Advanced Scalability for Light Field Image Coding

Hadi Amirpour, *Member, IEEE*, Christine Guillemot, *Fellow, IEEE*,
Mohammad Ghanbari, *Life Fellow, IEEE*, and Christian Timmerer, *Senior Member, IEEE*

***Abstract*— Light field imaging, which captures both spatial and angular information, improves user immersion by enabling post-capture actions, such as refocusing and changing view perspective. However, light fields represent very large volumes of data with a lot of redundancy that coding methods try to remove. State-of-the-art coding methods indeed usually focus on improving compression efficiency and overlook other important features in light field compression such as scalability. In this paper, we propose a novel light field image compression method that enables *(i)* viewport scalability, *(ii)* quality scalability, *(iii)* spatial scalability, *(iv)* random access, and *(v)* uniform quality distribution among viewports, while keeping compression efficiency high. To this end, light fields in each spatial resolution are divided into sequential viewport layers, and viewports in each layer are encoded using the previously encoded viewports. In each viewport layer, the available viewports are used to synthesize intermediate viewports using a video interpolation deep learning network. The synthesized views are used as virtual reference images to enhance the quality of intermediate views. An image super-resolution method is applied to improve the quality of the lower spatial resolution layer. The super-resolved images are also used as virtual reference images to improve the quality of the higher spatial resolution layer. The proposed structure also improves the *flexibility* of light field streaming, provides *random access* to the viewports, and increases *error resiliency*. The experimental results demonstrate that the proposed method achieves a high compression efficiency and it can adapt to the display type, transmission channel, network condition, processing power, and user needs.**

***Index Terms*— Light field, compression, scalability, random access, deep learning.**

I. Introduction

LIGHT field imaging is a promising technology for providing an immersive experience to the users [1]. Unlike

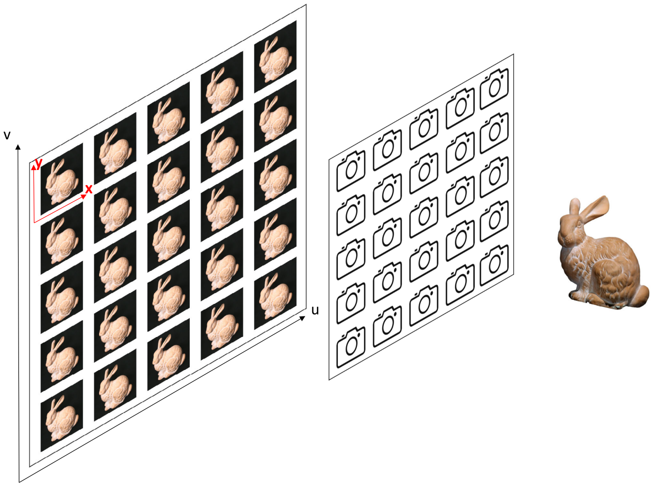


Fig. 1. Light fields are typically represented by multiview images. (u,v) represents the view location while (x,y) denotes the pixel location in each view.

traditional photography that integrates angular information into a 2D image, light field imaging collects both spatial and angular information, resulting in a grid of 2D views, enabling functionalities such as changing viewport, synthesizing new views, and immersive navigation within the captured scene. However, light fields come with a huge amount of data for transmission and/or storage, making their compression and transmission a challenging task. Therefore, a highly efficient light field compression method is required to deal with these images for transmission/storage. Light field compression methods are mainly categorized into two groups [2]: *(i)* transform-based coding and *(ii)* predictive-based coding methods.

The Discrete Cosine Transform (DCT) [3], Discrete Wavelet Transform (DWT) [4], Karhunen Loeve Transform (KLT) [5], and Graph Fourier Transform (GFT) [6] are among the transformations that have been applied to light fields to reduce their redundancy in the transform domain. Such a transform-based solution has been adopted in the 4D transform mode, also known as the Multidimensional Light field Encoder (MuLE) [3] of JPEG Pleno. The 4D redundancy of light fields is exploited by applying a *4D-DCT* transform to 4D spatio-angular blocks. Rizkallah et al. [7] propose a graph-transform based light field compression method using a rate-distortion optimized graph coarsening and partitioning algorithm.

Predictive-based coding approaches are typically based on *(i)* non-local spatial prediction, *(ii)* inter-view prediction, and *(iii)* view synthesis methods. Non-local spatial prediction approaches have been used to reduce the redundancy

Manuscript received 28 November 2021; revised 4 August 2022 and 19 October 2022; accepted 9 November 2022. Date of publication 29 November 2022; date of current version 7 December 2022. This work was supported in part by the European Union (EU) H2020 Research and Innovation Programme (European Research Council (ERC) Advanced Grant Computational Light Field Imaging (CLIM), and in part by the French Agence Nationale de la Recherche (ANR) Research Agency in the Context of the Project DeepCIM) under Agreement 694122. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yulan Guo. *(Corresponding author: Hadi Amirpour.)*

Hadi Amirpour and Christian Timmerer are with the Christian Doppler Laboratory ATHENA, Institute of Information Technology (ITEC), University of Klagenfurt, 9020 Klagenfurt, Austria (e-mail: hadi.amirpour@aau.at).

Christine Guillemot is with the Inria Rennes-Bretagne Atlantique, 35042 Rennes, France (e-mail: christine.guillemot@inria.fr).

Mohammad Ghanbari is with the Christian Doppler Laboratory ATHENA, Institute of Information Technology (ITEC), University of Klagenfurt, 9020 Klagenfurt, Austria, and also with the School of Computer Science and Electronic Engineering, University of Essex, CO4 3SQ Colchester, U.K. (e-mail: ghan@essex.ac.uk).

Digital Object Identifier 10.1109/TIP.2022.3223787

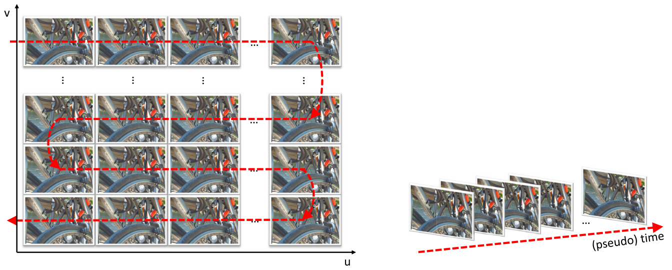


Fig. 2. Example of converting multiview images of a light field into a PVS using the serpentine scan order.

within a lenslet image [8], [9]. *High Efficiency Video Coding* (HEVC) [10] or *Versatile Video Coding* (VVC) [11] coding standards have also been used to reduce the redundancy between light field views thanks to inter-view prediction methods. Light field views are reordered as a pseudo video sequence (PVS) and the generated PVS is fed into the video codec. A predefined scan order such as raster and spiral [12], [13] is typically used to generate a PVS. Fig. 2 depicts the conversion of the multiview images to a PVS using the serpentine scan order. Wang et al. [14] analyze the relationship of the inter-view prediction structure with the coding performance and propose an efficient prediction structure for light field coding.

In synthesized-based approaches, a sparse set of light field views is first encoded and used to synthesize (predict) the remaining views using view synthesis methods, including *(i)* Depth Image Based Rendering (DIBR), as in the Warping, merging and Sparse Prediction encoder (WaSP) [15], which has been adopted in the JPEG Pleno coding standard, or in [16], *(ii)* transform-assisted [17], and *(iii)* learning-based view synthesis [18], [19] approaches.

Dib et al. [20] use a transform-assisted view synthesis method to compress light fields. A subset of views is first inter-coded and then used to synthesize the next subset of views using the Fourier Disparity Layer (FDL) representation. The prediction residuals are then inter-coded and used to enhance the quality of synthesized views and refine the FDL representation. Ahmad et al. [21] divide light field views into two groups, namely, key views and decimated views. Key views are encoded using MV-HEVC. They are then used to synthesize the decimated views using the shearlet transform. The residuals of synthesized views are then encoded as a single PVS.

Hou et al. [18] propose a bi-level compensation approach which uses the learning-based view synthesis Deep Neural Network (DNN) proposed in [22] for light field compression. The four corner views are inter-coded first and after decoding, they are fed to the DNN to synthesize the remaining views. The residuals between the synthesized views and their corresponding target views are reordered as a PVS and inter-coded. Jia et al. [23] propose a light field compression method based on a Generative Adversarial Network (GAN). They first generate a PVS by sparsely sampling light field views following a chessboard pattern. The intermediate views are then synthesized from the decoded PVS views using the GAN. The residuals between synthesized views and their corresponding target views are then inter-coded to enhance the quality of the synthesized views. Hu et al. [19] propose an adaptive two-layer light field compression method based on Graph Neural Network (GNN) reconstruction. Low- and high-frequency components are encoded using different approaches. The high-frequency view components are converted into a PVS and encoded using HEVC. The low-frequency components of the views are resampled in the angular dimension and the selected views are inter-coded. The discarded views are synthesized using the GNN. Bakir et al. [24] use VVC's temporal scalability structure to encode key views which are then fed to a GAN to synthesize the remaining views.

Some approaches provide a form of scalability when coding light fields. Conti et al. [25] propose a viewport scalable coding solution for 3D light fields based on an inter-layer prediction scheme that exploits the redundancy between multiview and lenslet representations. Li et al. [26] propose a three layers disparity-compensated scheme for scalable coding of lenslet images. Garrote et al. [27] propose a scalable scheme based on the wavelet transform for lenslet image coding. Conti et al. [28], [29] propose a light field coding solution with field of view scalability, which supports region of interest enhancement. Komatsu et al. [30] propose a light field coding using weighted binary images with the support of quality scalability. Rüefenacht et al. [31] propose a scalable light field coding approach based on the base-anchored representation, including scalable compression of the disparity information itself.

In this paper, we propose a *flexible* light field compression method that can be adapted to the user's needs by supporting the following functionalities: *(a)* viewport scalability, *(b)* spatial scalability, *(c)* quality scalability, *(d)* random access, and *(e)* uniform quality distribution. The proposed framework extends the method described in [32] in several ways. It first adds spatial scalability based on a single image super-resolution approach which is shown to give a very high rate-distortion performance for each target spatial resolution. The flexibility of the encoding structure has been increased by adding spatial scalability in addition to the viewport and quality scalabilities. This increased flexibility allows us to better address the various trade-offs between encoding efficiency, random access, and the different forms of scalability. A comprehensive analysis is carried out using a light field dataset with a large parallax which is more challenging in terms of encoding efficiency as well as low parallax light fields.

In a nutshell, we first downscale light field views to a lower resolution to make two spatial layers: *(i)* Spatial Layer 1 (SL_1) and *(ii)* Spatial Layer 2 (SL_2). Views in each spatial layer are divided into Viewport Layers (VLs). Fig. 3 depicts the structuring of 5×5 light field views into spatial and viewport layers. In each VL, the available views are used to synthesize intermediate views and the synthesized views are used as virtual reference images to predict their corresponding views. To encode views in SL_2, super-resolution is applied to their corresponding encoded viewports in SL_1 and they are also added to the reference image list.

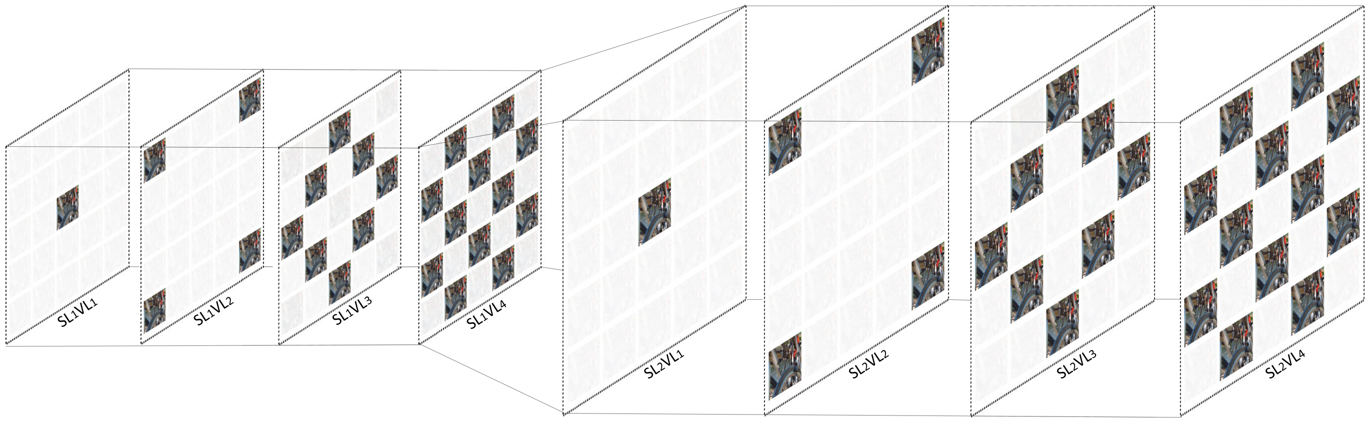


Fig. 3. Light field view structuring in spatial and viewport layers.

The remainder of the paper is organized as follows. The theoretical background for light field imaging is introduced in Section II. The functionalities supported by our proposed method are introduced in Section III. Section IV presents the proposed light field encoding method. Experimental results are provided in Section V and Section VI presents the concluding remarks.

II. Light Fields

A light field is a quantized representation of the 7D plenoptic function [33], *i.e.*,

$$P = P(\phi, \theta, x, y, z, \lambda, t) \tag{1}$$

where all light rays at every possible location (x, y, z), at every possible direction (θ, ϕ), at any time (t), over any range of wavelengths (λ) are recorded. The light field representation can be simplified based on some assumptions. First, light rays are considered time-invariant, and monochromatic, resulting in removing time (t) and wavelength (λ) dimensions. Second, the light rays are assumed to travel in a free space, which leads to removing another dimension. Therefore, a light field is represented by a 4D function as follows:

$$LF = P(x, y, u, v) \tag{2}$$

where (u,v) represents the view location, and (x,y) denotes the pixel location in each view. A two-plane parameterization can be used to model light fields, and they are represented as multiview images as shown in Fig. 1. To acquire light fields, multi-array or lenslet cameras are used. For lenslet cameras, the spatial and angular domains are multiplexed into a single 2D image, known as a lenslet image. The lenslet image can be converted into a multiview representation [34].

III. Functionalities in Light Field Compression

In this section, we highlight the functionalities supported by our proposed light field coding method.

A. Viewport Scalability

Viewport scalability for light fields is provided by grouping light field views into different layers. In this way the adaptation to *(i)* capturing device, *(ii)* display, *(iii)* network condition, *(iv)* processing power, and *(v)* storage capacity is enhanced. For example, 2D displays might require the central view, while 3D/stereo displays need only the central view and two of its side views. For light field displays, layers can be transmitted, decoded, and displayed one after another. PVS-based methods make all the views dependent on each other to highly utilize redundancy among the views and increase the compression efficiency. However, to access an arbitrary view, *e.g.*, the central view on a 2D display, all light field views should be encoded, transmitted, and decoded. This will lead to both bandwidth and processing power wastage as well as decoding delay [35]. Monteiro et al. [36] divide the light field views into multiple viewport layers and encode the views in each layer by using the previously encoded/decoded views in the same layer or in prior layers as references.

B. Quality Scalability

Through quality scalability, the adaptation to the network condition is provided. In this way, light fields are encoded in two (or more) quality layers and the quality of light fields can be improved by transmitting enhancement layers when enough bandwidth or processing power is available. In synthesizing views, some approaches introduced in the previous section, *e.g.*, [18], [21] encode their residuals as a quality enhancement layer to improve the quality of the synthesized image.

C. Spatial Scalability

To address various devices and display resolutions it is important to provide spatial scalability. In this regard, images are encoded at two (or more) spatial resolutions. The lower resolution is encoded as the base layer and it is used as a reference to encode the higher resolution(s), *i.e.*, enhancement layer(s).

D. Viewport Random Access

Navigation between various viewports is another important factor to be considered in light field encoding solutions. Since light field views in an inter-view prediction are highly dependent on each other, navigation between different views may require a huge amount of views to be decoded which can have a high cost on decoding delay, bandwidth requirement,

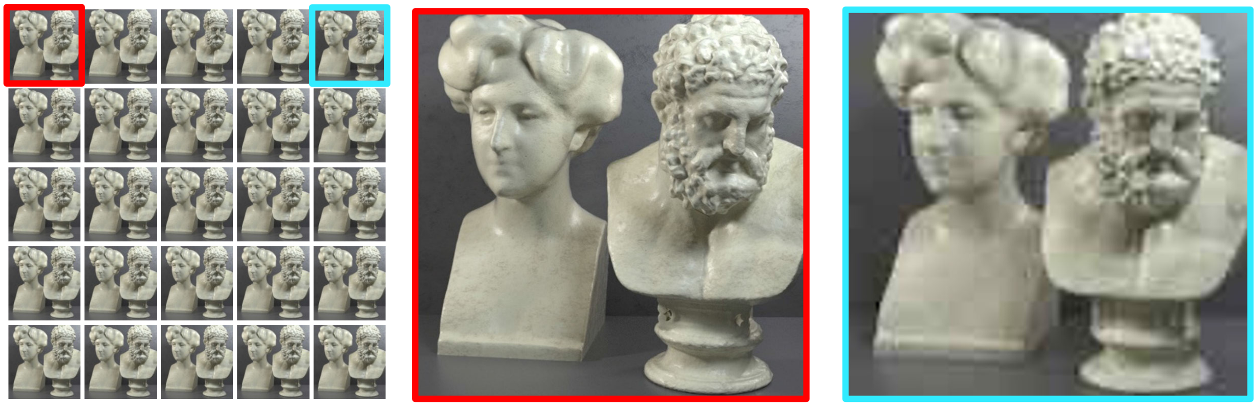


Fig. 4. Quality variation when a user navigates between the top-left and top-right views.

and processing power. To avoid these problems, random access to the image views should be considered in light field coding [35], [37]. Therefore, JPEG Pleno defines various metrics within its light field coding common test conditions [38]. The random access metric (RA) is defined as:

$$RA = \mathit{Total\ amount\ of\ encoded\ bits\ required\ to\ access\ a\ view} \tag{3}$$

The random access penalty metric (RA_p) is considered as the maximum RA among all views as:

$$RA_p = \max_{all\ views} RA \tag{4}$$

The relative random access penalty metric (RRA_p) is defined as:

$$RRA_p = \frac{RA_p}{\mathit{Total\ amount\ of\ encode\ bits\ to\ decode\ the\ full\ light\ field}} \tag{5}$$

In PVS-based light field coding solutions, RRA_p is equal to 1, which means to access a view, the whole encoded light field should be transmitted and the whole bitstream should be decoded (to access, *e.g.*, the last view). In encoding light fields, some compression methods focus on improving random access to arbitrary views [36], [37], [39], [40], [41], [42], [43].

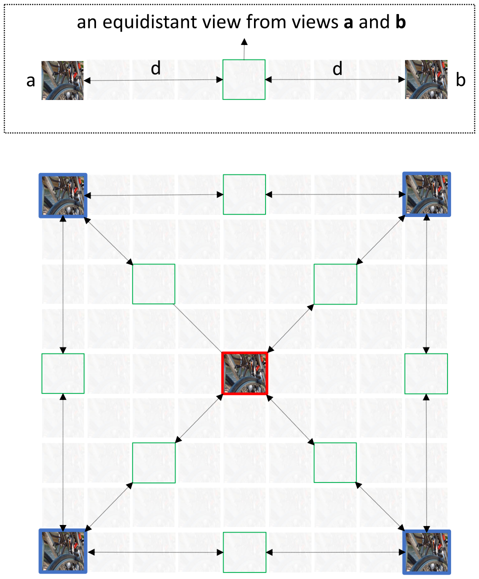


Fig. 5. VL_3 comprises the views that are equidistant from views in VL_1 and VL_2. 🟥 represents the view of VL_1, 🟦 represents views of VL_2, and 🟩 represents views of VL_3.

E. Uniform Quality Distribution

Light field views in a given number of encoded bits should have similar quality at any view. It is undesirable to provide light field views in a way that users face different quality levels when navigating between viewports. Fig. 4 illustrates the quality variation when a user navigates between the top-left and top-right views in case there is a significant difference between the quality of those image views.

IV. Scalable Light Field Coding

To address the above-mentioned functionalities, a *flexible* light field compression method is proposed in this paper. To provide spatial scalability, a light field LF is spatially downscaled to a lower resolution ($\times\frac{1}{2}$ in each direction). Therefore, the light field views are provided in two spatial layers; *(i)* SL_1 (low resolution), and *(ii)* SL_2 (original resolution).

To support viewport scalability, both spatial layers are divided into multiple viewport layers, each containing a subset of views. *(i)* SL_xVL_1 consists of only the central view of the spatial layer x. *(ii)* SL_xVL_2 consists of four corner views of the spatial layer x. *(iii)* SL_xVL_m $(3 \leq m \leq n)$ comprises the views that are equidistant from views in viewport layers *1* to *m-1* in the spatial layer x, *i.e.*, SL_xVL_1 to SL_xVL_{m-1}. A view is equidistant from two other views if it is the same distance from them. For instance, as shown in Fig. 5, VL_3 comprises the views that are equidistant from views in VL_1 and VL_2.

The maximum number of viewport layers (n) is determined by the angular resolution of light fields. For example, a light field of 5 × 5 views will be decomposed into four viewport layers ($n = 4$), a light field of 9×9 views will be decomposed into five viewport layers ($n = 5$), and a light field with 17 × 17 angular resolution will be decomposed into six viewport layers ($n = 6$) for each spatial resolution. Fig. 3 shows the way a light field with an angular resolution of 5 × 5 views is structured into two spatial resolution layers, SL_1 and SL_2, and four viewport layers per each spatial layer.

A. Compression of SL_1

We encode views at different viewport layers in different way. *(i)* SL_1VL_1: the central view is intra-coded, hence it can be accessed independently. *(ii)* SL_1VL_2: views in the second layer are encoded *independently* of each other, however, using inter-coding taking the central view as a reference image. *(iii)* SL_1VL_m $(3 \leq m \leq n)$: the remaining views are encoded using a predictor based on a view interpolation method as described in the following.

In video frame interpolation methods, the optical flow between two input frames, *i.e.*, a per pixel translational displacement, is estimated and subsequently, the intermediate frame guided by motion is synthesized. DNNs are promising techniques to generate intermediate frames or – in our case – views. Many video frame interpolation methods using DNNs have been introduced [44], [45]. In this paper, we use RIFE [46] for view interpolation as it allows real-time flow estimation without any limit on the maximum number of interpolated views, which makes it flexible to support a varying number of viewport layers (cf. Section V-F).

Fig. 6.a illustrates the use of *RIFE* to synthesize the top view in the 3^{rd} viewport layer (SL_1VL_3) from two input images, *i.e.*, the top-left and top-right views of the second viewport layer (SL_1VL_2). The residual images between the ground truth top view in SL_1VL_3 and these three images are also shown in Fig 6.b. It is seen that the synthesized view has more correlation with the target view and, thus, it can serve as a better reference for predicting the top view in the 3^{rd} viewport layer (SL_1VL_3). We therefore use these three views, *i.e.*, top-left, top-right, and synthesized views, as reference images in the reference lists of the standard video codec VVC [11] to inter-code the top view in the 3^{rd} viewport layer (SL_1VL_3). The Rate-Distortion (RD) performance (see Fig. 6.c) shows a significant improvement when the synthesized view is used as the reference.

When a synthesized view is used for prediction, it is added as a *virtual reference frame* to the Decoded Picture Buffer (DPB), which stores pictures for future use as reference, and into the two Reference Picture Lists (RPLs), *i.e.*, RPL0 and RPL1 [11]. To encode such "intermediate" view, four references are thus needed for inter-coding: *(i)* the central view, *(ii, iii)* two views that are used for interpolation, and *(iv)* the synthesized view. It should be noted that the synthesized view corresponds to a first level of quality in all the viewport layers, a second level of quality being obtained by transmitting a prediction residue.

B. Compression of SL_2

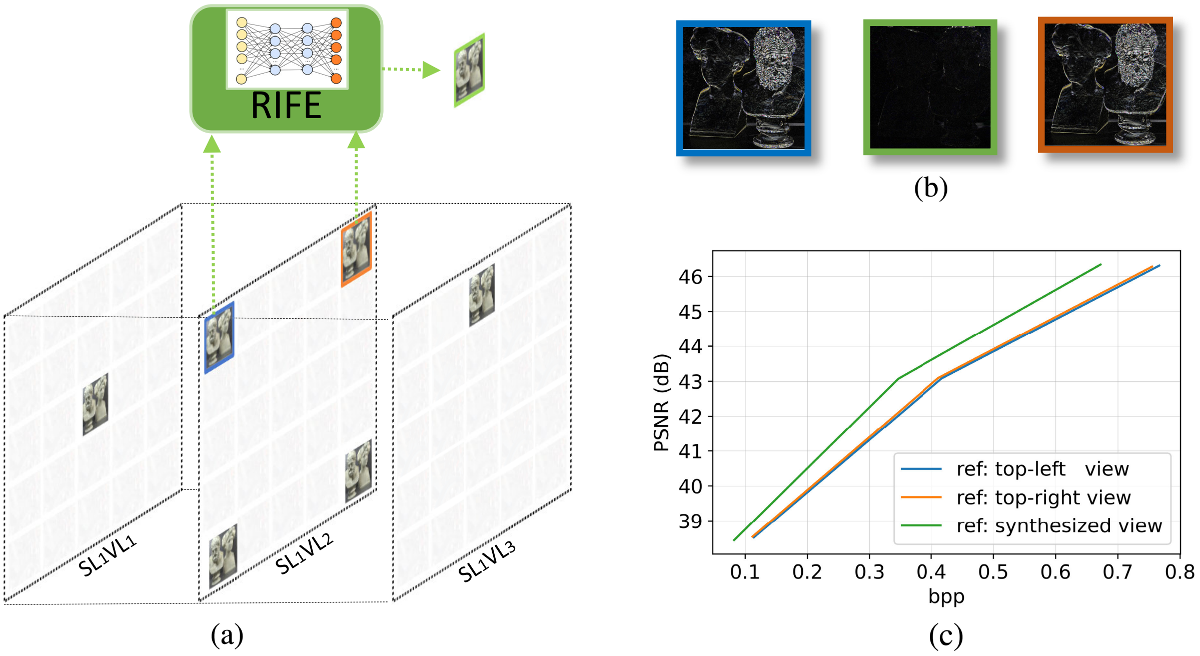
An upscaled view of SL_1 can be used as an additional reference to inter-code its corresponding view in SL_2. The views of the second spatial layer are encoded in a different way depending on the viewport layer to which they belong to. *(i)* SL_2VL_1: the central view of the second spatial layer is inter-coded using the upscaled central view in SL_1VL_1 as the reference image. *(ii)* SL_2VL_2: the views of the second viewport layer of the second spatial layer are encoded *independently* of each other but using inter-coding, taking *(a)* the central view in SL_2 and *(b)* the upscaled version of the co-located view in SL_1VL_2 as reference images. *(iii)* SL_2VL_m $(3 \leq m \leq n)$: three references are used for inter-coding views in SL_2VL_m: *(a)* the central view in SL_2, *(b)* the synthesized view, and *(c)* the upscaled version of the co-located view in SL_1.

The views in SL_2VL_m $(3 \leq m \leq n)$ are synthesized similar to the views in SL_1VL_m $(3 \leq m \leq n)$. That is, views in SL_2VL_1 to SL_2VL_{m-1} are used to synthesize those views which are equidistant from them in SL_2VL_m using RIFE.

To upscale images, DNN based super-resolution methods have shown a significant gain over the traditional methods. Some methods have been proposed specifically for light field super-resolution [47], [48], [49]. However, they typically use all or a set of low resolution light field views for the super-resolution task, which impairs the random access functionality (cf. Section V-F). To avoid this problem, we use a conventional single image super-resolution method in this paper, *i.e.*, DASR [50]. It should be noted that in SL_2, for the first quality level, intermediate views can be either *(i)* synthesized using a view interpolation method or *(ii)* reconstructed by applying a super-resolution approach to the co-located view in SL_1. To produce the second quality level, they are enhanced by adding the prediction residue to the above-mentioned reference images. Fig. 7 shows the encoding workflow for the top view in SL_2VL_3. The co-located view in SL_1, *i.e.*, the top view located in SL_1VL_3, is upscaled using DASR and it is added to the reference list. The central view in SL_2, *i.e.*, the view located in SL_2VL_1 is also added to the reference list. Finally, two views that the top view is equidistant from them, *i.e.*, the top-left and top-right views of the second viewport layer (SL_2VL_2), are used as inputs of RIFE, and the output of RIFE, *i.e.*, the synthesized view, is also added to the reference list. The top view is inter-coded and the prediction residue is added to the bitstream as the quality enhancement layer.

C. Bit Allocation and Quality Distribution

The bit allocation to different layers and views is flexible, allowing users to allocate bits in a way that meets their needs. In this paper, we allocate bits to provide *uniform quality distribution* among the views. To this end, we encode SL_1VL_1 with a base QP, and consider its quality as the reference quality (q_{c1}). We then empirically determine QPs for the views in SL_1VL_2 in a way that similar quality to the reference quality is achieved for views in SL_1VL_2, *i.e.*, $|q_{view} - q_{c1}| \leq \epsilon$, where ϵ is a threshold. When views in SL_1VL_m $(3 \leq m \leq n)$ are synthesized, the prediction residue is encoded if the quality of the synthesized view (*i.e.*, interpolated view) does not meet the uniform quality distribution criterion, *i.e.*, $|q_{view} - q_{c1}| \nleq \epsilon$. QP is empirically determined for the prediction residue to achieve $|q_{view} - q_{c1}| \leq \epsilon$. For SL_2VL_1, we consider the super-resolved image of SL_1VL_1 as the first quality level and we encode the prediction residue with the base QP to



(a) (b) (c)

Fig. 6. (a) The top-left and top-right views in VL_2 of spatial layer SL_1 are used as inputs to synthesize the top view in VL_3 of that layer. (b) The residual images between the ground truth top view in VL_3 and these three images are shown. The residual between the ground truth top view and the synthesized view has less information. (c) These views (*i.e.*, top-left, top-right, and synthesized views) are used as reference images in the reference list of the standard codec VVC to compress the top view in VL_3. The encoding efficiency of these three reference images shows a significant gain when the synthesized view is used as a reference image.

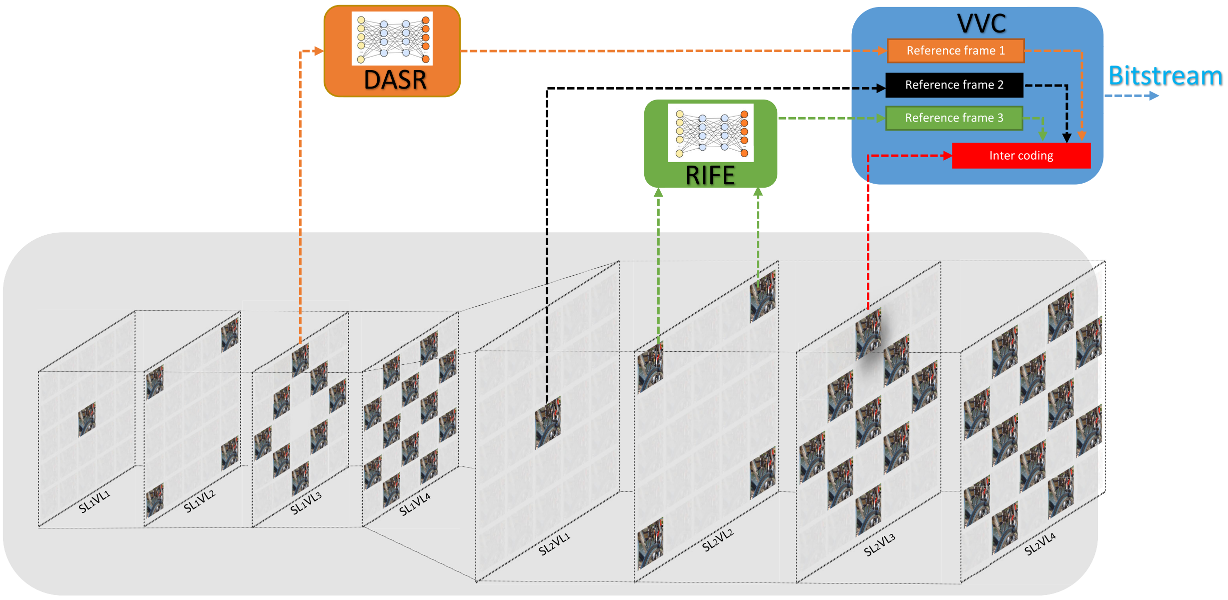


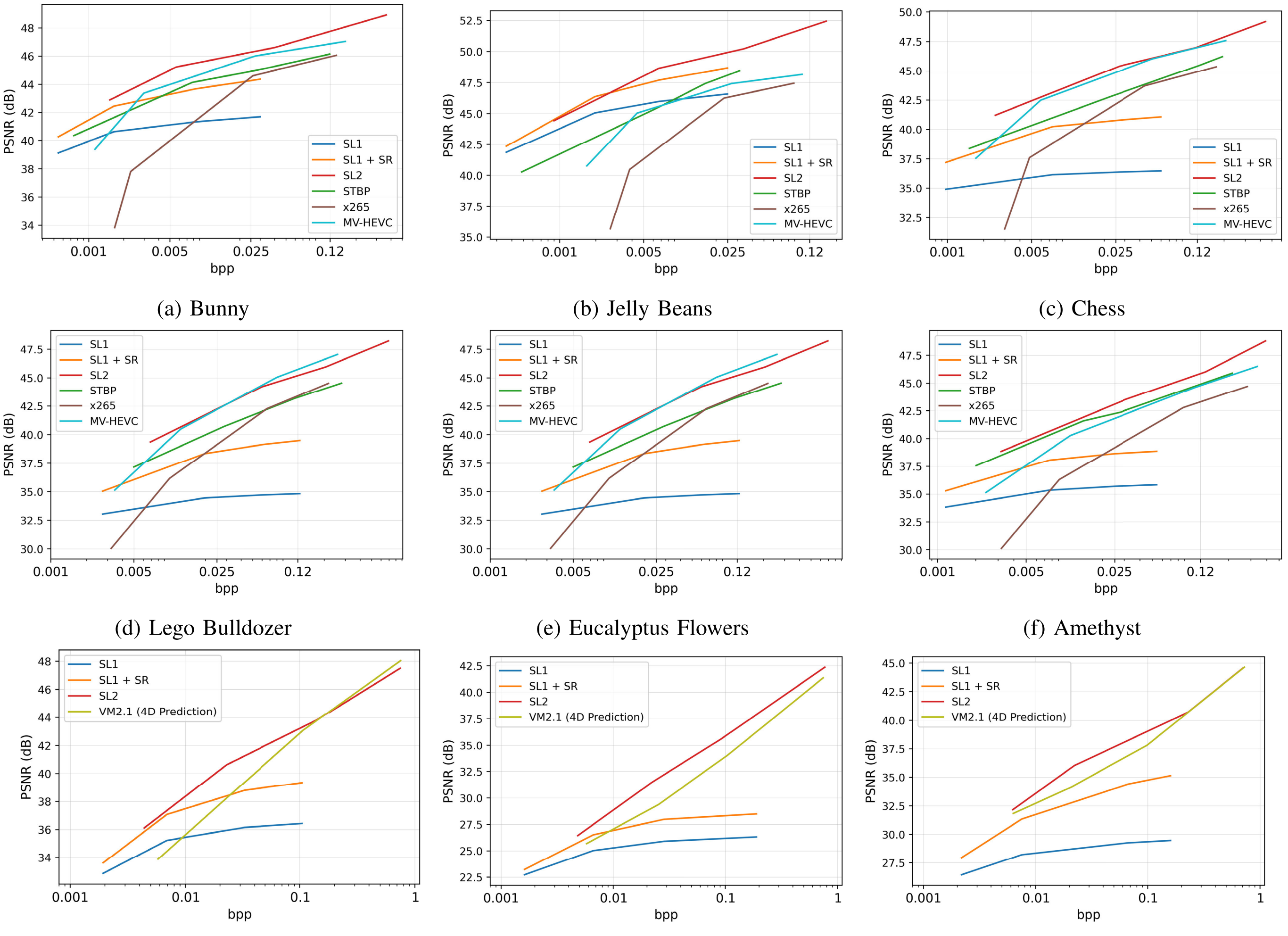
Fig. 7. Encoding workflow for the top view located in SL_2VL_3.

provide quality scalability for SL_2VL_1 and its final quality is referred to as q_{c2}. For the views in SL_2VL_2, we consider the super-resolved image of co-located views in SL_2VL_1 as the first quality level and we encode the prediction residue if $|q_{view} - q_{c2}| \nleq \epsilon$ by determining empirically QP to meet $|q_{view} - q_{c2}| \leq \epsilon$.

For views in SL_2VL_m $(3 \leq m \leq n)$, the reconstruction quality of the interpolated (synthesized) view and upscaled image by super-resolution is measured and their maximum value is calculated (q_{view}) for each view. q_{view} is then compared with the reconstructed quality of the central view (q_{c2}). If the difference between q_{view} and q_{c2} is not less than or equal to the threshold (ϵ), *i.e.*, $|q_{view} - q_{c2}| \nleq \epsilon$, the prediction residue is added to ensure $|q_{view} - q_{c2}| \leq \epsilon$ and consequently *uniform quality distribution* is guaranteed. Adding an enhancement layer is equivalent to providing quality scalability. Note that in this paper, the quality enhancement layer is not provided for views in SL_1VL_1 and SL_1VL_2, which

TABLE I
LIGHT FIELD TEST IMAGES TAKEN FROM THE STANFORD DATASET [39] AND JPEG PLENO DATASET [38]

Dataset	Name	Angular resolution	Spatial resolution	QP_1	QP_2	QP_3	QP_4
Stanford [39]	Bunny	17 × 17	1024 × 1024	30	22	17	15
	Jelly Beans	17 × 17	1024 × 512	31	17	15	13
	Chess	17 × 17	1400 × 800	31	18	16	15
	Lego Bulldozer	17 × 17	1536 × 1152	30	18	17	16
	Eucalyptus Flowers	17 × 17	1280 × 1536	30	18	17	16
	Amethyst	17 × 17	768 × 1024	30	18	17	16
JPEG Pleno [38]	Greek	9 × 9	512 × 512	35	25	18	15
	Sideboard	9 × 9	512 × 512	44	35	29	21
	Tarot	17 × 17	1024 × 1024	37	28	22	20



(a) Bunny (b) Jelly Beans (c) Chess

(d) Lego Bulldozer (e) Eucalyptus Flowers (f) Amethyst

(g) Greek (h) Sideboard (i) Tarot

Fig. 8. RD curves for light field test images of the Stanford dataset [39] and the JPEG Pleno dataset [38]. SL_1 represents the first spatial resolution after applying bicubic upsampling, $SL1 + SR$ represents the first spatial resolution after applying super-resolution, and SL_2 represents the compression efficiency of the overall proposed method.

can be provided depending on the user's need. Additionally, in this paper, for the views that the uniform quality distribution is satisfied with the interpolated or super-resolved images, the quality enhancement layer is not provided. However, the flexibility of the proposed method allows for a quality enhancement layer for all views according to the user's needs.

V. EXPERIMENTAL RESULTS

In this Section, we first introduce the test condition that we used in this paper. We then provide experimental results for compression efficiency and other functionalities that have been discussed in the previous sections.

A. Test Condition

To evaluate the performance of the proposed method, we have selected six light fields from the Stanford[1] dataset [39] and three light fields from the JPEG Pleno[2] dataset [38] to cover light fields from large to narrow parallaxes. The

[1] http://lightfield.stanford.edu/lfs.html; last access: Nov. 26, 2021

[2] http://plenodb.jpeg.org/lf/pleno_lf; last access: Nov. 26, 2021

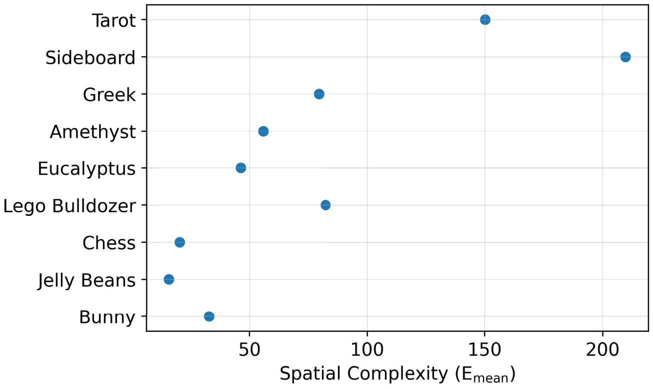


Fig. 9. Average spatial complexity of views (E_{mean}) for light field test images.

characteristics of these images are summarized in Table. I. The Stanford light field views were converted to 8-bits YUV420 format and the JPEG Pleno light field views were converted to 10-bits YUV444 format to match the coding conditions of the baseline codecs selected for comparison. VTM Encoder Version 10.2,[3] was used as the standard encoding software for VVC. We encode light fields at four quality levels. The base QPs used to encode each light field test image at four quality levels are also summarized in Table. I. QPoffsets for each viewport layer are selected in a way that the quality of encoded views remains similar to each other. In this paper, ϵ, was set to 1dB, which means that the quality difference of all views and the central view at each quality level is less than 1dB. For video interpolation, RIFE,[4] and for video super-resolution, DASR[5] were used without fine tuning.

B. Compression Efficiency and Quality Distribution

To evaluate the compression efficiency of the proposed method, we consider three points in its workflow: *(i)* SL_1: the compression efficiency of the first spatial resolution after applying the bicubic upsampling, *(ii)* $SL_1 + SR$: the compression efficiency of the first spatial resolution after applying super-resolution, and *(iii)* SL_2: the compression efficiency of the overall proposed method. We compare the encoding efficiency of these three points with the JPEG Pleno anchor (x265) [38], MV-HEVC [51], and Shearlet Transform Based Prediction (STBP) approach [21] for Stanford light fields, and with the JPEG Pleno Verification model 2.1 (4D Prediction) (VM2.1) [38] for JPEG Pleno light fields. Note that different baseline codecs have been selected for each dataset since they perform differently on each of them. VM2.1 performs well on the JPEG Pleno dataset, which mainly includes light fields with a narrow disparity. However, it does not perform well for large disparity light fields such as those of the Stanford dataset. On the other hand, STBP, which is based on MV-HEVC, provides limited compression efficiency for narrow disparity light fields [21]. Fig. 8 shows the RD curves using the mean PSNR of the Y component of all the views as the objective metric.

For the $Eucalyptus\ Flower$ light field, which has lots of fine geometry, the proposed method fails to outperform the state-of-the-art scheme. This might happen because of the inefficiency of video frame interpolation or super-resolution DNNs for these images or the lack of this type of image in their training dataset. For other light fields the proposed method (SL_2) shows superior performance compared to its competitors, particularly at lower bitrates. This is more significant for a light field with simple geometry such as $Jelly\ Beans$. The superiority of $SL_1 + SR$ to SL_1 shows the importance of super-resolution in improving the compression efficiency.

Note that the compression efficiency of SL_1 and $SL_1 + SR$ is low for some light fields such as $Sideboard$ and $Tarot$, while it is high for some light fields such as $Jelly\ Beans$. We have calculated the spatial complexity (E) for each light field view using Video Complexity Analyzer (VCA[6]) [52] and computed their average value (E_{mean}). The E_{mean} values for all test light fields are shown in Fig. 9. It is observed that, with increasing the spatial complexity, the compression efficiency is reduced.

C. Scalability

In this paper, to support spatial scalability, the light fields are compressed at two spatial resolutions. Therefore, the final bitstream consists of two parts: *(i)* b_{SL_1}: the bits allocated to compress the lowest resolution, and *(i)* b_{SL_2} the bits allocated to compress the highest resolution. The allocated bits to each spatial layer are also divided into multiple viewport layers (*i.e.*, $\{b_{VL_1}, \ldots, b_{VL_n}\}$) to support viewport scalability and uniform quality distribution. Finally, the allocated bits to each viewport layer are used to improve the quality of viewports in that layer, in other words, to support quality scalability. Fig. 10 shows the bits allocated to spatial and viewport layers the encoded $Bunny$ light field. It is observed that with increasing the number of encoding bits, the larger portion of the whole bitstream is allocated to SL_2. It is also observed that at the higher number of encoding bits, the smaller portion of each spatial resolution is allocated to the first viewport layer of each spatial layer, *i.e.*, SL_1VL_1 and SL_2VL_1, which have been differentiated from the other viewport layers in Fig. 10. To subjectively analyze the scalability of the proposed method, Fig. 11 shows the $Eucalyptus\ Flower$ light field when the whole light field is encoded at 0.04 bits per pixel (bpp). The central view of SL_1, before and after applying super-resolution, as well as the central view of SL_2 are compared with the original central view. It is shown how applying super-resolution and adding an enhancement layer improves the quality of the decoded central view.

D. Random Access

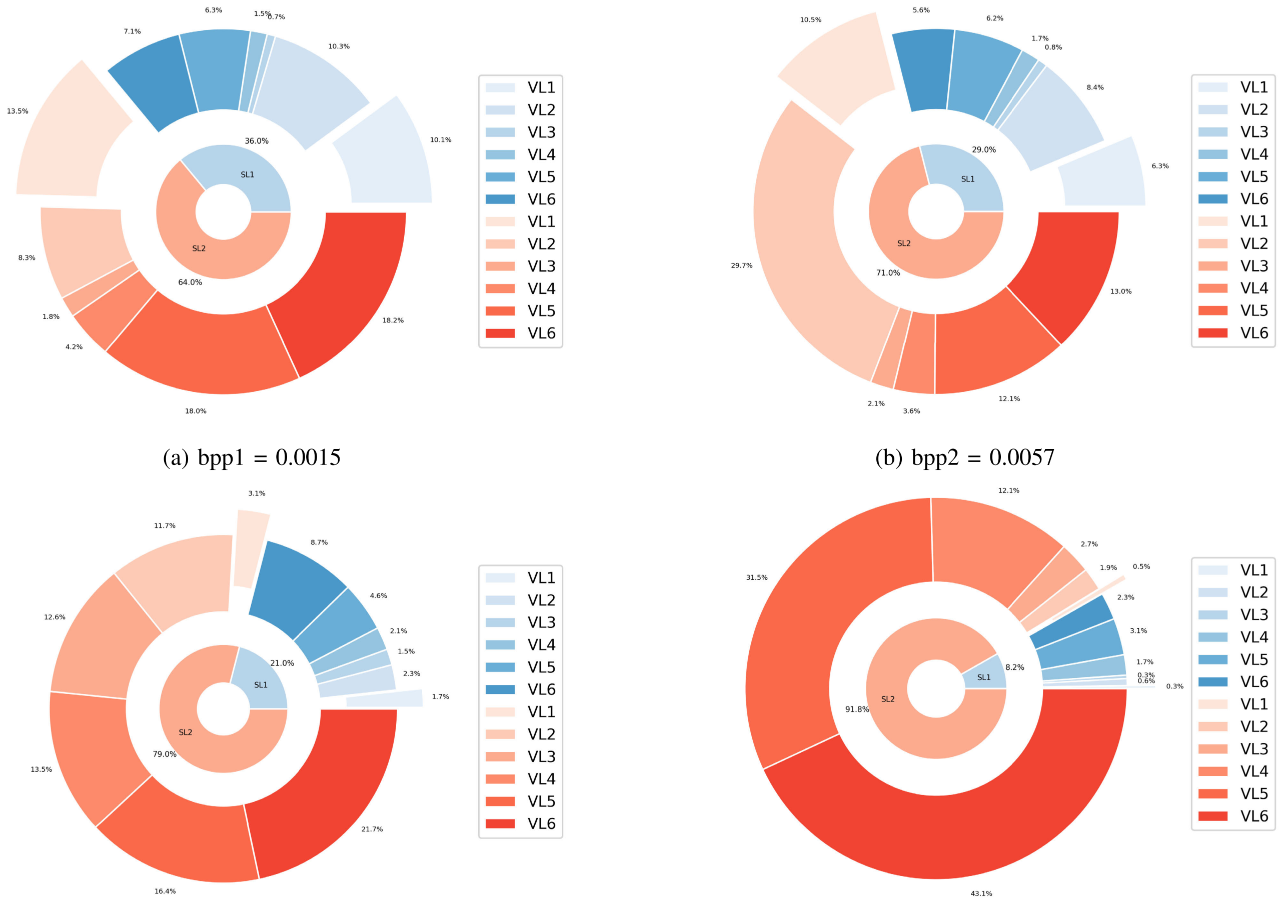
Random access to an arbitrary view decreases memory footprint and bandwidth requirements. The bitrates required

[3] https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM; last access: Nov. 26, 2021

[4] https://github.com/hzwer/arXiv2020-RIFE; last access: Nov. 26, 2021

[5] https://github.com/LongguangWang/DASR; last access: Nov. 26, 2021

[6] https://cd-athena.github.io/VCA/; last access: Jul. 20, 2022



(a) bpp1 = 0.0015

(b) bpp2 = 0.0057

(c) bpp3 = 0.0399

(d) bpp4 = 0.364

Fig. 10. Division of *Bunny* bitstream into spatial and viewport layers. The blue portions represent the portion of bits allocated to SL_1, and the red portions represent the portion of bits allocated to SL_2. With increasing the number of encoding bits, the larger portion of the whole bitstream is allocated to SL_2.

to access views and their maximum (RA_p) are shown in Fig. 12. RRA_p is also shown in Fig. 12 as embedded plots. It is seen that at the higher number of encoding bits, where random access is crucial, only a small portion of the whole bitstream is required to access an arbitrary view. Note that the *flexibility* of the proposed method allows to address the trade-off between the compression efficiency and random access. For instance, if the synthesized views are removed from the reference list and only the super-resolved images are used as virtual reference images to encode views in SL_2, random access is improved while the compression efficiency is reduced. It should be mentioned that the baseline codecs, *i.e.*, JPEG Pleno anchor (x265), MV-HEVC, STBP, and VM2.1 (4D Prediction) show low random access performance since they are highly dependent on the inter-view prediction between the different views. JPEG Pleno anchor (x265) converts all views into a single PVS and encodes them sequentially; thus, it does not provide random access to views. Similarly, in STBP, the prediction residuals of all views are converted to a PVS and compressed with a video encoder, which makes all views dependent on each other and significantly impairs the random access performance. VM2.1 (4D Prediction), which is based on WaSP, is also highly dependent on the amount of reference views that are warped and merged using one optimal least-squares merger. Fig. 13 compares the performance of RRA_P of the proposed method with the one of MV-HEVC for the *Bunny* light field. It is shown that the proposed method achieves a better random access performance compared to MV-HEVC. The superiority is more significant at higher number of encoding bits, where random access is more crucial.

E. Error Resiliency

Compressed data is always vulnerable to channel errors and bandwidth constraints. However, our proposed method can synthesize all views even with a small portion of the whole bitstream, *i.e.*, $b_{SL_1VL_1}$ and $b_{SL_1VL_2}$. When corner views are available in the first spatial layer, all other views can be synthesized and super resolved to generate the whole image views but at a lower quality. For example, as shown in Fig. 10, at $bpp3$, with only $b_{SL_1VL_1} + b_{SL_1VL_2} = 1.7\% + 2.3\% = 4\%$ of the whole bitstream, all other views can be synthesized. To show how much quality improvement can be achieved by additionally downloading each layer (and loosing next layers), we plot quality vs. downloaded bits for the *Bunny* light field in Fig. 14. It is seen that the proposed method is resilient to channel errors and can retrieve image views even when a significant portion of a bitstream is lost.

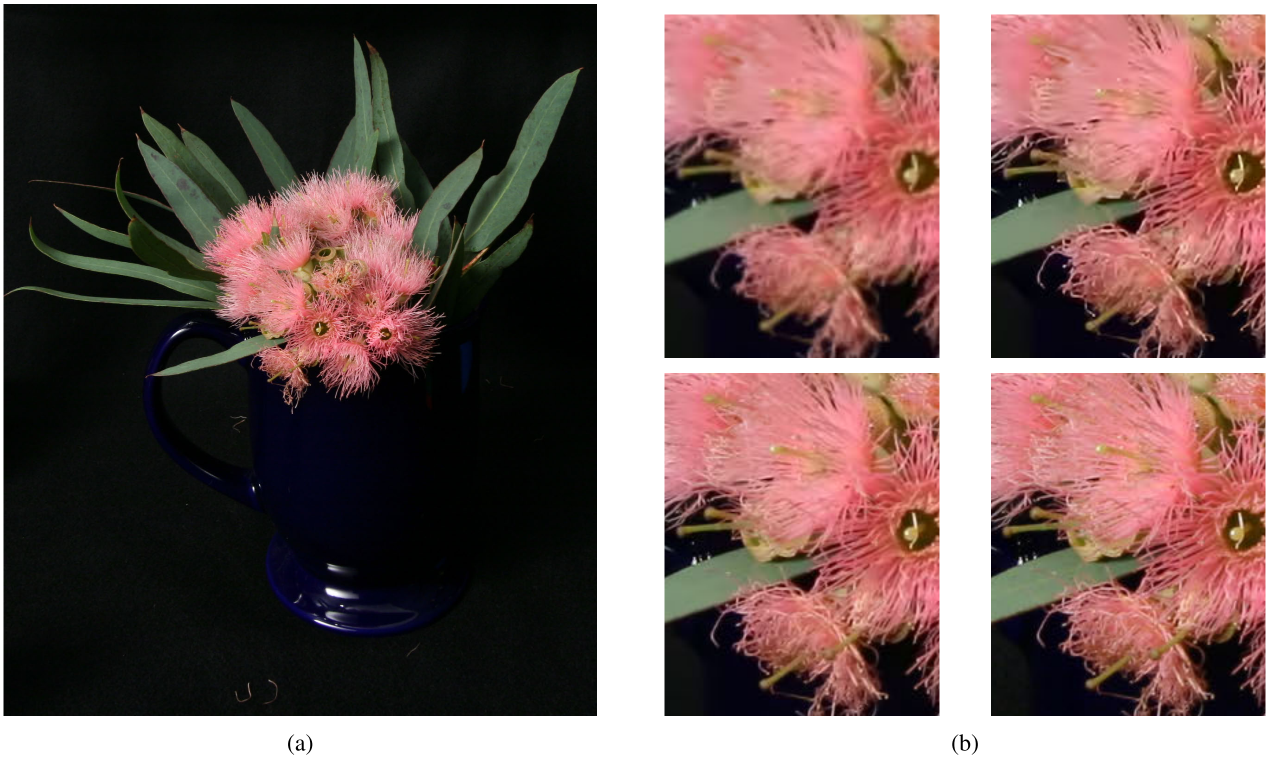


Fig. 11. Subjective evaluation of the scalability of the proposed method for the *Eucalyptus Flowers* test image when the whole test image is encoded at 0.04 bpp. (a) central view, (b) [top-left] SL_1, (b) [top-right] $SL_1 + SR$, (b) [bottom-left] SL_2, (b) [bottom-right] Original image.

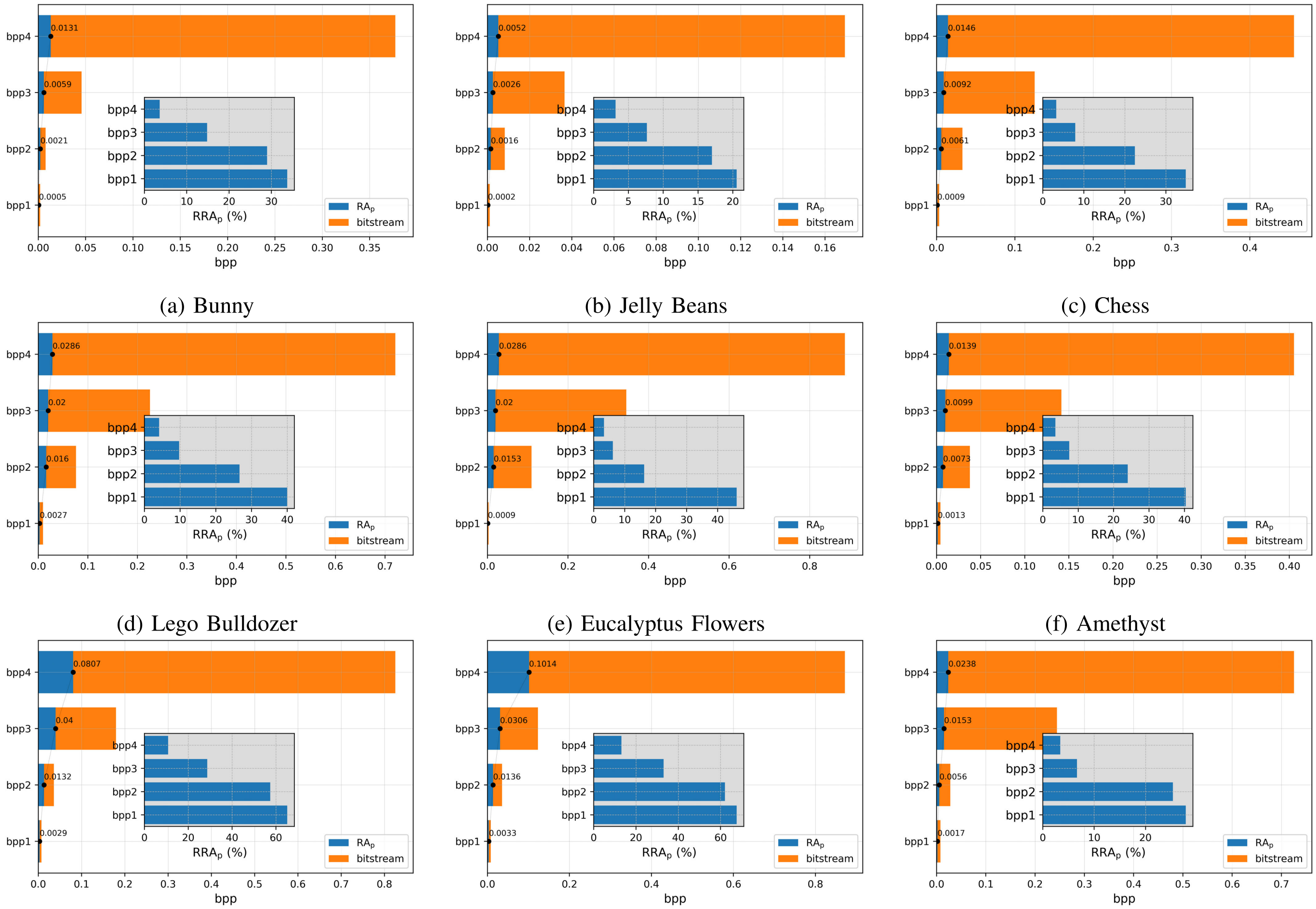
F. Flexibility

Due to its high flexibility, the proposed approach can address different trade-offs including compression efficiency, random access, uniform quality distribution, and error resiliency with adaptive bit allocation to different layers. In this paper, the bits were empirically allocated among different layers in a way that they yield image views with similar qualities. For example, Fig.15a shows the standard deviation for PSNR of views of the *Bunny* light field for SL_1, $SL_1{+}SR$, and SL_2 points. The scatter plot for the absolute difference between PSNR of each view and PSNR of the central view (for SL_2) is also shown in Fig.15b to validate the uniform quality distribution. It is seen that the criterion of $|q_{view} - q_c| < 1$dB for all views has been met. However, the bits can be allocated in a way to yield a higher compression efficiency or random access performance.

RIFE is capable of interpolating intermediate viewports without any limits on the maximum number of interpolated views at the same inference time. In this paper, it is used to interpolate only one intermediate view, *i.e.*, equidistant intermediate views. However, with interpolating more than one intermediate view, each viewport layer may contain more views, and the number of viewport layers and the inference time for interpolation may be reduced. This will allow us to have flexibility in the number of viewport layers (n). For example, we encode only the first and second viewport layers in the first spatial layer (*i.e.*, SL_1VL_1 and SL_1VL_2), and we then use corner views in SL_1VL_2 as inputs of RIFE to interpolate all intermediate views between the corner views without adding any quality enhancement layer. In this way, we need to run RIFE at most thrice to access any arbitrary view in SL_1 and additionally DASR once to access any arbitrary view in SL_2 without any need to encode/decode any enhancement layer (See Fig. 16a). Note that in this structure, the number of viewport layers (*i.e.*, four viewport layers for SL_1 and one viewport for SL_2) is independent of the light field's angular resolution. The compression efficiency of the above-mentioned structure ($SL_1 + SR$ (2)) for the *Bunny* light field is shown in Fig. 16b. It is seen that this structure shows lower performance in terms of compression efficiency; however, it results in fast access to any arbitrary view. Additionally, since the quality enhancement layer is not applied to views, the average standard deviation of PSNR of views for all quality levels is increased from 0.24 for SL_2 to 1.04 for $SL1 + SR(2)$.

Light field super-resolution (LFSR) approaches [53], [54], [55] may result in views with higher reconstruction quality compared to single image super-resolution (SISR) approaches [50], [56] since they better preserve angular consistency. However, it should be noted that LFSR approaches usually utilize all or a huge set of low resolution views as inputs to super resolve all of them, which harms the random access performance and viewport scalability. To evaluate the impact of super-resolution on the performance of the proposed method, we take the 5×5 central views of the *Tarot* light field and encode SL_1 with the proposed method (Section IV-A). For super-resolution, we select EDSR [56] as an SISR approach and LFT [55] as an LFSR approach from BasicLFSR,[7] an

[7] https://github.com/ZhengyuLiang24/BasicLFSR



(a) Bunny (b) Jelly Beans (c) Chess

(d) Lego Bulldozer (e) Eucalyptus Flowers (f) Amethyst

(g) Greek (h) Sideboard (i) Tarot

Fig. 12. RA_p and RRA_p for light field test images. RA_p denotes the maximum number of encoded bits requited to access an arbitrary view at each bitstream. The embedded plots represent RRA_p, *i.e.*, the relative number of encoded bits required to access an arbitrary view at each bitstream.

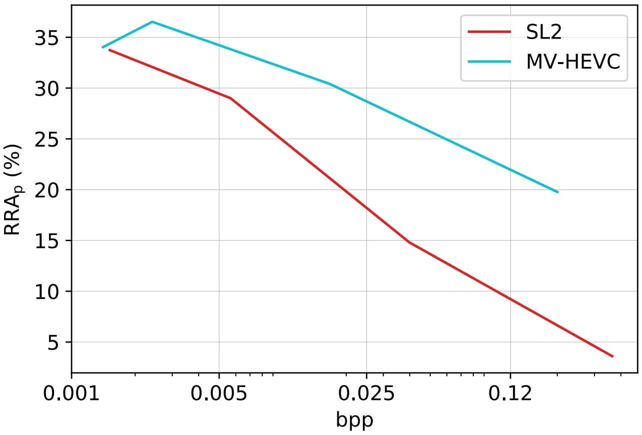


Fig. 13. RRA_P of the proposed method (SL_2) and MV-HEVC for the *Bunny* light field. The lower is RRA_p, the better is the random access performance.

open-source light field super-resolution toolbox. EDSR and LFT have been selected since they have been both trained with the same light fields allowing a fair comparison. We super resolve views using EDSR and encode views in SL_2 using the proposed method (Section IV-B). When EDSR is replaced with LFT, all views in $SL1$ are used as inputs of LFT and the output of LFT will be all views that have been super resolved.

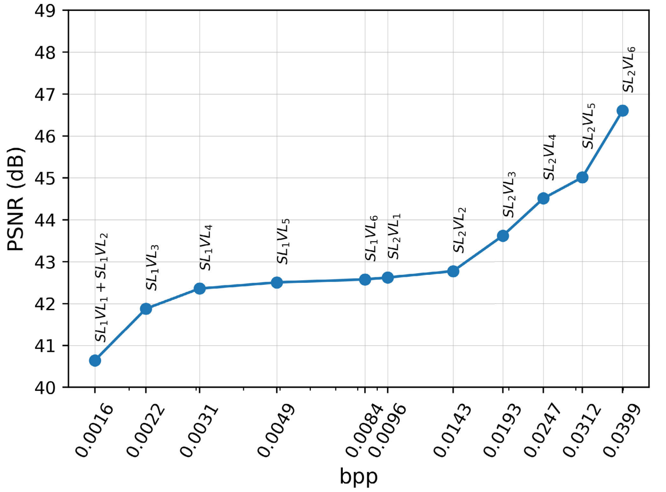
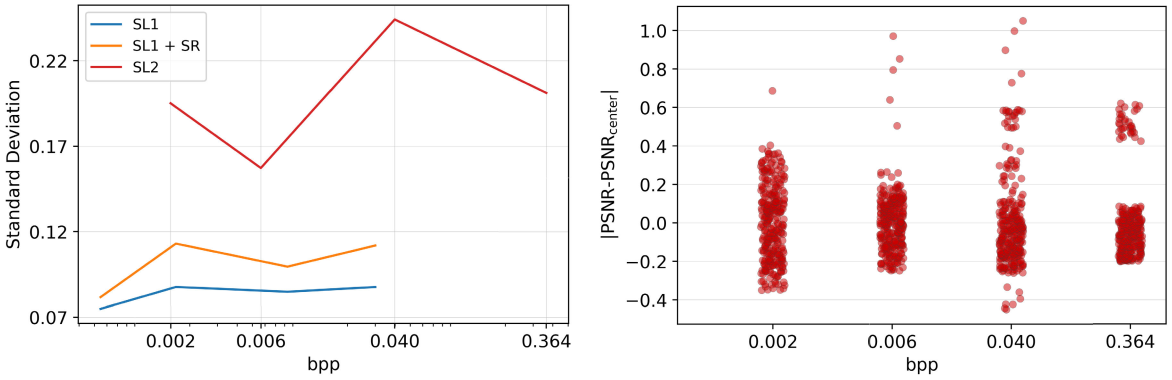


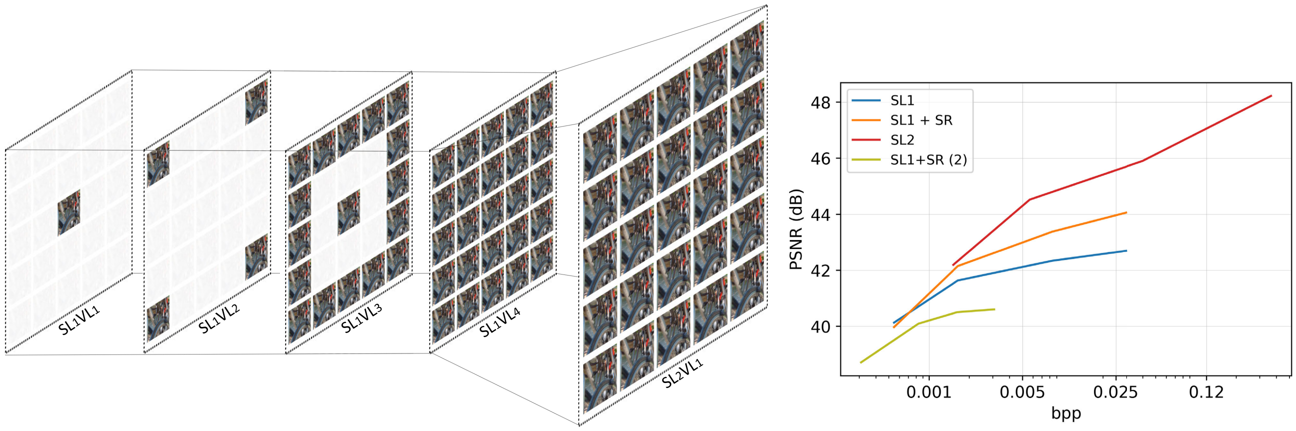
Fig. 14. PSNR vs. downloaded bits for the *Bunny* light field. The more the layers are downloaded, the higher is the quality. In this way, error resiliency is achieved in the case of channel errors and/or bandwidth constraints.

Therefore, SL_2 comprises only one viewport layer with LFT approach. We show the compression efficiency and random access performance of both methods in Fig. 17. It is seen that utilizing the LFSR approach for super-resolution improves the



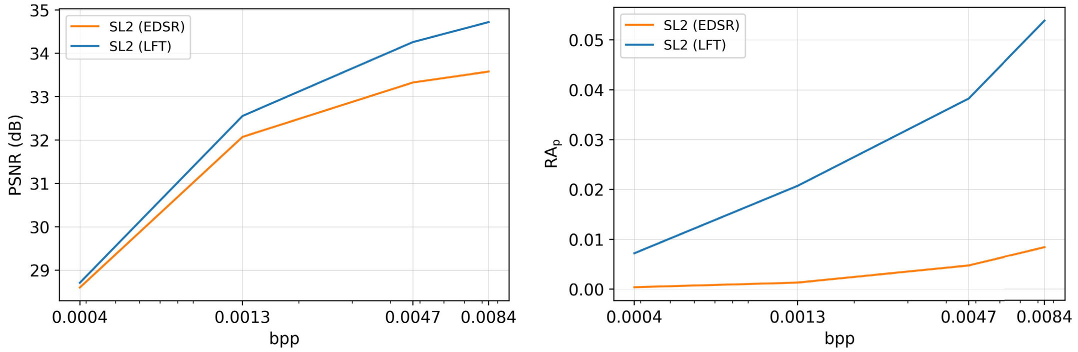
(a) (b)

Fig. 15. (a) The standard deviation of the quality of views for the *Bunny* light field. (b) The absolute quality difference between all views and the central view in SL_2.



(a) (b)

Fig. 16. Corner views in the first spatial layer are used as inputs of RIFE to interpolate all intermediate views between the corner views without adding any quality enhancement layer. In this way, RIFE is run at most thrice to access any arbitrary view in SL_1. Additionally, DASR is run once to access any arbitrary view in SL_2 without any need to encode/decode any enhancement layer. (a) The example structure for a 5 × 5 light filed. (b) The compression efficiency of the *Bunny* light field using this structure.



(a) (b)

Fig. 17. Comparison between the use of EDSR (SISR) and LFT (LFSR) approaches on the performance of the proposed method in terms of (a) compression efficiency and (b) random access.

compression efficiency at the cost of reduced random access performance.

G. Future Directions

RIFE has been trained for video frame interpolation and its training for light field view synthesis may improve its efficiency for the view synthesis. Both RIFE and DASR have been trained with uncompressed images but we deploy them to interpolate and super resolve compressed images. Fine tuning these DNNs with compressed images may improve their accuracy.

VI. Conclusion

In this paper, we propose a novel light field compression method based on video interpolation and image super-resolution techniques. Light field views are compressed in two spatial layers to support spatial scalability. Views at each spatial layer are divided into various viewport layers. The previously encoded views are used to synthesize their equidistant intermediate views and the synthesized views are then used as virtual reference frames to inter-code the intermediate views and improve their quality. A super-resolution method is applied to the compressed views at the lowest resolution and they are used as additional reference images to inter-code their corresponding views at the highest resolution. In addition to the spatial, viewport, and quality scalabilities, the proposed structure improves the flexibility of light field compression, provides random access to the viewports, and increases error resiliency.

Acknowledgment

The financial support of the Austrian Federal Ministry for Digital and Economic Affairs, the National Foundation for Research, Technology and Development, and the Christian Doppler Research Association are gratefully acknowledged. Christian Doppler Laboratory ATHENA: https://athena.itec.aau.at/.

References

[1] G. Wu et al., “Light field image processing: An overview,” *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 7, pp. 926–954, Oct. 2017.

[2] C. Conti, L. D. Soares, and P. Nunes, “Dense light field coding: A survey,” *IEEE Access*, vol. 8, pp. 49244–49284, 2020.

[3] B. M. D. Carvalho et al., “A 4D DCT-based lenslet light field codec,” in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 435–439.

[4] A. Aggoun, “Compression of 3D integral images using 3D wavelet transform,” *J. Display Technol.*, vol. 7, no. 11, pp. 586–592, Nov. 2011.

[5] H.-H. Kang, D.-H. Shin, and E.-S. Kim, “Compression scheme of sub-images using Karhunen-Loeve transform in three-dimensional integral imaging,” *Opt. Commun.*, vol. 281, no. 14, pp. 3640–3647, Jul. 2008.

[6] V. Elias and W. Martins, “On the use of graph Fourier transform for light-field compression,” *J. Commun. Inf. Syst.*, vol. 33, no. 1, pp. 92–103, 2018.

[7] M. Rizkallah, T. Maugey, and C. Guillemot, “Rate-distortion optimized graph coarsening and partitioning for light field coding,” *IEEE Trans. Image Process.*, vol. 30, pp. 5518–5532, 2021.

[8] C. Conti, P. Nunes, and L. D. Soares, “HEVC-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.

[9] R. Monteiro et al., “Light field HEVC-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.

[10] J. G. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, “Overview of the high efficiency video coding (HEVC) standard,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

[11] B. Bross, J. Chen, J.-R. Ohm, J. G. Sullivan, and Y.-K. Wang, “Developments in international video coding standardization after AVC, with an overview of versatile video coding (VVC),” *Proc. IEEE*, vol. 109, no. 9, pp. 1463–1493, Sep. 2021.

[12] F. Dai, J. Zhang, Y. Ma, and Y. Zhang, “Lenselet image compression scheme based on subaperture images streaming,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2015, pp. 4733–4737.

[13] A. Vieira, H. Duarte, C. Perra, L. Tavora, and P. Assuncao, “Data formats for high efficiency coding of Lytro-Illum light fields,” in *Proc. Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Nov. 2015, pp. 494–497.

[14] G. Wang, W. Xiang, M. Pickering, and C. W. Chen, “Light field multi-view video coding with two-directional parallel inter-view prediction,” *IEEE Trans. Image Process.*, vol. 25, no. 11, pp. 5104–5117, Nov. 2016.

[15] P. Astola and I. Tabus, “WaSP: Hierarchical warping, merging, and sparse prediction for light field image compression,” in *Proc. 7th Eur. Workshop Vis. Inf. Process. (EUVIP)*, Nov. 2018, pp. 1–6.

[16] X. Jiang, M. L. Pendu, and C. Guillemot, “Light field compression using depth image based view synthesis,” in *Proc. IEEE Int. Conf. Multimedia Expo Workshops (ICMEW)*, Jul. 2017, pp. 19–24.

[17] W. Ahmad, S. Vagharshakyan, M. Sjostrom, A. Gotchev, R. Bregovic, and R. Olsson, “Shearlet transform based prediction scheme for light field compression,” in *Proc. Data Compress. Conf.*, Mar. 2018, p. 396.

[18] J. Hou, J. Chen, and L.-P. Chau, “Light field image compression based on Bi-level view compensation with rate-distortion optimization,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 2, pp. 517–530, Feb. 2019.

[19] X. Hu, J. Shan, Y. Liu, L. Zhang, and S. Shirmohammadi, “An adaptive two-layer light field compression scheme using GNN-based reconstruction,” *ACM Trans. Multimedia Comput., Commun., Appl.*, vol. 16, no. 2, pp. 1–23, Apr. 2020.

[20] E. Dib, M. L. Pendu, and C. Guillemot, “Light field compression using Fourier disparity layers,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2019, pp. 3751–3755.

[21] W. Ahmad, S. Vagharshakyan, M. Sjostrom, A. Gotchev, R. Bregovic, and R. Olsson, “Shearlet transform-based light field compression under low bitrates,” *IEEE Trans. Image Process.*, vol. 29, pp. 4269–4280, 2020.

[22] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, “Learning-based view synthesis for light field cameras,” *ACM Trans. Graph.*, vol. 35, no. 6, p. 193, 2016.

[23] C. Jia, X. Zhang, S. Wang, S. Wang, and S. Ma, “Light field image compression using generative adversarial network-based view synthesis,” *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 1, pp. 177–189, Mar. 2019.

[24] N. Bakir, W. Hamidouche, S. A. Fezza, K. Samrouth, and O. Deforges, “Light field image coding using VVC standard and view synthesis based on dual discriminator GAN,” *IEEE Trans. Multimedia*, vol. 23, pp. 2972–2985, 2021.

[25] C. Conti, P. Nunes, and L. D. Soares, “Inter-layer prediction scheme for scalable 3-D holoscopic video coding,” *IEEE Signal Process. Lett.*, vol. 20, no. 8, pp. 819–822, Aug. 2013.

[26] Y. Li, M. Sjöström, R. Olsson, and U. Jennehag, “Scalable coding of plenoptic images by using a sparse set and disparities,” *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 80–91, Jan. 2016.

[27] J. Garrote, C. Brites, J. Ascenso, and F. Pereira, “Lenslet light field imaging scalable coding,” in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 2150–2154.

[28] C. Conti, L. D. Soares, and P. Nunes, “Scalable light field coding with support for region of interest enhancement,” in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 1855–1859.

[29] C. Conti, L. D. Soares, and P. Nunes, “Light field coding with field-of-view scalability and exemplar-based interlayer prediction,” *IEEE Trans. Multimedi*, vol. 20, no. 11, pp. 2905–2920, 2018.

[30] K. Komatsu, K. Takahashi, and T. Fujii, “Scalable light field coding using weighted binary images,” in *Proc. 25th IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2018, pp. 903–907.

[31] D. Rüfenacht, A. T. Naman, R. Mathew, and D. Taubman, “Base-anchored model for highly scalable and accessible compression of multiview imagery,” *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3205–3218, Jul. 2019.

[32] H. Amirpour, C. Timmerer, and M. Ghanbari, “SLFC: Scalable light field coding,” in *Proc. Data Compress. Conf. (DCC)*, Mar. 2021, pp. 43–52.

[33] E. H. Adelson and J. R. Bergen, “Light fields and computational imaging,” in *Computational Models of Visual Processing*. Cambridge, MA, USA: MIT Press, 1991.

[34] D. G. Dansereau, O. Pizarro, and S. B. Williams, “Decoding, calibration and rectification for lenselet-based plenoptic cameras,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1027–1034.

[35] R. J. S. Monteiro, “Scalable light field representation and coding,” Instituto Universitário de Lisboa (ISCTE), Portugal, Tech. Rep., 2020. [Online]. Available: http://hdl.handle.net/10071/20584

[36] R. J. S. Monteiro, N. M. M. Rodrigues, S. M. M. Faria, and P. J. L. Nunes, “Light field image coding with flexible viewpoint scalability and random access,” *Signal Process., Image Commun.*, vol. 94, May 2021, Art. no. 116202.

[37] H. Amirpour, A. Pinheiro, M. Pereira, F. J. P. Lopes, and M. Ghanbari, “Light field image compression with random access,” in *Proc. Data Compress. Conf. (DCC)*, Mar. 2019, p. 553.

[38] G. Alves, C. L. Pagliari, and P. G. Freitas, *JPEG Pleno Light Field Coding Common Test Conditions V3. 2*, document ISO /IEC JTC 1/SC 29 /WG 1 N83029, 2019.

[39] M. Magnor and B. Girod, “Data compression for light-field rendering,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 3, pp. 338–343, Apr. 2000.

[40] S. Pratapa and D. Manocha, “RLFC: Random access light field compression using key views,” 2018, *arXiv:1805.06019*.

[41] S. Pratapa and D. Manocha, “HMLFC: Hierarchical motion-compensated light field compression for interactive rendering,” *Comput. Graph. Forum*, vol. 38, no. 8, pp. 1–12, Nov. 2019.

[42] C. Zhang and J. Li, “Compression of lumigraph with multiple reference frame (MRF) prediction and just-in-time rendering,” in *Proc. DCC. Data Compress. Conf.*, 2000, pp. 253–262.

[43] P. Gomes and L. A. D. S. Cruz, “Pseudo-sequence light field image scalable encoding with improved random access,” in *Proc. 8th Eur. Workshop Vis. Inf. Process. (EUVIP)*, Oct. 2019, pp. 16–21.

[44] S. Niklaus, L. Mai, and F. Liu, “Video frame interpolation via adaptive convolution,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 670–679.

[45] S. Niklaus, L. Mai, and F. Liu, “Video frame interpolation via adaptive separable convolution,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 261–270.

[46] Z. Huang, T. Zhang, W. Heng, B. Shi, and S. Zhou, “RIFE: Real-time intermediate flow estimation for video frame interpolation,” 2020, *arXiv:2011.06294*.

[47] N. Meng, H. K.-H. So, X. Sun, and E. Y. Lam, “High-dimensional dense residual convolutional neural network for light field reconstruction,” *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 43, no. 3, pp. 873–886, Mar. 2021.

[48] J. Jin, J. Hou, J. Chen, and S. Kwong, “Light field spatial super-resolution via deep combinatorial geometry embedding and structural consistency regularization,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2257–2266.

[49] S. Zhang, Y. Lin, and H. Sheng, “Residual networks for light field image super-resolution,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11038–11047.

[50] L. Wang et al., “Unsupervised degradation representation learning for blind super-resolution,” in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 10581–10590.

[51] 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.

[52] V. V. Menon, C. Feldmann, H. Amirpour, M. Ghanbari, and C. Timmerer, “VCA: Video complexity analyzer,” in *Proc. 13th ACM Multimedia Syst. Conf. (MMSys)*, New York, NY, USA, 2022, pp. 259–264.

[53] Y. Wang et al., “Light field image super-resolution using deformable convolution,” *IEEE Trans. Image Process.*, vol. 30, pp. 1057–1071, 2021.

[54] Y. Wang, F. Liu, K. Zhang, G. Hou, Z. Sun, and T. Tan, “LFNet: A novel bidirectional recurrent convolutional neural network for light-field image super-resolution,” *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4274–4286, Sep. 2018.

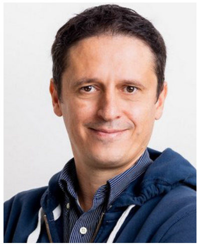
[55] Z. Liang, Y. Wang, L. Wang, J. Yang, and S. Zhou, “Light field image super-resolution with transformers,” *IEEE Signal Process. Lett.*, vol. 29, pp. 563–567, 2022.

[56] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, “Enhanced deep residual networks for single image super-resolution,” in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 136–144.

Hadi Amirpour (Member, IEEE) received the two B.Sc. degree in electrical and biomedical engineering and the M.Sc. degree in electrical engineering from the K. N. Toosi University of Technology and the Ph.D. degree in computer science from the University of Klagenfurt in 2022. He is a currently a Postdoctoral Research Fellow with the Christian Doppler (CD) Laboratory ATHENA, University of Klagenfurt. He was involved in the Project EmergIMG, a Portuguese consortium on emerging imaging technologies, funded by the Portuguese funding agency and H2020. His research interests include: video streaming, image and video compression, quality of experience, emerging 3D imaging technology, and medical image analysis. Further information at https://hadiamirpour.github.io.

Christine Guillemot (Fellow, IEEE) received the Ph.D. degree from Ecole Nationale Superieure des Telecommunications (ENST), Paris, and the Habilitation degree in research direction from the University of Rennes. From 1985 to October 1997, she was at FRANCE TELECOM, where she has been involved in various projects in the areas of image and video coding and processing for TV, HDTV, and multimedia. From January 1990 to mid 1991, she was worked at the Bellcore, NJ, USA, as a Visiting Scientist. She is currently the Director of Research with INRIA. Her research interests include: signal and image processing and computer vision. She has served as a Senior Member of the Editorial Board of the IEEE Journal of Selected Topics in Signal Processing (2013–2015) and has been a Senior Area Editor of IEEE Transactions on Image Processing (2016–2020). She was served as an Associate Editor for IEEE Transactions on Image Processing (from 2000 to 2003, as well as 2014–2016), for IEEE Transactions on Circuits and Systems for Video Technology (from 2004 to 2006), and for IEEE Transactions on Signal Processing (2007–2009).

Mohammad Ghanbari (Life Fellow, IEEE) is currently an Emeritus Professor with the School of Computer Science and Electronic Engineering, University of Essex, U.K. He is currently involved in the Athena Project with the Universitat Klagenfurt, Austria. He is internationally best known for the pioneering work on layered video coding, which earned him IEEE Fellowship in 2001 and he was also promoted IEEE Life Fellow in 2014. He has registered for thirteen international patents and published more than 800 technical articles on various aspects of video networking, many of which have had fundamental influences in this field. These include: video/image compression, layered/scalable video coding, video transcoding, motion estimation, and video quality metrics. He is the author and coauthor of eight books, and his book *Video coding: an introduction to standard codecs* (IET Press, 1999). In 2021, he was elected a fellow of Asia-Pacific Artificial Intelligence Association (FAAIA). He received the Rayleigh Prize as the best book in 2000 by IET. He was the General Chair of 1997 Packet Video Work Shop. He was one of the founding Associate Editors of IEEE Transactions on Multimedia from 1998 to 2002.



Christian Timmerer (Senior Member, IEEE) is currently an Associate Professor with the Institute of Information Technology (ITEC) and also the Director of the Christian Doppler (CD) Laboratory ATHENA (https://athena.itec.aau.at/). His research interests include: immersive multimedia communication, streaming, adaptation, and quality of experience, where he has coauthored seven patents and more than 200 articles. He was the General Chair of WIAMIS 2008, QoMEX 2013, MMSys 2016, and PV 2018, and has participated in several EC-Funded projects, notably DANAE, ENTHRONE, P2P-Next, ALICANTE, SocialSensor, COST IC1003 QUALINET, and ICoSOLE. He also participated in ISO/MPEG work for several years, notably in the area of MPEG-21, MPEG-M, MPEG-V, and MPEG-DASH, where he also served as a Standard Editor. In 2013, he was cofounded Bitmovin (http://www.bitmovin.com/) to provide professional services around MPEG-DASH where he holds the position of the chief innovation officer (CIO)—head of research and standardization. Further information at http://timmerer.com.