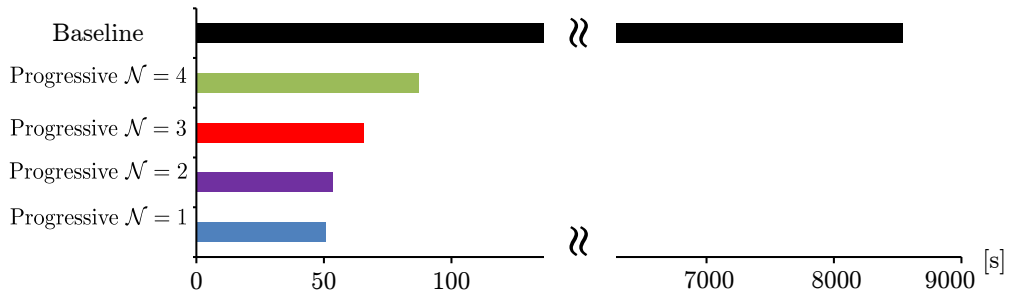
(b) Encoding time with $N = 12$

Figure 3.7: Rate-distortion performance and encoding time with and without progressive framework
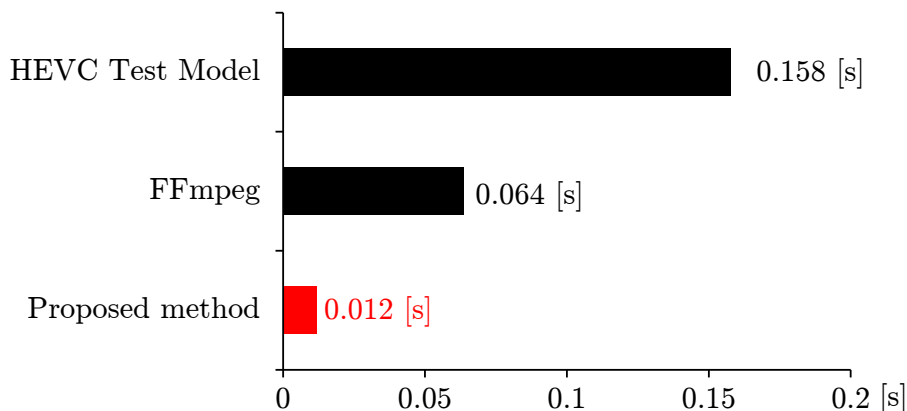
Figure 3.8: Comparison of decoding time

formances over several grayscale datasets taken from [25]. Performances of the baseline and progressive method appear to depend on the image differences between the viewpoints. They achieved a good rate-distortion performance for a dataset with small differences, such as Amethyst, but could not achieve high PSNR values for a dataset including large differences, such as Bulldozer. They achieved a reasonable performance overall, comparable to that of FFmpeg but moderately inferior to that of the HEVC Test Model. We believe these results are promising because HEVC is the state-of-the-art video coding standard that has been optimized with a significant amount of labor and time.

Finally, we compared the decoding times between our method and HEVC codecs using the truck dataset in grayscale. We used the Desktop PC running Windows 10 pro equipped with Intel (R) Core (TM) i5-4590 3.3-GHz CPU and 8.0 GB main memory, but we measured the decoding time on Ubuntu 16.04 installed on Virtual Box ver 5.1.16 due to the availability of software. We measured the user time for executing decoding processes with "time" command. We repeated the measurement 100 times and obtained the average for each method, which is plotted in Figure 3.8 [47]. For our method, N was set to 10 and the outputs were written as pgm files. The time for unzip process was negligible. For FFmpeg and the HEVC Test Model, the outputs were respectively written in pgm files and YUV files. We set the parameters for these codecs so
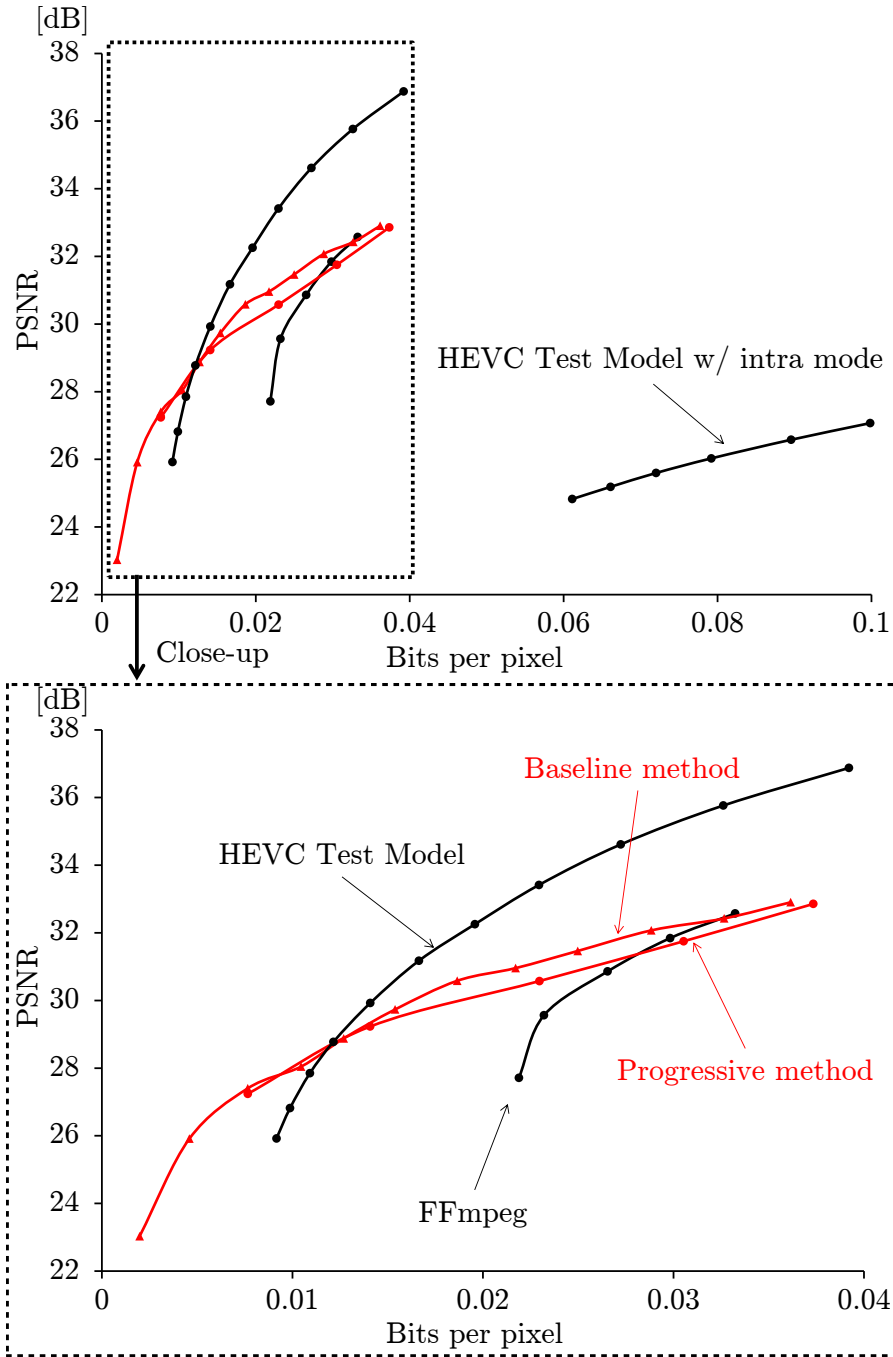
Figure 3.9: Comparison of R-D curves between proposed methods and HEVC for truck

Figure 3.10: Comparison of R-D curves between proposed methods and HEVC for amethyst
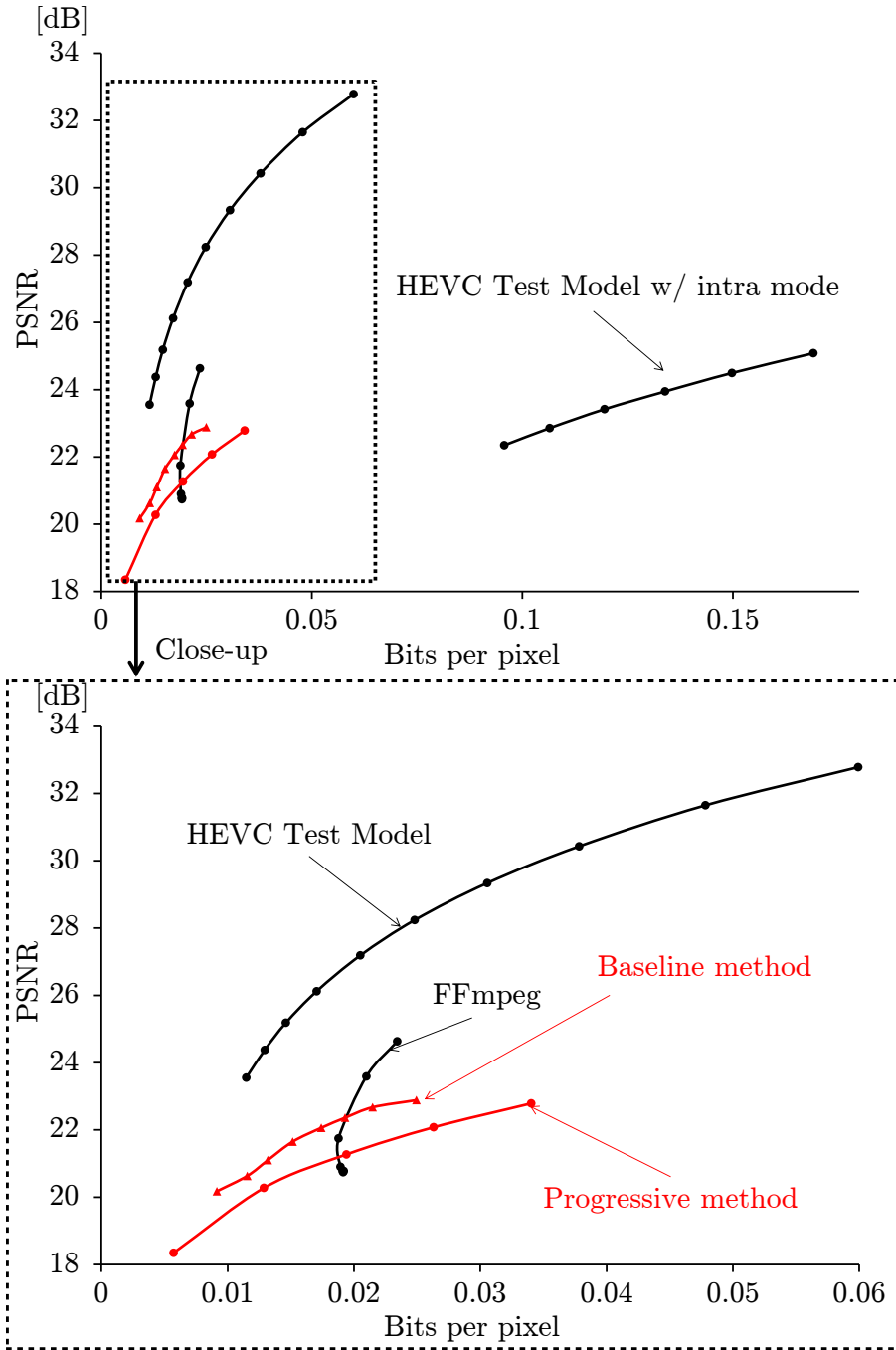
Figure 3.11: Comparison of R-D curves between proposed methods and HEVC for bulldozer

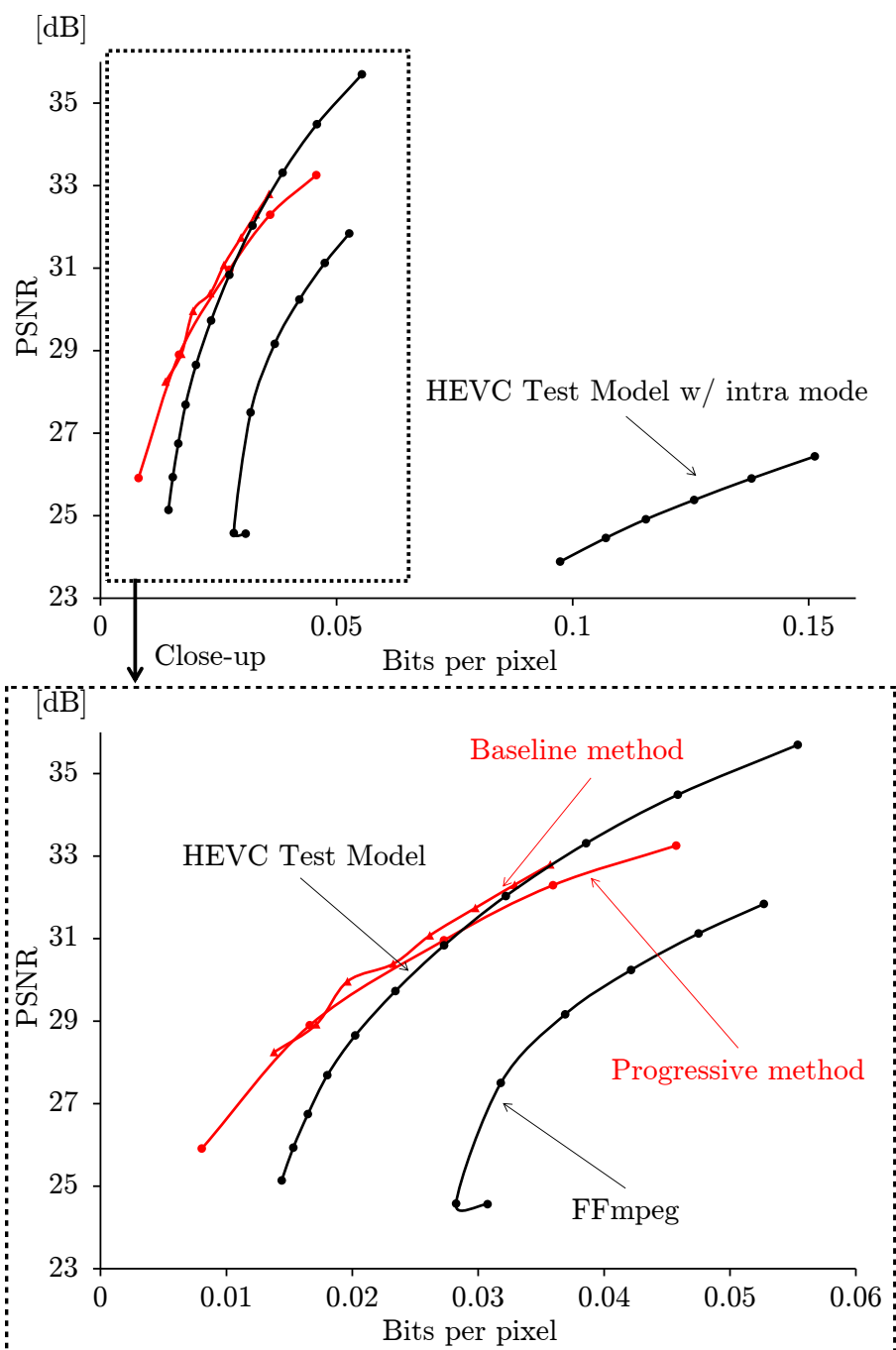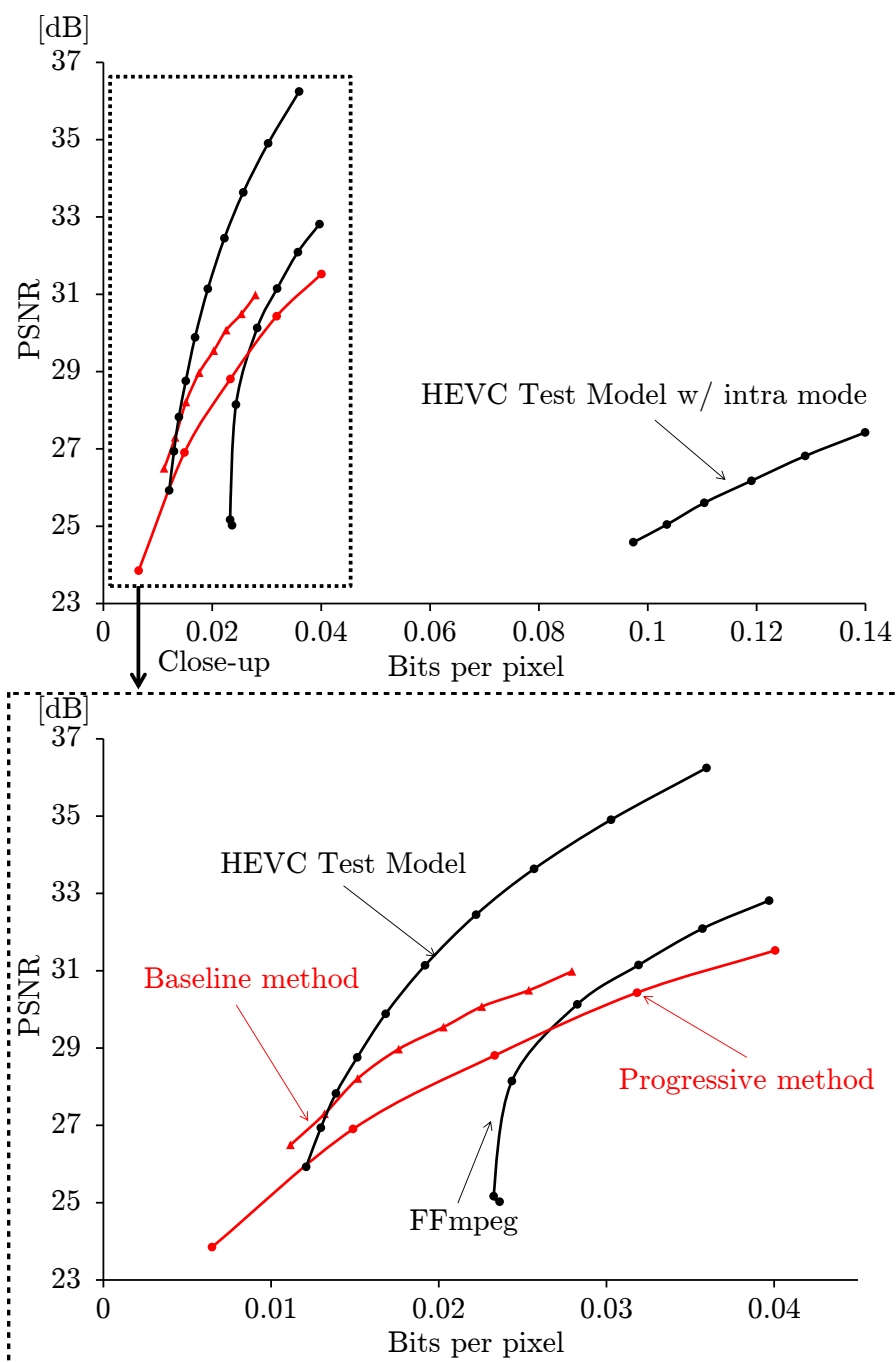Figure 3.12: Comparison of R-D curves between proposed methods and HEVC for bunny
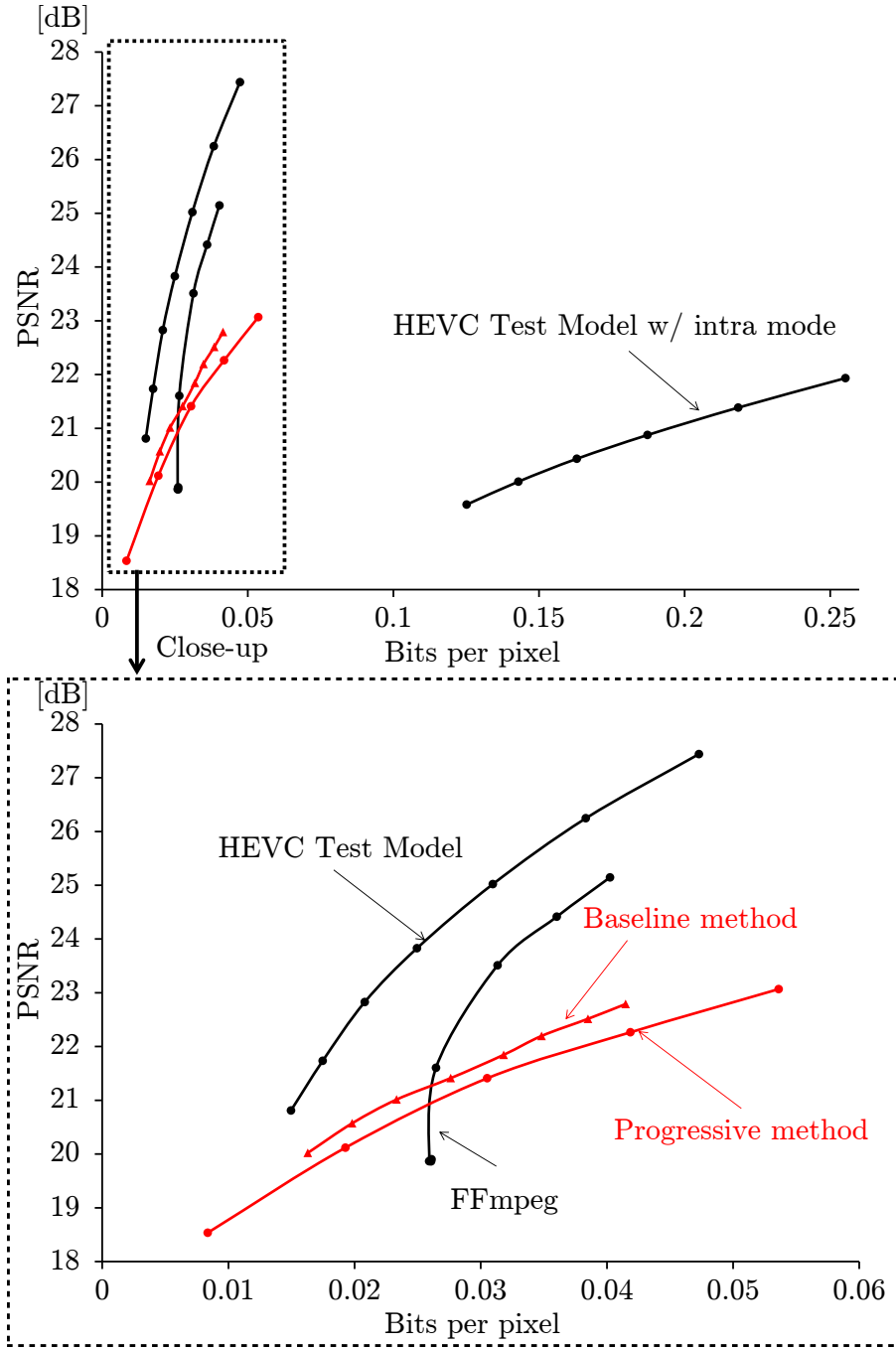
Figure 3.13: Comparison of R-D curves between proposed methods and HEVC for crystal
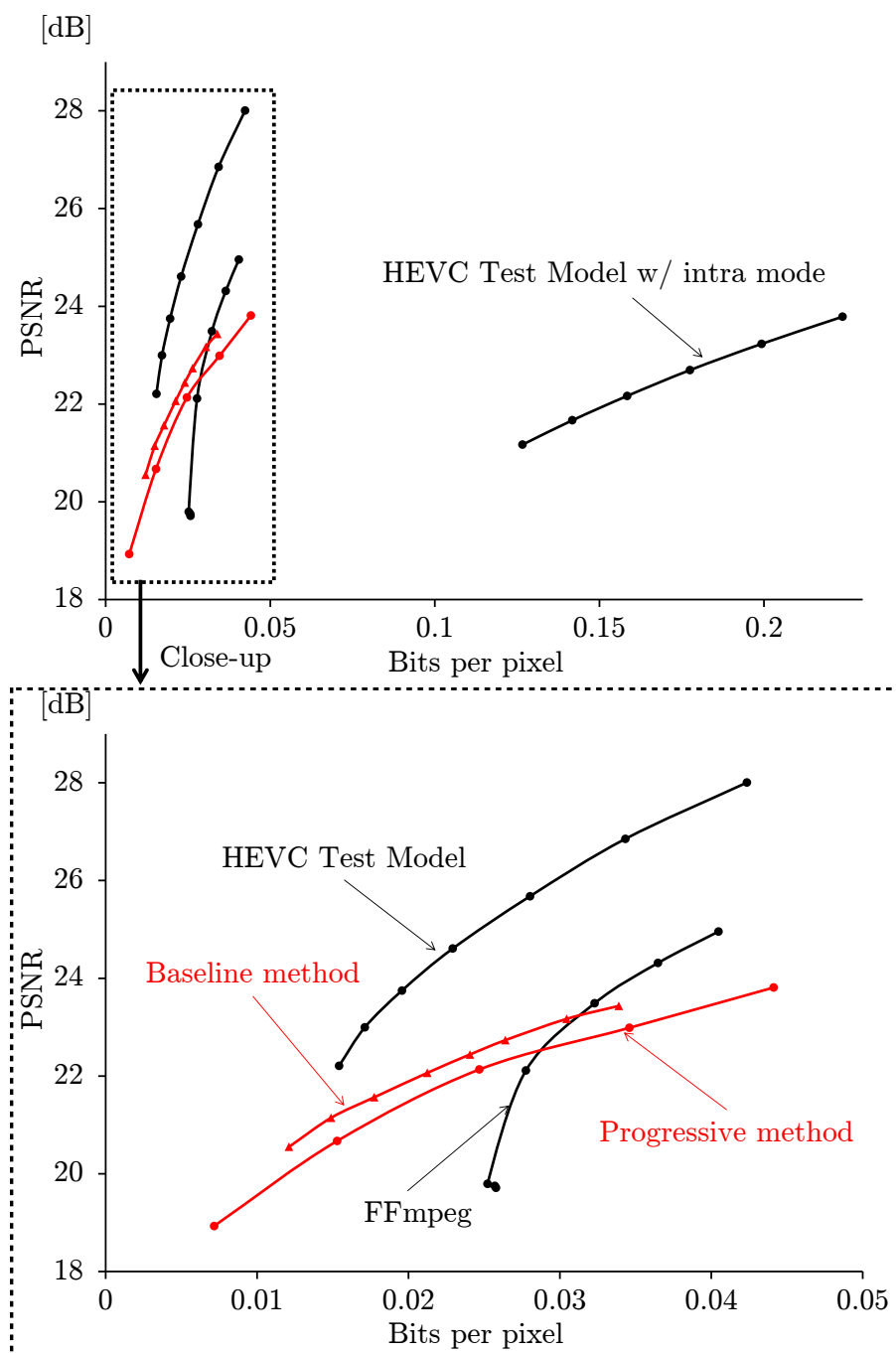
Figure 3.14: Comparison of R-D curves between proposed methods and HEVC for knight

that they resulted in almost the same PSNR as that of the scheme using weighted binary patterns. It can be seen from the graph that the method with weighted binary patterns runs much faster than HEVC codecs. This can be attributed to the simplicity of the scheme; as mentioned earlier, its decoding process is carried out with simple sum-of-product operations, while HEVC requires complex inter/intra-frame prediction and transforms as described in Section 2.3.

3.3   Disparity Compensation Framework for Binary Patterns

In this section, I propose the method of introducing a disparity compensation framework to the progressive light field coding scheme described in subsection 3.2.2 to improve the rate-distortion performance while avoiding infeasible computational complexity. In the conventional scheme, all the images at different viewpoints are approximated using the same binary patterns with different weights. This means that the binary patterns represent the common components among multi-view images and that the weights represent viewpoint-dependent components. Meanwhile, if disparity compensation is applied to the binary patterns, they can represent not only the common components but also the viewpoint-dependent components. Thus, applying appropriate disparity compensation might improve the representation capability of the binary patterns. The amount of disparities included in a light field depend on a captured scene; therefore, the appropriate disparity values should be adaptively searched for according to the scene. The contents of this section have been published in a journal paper [3] in the publication list.

3.3.1 Optimization for Disparity-compensated Binary Patterns and Weights

With disparity compensation applied, the approximation of a light field can be formulated as follows:

$$L(s,t,x,y) \simeq \sum_{n=1}^{N} B_n(x - sd_n, y - td_n)r_n(s,t), \qquad (3.9)$$

where d_n is a disparity value with which the n-th binary pattern is compensated. In the case where $x - sd_n$ and $y - td_n$ become non-integer values, pixel values at integer position are interpolated by using the neighbouring pixel values existing at non-integer position. If $x - sd_n$ and $y - td_n$ indicate the out of an image, in this thesis, I assume that the pixel value at the corner position continuously exists at the out of an image in this thesis although there are several approaches for dealing with this case. All pixels of $B_n(x,y)$ are shifted according to the viewpoint position (s,t) so that each viewpoint image is approximated by using slightly shifted binary patterns depending on the viewpoint position. According to Equation (3.9), the optimization problem to find the binary patterns, weights, and disparity value is defined as follows:

$$\underset{\substack{B_n(x,y) \\ r_n(s,t) \\ d_n}}{\arg\min} \sum_{s,t,x,y} \|L(s,t,x,y) - L'(s,t,x,y)\|^2, \qquad (3.10)$$

$$L'(s,t,x,y) = \sum_{n=1}^{N} B_n(x - sd_n, y - td_n)r_n(s,t). \qquad (3.11)$$

Although it would be preferable if the global-optimum solution for Equation (3.10) can be obtained, solving this optimization is quite difficult because it includes three sets of unknowns. Equation (3.9) is reformulated with a restriction on disparity values d_n and then introduce a disparity compensation framework into the progressive light field coding scheme presented in subsection 3.2.2.

I propose a progressive light field coding method that is formulated as

follows:

$$L(s,t,x,y) \simeq \sum_{m=1}^{M} L'_m(s,t,x,y) \tag{3.12}$$

$$L'_m(s,t,x,y) = \sum_{n=1}^{\mathcal{N}} B_n^{(m)}(x - sd_m, y - td_m) r_n^{(m)}(s,t), \tag{3.13}$$

where d_m is a disparity value used for the disparity compensation at the m-th layer. As a restriction, I use only one disparity value for each group of binary patterns. At each layer, binary patterns, weights, and a disparity value are obtained by solving the following optimization:

$$\underset{\substack{B_n^{(m)}(x,y) \\ r_n^{(m)}(s,t) \\ d_m}}{\arg\min} \sum_{s,t,x,y} \|L_m(s,t,x,y) - L'_m(s,t,x,y)\|^2, \tag{3.14}$$

$$L_m(s,t,x,y) = L(s,t,x,y) - \sum_{i=1}^{m-1} L'_i(s,t,x,y). \tag{3.15}$$

Equation (3.14) still includes three sets of unknowns like Equation (3.10). The range of disparities included in a dense light field is basically narrow because of its very small viewpoint interval; thus, I simplify the problem by manually defining a set of candidate disparities $\mathcal{D}$. The specific algorithm is shown in Algorithm 2, where $B_n^{(m)*}(x,y)$, $r_n^{(m)*}(s,t)$, and d_m^* denote a solution for Equation (3.14) at the m-th layer. The binary patterns and weights are calculated for each candidate disparity value $d \in \mathcal{D}$. From the set of the obtained binary patterns and weights, I use the one that achieves the best approximation accuracy. Consequently, the proposed method adaptively searches for the appropriate disparity value at each layer depending on a captured scene by simple brute-force search.

To practically solve Equation (3.14), it is reformulated as follows:

$$\underset{\substack{B_n^{(m)}(x,y) \\ r_n^{(m)}(s,t) \\ d_m}}{\arg\min} \sum_{s,t,x,y} \|L_m(s,t,x' + sd_m, y' + td_m)$$

$$- L'_m(s,t,x' + sd_m, y' + td_m)\|^2, \tag{3.16}$$

$$L'_m(s,t,x' + sd_m, y' + td_m) = \sum_{n=1}^{\mathcal{N}} B_n^{(m)}(x',y') r_n^{(m)}(s,t), \tag{3.17}$$

Algorithm 2 Progressive light field coding using disparity-compensated and weighted binary patterns

---

Input: $L(s, t, x, y), \mathcal{D}$

Output: $B_n^{(m)*}(x, y), r_n^{(m)*}(s, t), d_m^* \ (n = 1, \cdots, \mathcal{N}, m = 1, \cdots, M)$

Initialize $L_1(s, t, x, y) \Leftarrow L(s, t, x, y)$

for $i = 1$ to M **do**

    BEST_PSNR $\Leftarrow 0.0$

    for each disparity $d \in \mathcal{D}$ **do**

        Obtain $B_n^{(i)}(x, y), r_n^{(i)}(s, t)$ using Eq. (3.14) with fixed $d_m = d$

        $p \Leftarrow$ PSNR of $L(s, t, x, y)$ from Eq. (3.12) with $M = i$

        if BEST_PSNR $< p$ **then**

            $B_n^{(i)*}(x, y) \Leftarrow B_n^{(i)}(x, y)$

            $r_n^{(i)*}(s, t) \Leftarrow r_n^{(i)}(s, t)$

            $d_i^* \Leftarrow d$

            BEST_PSNR $\Leftarrow p$

        end if

    end for

    Carry over the residual using Eq. (3.15) with $m = i + 1$

    $i \Leftarrow i + 1$

end for

---

where $x' = x - sd_m$, and $y' = y - td_m$. The right side of Equation (3.17) takes the same form as the second term in Equation (3.3). I first apply disparity compensation to the target light field $L_m(s, t, x, y)$ to obtain $L_m(s, t, x' + sd, y' + td)$; after that, the solution for (3.16) is obtained in the same manner as solving Equation (3.3). The desired light field $L'(s, t, x, y)$ can be obtained by applying inverse disparity compensation to the light field $L'(s, t, x' + sd_m, y' + td_m)$.

By manually defining a set of candidate disparity values, the proposed method finds the best disparity values while keeping feasible computational complexity. The difference between the proposed scheme and the conventional progressive method with respect to computational complexity is the brute-force search for disparity values. Assuming that the number of elements of $\mathcal{D}$, i.e., the number of candidate disparity values, is denoted as D, the computational complexity for optimizing binary patterns in the proposed scheme is $\mathcal{O}(DM \cdot 2^{\mathcal{N}})$ because the proposed method

obtains the binary patterns and weights with all candidate disparity values at each layer. Although the proposed method takes much more time for encoding than the conventional progressive coding, it still can find the solution with feasible computational complexity. As I mentioned in subsection 3.2.2, the main bottleneck is optimizing binary patterns; thus, if $\mathcal{N}$ is kept small, the computational complexity of the proposed method never increases explosively like that of the conventional baseline scheme as the total number of binary patterns N increases. In the case with $N = 24$, $M = 8$, $\mathcal{N} = 3$, and $D = 10$, the proposed scheme takes 10 times longer than the conventional progressive scheme, but the computational cost of the proposed scheme $10 \times 8 \times 2^3 = 640$ is still feasible compared with that of the baseline scheme 2^{24}.

When a target light field is represented by using N binary patterns and the corresponding weights, which are divided in M groups, the compression ratio of the proposed scheme is calculated as follows:

$$\text{compression ratio} = \frac{N(XY + STb_r) + Mb_d}{STXYb_L},\tag{3.18}$$

where b_d denotes a bit depth used to describe the used disparity values. The increase of total bits compared with Equation (3.4) is less of an issue because one byte ($b_d = 8$) is enough to describe a disparity value when D is set to 10–20.

3.3.2 Experimental Results

The proposed method was implemented using the software made available from our website [61] in order to evaluate its performance. Six light field datasets [25] shown in Figure 3.2.3, each of which consists of 17×17 gray-scale multi-view images, were used in the experiments. The number of iterations for alternating optimization to calculate binary patterns and weights was fixed at 20. The set of candidate disparity values $\mathcal{D}$ in the proposed method was given as $\mathcal{D} = \{0.0, \pm0.2, \pm, 0.5, \pm0.8, \pm1.0, \pm1.5, \pm2.0\}$; namely, the number of candidate disparity values is $D = 13$. All binary patterns $B_n(x, y)$ were initialized by the result of binary thresholding to the most top-left image of the input light field. The number of binary

Table 3.1: Selected disparity values in each layer

layer number	1	2	3	4	5	6	7	8
Truck	0.2	0.0	0.2	0.0	0.2	0.0	0.0	0.2
Bulldozer	0.2	0.2	0.8	0.2	0.2	0.2	0.5	-0.2
Amethyst	0.0	0.0	0.0	0.0	0.0	-0.2	0.0	0.0
Bunny	0.0	0.0	0.0	0.0	0.0	-0.2	0.0	0.0
Crystal	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
Knight	0.0	0.0	0.0	0.0	0.0	0.0	-0.5	0.0

patterns in each layer was set as $\mathcal{N} = 3$. The approximation accuracy was evaluated using PSNR, which is calculated from the mean square errors over all the viewpoints and pixels such like Equation (2.10) in Section 2.3.

I first investigated the effectiveness of the disparity compensation framework in the progressive light field coding scheme on the basis of rate-distortion performance. The proposed method, the conventional progressive coding, and the baseline coding were compared. The number of binary patterns N for the proposed method and the conventional progressive coding was varied from 3 to 24, but N for the baseline coding was limited from 3 to 12 because of the high computational complexity. The bitrate was calculated from raw binary patterns, weights, and used disparity values without gzip compression.

Figures 3.15–3.20 show rate-distortion curves for six datasets. The conventional baseline and progressive methods are called "Baseline" and "Progressive", respectively, and the proposed method is called "Progressive + disp. comp." in the results. Compared to the conventional progressive coding, the proposed method remarkably improves rate-distortion performances for truck and bulldozer while showing almost the same performances for the other datasets. The rate-distortion performances of the proposed method for truck and bulldozer even outperform those of the baseline method. Table 3.1 indicates selected disparity values in the proposed method with $N = 24$ for each layer of six datasets. The proposed method finds and uses non-zero disparity values in most of the layers for truck and bulldozer; thus, the disparity compensation outstandingly makes a difference for the two datasets. Meanwhile, for the other datasets,
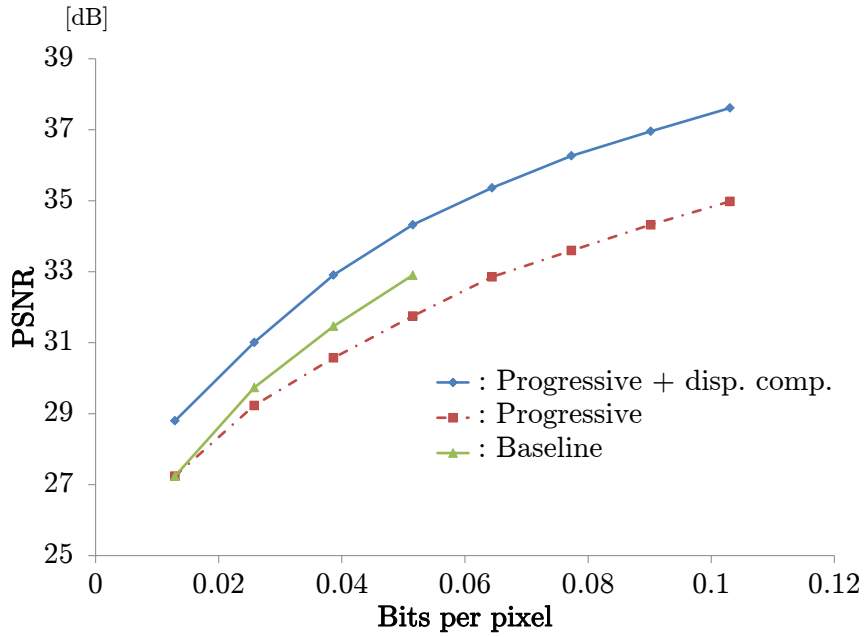
Figure 3.15: R-D curve comparison among proposed schemes for truck
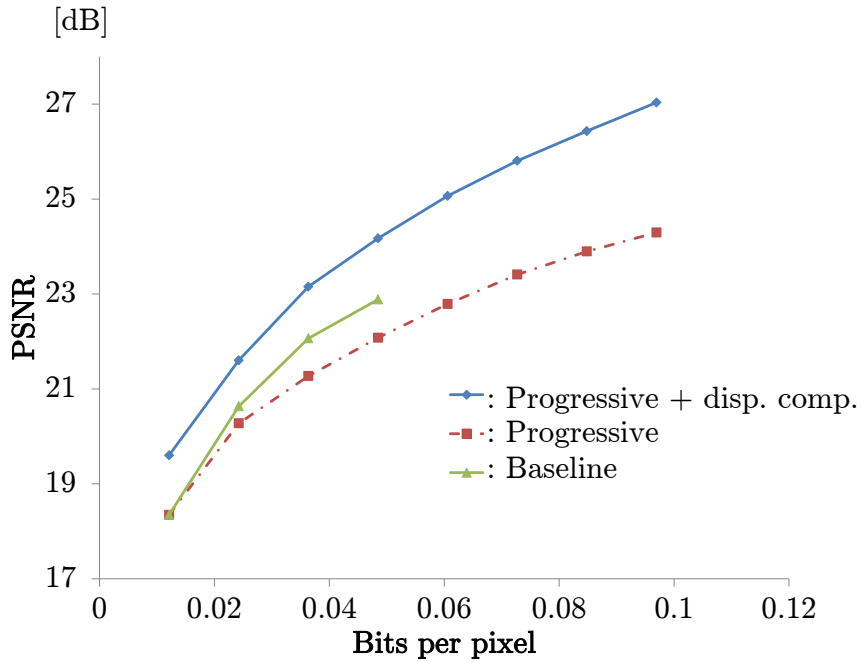

Figure 3.16: R-D curve comparison among proposed schemes for bulldozer

Figure 3.17: R-D curve comparison among proposed schemes for amethyst

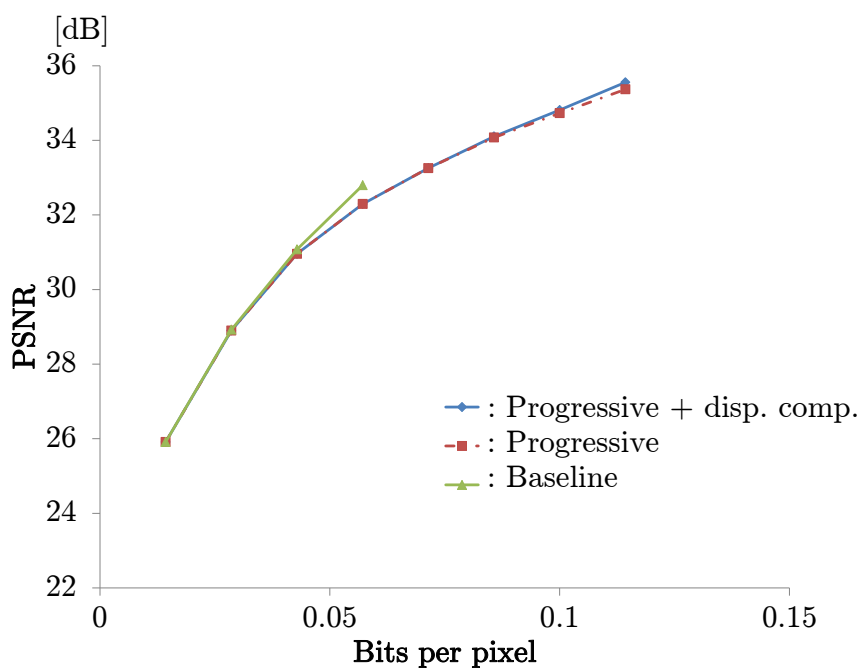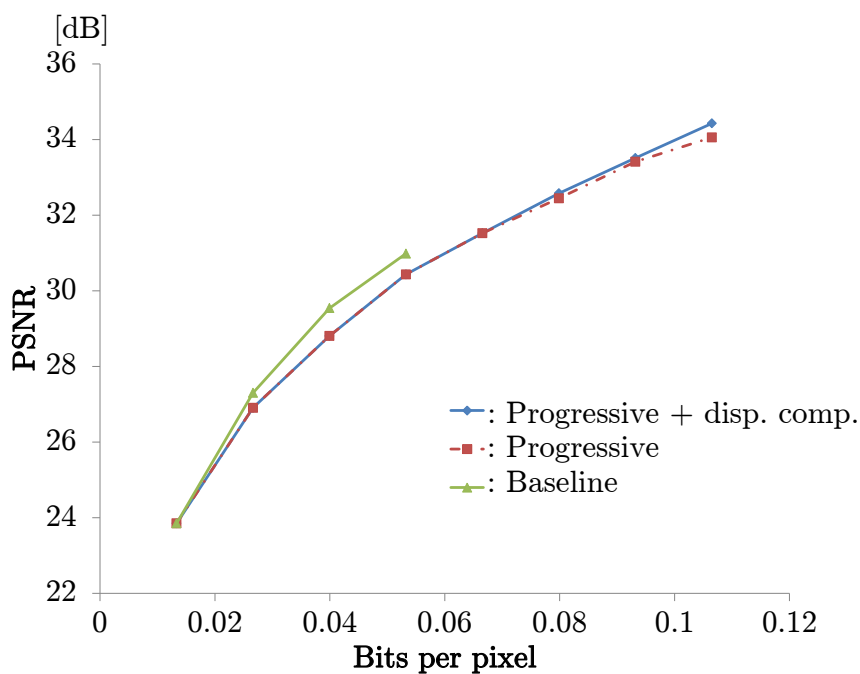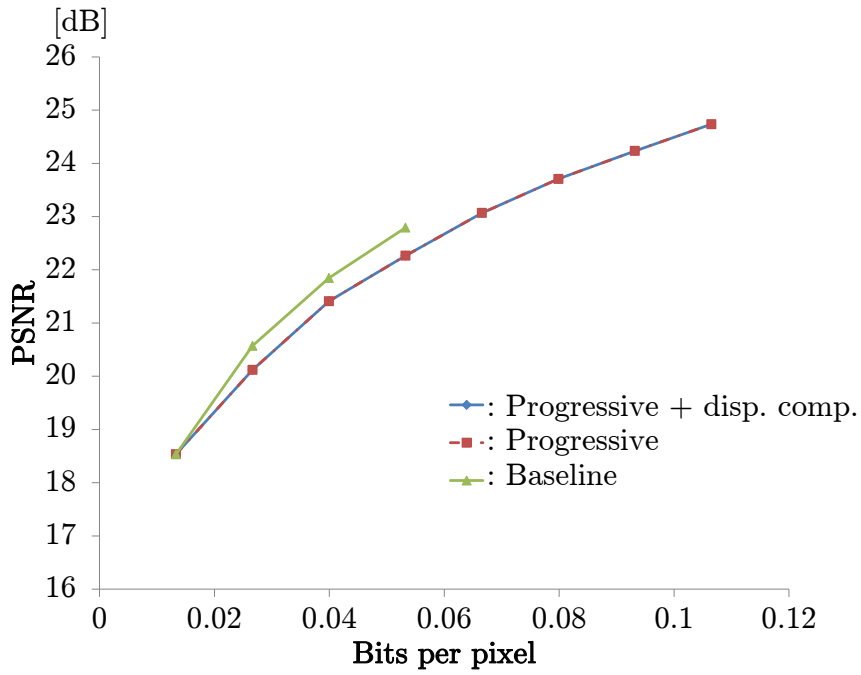Figure 3.18: R-D curve comparison among proposed schemes for bunny

Figure 3.19: R-D curve comparison among proposed schemes for crystal

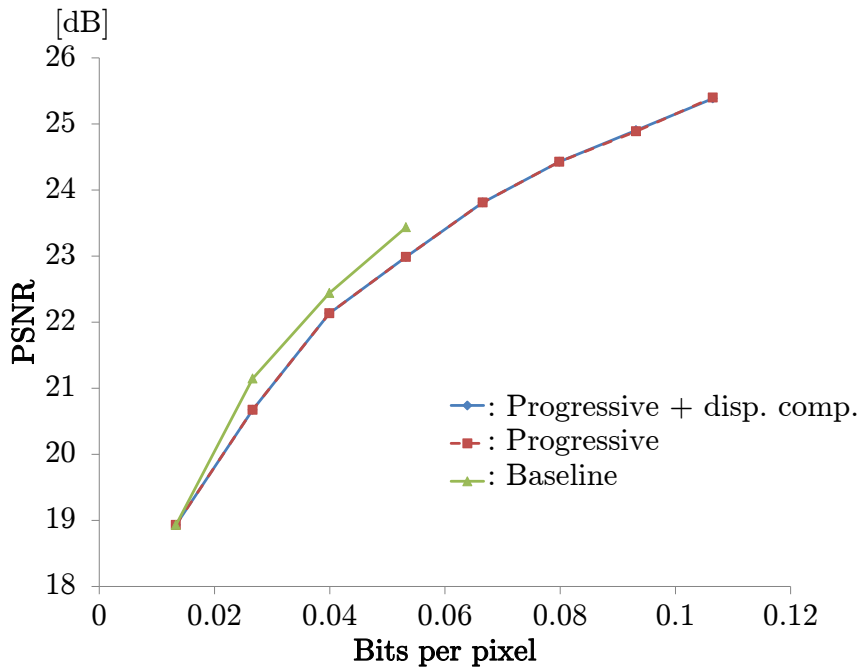Figure 3.20: R-D curve comparison among proposed schemes for knight

(i) Progressive (ii) Progressive + disp. comp.

(a) Truck

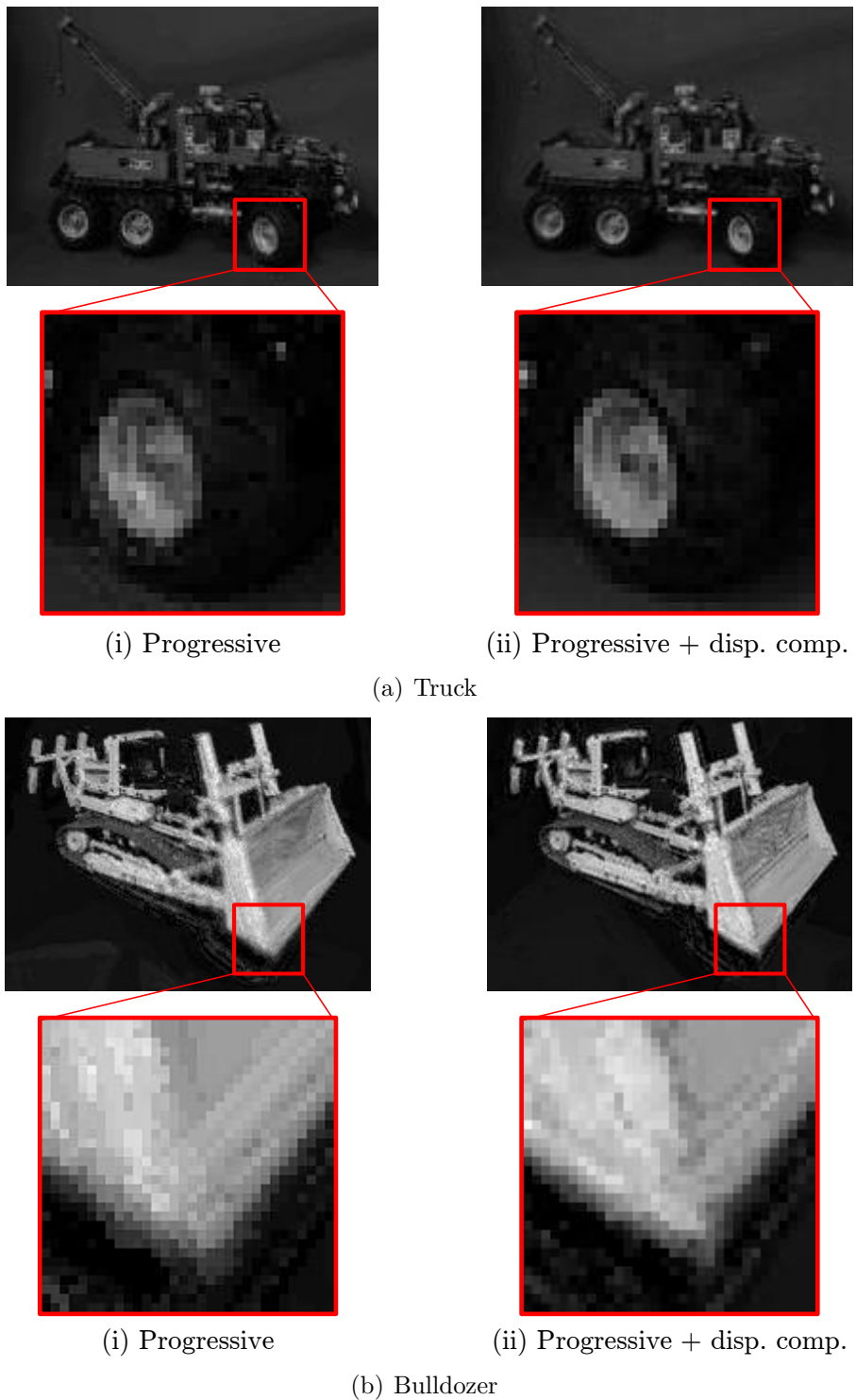
(i) Progressive (ii) Progressive + disp. comp.

(b) Bulldozer

Figure 3.21: Visual comparison of most top-left image ($N = 24$)

Table 3.2: Encoding time for Truck [s]

# of binary patterns	3	6	9	12
Baseline	16.9	106	876	7521
Progressive	16.8	32.0	49.0	64.1
Progressive + disp. comp.	263	514	774	1026

Table 3.3: Encoding time for Knight [s]

# of binary patterns	3	6	9	12
Baseline	13.2	94.6	782	6768
Progressive	13.5	26.0	39.4	52.0
Progressive + Disp. comp.	212	422	643	852

the selected disparities were zero for almost all the layers, which explains the reason why our method and the progressive method performed similarly in Figures 3.17–3.20; these datasets have only small disparities, and thus, the progressive method without disparity compensation was sufficiently effective for them. Figure 3.21 presents visual comparisons between the proposed method and the conventional progressive coding. It seems that the proposed method achieves better approximation accuracy by alleviating blurs on the object's parts having large disparities.

Tables 3.2 and 3.3 show the comparison of encoding time of three methods for truck and knight datasets, respectively. The encoding time was measured on a desktop PC running Windows 10 Pro equipped with an Intel Core (TM) i7-6700 3.4-GHz CPU and 16.0-GB main memory. The proposed method takes much more time for encoding than the conventional progressive method as described in Section 3.2.2. However, the encoding time of the proposed method linearly increases as the number of binary patterns increases, while the encoding time of the baseline method explosively increases. The experimental results prove that the proposed method improves rate-distortion performance while avoiding a computational complexity explosion.

Next, the proposed method was compared with the modern video coding standard H.265/HEVC. As general implementations of H.265/HEVC, the HEVC Test Model [62] was used with random access configuration and FFmpeg ver. 4.1. To apply these video codecs to light-field datasets, im-

ages in the dataset in the row-major order were aligned and were regarded as a video sequence. The bitrate of the proposed method was calculated from the binary patterns, weights, and used disparity values, which are compressed by gzip ver. 1.9. As a reference, the baseline method was also compared with the above methods.

Figures 3.22–3.27 show rate-distortion performances of three methods for six datasets. The proposed method shows better performance than that of the baseline method in truck dataset, and the performance can be more comparable to that of the HEVC Test Model. However, the proposed method still shows slightly inferior performance to the HEVC Test Model in the other datasets. As I mentioned in previous section, we still believe that the performance of the proposed method is promising because it achieves superior or comparable performance to that of FFmpeg and the HEVC Test Model, which has been optimized using enormous labor and time. The excellent performance of HEVC comes from the combination of many sophisticated coding techniques such as intra/inter prediction, transform coding, and arithmetic coding, where the optimal coding modes (e.g. prediction mode and block partition) are selected in accordance with the image content. Meanwhile, our method is merely constructed on a very simple framework using weighted binary patterns with disparity compensation.

3.4 Summary

In this chapter, we proposed a novel light field coding scheme using weighted binary patterns, its progressive extension for accelerating encoding process, and a method of introducing a disparity compensation framework into the progressive framework. Our idea is to encode a light field with several binary images and the corresponding weight values. The binary images and weight values were optimized iteratively so as to best approximate the original light field. I also mentioned how this coding scheme can be made progressive.

Experimental results demonstrated that our method has several promising properties. First, it can achieve reasonable coding efficiency (rate-

distortion performance) compared with those of highly-sophisticated standard video codecs. Moreover, its decoding process is extremely simple and can be executed much faster than the standard video codecs. Furthermore, the computational complexity of the encoding process can be substantially reduced by the progressive coding scheme.

Additionally, I reformulated progressive coding framework to integrate disparity compensation into binary patterns and proposed an optimization approach to obtain binary patterns, weights, and disparity values which approximate light field as accurate as possible. I simplified the problem by searching disparity values from a manually-defined set of candidate disparities $\mathcal{D}$. The proposed method adaptively finds the best disparity value at each layer.

Experimental results showed that, compared with the conventional progressive light-field coding, the proposed method improves the approximation accuracy in several datasets thanks to the disparity compensation. The proposed method outperforms not only the progressive method but also the original baseline method for some datasets. Although the proposed method takes much more time for encoding than the conventional progressive coding, the computational complexity never increases explosively like that of the conventional baseline coding.
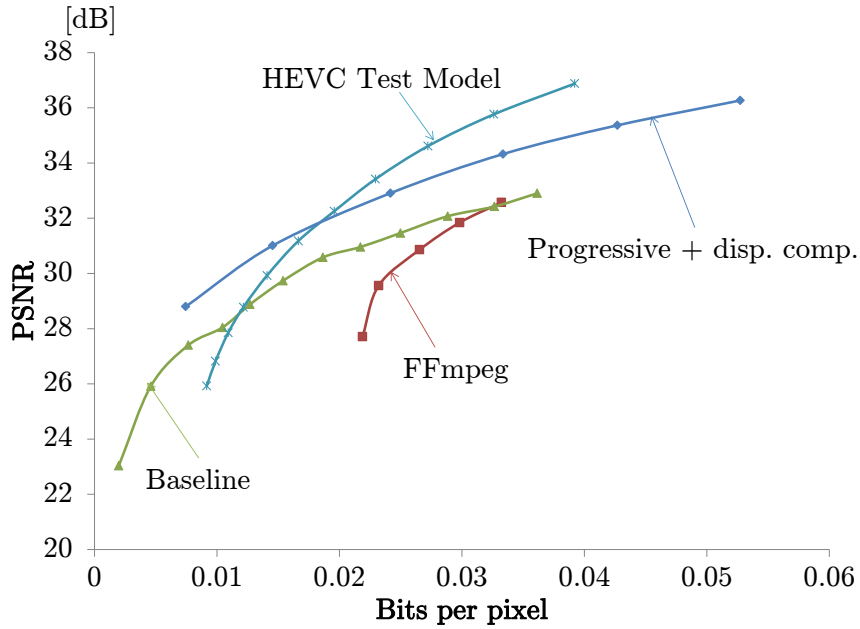
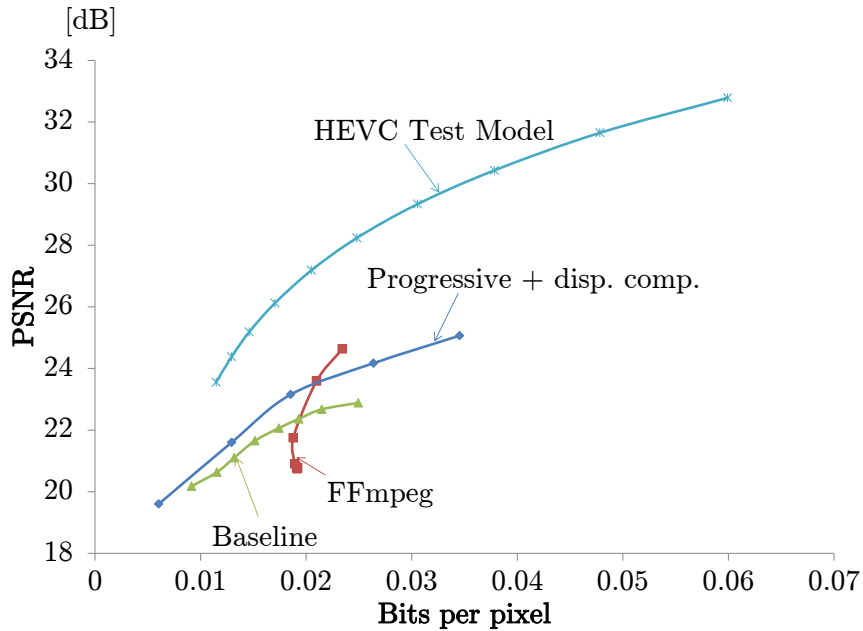Figure 3.22: R-D curve comparison with H.265/HEVC for truck

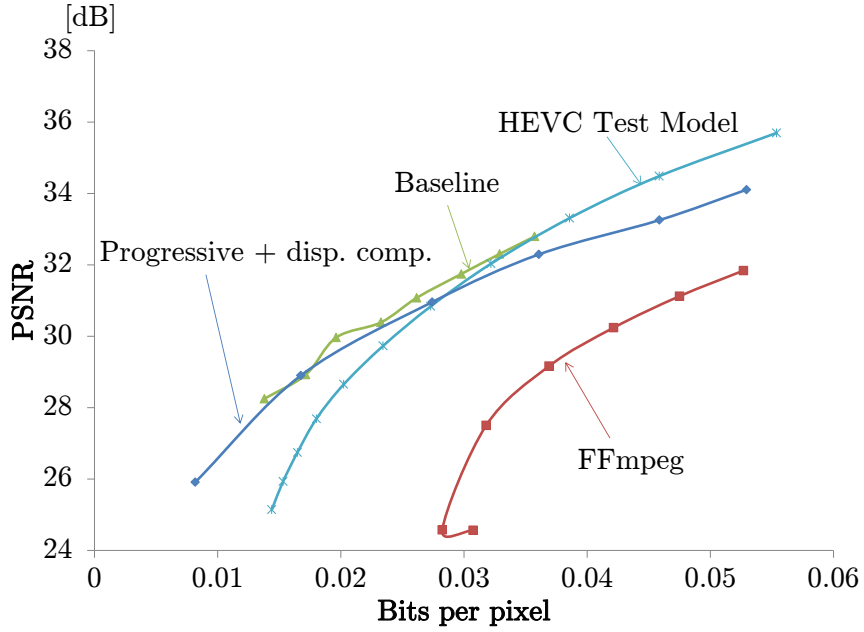Figure 3.23: R-D curve comparison with H.265/HEVC for bulldozer

Figure 3.24: R-D curve comparison with H.265/HEVC for amethyst
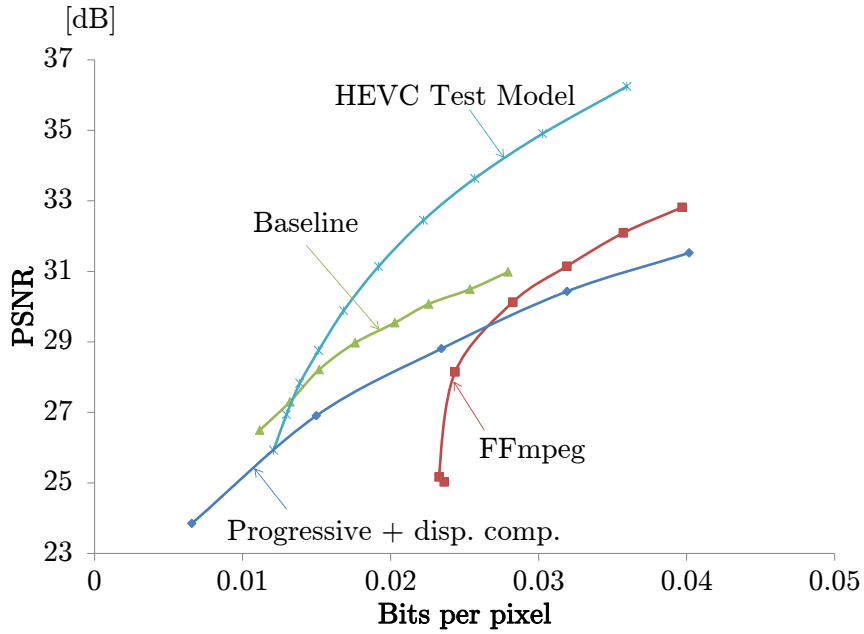

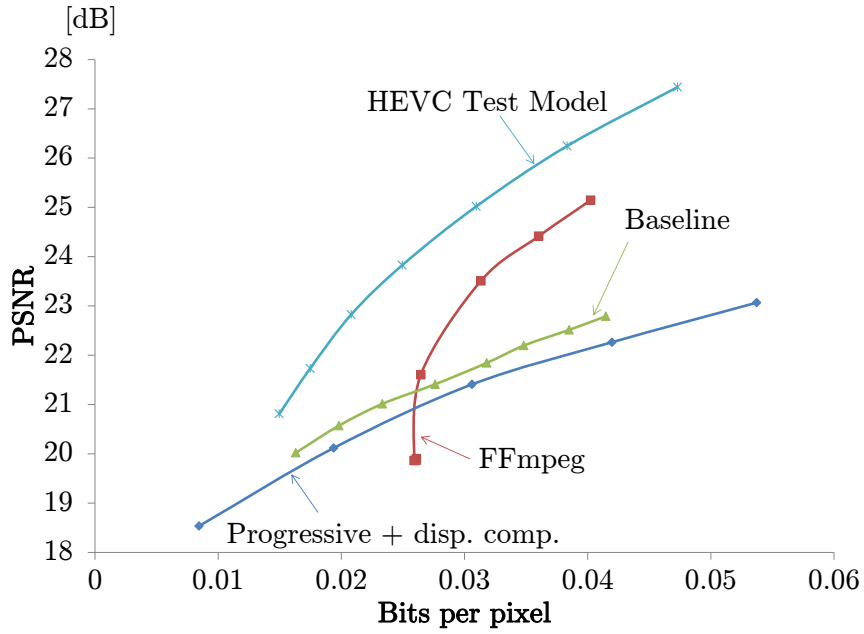Figure 3.25: R-D curve comparison with H.265/HEVC for bunny

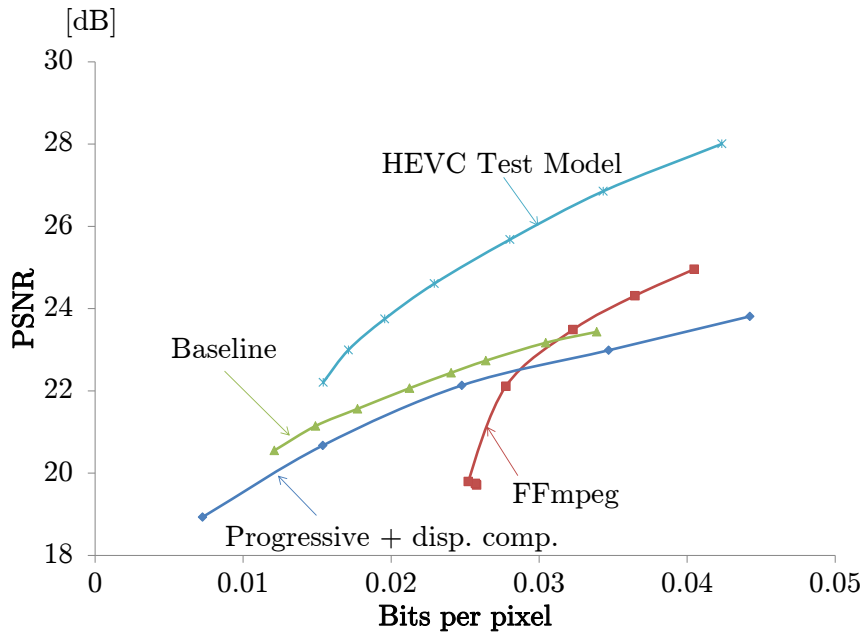Figure 3.26: R-D curve comparison with H.265/HEVC for crystal

Figure 3.27: R-D curve comparison with H.265/HEVC for knight

Chapter 4

Light Field Coding for Compressive 3D Display

This chapter considers a light field coding problem under a practical scenario using a compressive 3D display. In contrast to the previous sections, this chapter tackles a light field coding problem in a more practical scene. First, the practical scenario is defined in detail, and two possible light field transmission frameworks in the scenario are supposed. To evaluate their rate-distortion performances, simulative experiments are conducted. At the end of this chapter, the whole of contents is summarized. The contents of this chapter have been published in a journal paper [1] in the publication list.

4.1   Introduction

This chapter investigates coding efficiency of a light field under a scenario where the light field is reproduced by a compressive 3D display. For this scenario, a 3D video transmission system is supposed. In the system, the compressive 3D display is adopted as the receiving terminal. As described in the Section 2.3, many researchers have studied on light field image/video coding using video coding standard H.265/HEVC [?] with small modifications [41–46, 55–57]. A light field coding scheme using weighted binary patterns has been also investigated as a novel scheme without using H.265/HEVC. These studies basically focused on the light

field coding, but do not carefully consider a practical scenario for light field image/video transmission. Thus, I consider a practical transmission scenario where compressive 3D displays are used as receiving terminals. Under this scenario, either light fields or pre-calculated layer patterns should be transmitted from the sender to the receivers, so that the compressive displays at the receiver sides can reproduce the light fields. It should be noted that the layer patterns by themselves are compressive representations, because the entire light field (tens to hundreds of images) are reduced into only a few layer patterns. However, the layer patterns are different from natural images as can be seen from Figure 2.5(c), and thus, it is unclear how much the encoding errors of the layer patterns affect the quality of reproduced light fields. In this chapter, the coding efficiency of the light fields and layer patterns is experimentally investigated under the communication scenario with the compressive displays.

4.2 Transmission Framework for Compressive 3D Display

I assume a 3D video transmission system where the compressive displays are adopted as the receiving terminals. The receiver terminals finally need the layer patterns with which the light fields are reproduced. Under this scenario, two possible transmission frameworks can be considered as shown in Figures 4.1 and 4.2. In the first framework (i), light fields are encoded at the sender side and transmitted through the communication channel, and then the corresponding layer patterns are obtained at the receiver sides using the decoded light fields. Meanwhile, in the second framework (ii), layer patterns are calculated in advance and encoded at the sender side, and the received and decoded layer patterns are directly displayed at the receiver sides. Note that in the second framework (ii) the data size is greatly reduced at the point where a light field (tens to hundreds of images) is transformed into a set of layer patterns (only a few transmittance images); it is expected that the coding efficiency of the framework (ii) would be better than that of the framework (i). However,
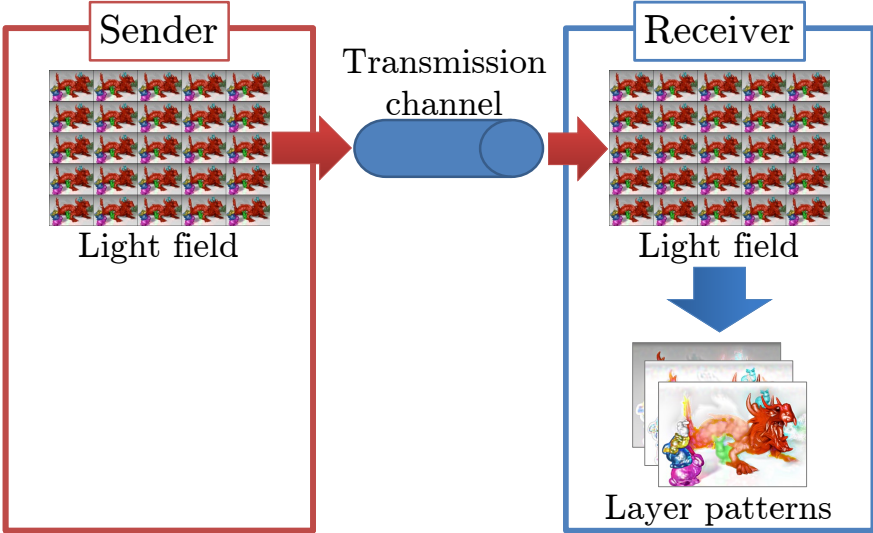
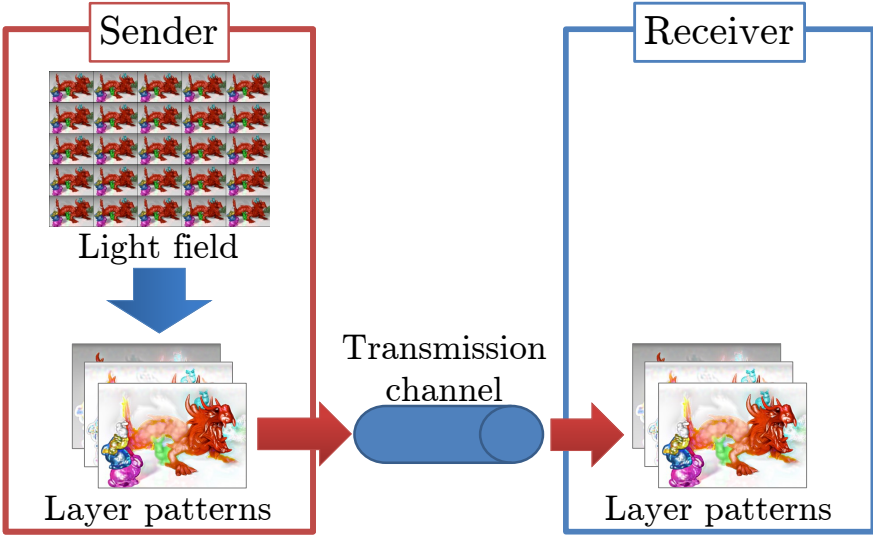Figure 4.1: Framework (i): transmitting light field

Figure 4.2: Framework (ii): transmitting layer patterns

the effect of encoding errors on the layer patterns should be taken into consideration, because they are significantly different from natural images.

4.3   Experimental Results

To verify the coding efficiency of two frameworks described above, simulative experiments were conducted using two light field still-image datasets and a light field video dataset. Published software is utilized for calculating layer patterns published by [64]. In experiments, a light field and layer patterns are encoded, and then reproduction of light field with the compressive display is simulated using decoded materials. The quality of the displayed light field is evaluated with PSNR against the original light field. The coding efficiency is evaluated with rate-distortion characteristics describing the trade-off between the total bits of the encoded light field or layer patterns and the PSNR of the displayed light field.

Light field datasets truck and amethyst from [25] were used as input light fields, which are RGB color images. The center view images of these dataset are shown in Figures 4.3 (a) and (b). Both datasets consist of 17×17 images, which were cropped to 640×480 and 384×512, respectively. The number of layers was set to three, and the number of time division was set to one (without time division multiplexing) and two. HEVC Test Model (HTM) 16.6 provided as H.265/HEVC reference software [53] was employed for encoding a light filed and layer patterns. For the first transmission framework, multi-view images are aligned in the raster-scan order and regarded as a single-viewpoint video. For the second framework, three layer-pattern images were also regarded as a single-viewpoint video. When the time division multiplexing was used, the first and second three patterns are individually regarded as single-viewpoint videos, and then the two videos were concatenated. The quantization parameter QP in HTM was set to 1, 10, 15, 22, 27, 32, 37, 42, and 50 to draw a curve of the rate-distortion trade-off.

Figure 4.4 shows the rate-distortion characteristic with truck and amethyst without time-division multiplexing. "Non-compressed" indicates the reproduction quality of the compressive display obtained with the original

(a) Truck   (b) Amethyst
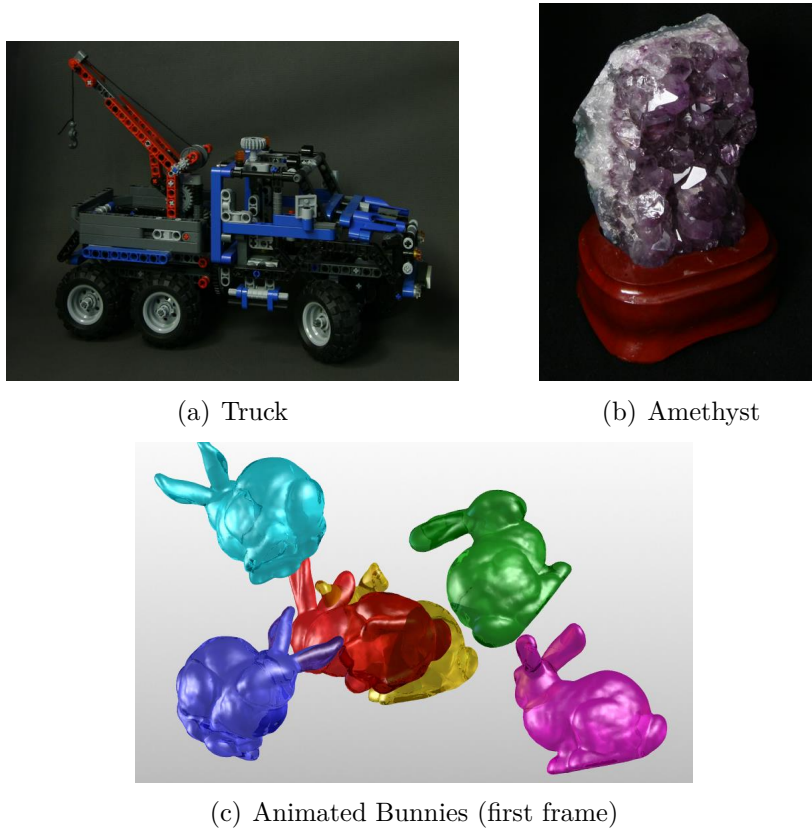

(c) Animated Bunnies (first frame)

Figure 4.3: Center view image of each dataset

light field datasets; it is the upper-bound performance of the communication scenario under consideration. The center view images of the reproduced truck without TDM are shown in Figure 4.5, where it is difficult to visually distinguish the difference among these images. Figure 4.4 demonstrates that framework (ii) outperforms framework (i) in terms of the coding efficiency; framework (ii) achieves lower bits than framework (i) with less distortion. However, the difference is not so significant as expected, given the fact that the number of images given to the HEVC encoder was 298 for framework (i) while it was only 3 for framework (ii). This means that light fields can be more efficiently compressed by using HEVC. Additionally, it seems that layer patterns are difficult to be compressed with HEVC because the layer patterns are different from natural image, for which HEVC and other video codec ware optimized. Figures 4.6
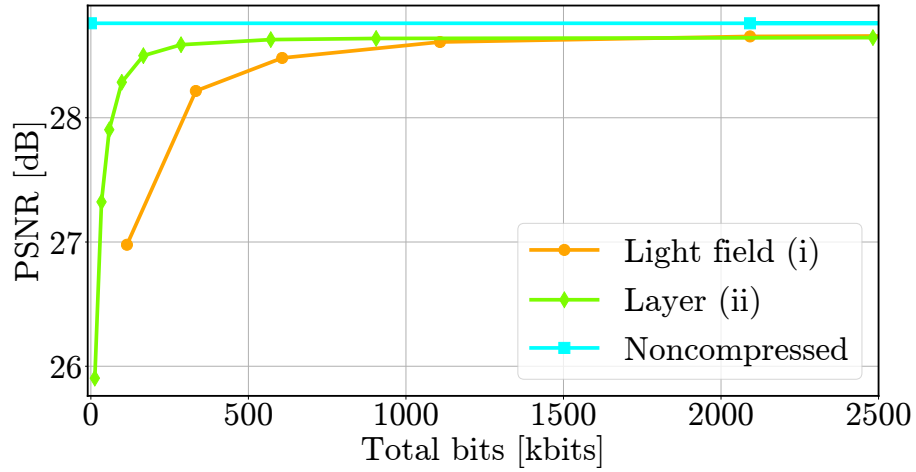
(a) Truck
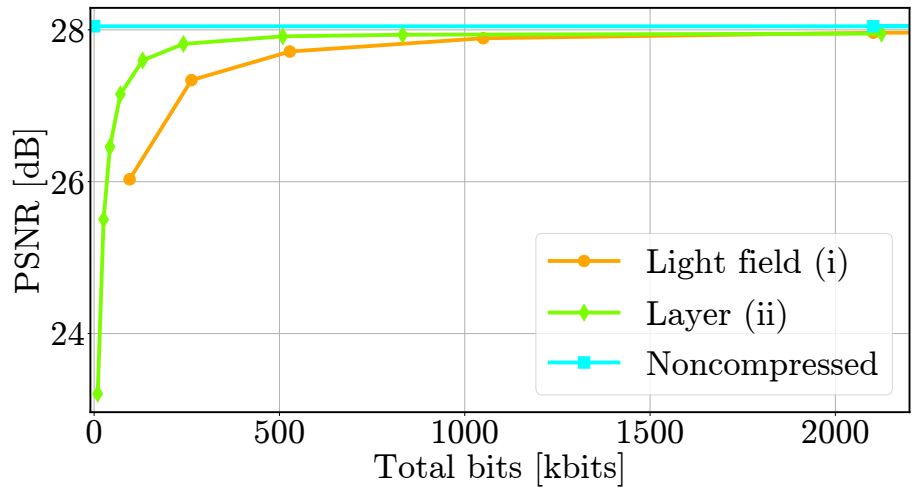
(b) Amethyst

Figure 4.4: Rate-distortion curves without TDM

(a) Noncompressed (28.76 dB)

(b) Light field (i) (28.68 dB)

(c) Layer (ii) (28.64 dB)

Figure 4.5: Reproduced center view without TDM at QP=10 (Truck)

(a) Entire bit range
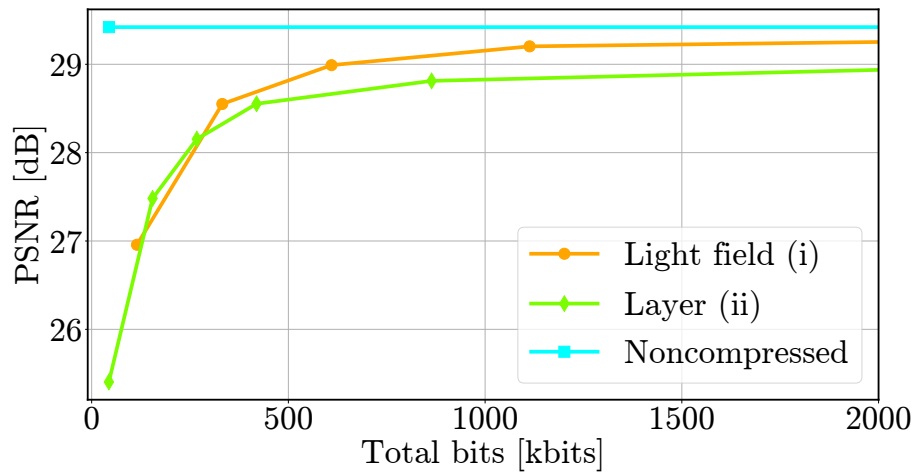
(b) Lower bit range only

Figure 4.6: Rate-distortion curves of truck with TDM

(a) Entire bit range

(b) Lower bit range only

Figure 4.7: Rate-distortion curves of amethyst with TDM

(a) Noncompressed (29.42 dB)
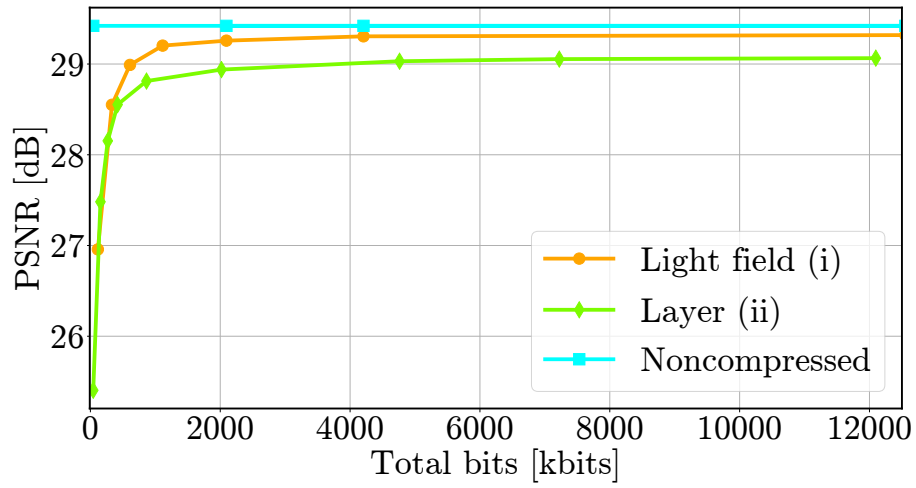

(b) Light field (i) (29.33 dB)
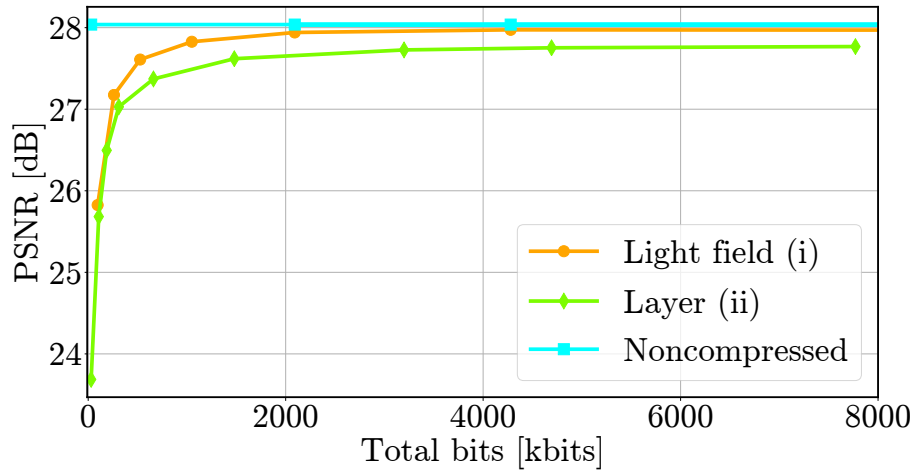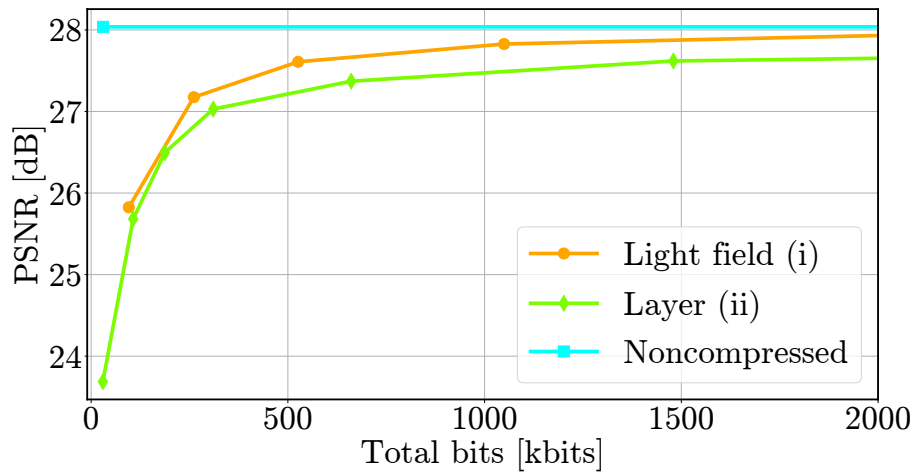

(c) Layer (ii) (29.05 dB)

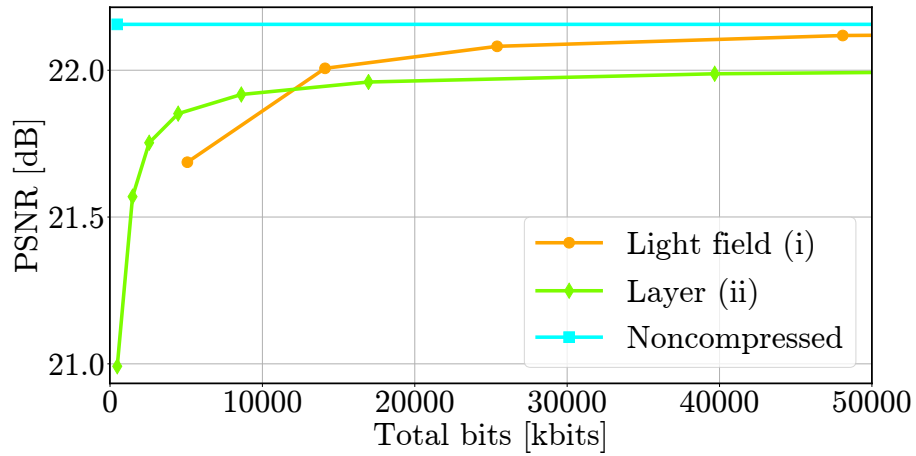Figure 4.8: Reproduced center view with TDM at QP=10 (Truck)

and 4.7 show the rate-distortion characteristic with time-division multiplexing. The center view images of the reproduced truck with TDM are shown in Figure 4.8. It is also difficult to visually identify the difference among these images. Using TDM doubles the number of layer patterns to be encoded, and thus, more total bits were required than that without TDM in framework (ii). In this case, framework (ii) is no longer superior to framework (i). However, framework (ii) still can achieve lower total bits than framework (i) in the case with the largest QP (QP=50) because of the small number of images to be encoded.

As a light field video dataset, Animated Bunnies from [23] was used. Figure 4.3 (c) shows its center view image of the first frame. Animated Bunnies was composed of 9×3 images with 89 frames and each of the images was cropped to 840×512. The number of layers was set to three, and time division multiplexing was not employed. For the first transmission framework, the light field video was treated as a multi-view videos and each video (corresponding to each viewpoint) was individually encoded using HTM. In this case, the light field video is regarded as 27 single-viewpoint videos, each of which consists of 89 frames. In the second framework, the patterns for each layer over time were treated as a video, and individually encoded using HTM. In this case, the set of layer patterns is regarded as 3 videos, each of which consists of 89 frames. QP was set to the same values as the experiments with the still images. The value of PSNR was obtained from the MSE calculated over all frames such like Equation (2.10) in Section 2.3, instead of taking the average of the PSNR values obtained from individual frames.

Figure 4.9 shows the rate-distortion characteristic of Animated Bunnies. The center view images of the reproduced first frame are shown in Figure 4.10. Again, frameworks (ii) surpasses framework (i) in the lower bit range. However, the upper bound of PSNRs obtained by frameworks (i) and (ii) is different. It could be because coding errors by HEVC are accumulated for layer patterns due to a large number of frames. The layer patterns are different from natural images as mentioned before, so that tendency of coding errors might be different and the amount of errors could be more accumulated during prediction procedure than that of
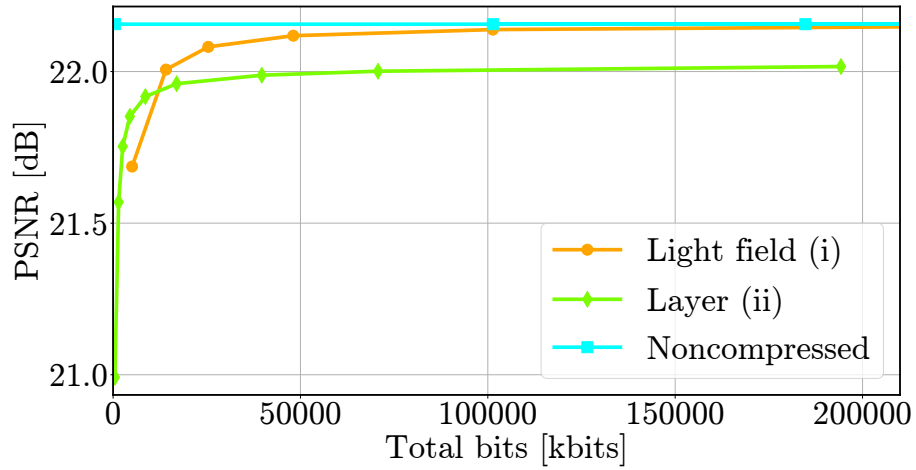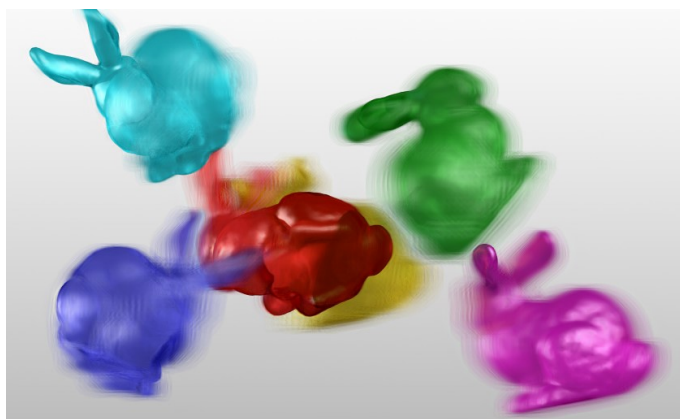
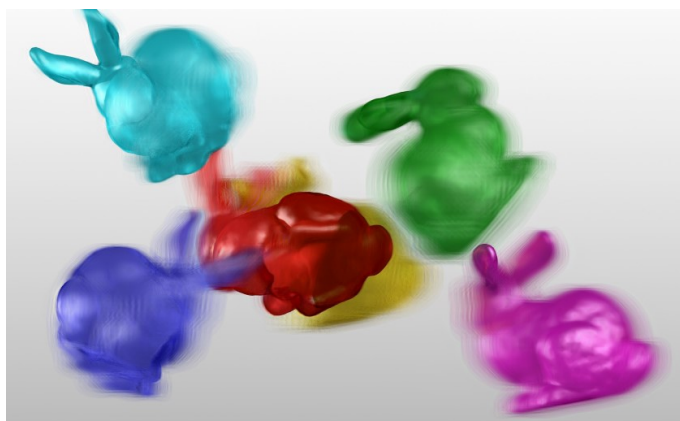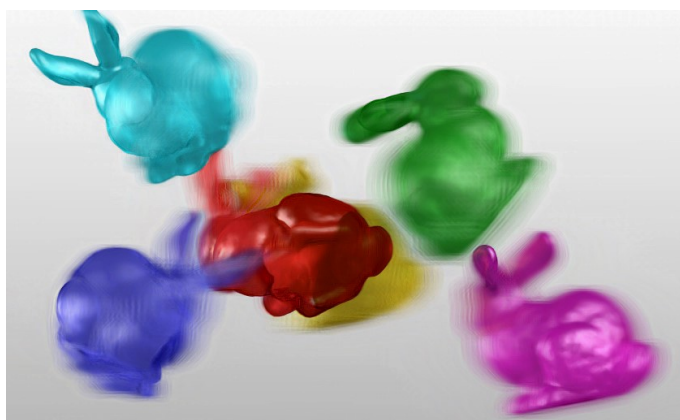(a) Entire bit range

(b) Lower bit range only

Figure 4.9: Rate-distortion curves of Animated Bunnies

(a) Noncompressed (22.16 dB)

(b) Light field (i) (22.15 dB)

(c) Layer (ii) (22.00 dB)

Figure 4.10: Reproduced center view of first frame at QP=10 (Animated Bunnies)

natural images.

To examine influence of the number of viewpoints for coding efficiency, additional simulative experiments were conducted. Truck dataset [25] is used for this experiments, but prepare 9×9 and 5×5 light fields by picking up a part of viewpoint images as shown in Figure 4.11, where viewpoint images are picked up in two manners to conduct experiments using light field with different viewpoint distances. Sparse means that viewpoint images are picked up so that viewpoint distance of a resulting light field becomes large while dense means that the distance becomes small. The number of layer patterns is three, and the light field and layer patterns are encoded using H.265/HEVC in the same manner as the previous experiments. Therefore, the number of frames of encoded video sequence is 81 and 25 for 9×9 and 5×5 light field, respectively. The coding efficiency is also evaluated with rate-distortion characteristics by following the previous experiments.

Figures 4.12 and 4.13 show the rate-distortion curves for dense and sparse datasets, respectively. Figure 4.12 indicates that framework (ii) achieves better rate-distortion performance than framework (i) for a 9×9 light field similarly to the previous experiments. However, the rate-distortion performance of framework (ii) is inferior to that of framework (i) for a 5×5 light field. This is because the difference of transmitted data size for 5×5 light fields in frameworks (i) and (ii) is smaller that that for 9×9 and 17×17 light fields. In the case of 5×5 light fields, the number of images is 25 but the number of layer patterns is still 3; thus, compression using H.265/HEVC overcomes this original data-size difference. The superiority of compressive expression is lost in the case of small viewpoint-number light fields. This tendency can be found also in the sparse light field as shown in Figure 4.13.

4.4 Summary

In this chapter, the coding efficiency of light fields was examined for compressive 3D displays as a practical application of light field coding. I supposed a transmission system where such compressive displays were

adopted as the receiving terminals and compared two transmission frameworks where the light fields or the layer patterns were encoded and transmitted. Simulative experiments demonstrated that the latter framework achieves better rate-distortion performance in lower bit ranges. However, this superiority was lost in the case of using time-division multiplexing and light fields with small-number viewpoints. Additionally, I confirmed that compressing layer patterns using HEVC is more difficult than light field images because layer patterns are different from natural images.
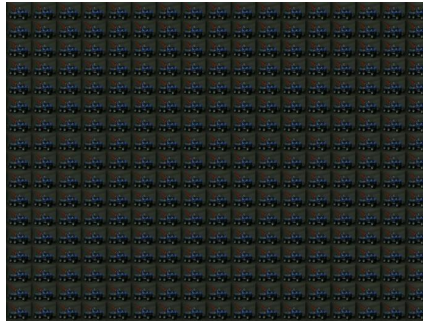
17×17 views
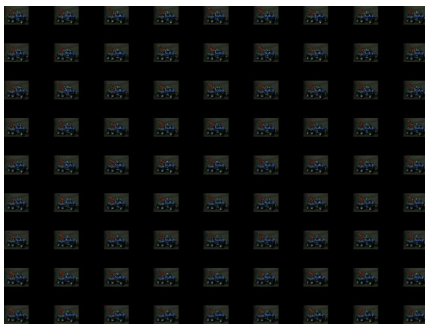
9×9 views (sparse)
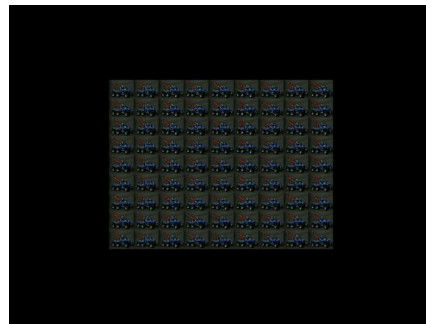
9×9 views (dense)
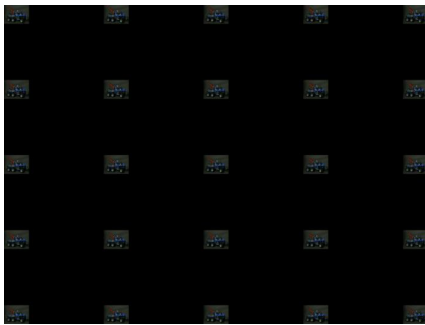
5×5 views (sparse)
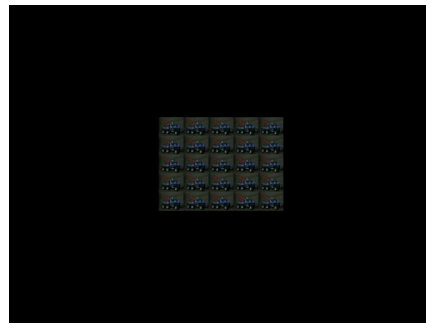
5×5 views (dense)

Figure 4.11: 9×9 and 5×5 light fields from 17×17 truck dataset
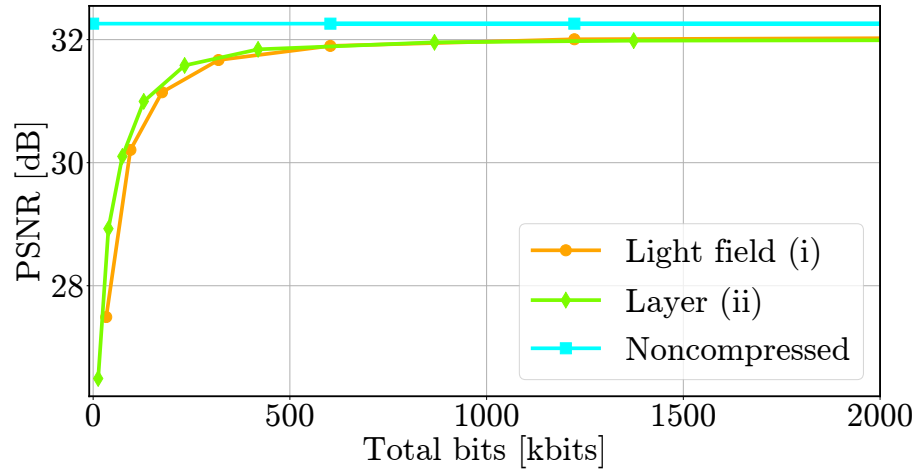
(a) 9×9 light field
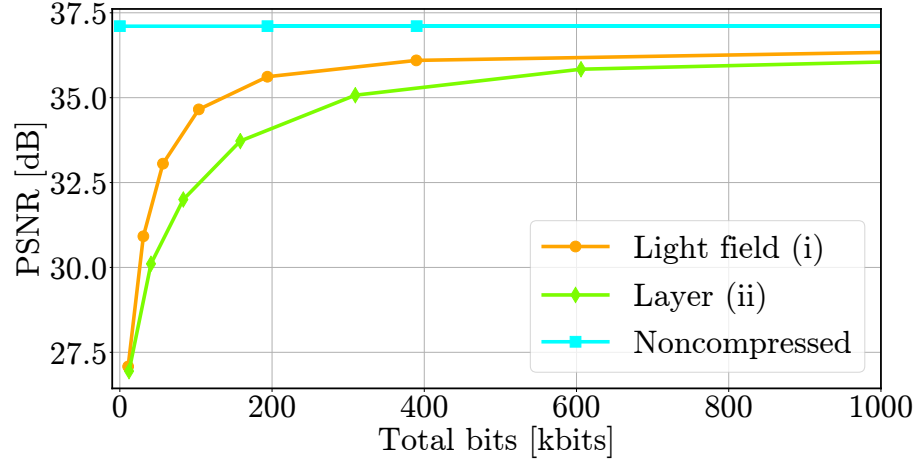
(b) 5×5 light field

Figure 4.12: Rate-distortion curves of dense dataset
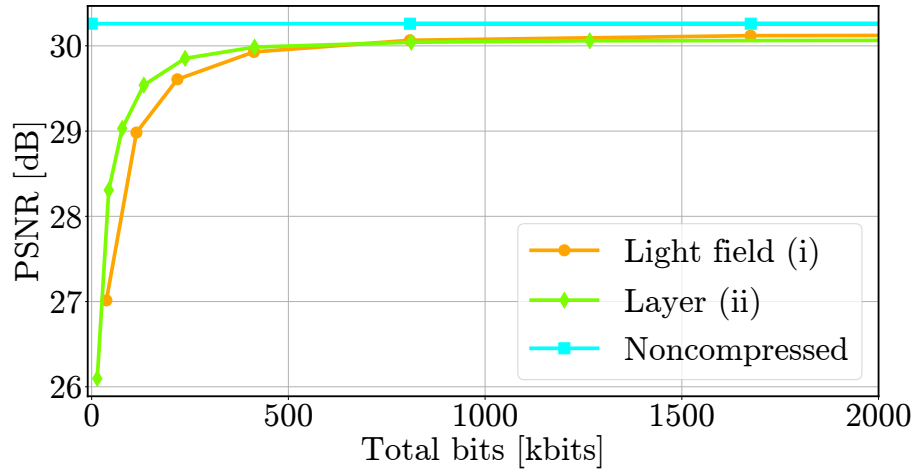
(a) 9 × 9 light field
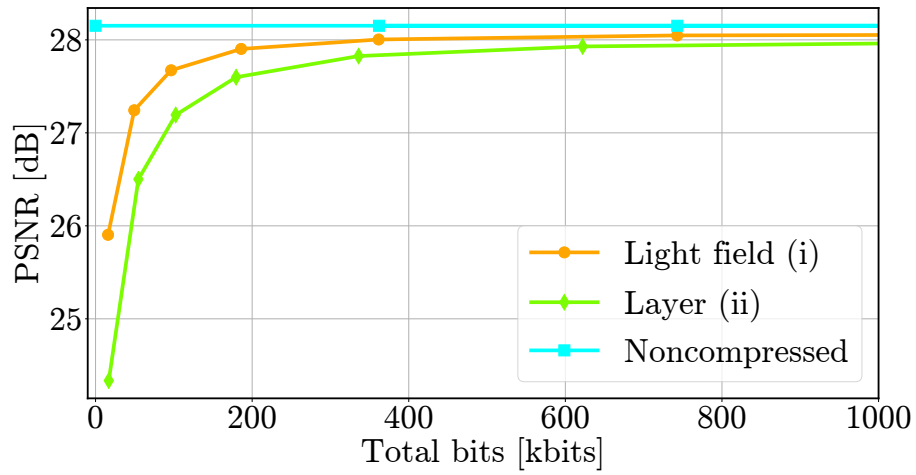
(b) 5 × 5 light field

Figure 4.13: Rate-distortion curves of sparse dataset

Chapter 5

Relationship to Real-World Data Circulation

This chapter discusses the relationship between Real-World Data Circulation (RWDC) [51] and our study. The concept of RWDC is firstly introduced, and then data circulation in light field processing is described. At the end of this chapter, data circulation in a study on light field coding is also described.

5.1   Real-World Data Circulation

RWDC is a novel field which involves engineering, information science, medicine, and economics. RWDC makes data circulation in real world by integrating acquisition, analysis, and implementation of real-world data in order to create social values. Thanks to the development of information technology, various kinds of data can be obtained and accumulated from real world. To produce social values, it is important to not only just create a product and service but also reflect demands of people in real world. "Data acquisition" involves observing enormous digital data of real-world phenomenon. "Data analysis" reveals hidden demands from the acquired real-world data based on information science. Then, "implementation" realizes innovative products and services based on the analysis result. After publishing the implemented products and services, new data from users can be obtained through various methods of "data acquisition".
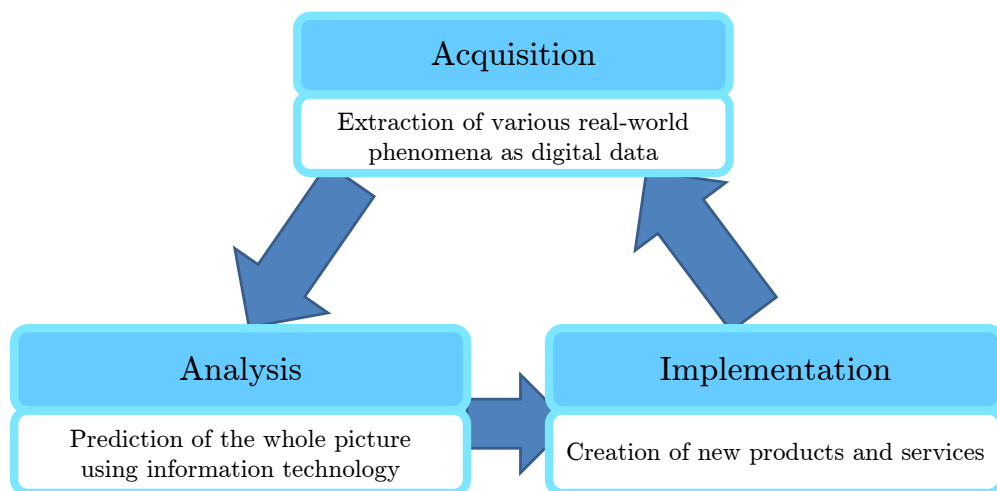
Figure 5.1: Real-world data circulation
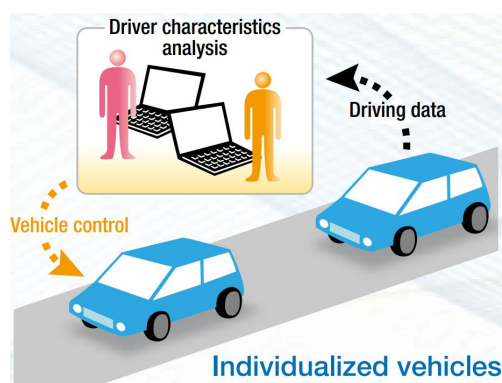
Figure 5.2: Real-world data circulation in autonomous driving

By analysing the newly obtained data, the better or novel products and service can be implemented. This can be considered as circulation of data as shown in Figure 5.1. This circulation can be utilized for many applications in various fields where data caused by real-world phenomenon can be obtained.

An autonomous driving system can be considered as an example of data circulation. Figure 5.2 [51] conceptually illustrates the data circulation in an autonomous driving system. During a car driving, driving data such as speed, position, car camera image/video, and so on can be
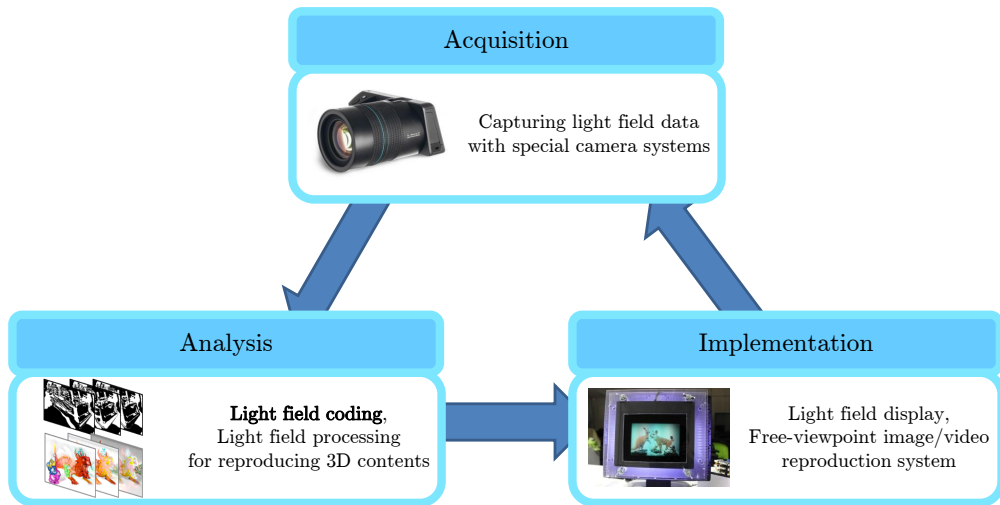
Figure 5.3: Real-world data circulation in light field processing

measured by using various sensors. Characteristics of drivers and incidents which should be avoided can be estimated by analyzing the acquired data. Localization map can also be obtained by the analysis. In addition, important knowledge for autonomous driving can be obtained by analyzing driving data. The vehicle control system can be updated using based on the result of analysis. Autonomous driving system can continuously be improved by repeating the data circulation. This data circulation is able to reflect a transition of typical driver characteristics in the autonomous driving system.

5.2 Data Circulation in Light Field Processing

This section discusses the data circulation in light field processing. I consider that light field processing consists of data acquisition, analysis, and implementation in real world as shown in Figure 5.3. For data acquisition, light field data is captured by using camera systems such as a multi-camera array system and light field camera. For data analysis, various light field processing technology can be considered as a light field analysis such as
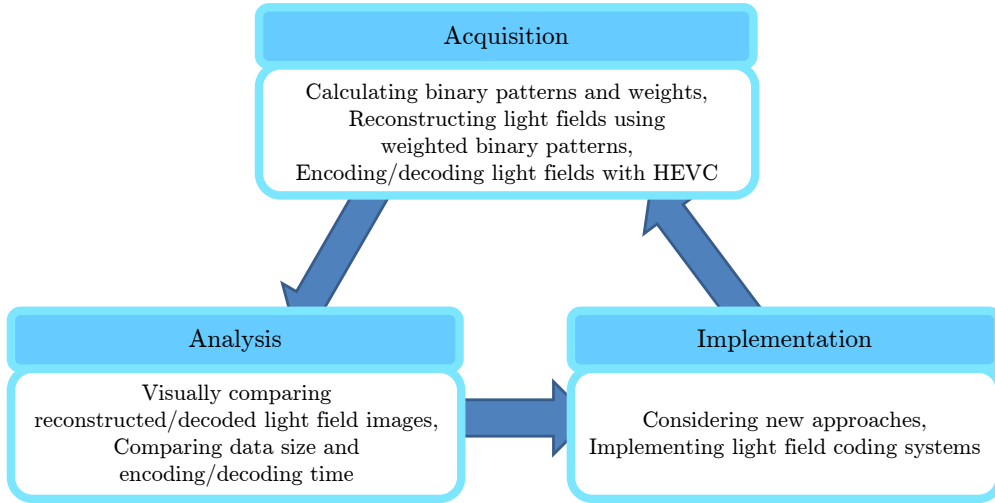
Figure 5.4: Real-world data circulation in this thesis

compression, calculation of layer patterns for a compressive display, 3D object shape estimation, etc. For implementation, 3D image/video systems such as light field display and free-viewpoint image/video reproduction system can be realized by using the analyzed data. Our laboratory inclusively studies on light field capturing, analysis, and display systems; this thesis especially focuses on light field compression classified into analysis part in the data circulation. As one of examples for light field data circulation, light field streaming system. A target 3D object is captured by using a light field camera; and then, captured light field data should be compressed to transmit it. 3D display systems such as the compressive display reproduces the 3D object using transmitted light field data. For the output data of the display systems, subjective evaluation can be obtained from viewers. By analyzing the result of evaluation, what is the important for 3D display systems would be found. Although what is important for 3D display system depends on the purpose of the system, each circulation stage could also be improved depending on the subjective evaluation; consequently, light field data circulation could be realized.

More specifically, I also consider data circulation in this study as shown in Figure 5.4. I first find room for improving compression schemes by considering current performance of coding schemes and demands for practi-

cal systems. The performance of existing schemes such as rate-distortion characteristics, encoding/decoding times, and reconstructed images are obtained by experiments, where an input light field is encoded/decoded and data size, computational time for encoding/decoding, and visual quality are obtained. In the case of methods based on the modern video coding standards, these data can be obtained by decoding the bitstream, which is an output of the coding standards. Then, room for improving the scheme can be discussed based on the obtained data. For instance, degradation of decoded images can be confirmed by visual comparison and cause of the degradation could be found by carefully observing it. When I considered the method of applying disparity compensation to the progressive coding, large degradation of decoded images was able to be seen on a part having large disparities by comparing the decoded images with original one. Based on the result of the discussion, I could consider approaches for light field compression. After considering approaches, I actually implement the approaches and conduct experiments. Through the experiments, I can obtain new data of the considered light field coding scheme and discuss the remaining issues and new approaches based on the obtained data. This procedure could be considered as one of data circulation.

Chapter 6

Conclusion and Future Works

This thesis has focused on a light field coding technology and its practical application. I have especially discussed a light field coding scheme using weighted binary patterns and coding efficiency of light field transmission frameworks for a compressive 3D display.

Chapter 2 has described background knowledge and related works to get a better understanding of light field coding problems. The definition of light field was first introduced, and several architectures for a 3D display were compared and the advantages of a compressive 3D display was described. The basic knowledge of modern video coding standards was briefly reviewed since they can be applied to light field coding. As related works, studies on light field coding utilizing modern video coding standards were described. This chapter described that the related works basically focused on applying the modern video coding standards to light field coding with small modifications.

Chapter 3 has discussed a light field coding scheme using weighted binary patterns. This chapter first introduced the light field coding scheme with weighted binary patterns and optimization process for encoding, i.e. derivation of binary patterns and weights. The key concept of the scheme is approximating a light field using a weighted sum of only several binary patterns. We have proposed the scheme based on an idea that the standard video coding techniques are not necessarily the most suitable for a dense light field. The proposed scheme achieves comparable performance to that of the video coding standards applied to a light field

though the video coding standards have been optimized using enormous labor and time. An important advantage of this scheme is dramatically simple decoding process, which allows us to implement a faster and less power-hungry decoder. Then, I have also proposed a novel method of applying disparity compensation framework to the binary patterns to address a trade-off problem between computational complexity of the encoding process and rate-distortion performance in the proposed scheme. Experimental results show that the proposed method improves rate-distortion performance while keeping feasible computational complexity. This improvement is the first contribution of this thesis.

Chapter 4 has discussed a practical light field coding scenario where a compressive 3D display is adopted as a receiving terminal. First, the detail of practical scenario was described. Then, two light field transmission frameworks under the scenario were introduced. First framework transmits a light field directly and layer patterns are calculated with the transmitted light field in receiver side. Second framework calculates layer patterns from the original light field in advance and transmits the obtained patterns. Experimental results demonstrate that the second framework achieves superior rate-distortion performance to that of first framework in lower bit range; namely, coding efficiency of layer patterns is better than that of a light field. This investigation of coding efficiency under the practical scenario is the second contribution of this thesis.

Chapter 5 has described a relationship between real-world data circulation (RWDC) and this thesis. RWDC is a novel academic field including engineering, information science, economics, and medicine proposed by a special curriculum for Ph.D. students named real-world data circulation leaders, program for leading graduate schools, Nagoya University. The basic concept and example of RWDC were first introduced; then, I discussed data circulation in light field processing and a study on light field coding. Light field processing includes data circulation and the concept of data circulation is helpful for executing this study.

As future works of this study, I first should consider further improvement and extension of light field coding using weighted binary patterns. In practical applications, light field images with a high resolution is desir-

able to provide high quality 3D experiences, and various kinds of 3D scenes should be handled. Although the effectiveness of the proposed method was demonstrated, it is examined for only limited cases where a resolution of light field images is small. Approximating larger light field images using weighted binary patterns would be difficult because they include large disparities and their redundancy among viewpoints becomes small. This difficulty should be addressed for further improvements of the scheme. Additionally, we investigated how to apply the baseline and progressive methods to color light field images in [47], but the proposed method using disparity compensation is applied to only grayscale light field images in this thesis; thus, I also should consider how to apply disparity compensation framework to color light field images. After addressing the above issues, I should consider how to apply the proposed method to practical applications such as a light field display system and a free-viewpoint synthesis system.

Next, I will further be able to investigate the coding efficiency of a light field and layer patterns. I simply adapted H.265/HEVC to a light field and layer patterns in this thesis. However, there are many researches for efficiently compressing a light field using the video coding standards and applying the modern video codecs to layer patterns might not be suitable because the codecs have been optimized for natural image/video sequences and layer patterns are different from natural images. There is a room for further investigation of coding efficiency in the scenario.

Acknowledgments

I would like to first sincerely express my great gratitude to my supervisor, Professor Toshiaki Fujii, for receiving me as his student and giving me the opportunity of research. Thanks to his forceful and kindful encouragement and support, I could complete the doctoral course and have learned not only technical knowledge but also how to face enormous difficulties in life. I would like to also thank to his special support for my activities in the special curriculum, real-world data circulation leaders. I could continue to actively work on my research by his warm support and guidance.

I would like to sincerely thank Associate Professor Keita Takahashi in the lab. for often taking a long times to discuss my research. His precise discussion and direction in research have always helped my research work, and I learned what researcher should be from his sincere attitude to research. I also would like to express my gratitude to Associate Professor Ryutaroh Matsumoto in the lab., for his thoroughly logical insights and directions, and having time for discussing my researches.

I would like to sincerely appreciate for Professor Takaya Yamazato and Professor Ichiro Matsuda of Tokyo University of Science having time for reviewing my thesis. I also would like to thank Professor Mehrdad Teratani of Université Libre de Bruxelles, for suggestions and kind encouragement in Fujii Lab. He kindly supported me for not only research but also activities in the special curriculum while belonging to Nagoya University and the curriculum.

My study is supported by the special curriculum called real-world data circulation leaders, program for leading graduate schools, Nagoya University. The curriculum gave me numerous opportunities to get knowledge and experiences of not only research but also global activities. I would like to appreciate for great support from the curriculum and to thank Professor Kazuya Takeda, a coordinator of the curriculum, and all designed professors in the curriculum.

I would like to thank Professor Yoshimitsu Kuroki of National Institute of Technology, Kurume College who was my supervisor when I received bachelor's degrees. He provided me various opportunities to learn the foundation of research and to get global experiences.

Many thanks to all students for making my time at Fujii Lab such a pleasure. Finally, I wish to thank my parents who gave me special support in my life.

Bibliography

[1] M. Levoy and P. Hanrahan, "Light field rendering," Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '96, New York, NY, USA, pp.31–42, ACM, 1996.

[2] S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen, "The lumigraph," ACM SIGGRAPH, pp.43–54, 1996.

[3] T. Fujii, T. Kimoto, and M. Tanimoto, "Ray space coding for 3D visual communication," Picture Coding Symposium (PCS), pp.447–451, 1996.

[4] B. Wilburn, N. Joshi, V. Vaish, E.V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy, "High performance imaging using large camera arrays," ACM Transactions on Graphics, vol.24, no.3, pp.765–776, July 2005.

[5] T. Fujii, K. Mori, K. Takeda, K. Mase, M. Tanimoto, and Y. Suenaga, "Multipoint measuring system for video and sound - 100-camera and microphone system," 2006 IEEE International Conference on Multimedia and Expo, pp.437–440, July 2006.

[6] E.H. Adelson and J.Y.A. Wang, "Single lens stereo with a plenoptic camera," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.14, no.2, pp.99–106, Feb. 1992.

[7] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," Computer Science Technical Report CSTR, vol.2, no.11, pp.1–11, 2005.

[8] A. Lumsdaine and T. Georgiev, "The focused plenoptic camera," IEEE International Conference on Computational Photography (ICCP), pp.1–8, 2009.

[9] C. Perwaß and L. Wietzke, "Single lens 3D-camera with extended depth-of-field," Human Vision and Electronic Imaging XVII, pp.45–59, Feb. 2012.

[10] T. Georgiev and A. Lumsdaine, "The multi-focus plenoptic camera," Digital Photography VIII, pp.69–79, Jan. 2012.

[11] S. Wanner and B. Goldluecke, "Variational light field analysis for disparity estimation and super-resolution," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.36, no.3, pp.606–619, March 2014.

[12] T. Suzuki, K. Takahashi, and T. Fujii, "Disparity estimation from light fields using sheared EPI analysis," 2016 IEEE International Conference on Image Processing (ICIP), pp.1444–1448, Sep. 2016.

[13] T. Suzuki, K. Takahashi, and T. Fujii, "Sheared EPI analysis for disparity estimation from light fields," IEICE Transactions on Information and Systems, vol.E100.D, no.9, pp.1984–1993, 2017.

[14] G. Houben, S. Fujita, K. Takahashi, and T. Fujii, "Fast and robust disparity estimation for noisy light fields," 2018 25th IEEE International Conference on Image Processing (ICIP), pp.2610–2614, Oct. 2018.

[15] L. Shi, H. Hassanieh, A. Davis, D. Katabi, and F. Durand, "Light field reconstruction using sparsity in the continuous Fourier domain," ACM Transactions on Graphics, vol.34, no.1, pp.12:1–12:13, Dec. 2014.

[16] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, "Deepstereo: Learning to predict new views from the world's imagery," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.5515–5524, June 2016.

[17] D. Lanman, M. Hirsch, Y. Kim, and R. Raskar, "Content-adaptive parallax barriers: Optimizing dual-layer 3D displays using low-rank light field fac-

torization," ACM Transactions on Graphics, vol.29, no.6, pp.163:1–163:10, 2010.

[18] G. Wetzstein, D. Lanman, W. Heidrich, and R. Raskar, "Layered 3D: Tomographic image synthesis for attenuation-based light field and high dynamic range displays," ACM Transactions on Graphiscs, vol.30, no.4, pp.95:1–95:12, 2011.

[19] G. Wetzstein, D. Lanman, M. Hirsch, and R. Raskar, "Tensor displays: Compressive light field synthesis using multilayer displays with directional backlighting," ACM Transactions on Graphics, vol.31, no.4, pp.80:1–80:11, July 2012.

[20] K. Maeno, H. Nagahara, A. Shimada, and R. Taniguchi, "Light field distortion feature for transparent object recognition," IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.2786–2793, 2013.

[21] T. Saito, Y. Kobayashi, K. Takahashi, and T. Fujii, "Displaying real-world light fields with stacked multiplicative layers: Requirement and data conversion for input multiview images," Journal of Display Technology, vol.12, no.11, pp.1290–1300, Nov. 2016.

[22] S. Lee, C. Jang, S. Moon, J. Cho, and B. Lee, "Additive light field displays: Realization of augmented reality with holographic optical elements," ACM Transactions on Graphics, vol.35, no.4, pp.60:1–60:13, July 2016.

[23] "Synthetic light field archive." `https://www.media.mit.edu/~gordonw/SyntheticLightFields/`, 2008.

[24] M. Rerabek and T. Ebrahimi, "New light field image dataset," 8th International Conference on Quality of Multimedia Experience(QoMEX), pp.1–2, 2016.

[25] "The (new) stanford light field archive." `http://lightfield.stanford.edu/lfs.html`, 2008.

[26] F.E. Ives, "Parallax stereogram and process of making same." U.S. Patent 725, 567, 1903.

[27] H. Isono, M. Yasuda, and H. Sasazawa, "Autostereoscopic 3D LCD display using LCD-generated parallax barrier," Japan Display '92, vol.303, pp.303–306, 1992.

[28] T. Peterka, R.L. Kooima, D.J. Sandin, A. Johnson, J. Leigh, and T.A. DeFanti, "Advances in the dynallax solid-state dynamic parallax barrier autostereoscopic visualization display system," IEEE Transactions on Visualization and Computer Graphics, vol.14, no.3, pp.487–499, 2008.

[29] S.K. Kim, K.H. Yoon, S.K. Yoon, and H. Ju, "Parallax barrier engineering for image quality improvement in an autostereoscopic 3D display," Optics Express, vol.23, no.10, pp.13230–13244, 2015.

[30] G. Lippmann, "Épreuves réversibles donnant la sensation du relief," Journal de Physique Théorique et Appliquée, vol.7, no.1, pp.821–825, 1908.

[31] T. Okoshi, Three-Dimensional Imaging Techniques, Academic Press, 1976.

[32] R. Börner, "Autostereoscopic 3D-imaging by front and rear projection and on flat panel displays," Displays, vol.14, no.1, pp.39–46, 1993.

[33] M. McCormick, "Integral 3D imaging for broadcast," Proceedings of the 2nd International Display Workshop, pp.77–80, 1995.

[34] J. Arai, F. Okano, M. Kawakita, M. Okui, Y. Haino, M. Yoshimura, M. Furuya, and M. Sato, "Integral three-dimensional television using a 33-megapixel imaging system," Journal of Display Technology, vol.6, no.10, pp.422–430, 2010.

[35] D. Lanman, M. Hirsch, Y. Kim, and R. Raskar, "Content-adaptive parallax barriers: Optimizing dual-layer 3D displays using low-rank light field factorization," ACM Transactions on Graphics, vol.29, no.6, pp.163:1–163:10, Dec. 2010.

[36] G. Wetzstein, D. Lanman, W. Heidrich, and R. Raskar, "Layered 3D: Tomographic image synthesis for attenuation-based light field and high dynamic range displays," ACM Transactions on Graphics, vol.30, no.4, pp.95:1–95:12, July 2011.

[37] S. Suyama, H. Takada, and S. Ohtsuka, "A direct-vision 3-D display using a new depth-fusing perceptual phenomenon in 2-D displays with different depths," IEICE Transactions on Electronics, vol.85, no.11, pp.1911–1915, Nov. 2002.

[38] D. Lanman, G. Wetzstein, M. Hirsch, W. Heidrich, and R. Raskar, "Polarization fields: Dynamic light field display using multi-layer lcds," ACM Transactions on Graphics, vol.30, no.6, pp.186:1–186:10, Dec. 2011.

[39] "Light field display project." `http://www.fujii.nuee.nagoya-u.ac.jp/~takahasi/Research/LFDisplay/index.html`, 2018.

[40] "ITU-T Recommendation H.265 and ISO/IEC 23008-2: High efficiency video coding," Feb., 2018.

[41] G. Tech, Y. Chen, K. Müller, J.R. Ohm, A. Vetro, and Y.K. Wang, "Overview of the multiview and 3D extensions of high efficiency video coding," IEEE Transactions on Circuits and Systems for Video Technology, vol.26, no.1, pp.35–49, Jan. 2016.

[42] Y. Li, M. Sjöström, R. Olsson, and U. Jennehag, "Efficient intra prediction scheme for light field image compression," 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp.539–543, May 2014.

[43] A. Vieira, H. Duarte, C. Perra, L. Tavora, and P. Assuncao, "Data formats for high efficiency coding of lytro-illum light fields," 2015 International Conference on Image Processing Theory, Tools and Applications (IPTA), pp.494–497, Nov. 2015.

[44] D. Liu, L. Wang, L. Li, Z. Xiong, F. Wu, and W. Zeng, "Pseudo-sequence-based light field image compression," 2016 IEEE International Conference on Multimedia Expo Workshops (ICMEW), pp.1–4, July 2016.

[45] F. Hawary, C. Guillemot, D. Thoreau, and G. Boisson, "Scalable light field compression scheme using sparse reconstruction and restoration," 2017 IEEE International Conference on Image Processing (ICIP), pp.3250–3254, Sep. 2017.

[46] X. Jiang, M.L. Pendu, R.A. Farrugia, and C. Guillemot, "Light field compression with homography-based low-rank approximation," IEEE Journal of Selected Topics in Signal Processing, vol.11, no.7, pp.1132–1145, Oct. 2017.

[47] K. Komatsu, K. Isechi, K. Takahashi, and T. Fujii, "Light field coding using weighted binary images," IEICE Transactions on Information and Systems, vol.E102-D, no.11, pp.2110–2119, 2019.

[48] G. Koutaki, "Binary continuous image decomposition for multi-view display," ACM Transactions on Graphics, vol.35, no.4, pp.69:1–69:12, July 2016.

[49] K. Komatsu, K. Takahashi, and T. Fujii, "Multi-view image coding using disparity-compensated and weighted binary patterns," Proceedings of International Workshop on Advanced Image Technology (IWAIT) 2019, 2019.

[50] Y. Kobayashi, S. Kondo, K. Takahashi, and T. Fujii, "A 3-D display pipeline: Capture, factorize, and display the light field of a real 3-D scene," ITE Transactions on Media Technology and Applications, vol.5, no.3, pp.88–95, 2017.

[51] "Nagoya university program for leading graduate schools: Graduate program for real-world data circulation leaders." `http://www.rwdc.is.nagoya-u.ac.jp/index-e.php`, 2018.

[52] E.H. Adelson and J.R. Bergen, "The plenoptic function and the elements of early vision," Computational Models of Visual Processing, pp.3–20, 1991.

[53] G.J. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," IEEE Transactions on Circuits and Systems for Video Technology, vol.22, no.12, pp.1649–1668, Dec. 2012.

[54] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," Proceedings of the ITU-T Video Coding Experts Group (VCEG) Thirteenth Meeting, 2001.

[55] F. Dai, J. Zhang, Y. Ma, and Y. Zhang, "Lenselet image compression scheme based on subaperture images streaming," 2015 IEEE International Conference on Image Processing (ICIP), pp.4733–4737, Sep. 2015.

[56] S. Zhao, Z. Chen, K. Yang, and H. Huang, "Light field image coding with hybrid scan order," 2016 Visual Communications and Image Processing (VCIP), pp.1–4, Nov. 2016.

[57] R. Conceição, M. Porto, B. Zatt, and L. Agostini, "LF-CAE: Context-adaptive encoding for lenslet light fields using HEVC," 2018 25th IEEE International Conference on Image Processing (ICIP), pp.3174–3178, Oct. 2018.

[58] K. Imaeda, K. Isechi, K. Takahashi, T. Fujii, Y. Bando, T. Miyazawa, S. Takamura, and A. Shimizu, "LF-TSP: Traveling salesman problem for HEVC-based light-field coding," 2019 IEEE Visual Communications and Image Processing (VCIP), pp.1–4, Dec. 2019.

[59] R.J. Gove, "DMD display systems: The impact of an all-digital display," Society for Information Display International Symposium, pp.1–12, June 1994.

[60] L.J. Hornbeck, "Projection displays and MEMS: timely convergence for a bright future," SPIE Micromachining and Microfabrication Process Technology, pp.27–40, 1995.

[61] "Light field compression project." `http://www.fujii.nuee.nagoya-u.ac.jp/~takahasi/Research/LFCompression/index.html`, 2018.

[62] "Fraunhofer heinrich hertz institute high efficiency video coding (HEVC)." https://hevc.hhi.fraunhofer.de, 2018.

[63] F. Bossen, "Common HM test conditions and software reference configurations," 2013.

[64] K. Takahashi, Y. Kobayashi, and T. Fujii, "From focal stack to tensor light-field display," IEEE Transactions on Image Processing, vol.27, no.9, pp.4571–4584, Sep. 2018.

Publications

Journal

[1] Kohei Isechi, Yuto Kobayashi, Keita Takahashi, and Toshiaki Fujii, "Light Field Compression for Compressive 3D Display," International Journal of Machine Learning and Computing (IJMLC), vol. 9, no. 2, pp. 208–212, April 2019.

[2] Koji Komatsu, Kohei Isechi, Keita Takahashi, and Toshiaki Fujii, "Light Field Coding Using Weighted Binary Images," IEICE Transactions on Information and Systems, vol. E102-D, no. 11, pp. 2110–2119, November 2019.

[3] Kohei Isechi, Keita Takahashi, and Toshiaki Fujii, "Disparity Compensation Framework for Light-Field Coding Using Weighted Binary Patterns," ITE Transactions on Media Technology and Applications, vol. 9, no. 1, pp. 40–48, January 2020.

International conference

[1] Kohei Isechi, Yuto Kobayashi, Keita Takahashi, and Toshiaki Fujii, "Light Field Compression for Compressive 3D Display," 2018 International Conference on Computer and Communication Systems (ICCCS 2018), (2018.4).