



## Article

# Novel Projection Schemes for Graph-Based Light Field Coding

Nguyen Gia Bach <sup>1</sup>, Chanh Minh Tran <sup>1</sup>, Tho Nguyen Duc <sup>1</sup>, Phan Xuan Tan <sup>2,\*</sup> and Eiji Kamioka <sup>1</sup>

<sup>1</sup> Graduate School of Engineering and Science, Shibaura Institute of Technology, Tokyo 135-8548, Japan; mg21501@shibaura-it.ac.jp (N.G.B.); nb20502@shibaura-it.ac.jp (C.M.T.); nb20501@shibaura-it.ac.jp (T.N.D.); kamioka@shibaura-it.ac.jp (E.K.)

<sup>2</sup> Department of Information and Communications Engineering, Shibaura Institute of Technology, Tokyo 135-8548, Japan

\* Correspondence: tanpx@shibaura-it.ac.jp

**Abstract:** In light field compression, graph-based coding is powerful to exploit signal redundancy along irregular shapes and obtains good energy compaction. However, apart from high time complexity to process high dimensional graphs, their graph construction method is highly sensitive to the accuracy of disparity information between viewpoints. In real-world light field or synthetic light field generated by computer software, the use of disparity information for super-rays projection might suffer from inaccuracy due to vignetting effect and large disparity between views in the two types of light fields, respectively. This paper introduces two novel projection schemes resulting in less error in disparity information, in which one projection scheme can also significantly reduce computation time for both encoder and decoder. Experimental results show projection quality of super-pixels across views can be considerably enhanced using the proposals, along with rate-distortion performance when compared against original projection scheme and HEVC-based or JPEG Pleno-based coding approaches.

**Keywords:** light field; compression; super-rays; graph transform; super-ray projection



**Citation:** Bach, N.G.; Tran, C.M.; Duc, T.N.; Tan, P.X.; Kamioka, E. Novel Projection Schemes for Graph-Based Light Field Coding. *Sensors* **2022**, *22*, 4948. <https://doi.org/10.3390/s22134948>

Academic Editor: Yitzhak Yitzhak

Received: 2 June 2022

Accepted: 26 June 2022

Published: 30 June 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Light field (LF) is an emerging technology in multimedia research areas that allows capturing different light rays in many directions, emitted from every point of an object or a scene [1]. Hence, it brings significantly improved immersiveness, depth, intensity, color and perspectives from a range of viewpoints. As the result, it reveals promising application opportunities into vast areas such as Virtual Reality (VR), Augmented Reality (AR) [2], 3D television [3], biometrics recognition [4], medical imaging [5], post-capture processing techniques such as depth estimation and refocusing [6], or semantic segmentation [7]. However, the rich quality trades off with a high volume of redundant data from both within and between viewpoints, leading to the need of obtaining efficient compression approaches.

Recently, graph-based coding has proven to be an efficient approach to LF compression [8–10] in comparison with conventional 2D image-based compression methods, e.g., HEVC [11], JPEG Pleno [12]. This is because the conventional methods use rectangular blocks which often contain non-uniform intensities or sub regions with different statistical properties. Such non-uniform representation of signal achieves low energy compaction when transformed into frequency domain, leading to higher bitrate required for coding. Meanwhile, graph-based coding can efficiently exploit the redundancy within pixels blocks with irregular shape, adhering closely to object boundaries. More concretely, graphs with arbitrary shapes containing mostly uniform pixel intensities are transformed into frequency domain using Graph Fourier Transform (GFT). As the result, better energy compaction of coefficients can be achieved. Among exiting graph-based LF coding methods, the one in [10] achieves the best rate-distortion performance by proposing graph coarsening and partitioning in a rate-distortion sense. Indeed, in comparison with the methods in [8,9], this

method is capable of reducing graph vertices and obtaining smaller graphs from the original high dimensional graphs. At the same time, it assures that the redundancies within and between views can still be efficiently exploited at some target coding bitrates. This allows the redundancy in bigger pixel regions where the signal is smooth to be efficiently exploited. As the result, high rate-distortion performance can be achieved. However, compared to HEVC Lozenge [11] and JPEG-Pleno [12], the method in [10] remains outperformed for real LF suffering from vignetting effect and synthetic LF at high bitrates, despite having highest rate-distortion at low bitrates. Additionally, their execution time is reported to be 10 times higher than HEVC for a single LF at the same target quality, mainly due to time complexity of Laplacian eigen-decomposition.

It is believed that the main reason why [10] does not perform well on high coding bitrates relates to the error in disparity information used for super-ray projection. To elaborate this point, a closer look at the concept of super-rays as the common support of graph-based LF coding studies is needed. It is an extension of super-pixels over-segmentation in 2D images [13]. In other words, upon views of LF, each super-ray is a group of corresponding super-pixels across all views. The purpose is to group similar light rays coming from the same object in the 3D space to different viewpoints, as an analogy to grouping perceptually similar pixels which are close to each other in 2D image. The similarity contains high redundancy, and thus good energy compaction can be obtained in the frequency domain. In details, existing graph-based LF coding studies [8–10] segments top-left view into super-pixels, computes the median disparity per super-pixel based on the estimated disparity of top-left view, then applies disparity shift for the projection of a super-pixel from the reference view to remaining views at both encoder and decoder. Due to the similar geometry (structures of objects) and optical characteristics (distance from camera to objects) between the viewpoints, scaling of the disparity value can be used to shift the pixels from one viewpoint to any other viewpoint. This emphasizes the importance of the accuracy of disparity information to the projection of super-pixels.

However, in the case of real LF captured by plenoptic camera or camera array, if the selected reference view suffers from vignetting effect, the estimated disparity would not be accurate, and thus the projection of super-pixels would also suffer errors, leading to incorrect position of corresponding super-pixel in target view. For synthetic LF generated by software, the baseline distance between every two viewpoints has no constraint, and thus it usually has much larger disparity between views compared to real LF, whose baseline is limited by aperture size of a plenoptic camera. Hence, using only one median disparity per super-ray would make the super-ray projection less accurate, particularly when super-pixel size is large. These issues are verified and further explained in Section 3.

To this end, in this paper, two novel projection schemes related to selection of reference views for super-ray projection for real LF and synthetic LF, are proposed, to tackle error in disparity information and improve the super-ray projection quality. For real LF with vignetting issue, instead of using top-left view as a reference view, the center view is proposed to be used. This allows the projection to spread out to neighboring views symmetrically in both directions. As a result, the properties of the obtained depth map are preserved. For synthetic LF with a large disparity, instead of choosing only a single reference view in the top-left corner, multiple views in a sparse distribution are proposed. This allows to perform projection to closer views. As a result, the error of median disparity per super-ray used for projection can be reduced. Moreover, each reference view would be associated with a distinct global graph (a set of all super-ray graphs), and thus the original global graph is divided into smaller sub global graphs. In this way, they can be processed in parallel, decreasing computation time. In order to determine the optimal number of views in this proposal, a Lagrangian minimization problem is solved. The purpose is to avoid increasing bitrates during transmission of reference segmentation maps and disparity maps.

The experimental results demonstrate that by using the proposed projection schemes, higher rate-distortion performance and lower computation time are generally achieved, in comparison with various baselines. The main contributions of the paper are as follows:

- How vignetting effect results in inaccurate depth estimation, how large disparity between views leads to higher median disparity error for projection, and how these issues affect the projection quality are examined qualitatively and quantitatively;
- A center view projection scheme is proposed for real LF with large parallax, suffering from vignetting effect in peripheral views, in which the center view is selected as the reference instead of top-left view. This scheme outperforms both original scheme [10] and state-of-the-art coders such as HEVC or JPEG Pleno at low and high bitrates;
- A multiple views projection scheme is proposed for synthetic LF, in which the positions of reference views are optimized by a minimization problem, so that projection quality is improved and inter-views correlations can still be efficiently exploited. In results, this proposal significantly outperforms the original scheme [10] in terms of both Rate Distortion and computation time, by parallel processing sub global graphs with smaller dimensions;
- A comparative analysis with qualitative and quantitative results is given on rate-distortion performance between the two proposals and original projection scheme [10], as well as HEVC-Serpentine and JPEG Pleno 4DTM.

The rest of the paper is organized as follows: Section 2 introduces LF compression categories and recent studies on graph-based LF compression. Section 3 provides a verification of the issues resulting in the error of disparity information. A detailed description of the two projection schemes is given in Section 4. In Sections 5 and 6, experimental results and analysis are discussed to evaluate the performance of proposals. Conclusion is given in Section 7.

## 2. Related Work

In this section, the paper first provides some background on LF compression with their representations and the recent associated compression approach, then further surveys existing studies on graph-based LF compression. The goal is to understand the current progress of LF compression, the potentials of graph-based LF coding and clarify the benefit of graph coarsening and partitioning over other recent graph-based approach, as well as its existing issues.

### 2.1. Light Field Compression

LF compression can be generally based on two approaches: compressing the raw lenslet image (2D image) or compressing multiple views (array of 2D images) extracted from the raw data.

The first category aims at LF with lenslet-based representation, which is a 2D image containing a grid of microlens images, and most of its solutions [14–17] take advantage of existing HEVC by extending new intra prediction modes exploiting correlation between micro-images, each of which is the captured image from each micro-lens. Other standards have also been considered, such as JPEG-2000 in [18], to code the residuals after depth-based sparse prediction of micro-images. More recently, authors in [19] proposed a lossless codec architecture for raw lenslet LF images using sparse relevant regressors and contexts (SRRC). The encoder splits the raw image into rectangular patches, each corresponding to a micro-lens. The codec exploits inter-patch correlation based on patch-by-patch prediction mechanism, and intra-patch correlation is exploited by designing a sparse predictor for each pair of patch and the label of that patch, which is classified based on either the Bayer mask colors or depth information. The results of their best method (SRRC-PHASE) have shown to considerably reduce file size and outperform well-known predictive standards, i.e., 18.5% less bits than JPEG 2000, 22.4% less bits than JPEG-LS on average.

Methods in the second category aim at pseudo-video-sequence-based, multiview-based, volumetric-based, and geometry-assisted based LF representations, all of which can

be generated from lenslet acquisition, or multiview acquisition. In the case of raw lenslet acquisition, captured by plenoptic camera, the image is first preprocessed by de-vignetting and demosaicing, then a dense array of views (micro-images or sub-aperture images) are extracted. For multiview acquisition, captured by an array of cameras, multiple views with full parallax can be used directly without preprocessing. The variety of ways the views are stacked together inspire different LF compression approaches.

For pseudo-video-sequence (PVS)-based representations, the 2D array of viewpoints are scanned following a specific pattern to form a 1D array of views. This array represents the (pseudo) temporal relationship between the views, and thus any conventional 2D video coder can be implemented to exploit inter-view correlation inside the pseudo video. In addition to evaluating compression efficiency, recent studies on this category also try to tackle trade-off between the viewpoint scalability, random access functionality and compression performance. A new coding framework was proposed in [20] using HEVC PVS-based LF encoder with two novel profiles, namely, HEVC-SLF to include scalable functionalities and HEVC-SLF-RA to include both scalable and viewpoint random access features. HEVC-SLF attempts to increase the number of scalability layers and the flexibility in their selection, while HEVC-SLF-RA proposes two reference picture selection (RPS) variants to increase random access at the cost of reducing coding efficiency. With various flexible encoding profiles, their results have shown to achieve 44% bitrate savings when compared with the original PVS-based HEVC, 37% and 47% when compared with MuLE (applied in JPEG Pleno 4DTM) and WaSP (applied in JPEG Pleno 4DPM), respectively.

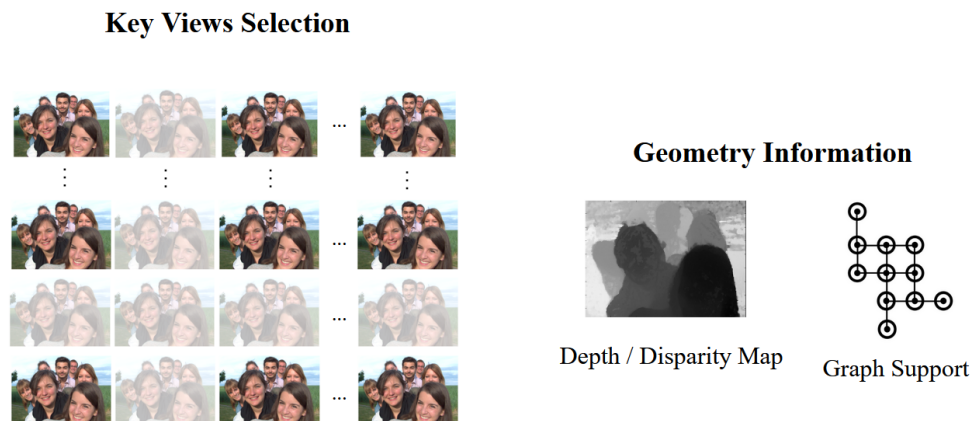
Similar to PVS, volumetric-based representation also scans viewpoints and forms a 1D array of views, but considers the stack as a 3D volume instead, without temporal dimension. While PVS uses standard coders to partition each view (pseudo video frame) into 2D blocks for encoding process, volumetric-based coding partitions whole 3D volume of LF content into 3D blocks, then advanced volumetric data compression standards such as JP3D (JPEG 2000 part 10 [21]) can be used to exploit intra-view (within view) and inter-view (between views) redundancy.

Multiview-based LF representation stacks 2D array of viewpoints into 1D array of multiple PVSs as a conventional 3D multiview format, and thus can be coded with common 3D video coders such as MVC and MVC-HEVC. If PVS-based LF coding can exploit only spatial (intra-view) and (pseudo) temporal correlations, multiview-based coding exploits all dimensions including inter-view correlation. MVC-based multiview coding has been widely introduced in LF coding [22–26], whereas the MVC-HEVC based approach is more recent [27].

The most recent geometry-assisted based LF representation does not rely heavily on stacking viewpoints or try to consider the whole LF content as a 2D/3D video, hence, it depends less on traditional coders, and has high potential for improvement. Instead, research into this category focuses on key view selection and geometry estimation problems (i.e., depth estimation for LF [28–32]), as depicted in Figure 1.

Geometry-assisted based LF representation is accompanied with view synthesis based LF compression, which has been adopted in the 4DPM (4D prediction) mode of JPEG Pleno, a new standard project within the ISO/IEC JTC 1/SC 29/WG 1 JPEG Committee, specialized in novel image modalities such as textured-plus-depth, light field, point cloud or holograms. JPEG Pleno implements two strategies to exploit LF redundancy, 4DTM and 4DPM. The 4DTM mode utilizes a 4D transform approach, and targets real LF with high angular view density obtained by plenoptic cameras. Raw LF in lenslet format is first converted into multiview representation, and 4DTM partitions LF into variable-size 4D blocks (two spatial and two angular dimensions), then each block is transformed using 4D DCT. On the other hand, the 4DPM mode divides multiple views of LF into a set of reference views and intermediate views. Texture and geometric depth of reference views are encoded using JPEG 2000, then at the decoder side, a hierarchical depth-based prediction technique is used to obtain depth maps of discarded views, and their textures are warped from the references based on obtained depths. Hence, the 4DPM mode can encode LF very efficiently

under reliable depth information. However, at the time of writing of this paper, the 4DPM mode is not yet available in the open source code of JPEG Pleno Reference Software [33], and thus, the 4DTM mode is used for comparison in this paper instead.



**Figure 1.** Geometry-assisted representation for LF. (**Left side**) Key views selection, in which the discarded views are faded. The remaining selected views are used to estimate geometry information (**Right side**), including depth/disparity estimation, and obtaining an efficient graph model.

Deep Learning has also been introduced into view synthesis-based LF reconstruction. From a sparse set of decoded images, a residual network model was proposed in [34] to reconstruct densely sampled LF images. Instead of training a model with sparse sampled viewpoints of the same scene, the raw lenslet image is directly used, and thus, the image reconstruction task is transformed into image-to-image translation. Training with raw lenslet images, the network can understand and model the relationship between viewpoint images well, enabling more texture details to be restored and ensuring better reconstruction quality. The features of small-baseline LF was extracted to define the target images to be reconstructed using the nearest-view method. Their proposal improved the average PSNR over the second-best method (Zhang et al. [35]) by 0.64 dB. Authors in [36] proposed a Lightweight Deformable Deep Learning Framework to resolve the problem of disparity in LF reconstruction, by feature extraction and angular alignment using the deformable convolution network approach, without using a loss function. Additionally, a novel activation function was introduced to reduce time complexity for LF super-resolution images. Their reconstruction quality results have been shown to outperform state-of-the-art LF image reconstruction methods, while reducing 37% training time and 40% execution time using super-resolution activation function.

## 2.2. Graph-Based Light Field Coding

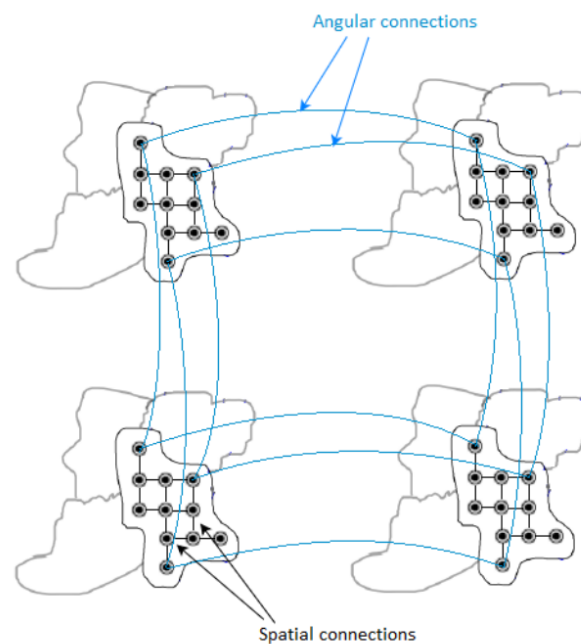
Graph-based light field coding falls into the second category of LF compression which compresses a dense array of 2D images (micro-images or sub-aperture images) extracted from the raw lenslet LF, aiming at geometry-assisted based LF representation. Graph vertices are used to describe colors with pixel intensities as graph signals, while graph connections reflect geometry dependencies intra-view or inter-view. The graph signals are transformed into the frequency domain to exploit energy compaction using Graph Fourier Transform (GFT), then quantized and encoded to send to the decoder, while the graph support (Laplacian matrix) can be encoded using a separate lossless coder.

In [8], a graph-based solution is proposed with graph support defined on the super-ray segmentation, first introduced in [13] to group light rays of similar color being close in 3D space, as an extension to the concept of super-pixels obtained by SLIC segmentation in 2D image [37]. A super-pixel groups perceptually similar pixels within a view, and a super-ray groups corresponding super-pixels across views, and total super-rays form up the LF image. Their proposal first selects top-left view as the key view, obtains super-pixel segmentation

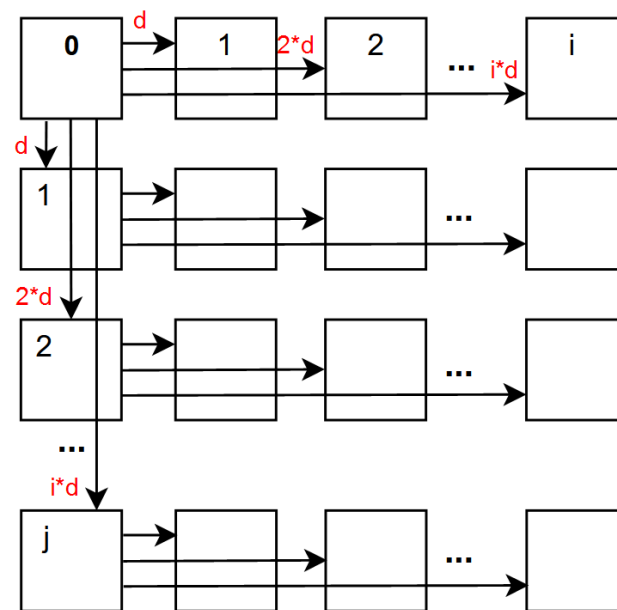
labels using SLIC algorithm [28], computes its disparity map using method in [29], then projects super-pixel labels to other views based on disparity shift, and construct local graphs of super-rays across all views. Spatial edges connect pixels within a super-pixel, and angular edges form connections between corresponding pixels of the same super-pixel at every four views. An example of the process is illustrated in Figures 2–4. Their results have shown that it outperforms HEVC Lozenge [11] at high bitrates for all real LF datasets, but performs worse at low bitrates. This can be explained by the fact that using a limited size for graph support might overcome with computational complexity of high-dimensional non-separable graph, yet it may not enable GFT to exploit long-term spatial or angular redundancy of signal.



**Figure 2.** An example of super-pixel segmentation for real-world LF on dataset *Fountain\_Vincent\_2*, with number of super-pixels set at 2000.



**Figure 3.** A super-ray graph consisting of spatial graphs connecting pixels within a super-pixel and angular graphs connecting corresponding pixels across views.  $I_{1,1}$ ,  $I_{1,2}$ ,  $I_{2,1}$ ,  $I_{2,2}$  are four adjacent views.



**Figure 4.** Top-left view projection scheme based on median disparity per super-pixel.  $d$  denotes the median disparity in a specific super-pixel.  $*$  denotes the multiplication between the median disparity and any position of the view.

In [9], the authors improve on their previous work in [8] by addressing the issue of limited local graph size, and propose sampling and prediction schemes for local graph transform to exploit correlation beyond limits of local graph, without extending graph size. Their proposal first samples the LF data based on graph sampling theory to form a new image of reference samples, then encodes it with conventional 2D image coder with powerful intra-prediction ability. The encoder sends the coded reference image along with only high frequency coefficients of graph transforms. At the decoder side, a prediction mechanism in the pixel domain is introduced to predict the low frequency coefficients using the obtained reference image and high frequency coefficients. Their schemes are designed for quasi-lossless (high quality) coding and have shown substantial RD gain compared to HEVC-Inter Raster scan at lossless mode. However, their performance can drop drastically with lower bitrate, as the prediction scheme is highly dependent on high frequency coefficients, in which a tiny change (i.e., small rounding) may lead to significant reduction in reconstruction quality.

Their most recent study [10] also concerns the high complexity of non-separable graph in [8] using graph coarsening and partitioning, guided by a rate-distortion model for graph optimization. Graph coarsening is performed to reduce the number of vertices inside a super-ray graph, below a threshold leading to acceptable complexity, while retaining basic properties of the graph. If signal approximation of a reduced graph gives too coarse a reconstruction of the original signals, or contains two regions with different statistics properties despite having acceptable number of vertices, the local graph is partitioned into two sub-graphs instead. Their experiment results have shown to surpass other state-of-the-art coders like HEVC Lozenge [11] and JPEG Pleno [12] for ideal real LF, but outperformed by most coders at high bitrates (quasi-lossless) for real LF suffered from vignetting effect, and synthetic LF, even though their proposal's performance still exceeds others at low bitrate.

Importantly, both [9,10] implement the same super-ray projection mechanism as in [8]. They select the top-left view as the reference view, compute its disparity map and segmentation map, then project the super-pixels labels to all other views based on the median disparity per super-pixel, as illustrated in Figure 4. The projection scheme proceeds row by row, with horizontal projection from left to right in each row, and one vertical projection from above for the first view of every row. However, the top-left view might not

be the optimal selection for reference view on real LF due to vignetting, and choosing one median disparity per super-ray might incur high disparity error on synthetic LF, if every pixel in a super-pixel has high disparity, especially when the super-pixel is large. This research's purpose is to clarify how the above issues have a negative impact on disparity information, then to propose two novel projection schemes to mitigate the issues, and obtain enhanced projection quality of super-pixels. The main focus is to improve the most recent graph-based solution [10] with already better performance among the other two approaches [8,9], so that it can perform well on the full range of coding bitrates for all types of LF, using the two proposed projection schemes.

### 3. Impact of Disparity Information on Projection Quality

In this section, an overview of the evaluated LF datasets is given, and the issues related to affecting reconstructed view quality in [10] for each type of LF are shown as follows:

- How vignetted real LF affects its disparity estimation;
- How synthetic LF with large disparity leads to higher median disparity error;
- How do both issues affect the quality of super-ray projection? To support the verification, SSIM metric [38] was used to compute the projection quality for each view with top-left view projection.

#### 3.1. Datasets

In this paper, both real-world LF and synthetic LF were examined to verify the existence of vignetting effect and high disparity for each type of LF, respectively. Additionally, they were also used for the evaluation of compression efficiency for each LF coding method. The datasets were carefully selected following the LF Common Test Conditions Document [39] in order to provide a wide range of scenarios that would challenge compression algorithms, in terms of acquisition technology (Lenslet Lytro Illum camera, Synthetic creation), scene geometry, spatial resolution, number of viewpoints, bit depth and texture.

Real-world LF acquired with plenoptic cameras were downloaded from the EPFL dataset [40], which contains natural scenes with wide baseline. The contents were captured with a Lytro Illum camera and pre-processed with Light Field Matlab Toolbox to obtain  $15 \times 15$  viewpoint images, each with a resolution of  $625 \times 434$  pixels at 10-bit depth. However, only central  $13 \times 13$  views were considered in this paper, then color and gamma corrections were applied on each view point image to reduce the strong vignetting effect. *Danger\_de\_Mort* and *Fountain\_Vincent\_2* were selected as vignetted real LF scenes with large parallax  $13 \times 13$  views in this dataset. The scenes were categorized into 10 groups, including 'buildings', 'grids', 'mirrors and transparency', 'landscapes', 'nature', 'ISO and Color Charts', 'People', 'Studio', 'Urban', 'Lights'. *Danger\_de\_Mort* belongs to the 'grids' group, which contains shots of different highly detailed grid patterns close to camera with a wide depth of field. Whereas, *Fountain\_Vincent\_2* is classified into the group 'people', capturing portrait shots of one to seven people at different depth positions.

Synthetic LF considered in this paper are *Greek* and *Sideboard* from HCI 4D Light Field Benchmark dataset [41], containing 4 stratified and 20 photorealistic scenes of  $9 \times 9$  views,  $512 \times 512$  pixels per view at 8-bit depth. The scenes were generated using Blender software with LF plugin. Per-pixel ground truth disparity map is also available for each scene, which is beneficial for super-ray projection based on disparity, whereas the projection in real world LF requires an extra step of depth estimation. In the considered scenes, *Greek* has higher disparity range due to objects being close to camera, whereas *Sideboard* has smaller disparity because most objects are further away from the viewpoints, but it contains more objects with various shapes, and thus the complex geometry can be more challenging.

#### 3.2. Vignetting Effect Degrades Disparity Estimation

Vignetting has been extensively surveyed regarding its impact on traditional 2D stereo correspondence problems [42] among other radiometric differences such as image noises, different camera settings, etc. Stereo correspondence, or stereo matching, is an active topic

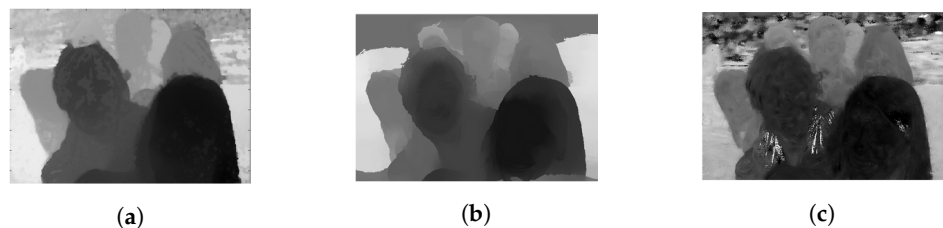


in computer vision with the goal to estimate depth information for 2D images from an image pair. However, the optical flow-based approach has shown to outperform stereo-based algorithms [30,31] in the task of depth estimation for LF in recent literature [28,29]. This section provides a demonstration on how vignetting impacts optical flow-based LF depth estimation in a subjective manner.

Due to the inability to efficiently capture light rays in peripheral lens, plenoptic cameras usually produce vignetted border views in a multiview-based representation of LF content captured at wide angle, which is essential to provide high parallax. First, considering the case of real LF with medium parallax ( $9 \times 9$  views), this experiment examines the difference between the top-left and center views of dataset *Friends* [40] qualitatively. As depicted in Figure 5, the two images are almost identical, with a tiny shift in the position of objects, but no visual distortion in terms of colors or blurring occurs. The depth map of top-left view is then computed using an optical flow-based method in [29] and compared with their original result, also with two other state-of-the-art stereo-based disparity estimators [30,31]. From Figure 6, it is apparent that the obtained disparity map adheres closely to the basic depth properties which all methods have in common.

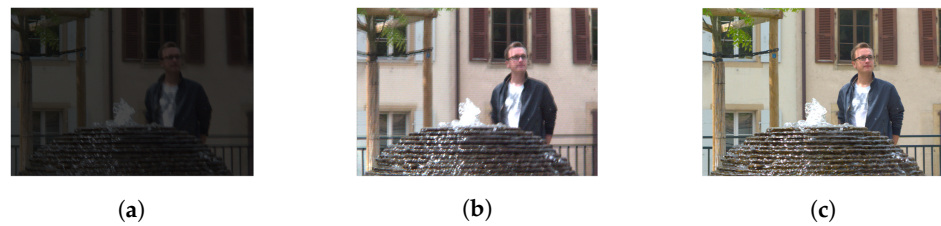


**Figure 5.** Samples of views in dataset *Friends*. (a) Top-left view. (b) Center view.



**Figure 6.** Disparity maps obtained using optical-flow-based and stereo-based depth estimators in the dataset *Friends*. (a) Estimated disparity map of top-left view using [29]. (b) Estimated disparity map of center view using [30]. (c) Estimated disparity map of center view using [31].

However, in the case of real LF with high parallax ( $13 \times 13$  views), the obtained depth map might lose these properties due to the vignetting effect, even after being de-vignetted using color correction methods. A comparison between the top-left view and center view of dataset *Fountain\_Vincent\_2* [40] with  $13 \times 13$  views is shown in Figure 7a,c. The original top-left view can be seen with significant degradation in intensity and color. Gamma correction can be used for color calibration, and thus help reduce vignetting considerably, as seen in Figure 7b. However, a close look at the calibrated top-left view reveals that object details are blurry with minor distortion in the colors, yet its estimated disparity map is severely affected. It is much worse compared to the output in center view using the same optical flow-based [29] or stereo-based [30,31] depth estimation, illustrated in Figure 8. For example, the fountain, which is closest to the camera, now shares the same depth as some of objects in the background (i.e., tree), whereas a segment of the wall now becomes the closest, according to its depth. The instances which have shown even a minor vignetting effect can result in significant deterioration of the estimated depth map in peripheral views.



**Figure 7.** Sample of views in dataset *Fountain\_Vincent\_2*. (a) Top-left view (original). (b) Top-left view (de-vignetted using gamma correction). (c) Center view.




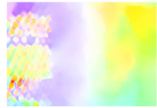






**Figure 8.** Disparity maps obtained using optical-flow-based and stereo-based depth estimators in dataset *Fountain\_Vincent\_2*. (a) Estimated disparity map of top-left view (gamma corrected) using [29]. (b) Estimated disparity map of center view using [29]. (c) Estimated disparity map of center view using [30]. (d) Estimated disparity map of center view using [31].

### 3.3. Synthetic LF with Large Disparity Leads to High Median Disparity Error for a Super-Pixel

While real LF captured using plenoptic camera has a baseline limited by the aperture size of the camera lens, synthetic LF can be generated using graphics software without baseline restraint, enabling wide parallax for the viewer. Therefore, synthetic LF may present a much larger disparity between views than real LF, leading to higher median disparity error per super-ray. This can considerably affect the projection of super-pixel locations from reference view to remaining views. In Table 1, the previews of two real LF and two synthetic LF are displayed, along with their optical flow between a pair of views, and the disparity range. The intensity of color in the optical flow is proportional to the disparity. Additionally, the disparity range displays the maximum and minimum of disparity value in the whole LF. It can be seen that synthetic LF has distinctively more intense optical flow and a wider range of disparity.

**Table 1.** Comparison of disparity range between real LF and synthetic LF.

Datasets	Center View	Optical Flow	Disparity Range
<i>Fountain_Vincent_2</i> [40] (real LF)			$-0.495 \rightarrow 0.798$
<i>Danger_de_Mort</i> [40] (real LF)			$-1.306 \rightarrow 0.683$
<i>Greek</i> [41] (synthetic LF)			$-2.880 \rightarrow 3.637$
<i>Sideboard</i> [41] (synthetic LF)			$-1.513 \rightarrow 1.845$

In order to verify this issue, a mathematical explanation is given. This scenario considers the projection scheme based on disparity shift, which was used by the authors in [8–10]. As illustrated in Figure 4, the disparity shift scheme increases disparity proportionally to the distance between the target and reference view. Our goal is to verify that the further the target view, the higher the median disparity error, leading to a worse projection of super-pixel labels. Let us consider an arbitrary super-pixel in the reference view (top-left), depicted in Figure 9.



**Figure 9.** An example of super-pixel segmentation for synthetic LF on dataset Greek, with the number of super-pixels set at 1200. The selection of an arbitrary super-pixel is highlighted in red.

Let us denote all disparity values of every pixel in this super-pixel as  $D = d_1, d_2, \dots, d_n$  where  $n$  is the total number of pixels, and  $d_m$  is the median disparity value of this set. Since a super-pixel follows well any object's boundary, it is safe to assume that no super-pixel contains pixels of two separate objects at two distinct depth planes, and thus the disparity variation is smooth, or a normal distribution of disparity values in a super-pixel is obtained most often. Therefore, without loss of generality for all super-pixels, this verification considers the median disparity is equal to the mean value, as in Equation (1),

$$d_m = \frac{\sum_{i=1}^n d_i}{n} \quad (1)$$

The median disparity error is calculated as follows,

$$mse = \frac{1}{n} \sum_{i=1}^n (d_i - d_m)^2 = \frac{1}{n} \sum_{i=1}^n \Delta^2$$

Denote  $mse_1, \Delta_1$  as the median disparity error and the disparity error, respectively calculated at view 1 (reference view), and  $mse_k, \Delta_k$  at view  $k$ . Since disparity of a pixel at view  $k$  is  $k$  times higher than at view 1, the following equations are derived,

$$mse_1 = \frac{1}{n} \sum_{i=1}^n \Delta_1^2$$

$$mse_k = \frac{1}{n} \sum_{i=1}^n \Delta_k^2$$

$$\begin{aligned}
\Delta_1 &= d_i - d_m = d_i - \frac{\sum_{i=1}^n d_i}{n} \\
\Delta_k &= k * d_i - d'_m = k * d_i - \frac{\sum_{i=1}^n k * d_i}{n} = k * (d_i - \frac{\sum_{i=1}^n d_i}{n}) = k * \Delta_1 \\
\Rightarrow mse_k &= \frac{1}{n} \sum_{i=1}^n k^2 * \Delta_1^2 = k^2 * \frac{1}{n} \sum_{i=1}^n \Delta_1^2 = k^2 * mse_1
\end{aligned} \tag{2}$$

Hence, if using the original projection scheme proposed by the authors [8–10], the median disparity error of a specific super-pixel at target view  $k$  is  $k^2$  higher than the reference view. As the consequences, the further the target view, the more inaccurately that super-pixel is relocated. Furthermore, the issue becomes worse when the disparity is already large, as in synthetic LF.

### 3.4. Inaccurate Disparity Information Leads to Poor Super-Ray Projection

To evaluate how the super-ray projection schemes used in [8–10] are affected by disparity error in a quantitative manner, the SSIM [38] metric was used to measure the similarity of the segmentation labels in a target view, between ground truth labels (segmented using SLIC algorithm [28]) and projected labels from reference view, with the projection scheme depicted in Figure 4. Specifically, the evaluation formula is given in Equation (3),

$$quality = SSIM(L_{i,j}, L'_{i,j}) \tag{3}$$

where  $i$ , or  $j$  is the location of the target view in 2D array of views;  $L_{i,j}$  is denoted as “ground truth” image where every pixel intensity is the label value of the super-pixel it belongs to, assigned by SLIC;  $L'_{i,j}$  is also an image of labels, but assigned by super-ray projection from reference view. The SSIM value range is between 0 and 1, with 1 indicating best quality, or the two images being identical, and lower scores indicate worse quality.


#### 3.4.1. For Real-World LF with High Parallax (Vignetting)

First, the real world LF dataset *Fountain\_Vincent\_2* [40]  $13 \times 13$  views was considered. This dataset suffers from vignetting, which then leads to an inaccurate disparity map. The ground truth labels of each view were obtained using the Python SLIC segmentation library, with parameter values as *compactness* = 30 and *n\_segments* = 2000. The projection quality of  $13 \times 13$  views was computed, as shown in Table 2. Results have shown that the view with highest quality is apparently the reference view (top-left), since no projection was made, and the quality gradually deteriorates when the projection is made to further target views. The disparity error accumulated for lower views was selected as reference for horizontal projection in the first column, and the views in the bottom-right corner obtained the worst quality, which means the reconstructed super-pixels of those views might have been located at highly inaccurate positions, hence the distortion. Similarly, the projection quality of dataset *Danger\_de\_Mort* [40] is given in Table 3.

#### 3.4.2. For Synthetic LF with High Median Disparity Error per Super-Ray

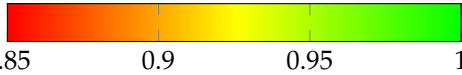
The projection quality was examined in the synthetic dataset *Greek* [41]  $9 \times 9$  views using SLIC with the same parameters as for real LF datasets. The results in Table 4 reveal that the quality of each view deteriorated much faster in both directions, due to the large disparity between the views. Despite having fewer views than vignettted real LF, dataset *Greek* ended up having worse projection quality at bottom-right views. Similarly, Table 5 illustrates projection quality in synthetic dataset *Sideboard* [41].

**Table 2.** Projection quality of  $13 \times 13$  views in dataset *Fountain\_Vincent\_2* using top-left view projection scheme. Each number is the computed SSIM result corresponding to a view. Reference view with absolute quality is highlighted in pure green. The color transition from green to red corresponds to the degradation of projection quality.




	0.85	0.9	0.95	1								
1.000	0.971	0.963	0.958	0.953	0.952	0.950	0.950	0.950	0.945	0.945	0.942	0.941
0.956	0.961	0.955	0.951	0.947	0.947	0.945	0.946	0.945	0.942	0.941	0.939	0.936
0.943	0.941	0.939	0.939	0.935	0.934	0.932	0.932	0.931	0.932	0.930	0.928	0.926
0.935	0.933	0.930	0.930	0.929	0.927	0.927	0.923	0.923	0.925	0.923	0.922	0.921
0.921	0.920	0.922	0.923	0.921	0.920	0.918	0.917	0.915	0.916	0.915	0.913	0.915
0.914	0.913	0.915	0.915	0.915	0.914	0.910	0.909	0.909	0.909	0.908	0.909	0.909
0.907	0.907	0.905	0.906	0.906	0.905	0.903	0.903	0.904	0.903	0.903	0.903	0.902
0.902	0.902	0.900	0.900	0.900	0.899	0.899	0.899	0.899	0.899	0.897	0.897	0.896
0.895	0.897	0.895	0.896	0.896	0.896	0.896	0.895	0.895	0.894	0.893	0.892	0.891
0.892	0.893	0.892	0.890	0.891	0.891	0.892	0.892	0.890	0.890	0.889	0.888	0.889
0.886	0.888	0.887	0.884	0.885	0.885	0.886	0.886	0.886	0.886	0.884	0.882	0.883
0.877	0.883	0.885	0.882	0.883	0.882	0.882	0.881	0.882	0.881	0.880	0.880	0.880
0.873	0.875	0.876	0.880	0.880	0.878	0.879	0.878	0.877	0.876	0.877	0.877	0.876

**Table 3.** Projection quality of  $13 \times 13$  views in dataset *Danger\_de\_Mort* using top-left view projection scheme. Each number is the computed SSIM result corresponding to a view. Reference view with absolute quality is highlighted in pure green. The color transition from green to red corresponds to the degradation of projection quality.




	0.85	0.9	0.95	1								
1.000	0.969	0.961	0.959	0.951	0.951	0.948	0.952	0.949	0.943	0.943	0.941	0.939
0.955	0.959	0.953	0.949	0.946	0.945	0.944	0.945	0.943	0.941	0.939	0.937	0.934
0.941	0.940	0.938	0.938	0.933	0.933	0.930	0.931	0.929	0.930	0.929	0.926	0.925
0.933	0.932	0.928	0.929	0.929	0.926	0.925	0.921	0.922	0.923	0.921	0.921	0.920
0.920	0.918	0.920	0.921	0.919	0.919	0.917	0.915	0.913	0.915	0.913	0.912	0.913
0.913	0.914	0.914	0.914	0.913	0.912	0.911	0.908	0.907	0.908	0.909	0.908	0.907
0.906	0.905	0.904	0.904	0.905	0.903	0.901	0.902	0.903	0.901	0.901	0.901	0.900
0.900	0.901	0.898	0.901	0.899	0.897	0.898	0.897	0.897	0.897	0.897	0.894	0.893
0.893	0.896	0.894	0.895	0.895	0.894	0.894	0.894	0.893	0.891	0.892	0.889	0.888
0.891	0.891	0.892	0.889	0.890	0.890	0.890	0.891	0.888	0.889	0.888	0.886	0.886
0.888	0.887	0.886	0.882	0.884	0.883	0.885	0.883	0.884	0.883	0.883	0.879	0.882
0.876	0.881	0.884	0.880	0.882	0.880	0.881	0.880	0.879	0.878	0.879	0.878	0.877
0.872	0.874	0.875	0.878	0.879	0.877	0.877	0.877	0.874	0.873	0.876	0.874	0.873

**Table 4.** Projection quality of  $9 \times 9$  views in dataset *Greek* using a single-view projection scheme. Each number is the computed SSIM result corresponding to a view. Reference view with absolute quality is highlighted in pure green. The color transition from green to red corresponds to the degradation of projection quality.



1.000	0.958	0.941	0.931	0.927	0.924	0.921	0.922	0.918
0.941	0.932	0.925	0.919	0.916	0.913	0.911	0.910	0.906
0.915	0.912	0.908	0.906	0.904	0.901	0.900	0.898	0.896
0.898	0.897	0.898	0.896	0.895	0.893	0.891	0.889	0.887
0.889	0.889	0.887	0.886	0.886	0.885	0.883	0.882	0.883
0.887	0.886	0.884	0.884	0.882	0.882	0.882	0.882	0.883
0.889	0.888	0.886	0.886	0.884	0.884	0.883	0.884	0.884
0.884	0.880	0.880	0.878	0.878	0.878	0.879	0.879	0.875
0.877	0.875	0.872	0.872	0.871	0.871	0.872	0.870	0.870

**Table 5.** Projection quality of  $9 \times 9$  views in dataset *Sideboard* using single-view projection scheme. Each number is the computed SSIM result corresponding to a view. Reference view with absolute quality is highlighted in pure green. The color transition from green to red corresponds to the degradation of projection quality.



1.000	0.952	0.927	0.914	0.911	0.906	0.901	0.901	0.895
0.949	0.934	0.921	0.911	0.906	0.902	0.899	0.897	0.892
0.919	0.914	0.907	0.898	0.892	0.891	0.886	0.884	0.885
0.904	0.897	0.891	0.888	0.882	0.884	0.881	0.877	0.877
0.882	0.885	0.881	0.878	0.876	0.877	0.874	0.874	0.868
0.877	0.875	0.873	0.872	0.870	0.870	0.869	0.866	0.863
0.867	0.869	0.869	0.868	0.867	0.867	0.865	0.862	0.863
0.863	0.862	0.863	0.864	0.863	0.865	0.864	0.861	0.859
0.860	0.860	0.860	0.859	0.861	0.862	0.860	0.858	0.857

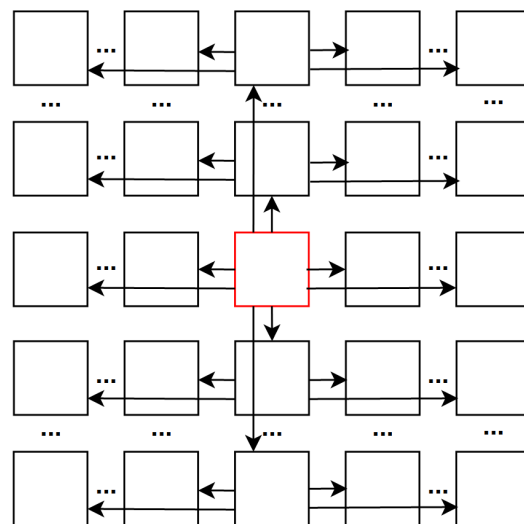
## 4. Proposals

In this paper, two novel projection schemes are proposed for real LF and synthetic LF as follows:

- For real LF with many viewpoints suffering from vignetting effect, the proposed approach is that super-ray projection be carried out on the center view as the reference, then spread out to surrounding views, instead of the top-left one with inaccurate disparity;
- For synthetic LF with large disparity, a projection scheme using multiple views in a sparse distribution as references is proposed, aiming to reduce the distance between target and reference views. In addition, using multiple reference views can create multiple sub global graphs which are processed simultaneously. This allows to mitigate computational time for both encoder and decoder.

### 4.1. Center-View Projection Scheme

The proposed center-view projection scheme is illustrated in Figure 10. The purpose of this proposal is to improve rate distortion performance of [10] in real-world LF data with large parallax, which suffers from vignetting in peripheral views. From a center view, the projection spreads out to neighboring views, instead of proceeding row by row in one direction, as in [8–10]. Specifically, for a  $N \times N$  views real LF, this scheme performs a horizontal projection from the center view  $I_{\frac{N+1}{2}, \frac{N+1}{2}}$  to remaining  $\frac{N}{2}$  views symmetrically on the center row  $R_{\frac{N+1}{2}}$ . Vertical projection is also performed in the center column symmetrically in both directions, with  $I_{\frac{N+1}{2}, \frac{N+1}{2}}$  as reference. Then, for each remaining  $N - 1$  rows, its center view is now used as reference for the horizontal projection, covering all views of the remaining  $N - 1$  columns. This projection scheme not only avoids inaccurate disparity estimation in top-left view due to vignetting, but also allows projection to closer views (half the distance compared to top-left view projection), hence, the quality of more views can be improved.



**Figure 10.** Center-view projection scheme.

### 4.2. Multiple Views Projection Scheme

The purpose of this proposal is to improve rate distortion performance and reduce computation time when applying the approach in [10] into synthetic LF data with large disparity between views. Equation (2) has shown that the median disparity error of any super-pixel in a target view is  $k^2$  higher than of corresponding super-pixel in reference view, in which the target view is the  $k$ th view away from the reference. This negatively affects the projection quality. In addition, using a single view as a reference will result in a single global graph of very high dimension, which leads to high encoding and decoding time. Hence, to approach these two problems at the same time, a novel idea is to increase

the number of reference views in a row or column. Thereby, the distance from reference views to the to-be-projected views will be decreased. At the same time, multiple smaller global graphs can be created, which enable leveraging the power of parallel computing to improve the execution time.

The question is how many references should be sufficient. Too many references would lead to an inability to efficiently exploit angular correlation across views of the whole LF, whereas few references would cause each reference view to project to further views, and thus increase projection error. Another interpretation of this question is how many views should each reference view project to, in a row or a column.

The question can be answered by finding a target view with worst projection quality, while being close to the reference view as much as possible, then using its ground truth segmented labels as a new reference view for a new projection chain. This can be interpreted as a Lagrangian minimization problem,

$$\min(k[x] + \lambda * SSIM[x]) \quad (4)$$

or

$$\min(SSIM[x] + \lambda * k[x]), \quad (5)$$

where  $k$  can be considered as an array of distances between target and reference view in a single direction (with number of views as unit)  $k = k_1, k_2, \dots, k_{n-1}$ ;  $SSIM$  is an array of projection quality computed using Equation (3) in the same direction (multiplied with 100 to be on the same scale as  $k$ )  $SSIM = q_1, q_2, \dots, q_{n-1}$ ; and  $n$  is the number of views for that row or column.  $k$  and  $SSIM$  arrays exclude the reference view. Equation (4) or Equation (5) can be re-expressed as,

$$\min_x k[x] \quad w.r.t \quad SSIM[x] = SSIM_{target} = const \quad (6)$$

or

$$\min_x SSIM[x] \quad w.r.t \quad k[x] = k_{target} = const \quad (7)$$

The optimal Lagrange multiplier  $\lambda^*$  is not known in advance, and can be varied with the desired distance of views ( $k_{target}$ ) or quality ( $SSIM_{target}$ ).

The optimal solutions to Equation (6) or Equation (7) are the optimal points ( $k[x^*]$ ,  $SSIM[x^*]$ ) lying on the lower half convex hull of the scatter plot of  $SSIM$  and  $k$ . For the sake of simplicity, the median point on the convex hull ( $k[x^{**}]$ ,  $SSIM[x^{**}]$ ) is chosen as the new reference view to avoid being too close or too far from the reference view.

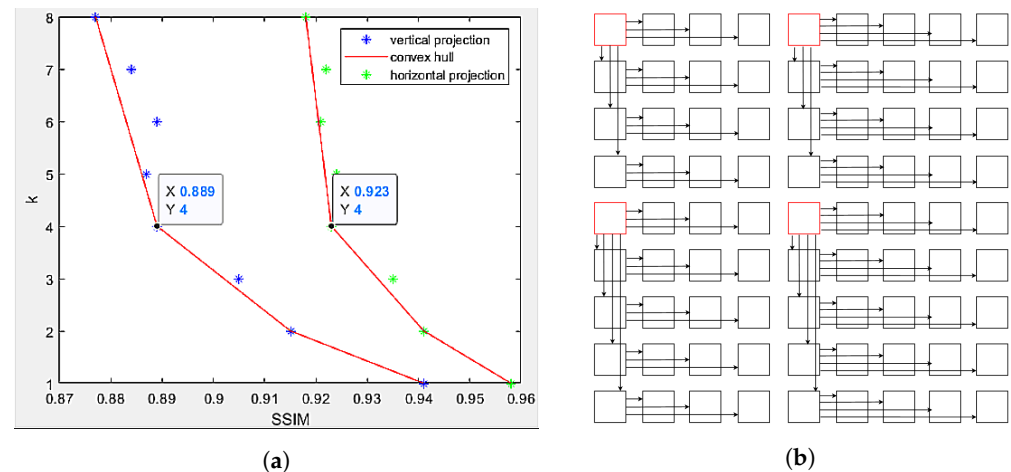
An example is shown in Figure 11, using data in Table 4 of dataset *Greek*  $9 \times 9$  views, the  $SSIM$  values are plotted against  $k$  values for horizontal projection (first row) and vertical projection (first column). For each type of projection, this method finds its corresponding convex hull, then determines the median point as the next reference. Figure 11a reveals  $k[x^{**}] = 4$  for both directions, or the new reference for horizontal projection is the fourth view away from the original top-left reference view, while vertical projection also selects the fourth view. The new projection scheme can be seen in Figure 11b with four reference views, instead of one. It should be noted that views  $I_{1,9}$  and  $I_{5,9}$  are not selected as references because no more views to be projected after them, despite being the fourth view away from the previous references.

We denote a local graph to be the graph with spatial and angular connections within a super-ray, and a global graph to be the set of all local graphs. The original high dimensional global graph is now partitioned into four sub-global graphs with better projection quality and less complexity, while still exploiting angular correlations of at least four views in every direction.

Additionally, depending on the technical implementation, all four sub-global graphs can be processed simultaneously by taking advantage of parallel programming. The main complexity of graph-based LF coding lies in its Laplacian diagonalization of each local graph, which is  $O(n^3)$ , with  $n$  as the number of nodes. By partitioning into four sub global

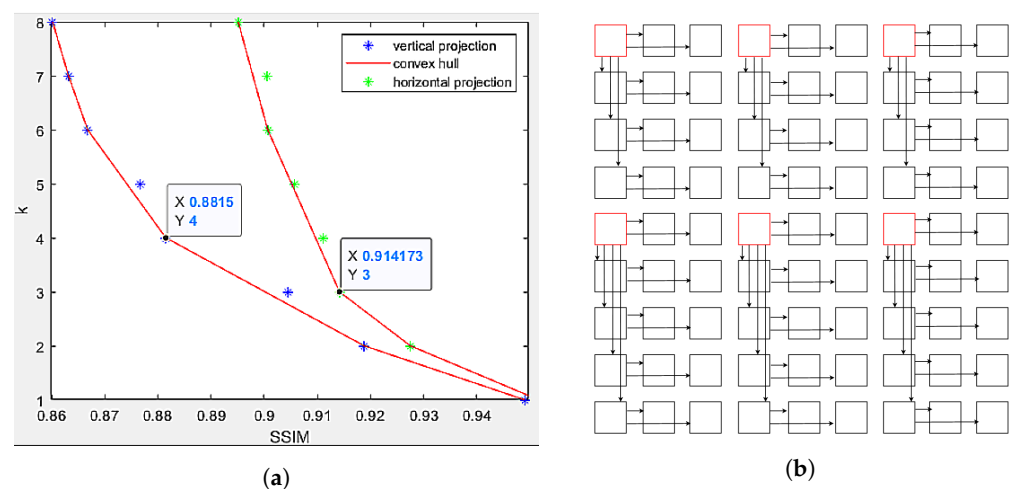


graphs,  $n$  is reduced by a fourth approximately for each global graph, if not accounting for graph coarsening, and thus computation time can decrease significantly. With graph coarsening enabled to reduce vertices by approximating original graph, as detailed in [10], the number of nodes in the original scheme might be smaller than the total number of nodes in all sub global graphs, because the original graph with higher dimensions may have more coarsened local graphs than each of sub global graphs with lower dimensions in the multi-references scheme. However, the number of nodes in each sub global graph is significantly smaller than the original, and each sub-global graph is processed independently, hence computation time for both encoder and decoder can still be reduced considerably.



**Figure 11.** Multi-view projection scheme in dataset *Greek*. (a) Scatter plot of target views based on its distance to reference view and its projection quality. Potential candidates selected as the next reference view lie on the convex hull. (b) Multi-view projection scheme using every 4th view horizontally and vertically as reference views.

The selection of references for dataset *Sideboard*  $9 \times 9$  views is shown in Figure 12a,b, in which horizontal projection selects every 3rd view as the new reference, and vertical projection selects the 4th view. The original graph is partitioned into six sub-graphs, exploiting angular correlations of at least three views in every direction, and having better projected segmentation maps for all the views, compared to the original projection scheme.



**Figure 12.** Multi-view projection scheme in dataset *Sideboard*. (a) Scatter plot of target views based on its distance to reference view and its projection quality. Potential candidates selected as the next reference view lie on the convex hull. (b) Multi-view projection scheme using every 3rd view horizontally and every 4th view vertically as reference views.

## 5. Performance Evaluation

In this section, evaluation of super-ray projection quality for each view is analyzed quantitatively to show that it can be improved by the two proposed center view, and multiple views projection schemes. Then, a set of experiments are designed to evaluate the impact of enhanced projection quality on overall compression efficiency. Finally, experimental results are presented and analyzed.

### 5.1. Projection Quality Evaluation


#### 5.1.1. Center View Projection Scheme

Using Equation (3), this experiment computed the projection quality of a vignettted LF, but with center view as the reference. Tables 6 and 7 show SSIM quality results on all the  $13 \times 13$  views of *Fountain\_Vincent\_2* and *Danger\_de\_Mort*, with absolute SSIM on center view. Thanks to the accurate disparity map, it can be seen that the quality deterioration from the center view to further views horizontally and vertically were slower than in the case of projecting from top-left view, described in Tables 2 and 3. Additionally, the quality of more views was improved because center view projected to more closer views (smaller disparity) to four directions, whereas corner view projected to further views (higher disparity) to two directions.

#### 5.1.2. Multiple Views Projection Scheme


Tables 8 and 9 show the projection quality of synthetic LF *Greek* and *Sideboard* using multiple views projection scheme. Absolute SSIM was found in all reference views. Although the deteriorating rate of quality remained fast due to large disparity, the quality of remaining views was highly improved, compared to Tables 4 and 5. This can be explained by the fact that views with the worst projection quality used ground truth segmentation labels instead, then they became new reference views with accurate segmentation for new projection chains.

**Table 6.** Projection quality of  $13 \times 13$  views in dataset *Fountain\_Vincent\_2* using center view projection scheme. The reference view with absolute quality is highlighted in pure green. The color transition from green to red corresponds to the degradation of projection quality.



0.917	0.918	0.919	0.919	0.918	0.918	0.921	0.919	0.919	0.918	0.917	0.919	0.917
0.922	0.925	0.925	0.927	0.926	0.926	0.929	0.926	0.926	0.924	0.924	0.924	0.923
0.927	0.928	0.929	0.929	0.929	0.931	0.930	0.928	0.927	0.926	0.925	0.926	0.927
0.938	0.940	0.940	0.941	0.943	0.944	0.946	0.943	0.941	0.938	0.938	0.939	0.938
0.939	0.940	0.940	0.944	0.945	0.947	0.949	0.944	0.942	0.941	0.940	0.941	0.941
0.955	0.954	0.955	0.959	0.962	0.968	0.979	0.965	0.960	0.959	0.957	0.955	0.954
0.959	0.959	0.961	0.965	0.969	0.977	1.000	0.974	0.970	0.968	0.966	0.962	0.958
0.955	0.957	0.957	0.958	0.962	0.965	0.974	0.968	0.964	0.961	0.962	0.960	0.956
0.943	0.944	0.945	0.946	0.946	0.948	0.953	0.949	0.947	0.947	0.944	0.943	0.940
0.940	0.941	0.941	0.941	0.943	0.944	0.946	0.945	0.946	0.944	0.942	0.941	0.939
0.928	0.929	0.931	0.931	0.930	0.930	0.931	0.930	0.931	0.930	0.928	0.924	0.924
0.922	0.925	0.927	0.928	0.928	0.927	0.927	0.926	0.925	0.924	0.920	0.921	0.918
0.911	0.913	0.914	0.917	0.917	0.918	0.916	0.915	0.914	0.913	0.912	0.911	0.908


**Table 7.** Projection quality of  $13 \times 13$  views in dataset *Danger\_de\_Mort* using a center view projection scheme. The reference view with absolute quality is highlighted in pure green. The color transition from green to red corresponds to the degradation of projection quality.



0.85	0.9	0.95	1
------	-----	------	---

0.917	0.916	0.917	0.920	0.917	0.916	0.919	0.920	0.918	0.917	0.916	0.918	0.916
0.920	0.924	0.924	0.925	0.924	0.925	0.928	0.924	0.924	0.923	0.922	0.922	0.921
0.925	0.926	0.927	0.927	0.927	0.929	0.929	0.927	0.926	0.925	0.924	0.925	0.926
0.936	0.939	0.939	0.939	0.943	0.942	0.944	0.941	0.939	0.937	0.937	0.938	0.936
0.938	0.939	0.939	0.943	0.943	0.945	0.947	0.943	0.940	0.939	0.939	0.939	0.939
0.954	0.955	0.953	0.958	0.960	0.967	0.981	0.963	0.958	0.957	0.959	0.953	0.952
0.957	0.957	0.960	0.963	0.968	0.976	1.000	0.973	0.968	0.966	0.965	0.961	0.956
0.954	0.956	0.956	0.960	0.960	0.963	0.972	0.966	0.962	0.959	0.960	0.959	0.954
0.942	0.942	0.943	0.944	0.945	0.946	0.951	0.948	0.945	0.945	0.943	0.942	0.939
0.938	0.939	0.941	0.939	0.942	0.942	0.945	0.943	0.944	0.943	0.941	0.940	0.937
0.929	0.927	0.930	0.930	0.928	0.929	0.929	0.929	0.930	0.928	0.927	0.922	0.923
0.921	0.923	0.926	0.926	0.926	0.926	0.926	0.924	0.924	0.923	0.919	0.920	0.917
0.909	0.912	0.912	0.916	0.915	0.917	0.914	0.913	0.913	0.912	0.910	0.910	0.906


**Table 8.** Projection quality of  $9 \times 9$  views in dataset *Greek* using a multi-view projection scheme. The reference view with absolute quality is highlighted in pure green. The color transition from green to red corresponds to the degradation of projection quality.



0.85	0.9	0.95	1
------	-----	------	---

1.000	0.958	0.941	0.931	1.000	0.961	0.945	0.935	0.926
0.941	0.932	0.925	0.919	0.940	0.931	0.926	0.920	0.913
0.915	0.912	0.908	0.906	0.913	0.912	0.910	0.905	0.900
0.898	0.897	0.898	0.896	0.901	0.897	0.896	0.894	0.890
1.000	0.956	0.941	0.929	1.000	0.959	0.942	0.931	0.923
0.933	0.925	0.920	0.917	0.936	0.927	0.922	0.917	0.912
0.909	0.907	0.902	0.901	0.910	0.906	0.904	0.900	0.898
0.895	0.894	0.893	0.891	0.895	0.892	0.891	0.890	0.887
0.888	0.885	0.882	0.883	0.887	0.887	0.885	0.883	0.880

**Table 9.** Projection quality of  $9 \times 9$  views in dataset *Sideboard* using multi-view projection scheme. The reference view with absolute quality is highlighted in pure green. The color transition from green to red corresponds to the degradation of projection quality.



1.000	0.952	0.927	1.000	0.941	0.921	1.000	0.946	0.923
0.949	0.934	0.921	0.942	0.925	0.912	0.946	0.928	0.911
0.920	0.914	0.907	0.915	0.906	0.898	0.912	0.905	0.901
0.902	0.897	0.891	0.896	0.890	0.887	0.900	0.891	0.885
1.000	0.945	0.925	1.000	0.945	0.918	1.000	0.943	0.915
0.946	0.929	0.914	0.942	0.927	0.909	0.943	0.924	0.908
0.910	0.907	0.902	0.915	0.908	0.898	0.916	0.905	0.898
0.894	0.889	0.889	0.897	0.893	0.889	0.898	0.891	0.887
0.881	0.880	0.879	0.882	0.884	0.880	0.885	0.880	0.878

## 5.2. Compression Efficiency Evaluation

In order to evaluate the impact of enhanced projection quality on overall performance, this section assesses Rate Distortion performance and quality of reconstructed LF of the two proposed center view and multi-view projection schemes, and computation time for the multi-view projection scheme. This allows to demonstrate the improvement of quality in both proposals, as well as the running time for multiple views projection. In Rate Distortion quantitative results, the proposals were compared against the original top-left view projection scheme and two state-of-the-art coders: HEVC with Serpentine scanning topology, and JPEG Pleno with 4D transform mode (4DTM).

### Experiment Setup

The encoder and decoder were run on Python 3 under Ubuntu 20.04 with 64 GB RAM, and utilizing Python's Ray library for the parallel processing of super-rays or sub-global graphs. The disparity estimation technique was used from [29] to compute the disparity map for center view of real LF, and multiple reference views for synthetic LF. Their segmentation mask was obtained using SLIC algorithm [28]. Due to lack of memory resources in this experimental environment, the initial number of super-rays was set as 2000 and 1200 for real LF  $13 \times 13$  views and synthetic LF  $9 \times 9$  views, respectively, instead of 500 as originally used in [10]. The more super-rays, the smaller the local graphs, and thus they would consume less resources, but with the trade-off of inefficient decorrelation of signals. On the other hand, having a smaller number of super-rays leads to bigger sizes of graphs, which implies significant increase in time complexity of eigen-decomposition for the Laplacian matrix.

The subjective results were obtained when running encoder and decoder with parameter PSNR<sub>min</sub> set at 20, instead of 45 (max value). This parameter was used to guide the rate of graph coarsening and partitioning. Setting at 20 would return results of low quality, and thus, it would be easier to visually differentiate results of the original and proposed projection schemes, for the purpose of reading the paper.

The x.265 implementation of HEVC-Serpentine used in the experiments was run with source version 2.3, following LF common test conditions [39]. The base quantization parameters (QP) were set to 10, 28, 35, 50. JPEG Pleno 4DTM was used within Part 4 (Reference Software) with base Lambdas (quantization parameter) of 25, 1000, 20,000, 500,000.

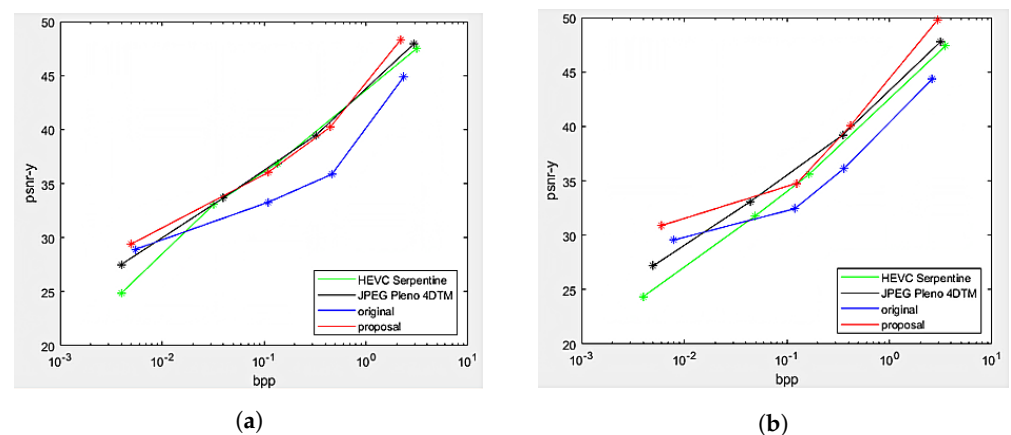
Regarding QPs used in both original and proposed schemes, this work implemented an adaptive quantization approach. After GFT was used to transform signals into the frequency domain, the super-ray coefficients were divided into 32 sub-groups. Since the first group contains low frequency coefficients, which represent fundamental properties of the signals, and it usually has much higher energy than the next groups, more quantization steps should be assigned for the first group to obtain more accurate reconstruction than other groups, for containing more important coefficients. In other words, the base QPs were set adaptively for the first group and remaining groups. Then, optimized quantization step sizes were found based on the rate-distortion optimization approach, as described in [10], with parameter QP set according for each group. Using the optimized quantization steps sizes, the coefficients in each group were quantized and arithmetically coded with a public version of Context Adaptive Binary Arithmetic Coder (CABAC) [43]. At high-quality coding, the QPs were set as 4 for the first group, and 10 for the remaining groups. The reference segmentation mask was encoded using arithmetic edge coder EAC [44], and disparity values (median disparity per super-ray) were encoded using original arithmetic coder of the authors [10].

Additionally, all reconstructed LFs using any of the four methods in the experiments were converted into 8-bit for evaluation at the same conditions and their PSNR of the luminance channel (PSNR-Y) were computed with the same formula, following the LF common test conditions [39].

### 5.3. Analysis of Center View Projection Scheme

#### 5.3.1. Rate Distortion Analysis

The Rate Distortion performance of the proposed projection with center view as reference was compared against top-left view projection [8], direct encoding of the views as a PVS using HEVC-Serpentine, and 4D transform solution utilizing JPEG Pleno 4DTM. The performance comparison was made on the two datasets *Fountain\_Vincent\_2* and *Danger\_De\_Mort*, as shown in Figure 13. Substantial gains in the Center view projection proposal can be seen compared to the original scheme at all bitrates for both datasets, as well as the proposed scheme outperformed HEVC-Serpentine and JPEG Pleno 4DTM, especially at low and high bitrates.



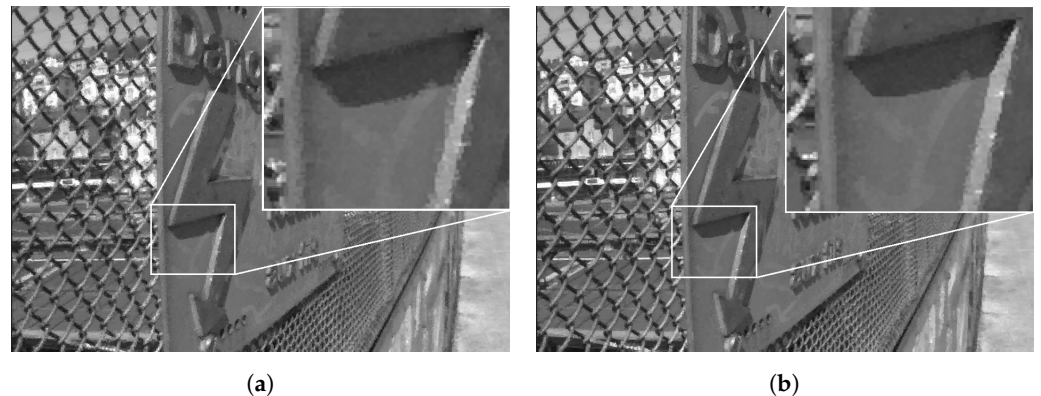
**Figure 13.** Rate Distortion performance between center view projection scheme (proposal), top-left view projection scheme (original [10]), and state-of-the-art codecs HEVC-Serpentine, JPEG Pleno 4DTM. (a) *Fountain\_Vincent\_2*. (b) *Danger\_de\_Mort*.

### 5.3.2. Qualitative Analysis for Reconstructed LF

At the decoder side, the luminance channel of LF was reconstructed from the quantized coefficients sent by the encoder. The output results using the original and proposed projection scheme are shown subjectively in Figure 14 for *Fountain\_Vincent\_2*, and Figure 15 for *Danger\_de\_Mort*. It can be seen that in both datasets, the proposed Center view projection returned sharper results, clearly visible in edges around texture, whereas the original scheme's results seemed to be blurry in these edges. The blur effect could be caused by super-pixels reconstructed at inaccurate positions, resulting from poor depth estimation, as a consequence of vignetted top-left view.



**Figure 14.** Reconstructed luminance channel of center view using projection scheme from top-left view and center view, in dataset *Fountain\_Vincent\_2*. (a) Top-left view projection scheme [10] (original). (b) Center view projection scheme (proposal).



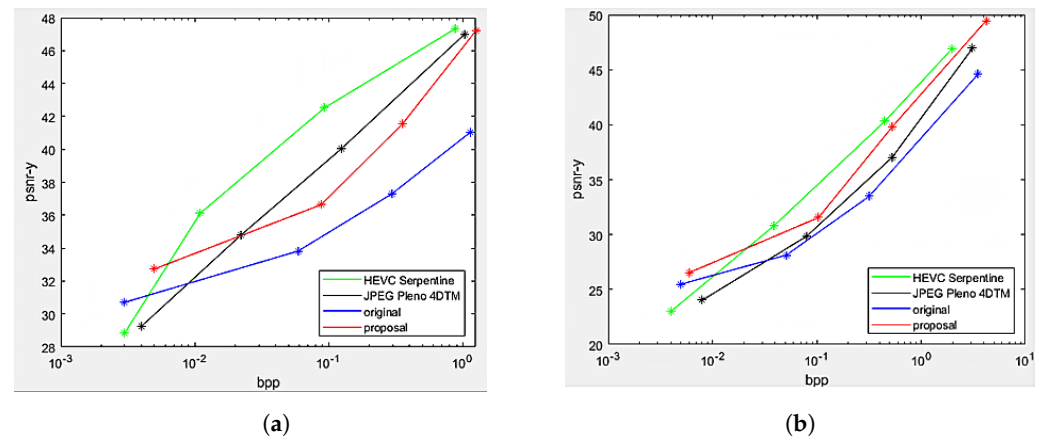
**Figure 15.** Reconstructed luminance channel of center view using projection scheme from top-left view and center view, in dataset *Danger\_de\_Mort*. (a) Top-left view projection scheme [10] (original). (b) Center view projection scheme (proposal).

## 5.4. Analysis of Multiple Views Projection Scheme

### 5.4.1. Rate Distortion Analysis

Rate Distortion performance of the multi-view projection scheme was illustrated in Figure 16, comparing with original single view projection scheme, HEVC, and JPEG Pleno in datasets *Greek* and *Sideboard*. The proposed scheme significantly outperformed the original projection at all bitrates, having better projection quality, and surpassed the other two conventional coders at low bitrates. However, HEVC-Serpentine remained the best compressor for synthetic LF at medium and high bitrates. This can be explained by the fact that the two synthetic LF are free of imperfections such as image noises, and thus, the performance of classical coders HEVC and JPEG Pleno was not degraded, thus achieving better Rate Distortion than their results in real-world LF. Nevertheless, the proposed method performed slightly worse on dataset *Greek*, compared to *Sideboard*, because the disparity between views in *Greek* is higher than other datasets, as shown in Table 1, leading

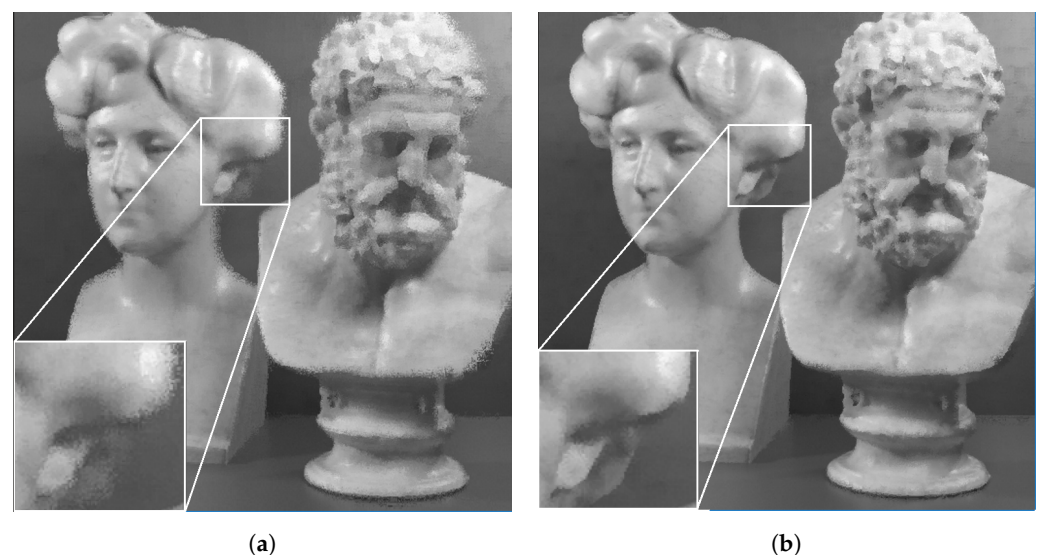
to higher median disparity error per super-ray used for projection. In addition, more sub-global graphs can be found in *Sideboard* than *Greek* after finding optimized positions for reference views, resulting in more views with better projection quality.



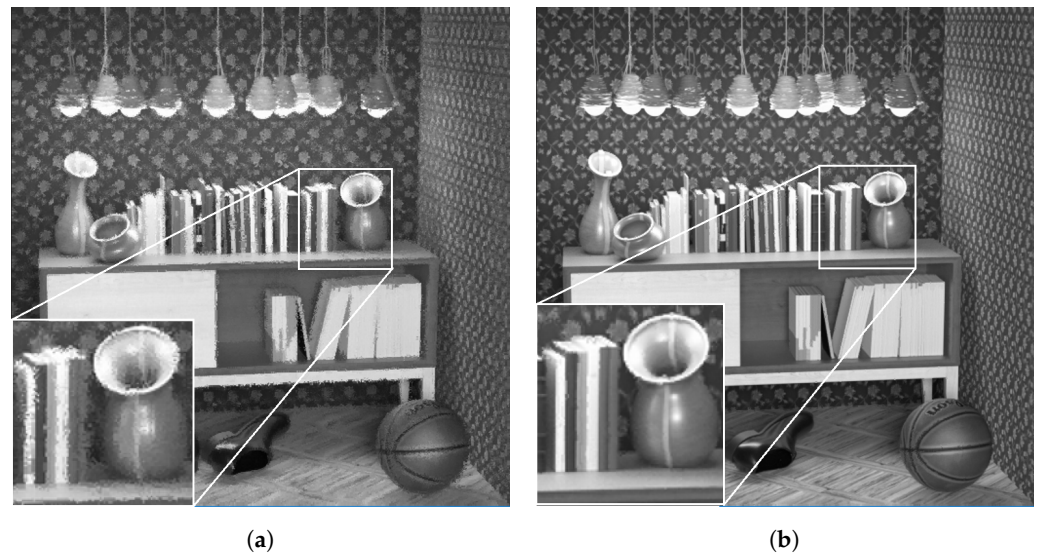
**Figure 16.** Rate Distortion performance between multi-view projection scheme (proposal), top-left view projection scheme (original [10]), and state-of-the-art codecs HEVC-Serpentine, JPEG Pleno 4DTM. (a) *Greek*. (b) *Sideboard*.

#### 5.4.2. Qualitative Analysis for Reconstructed LF

The qualitative results of luminance reconstruction for *Greek* and *Sideboard* datasets using original or multi-view projection schemes are shown in Figures 17 and 18. Same as previous subjective results of real LF, the proposed multi-view projection returned sharper results for synthetic LF, especially around edges of textures, for having more accurate projection of super-pixels than the single-view projection scheme.



**Figure 17.** Reconstructed luminance channel of center view using single-view and multi-view projection scheme, in dataset *Greek*. (a) Single-view projection scheme [10] (original). (b) Multiple-view references scheme (proposal).



**Figure 18.** Reconstructed luminance channel of center view using single-view and multi-view projection scheme, in dataset *Sideboard*. (a) Single-view projection scheme [10] (original). (b) Multiple-view reference scheme (proposal).

#### 5.4.3. Computation Time Analysis

Apart from achieving substantial gains in compression performance compared to the single-view projection scheme, multi-view projection can also significantly reduce computation time of both encoder and decoder, with a slight trade-off in increasing bitrates. Experimental results are given in Tables 10 and 11, analyzing the parameters and time duration when running encoder and decoder on dataset *Greek* and *Sideboard* with PSNR<sub>min</sub> set at 40, along with output quality PSNR-Y and required bitrate at high quality coding. The high-dimensional graph was separated into four sub-global graphs in the multi-view proposal, as optimized by a minimization problem. The first three columns (param, obtained *num\_SR*, total # of nodes) bring interesting results. It can be seen that, for the same initial *num\_SR* (number of super-rays/number of local graphs), the output *num\_SR* and total number of nodes after graph coarsening and partitioning of single-view scheme (original) were much greater than the output of each sub-global graph (proposal). This means graph coarsening and partitioning rates were higher in the original graph than in each sub-global graph. Thus, the proposed multi-view scheme could retain more accurate graph information of vertex signals and edges in each sub-global graph, in addition to having higher quality for the projection of super-pixels, since each reference view projected to closer views. Essentially, the total number of nodes determines the time complexity for eigen-decomposition of the Laplacian matrix, and it was smaller in each sub-global graph. Moreover, the four sub-global graphs were encoded or decoded simultaneously by running in parallel, and thus, the total approximate encoding and decoding time were reduced by more than half, compared to processing the original high-dimensional graph. Nevertheless, the bitrate slightly increased as more reference segmentation masks and disparity maps were required to be coded and transmitted alongside the graph coefficients.



**Table 10.** Encoding time and decoding time using the single-view (original) and multi-view (proposal) projection scheme on dataset *Greek*.

	Param	Obtained num_SR (after Graph Coarsening and Partitioning)	Total # of Nodes (after Graph Coarsening and Partitioning)	Approx. Encoding Time (s)	Approx. Decoding Time (s)	psnr_y (Adaptive QP = 4/10)	bpp (Adaptive QP = 4/10)
original	num_SR = [10] 1200	3130	12,294,870	25,835	23,376	41.04 db	<b>1.14 bpp</b>
proposal	num_SR = 1200	sub_graph_1: 1162	4,190,290	<b>10,232</b>	<b>9533</b>	<b>47.19 db</b>	1.25 bpp
		sub_graph_2: 1187	5,139,382				
		sub_graph_3: 1181	5,139,424				
		sub_graph_4: 1294	5,958,210				

**Table 11.** Encoding time and decoding time using single-view (original) and multi-view (proposal) projection scheme on dataset *Sideboard*.

	Param	Obtained num_SR (after Graph Coarsening and Partitioning)	Total # of Nodes (after Graph Coarsening and Partitioning)	Approx. Encoding Time (s)	Approx. Decoding Time (s)	psnr_y (Adaptive QP = 4/10)	bpp (Adaptive QP = 4/10)
original	num_SR [10] = 700	5974	19,147,151	34,516	27,585	44.58 db	<b>3.51 bpp</b>
proposal	num_SR = 700	sub_graph_1: 821	3,100,441	<b>16,072</b>	<b>14,869</b>	<b>49.47 db</b>	4.21 bpp
		sub_graph_2: 825	3,092,246				
		sub_graph_3: 845	3,092,751				
		sub_graph_4: 1080	3,796,717				
		sub_graph_5: 1080	3,780,491				
		sub_graph_6: 1099	3,775,655				

## 6. Discussion

Based on evaluation results, it has been shown that the center view and multiple views projection scheme can bring an overall improvement for super-rays projection quality in all views by having accurate disparity information, leading to better compression efficiency, especially at high bitrates, compared to the original top-left view projection. Additionally, combining accurate geometry information with the advantage of graph coarsening proposed in [10], the graph-based approach can also outperform state-of-the-art coders HEVC and JPEG Pleno at low bitrates.

The benefit of graph coarsening for graph-based approaches is clear, for low bitrates, aside from its ability to exploit correlations for irregular patterns in textures, which allows them to outperform the other two state-of-the-art coders HEVC and JPEG Pleno. Graph coarsening retains total variations of signals on the reduced graphs, while the number of coefficients to be coded also substantially decreases, leading to good Rate Distortion performance at low bitrates. Additionally, besides the vignetting effect, real-world LF might also suffer from image noises, degrading the performance of traditional coding considerably, but not affecting graph coarsening, which utilizes low rank model approximation, and thus, the noises can be removed.

High-quality coding at high bitrates requires particularly accurate super-ray positions, which depends entirely on the performance of the depth estimation algorithm. For real LF, as verified in previous section, the vignetting effect significantly degrades the output

depth map of top-left view, leading to inaccurate super-ray projections, hence, the original projection scheme obtains the lowest Rate Distortion performance. On the other hand, the Center view projection scheme has more accurate depth estimation maps, leading to higher compression performance than HEVC and JPEG Pleno, which might also be supported by its ability to decorrelate signals in irregular-shape textures.

For high-quality coding of synthetic LF, although the proposed multiple views projection scheme significantly surpasses the performance of top-left projection scheme, they are still outperformed by HEVC-Serpentine. The potential solution to obtaining competitive performance with HEVC is to use more reference views, but with a trade-off of increasing bitrates for transmission because more segmentation and disparity information of the references are needed to be coded and sent to the decoder side. Another possible solution could be using two median disparity values for each super-pixel within the reference view, then each half super-pixel would be projected separately to other views, based on the corresponding median disparity value. The idea is motivated by the fact that the smaller size the super-pixel possesses, the smaller the median disparity error becomes, with respect to all disparity values within the super-pixel. This approach might be discussed further in future work.

Furthermore, for the multi-view projection scheme, time execution for both encoder and decoder can be considerably reduced by processing all sub global graphs in parallel, while ensuring correlations between views can still be efficiently exploited by optimizing positions of reference views through a minimization problem. There may be a slight increase in the coding bitrates due to increased reference segmentation and more disparity maps to be coded. Additionally, based on Figures 11a and 12a, it should be noted that solving the minimization problem to find the optimal reference view might not make a significant improvement for multiview-based LF representation, compared to directly choosing center view of every projection direction as the new reference, since there are only a few views to be evaluated, and most of them lie closely on the convex hull. However, this approach can be applied to lenslet-based LF representation, in which the number of views is large, and thus finding the views lying on the convex hull can be more efficient. This idea can also be further discussed in future research.

Another limitation of the proposed multi-view projection scheme for graph-based LF coding is the execution time for both encoder and decoder remains relatively high (1200 super-rays with parallel diagonalization of 15 super-rays took about 2.8 h in existing experiment environment), compared to other standards like JPEG Pleno (in about 15 min with MATLAB implementation [45]), despite having significant improvement from the original single-view projection scheme (in about 7.2 h). The super-rays are potentially suitable for parallel Laplacian diagonalizations with the use of GPU-based computing libraries like MAGMA and magmaFast. Additionally, fast GFT can be used to directly reduce the Laplacian diagonalization and transform time by a factor of up to 27, which has been recently reported in [46,47]. These solutions can be investigated in future work.

## 7. Conclusions

In this paper, two novel projection schemes for graph-based light field coding are introduced, including center view and multiple views projection. The proposals significantly outperformed original Top-left view projection scheme and generally obtained competitive rate-distortion performance with state-of-the-art coders HEVC (Serpentine scanning) and JPEG Pleno (4DTM mode). This can only be achieved by having accurate disparity estimation for center view projection in real LF with large parallax, and smaller median disparity error for multiple views projection in synthetic LF. In addition to improving overall compression efficiency, multiple views projection can also reduce end-to-end computation time by processing smaller sub global graphs in parallel. This has shown the potential of further improvement for graph-based LF coding in order to achieve both competitive performance in both compression efficiency and computation time, compared to state-of-the-art coders.

**Author Contributions:** Conceptualization, N.G.B., C.M.T. and T.N.D.; Methodology, N.G.B., C.M.T. and P.X.T.; Supervision, P.X.T. and E.K.; Writing—original draft, N.G.B., C.M.T. and T.N.D.; Writing—review and editing, P.X.T. and E.K. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Ng, R. Light Field Photography. Ph.D. Thesis, Department Computer Science, Stanford University, Stanford, CA, USA, 2006.
2. Wang, J.; Xiao, X.; Hua, H.; Javidi, B. Augmented reality 3D displays with micro integral imaging. *J. Disp. Technol.* **2015**, *11*, 889–893. [[CrossRef](#)]
3. Arai, J.; Kawakita, M.; Yamashita, T.; Sasaki, H.; Miura, M.; Hiura, H.; Okui, M.; Okano, F. Integral three dimensional television with video system using pixel offset method. *Opt. Express* **2013**, *21*, 3474–3485. [[CrossRef](#)] [[PubMed](#)]
4. Raghavendra, R.; Raja, K.B.; Busch, C. Presentation attack detection for face recognition using light field camera. *IEEE Trans. Image Process.* **2015**, *24*, 1060–1075. [[CrossRef](#)] [[PubMed](#)]
5. Chen, B.R.; Buchanan, I.A.; Kellis, S.; Kramer, D.; Ohiorhenuan, I.; Blumenfeld, Z.; Grisafe, D.J., II; Barbaro, M.F.; Gogia, A.S.; Lu, J.Y.; et al. Utilizing Light field Imaging Technology in Neurosurgery. *Cureus* **2018**, *10*, e2459. [[CrossRef](#)] [[PubMed](#)]
6. Georgiev, T.G.; Lumsdaine, A. Focused plenoptic camera and rendering. *J. Electron. Imaging* **2010**, *19*, 021106.
7. Zhou, J.; Yang, D.; Cui, Z.; Wang, S.; Sheng, H. LRFNet: An Occlusion Robust Fusion Network for Semantic Segmentation with Light Field. In Proceedings of the 2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI), Washington, DC, USA, 1–3 November 2021; pp. 1168–1178. [[CrossRef](#)]
8. Rizkallah, M.; Su, X.; Maugey, T.; Guillemot, C. Geometry-Aware Graph Transforms for Light Field Compact Representation. *IEEE Trans. Image Process.* **2019**, *29*, 602–616. [[CrossRef](#)]
9. Rizkallah, M.; Maugey, T.; Guillemot, C. Prediction and Sampling With Local Graph Transforms for Quasi-Lossless Light Field Compression. *IEEE Trans. Image Process.* **2019**, *29*, 3282–3295. [[CrossRef](#)]
10. Rizkallah, M.; Maugey, T.; Guillemot, C. Rate-Distortion Optimized Graph Coarsening and Partitioning for Light Field Coding. *IEEE Trans. Image Process.* **2021**, *30*, 5518–5532. [[CrossRef](#)]
11. Rizkallah, M.; Maugey, T.; Yaacoub, C.; Guillemot, C. Impact of light field compression on focus stack and extended focus images. In Proceedings of the 24th European Signal Processing Conference (EUSIPCO), Budapest, Hungary, 29 August–2 September 2016; pp. 898–902.
12. Perra, C.; Freitas, P.G.; Seidel, I.; Schelkens, P. An overview of the emerging JPEG Pleno standard, conformance testing and reference software. *Proc. SPIE* **2020**, *11353*, 33. [[CrossRef](#)]
13. Hog, M.; Sabater, N.; Guillemot, C. Superrays for efficient light field processing. *IEEE J. Sel. Top. Signal Process.* **2017**, *11*, 1187–1199. [[CrossRef](#)]
14. Conti, C.; Soares, L.D.; Nunes, P. HEVC-based 3D holoscopic video coding using self-similarity compensated prediction. *Signal Process. Image Commun.* **2016**, *42*, 59–78. [[CrossRef](#)]
15. Li, Y.; Olsson, R.; Sjöström, M. Compression of unfocused plenoptic images using a displacement intra prediction. In Proceedings of the 2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Seattle, WA, USA, 11–15 July 2016; pp. 1–4.
16. Conti, C.; Nunes, P.; Soares, L.D. New HEVC prediction modes for 3D holoscopic video coding. In Proceedings of the 2012 19th IEEE International Conference on Image Processing, Orlando, FL, USA, 30 September–3 October 2012; pp. 1325–1328.
17. Conti, C.; Nunes, P.; Soares, L.D. HEVC-based light field image coding with bi-predicted self-similarity compensation. In Proceedings of the 2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Seattle, WA, USA, 11–15 July 2016; pp. 1–4.
18. Tabus, I.; Helin, P.; Astola, P. Lossy compression of lenslet images from plenoptic cameras combining sparse predictive coding and JPEG 2000. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 4567–4571.
19. Tabus, I.; Palma, E. Lossless Compression of Plenoptic Camera Sensor Images. *IEEE Access* **2021**, *9*, 31092–31103. [[CrossRef](#)]
20. Monteiro, R.J.; Rodrigues, N.M.; Faria, S.M.; Nunes, P.J.L. Light field image coding with flexible viewpoint scalability and random access. *Signal Process. Image Commun.* **2021**, *94*, 16202. [[CrossRef](#)]
21. Information Technology-JPEG 2000 Image Coding System: Extensions for Three-Dimensional Data [Online]. ITU-T Recommendation Document T.809. May 2011. Available online: <https://www.iso.org/standard/61534.html> (accessed on 1 June 2022).

22. Vetro, A.; Wiegand, T.; Sullivan, G.J. Overview of the stereo and multiview video coding extensions of the H.264/MPEG-4 AVC standard. *Proc. IEEE* **2011**, *99*, 626–642. [[CrossRef](#)]
23. Tech, G.; Chen, Y.; Muller, K.; Ohm, J.-R.; Vetro, A.; Wang, Y.-K. Overview of the multiview and 3D extensions of high efficiency video coding. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *26*, 35–49. [[CrossRef](#)]
24. Adedoyin, S.; Fernando, W.A.C.; Aggoun, A. A joint motion & disparity motion estimation technique for 3D integral video compression using evolutionary strategy. *IEEE Trans. Consum. Electron.* **2007**, *53*, 732–739.
25. Adedoyin, S.; Fernando, W.A.C.; Aggoun, A.; Weerakkody, W.A.R.J. An ES based efficient motion estimation technique for 3D integral video compression. In Proceedings of the 2007 IEEE International Conference on Image Processing, San Antonio, TX, USA, 16 September–19 October 2007; Volume 3, p. III-393.
26. Wei, J.; Wang, S.; Zhao, Y.; Jin, F. Hierarchical prediction structure for subimage coding and multithreaded parallel implementation in integral imaging. *Appl. Opt.* **2011**, *50*, 1707. [[CrossRef](#)]
27. Ahmad, W.; Olsson, R.; Sjostrom, M. Interpreting plenoptic images as multi-view sequences for improved compression. In Proceedings of the 2017 IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 4557–4561.
28. Chen, Y.; Alain, M.; Smolic, A. Fast and accurate optical flow based depth map estimation from light fields. In Proceedings of the Irish Machine Vision and Image Processing Conference (IMVIP), Maynooth, Ireland, 30 August–1 September 2017.
29. Jiang, X.; Pendu, M.L.; Guillemot, C. Depth estimation with occlusion handling from a sparse set of light field views. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 634–638.
30. Jeon, H.G.; Park, J.; Choe, G.; Park, J.; Bok, Y.; Tai, Y.W.; So Kweon, I. Accurate depth map estimation from a lenslet light field camera. In Proceedings of the International Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015.
31. Zhang, S.; Sheng, H.; Li, C.; Zhang, J.; Xiong, Z. Robust depth estimation for light field via spinning parallelogram operator. *J. Comput. Vis. Image Underst.* **2016**, *145*, 148–159. [[CrossRef](#)]
32. Rogge, S.; Schiopu, I.; Munteanu, A. Depth Estimation for Light-Field Images Using Stereo Matching and Convolutional Neural Networks. *Sensors* **2020**, *20*, 6188. [[CrossRef](#)]
33. JPEG Pleno Reference Software [Online]. Available online: <https://gitlab.com/wg1/jpeg-pleno-refsw> (accessed on 1 June 2022).
34. Salem, A.; Ibrahim, H.; Kang, H.-S. Light Field Reconstruction Using Residual Networks on Raw Images. *Sensors* **2022**, *22*, 1956. [[CrossRef](#)]
35. Zhang, S.; Chang, S.; Shen, Z.; Lin, Y. Micro-Lens Image Stack Upsampling for Densely-Sampled Light Field Reconstruction. *IEEE Trans. Comput. Imaging* **2021**, *7*, 799–811. [[CrossRef](#)]
36. Ribeiro, D.A.; Silva, J.C.; Lopes Rosa, R.; Saadi, M.; Mumtaz, S.; Wuttisittikulij, L.; Zegarra Rodríguez, D.; Al Otaibi, S. Light Field Image Quality Enhancement by a Lightweight Deformable Deep Learning Framework for Intelligent Transportation Systems. *Electronics* **2021**, *10*, 1136. [[CrossRef](#)]
37. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [[CrossRef](#)]
38. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image Quality Assessment: From Error Visibility to Structural Similarity. *Image Processing. IEEE Trans. Image Process.* **2004**, *13*, 600–612. [[CrossRef](#)]
39. JPEG Pleno Light Field Coding Common Test Conditions v3.3 [Online]. Available online: [http://ds.jpeg.org/documents/jpegpleno/wg1n84049-CTQ-JPEG\\_Pleno\\_Light\\_Field\\_Common\\_Test\\_Conditions\\_v3\\_3.pdf](http://ds.jpeg.org/documents/jpegpleno/wg1n84049-CTQ-JPEG_Pleno_Light_Field_Common_Test_Conditions_v3_3.pdf) (accessed on 1 June 2022).
40. EPFL Light Field Image Dataset [Online]. 2016. Available online: <http://mmspg.epfl.ch/EPFL-light-field-image-dataset> (accessed on 1 June 2022).
41. Honauer, K.; Johannsen, O.; Kondermann, D.; Goldluecke, B. A dataset and evaluation methodology for depth estimation on 4D light fields. In Proceedings of the Asian Conference on Computer Vision (ACCV), Taipei, Taiwan, 20–24 November 2016; Springer: Cham, Switzerland, 2016; pp. 19–34.
42. Hirschmuller, H.; Scharstein, D. Evaluation of Stereo Matching Costs on Images with Radiometric Differences. *Pattern Analysis and Machine Intelligence. IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 1582–1599. [[CrossRef](#)]
43. Context Adaptive Binary Arithmetic Coder (CABAC) [Online]. Available online: <https://github.com/christianrohlfig/ISScabac/> (accessed on June 2022).
44. Daribo, I.; Cheung, G.; Florencio, D. Arithmetic edge coding for arbitrarily shaped sub-block motion prediction in depth video compression. In Proceedings of the 2012 19th IEEE International Conference on Image Processing, Orlando, FL, USA, 30 September–3 October 2012; pp. 1541–1544.
45. *JPEG Pleno Light Field Coding Vm 1.1*, document N81052, ISO/IEC JTC. 1/SC29/WG1 JPEG. October 2018. Available online: <https://jpeg.org/jpegpleno/documentation.html> (accessed on 1 June 2022).
46. Magoarou, L.L.; Gribonval, R.; Tremblay, N. Approximate fast graph Fourier transforms via multi-layer sparse approximations. *IEEE Trans. Signal Inf. Process. Over Netw.* **2018**, *4*, 407–420. Available online: <https://hal.inria.fr/hal-01416110> (accessed on 1 June 2022). [[CrossRef](#)]
47. Lu, K.; Ortega, A. Fast graph Fourier transforms based on graph symmetry and bipartition. *IEEE Trans. Signal Process.* **2019**, *67*, 4855–4869. [[CrossRef](#)]