Quadtree-Based Coding Framework for High-Density Camera Array-Based Light Field Image

Li Li, *Member, IEEE*, Zhu Li, *Senior Member, IEEE*, Bin Li, *Member, IEEE*, Dong Liu, *Senior Member, IEEE*, and Houqiang Li, *Senior Member, IEEE*

***Abstract*— The size of a high-density-camera-array (HDCA)-based light field image (LFI) is usually very large, containing hundreds of high-resolution views. Therefore, there is an urgent need to efficiently compress it. Currently, no compression algorithms, specially, for the HDCA-based LFI have been designed. In this paper, we propose an algorithm based on a quadtree-based 2D hierarchical coding framework for the HDCA-based LFI data compression. The proposed framework has the following contributions. First, we organize the views of the HDCA-based LFI into a quadtree-based coding structure. Under this structure, all of the views are divided into four quadrants at the first level. Each quadrant is further sub-divided into four quadrants at each subsequent level. The process continues until the desired depth is reached. This quadtree-based coding structure can make full use of the strong inter-view correlations to improve the coding efficiency. In addition, the proposed quadtree-based structure can be easily extended to a general 2D hierarchical structure with variable group of pictures (GOP) sizes to adapt to the reference frame buffer constraint. Second, we try to improve the performance of the 2D hierarchical coding framework using the distance-based criteria for both the reference frame selection and motion vector scaling. Third, a one-pass optimal bit allocation scheme is proposed to further optimize the performance by taking the quality dependencies among various views into consideration. The proposed framework is implemented in the newest video coding standard, high efficiency video coding (HEVC). The experimental results show that the proposed quadtree-based 2D hierarchical coding framework can achieve an average of over 25% bitrate saving compared with the 1D hierarchical coding structure.**

***Index Terms*— High density camera array (HDCA), high efficiency video coding (HEVC), light field image (LFI) compression, quadtree-based coding structure, 2D hierarchical coding structure.**

Manuscript received December 19, 2018; revised April 5, 2019 and May 5, 2019; accepted June 5, 2019. Date of publication June 21, 2019; date of current version August 4, 2020. This work was supported in part by the Phase I IUCRC University of Missouri-Kansas City: Center for Big Learning under Award Number 1747751. This article was recommended by Associate Editor H. Schwarz. *(Corresponding author: Zhu Li.)*

L. Li is with the Department of Computer Science and Electrical Engineering, University of Missouri–Kansas City, Kansas City, MO 64110 USA, and also with the CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, University of Science and Technology of China, Hefei 230027, China (e-mail: lil1@umkc.edu).

Z. Li is with the Department of Computer Science and Electrical Engineering, University of Missouri–Kansas City, Kansas City, MO 64110 USA (e-mail: lizhu@umkc.edu).

B. Li is with Microsoft Asia, Beijing 100010, China (e-mail: libin@microsoft.com).

D. Liu and H. Li are with the CAS Key Laboratory of Technology in Geo-Spatial Information Processing and Application System, University of Science and Technology of China, Hefei 230027, China (e-mail: dongeliu@ustc.edu.cn; lihq@ustc.edu.cn).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2019.2924313

I. Introduction

THE light-field image (LFI) [1], also known as the plenoptic image, contains the information not only about the intensity of the light in a scene but also the light rays in space. Therefore, the LFI is a very promising solution for many areas of 3D research such as 3D television [2] and medical imaging [3]. There are two common types of LFI data capture methods, lenslet-based LFI, and high-density-camera-array-based (HDCA-based) LFI. As their names imply, the lenslet-based LFI operates by placing an array of micro-lenses in front of a conventional image sensor, while the HDCA-based LFI uses multiple camera arrays. The lenslet-based LFI is simple and cheap, but it sacrifices the spatial resolution to improve the view resolution. The HDCA-based LFI is more expensive, but guarantees both high spatial and view resolutions.

Due to the high spatial and view resolutions, the need for compressing the HDCA-based LFI is much higher. Currently, the ISO/IEC SC29WG11 JPEG Pleno standardization group (JPEG Pleno) [4], [5] is making efforts to develop a standard on LFI compression. For a typical HDCA-based LFI in the JPEG Pleno dataset from Fraunhofer IIS [6], the view resolution is 101×21, the spatial resolution of each view is 3840×1920, and the bit depth of each pixel is 10. Therefore, the size of such a LFI can be as large as $100GBytes$. Fig. 1 shows four typical views of a HDCA-based LFI. We can obviously see from Fig. 1 that different views are very similar to each other, which means that there are lots of redundancies among various views. In this paper, we will try to take full advantage of these correlations to compress the HDCA-based LFI efficiently.

The current researches on LFI compression can be roughly divided into two groups according to different types of LFIs: the lenslet-based LFI compression and the HDCA-based LFI compression. The lenslet-based LFI compression methods can be further divided into two groups: the self-similarity-based method [7], [8] and the pseudo-sequence-based method [9], [10]. The self-similarity-based

Fig. 1. Four corner views of a 21 × 21 camera array cropped from the HDCA-based LFI Set2.

framework tries to compress the LFI using the image compression framework, while the pseudo-sequence-based method decomposes the LFI into multiple views and uses the video compression framework. Since the HDCA-based LFI has multiple views, the pseudo-sequence based method seems to be a natural fit. Nevertheless, the state-of-the-art pseudo-sequence-based method [10] is unsuitable for the HDCA-based LFI for the following reasons. First, since the spatial resolution of various views of a lenslet-based LFI is very small, the previous works usually organize all the views as a group of pictures (GOP). However, this method is unsuitable for the HDCA-based LFI compression due to the reference frame buffer constraints. As the spatial resolution of the HDCA-based LFI increased, continued storage of the previously coded frames in the reference frame buffer becomes infeasible. Second, the bit allocation algorithm provided in [10] can lead to better R-D performance, however, it introduces multi-pass encoding. The multi-pass encoding problem is much more serious for the HDCA-based LFI since the complexity of the one-pass encoding is already quite large. We believe that the multi-pass encoding problem can be managed, since many views of the HDCA-based LFI are similar.

The most typical work for the HDCA-based LFI compression [11] organizes all the views into a multi-view sequence and compresses them using the multi-view HEVC (MV-HEVC). However, as the MV-HEVC is designed for the multi-view sequences instead of the HDCA-based LFI, it presents the following disadvantages when it is applied to the LFI compression.

- The MV-HEVC is not flexible enough to fully exploit the correlations among various views in the HDCA-based LFI. Both the reference structure and the encoding structure suffer in various ways. For the MV-HEVC reference structure, the current frame can only use the frames with the same POC or the same view index as references. The HDCA based LFI will suffer from this approach since the diagonal high-quality references are quite similar and can lead to significant gains in coding efficiency. For the MV-HEVC encoding structure, the number of views is restricted by the number of vertical cameras in the camera arrays. As the number of vertical camera increases, the number of high resolution views in the vertical direction increases. This increases the size of the reference buffers, which will have a natural upper limit due to the processing capabilities.
- In MV-HEVC, the view correlations are considered less important than the inter correlations. For example, different views with the same POC may lead to inaccurate motion predictions, yet, the frame-to-frame predictions are accurate. In the HDCA-based LFI, this issue should be addressed by considering both horizontal and vertical views equally.
- The bit allocation algorithm in the MV-HEVC is unsuitable for the HDCA-based LFI. The very high correlations among various view in HDCA-based LFI cannot be captured by the current bit allocation algorithms designed for MV-HEVC.

Therefore, in this paper, we propose a quadtree-based 2D hierarchical coding framework to compress the HDCA-based LFI. We implement a modified High Efficiency Video Coding (HEVC) [12] encoder and decoder for the proposed framework. The proposed framework mainly has the following contributions.

- We organize all the views of the HDCA-based LFI into a quadtree-based coding structure. Under this structure, all of the views are divided into four quadrants at the first level. Each quadrant is further sub-divided into four quadrants at each subsequent level. This process continues until the desired depth is reached. The quadtree-based coding structure is able to fully exploit the correlations among various views to improve the coding efficiency. In addition, it can be easily extended to a general 2D hierarchical coding structure with variable GOP sizes. The GOP size is selected to accommodate a reference buffer size that will achieve the best performance.
- We improve the 2D hierarchical coding framework using the distance-based criteria for both the reference frame selection and motion vector (MV) scaling. The distance-based reference frame selection can put the frames with higher correlations to the current frame in the earlier positions of the reference picture lists to save the bits for the header information. The distance-based MV scaling process can estimate the MV between various views very accurately due to the regular structure of the HDCA-based LFI.
- We further optimize the 2D hierarchical coding framework using a one-pass optimal bit allocation algorithm. Through utilizing the correlations among various views, quality dependencies can be predicted from previous GOP, and thus multi-pass encoding can be avoided. This scheme can improve performance without an increase in complexity.

Note that we have proposed the distance-based reference frame selection and MV scaling in our previous work for the lenslet-based LFI [10]. Since the motions between various views in HDCA-based LFI are more regular than for the lenslet-based LFI, using distance-criteria can lead to better performance. Therefore, we propose using distance-based criteria to further improve the 2D hierarchical coding structure.

The rest of this paper is organized as follows. In Section II, we will review some related works on LFI compression. Then we will introduce the proposed HDCA-based LFI compression framework including the quadtree-based coding structure, the distance based reference management, and the MV scaling in Section III. The one-pass optimal bit allocation algorithm will be introduced in Section IV. We will show the detailed experimental results in Section V to demonstrate the effectiveness of the proposed algorithms. Finally, we will conclude the whole paper in Section VI.

II. Related Work

A. Lenslet-Based LFI Compression

The lenslet-based LFI compression can be divided into two groups: the self-similarity-based method and the pseudo-sequence-based algorithm. As its name implies, the self-similarity-based method incorporates the self-similarity compensated prediction and estimation, which can substantially exploit the correlations in the LFI, into the existing image compression framework. For example, Conti *et al.* [13] first introduced the concept of self-similarity to H.264/Advanced video coding [14] to compress the LFI efficiently. This work was then extended to HEVC to take advantage of flexible partitions [7]. Most recently, the bi-directional self-similarity compensated prediction [15] was introduced to the self-similarity based framework to further improve the coding efficiency. Template matching [16] was also used to find multiple similar blocks to get a combined prediction block to improve the prediction accuracy. In addition to the translational motion model, Monteiro *et al.* [17] proposed to use the high-order geometric transform to predict each block more accurately. Furthermore, the macro-pixel in LFI is not aligned with the standard-based coding structure which may lead to degraded compression performance. Jin *et al.* [18] proposed reshaping the image to align the macro-pixel with the grids of the block-based image compression structure. Other advanced prediction modes have been proposed in [19], which also aim to improve the compression performance.

The pseudo-sequence-based algorithm decomposes the LFI into multiple views and organizes all the views into a pseudo sequence to utilize the efficient inter predictions in the video coding framework. For example, Perra and Assuncao [20] proposed dividing the LFI into multiple tiles and organizing them into a sequence. The correlations between the frames were then exploited using the HEVC inter prediction. Liu *et al.* [9] first proposed decomposing the LFI into multiple views and organizing them into a sequence. The sequence was then encoded using a pre-defined 2D hierarchical coding structure. Li *et al.* [10] further considered the encoding orders, the reference frame management, and the bit allocation to optimize the 2D hierarchical coding structure as much as possible. Instead of the hierarchical coding order, Hariharan *et al.* [21] introduced the circular reordering to generate a pseudo sequence which is more suitable for the low delay coding structure. Zhang *et al.* [22] investigated a sift-based reference frame selection to optimize encoding. Ahmad *et al.* [23] directly treated the LFI as a multiple view sequence and compressed it using MV-HEVC [24].

Rather than applying any of the variations of HEVC directly to the LFI, Jiang *et al.* [25] proposed first computing a homography-based low-rank representation of the LFI. They then coded this representation and corresponding disparity information using HEVC. Since lenslet-based LFIs are dense, methods where they are split into encoding and synthesizing subsets have been proposed. Zhao and Chen [26] proposed encoding half of the views using the video compression standard while synthesizing the other half using a linear approximation method. Chen *et al.* [27] proposed deriving some structural key views from the original LFI and predicting other views using sparse coding methods. Tabus *et al.* [28] proposed compressing some key views and corresponding depth maps, then using these to synthesize additional views in the decoder. Even more compression-based research focused on convex optimization-based bit allocation and adaptive loop filtering have been proposed in [29] and [30].

B. HDCA-Based LFI Compression

Many works have proposed using currently existing compression methods to encode the HDCA-based LFI. Zhu *et al.* [31] proposed computing the correlations between the views of large camera arrays using a distributed compression scheme. Some views were acquired using JPEG [32] and the other views were compressed using Wyner-Ziv encoders [33]. Ahmad *et al.* [11] proposed treating the views with different horizontal axis as different frames, and the views with the same horizontal axis but different vertical axis as multiple views of one frame. All the views were organized into a multi-view sequence and encoded using MV-HEVC [24]. The MV-HEVC-based method achieves much better compression performance compared with HEVC. However, it has not fully considered the characteristics of HDCA-based LFI as we have analyzed in Section I. Therefore, it is unable to fully exploit the correlations among various views of the HDCA-based LFI and optimize the compression performance.

Other related works focused on smaller sized data sets. Adhikarla *et al.* [34] proposed a fast and efficient data reduction approach for MCLF display systems. However, this approach directly drops data in the acquisition side of the system, which is unsuitable for the general compression and transmission applications. Cornwell *et al.* [35] introduced a combined global perspective motion and local affine motion model to characterize the complex motions among various views for the small camera array. However, those methods are not well suited for the HDCA-based LFI, for which the motions can be well approximated by the translational motion model.

III. Proposed HDCA-Based LFI Compression Framework

The proposed HDCA-based LFI compression framework can be roughly divided into two parts. We will first describe the quadtree-based coding structure in subsection III-A. Then

Fig. 2. This figure depicts the 21×21 views to be compressed. The views are assigned picture order count 0 to 440 shown in the figure. All the views are first divided into 4 quadrants as indicated by the yellow rectangles. The quadrants are further divided into 16 GOPs as indicated by the blue rectangles.

we will introduce the distance-based reference frame selection and MV scaling in subsection III-B.

A. Quadtree-Based Hierarchical Coding Structure

Taking the 21×21 views derived from the 101×21 dataset as an example, the proposed quadtree-based coding structure is shown in Fig. 2. In Fig. 2, each square image represents a view taken by a camera in the corresponding spatial position. All the views are first split into 4 quadrants as indicated by the yellow rectangles. Each quadrant will be further split into 4 GOPs as shown by the blue rectangles. In the following descriptions, we will consider the 4 zones after the first split as quadrants, and the 16 zones after the second split as GOPs. In this way, all the views are divided into 16 GOPs using split depth 2. At each split depth, we assign the picture order count (POC) to all views GOP by GOP in a counter-clockwise order, as shown in Fig. 2. For example, the first GOP is assigned POCs 0 to 35 and the following GOP is assigned POCs 36 to 65. Within each GOP, the POC is assigned from top left to bottom right with the POC increases one by one in the vertical direction.

In such a quadtree-based coding structure, there are two points to be further clarified. First, the POC here is just a symbol to represent each view instead of the display order in general videos. Therefore, we build a coordinate system for the spatial positions of each view, which will be used for the distance-based reference frame selection and MV scaling algorithms in the next subsection. In the coordinate system, the coordinates of the top left view, the top right view, the bottom left view, and the bottom right view are $(0, 0)$, $(20, 0)$, $(0, 20)$, and $(20, 20)$, respectively. Second, such a coding structure can be easily applied to other HDCA-based datasets with a different number of views by setting different

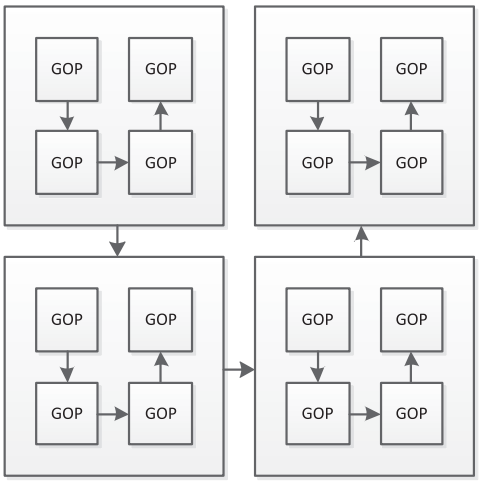


Fig. 3. GOP and quadrant encoding order.

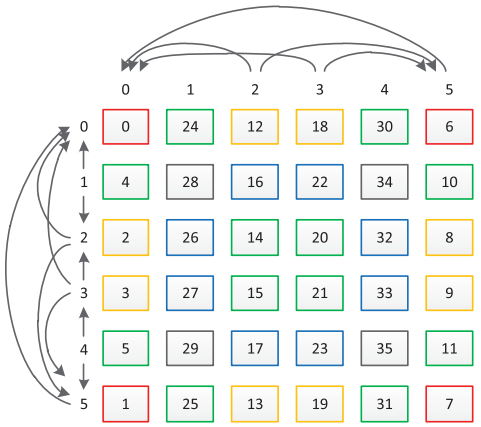


Fig. 4. Encoding order and reference relationship of a typical GOP.

split depths and GOP sizes. The larger camera arrays will have larger split depths and vice versa.

Using the proposed quadtree-based hierarchical coding structure, all the views will be encoded GOP by GOP. In the proposed algorithm, a counter-clockwise order is used in each split depth to encode all the quadrants and GOPs, as shown in Fig. 3. The arrows in the figure reflect exactly the order we used to encode all the GOPs. Compared with the raster scan, the counter-clockwise order has better chance of keeping reference frames that are in close proximity to the current frame in the reference frame buffer. Compared with the zig-zag scan, the counter-clockwise order is more likely to include frames along the horizontal axis, thereby, keeping frames in close proximity in the reference frame buffer.

Within each GOP, the depth-first hierarchical coding order is used to make full use of the reference frame buffer. Taking the first GOP from Fig. 2 as an example, the hierarchical coding orders are shown in Fig. 4. The number in the rectangles means the encoding order of the corresponding view. Note that the column-based encoding order, in which we encode one column after another, is used in our implementation. There are no obvious differences whether the column-based encoding order or the row-based encoding order is used. In both the horizontal and vertical directions, the proposed encoding order is 0, 5, 2, 3, 1, and 4 following the hierarchical coding structure. To be more specific, we will first encode the $0th$ and $5th$ row. Then according to the hierarchical coding structure in the horizontal direction, we will encode the $2nd$ and $3rd$ row followed by the $1st$ and $4th$ row. Within each row, the order of 0, 5, 2, 3, 1, and 4 is also used. The encoding orders of the other GOPs are similar to that of the first GOP. In essence, the other GOPs can be considered as the first GOP without the first column, the first row, the last column, or the last row. The frames are encoded in the same order as the first GOP with the positions of the missing frames ignored.

After determining the encoding orders, we will then decide the reference relationship of all the views. As shown in Fig. 4, all the views are divided into 5 temporal layers according to their positions in each GOP.

- The corner frames with red squares. These frames are the so-called key frames. The key frames and the frames within the spatial positions of the key frames build a GOP. The key frames will not only be referenced by the frames in the current GOP but also by the frames in the other GOPs.
- The frames with yellow squares. These frames are the second most frequently referenced frames. They will be referenced by all the other frames in the current GOP except for the red frames.
- The frames with green squares. These frames are the third most frequently referenced frames. In the $1st$ and $4th$ column, they are in the lowest temporal layer and will be referenced by all the frames in the same column.
- The frames with blue squares. These frames are the least referenced frames. They will only be referenced by the other frames with blue squares and the frames with black squares.
- The frames with black squares. These frames are the non-reference frames.

Note that all the other frames except for the key frames will only be referenced by the current GOP in the proposed hierarchical structure. At the beginning of encoding each GOP, we will re-initialize the reference buffer with only the key frames. For example, when encoding the key frame 116, which is the first frame to be encoded in the $4th$ GOP, we will initialize the reference frame buffer as $(30, 35, 40, 65, 90, 95)$. The key frames 0 and 5 have already been popped out of the reference frame buffer since they have relatively larger distances with all the frames to be encoded.

In the experiments, we set the maximum number of reference frames as 16 to satisfy the reference frame constraint provided by HEVC. When the number of reference frames is full, we will always keep the key frames which may still be used by the current GOP and the other GOPs in the future. The other frames, which are not as important as the key frames, will be popped out of the reference frame buffer with the order of first-in-first-out. Also, in accordance with the encoding order, we also use the vertical direction as the main direction when considering the reference relationship. As can be seen from Fig. 4, we always guarantee the complete hierarchical

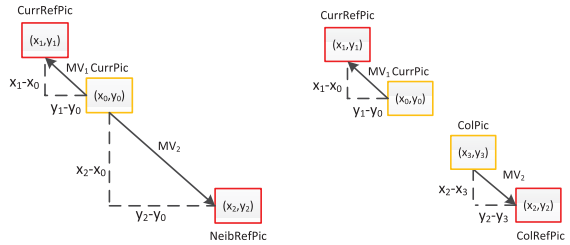
reference relationship in the vertical direction, while we can only guarantee the key frames being used as reference frames in the horizontal direction.

We also would like to emphasize that the proposed quadtree-based hierarchical coding structure can not only be used for the HDCA-based LFI, but also for more general applications. Different from the previous work [10] considering all the views as a GOP, the proposed quadtree-based coding structure with multiple GOPs can be easily extended to the general form of a 2D hierarchical coding structure by giving the number of frames in the horizontal and vertical directions in a GOP. For example, the proposed quadtree-based hierarchical coding structure as shown in Fig. 2 can be considered as a general 2D hierarchical coding structure with 5 frames in both the horizontal and vertical directions in a GOP. In summary, the generalized 2D hierarchical coding structure mainly has the following three key differences compared with the 1D hierarchical coding structure.

- Different from the 1D hierarchical coding structure [36] with fixed GOP size 8 or 16, the general 2D hierarchical coding structure has an alterable GOP size. For example, the sizes of the GOPs in the top left quadrant of the proposed hierarchical coding structure are 35, 30, 25, and 30, respectively. The difference comes from the fact that we will have the overlaps of one-column or one-row views under the 2D hierarchical coding structure between the neighboring GOPs. However, the overlap of the neighboring GOP under the 1D hierarchical coding structure will be only the key frame.
- In the 1D hierarchical coding structure, the POCs of the key frames are always multiples of the GOP size. In the proposed 2D coding structure, the key frames are those whose horizontal and vertical coordinates are a multiple of either the number of frames in each column or row in a GOP, respectively.
- The reference relationships in the 2D hierarchical structure involve two directions, whereas the relationships in a 1D hierarchical structure only involve one direction.

B. Distance-Based Reference Frame Selection and MV Scaling

In the 1D hierarchical coding structure, the POC difference can mostly indicate the similarity between the current frame and the reference frames. Therefore, the POC-based reference frame selection is used in the 1D hierarchical coding structure by putting the reference frames with smaller POC differences compared with the current frame in relatively earlier positions in the reference frame lists. However, in the 2D hierarchical coding structure, the POC differences cannot reflect the spatial distances between neighboring frames at all since POC is just a symbol to represent each frame. For example, the POC difference between the frame 0 and the frame 7 is larger than that between the frame 0 and the frame 5. However, the distance between the frame 0 and the frame 7 is $\sqrt{2}$, which is smaller than the distance between the frame 0 and the frame 5. Therefore, we need to calculate the distances ourselves according to the spatial position of each view.



(a) spatial case (b) temporal case

Fig. 5. Distance-based MV scaling.

To accurately calculate the distances between various views, we have established a coordinate system to derive the spatial coordinates of all the views in Fig. 2. Then the distance d between two views with coordinates (x_1, y_1) and (x_2, y_2) can be calculated using the Euclidean distance

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}. \tag{1}$$

After the distance between two frames is well defined, we then try to construct the two reference picture lists: list0 and list1. We consider the frames with smaller POCs compared with the current frame as the forward frames and the frames will larger POCs as the backward frames. Just like the reference list construction process in the 1D hierarchical structure, the list0 is constructed from the nearest forward frame to the farthest forward frame, from the nearest backward frame to the farthest backward frame, and the list1 is constructed from the nearest backward frame to the farthest backward frame, from the nearest forward frame to the farthest forward frame. For example, frame 14 has the frames (0, 1, 2, 3, 4, 5, 12, 17, 30, 31, 32, 33, 34, 35) in the reference pictures lists. Among them. the forward frames include (0, 1, 2, 3, 4, 5, 12), and the backward frames include (30, 31, 32, 33, 34, 35). We will put the frames with smaller distances with the current frame in the earlier positions of the reference lists. In the experiments, we use 4 reference frames in both list0 and list1. According to the distance-based criteria, the list0 and list1 will be (2, 12, 3, 1) and (32, 17, 33, 31), respectively. It should also be mentioned here that the reference frames in higher temporal layers will not be used as references for the frames in lower temporal layers.

The distance will not only have influences on the reference frame construction but also will influence the MV scaling processes in the merge and advanced motion vector prediction (AMVP) processes. The MV scaling processes are used when the spatial neighboring or temporal co-located blocks are referencing a different frame from the current block. The MV scaling processes can be divided into spatial scaling and temporal scaling. They are used when the spatial neighboring or temporal co-located blocks are referencing a different frame from the current block. Under the quadtree-based coding structure, we should perform the MV scaling based on the distance in x and y directions instead of POC. The detailed processes can be seen from Fig. 5.

In the spatial scaling case, the coordinate of the current block is (x_0, y_0). The coordinates of the reference frames of the current block and the neighboring block are (x_1, y_1) and

(x_2, y_2), respectively. Assume that the MV of the neighboring block is $(MV_{2,x}, MV_{2,y})$, then the MV prediction of the current block $(MV_{1,x}, MV_{1,y})$ can be calculated as

$$MV_{1,x} = \frac{MV_{2,x}}{x_2 - x_0} \cdot (x_1 - x_0), \tag{2}$$

$$MV_{1,y} = \frac{MV_{2,y}}{y_2 - y_0} \cdot (y_1 - y_0). \tag{3}$$

In the temporal scaling case, one more coordinate of the co-located block (x_3, y_3) is involved in the scaling process. Assume that the MV of the co-located block is $(MV_{2,x}, MV_{2,y})$, then the MV prediction of the current block $(MV_{1,x}, MV_{1,y})$ can be calculated as

$$MV_{1,x} = \frac{MV_{2,x}}{x_2 - x_3} \cdot (x_1 - x_0), \tag{4}$$

$$MV_{1,y} = \frac{MV_{2,y}}{y_2 - y_3} \cdot (y_1 - y_0). \tag{5}$$

We have proposed the idea of using the distance-based MV scaling instead of the POC-based MV scaling in our previous works for the lenslet-based LFI to solve such a problem [10]. The same rules are followed here to perform the distance-based MV scaling including the spatial and temporal MV scaling. Note that the distance-based MV scaling becomes a much more important coding tool for the HDCA-based LFI compared with the lenslet-based LFI as will be shown in the experimental results since different views in the HDCA-based LFI are captured by the cameras placed in equal distances.

IV. One-Pass Optimal Bit Allocation

To further boost the performance of the proposed 2D hierarchical coding structure, we also propose a one-pass optimal bit allocation scheme to utilize the correlations among various GOPs. For each GOP, the optimal bit allocation is to minimize the sum of the distortions of all the pictures under the total bitrate constraint. As indicated in the λ-domain rate control and bit allocation algorithms [37], [38], the Lagrangian multiplier λ is in essence the key factor to determine the bitrate R and the distortion D. On the one hand, since λ is the slope of the line tangent to the R-D curve, there is a one-to-one correspondence between R or D and λ. On the other hand, λ can not only determine the residual bitrate, but also determine the non-residual bitrate including MV, coded block flag (cbf) through influencing the mode decision and motion estimation processes. Therefore, we propose to determine a proper λ_j for picture j in each GOP to achieve the optimal bit allocation.

In this way, the optimal bit allocation problem is formulated as

$$\min_{\lambda_j} \sum_{i=1}^{N_S} D_i, \quad s.t. \sum_{i=1}^{N_G} R_i \le R_t, \quad j = 1, 2, ..., N_G, \tag{6}$$

where N_S and N_G are the number of frames in a sequence or a GOP, respectively. Since the key frames in the current GOP may have significant influences on the other GOPs, we need to consider the distortion of the whole sequence when we allocate the bits of all the views in the current GOP. The D_i and R_i are the distortion and bits for frame i, respectively. The constrained optimization problem can be converted to the following unconstrained problem by introducing the Lagrange multiplier λ,

$$\min_{\lambda_j} \sum_{i=1}^{N_S} D_i + \lambda \sum_{i=1}^{N_G} R_i. \tag{7}$$

Then, applying the Lagrangian method and setting the derivative of the total cost with respect to λ_j to 0, the unconstrained problem becomes

$$\frac{\partial \sum_{i=1}^{N_S} D_i}{\partial D_j} \cdot \frac{\partial D_j}{\partial \lambda_j} + \lambda \frac{\partial R_j}{\partial \lambda_j} = 0. \tag{8}$$

Note in (8), we assume that the bitrate of current picture is only determined by the coding parameters of the current picture [39]. Since the Lagrange multiplier for picture j is the slope of the rate distortion curve, we can have $\lambda_j = -\frac{\partial D_j}{\partial R_j}$. In this way, the solution of (8) can be expressed as

$$\lambda_j = \frac{\lambda}{\frac{\partial \sum_{i=1}^{N_S} D_i}{\partial D_j}} = \frac{\lambda}{1 + \frac{\partial \sum_{i=j+1}^{N_S} D_i}{\partial D_j}} \triangleq \frac{\lambda}{1 + \Omega_j}, \tag{9}$$

where Ω_j represents the influence of the distortion of the picture j on the subsequent pictures. Note that the Ω_j will be equal to 0 when picture j is a non-reference picture. Therefore, the λ_j of the non-reference pictures is equal to the sequence-level λ.

Then the only problem is to determine the Ω_j of the other reference pictures to derive the encoding parameter λ_j of each picture. As we can see from Section III, the reference relationship between various frames can be rather complex including uni-directional prediction and bi-directional prediction. We need to consider both, the influence the current picture has on to-be-encoded pictures, and the propagated influence that the to-be-encoded pictures has on other to-be-encoded pictures. These two kinds of influences will be called as direct influence and indirect influence in the following descriptions.

We first consider a simple case of direct influence. Under the rate distortion function in the high bitrate case, the relationship between the reconstruction error D_i and the prediction error D_i^P of a prediction unit (PU) can be expressed as

$$D_i = e^{-bR_i} \cdot D_i^P, \tag{10}$$

where b is the model parameter of the video source. Under the uni-directional prediction case, the D_i^P can be estimated using the distortion of the reference block D_j,

$$D_i^P = \alpha D_j + C, \tag{11}$$

where α is set as 0.94 according to the experience of some previous works [39], [40]. C is a constant determined by the video source. If we combine (10) and (11) together, we can have

$$D_i = e^{-bR_i} \cdot (\alpha D_j + C). \tag{12}$$

Therefore, we can derive the distortion propagation from block j to block i as

$$\frac{\partial D_i}{\partial D_j} = e^{-bR_i} \cdot \alpha \triangleq \gamma_{j,i}. \tag{13}$$

From (10), the $\gamma_{j,i}$ can be calculated as

$$\gamma_{j,i} = \alpha \cdot D_i / D_i^P. \tag{14}$$

The $\gamma_{j,i}$ reflects the influence of one block in frame j on a corresponding block in frame i.

Under the bi-directional prediction case, the problem is similar but more complex. The D_i^P in (11) is not only related to a block in the forward frame D_j, but also related to a block in the backward frame D_k.

$$D_i^P = \beta \cdot (D_j^P + D_k^P) = \beta \cdot (\alpha D_j + \alpha D_k + C), \tag{15}$$

where the β is approximated as 0.3 according to the previous work on the 1D hierarchical coding structure [39]. In this way, the $\gamma_{j,i}$ in bi-directional case can be calculated as

$$\gamma_{j,i} = \beta \cdot \alpha \cdot D_i / D_i^P. \tag{16}$$

After deriving the distortion propagation of one block to another block, we then accumulate the weighted sum of $\gamma_{j,i}$ of all the blocks in frame i using frame j as reference to obtain $\Gamma_{j,i}$, the influence of frame j on frame i.

$$\Gamma_{j,i} = \sum_{m=1}^{N_{j,i}} w_m \cdot \gamma_{j,i}^m, \tag{17}$$

where the weight w_m is calculated by the area of the mth PU divided by the size of the frame. The $N_{j,i}$ is the number of blocks in frame i using frame j as reference. Since frame i has not been encoded yet, we use the co-located frame in the previous GOP to estimate the $\Gamma_{j,i}$. Note that the prediction can be quite accurate since the reference relationships and the motions between neighboring GOPs are almost the same.

Then we will consider the indirect influence combined with the direct influence using an iterative method. Since a frame cannot have influences on the frames coded before it, we try to calculate the indirect influence using the reverse-encoding order. The indirect influence combined with the direct influence $\omega_{j,i}$ can be calculated as

$$\omega_{j,i} = \Gamma_{j,i} \cdot (1 + \Omega_i). \tag{18}$$

Since we calculate the influence based on the reverse-encoding order, the Ω_i is already available before calculating the $\omega_{j,i}$.

Finally, the influence of the frame j on the whole sequence can be calculated as

$$\Omega_j = \sum_{i=j+1}^{N-1} \omega_{j,i}. \tag{19}$$

After determining the Ω_j, we can calculate the λ_j of each picture according to (9). The quantization parameter (QP) will be calculated as in [41], by

$$QP_j = 4.3281 \cdot log(\lambda_j) + 14.4329. \tag{20}$$

Then we can finish the encoding process based on the λ_j and QP_j. Note that no previous information can be used when encoding the first GOP. The initial QPs of the different temporal layers are set according to our experience as shown in Table I. The QP_b in the table is the QP of the intra frame. A similar bit allocation algorithm can be extended to the conventional 1D hierarchical coding structure or the MV-HEVC coding structure.

TABLE I
INITIAL QP SETTINGS OF DIFFERENT TEMPORAL LAYERS

temporal layer	QP
0	QP_b + 3
1	QP_b + 5
2	QP_b + 6
3	QP_b + 7
4	QP_b + 8

TABLE II
CONFIGURATIONS OF THE 1D HIERARCHICAL CODING STRUCTURE

Notation	Value
GOP size	8
Frame encoding order	$0, 8, 4, 2, 1, 3, 6, 5, 7$
QP setting	$Level_f + QP_b + 1$

TABLE III
CONFIGURATIONS OF THE MV-HEVC CODING STRUCTURE

Notation	Value
Number of layers	17
GOP size	16
View encoding order	$8, 0, 4, 2, 1, 3, 6, 5, 7, 16, 12, 10, 9, 11, 14, 13, 15$
Frame encoding order	$0, 16, 8, 4, 2, 1, 3, 6, 5, 7, 12, 10, 9, 11, 14, 13, 15$
Interview references	3
Temporal references	Same as 1-D temporal structure with GOP 16
QP setting	$Level_v + Level_f + QP_b + 1$

V. EXPERIMENTAL RESULTS

A. Simulation Setup

We implement the proposed framework in HEVC reference software HM-16.7 [42] to compare with the 1D hierarchical coding structure to demonstrate the effectiveness of the proposed algorithm. We employ the default HEVC random access coding structure with GOP size 8 as the 1D hierarchical coding structure [36]. The detailed configurations are shown in Table II. The $Level_f$ means the hierarchical level. In addition, we also compare the proposed framework with the state-of-the-art HDCA-based LFI compression algorithm using MV-HEVC [11]. Since the original HDCA-based LFI compression algorithm in [11] used the low delay coding structure, which cannot fully utilize the correlations among various views, we just follow the interview coding structure in [11] but replace the frame coding structure with the random access coding structure for a more direct comparison. The detailed configurations including GOP and reference frame structures are shown in Table III. In Table III, the $Level_v$ and $Level_f$ mean the hierarchical level in the view encoding order and frame encoding order, respectively. The higher the level of the current view or frame, the larger the QP will be. We will first present the experimental results on the overall framework. Then the benefits of the proposed

framework will be carefully analyzed from the following three aspects: the proposed quadtree-based reference structure, the distance-based MV scaling, and the one-pass optimal bit allocation.

When measuring the performance of the proposed quadtree-based 2D hierarchical coding structure and the distance-based MV scaling, the QPs of various temporal layers of all the GOPs are set as shown in Table I. When measuring the performance of the optimal bit allocation where the QPs of various LFIs may be different according to the corresponding content, we use the following method to guarantee that the optimization targets of the proposed framework and the 2D hierarchical coding structure without the optimal bit allocation are the same. We first generate the anchor using the 2D hierarchical coding structure without the optimal bit allocation using QPs 22, 27, 32, and 37. Then the λ of the non-reference pictures is recorded and used as the sequence-level λ of the proposed framework. The λs of the other pictures will then be calculated using (9).

In our experiments, both the Y-PSNR and YUV-PSNR are used as the objective quality measurements. Since the test sequences are in 4:2:0 YUV format, the YUV-PSNR shown in this manuscript is calculated as

$$MSE_{YUV} = \frac{4 \cdot MSE_Y + MSE_U + MSE_V}{6}, \quad (21)$$

$$PSNR_{YUV} = 10 \log_{10} \frac{MAX^2}{MSE_{YUV}}, \quad (22)$$

where MSE_Y, MSE_U, and MSE_V are the mean square error (MSE) of the Y, U, and V components, respectively. The MSE_{YUV} is a weighted combination of them. The MAX is the peak value of the YUV signal, which is equal to 1024 and 256 for the test sequences with 10 and 8 bits, respectively. Since the output bitrates of various algorithms are not the same, the Bjontegaard-Delta-Rate (BD-rate) and Bjontegaard-Delta-Peak-Signal-to-Noise-Ratio (BD-PSNR) [43] are employed in our experiments for a direct comparison.

We test two HDCA LFI datasets to verify the performance of the proposed algorithm: the Fraunhofer IIS HDCA dataset [6] and the Stanford HDCA dataset [44]. For the HDCA-based LFIs from Fraunhofer IIS with view resolution 101×21, we cropped the left 21×21 views as our test set to save some encoding time. Among all the HDCA-based LFIs in the Fraunhofer HDCA data set, we choose Set2, Set6, Set9, and Set10 as the test images. The HDCA-based LFIs from Stanford are with view resolution 17×17. Since various HDCA-based LFIs in this dataset have different characteristics, we adopt all of them as the test images. The detailed characteristics of the test HDCA-based LFIs including resolution and bit depth are shown in Table IV. Note that due to the large GOP size and view resolution of the test LFIs from the Fraunhofer IIS HDCA, the MV-HEVC coding structure, which is designed for a small number of views, will crash. Therefore, we only show the experimental results of Stanford HDCA dataset under the MV-HEVC structure.

TABLE IV
CHARACTERISTICS OF THE HDCA TEST SEQUENCES

DataSet	Sequence	Resolution	BitDepth
Fraunhofer IIS HDCA	Set2	3840×2160	10
	Set6	3840×2160	10
	Set9	3840×2160	10
	Set10	3840×2160	10
Stanford HDCA	Chess	1400×800	8
	Bulldozer	1536×1152	8
	Truck	1280×960	8
	Flowers	1280×1536	8
	Amethyst	768×1024	8
	Bracelet	1024×640	8
	Bunny	1024×1024	8
	Beans	1024×512	8
	Knights	1024×1024	8
	Cards	1024×1024	8
	Treasure	1536×1280	8
	Gantry	640×1024	8

B. Performance of the Overall Framework

The performance of the overall framework compared with the 1D hierarchical structure and the MV-HEVC is shown in Table V. In Table V, the Y-Rate and YUV-Rate mean the BD-rate for the Y and YUV components. The negative values indicate performance improvements. The Y-PSNR and YUV-PSNR mean the BD-PSNR for the Y and YUV components. The positive values indicate performance improvements. As we can see from Table V, the proposed framework shows performance improvements of 27.0% and 26.3% compared with the 1D hierarchical structure for the Y and YUV components, respectively. For the two test datasets, the average R-D performance improvement for Fraunhofer IIS HDCA is much better than for Standford HDCA data. This is because the proposed framework is designed for very complex datasets, and the Fraunhofer IIS HDCA data is much more complex than the Standford HDCA data.

The proposed framework can show a significant improvement for most sequences compared with the 1D Hierarchical coding structure. However, for the "Bunny" and "Beans" sequences in the Stanford HDCA dataset, the performance improvements are limited for the following two reasons. First, since the texture of the sequence is quite simple, a good prediction block is easy to obtain regardless of the distance between the reference frame and the current frame. The differences between various coding structures will be minimal. Second, since the total number of bytes used for these two sequences is relatively small, the bits cost of the reference picture set makes up a large proportion of the total bytes. This will reduce amount of improvement the proposed framework can have. It should also be noted that the proposed framework brings about 52.7% performance improvement for the test sequence "Set2", which are the highest among all the test sequences. As far as we can see, the benefit mainly comes from the very complex textures in this sequence. The correlations among the complex textures cannot be well represented by the 1D hierarchical structure but can be captured by the proposed framework.

In addition, we can also see from Table V that the proposed framework can achieve an average of 12.6% and 9.8% perfor-

TABLE V
PERFORMANCE OF THE OVERALL FRAMEWORK

DataSet	Sequence	Anchored on 1-D hierarchical coding structure				Anchored on the MV-HEVC structure			
		Y-Rate	YUV-Rate	Y-PSNR	YUV-PSNR	Y-Rate	YUV-Rate	Y-PSNR	YUV-PSNR
Fraunhofer IIS HDCA	Set2	–52.7%	–51.7%	2.34	2.25	–	–	–	–
	Set6	–35.1%	–34.5%	1.44	1.37	–	–	–	–
	Set9	–36.0%	–35.1%	1.54	1.46	–	–	–	–
	Set10	–44.1%	–42.9%	2.05	1.93	–	–	–	–
Stanford HDCA	Chess	–12.0%	–11.0%	0.36	0.31	2.9%	5.3%	–0.09	–0.15
	Bulldozer	–21.9%	–21.0%	0.85	0.78	–15.9%	–13.4%	0.67	0.53
	Truck	–12.9%	–12.6%	0.37	0.35	–2.2%	0.1%	0.08	0.01
	Flowers	–34.1%	–33.4%	0.97	0.90	–27.2%	–24.7%	0.80	0.69
	Amethyst	–31.7%	–30.9%	1.08	0.99	–24.8%	–22.5%	0.93	0.80
	Bracelet	–29.6%	–29.6%	0.78	0.76	–24.4%	–21.6%	0.71	0.61
	Bunny	–4.8%	–2.9%	0.20	0.15	6.4%	12.0%	–0.17	–0.32
	Beans	–2.8%	–1.9%	0.17	0.14	–3.5%	0.6%	0.20	0.03
	Knights	–21.0%	–21.0%	0.87	0.84	–14.0%	–13.0%	0.62	0.55
	Cards	–27.5%	–26.6%	0.74	0.69	–12.6%	–8.3%	0.38	0.25
	Treasure	–34.1%	–34.2%	1.12	1.06	–27.7%	–25.7%	0.94	0.83
	Gantry	–31.5%	–31.4%	1.10	1.05	–7.7%	–6.6%	0.26	0.22
Avg. of Fraunhofer IIS		–42.0%	–41.0%	1.84	1.75	–	–	–	–
Avg. of Stanford		–22.0%	–21.4%	0.72	0.67	–12.6%	–9.8%	0.44	0.34
Avg.		–27.0%	–26.3%	1.00	0.94	–12.6%	–9.8%	0.44	0.34

TABLE VI
PERFORMANCE OF THE PROPOSED ALGORITHMS WITH ONLY NON-NORMATIVE CHANGES

DataSet	Sequence	Anchored on 1-D hierarchical coding structure				Anchored on the MV-HEVC structure			
		Y-Rate	YUV-Rate	Y-PSNR	YUV-PSNR	Y-Rate	YUV-Rate	Y-PSNR	YUV-PSNR
Fraunhofer IIS HDCA	Set2	–49.2%	–47.9%	2.10	2.01	–	–	–	–
	Set6	–31.2%	–30.8%	1.24	1.19	–	–	–	–
	Set9	–32.4%	–31.4%	1.35	1.27	–	–	–	–
	Set10	–41.8%	–40.4%	1.89	1.77	–	–	–	–
Stanford HDCA	Chess	–8.6%	–7.8%	0.26	0.22	6.6%	8.8%	–0.20	–0.25
	Bulldozer	–17.8%	–17.0%	0.67	0.61	–11.4%	–8.8%	0.47	0.34
	Truck	–9.6%	–9.6%	0.28	0.26	1.4%	3.5%	–0.02	–0.08
	Flowers	–31.2%	–30.6%	0.88	0.81	–24.0%	–21.5%	0.70	0.59
	Amethyst	–29.7%	–29.0%	1.00	0.92	–22.7%	–20.5%	0.85	0.72
	Bracelet	–26.8%	–26.9%	0.70	0.68	–21.5%	–18.8%	0.62	0.53
	Bunny	0.4%	2.3%	0.07	0.03	12.3%	18.1%	–0.33	–0.48
	Beans	0.2%	0.8%	0.09	0.07	–0.5%	3.5%	0.09	–0.06
	Knights	–18.2%	–18.3%	0.74	0.72	–10.8%	–9.9%	0.47	0.42
	Cards	–24.7%	–24.0%	0.65	0.61	–9.2%	–5.0%	0.28	0.17
	Treasure	–31.5%	–31.7%	1.02	0.97	–24.8%	–22.8%	0.83	0.73
	Gantry	–28.1%	–28.0%	0.96	0.92	–3.1%	–2.0%	0.11	0.08
Avg. of Fraunhofer IIS		–38.7%	–37.6%	1.65	1.56	–	–	–	–
Avg. of Stanford		–18.8%	–13.3%	0.61	0.57	–9.0%	–6.3%	0.32	0.23
Avg.		–23.8%	–23.1%	0.87	0.82	–9.0%	–6.3%	0.32	0.23

mance improvements compared with the MV-HEVC structure for the Y and YUV components, respectively. The experimental results obviously demonstrate the effectiveness of the proposed framework. Note that we can also observe from the table that some sequences may suffer from a few performance losses. These performance losses are due to the fact that the maximum number of reference views in the reference frame buffer is 16 under the proposed framework while the maximum number of reference frames in the reference frame buffer under MV-HEVC is more than 80. However, the choices of the reference frames are restricted by the horizontal or vertical directions under the MV-HEVC, while we can use the reference frames from any directions under the proposed framework. In addition, the motion prediction in the vertical directions under MV-HEVC is inaccurate. Therefore, the proposed algorithms still show significant benefits on average compared with the MV-HEVC even with a smaller number of reference views.

C. Performance of the Proposed Algorithms With Only Non-Normative Changes

Among the three parts of the proposed algorithms, both the 2D hierarchical coding structure and the optimal bit allocation can be implemented with only non-normative changes to the HEVC encoder. It is very interesting to show the compression efficiency of the proposed algorithms without changing the HEVC decoder. The performance with only non-normative changes compared to the 1D hierarchical structure and the MV-HEVC is shown in Table VI. From Table VI, we can see that the non-normative changes can bring over 23% bitrate savings compared with the 1D hierarchical coding structure. Compared with the MV-HEVC, the proposed algorithms can bring, on average, 9.0% and 6.3% bitrate savings for the Y and YUV components, respectively. The experimental results demonstrate that the proposed framework with only non-normative changes to HEVC is very effective at compressing the HDCA-based LFI.

TABLE VII
PERFORMANCE OF THE 2D HIERARCHICAL CODING STRUCTURE

DataSet	Sequence	Anchored on 1-D hierarchical coding structure				Anchored on the MV-HEVC structure			
		Y-Rate	YUV-Rate	Y-PSNR	YUV-PSNR	Y-Rate	YUV-Rate	Y-PSNR	YUV-PSNR
Fraunhofer IIS HDCA	Set2	–43.9%	–44.1%	1.90	1.88	–	–	–	–
	Set6	–20.0%	–20.1%	0.78	0.76	–	–	–	–
	Set9	–22.7%	–22.6%	0.96	0.93	–	–	–	–
	Set10	–35.4%	–35.2%	1.67	1.61	–	–	–	–
Stanford HDCA	Chess	–1.7%	–1.5%	0.03	0.03	11.9%	13.3%	–0.38	–0.40
	Bulldozer	–12.3%	–12.7%	0.49	0.48	–7.5%	–6.3%	0.32	0.26
	Truck	–4.5%	–5.2%	0.11	0.13	5.1%	6.3%	–0.14	–0.17
	Flowers	–26.3%	–26.8%	0.79	0.76	–18.5%	–17.6%	0.58	0.52
	Amethyst	–22.9%	–23.6%	0.81	0.79	–15.4%	–14.8%	0.63	0.57
	Bracelet	–21.9%	–23.2%	0.54	0.56	–14.6%	–13.6%	0.48	0.43
	Bunny	3.3%	3.1%	–0.11	–0.10	12.1%	15.7%	–0.35	–0.44
	Beans	10.4%	10.5%	–0.26	–0.26	6.8%	10.3%	–0.14	–0.28
	Knights	–10.8%	–11.4%	0.46	0.47	–4.0%	–3.9%	0.19	0.18
	Cards	–20.9%	–21.2%	0.55	0.54	–5.4%	–3.1%	0.18	0.11
	Treasure	–27.7%	–28.6%	0.96	0.95	–20.0%	–19.3%	0.72	0.67
	Gantry	–21.0%	–21.7%	0.73	0.73	6.5%	6.4%	–0.18	–0.17
Avg. of Fraunhofer IIS		–30.5%	–30.5%	1.33	1.29	–	–	–	–
Avg. of Stanford		–13.0%	–13.5%	0.43	0.42	–3.6%	–2.2%	0.16	0.11
Avg.		–17.4%	–17.8%	0.65	0.64	–3.6%	–2.2%	0.16	0.11

TABLE VIII
PERFORMANCE OF THE DISTANCE-BASED MV SCALING PLUS 2D HIERARCHICAL CODING STRUCTURE

DataSet	Sequence	Anchored on 1-D hierarchical coding structure				Anchored on the MV-HEVC structure			
		Y-Rate	YUV-Rate	Y-PSNR	YUV-PSNR	Y-Rate	YUV-Rate	Y-PSNR	YUV-PSNR
Fraunhofer IIS HDCA	Set2	–46.2%	–46.3%	2.04	2.00	–	–	–	–
	Set6	–23.5%	–23.5%	0.93	0.91	–	–	–	–
	Set9	–26.3%	–26.1%	1.15	1.11	–	–	–	–
	Set10	–37.9%	–37.7%	1.81	1.74	–	–	–	–
Stanford HDCA	Chess	–4.3%	–4.1%	0.12	0.10	9.0%	10.5%	–0.29	–0.31
	Bulldozer	–15.5%	–15.8%	0.63	0.61	–11.0%	–9.7%	0.47	0.40
	Truck	–6.7%	–7.2%	0.18	0.19	2.8%	4.1%	–0.07	–0.10
	Flowers	–28.4%	–28.8%	0.86	0.83	–21.0%	–20.0%	0.65	0.59
	Amethyst	–24.9%	–25.5%	0.90	0.88	–17.7%	–17.0%	0.71	0.64
	Bracelet	–24.1%	–25.2%	0.62	0.64	–16.7%	–15.6%	0.55	0.50
	Bunny	–1.3%	–1.4%	0.03	0.02	7.2%	10.8%	–0.20	–0.31
	Beans	7.2%	7.6%	–0.17	–0.18	3.5%	7.1%	–0.02	–0.17
	Knights	–13.5%	–14.0%	0.59	0.59	–7.0%	–6.8%	0.32	0.30
	Cards	–23.4%	–23.6%	0.64	0.62	–8.4%	–6.0%	0.27	0.19
	Treasure	–29.3%	–30.2%	1.03	1.01	–21.9%	–21.1%	0.79	0.73
	Gantry	–23.5%	–24.1%	0.83	0.82	3.4%	3.3%	–0.09	–0.08
Avg. of Fraunhofer IIS		–33.5%	–33.4%	1.48	1.44	–	–	–	–
Avg. of Stanford		–15.6%	–16.0%	0.52	0.51	–6.5%	–5.0%	0.26	0.20
Avg.		–20.1%	–20.4%	0.76	0.74	–6.5%	–5.0%	0.26	0.20

D. Performance of the Separate Three Parts

1) Proposed 2D Hierarchical Coding Structure: The performance of the proposed 2D hierarchical structure individually compared with the 1D hierarchical structure and the MV-HEVC structure is shown in Table VII. We can see from Table VII that the proposed 2D hierarchical coding structure can bring an average of 30.5% and 13.0% bitrate savings for the Fraunhofer IIS HDCA and Stanford HDCA datasets, respectively. Comparing with the MV-HEVC structure, the proposed algorithm can show a 3.6% performance improvement on average to the Y component. The experimental results show that the proposed 2D hierarchical coding structure can better exploit the correlations among various views leading to improved coding performance. For most individual test sequences, significant improvements are achieved compared with the 1D hierarchical structure. Only two test sequences "Bunny" and "Beans" suffer performance losses. Since the 2D hierarchical coding structure has not yet been optimized to include distance-based MV scaling and optimal bit allocation, it is possible that the proposed coding structure may lead to some performance losses for sequences with simple textures.

2) Performance of the Distance-Based MV Scaling: The performance of the distance-based MV scaling plus the 2D hierarchical structure compared with the 1D hierarchical coding structure and MV-HEVC structure with POC-based MV scaling is shown in Table VIII. Through the comparison between Table VIII and Table VII, the distance-based MV scaling brings an average of around 3% bitrate savings compared with the POC-based MV scaling for the Fraunhofer HDCA and Stanford HDCA datasets. Under distance-based MV scaling, the proposed algorithm can more accurately predict the MV, thereby leading to significantly reduced

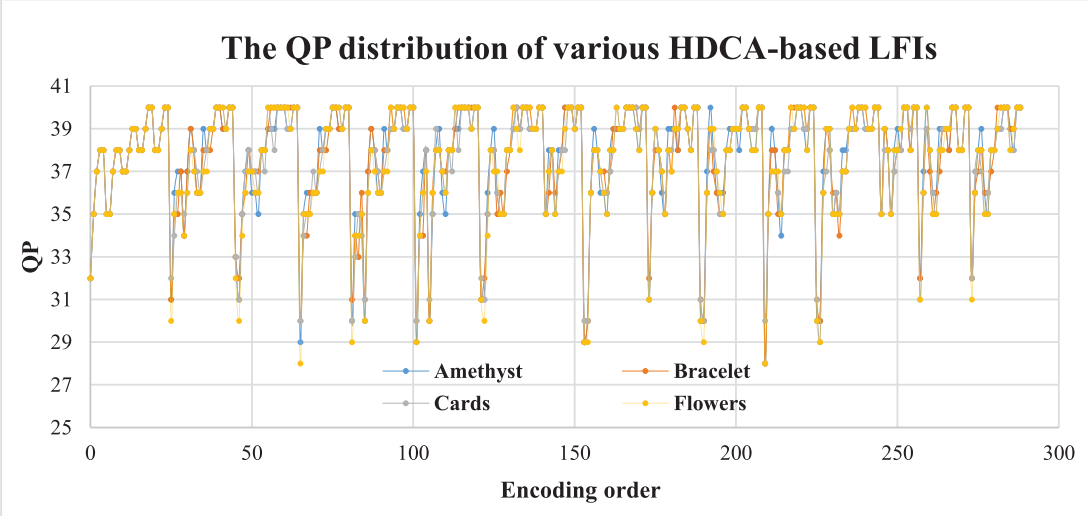


Fig. 6. QP distribution of various HDCA-based LFIs.

bits cost of the MV for all test sequences. Therefore, the distanced-based MV scaling provides consistent bitrate savings across all test sequences. As shown in our previous work [10], the distance-based MV scaling can only provide 0.6% performance improvements for the lenslet-based LFI. The experimental results show that the distance-based MV scaling for HDCA-based LFI are much better than for lenslet-based LFI.

3) Performance of the Optimal Bit Allocation: The performance of the optimal bit allocation compared with the framework without optimal bit allocation can be seen from the comparison between Table V and Table VIII. As we can see from these two tables, the optimal bit allocation method can bring about 6% performance improvements on average. To better explain the performance of the proposed bit allocation algorithm, the QP distributions of several typical sequences are shown in Fig. 6. First, we can see that the QP distributions of all these sequences follow the 2D hierarchical coding structure. The key frames which have the largest influences on the overall quality of the LFI are always coded using the smallest QP, while the non-reference frames are coded using the largest QP. Second, the QP distribution differences among all these LFIs are not as large as those of the general video sequences since they are captured through the same HDCA. However, there are still noticeable differences among various LFIs due to the different quality dependencies of various LFIs. The experimental results obviously demonstrate that the proposed bit allocation algorithm can capture the different quality dependencies among various views for each specified LFI and further optimize the 2D hierarchical coding structure.

As we can summarize from Table V, Table VII, and Table VIII, the proposed 2D hierarchical coding structure, the proposed distance-based MV scaling, and the proposed one-pass bit allocation achieve average performance improvements of 17.4%, 2.7%, and 6.9% compared with the 1D hierarchical coding structure, respectively. The experimental results obviously show that the proposed 2D hierarchical coding structure is the key step to improve the performance compared with the 1D hierarchical coding structure. The proposed distance-based MV scaling and one-pass bit allocation algorithms can further optimize the 2D hierarchical coding structure.

E. Some Examples of R-D Curves

To better explain the benefits of the proposed algorithm, we also present some examples of the R-D curves as shown in Fig. 7. The R-D curves obviously demonstrate that the proposed algorithm outperforms the 1D hierarchical and the MV-HEVC structures significantly. Also, we can see from the R-D curves that the proposed framework is able to achieve better R-D performance improvements in the high bitrate case instead of the low bitrate case. The differences of using different reference frames will be less in the low bitrate case due to the quantization errors. In addition, the bits cost of the reference picture set will have larger influences on the low bitrate case.

F. Subjective Quality

Fig. 8 gives two typical examples of the subjective quality comparisons between the 1D hierarchical coding structure and the proposed 2D hierarchical coding structure. From the cropped view of the LFI Cards, we can see that the left border of the image under the proposed algorithm recovers much better compared with the anchor. In addition, from the cropped view of the LFI Set2, we can see obvious artifacts as the lines under the anchor are not straight. Since the proposed algorithm can always provide the reference frames with relatively near distances for the current frame in both forward and backward directions, it can effectively avoid the cases where some border regions may be unable to find the corresponding blocks. The experimental results demonstrate that the proposed 2D hierarchical coding structure is able to achieve much better subjective quality compared with the 1D hierarchical coding structure.

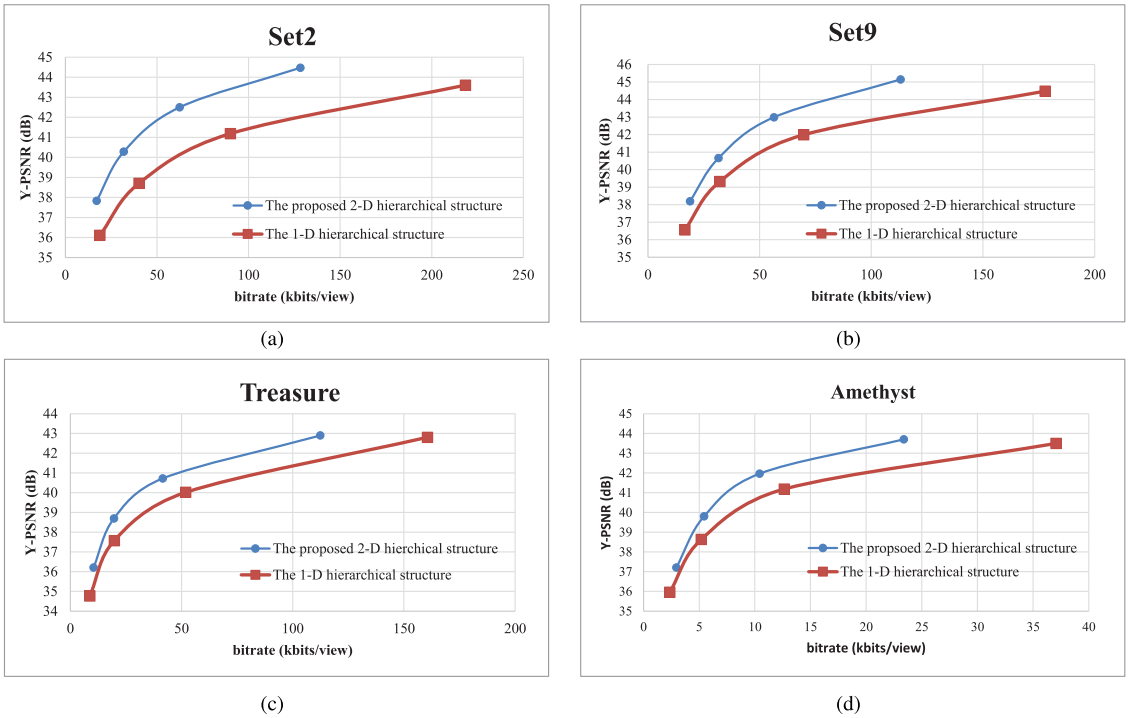


Fig. 7. Some examples of the R-D curves.

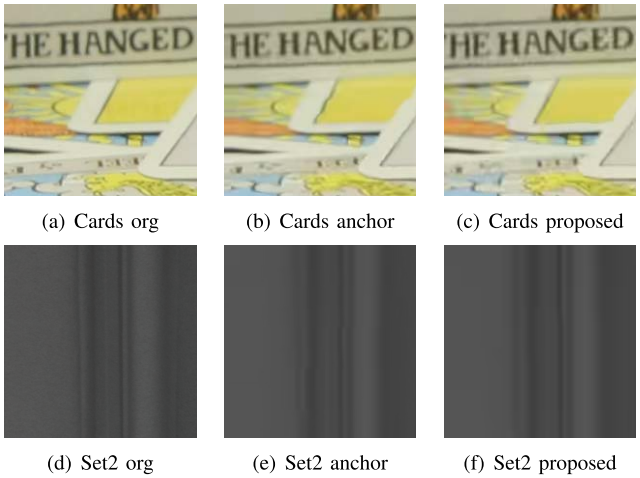


Fig. 8. Some examples of the subjective quality comparisons between the anchor and the proposed algorithm. (a)(b)(c): cropped 200×200 regions start from $(0, 350)$ on view $(0, 1)$ of the LFI Cards. (a) from the original picture, (b) from the 1D hierarchical coding structure with 6318 bits/view, (c) from the proposed algorithm with 5794 bits/view. (d)(e)(f): cropped 300×300 regions start from $(300, 0)$ on view $(0, 5)$ of the LFI Set2. (d) from the original picture, (e) from the 1D hierarchical coding structure with 18866 bits/view, (f) from the proposed algorithm with 17061 bits/view.

VI. Conclusion

In this paper, we propose using a quadtree-based 2D hierarchical coding framework to efficiently compress the HDCA-based LFI. The proposed framework is composed of the following three parts to fully exploit the correlations among various views. First, all the views are organized into a quadtree-based 2D hierarchical coding structure to fully exploit the correlations among various views. Second, the distance-based reference frame selection and MV scaling are proposed to improve the performance of the proposed coding structure. Third, a one-pass optimal bit allocation algorithm is provided to optimize the framework. The proposed framework is implemented based on the newest video coding standard High Efficiency Video Coding (HEVC). The experimental results show that the proposed framework can achieve over 25% and 10% bitrate savings compared with the 1D hierarchical coding structure and multi-view HEVC, respectively. The experimental results obviously demonstrate the effectiveness of the proposed framework.

The proposed framework can also be easily extended to a general form of the 2D hierarchical coding structure by specifying the number of frames in the horizontal and vertical directions within a GOP. We will delve into the general 2D hierarchical coding structure and try to apply it to more applications in our future work.

References

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *Proc. 23rd Annu. Conf. Comput. Graph. Interact. Techn. (SIGGRAPH)*. New York, NY, USA: ACM, 1996, pp. 31–42.

[2] J. Arai *et al.*, "Integral three-dimensional television using a 33-megapixel imaging system," *J. Display Technol.*, vol. 6, no. 10, pp. 422–430, Oct. 2010.

[3] Y. Xue *et al.*, "High-accuracy and real-time 3D positioning, tracking system for medical imaging applications based on 3D digital image correlation," *Opt. Lasers Eng.*, vol. 88, pp. 82–90, Jan. 2017.

[4] T. Ebrahimi, S. Foessel, F. Pereira, and P. Schelkens, "JPEG Pleno: Toward an efficient representation of visual reality," *IEEE MultiMedia*, vol. 23, no. 4, pp. 14–20, Oct./Dec. 2016.

[5] P. Schelkens *et al.*, "JPEG Pleno: Providing representation interoperability for holographic applications and devices," *ETRI J.*, vol. 41, no. 1, pp. 93–108, 2019.

[6] *High Density Camera Array Data Set from Fraunhofer IIS*. Accessed: 2019. [Online]. Available: https://www.iis.fraunhofer.de/en/ff/amm/dl/lightfielddataset.html

[7] C. Conti, L. D. Soares, and P. Nunes, "High Efficiency Video Coding-based 3D holoscopic video coding using self-similarity compensated prediction," *Signal Process. Image Commun.*, vol. 42, pp. 59–78, Mar. 2016.

[8] Y. Li, M. Sjöström, R. Olsson, and U. Jennehag, "Coding of focused plenoptic contents by displacement intra prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 7, pp. 1308–1319, Jul. 2016.

[9] D. Liu, L. Wang, L. Li, Z. Xiong, F. Wu, and W. Zeng, "Pseudo-sequence-based light field image compression," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2016, pp. 1–4.

[10] L. Li, Z. Li, B. Li, D. Liu, and H. Li, "Pseudo-sequence-based 2-D hierarchical coding structure for light-field image compression," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 1107–1119, Oct. 2017.

[11] W. Ahmad, M. Sjöström, and R. Olsson, "Compression scheme for sparsely sampled light field data based on pseudo multi-view sequences," *Proc. SPIE*, vol. 10679, May 2018, Art. no. 106790M.

[12] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (High Efficiency Video Coding) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[13] C. Conti, J. Lino, P. Nunes, L. D. Soares, and P. L. Correia, "Spatial prediction based on self-similarity compensation for 3D holoscopic image and video coding," in *Proc. 18th IEEE Int. Conf. Image Process.*, Sep. 2011, pp. 961–964.

[14] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, Jul. 2003.

[15] C. Conti, P. Nunes, and L. D. Soares, "High Efficiency Video Coding-based light field image coding with bi-predicted self-similarity compensation," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2016, pp. 1–4.

[16] R. Monteiro *et al.*, "Light field High Efficiency Video Coding-based image coding using locally linear embedding and self-similarity compensated prediction," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2016, pp. 1–4.

[17] R. J. S. Monteiro, P. J. L. Nunes, N. M. M. Rodrigues, and S. M. M. Faria, "Light field image coding using high-order intra-block prediction," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 1120–1131, Oct. 2017.

[18] X. Jin, H. Han, and Q. Dai, "Image reshaping for efficient compression of plenoptic content," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 1173–1186, Oct. 2017.

[19] X. Jin, H. Han, and Q. Dai, "Plenoptic image coding using macropixel-based intra prediction," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3954–3968, Aug. 2018.

[20] C. Perra and P. Assuncao, "High efficiency coding of light field images based on tiling and pseudo-temporal data arrangement," in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2016, pp. 1–4.

[21] H. P. Hariharan, T. Lange, and T. Herfet, "Low complexity light field compression based on pseudo-temporal circular sequencing," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2017, pp. 1–5.

[22] W. Zhang, D. Liu, Z. Xiong, and J. Xu, "SIFT-based adaptive prediction structure for light field compression," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2017, pp. 1–4.

[23] W. Ahmad, R. Olsson, and M. Sjöström, "Interpreting plenoptic images as multi-view sequences for improved compression," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 4557–4561.

[24] 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 Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 35–49, Jan. 2016.

[25] X. Jiang, M. Le Pendu, R. A. Farrugia, and C. Guillemot, "Light field compression with homography-based low-rank approximation," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 1132–1145, Oct. 2017.

[26] S. Zhao and Z. Chen, "Light field image coding via linear approximation prior," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 4562–4566.

[27] J. Chen, J. Hou, and L.-P. Chau, "Light field compression with disparity-guided sparse coding based on structural key views," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 314–324, Jan. 2018.

[28] I. Tabus, P. Helin, and P. Astola, "Lossy compression of lenslet images from plenoptic cameras combining sparse predictive coding and JPEG 2000," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 4567–4571.

[29] B. Guo, Y. Han, and J. Wen, "Convex optimization based bit allocation for light field compression under weighting and consistency constraints," in *Proc. Data Compress. Conf.*, Mar. 2018, pp. 107–116.

[30] C. Jia *et al.*, "Optimized inter-view prediction based light field image compression with adaptive reconstruction," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 4572–4576.

[31] X. Zhu, A. Aaron, and B. Girod, "Distributed compression for large camera arrays," in *Proc. IEEE Workshop Stat. Signal Process.*, Sep./Oct. 2003, pp. 30–33.

[32] G. K. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, no. 4, pp. 30–44, 1991.

[33] B. Girod, A. M. Aaron, S. Rane, and D. Rebollo-Monedero, "Distributed video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 71–83, Jan. 2005.

[34] V. K. Adhikarla, A. T. Islam, P. T. Kovács, and O. Staadt, "Fast and efficient data reduction approach for multi-camera light field display telepresence systems," in *Proc. 3DTV Vis. Beyond Depth (3DTV-CON)*, Oct. 2013, pp. 1–4.

[35] E. Cornwell, L. Li, Z. Li, and Y. Sun, "An efficient compression scheme for the multi-camera light field image," in *Proc. IEEE 19th Int. Workshop Multimedia Signal Process. (MMSP)*, Oct. 2017, pp. 1–6.

[36] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2006, pp. 1929–1932.

[37] B. Li, H. Li, L. Li, and J. Zhang, "λ-domain rate control algorithm for High Efficiency Video Coding," *IEEE Trans. Image Process.*, vol. 23, no. 9, pp. 3841–3854, Sep. 2014.

[38] L. Li, B. Li, H. Li, and C. W. Chen, "λ-domain optimal bit allocation algorithm for High Efficiency Video Coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 1, pp. 130–142, Jan. 2018.

[39] Y. Gao, C. Zhu, S. Li, and T. Yang, "Source distortion temporal propagation analysis for random-access hierarchical video coding optimization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 2, pp. 546–559, Feb. 2019.

[40] S. Wang, S. Ma, S. Wang, D. Zhao, and W. Gao, "Rate-GOP based rate control for High Efficiency Video Coding," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 6, pp. 1101–1111, Dec. 2013.

[41] K. Andersson *et al.*, *Non-normative HM Encoder Improvements*, document JCTVC-W0062, San Diego, CA, USA, Feb. 2016.

[42] *High Efficiency Video Coding test model, HM-16.7*. Accessed: 2019. [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/tags/

[43] G. Bjontegaard, *Calculation of Average PSNR Differences Between RD-Curves*, document VCEG-M33, Austin, TX, USA, Apr. 2001.

[44] *High Density Camera Array Data Set from Stanford*. Accessed: 2019. [Online]. Available: http://lightfield.stanford.edu/lfs.html

Li Li (M'17) received the B.S. and Ph.D. degrees in electronic engineering from the University of Science and Technology of China (USTC), Hefei, Anhui, China, in 2011 and 2016, respectively. He is currently a Post-Doctoral Researcher with the University of Missouri–Kansas City.

He is also with USTC. His research interests include image/video coding and processing. He received the Best 10% Paper Award at the 2016 IEEE Visual Communications and Image Processing (VCIP) Conference.

Zhu Li (M'02–SM'07) received the Ph.D. degree in electrical and computer engineering from Northwestern University, Evanston, in 2004.

He was a Principal Staff Research Engineer with the Multimedia Research Lab (MRL), Motorola Labs, from 2000 to 2008, an Assistant Professor with the Department of Computing, The Hong Kong Polytechnic University, from 2008 to 2010, the Sr. Staff Researcher/Media Analytics Lead with FutureWei (Huawei) Technology's Media Lab, Bridgewater, NJ, USA, from 2010 to 2012, and the Sr. Staff Researcher/Sr. Manager with the Samsung Research America's Multimedia Standards Research Lab, Richardson, TX, USA, from 2012 to 2015. He was the AFOSR SFFP Summer Visiting Faculty with the UAV Research Center, United States Air Force Academy (USAFA), from 2016 to 2018. He is currently an Associate Professor with the Department of Computer Science and Electrical Engineering (CSEE), University of Missouri–Kansas City, and the Director of the NSF I/UCRC Center for Big Learning (CBL), UMKC. His research interests include point cloud and light field compression, graph signal processing and deep learning in the next-generation visual compression, and image processing and understanding. He has 46 issued or pending patents and over 100 publications in book chapters, journals, and conferences in these areas. He has been serving as a Steering Committee Member for the IEEE ICME since 2015. He is also an elected member of the IEEE Multimedia Signal Processing (MMSP), the IEEE Image, Video, and Multidimensional Signal Processing (IVMSP), and the IEEE Visual Signal Processing and Communication (VSPC) Tech Committees. He received the Best Paper Award at the IEEE International Conference on Multimedia and Expo (ICME), Toronto, in 2006, and the Best Paper Award (DoCoMo Labs Innovative Paper) at the IEEE International Conference on Image Processing (ICIP), San Antonio, in 2007. He is also the Program Co-Chair of the IEEE International Conference on Multimedia and Expo (ICME) 2019 and the Co-Chair of the IEEE Visual Communication and Image Processing (VCIP) 2017. He has been an Associate Editor of the IEEE TRANSACTIONS ON MULTIMEDIA, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, and the *Journal of Signal Processing Systems* (Springer) since 2015.

Bin Li (M'14) received the B.S. and Ph.D. degrees in electronic engineering from the University of Science and Technology of China (USTC), Hefei, Anhui, China, in 2008 and 2013, respectively.

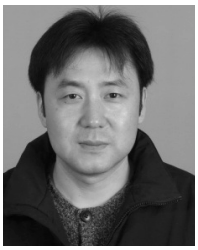
In 2013, he joined Microsoft Research Asia (MSRA), Beijing, China, where he is currently a Lead Researcher. He has authored or coauthored over 40 papers. He holds over 20 granted or pending U.S. patents in the area of image and video coding. He has more than 30 technical proposals that have been adopted by the Joint Collaborative Team on Video Coding. His current research interests include video coding, processing, and communication.

Dr. Li received the Best Paper Award at the the International Conference on Mobile and Ubiquitous Multimedia from Association for Computing Machinery in 2011, the Top 10% Paper Award at the 2014 IEEE International Conference on Image Processing, and the Best Paper Award at the 2017 IEEE Visual Communications and Image Processing. He was the Co-Chair of the Ad Hoc Group of Screen Content Coding Extensions Software Development. He is also the Vice Chair of the Ad Hoc Group of HEVC and HDRTools Software Development and Software Technical Evaluation. He has been an active contributor to the ISO/MPEG and ITU-T video coding standards.

Dong Liu (M'13–SM'19) received the B.S. and Ph.D. degrees in electrical engineering from the University of Science and Technology of China (USTC), Hefei, China, in 2004 and 2009, respectively.

In 2012, he joined USTC as an Associate Professor. He was a member of the Research Staff at the Nokia Research Center, Beijing, China, from 2009 to 2012. He has authored or coauthored more than 100 papers in international journals and conferences. He has 16 granted patents and one technical proposal adopted by AVS. His research interests include image and video coding, multimedia signal processing, and multimedia data mining.

Dr. Liu is also a Senior Member of CSIG. He received the 2009 IEEE Transactions on Circuits and Systems for Video Technology Best Paper Award and the Best 10% Paper Award at VCIP 2016. He and his team were the winners of several technical challenges that were held at ICME 2016, ACM MM 2018, ECCV 2018, and CVPR 2018, respectively. He has served as the Symposium Co-Chair for the WCSP 2014 and the Registration Co-Chair for the ICME 2019.



Houqiang Li (M'10–SM'12) received the B.S., M.Eng., and Ph.D. degrees in electronic engineering from the University of Science and Technology of China, Hefei, China, in 1992, 1997, and 2000, respectively.

He is currently a Professor with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. He has authored or coauthored over 100 papers in journals and conferences. His research interests include video coding and communication, multimedia search, and image/video analysis. He was a recipient of the Best Paper Award at the International Conference on Mobile and Ubiquitous Multimedia of the ACM (ACM MUM) in 2011, the Visual Communications and Image Processing (VCIP) in 2012, and the International Conference on Internet Multimedia Computing and Service (ICIMCS) in 2012. He is the Senior Author of the Best Student Paper at the 5th International Mobile Multimedia Communications Conference (MobiMedia) in 2009. He has served as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY from 2010 to 2013. He has been on the Editorial Board of the *Journal of Multimedia* since 2009.