HEVC-based Light-field Coding using Basis Images and Frame Reordering

Kota Imaeda†, Keita Takahashi (member)†, Toshiaki Fujii (member)†,
Yukihiro Bandoh (member)††, Seishi Takamura (member)††, Hideaki Kinata (member)††

Abstract An efficient coding method for light fields (LFs) is presented. The method is based on a sophisticated video coding standard called High Efficiency Video Coding (HEVC), but does not directly encode the LF images using the HEVC codec. Instead, the LF images are first transformed into a set of smaller number of images, called basis images, to remove the redundancies among the images. The basis images are then reordered to produce a temporally smooth video sequence, which is finally encoded using the HEVC codec. In the decoding process, the decoded frames are inversely transformed into the original LF. The first and final transformations are modeled using neural networks and optimized for the target LF. The frame reordering is formulated as a traveling salesman problem (TSP) and solved using a greedy method. The experimental results show that our method can achieve better rate-distortion performance than other HEVC-based light-field coding methods.

Key words: Light field, HEVC, compression

1. Introduction

Light fields (LFs) [1,2] are used to describe 3D visual information as light rays in 3D space and are utilized in various applications such as depth estimation [3], refocusing, and 3D display [4]. LF information can now be captured with light-field cameras [5,6], such as Lytro cameras. An LF is usually represented as a set of multi-view images, i.e., dozens of images aligned on a 2D squared grid, viewing the same objects from slightly different angles. In its original format, such an LF has a significant amount of data, so developing efficient compression schemes is an important research objective, as evidenced by the many reported studies [7–14]. Among the reported schemes, we focused on a coding scheme where the given LF (multi-view) images are regarded as a sequence of temporal video frames and are encoded using the latest video coding standard, the High Efficiency Video Coding (HEVC) [15,16].

To enhance the rate-distortion performance of this scheme, we focused on two important points. The first point is the issue of how to determine the frame order of the LF images. The frame order affects the temporal smoothness of the generated video sequence, which in turn affects rate-distortion performance. Various frame orders in addition to the straightforward raster order have been tested [12–14]. Furthermore, to enhance performance, adaption of the frame order to the specific LF rather than using a fixed order for all LFs has been proposed [17,18]. The second point is the limited ability of video codecs to handle redundancy among multiple images. Although inter-frame prediction schemes have become increasingly sophisticated, the images are still treated as individual frames. Instead of encoding all the images, it would be more efficient to encode only a limited number of images (we call them “basis images”) in which the information in the original images is “condensed” in some way. The result would be a set of basis images that are video codec friendly; i.e., they would be efficiently encoded as a natural video sequence.

Keeping these points in mind, we develop an HEVC-based LF coding method that uses basis images and optimized frame reordering. In this method, the set of original LF images is first transformed into a set of basis images, which are then reordered to create a temporally smooth video sequence that is finally encoded using the HEVC codec. In the decoding process, the frames decoded from the HEVC bit-stream are inversely transformed into the original light-field images. The first and final transformations are modeled using neural networks and optimized for the target LF. The frame reordering is formulated as a traveling salesman problem (TSP) and solved using a greedy method developed.
by Imada et al. [18], resulting in a content-dependent frame order. The parameters for the final transformation are transmitted to the receiver along with the HEVC-encoded bit-stream. The experimental results show that this method achieves better rate-distortion performance than other HEVC-based LF coding methods [12–14,18].

Reflecting the recent explosive prevalence of deep neural networks, learning-based video coding methods [19–22] have attracted much interest. However, these methods are still struggling to outperform the sophisticated video codecs. For example, although the method developed by Habibian et al. [22] has performance comparable to that of HEVC, their anchor, referred to as “HEVC”, was actually the FFmpeg with the default configuration, which has a performance substantially inferior to that of the well-tuned HEVC reference software, HM [23], maintained by the Joint Collaborative Team on Video Coding (JCT-VC). Meanwhile, our method can incorporate any HEVC codecs. Thus, our method benefits from both the excellent rate-distortion performance of well-designed HEVC codec and the data-driven optimization brought by neural networks; in our method, we use a highly sophisticated video codec for encoding the basis images, and neural networks for optimizing the transformations between the LF images and basis images.

2. Proposed method

2.1 Overview

An overview of our method is shown in Fig. 1. The LF to be compressed, consisting of $M$ multi-view images, is denoted as $\mathcal{L} := \{L_m(u,v) | m = 0, \ldots, M-1\}$, where $m$ is the viewpoint index, and $(u,v)$ specifies a pixel. Instead of directly encoding $\mathcal{L}$ using an HEVC codec, two steps are performed in advance to achieve better rate-distortion performance. First, $\mathcal{L}$ is transformed into a set of smaller number of images, denoted as $\mathcal{B} := \{B_n(u,v) | n = 0, \ldots, N-1\}$, where $N < M$. This reduces the intrinsic redundancy among the images. We call $B_n(u,v)$ a basis image. We approximately span the target LF with the set of basis images ($\mathcal{B}$). The basis images in $\mathcal{B}$ are then reordered to obtain a sequence of frames, described as $\mathcal{F} := \{F_n(u,v) | n = 0, \ldots, N-1\}$, which is then compressed using the HEVC codec. The frame order is optimized to boost the rate-distortion performance of the codec. In the decoding process, a set of decoded frames described as $\hat{\mathcal{F}} := \{\hat{F}_n(u,v) | n = 0, \ldots, N-1\}$ is first obtained, from which the target LF, denoted as $\hat{\mathcal{L}} := \{\hat{L}_m(u,v) | m = 0, \ldots, M-1\}$, is reconstructed.

The optimal LF coding is achieved by minimizing the squared error ($\|\mathcal{L} - \hat{\mathcal{L}}\|^2$) under a fixed bit-rate. This can be rewritten as

$$\min \|\mathcal{L} - \Phi_{\mathcal{F} \rightarrow \hat{\mathcal{F}}} \circ \Phi_{\hat{\mathcal{F}} \rightarrow \hat{\mathcal{L}}} \circ \Phi_{\mathcal{B} \rightarrow \mathcal{F}} \circ \Phi_{\mathcal{L} \rightarrow \mathcal{B}}(\mathcal{L})\|^2,$$

(1)

Fig. 1 Overview of our method. Input LF images $\mathcal{L}$ are first transformed into basis images $\mathcal{B}$, which in turn are reordered to produce a sequence of frames $\mathcal{F}$ and encoded using HEVC. The decoded frames $\hat{\mathcal{F}}$ are inversely transformed to reconstruct the LF images $\hat{\mathcal{L}}$. 

Paper » HEVC-based Light-field Coding using Basis Images and Frame Reordering
Fig. 2 Example of mapping operators for Bikes LF ((M, N) = (193, 128)).

where the transformation processes are denoted as mapping operators; e.g., the transformation from $\mathcal{L}$ to $\mathcal{B}$ is described as $\Phi_{\mathcal{L} \rightarrow \mathcal{B}}$. However, it is difficult to optimize all four mapping operators simultaneously, so they are optimized one by one. The optimization is conducted for each LF, so different mapping parameters are obtained for different LFs. Therefore, the parameters for the final mapping operator, $\Phi_{\hat{\mathcal{F}} \rightarrow \hat{\mathcal{L}}}$, are transmitted to the receiver along with the HEVC-encoded bit-stream.

### 2.2 Mapping operators

The first mapping operator, $\Phi_{\mathcal{L} \rightarrow \mathcal{B}}$, is defined as

$$B_n(u, v) = \sum_m w_{n,m} L_m(u, v). \quad (2)$$

This means that a basis image is synthesized as a weighted average over the input LF images. By imposing the constraints $\sum_m w_{n,m} = 1$ and $w_{n,m} \geq 0$, we ensure that the basis images look like the original LF images, with constant brightness across $n$. We also impose a constraint that the first basis image equals the central viewpoint image of the LF, based on our empirical finding that it works well. The weight values $(w_{n,m})$ are optimized so that the information in $\mathcal{L}$ is best preserved in $\mathcal{B}$. This optimization is done using a neural network, as described in Section 2.3. Note that the model (Eq. (2)) is similar to the one used by Inagaki et al. for compressive LF acquisition using a coded aperture camera [24], which inspired this study.

Mapping $\Phi_{\mathcal{B} \rightarrow \mathcal{F}}$ is aimed at producing a temporally smooth sequence that is suitable for the video codec. Since only the frame order in $\mathcal{B}$ is changed to produce $\mathcal{F}$, no information is lost. More specifically, the frame order is optimized to minimize the length, which is defined as

$$\text{length} = \sum_{n=1}^{N-1} \text{dist}(F_n(u, v), F_{n-1}(u, v)), \quad (3)$$

where “dist” is a distance measure between two adjacent frames. This is equivalent to the traveling salesman problem, in which the problem is to determine the shortest route for visiting all cities (in our case, frames). Following our previous work [18], we use

$$\text{dist}(F_n(u, v), F_{n-1}(u, v)) = \sum_{u,v} |F_n(u, v) - F_{n-1}(u, v)| \quad (4)$$

and the G+2 method of Imaeda et al. [18] to obtain the best possible order.

Mapping $\Phi_{\hat{\mathcal{F}} \rightarrow \hat{\mathcal{L}}}$ is performed by the HEVC codec, which is regarded as being sufficiently optimized. We consider $\bar{\mathcal{F}}$ to be a temporal sequence and simply apply HEVC reference software with a random access configuration to obtain a compressed bit-stream. From this bit-stream, we obtain a sequence of decoded frames, $\hat{\mathcal{F}}$.

Mapping operator $\Phi_{\hat{\mathcal{F}} \rightarrow \hat{\mathcal{L}}}$ is defined as

$$\hat{L}_m(u, v) = \sum_n \hat{w}_{m,n} \hat{F}_n(u, v). \quad (5)$$

This mapping corresponds to the inverse of composite mapping $\Phi_{\mathcal{B} \rightarrow \mathcal{F}} \circ \Phi_{\mathcal{L} \rightarrow \mathcal{B}}$, in which the inverse reordering and reconstruction of the LF images are performed in a single step. The parameters $\hat{w}_{m,n}$ are optimized so that the reconstructed LF ($\hat{\mathcal{L}}$) best approximates the
original one ($L$). This optimization is also conducted using a neural network and is described in Section 2.3. Constraint $w_{m,n} \geq 0$ is imposed to make $\tilde{w}_{m,n}$ sparse. All values for $\tilde{w}_{m,n}$ are transmitted to the receiver side.

The mapping operators obtained for the Bikes LF ($M = 193$) with $N = 128$ and QP = 22 are shown in Fig. 2. In plots (a), (c), and (d), the weights for $\Phi_{L \rightarrow B}$ ($w_{n,m}$ as a $N \times M$ matrix), the reordering pattern for $\Phi_{B \rightarrow F}$ (as a $N \times N$ permutation matrix), and the weights for $\Phi_{F \rightarrow L}$ ($\tilde{w}_{m,n}$ as a $M \times N$ matrix) are visualized. A color map (b) shows the contribution of each input viewpoint to the basis images, where each square corresponds to each viewpoint, and the value (also color-coded for visualization) in each square is obtained as

$$w_m = \sum_n w_{m,n}.$$  (6)

From this contribution map, we can see the extent of dependency the set of the basis images jointly has on each input viewpoint.

### 2.3 Implementation using neural network

The first and final mapping operators, $\Phi_{L \rightarrow B}$ and $\Phi_{F \rightarrow L}$, are implemented using neural networks.

The first mapping $\Phi_{L \rightarrow B}$ is optimized on the basis of principal component analysis, which is formulated as

$$\arg \min_{\Phi_{L \rightarrow B}} \| L - \Phi_{B \rightarrow \tilde{L}} \circ \Phi_{L \rightarrow B}(L) \|_2^2,$$  (7)

where $\Phi_{B \rightarrow \tilde{L}}$ is a pseudo-inverse operator given as

$$\tilde{L}_m(u,v) = \sum_n w_{m,n} B_n(u,v).$$  (8)

By combining this with Eq. (2), we can describe composite mapping $\Phi_{B \rightarrow F} \circ \Phi_{L \rightarrow B}$ as

$$\tilde{L}_m(u,v) = \sum_n w^*_{m,n} \sum_m w_{m,n} L_m(u,v).$$  (9)

Focusing on a specific pixel position $((u,v))$, we see that this calculation is implemented as a fully connected neural network with three layers, as shown in the top-left diagram in Fig. 1. The input, intermediate, and output layers in this network have $M$, $N$, and $M$ nodes, respectively. The input and output are vectors with $M$ elements corresponding to $\{L_m(u,v)\}_{m=0,\ldots,M-1}$ and $\{\hat{L}_m(u,v)\}_{m=0,\ldots,M-1}$, respectively, for a specific $(u,v)$. The intermediate layer corresponds to $B_n(u,v)$ $n=0,\ldots,N-1$ for the same $(u,v)$. The weight values between the input and intermediate layers correspond to $w_{n,m}$, which are optimized jointly with $w^*_{m,n}$ in the training stage. The output from the network is supervised by the original LF images, enforcing that $\tilde{L}_m \simeq L_m$ for all $m$, in a least-squares sense. We use all the pixels in the LF as the training dataset, which means that the obtained weight values ($w_{n,m}$) are optimized for all the pixels but only for the target LF.

In Table 1, we present the quality of a reconstructed LF ($\hat{L}$) to show how much information is preserved in the basis images without the HEVC coding. We used the same four LFs as those used in Section 3 ($M = 193$). We set $N$ to 193, 128, 64, 32. Here, the peak signal-to-ratio (PSNR) is calculated using the mean squared error of all 193 images. We can see the reconstruction quality depends on the number of basis images $N$; Generally, $N$ should be larger for higher reconstruction quality.

The final mapping ($\Phi_{F \rightarrow L}$) is also optimized:

$$\arg \min_{\Phi_{F \rightarrow L}} \| L - \Phi_{F \rightarrow L}(\hat{F}) \|_2^2.$$  (10)

In a manner similar to the one described above, Eq. (5) is implemented as a fully connected neural network for individual pixels, as shown in the top-right diagram in Fig. 1. The network has two layers, the input and output layers, which have $N$ and $M$ nodes, respectively. The input consists of the decoded frames ($\{\hat{F}_n(u,v)| m = 0, \ldots, N-1\}$) for a specific $(u,v)$, and the output consists of the reconstructed LF ($\{\hat{L}_m(u,v)| m = 0, \ldots, M-1\}$) for the same pixel. The weights between the layers correspond to $\tilde{w}_{m,n}$. The output from the network is supervised by the original LF images, enforcing that $\hat{L}_m \simeq L_m$ for all $m$ in a least-squares sense. We use all the pixels in the target LF as the training dataset. The training is conducted for each configuration (e.g., the quantization parameter (QP) value) of the HEVC codec because the configuration greatly affects the decoded frames ($\hat{F}$), which are the network input. All the parameters ($\tilde{w}_{m,n}$) are optimized on the sender side and transmitted to the receiver side along with the HEVC-encoded bit-stream.

The neural networks were implemented using Keras version 2.2.4 and the Adam Optimizer. The size of mini-batch was set to 16,384. The number of epochs was set to 3000 and 500 for $\Phi_{L \rightarrow F}$ and $\Phi_{F \rightarrow L}$, respectively.

<table>
<thead>
<tr>
<th>$N$</th>
<th>Bikes</th>
<th>Friends</th>
<th>Flowers</th>
<th>Stone Pillars</th>
<th>Outside</th>
</tr>
</thead>
<tbody>
<tr>
<td>193</td>
<td>51.74</td>
<td>48.43</td>
<td>50.84</td>
<td>45.16</td>
<td></td>
</tr>
<tr>
<td>128</td>
<td>45.16</td>
<td>47.69</td>
<td>46.49</td>
<td>47.65</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>39.56</td>
<td>43.65</td>
<td>41.40</td>
<td>42.53</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>35.82</td>
<td>40.58</td>
<td>37.67</td>
<td>38.83</td>
<td></td>
</tr>
</tbody>
</table>
3. Evaluation

We used grayscale LFs from the EPFL light-field data set [25], following the guidelines of the ICME 2016 Grand Challenge [26]. Each LF had $15 \times 15$ images with a resolution of $625 \times 434$ pixels. The viewpoints near the corners were removed because they were noisy and distorted, so the number of viewpoints used ($M$) was 193. HEVC reference software (version 16.2) [23] was used to encode and decode the intermediate video in YUV400 format. Random access was used for the inter-frame prediction structure, resulting in excellent rate-distortion performance. The PSNR averaged over all 193 images was used as the quality metric. The bit-rate (bits per pixel: BPP) was calculated using

$$\text{BPP} = \frac{R_e + R_w}{625 \times 434 \times 193},$$

(11)

where $R_e$ and $R_w$ represent the size (in bits) for the HEVC bit-stream and the weight parameters $w_{m,n}$, respectively. The weight parameters were rounded to 16 bit and compressed in LZMA format. Our experimental environment is summarized in Table 2.

The computation time for encoding an LF (we used Bikes LF) with $N = 193, 128, 64, 32$ and $\text{QP} = 22$ is shown in Table 3. Table 4 shows the breakdown between $R_e$ and $R_w$ for various configurations on Bikes LF. We can see that $R_w$ has only a limited ratio in the total bit length.

The performance of our method on four LFs is shown in Fig. 3(a). We tested different values for $N$ (32, 64, 128, and 193). In terms of the overall performance, $N = 128$ is the best choice among them. However, as shown in the close-ups, using $N = 32$ or 64 led to better rate-distortion performance in the low bit-rate. We also presented several contribution maps for the same LFs in Fig. 3(b)–(d). From these maps, we can see that depending on the target LF and the number of basis images, the extent of contribution from each input viewpoint is significantly different.

We also present two ablation studies. First, we nullified the frame reordering ($\Phi_{\Delta \rightarrow \gamma}$ in Fig. 1) and show some results in Fig. 4. From these results, we can see that the frame reordering contributes to the rate-distortion performance, in particular for a large value of $N$. Second, we removed the process of synthesizing basis images ($\Phi_{\Delta \rightarrow 2}$ in Fig. 1). In the ablated cases, we randomly or quasi-evenly selected the input views and used them as the basis images. Shown on the top and middle of Fig. 5 are examples of randomly selected views and quasi-evenly spaced views with different $N$ (32, 64, and 128). Here, each square corresponds to the location of each input viewpoint, and the number and color of each square indicate the frame order in which the selected views are aligned in a temporal sequence (the frame order was optimized using the same method as ours). The final mapping operator $\Phi_{\Delta \rightarrow \hat{L}}$ was optimized for each case in the same manner as ours. The rate-distortion performance is shown in the graphs at the bottom of Fig. 5, where we performed random selection for three times. We can see that our scheme of synthesizing basis images lead to better performance than random selection, in particular with a small number of $N$. Using quasi-evenly spaced viewpoints resulted in almost the same performance as our scheme for $N = 128$ and 64. However, the advantage of our scheme over the quasi-even one became obvious for $N = 32$.

We also compared our method against other HEVC-based methods. We tested four representative frame orders (Raster [12], Circular [12,13], LF-CAE [17] and LF-TSP (G+2) [18]), as shown in matrices (a)–(d) in Fig. 6. Note that the frame orders obtained by LF-CAE and LF-TSP (G+2) were different for different LFs. Following the original implementation [17], LF-CAE was implemented with Low delay B for inter-frame predic-
Fig. 3 Rate-Distortion performance and contribution maps of our method on four different LFs (from left to right, Bikes, Friends, Flowers, and Stone Pillars Outside)

(a) (top) Overall rate-distortion performance (bottom) close-ups for lower bit-rate.

(b) Contribution maps ($N = 128$)
(c) Contribution maps ($N = 64$)
(d) Contribution maps ($N = 32$)

4. Conclusion

We proposed an HEVC-based light-field coding method using basis images and frame reordering. Our method first condenses the original LF into a set of basis images and then reorders them to create a temporally smooth video sequence that is HEVC codec friendly.
Experimental results show that the design choices of our method were effective and our method has promising performance compared to previous works.

Future work will take several directions. First, we will apply our method to light-field datasets other than ones taken using Lytro cameras, which may result in a different setting for the number of basis images. Instead of the current video coding standard (HEVC), the upcoming video coding standard called Versatile Video Coding (VVC) [28] can also be combined with our method. We also intend to investigate other coding methods including 3D HEVC [7, 10], and learned video coding methods [19–22] to clarify the strength of each method and to evaluate the potential synergy with our method. Finally, we hypothesize that generating basis images is also effective for other applications such as coding of high-speed videos.

References
2) S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen: “The lumigraph”, Proceedings of the 23rd annual conference on Com-


Kota Imaeda received a B.E. degree in electrical engineering from Nagoya University, Nagoya, Japan, in 2019, where he is currently working toward a graduate degree in electrical engineering and computer science. His research project includes light field compression.

Keita Takahashi received B.E., M.S., and Ph.D. degrees in information and communication engineering from the University of Tokyo, Tokyo, Japan, in 2001, 2003, and 2006. He was a Project Assistant Professor at the University of Tokyo from 2006 to 2011 and was an Assistant Professor at the University of Electro-Communications from 2011 to 2013. He is currently an Associate Professor at the Graduate School of Engineering, Nagoya University, Nagoya, Japan. His research interests include computational photography, image-based rendering, and 3-D displays.

Toshiaki Fujii received B.E., M.E., and Dr.E. degrees in electrical engineering from the University of Tokyo, Tokyo, Japan, in 1990, 1992, and 1995. In 1995, he joined the Graduate School of Engineering, Nagoya University, where currently he is a Professor. From 2008 to 2010, he was with the Graduate School of Science and Engineering, Tokyo Institute of Technology. His current research interests include multidimensional signal processing, multi-camera systems, multi-view video coding and transmission, free-viewpoint television, and their applications.

Yukihiro Bandoh received the B.E., M.E., and Ph.D. degrees from Kyushu University, Japan, in 1996, 1998 and 2002, respectively. He granted JSPS Research Fellowship for Young Scientists from 2000 to 2002. In 2002, he joined Nippon Telegraph and Telephone (NTT) Corporation, where he has been engaged in research on efficient video coding for high realistic communication. He is currently a Distinguished Engineer at NTT Media intelligence Laboratories. His current research interests include efficient video coding and sparse modeling.

Seishi Takamura received B.E., M.E., and Ph.D. degrees in electrical engineering from the University of Tokyo, Tokyo, Japan, in 1991, 1993, and 1996. In 1996, he joined NTT Human Interface Laboratories, NTT Corporation, where currently he is a Senior Distinguished Engineer. From 2001 to 2006, he was a visiting scientist at Stanford University, CA, USA. His current research interests include efficient video coding and multi-modal signal information processing.

Kota Imaeda

Keita Takahashi

Toshiaki Fujii

Yukihiro Bandoh

Seishi Takamura

Hideaki Kimata received the B.E. and M.E. degrees in applied physics, and the Ph.D. degree in electrical engineering respectively from Nagoya University, Nagoya, Japan, in 1993, 1995, and 2006. He joined Nippon Telegraph and Telephone Corporation (NTT) in 1995, and has been involved in the research and development of video coding, realistic communication, computer vision, and video recognition based on machine learning (deep learning). He is currently a Senior Research Engineer and also the Supervisor at NTT Media Intelligence Laboratories. He is a Chair of Technical Committee on Image Engineering of the Institute of Electronics, Information and Communication Engineers of Japan.