Geometry-Aware Graph Transforms for Light Field Compact Representation

Mira Rizkallah, Xin Su, Thomas Maugey, Christine Guillemot

Geometry-Aware Graph Transforms for Light Field Compact Representation

Mira Rizkallah*, *Student Member, IEEE,* Xin Su†, *Member, IEEE,* Thomas Maugey†, *Member, IEEE,*
and Christine Guillemot†, *Fellow, IEEE*
* IRISA, Campus Universitaire de Beaulieu, 35042 Rennes, France
† INRIA Rennes Bretagne Atlantique, Campus Universitaire de Beaulieu, 35042 Rennes, France

***Abstract*—The paper addresses the problem of energy compaction of dense 4D light fields by designing geometry-aware local graph-based transforms. Local graphs are constructed on super-rays that can be seen as a grouping of spatially and geometry-dependent angularly correlated pixels. Both non separable and separable transforms are considered. Despite the local support of limited size defined by the super-rays, the Laplacian matrix of the non separable graph remains of high dimension and its diagonalization to compute the transform eigen vectors remains computationally expensive. To solve this problem, we then perform the local spatio-angular transform in a separable manner. We show that when the shape of corresponding super-pixels in the different views is not isometric, the basis functions of the spatial transforms are not coherent, resulting in decreased correlation between spatial transform coefficients. We hence propose a novel transform optimization method that aims at preserving angular correlation even when the shapes of the super-pixels are not isometric. Experimental results show the benefit of the approach in terms of energy compaction. A coding scheme is also described to assess the rate-distortion perfomances of the proposed transforms and is compared to state of the art encoders namely HEVC-lozenge [1], JPEG pleno 1.1 [2], HEVC-pseudo [3] and HLRA [4] .**

***Index Terms*—Light Fields, Energy Compaction, Transform coding, Super-rays, Graph Fourier Transform**

I. Introduction

Recently, there has been a growing interest in light field imaging. By sampling the radiance of light rays emitted by the scene along several directions, light fields enable a variety of post-capture processing techniques such as refocusing, changing perspectives and viewpoints, depth estimation, simulating captures with different depth of fields and 3D reconstruction [5], [6], [7]. This however comes at the expense of collecting large volumes of redundant high-dimensional data, which appears to be one key downside of light fields.

Research effort has been recently dedicated to the design of light field compression algorithms, by either adapting standardized solutions (in particular HEVC) to light field data (*e.g.* [3] [8] [9]), by proposing homography-based low rank models for reducing the angular dimension [4], or by investigating local Gaussian mixture models in the 4D ray space [10]. The authors in [11], use a depth-based segmentation of the light field into 4D spatio-angular blocks with prediction followed by JPEG-2000.

In this paper, we address the problem of graph transforms optimization for light fields energy compaction and compact representation. Indeed, light fields record illumination of light rays emitted by a scene in different orientations. The captured data for a static light field is represented by a 4D function $LF(m,n,x,y)$, and contains redundant information in both the spatial and angular dimensions. Those correlations could in principle be represented by a *huge* non separable graph connecting pixels within and across views of the entire light field. The basis functions of a graph Fourier transform [12] could then be used to decorrelate the color signal residing on the graph vertices. However, such a graph would have a very high number of vertices, each vertex corresponding to a light ray. This makes the diagonalization of the laplacian matrix unfeasible, hence, the computation of the graph Fourier transform not practical.

To lower the dimensionality of the problem, we propose to partition the big graph structure into smaller ones that are coherent and correlated inside and across the views. This can be viewed as cutting unreliable edges from the *global* graph. To perform this partitioning, we group similar pixels within and across views based on the concept of super-rays defining the supports of the set of local graph transforms. The concept of super-ray has been introduced in [13] as an extension to light fields of the concept of super-pixels.

The authors in [14] used super-rays as the supports of separable shape-adaptive Discrete Cosine Transform (DCT). Super-pixels are used in [15] as the supports of local graph transforms, and tested in a predictive scheme based on view synthesis. The angular transform is however applied on super-pixels that are co-located on all views, hence not exploiting scene geometry, due to the difficulty to design separable graph transforms that at the same time follow the scene geometrical information and preserve angular correlations. We come back on this point in the sequel.

In this paper, we address the problem of designing local super-ray based non separable and separable graph transforms following the scene geometry. Towards this goal, we first propose a specific super-ray construction method to limit shape variations of the super-pixels forming a given super-ray. Despite the local support of limited size defined by the super-rays, the Laplacian matrix remains of high dimension and its diagonalization to compute the transform eigen vectors is computationally expensive. An intuitive way to solve this problem is to perform the transform in a separable manner: a

This work has been supported in part by the EU H2020 Research and Innovation Programme under grant agreement No 694122 (ERC advanced grant CLIM).

first spatial transform applied per super-pixel inside each view, then an angular transform between corresponding super-pixels across the views to capture angular dependencies. We have however observed that if the shape of the super-ray undergoes a slight change between views, the basis functions computed from the graph laplacian have very different forms from one super-pixel to the corresponding ones in the other views, resulting in a decreased correlation between spatial transform coefficients.

The difficulty is therefore how to optimize the spatial transforms applied on each super-pixel of the super-ray in such a way that the angular correlation is well preserved. Preserving angular correlation is important in order to best compact the light field energy. The angular correlation is preserved, only if the eigen vectors of the spatial transforms computed independently on different shapes (the super-pixels forming the super-ray) are reasonably consistent, i.e. only when the shapes of the transform supports are approximately isometric. We propose in this paper a novel method to optimize the spatial transforms in such a way that the basis functions approximately diagonalize their respective Laplacians while being coherent across the views, given the scene geometry.

Experimental results show that the proposed super-ray construction method yields, for the light fields considered in the tests, up to 60 percent coherent supports out of all super-rays, which facilitates the application of a separable graph transform. The results also show that the optimized separable graph transform yields higher energy compaction, and significant rate-distortion performance gains, compared to the non optimized separable transform, when some super-rays are shape-varying across the views. The proposed simple coding scheme based on these local separable transforms is shown to outperform light field coding schemes based on HEVC-lozenge and JPEG Pleno [2] at high bitrate following the common test conditions.

In this paper, the contributions are as follows:

- We propose a novel light field representation approach based on geometry-aware local graph transforms with a support defined by super-rays.
- We define the notion of separable graph transforms that we apply locally in each super-ray. This allows us to capture both spatial and angular dependencies inside the light field.
- We develop a graph optimization method with a geometric association of nodes. We design consistent transforms for more than 2 graph supports to deal with the problem of inconsistent basis functions when the corresponding super-pixels inside different views are not isometric. The optimization is performed per group of frequency bands to reduce complexity and by fixing a reference to limit error propagation. Using the consistent transforms allows us to preserve angular correlations, and thus a good energy compaction of the separable graph transforms.

II. Related work

We first briefly review prior work on graph transforms design for signal (and in particular image) energy compaction, problem related to the core of the paper. For sake of completeness, the proposed transforms being validated in a complete coding scheme, we also give a brief overview of recent work on light field compression.

A. Graph Transforms

Recently, graph signal processing has been applied to different image and video coding applications, especially for piecewise smooth images. In [16], [17], the authors propose a graph-based coding method where the graph weights are defined considering pairwise similarities between pixel intensities. Another efficient graph construction method has been proposed in [18] for piecewise smooth images. For each signal in a block, they select the Graph Fourier Transform minimizing the rate distortion cost. A signed graph Fourier transform has also been proposed in [19] for depth map coding, accounting for negative weights between pixels.

For natural images, most of the work has focused on designing sparse graphs or using graph templates that capture principal gradient-based structures in images [20][21]. This is mostly useful in textured images. While most of the aforementioned transform coding strategies did not account for the graph coding cost, in a later work [22], a rate-distortion optimized graph learning approach has been proposed to code natural images while taking into account both the sparsity of the transformed coefficients and the graph coding cost. Several graph based approaches have also been proposed to code intra and inter predicted residual blocks in video compression, using generalized graph Fourier transform [23], simplified graph templates transforms [24], or separate line graph based transforms [25].

In this paper, we build graphs that follow the scene geometry and we then propose separable graph based transforms that best exploit light fields spatial and angular correlation.

B. Light Fields Compression

Existing light fields compression solutions can be broadly classified into two categories: approaches directly compressing the lenslet images or approaches coding the views extracted from the raw data. Methods proposed for compressing the lenslet images mostly extend HEVC intra coding modes by adding new prediction modes to exploit similarity between lenslet images (*e.g.* [26], [27], [8], [9]). The authors in [11] propose a lenslet-based compression scheme that uses depth, disparity and sparse prediction followed by JPEG-2000 residue coding.

The second category of methods consists in encoding the set of views which can be extracted from the lenslet images after de-vignetting, demosaicing and alignment of the micro-lens array on the sensor, following e.g. the raw data decoding pipeline in [28]. Several methods code the views as pseudo video sequences using HEVC [3], [1], or the latest JEM coder [29], or extend HEVC to multi-view coding [30]. Low rank models as well as local Gaussian mixture models in the 4D rays space are proposed in [4] and [31] respectively. View synthesis based predictive coding has also been investigated in [32] where the authors use a linear approximation computed

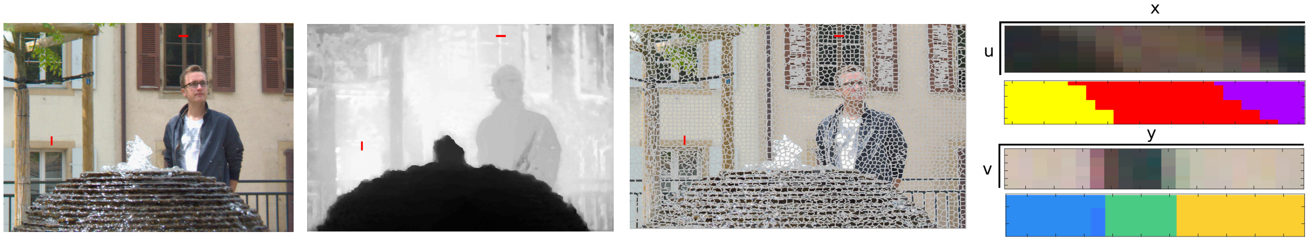


Fig. 1: The result of our proposed light field segmentation for a dense light field *Fountain Vincent* (from EPFL light fields dataset) with an estimated disparity. From left to right, the original view $I_{1,1}$, the disparity map of the original view, the super-pixel segmentation of $I_{1,1}$ and an example of vertical and horizontal epipolar segments taken from both original 4D RGB light field and the 4D segmentation labels (The red lines inside the images show from where the epipolar line is extracted).

with Matching Pursuit for disparity based view prediction. The authors in [33] and [34] use instead a the convolutional neural network (CNN) architecture proposed in [35] for view synthesis and prediction. The prediction residue is then coded using HEVC [33], or using local residue transforms (SA-DCT) and coding [34]. The proposed transforms could also be used for residue coding. However, to best assess their de-correlation advantage, in the experiments reported below, they are directly applied on the color values of the entire 4D light field data.

III. Super-rays and Graph construction

The compression efficiency of any coder based on block partitioning and transform coding does undeniably depend on the way the partitioning is done, and on how the resulting segmentation adheres to object boundaries. While traditional transforms such as 2D DCT applied on a square or rectangular support may fail due to high frequencies captured on the object boundaries, here we rely on a segmentation of the entire 4D light field into super-rays.

A. Light field Segmentation in Super-Rays

Segmentation is an important step of many editing algorithms. While this problem has been widely addressed for 2D images and videos, a few methods exist for light fields [36], [37], [38]. The regions or segments extracted by these methods, often corresponding to objects in the scene, are too large for defining local graph transform supports, as targeted here for reducing the complexity of the basis function computation. To overcome this problem, it is natural to consider instead light field over-segmentation. In [39], a depth estimation is used to propagate an initial over-segmentation into super-pixels of a reference view to all the views of the light field. The authors use the SLIC algorithm described in [40] for the reference view over-segmentation. This initial segmentation is then iteratively refined by optimizing an energy function based on segmentation smoothness inside and between the views along with a color, position and disparity uniformity prior. In this paper, we consider instead the concept of super-ray introduced in [41] as an extension of super-pixels [40] to group light rays coming from the same 3D object, *i.e.* to group pixels having similar color values and being close spatially in the 3D space. The method performs a k-means clustering of all light rays based on color and distance in the 3D space. To deal with dis-occlusions, a slightly modified formulation is proposed in [14] where the dense depth information is also used in the clustering. When the depth information is not fully reliable, this method results in inconsistent super-rays across views. In addition, the signalling cost of such a global light field segmentation is high. In order to make the super-rays more consistent across the views, we suggest a modified version where we compute super-pixels in the top-left view as shown in Figure 1. Then, using the disparity map, we project the segmentation labels to all the other views. Namely, having a segmentation map in the top left view and the corresponding disparity map, we compute the median disparity per super-pixel, and use it to project the segmentation mask to the other views. More precisely, the algorithm proceeds row by row. In the first row of views, we perform horizontal projections from the top-left $I_{1,1}$ to the $N-1$ views next to it. For each other row of views, a vertical projection is first carried out from the top view $I_{1,1}$ to recover the segmentation on view $I_{m,1}$, then $N-1$ horizontal projections from $I_{m,1}$ to the $N-1$ other views are performed, as shown in Figure 2.

An example of segmentation is shown in Figure 2, where we show a cropped area consisting of both background and foreground objects. The red and green super-pixels are computed in the initial segmentation of the top left view with SLIC. The blue are super-pixels obtained after disparity-based projection. At the end of each projection, some shapes are projected in all the views without interfering with others. Those typically represent flat regions inside objects (for example, the super-pixel in red in $I_{1,1}$). While others, mainly consisting of occluded and occluding segments end up superposed in some views. In this case, the occluded pixels are assigned the label of the neighboring super-ray corresponding to the foreground objects (*i.e.* having the higher disparity). As for appearing pixels, they will be clustered with the background super-rays (*i.e.* having the lower disparity). An example of super-ray that ends up with different shapes in the views is marked in green in the segmentation of $I_{1,1}$. The difference is not very remarkable since we deal with dense light fields where the disparity is not very high, it is a matter of one or two pixels at most.

This method performs well on light fields with small baselines. There are however some limitations that appear in the case of wider baselines. We may end up in this case with more occlusions due to higher disparities. With large disparities, objects (or pixels) that are completely occluded in the top-left view but visible in other views cannot be constructed in super-rays.

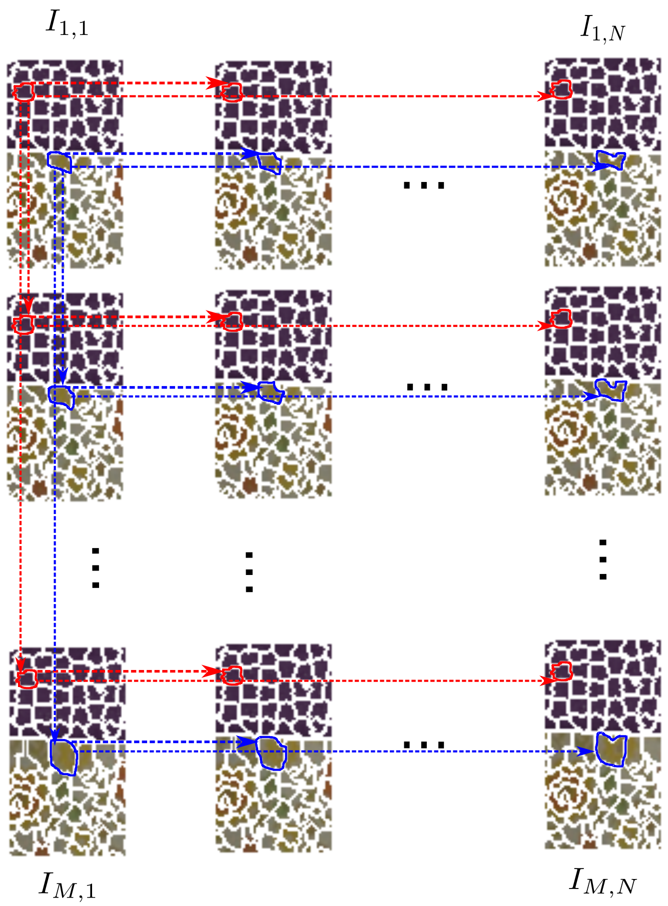


Fig. 2: Image showing the super-ray construction. The algorithm proceeds row by row. In the first row, only horizontal projections are performed. In every other row, first a vertical projection then $N-1$ horizontal projections are performed. The red super-pixel in $I_{1,1}$ is consistent across views, whereas the super-pixel in blue is shape-varying.

B. Graph Construction

In order to jointly capture spatial and angular correlations between pixels in the light field, we first consider a local non separable graph per super-ray. More precisely, if we consider the luminance values in the whole light field and a segmentation map S, the k^{th} super-ray SR_k can be represented by a signal $f_k \in \mathbb{R}^{N_k}$ defined on an undirected connected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ which consists of a finite set $\mathcal{V}$ of vertices corresponding to the pixels at positions $m, n, x, y s.t. S(m,n,x,y) = k$. A set $\mathcal{E}$ of edges are built as follows.

We first connect each pixel (m,n,x,y) in the set $\mathcal{V}$ and its 4-nearest neighbors in the spatial domain (i.e. the top, bottom, left and right neighbors with coordinates $(m,n,x-1,y)$, $(m,n,x+1,y)$, $(m,n,x,y-1)$, $(m,n,x,y+1)$). A pixel can have a maximum of four spatial connections if the four neighbors belong to the set $\mathcal{V}$, and can have less if it is on the border of the super-pixel.

We then find the median disparity value d of the pixels inside the super-ray k in the top-left view. Using this disparity value, we project each pixel in super-ray k with coordinates (m,n,x,y) in the 4 nearest neighboring views (i.e. the top, bottom, left and right neighboring view). We end up with four projected pixels with coordinates $(m-1,n,x-d,y)$, $(m+1,n,x+d,y)$, $(m,n-1,x,y-d)$, $(m,n+1,x,y+d)$. If a projected pixel belongs to the set of vertices $\mathcal{V}$, then we connect it to the original pixel (m,n,x,y). In this way, a maximum of four angular connections can be found for each pixel if the pixel is not occluded in the neighboring views. The weights of all connections are set to 1. An illustrative example of a graph built inside a super-ray is shown in Figure 3 for four views.

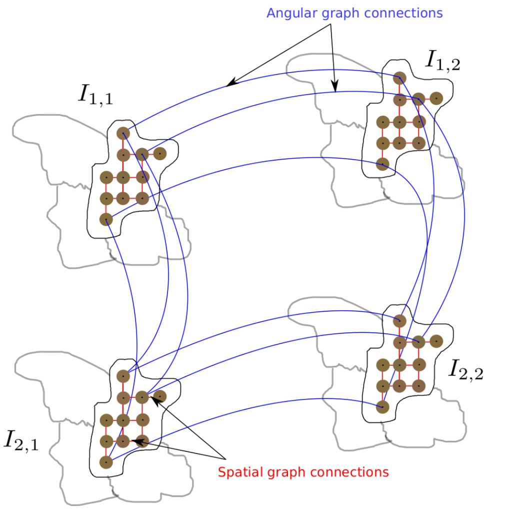


Fig. 3: Example of edges drawn inside a super-ray. We can see the connections within super-pixels in each view (i.e. spatial connections in red), as well as connections between pixels belonging to different views (i.e. angular connections in blue, only a subset of the connections is shown for illustration purpose). The color assigned to the vertices are the color values of the light field.

IV. Graph Transforms

In this section, we focus on the design of suitable transforms for the signals (color or residues) residing on the local graphs defined above.

A. Non Separable Graph Transform

Let us consider the k^{th} super-ray SR_k and its corresponding local graph $\mathcal{G}$. We start by defining its adjacency matrix $\mathbf{A}$ with entries $A_{mn} = 1$, if there is an edge $e = (m,n)$ between two vertices m and n, and $A_{mn} = 0$ otherwise. The adjacency matrix is used to compute the Laplacian matrix $\mathbf{L} = \mathbf{D} - \mathbf{A}$, where $\mathbf{D}$ is a diagonal degree matrix whose i^{th} diagonal element D_{ii} is equal to the sum of the weights of all edges incident to node i. The resulting Laplacian matrix $\mathbf{L}$ is symmetric positive semi-definitive and therefore can be diagonalized as:

$$\mathbf{L} = \mathbf{U}^\top \mathbf{\Lambda} \mathbf{U} \tag{1}$$

where $\mathbf{U}$ is the matrix whose rows are the eigenvectors of the graph Laplacian and $\mathbf{\Lambda}$ is the diagonal matrix whose diagonal elements are the corresponding eigenvalues. The laplacian eigenbases $\mathbf{U}$ are analogous to the Fourier bases in the Euclidean domain and allow representing the signals residing on the graph as a linear combination of eigenfunctions akin to Fourier Analysis. This is known as the Graph Fourier transform. For the signal f_k defined on the vertices of the local graph, the transformed coefficients vector $\hat{f}_k$ is defined in [12] as:

$$\hat{f}_k = \mathbf{U} f_k \tag{2}$$

The inverse graph Fourier transform is then given by

$$f_k = \mathbf{U}^\top \hat{f}_k \tag{3}$$

Although this would be the ideal decorrelating transform for the signal, the Laplacian of such graph, despite the locality, remains of high dimension (almost 6000 nodes per super-ray) leading to a high transform computational cost. To limit the computational cost, we then consider separable local transforms.

B. Coherent Separable Graph Transform

The separable graph transform is defined by a first spatial transform followed by a second angular transform as detailed below.

1) First spatial graph transform: If we consider the luminance values in only one sub-aperture image v of the light field and a segmentation map S, the k^{th} super-ray $SR_{k,v}$ can be represented by a signal $f_{k,v} \in \mathbb{R}^{N_{k,v}}$ defined on an local spatial graph with only connections in the spatial domain (*i.e.* between the neighboring pixels in a super-pixel, but not across the views in a super-ray). $N_{k,v}$ denotes the number of pixels in sub-aperture v that belong to the super-ray k. The matrix $\mathbf{U}_{k,v}$, being the eigen-vectors of the spatial laplacian $\mathbf{L}_{k,v}$, is used to compute the first spatial graph transform : For the signal $f_{k,v}$ defined on the vertices of the graph, the transformed coefficients vector $\hat{f}_{k,v}$ is defined in [12] as:

$$\hat{f}_{k,v} = \mathbf{U}_{k,v}^\top f_{k,v} \tag{4}$$

The inverse spatial graph Fourier transform is then given by

$$f_{k,v} = \mathbf{U}_{k,v} \hat{f}_{k,v} \tag{5}$$

2) Second angular graph transform: In order to capture inter-view dependencies and compact the energy into fewer coefficients, we perform a second graph based transform, in the angular dimension. Note that, for a given super-ray, we do not necessarily have the same number of pixels, hence coefficients resulting from the spatial transforms, in all the views. For a given band b (coefficients corresponding to the b^{th} eigenvectors of the spatial transforms), we construct a graph of N_b vertices corresponding to the views where the band b exists. Edges are drawn between each node and its direct four neighbors. Isolated nodes are connected to their nearest neighbor.

The Adjacency is used to compute the inter-view angular unweighted Laplacian as $\mathbf{L}_k^b = \mathbf{D}_k^b - \mathbf{A}_k^b$ with $\mathbf{D}_k^b$ the degree matrix. $\mathbf{L}_k^b$ can be diagonalized as:

$$\mathbf{L}_k^b = \mathbf{U}_k^b \mathbf{\Gamma} {\mathbf{U}_k^b}^\top \tag{6}$$

For a specific band number b and super-pixel k, the band signal is defined as $\hat{f}_k^b = \{\hat{f}_{k,v}(b), \quad v \subseteq [1, \cdots, M \times N]\} \in \mathbb{R}^{N_b}$. The angular Graph Transform consists of projecting the signal onto the eigenvectors of $\mathbf{L}_k^b$ as:

$$\hat{\hat{f}}_k^b = {\mathbf{U}_k^b}^\top \hat{f}_k^b \tag{7}$$

The inverse angular Graph Transform is then given by

$$\hat{f}_k^b = \mathbf{U}_k^b \hat{\hat{f}}_k^b \tag{8}$$

3) Coherence of spatial graph transforms in corresponding super-pixels: The spatial graphs in the different super-pixels forming one super-ray may not have the same shape. Furthermore, we have observed that for a specific super-ray, when the spatial graph topology in the corresponding super-pixels undergoes a slight change, the basis functions of each spatial graph transform are different and thus incompatible with each others (refer to Figure 4 before optimization), resulting in decreased correlation of the spatial transform coefficients across views. This is shown in the sequel to severely decrease the efficiency of the angular transform.

Basically, during the diagonalization procedure, the eigenfunctions are only defined up to sign flips for Laplacians having a simple spectrum (if the eigenvalues have a multiplicity of 1, for example connected graphs). Therefore, even having the same shape in two different views, we may end up with two opposite eigen-vectors for a specific eigenvalue during the diagonalization.

Moreover, eigenvectors computed independently on two different shapes (i.e. corresponding to two different Laplacians) can be expected to be reasonably consistent only when the shapes are approximately isometric. Whenever this assumption is violated, it is impossible to expect that the k^{th} eigenvector of a Laplacian $\mathbf{L}_{si}$ in view i will correspond to the k^{th} eigenvector of another Laplacian $\mathbf{L}_{sj}$ in view j. If the basis functions do not behave consistently on the corresponding points of the two shapes, the two signals defined on those two Laplacians will be projected onto incompatible basis functions (see Figure 4), and therefore we cannot guarantee any correlation to be preserved after performing the first spatial graph transform.

4) Coherent spatial graph transform: In order to overcome those limitations, we consider an approach which aims at finding *coupled* basis functions. We propose a graph optimization which differs from [42] in several manners:

- First, the association and correspondence points between the supports (i.e. super-pixels in different views) are defined based on the scene geometry and not manually as in [42]).
- Also, we design consistent transforms for more than 2 supports ($N = 2$ in [42]) by fixing one reference and optimizing the other basis functions in other views with respect to the reference in order to reduce the complexity of the overall problem.
- Third, thanks to fixing one reference, we limit the error propagation between different basis functions of different supports that may appear if we perform an iterative optimization.

More precisely, suppose that, in the super-ray k in a reference view o and a target view i, we have two Laplacians

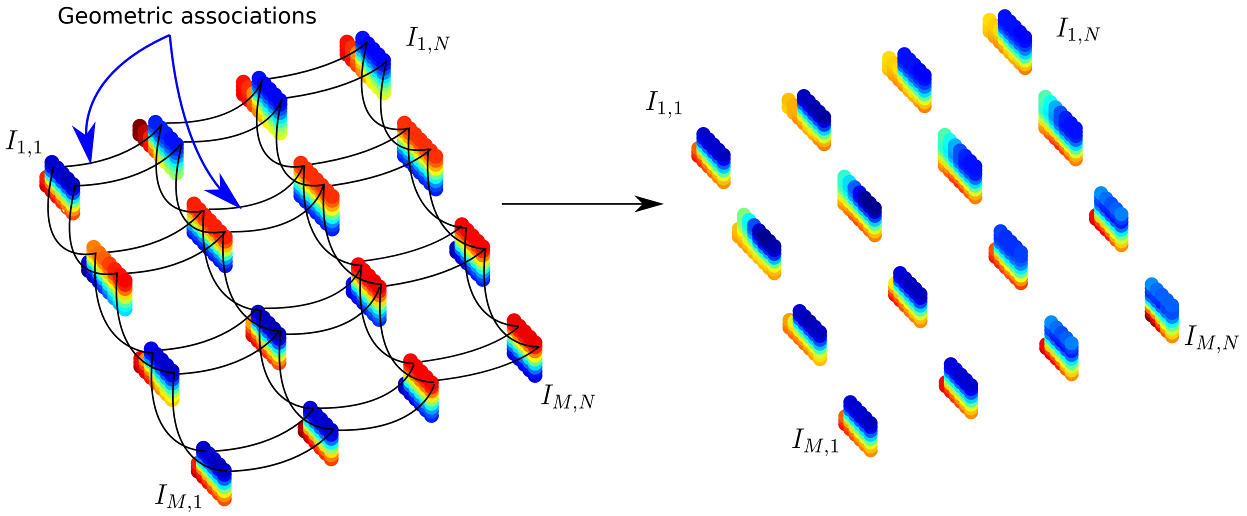
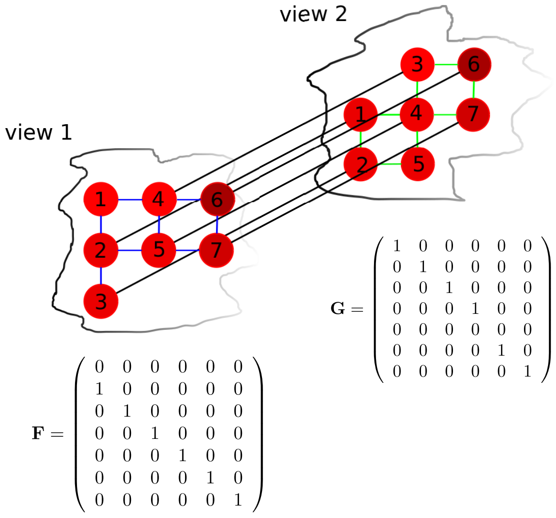


Fig. 4: Second eigenvector of different super-pixels belonging to the same super-ray before and after optimization. Only two sets of geometric associations are shown for illustration purposes.



$$\mathbf{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$\mathbf{F} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Fig. 5: Example of correspondence functions $\mathbf{F}$ and $\mathbf{G}$ computed for a small shape-varying super-pixel. The graph nodes are labeled in both graphs following a vertical scan line. In the second view, we have one disappearing node and another appearing one with respect to the first view.

$\mathbf{L}_{k,o}$ and $\mathbf{L}_{k,i}$ with size $(n_o \times n_o)$ and $(n_i \times n_i)$ respectively. They can be diagonalized as:

$$\begin{aligned} \mathbf{L}_{k,o} &= \mathbf{U}_{k,o}\boldsymbol{\Lambda}_o\mathbf{U}_{k,o}^\top \\ \mathbf{L}_{k,i} &= \mathbf{U}_{k,i}\boldsymbol{\Lambda}_i\mathbf{U}_{k,i}^\top \end{aligned} \tag{9}$$

If the two Laplacians are equal, we make sure that their eigenvectors are compatible with sign flips accordingly. We check the first value of the each eigenvector and flip its sign if the value is negative.

In the case where the super-pixel shapes in the sub-aperture images are not isometric, we propose to diagonalize one specific spatial graph Laplacian $\mathbf{L}_{k,o}$ and find $\mathbf{U}_{k,o}$. Then, we search for basis vectors $\hat{\mathbf{U}}_{k,i}$ that approximately diagonalize any other spatial graph Laplacian $\mathbf{L}_{k,i}$ and at the same time preserve correlations after the transform. We start by posing the problem as follows:

$$\begin{aligned} \hat{\mathbf{U}}^*_{k,i} = &\min_{\hat{\mathbf{U}}_{k,i}} \; off(\hat{\mathbf{U}}_{k,i}^\top\mathbf{L}_{k,i}\hat{\mathbf{U}}_{k,i}) + \alpha\left\|(\mathbf{F}^\top\mathbf{U}_{k,o} - \mathbf{G}^\top\hat{\mathbf{U}}_{k,i})\right\|_F^2, \\ &\text{s.t. } \hat{\mathbf{U}}_{k,i}^\top\hat{\mathbf{U}}_{k,i} = \mathbf{I}. \end{aligned} \tag{10}$$

where we seek to minimize the weighted sum of two terms subject to the orthonormality constraint of the computed basis functions $\hat{\mathbf{U}}_{k,i}$. The first term is a diagonalization term that aims at minimizing the energy residing on off-diagonal entries ($off(\mathbf{M}) = \sum_{i\neq j} m_{ij}$). The second term aims at enforcing coherence between the two spatial graph transforms and is defined as follows.

Based on the geometry information we have in hand, we can actually define, *a priori*, a set of correspondences between $\mathbf{L}_{k,o}$ and $\mathbf{L}_{k,i}$. More precisely, we suppose that we have a set of p corresponding functions represented by matrices $\mathbf{F}$ and $\mathbf{G}$ of sizes $(n_0 \times p)$ and $(n_i \times p)$ respectively. An example of $\mathbf{F}$ and $\mathbf{G}$ is shown in figure 5.

The basis functions of both Laplacians are supposed to be consistent if the Fourier coefficients of the functions $\mathbf{F}$ and $\mathbf{G}$ on $\mathbf{L}_{k,o}$ and $\mathbf{L}_{k,i}$ are approximately equal i.e. if $\mathbf{F}^\top\mathbf{U}_{k,o} \simeq \mathbf{G}^\top\hat{\mathbf{U}}_{k,i}$. To avoid over-determining the problem, we use the farthest point sampling technique restricting the correspondence points to a maximum of 15 points.

If we parametrize the new basis functions of $\mathbf{L}_{k,i}$ as being a linear combination of the old basis functions, we can write $\hat{\mathbf{U}}_{k,i} = \mathbf{U}_{k,i}\mathbf{B}$ where $\mathbf{B}$ is a matrix of combination coefficients, that plays a role of reflecting and rotating the original basis vectors in $\mathbf{U}_{k,i}$ so that they will align the best way with $\mathbf{U}_{s_0}$ while almost diagonalizing the laplacian $\mathbf{L}_{k,i}$. Using the diagonalizing property of $\mathbf{U}_{k,i}$, we can re-write Equation (10) as

$$\begin{aligned} \mathbf{B}^* = &\min_{\mathbf{B}} \; off(\mathbf{B}^\top\boldsymbol{\Lambda}_i\mathbf{B}) + \alpha\left\|(\mathbf{F}^\top\mathbf{U}_{k,o} - \mathbf{G}^\top\mathbf{U}_{k,i}\mathbf{B})\right\|_F^2, \\ &\text{s.t. } \mathbf{B}^\top\mathbf{B} = \mathbf{I}, \end{aligned} \tag{11}$$

It is important to note that the first term of the above problem does not guarantee a preserved increasing order of the eigenfunctions. It is therefore more convenient to use

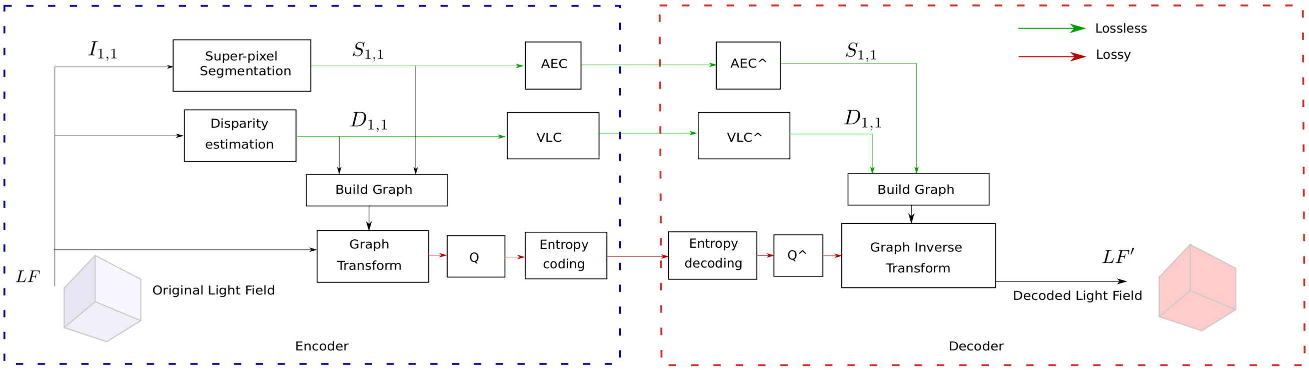


Fig. 6: Overview of proposed coding scheme.

an alternative penalty equal to $\left\|\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{\Lambda}_i\right\|_F^2$ that relates not only to the diagonalization property, but also to the distribution of the energies across the basis functions after the optimization.

$$\mathbf{B}^* = \min_{\mathbf{B}} \left\|\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{\Lambda}_i\right\|_F^2 + \alpha \left\|(\mathbf{F}^\top \mathbf{U}_{k,o} - \mathbf{G}^\top \mathbf{U}_{k,i}\mathbf{B})\right\|_F^2, \quad \text{s.t. } \mathbf{B}^\top \mathbf{B} = \mathbf{I}, \tag{12}$$

The problem in Equation (12) is a non linear optimization problem with an orthogonality constraint, which can be solved by iterative minimization algorithms. In our case, we used Matlab optimization toolbox (interior point method of the *fmincon* function) to solve it. The gradients of the cost function terms are given in appendix A.

Since we are dealing with large datasets and a large number of super-rays, it is convenient to use parallel computing to independently compute eigen-basis for the different super-rays. Also, in contrast with the way the optimization is performed in [42], in order to reduce the complexity of the problem, we propose to split it into smaller problems that are independent: we pick a small number z of eigenvectors to be optimized at a time. Then, for each disjoint group l of z eigenvectors in $\mathbf{U}_{k,i}$, we formulate a sub-problem by expressing z new eigenvectors as a linear combination of z old eigenvectors. Noticing that $\mathbf{U}_{k,i} = [\widetilde{\mathbf{U}}^1_{k,i}, \widetilde{\mathbf{U}}^2_{k,i}, ..., \widetilde{\mathbf{U}}^l_{k,i}]$ and

$$\mathbf{\Lambda}_i = \begin{pmatrix} \widetilde{\mathbf{\Lambda}}^1_i & 0 & 0 & 0 \\ 0 & \widetilde{\mathbf{\Lambda}}^2_i & 0 & 0 \\ 0 & 0 & .. & 0 \\ 0 & 0 & 0 & \widetilde{\mathbf{\Lambda}}^l_i \end{pmatrix} \tag{13}$$

For each group of z eigenvectors, we find $\widetilde{\mathbf{B}_l}$ of size $(z \times z)$ that will minimize the objective function on the subset of eigenvectors.

$$\widetilde{\mathbf{B}}^*_l = \min_{\widetilde{\mathbf{B}}_l} \left\|\widetilde{\mathbf{B}}_l^\top \widetilde{\mathbf{\Lambda}}^l_i \widetilde{\mathbf{B}}_l - \widetilde{\mathbf{\Lambda}}^l_i\right\|_F^2 + \alpha \left\|(\mathbf{F}^\top \widetilde{\mathbf{U}}^l_{k,0} - \mathbf{G}^\top \widetilde{\mathbf{U}}^l_{k,i} \widetilde{\mathbf{B}}_l)\right\|_F^2, \quad \text{s.t. } \widetilde{\mathbf{B}}_l^\top \widetilde{\mathbf{B}}_l = \mathbf{I}, \tag{14}$$

At the end of the optimization stage, most of the eigenvectors are thereby compatible across views and the transform will necessarily preserve any correlation already observed between views. An example of the second eigenvector of a super-ray before and after optimization is shown in Figure 4. While eigenvectors corresponding to higher frequencies are harder to adjust, the low frequency eigenvectors can be easily optimized. In our application, this is not a big problem since we have a high energy compaction in lower frequency bands, and those are the bands that matter the most for reconstruction. After performing the segmentation and two transforms, most of the energy of the color signal is indeed expected to be concentrated in a very small number of coefficients. In the following section, we aim at exploiting this energy compaction property to efficiently code the redundant information present in the light field using the tools introduced above.

V. Light Field Coding Scheme

The overall steps of the compression algorithm are shown in Figure 6. The top left view of the Light Field is separated into uniform regions using the SLIC algorithm to segment the image into super-pixels [40], and its disparity map is estimated. Using both the segmentation map and the geometry information, we construct consistent super-rays in all views as explained in section III. The non separable and separable transforms described above are then locally applied on each super-ray. The transformed coefficients are then quantized and encoded to be stored or transmitted. The segmentation map of the reference view and a disparity value per super-ray also need to be transmitted as side information to the decoder.

1) Segmentation map and disparity values coding: The segmentation map of the reference view is encoded using the arithmetic edge coder proposed in [43]. The contours are first represented by differential chaincode [44] and divided into segments. Then, to efficiently encode a sequence of symbols in a segment, *AEC* uses a linear regression model to estimate probabilities, which are subsequently used by the arithmetic coder. Disparity values are encoded using an arithmetic coder.

2) Grouping and transform coefficients coding: The energy compaction is not the same in all super-rays. This can be explained by the fact, that the segmentation may not well adhere to object boundaries, resulting in high angular frequencies after optimization of the first spatial transform.

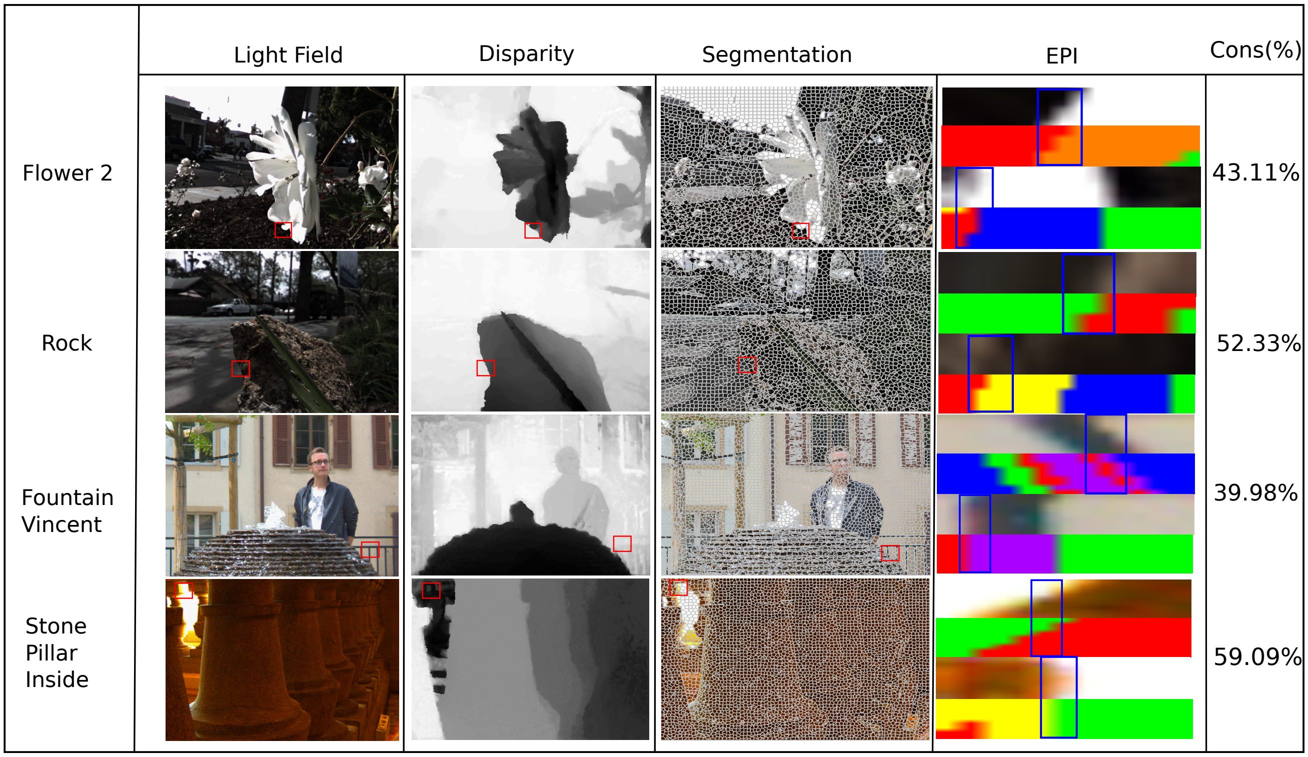


Fig. 7: Consistent Super-rays performance:In the first three columns, we have the original top left corner view, its corresponding disparity map and super pixel segmentation using the SLIC algorithm [40] respectively. In the fourth column, we show horizontal and vertical epipolar segments taken both from the 4D light field color and our final labeling in specific regions of the image(the red blocks). We use the prism color map in Matlab for the segmentation, just for illustration purposes.

To optimize the coding performance, we divide the set of super-rays into four classes, where each class is defined according to an energy compaction criterion.

First, we learn a scanning order. More precisely, at the end of the two graph transform stages, coefficients are grouped into a three-dimensional array $\mathbf{R}$ where $\mathbf{R}(i_{SR}, i_{bd}, v)$ is the v^{th} transformed coefficient of the band i_{bd} for the super-ray i_{SR}. Using the observations on all the super-rays in some training datasets (*Flower1*,*Friends*), we can find the best ordering for scanning and quantization. We sort the variances of coefficients with enough observations in decreasing order and we follow this decreasing order during the scanning process.

Then, each super-ray with N coefficients belongs to class i if the mean energy per high frequency coefficient is less than 1, where the high frequency coefficients are the last $round(N \times i/4)$ coefficients following the scanning order of the super-rays coefficients defined previously. We start by finding the super-rays in the first class than remove them from the search space before finding the other classes, and idem for the following steps.

We code a flag with an arithmetic coder to gives the information of the class of super-rays to the decoder side. In class i, the last $round(N \times i/4)$ coefficients of each super-ray are discarded. The rest of the coefficients are grouped into 32 uniform groups. The quantization step sizes in groups are defined with a rate-distortion optimization taking into account a big number of observed coefficients. At the end of this stage, for each class, each group is coded using the Context Adaptive Binary Arithmetic Coder (*CABAC*) from the HEVC H.265 reference coder.

VI. Experimental analysis

For performance evaluation, we consider real light fields captured by plenoptic cameras from the datasets used in [35] and [45]. We consider the 8×8 central sub-aperture images cropped to 364×524 in [35], and 9×9 cropped to 432×624 from [45] in order to avoid the strong vignetting and distortion problems on the views at the periphery of the light field. The disparity map of the top left view of each light field has been estimated using the method in [46]. The estimated disparity map is used to construct super-rays as described in Section III.

A. Assessment of the proposed super-ray construction method

In this section, we assess how the proposed super-ray construction method deals with occluded and dis-occluded parts, and to which extent the super-rays are consistent despite uncertainty on the disparity information. Figure 7 shows examples of super-rays obtained with different real light fields captured by a Lytro Ilum camera (*Flower 2*, *Rock* used in [35], and *FountainVincent*, *StonePillarInside* used in [45]). In the first three columns, we have the original top left corner view, its corresponding disparity map and super pixel segmentation using the SLIC algorithm [40] respectively. In the fourth column, we show horizontal and vertical epipolar segments taken both from the 4D light field color information and our final segmentation in specific regions of the image (the red blocks). We can see that we are following well the object borders, especially when the disparity map is reliable. Also, we have always attained a high percentage of coherent super-rays across views (higher than 40% as measured with Cons$(\%)$ in the fifth column). More precisely, Cons$(\%)$ gives the percentage of coherent super-rays: A super-ray is coherent when it is made of super pixels having the same shape in all the views, with or without a displacement.

At the end of this segmentation stage, we end up with a segmentation map with consistent super rays in flat objects and shape-varying super-rays mainly on the borders.

B. Analysis of proposed graph based optimized transforms

In this section, we analyze the performance of our optimization process described in section IV-B and its effect on the transform coding efficiency. In all the experiments, for each super-ray we find the super-pixel $\mathbf{L}_{s_o}$ that is on the top-left most of the light field, and fix it as reference for the coupling process. We therefore optimize the maximum number of eigenvectors defined as $floor(\frac{n_0}{10}) \times 10$ with n_0 being the number of pixels in the reference super-pixel. An example of input and output of the coupling process for a shape-varying super-ray is illustrated in Figure 8. We see that the consistency of eigenvectors in the different graphs is much better after our optimization. If we project the light field signal residing in the super-ray on the optimized coupled eigenvectors, the inter-view correlation is better preserved compared to the non optimized eigenvectors.

1) Energy Compaction of the spatial transform: Figure 9 shows the energy compaction observed in the spatial transform domain, then in the spatio-angular transform domain, *i.e.* after performing the first spatial transform and after performing both spatial and angular transforms on the color signal of the light fields. The energy compaction is computed for both optimized and non optimized cases. It denotes the percentage of energy if we keep some of the coefficients and discard others. For the spatial transform, we gather the transform coefficients of all super-pixels, and then we scan them following the intuitive order increasing order of the Laplacian eigenvalues to compute the compaction. For the spatio-angular compaction, we follow the learned sub-optimal scanning order using different observations from the different datasets as explained in section V-2.

If we compare the energy compaction of the spatial transforms only (red and blue curves) for different datasets, we observe that we may loose in terms of energy compaction for some datasets after optimization. In order to explain such loss, we analyze how the graphs are varying under the new basis functions after optimization. An example is shown in Figure 10 where edges between highlighted nodes are added implicitly in the graph after coupling. The new underlying Laplacian is computed as $\hat{\mathbf{L}}_{k,i} = \hat{\mathbf{U}}_{k,i}\mathbf{\Lambda}_{k,i}\hat{\mathbf{U}}^T_{k,i}$.

The underlying assumption behind the optimization procedure is that the signal can be modeled by a modified Gaussian distribution (Gaussian Markov Random Field) with a modified precision matrix which is equivalent to the new Laplacian matrix with some added small weights. Since this procedure is modifying the original graph structure, it may, in some cases, bring some high frequencies.

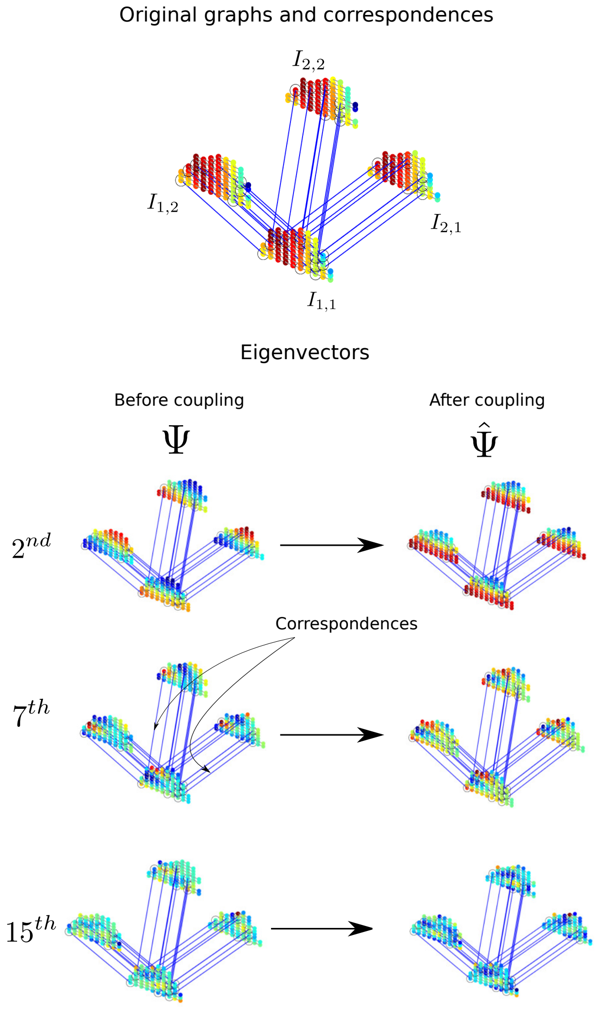


Fig. 8: Illustration of the output of the optimization process for a super-ray in 4 views. The first row corresponds to a super-ray accross four views of the light field. The signal on the vertices correspond to the color values lying on super-pixels corresponding to the same super-ray and the blue lines denote the correspondences. The second to fourth rows are illustrations of basis functions before and after optimization. The signals on the vertices are the eigenvectors values.

2) Correlation and Energy Compaction after angular transform: The gain in compaction after the spatio-angular transform is clear in Figure 9 when we perform the optimization. This is due to the fact that we are able to preserve angular correlations after the spatial transform, which will be subsequently exploited by the angular transform.

In order to assess the performance of our coupling process in preserving the correlation, we draw in Figure 11, the correlation matrices and the covariance matrices for some bands after the first transform with shape-varying super-rays. If we restrict our attention to the first column, We see that after the first transform that is not optimized, we have uncorrelated transform coefficients due to the perturbation of eigenvectors computed on super-pixels having slightly different shapes. This problem is almost resolved with our coupling procedure in the second column, where we can observe more correlation between the coefficients of the same band in neighboring

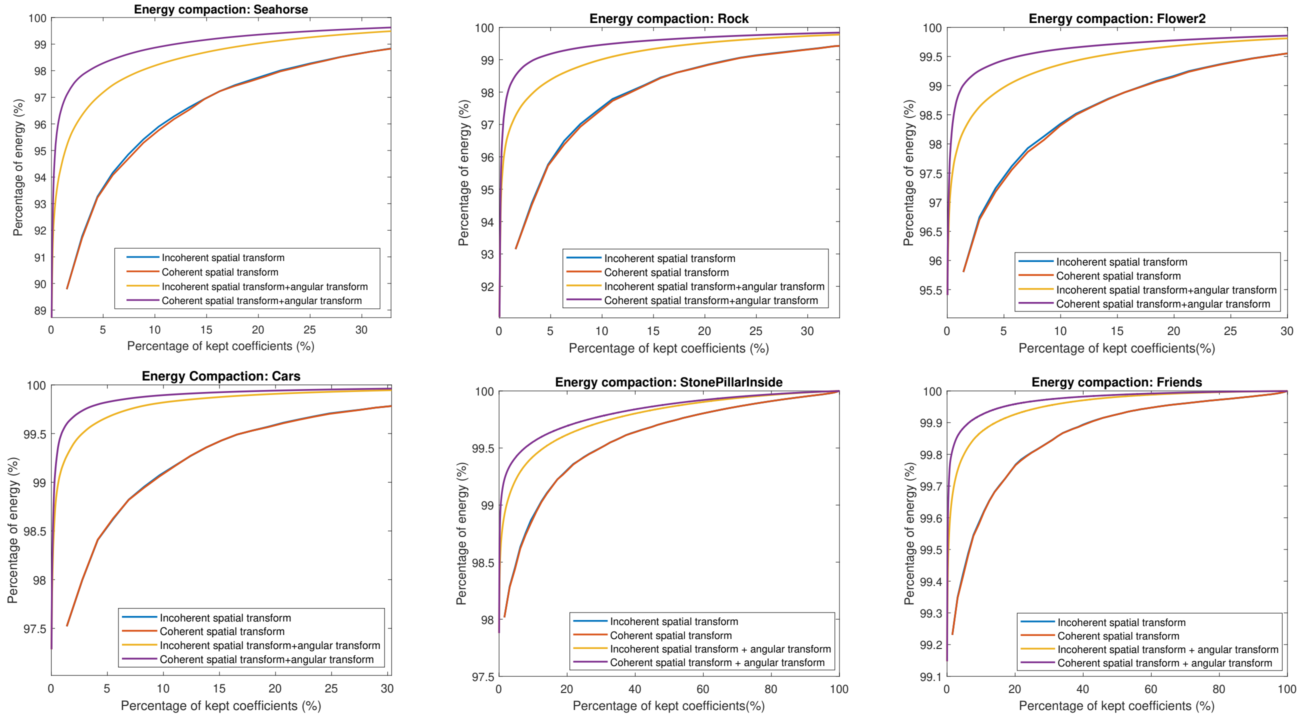


Fig. 9: Energy compaction with or without optimization of the first spatial transform for four datasets (Seahorse, Rock,Flower2 and Cars) from the dataset used in [35] and two others (Friends and StonePillarsInside) taken from the datasets in [45].

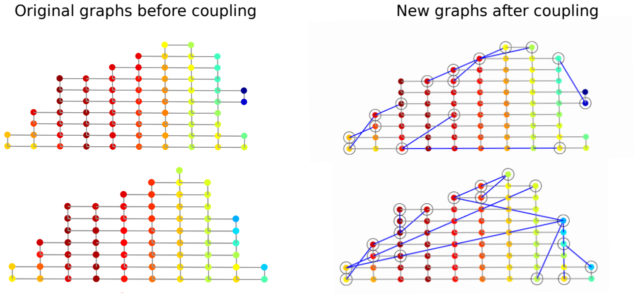


Fig. 10: Image showing the old graphs before coupling and the new graphs after optimization. New edges with absolute weight values larger than 0.04 are shown as blue lines connecting highlighted nodes.

views. Furthermore, the logarithm of the variances (values lying on the diagonal in the covariance matrices) being higher in the first low frequency bands and decreasing when moving further from the DC, shows the energy compaction of the first transform. As for the values of the off-diagonal elements of the covariance matrices, they show how correlated are the transformed coefficients after the first transform inside the views. If we observe the off-diagonal values and compare them with or without optimization, we find out that the optimization performs better for low frequencies than for high frequencies and is therefore more able to retrieve coherent basis functions.

After the second angular transform per band, for both cases with or without optimization, we compute the logarithm of coefficients' variances after the second transform and illustrate it in the third row where the x-axis and y-axis correspond to the band number and the view number respectively. A compaction of the energy in fewer coefficients is observed in the optimized case compared to the non-optimized case, especially when we focus on the top-left region. Some inter-view high frequencies are sometimes still there and might be due to the presence of some super-rays are made of super-pixels that adhere well to borders in some views while not adhering in some others due to disparity rounding effects.

3) Impact of disparity errors: When the disparity information is not reliable, dis-occluded pixels may be clustered with a wrong super-ray, resulting in high frequencies, hence poor energy compaction, after the spatial transforms in those specific regions. However, experiments with synthetic data sets (for which the ground truth depth is known) have shown that, with the considered disparity estimation method [46], the depth map errors lead to a mislabelling for less than 2% of the pixels, which has a negligible effect on the energy compaction and on the RD performance.

4) Impact of super-rays size: The size of super-rays may have an impact on the rate distortion performance especially when the disparity information is reliable and there is a lot of homogeneous objects. If we have large objects, we might want to merge some small super-rays which makes a non separable graph transform practically unfeasible. Here comes the advantage of an optimized separable graph transform where one can define the number of eigenvectors to be optimized depending on the homogeneity of the shape-varying super-rays inside the views. In this case, the segmentation and disparity costs will more likely drop also since we also have less contours and values to code.

In our experiments, however, we use a uniform segmentation into super-pixels. We fix the number of super-rays to 2800 for the light fields in [35], and 4000 for the light fields in [45]. We have observed that when we have a small number of super-rays, the disparity errors may have an impact on the

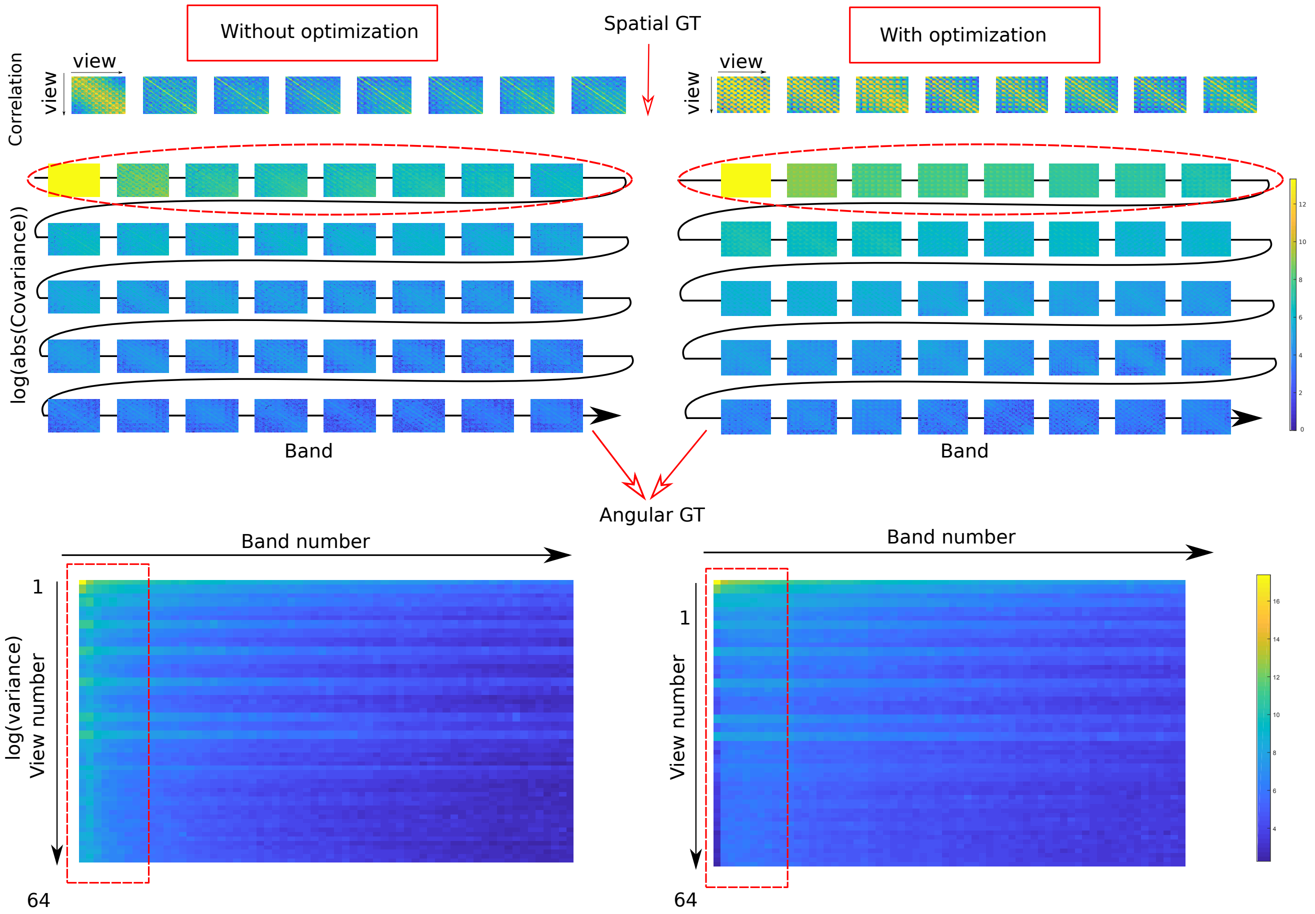


Fig. 11: Advantage of our optimization in terms of energy compaction. The three rows correspond to (1) correlation matrices of the spatial transformed coefficients of the first ten bands, (2) the log of the absolute value of the covariance matrices of the 64 first bands of the spatial transformed coefficients, and (3) the logarithm of the variance of the coefficients after the angular transform, respectively. The two columns show the two cases: without or with our optimization.

compensation and therefore result in a decreased PSNR-Rate performance. On the other hand, having a very large number of super-rays increases the rate needed for segmentation and limits the dimension of each super-ray, resulting in a smaller benefit in terms of de-correlation of the proposed spatio-angular transform.

C. Rate-distortion performance comparative evaluation

We assess the compression performance obtained with our graph based transform coding schemes against four schemes: encoding the views as a video sequence following a lozenge order (HEVC lozenge) [1], or using different prediction orders in the same vein as multi-view coding (HEVC pseudo), [3], JPEG Pleno VM 1.1 software [2] and HLRA [4]

In the simulations, the basic configuration files of JPEG Pleno VM have been used with small changes in order to be applied on 9×9 views. For HEVC-lozenge, the base QPs are set to 20, 26, 32, 38 and a GOP of 4 is used. The HEVC version used in the tests is HM-16.10.

In Figure 12, our coding scheme based on both non separable and separable graph transforms is investigated against HEVC-lozenge [1], JPEG pleno 1.1 [2], HEVC-pseudo ([3]) and HLRA ([4]) for three light fields with 9×9 views, from the ICIP 2017 Grand Challenge [45]. Further experiments are also depicted in Figure 13 for 8×8 light fields [1]. For the separable case, we compare the optimized and the non optimized graph transform. In Table I, we restrict our attention to the optimized separable graph based transform case that we denote by opt-separable GBT scheme that can be applied no matter how big the super-rays are. It shows the rate allocation of our method, at low and high bitrates, for the different light fields.

We can observe that, for most of the light fields used in our tests, the non separable graph transform yields a better rate-distortion performance compared to the separable case for a fixed number of super-rays. While the non optimized graph transform fails to compact the energy of the light field, the optimized graph transform is performing better and sometimes almost catches the non separable case. One major advantage of the separable optimized case is that it can be applied on super-rays of large dimensions without facing the basis functions computational complexity issue of the non separable case. More precisely, the use of the separable transform (including its optimization) leads to a time saving of around 60% compared with the non separable transform. Furthermore, the number of eigenvectors to be optimized can be defined by the encoder and does not have to be necessarily large.

[1] Visual results can be found on http://clim.inria.fr/research/GBT/GBT.html

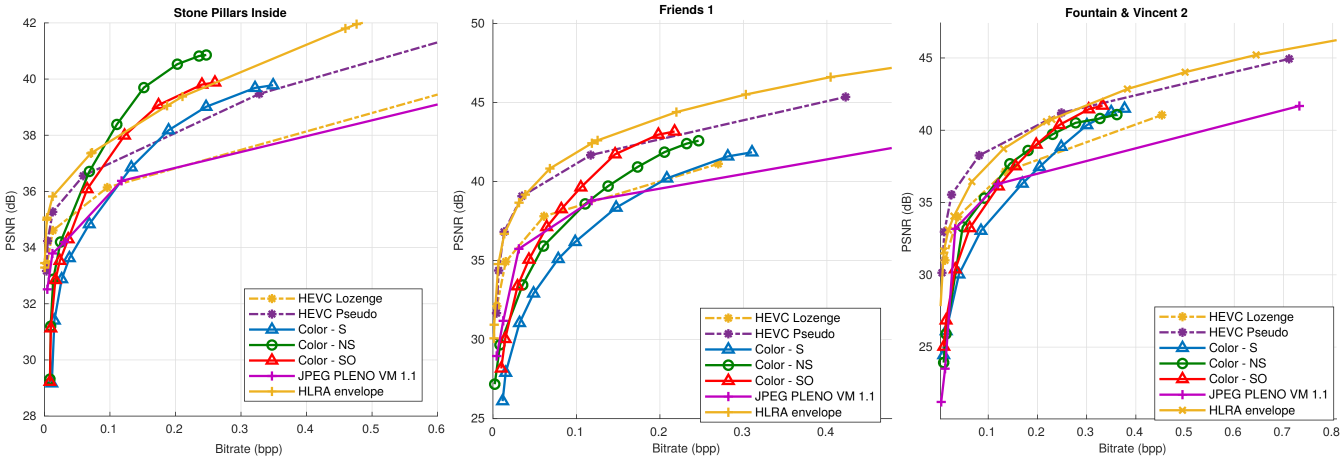


Fig. 12: Rate distortion performance of our graph based coding schemes (Non separable, not optimized and optimized separable graph transforms) compared to HEVC lozenge [1], HEVC pseudo [3], HLRA [4] and JPEG Pleno VM 1.1 [2] for the 9 × 9 light fields used in the ICIP 2017 Grand Challenge [45] following the common test conditions.

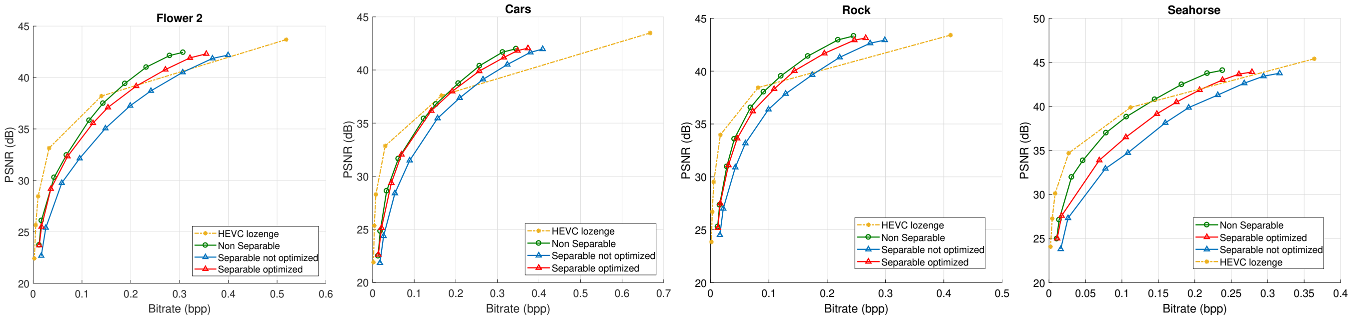


Fig. 13: Rate distortion performance of our graph based coding schemes (Non separable, not optimized and optimized separable graph transforms) compared to HEVC lozenge [1] for the 8 × 8 light fields of [35].

Light Field	Rate allocation(in %) for the opt-separable GBT scheme			
	Overall bitrate	*Segmentation*	*Disparity*	*Coefficients*
Cars (364 × 524)	0.2563 bpp (PSNR = 42.24dB)	2.69%	0.55%	96.76%
	0.0212 bpp (PSNR = 25.23dB)	32.55%	6.60%	60.85%
Flower2 (364 × 524)	0.2710 bpp (PSNR = 40.77dB)	2.69%	0.55%	96.76%
	0.0362 bpp (PSNR = 29.18dB)	20.17%	4.14%	75.69%
Rock (364 × 524)	0.1951 bpp (PSNR = 41.68dB)	4.00%	0.82%	95.18%
	0.0306 bpp (PSNR = 31.10dB)	25.49%	5.23%	69.28%
Seahorse (364 × 524)	0.2302 bpp (PSNR = 42.99dB)	2.65%	0.74%	96.61%
	0.0612 bpp (PSNR = 33.88dB)	9.97%	2.78%	87.25%
Friends (432 × 624)	0.1464 bpp (PSNR = 41.73dB)	3.89%	0.10%	96.01%
	0.0294 bpp (PSNR = 33.38dB)	19.39%	5.10%	75.51%
StonePillarInside (432 × 624)	0.2204 bpp (PSNR = 39.07dB)	2.59%	0.54%	96.87%
	0.0212 bpp (PSNR = 32.85dB)	26.89%	5.66%	67.45%
FountainVincent (432 × 624)	0.2448 bpp (PSNR = 40.37dB)	2.12%	0.57%	97.31%
	0.0330 bpp (PSNR = 30.38dB)	15.76%	4.24%	80.00%

TABLE I: Rate allocation performed by the proposed coding scheme with the optimized separable graph transform. The rate is divided into three parts used for coding the segmentation, disparity and transform coefficients.

Moreover, we can observe that the proposed method outperforms JPEG Pleno VM 1.1 [2] and HEVC lozenge [1] at high bitrate. Figure 12 also shows that it yields lower RD performances when compared with two other reference methods (HEVC-pseudo [3] and HLRA [4]). This is mainly due to the fact that the proposed scheme does not incorporate any spatial mechanism to exploit correlation between the local transform supports while the two reference methods benefit from the efficient HEVC intra prediction mechanisms.

Also, the bitrate allocated to the segmentation and disparity is very large, especially at low bitrate (almost reaching 30 percent for most datasets) and could be further reduced.

Note that the decoder needs to compute the optimized basis functions for the non consistent super-rays, inducing some computational complexity. However, the optimization can be performed independently on each super-ray, in a parallel manner.

VII. Conclusion

In this paper, we have addressed the problem of local geometry-aware graph transform design for light field energy compaction and compact representation. The transform support is based on super-rays constructed in a way that their shape remains coherent across the different views. We have first considered both non separable graph transforms.

Despite the limited size of the transform support, the Laplacian matrix of such graph remains of high dimension and its diagonalization to compute the transform eigenvectors is computationally expensive.

To solve this problem, we then considered a separable spatio-angular transform. We have shown that, when the shape of corresponding super-pixels in the different views undergoes small changes, the basis functions of the spatial transforms are not coherent, resulting in a decreased correlation between spatial transform coefficients. We hence proposed a novel transform optimization method that aims at preserving angular correlation even when the shapes of corresponding super-pixels (i.e. forming one super-ray) are not isometric. This procedure has been shown to increase energy compaction of the separable spatio-angular graph transforms and bring substantial rate-distortion performance gains compared to a non optimized case. The proposed optimized spatio-angular graph transforms can be applied on both color or residual signals and can be easily parallelized to reduce the complexity on the decoder side.

Acknowledgment

The authors would like to thank Elian Dib, Navid Mahmoudian Bidgoli and Pierre Allain from Inria for their help in extracting and running the JPEG Pleno VM 1.1 [2] and HEVC CABAC encoder. Also, we would like to thank Effrosyni Simou and Eda Bayram from EPFL, who were of a great help with various discussions about the subject.

Appendix A
Gradients of the objective function terms

The gradients of the two terms in the optimization of equation 12 are provided below:

$$\begin{aligned}
&\nabla_B \|\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{\Lambda}_i\|_F^2 \\
&= \nabla_B tr\left((\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{\Lambda}_i)^\top (\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{\Lambda}_i)\right) \\
&= \nabla_B tr\left((\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{\Lambda}_i^\top)(\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{\Lambda}_i)\right) \\
&= \nabla_B tr(\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B}\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B}\mathbf{\Lambda}_i \\
&\qquad\qquad - \mathbf{\Lambda}_i^\top \mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} + \mathbf{\Lambda}_i^\top \mathbf{\Lambda}_i) \\
&= 4(\mathbf{\Lambda}_i \mathbf{B}\mathbf{B}^\top \mathbf{\Lambda}_i \mathbf{B} - \mathbf{\Lambda}_i \mathbf{B} \mathbf{\Lambda}_i)
\end{aligned} \tag{15}$$

As for the coupling term, with a similar derivation as the first gradient and using the trace derivation properties in [47], we get:

$$\begin{aligned}
&\nabla_B(\left\|(\mathbf{F}^\top \mathbf{U}_{s_0} - \mathbf{G}^\top \mathbf{U}_{s_i} \mathbf{B})\right\|_F^2) \\
&= 2\mathbf{U}_{s_i}^\top \mathbf{G}(\mathbf{G}^\top \mathbf{U}_{s_i} \mathbf{B} - \mathbf{F}\mathbf{U}_{s_0})
\end{aligned} \tag{16}$$

References

[1] M. Rizkallah, T. Maugey, C. Yaacoub, and C. Guillemot, "Impact of light field compression on focus stack and extended focus images," in *24th European Signal Processing Conference (EUSIPCO)*. IEEE, 2016, pp. 898–902.

[2] I. J. S. JPEG, "Jpeg pleno light field coding vm 1.1," Doc. N81052, 2018.

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

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

[5] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, M. Adams, A.and Horowitz, and M. Levoy, "High performance imaging using large camera arrays," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 765–776, Jul. 2005.

[6] R. Ng, "Light field photography," Ph.D. dissertation, Stanford University, 2006.

[7] T. Georgiev and A. Lumsdaine, "Focused plenoptic camera and rendering," *J. of Electronic Imaging*, vol. 19, no. 2, Apr. 2010.

[8] C. Conti, L. D. Soares, and P. Nunes, "Hevc-based 3d holoscopic video coding using self-similarity compensated prediction," *Signal Processing: Image Communication*, vol. 42, pp. 59–78, 2016.

[9] Y. Li, R. Olsson, and M. Sjöström, "Compression of unfocused plenoptic images using a displacement intra prediction," in *2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, 2016, pp. 1–4.

[10] R. Verhack, T. Sikora, L. Lange, R. Jongebloed, G. Van Wallendael, and P. Lambert, "Steered mixture-of-experts for light field coding, depth estimation, and processing," in *2017 IEEE International Conference on Multimedia and Expo (ICME),*. IEEE, 2017, pp. 1183–1188.

[11] I. Tabus, P. Helin, and P. Astola, "Lossy compression of lenslet images from plenoptic cameras combining sparse predictive coding and jpeg 2000," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 4567–4571.

[12] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Processing Magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[13] M. Hog, N. Sabater, and C. Guillemot, "Super-rays for efficient light field processing," *IEEE J. on Selected Topics in Signal Processing, special issue on light field image processing*, Oct. 2017.

[14] X. Su, M. Rizkallah, T. Maugey, and C. Guillemot, "Rate-distortion optimized super-ray merging for light field compression," in *European Signal Processing Conference (EUSIPCO)*, 2018.

[15] M. Rizkallah, X. Su, T. Maugey, and C. Guillemot, “Graph-based transforms for predictive light field compression based on super-pixels,” in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

[16] G. Shen, W.-S. Kim, S. K. Narang, A. Ortega, J. Lee, and H. Wey, “Edge-adaptive transforms for efficient depth map coding,” in *Picture Coding Symposium (PCS)*. IEEE, 2010, pp. 566–569.

[17] W.-S. Kim, S. K. Narang, and A. Ortega, “Graph based transforms for depth video coding,” in *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2012, pp. 813–816.

[18] W. Hu, G. Cheung, A. Ortega, and O. C. Au, “Multiresolution graph fourier transform for compression of piecewise smooth images,” *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 419–433, 2015.

[19] W.-T. Su, G. Cheung, and C.-W. Lin, “Graph fourier transform with negative edges for depth image coding,” in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 1682–1686.

[20] E. Pavez, H. E. Egilmez, Y. Wang, and A. Ortega, “Gtt: Graph template transforms with applications to image coding,” in *Picture Coding Symposium (PCS)*. IEEE, 2015, pp. 199–203.

[21] I. Rotondo, G. Cheung, A. Ortega, and H. Egilmez, “Designing sparse graphs via structure tensor for block transform coding of images,” *APSIPA ACS, Hong Kong, China*, 2015.

[22] G. Fracastoro, D. Thanou, and P. Frossard, “Graph transform learning for image compression,” in *Picture Coding Symposium (PCS)*. IEEE, 2016, pp. 1–5.

[23] W. Hu, G. Cheung, and A. Ortega, “Intra-prediction and generalized graph fourier transform for image coding,” *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 1913–1917, 2015.

[24] H. E. Egilmez, A. Said, Y.-H. Chao, and A. Ortega, “Graph-based transforms for inter predicted video coding,” in *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2015, pp. 3992–3996.

[25] K.-S. Lu and A. Ortega, “Symmetric line graph transforms for inter predictive video coding,” in *Picture Coding Symposium (PCS)*. IEEE, 2016, pp. 1–5.

[26] C. Conti, P. Nunes, and L. D. Soares, “New hevc prediction modes for 3d holoscopic video coding,” in *2012 19th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2012, pp. 1325–1328.

[27] ——, “Hevc-based light field image coding with bi-predicted self-similarity compensation,” in *2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*. IEEE, 2016, pp. 1–4.

[28] P. David, M. Le Pendu, and C. Guillemot, “White lenslet image guided demosaicing for plenoptic cameras,” in *2017 IEEE 19th International Workshop on Multimedia Signal Processing (MMSP)*. IEEE, 2017, pp. 1–6.

[29] C. Jia, Y. Yang, X. Zhangy, X. Zhang, S. Wangx, S. Wang, and S. Ma, “Optimized inter-view prediction based light field image compression with adaptive reconstruction,” in *2017 IEEE International Conference on Image Processing ICIP*, 2017.

[30] W. Ahmad, R. Olsson, and M. Sjostrom, “Interpreting plenoptic images as multiview sequences for improved compression,” in *2017 IEEE International Conference on Image Processing ICIP*, 2017.

[31] R. Verhack, T. Sikora, L. Lange, R. Jongebloed, G. Van Wallendael, and P. Lambert, “Steered mixture-of-experts for light field coding, depth estimation, and processing,” in *2017 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2017, pp. 1183–1188.

[32] S. Zhao and Z. Chen, “Light field image coding via linear approximation prior,” in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 4562–4566.

[33] X. Jiang, M. Le Pendu, and C. Guillemot, “Light fields compression using depth image based view synthesis,” in *Hot3D workshop held jointly with IEEE Int. Conf. on Multimedia and Expo, ICME*. IEEE, July 2017.

[34] X. Su, M. Rizkallah, T. Maugey, and C. Guillemot, “Graph-based light fields representation and coding using geometry information,” in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017.

[35] N. K. Kalantari, T.-C. Wang, and R. Ramamoorthi, “Learning-based view synthesis for light field cameras,” *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 193, 2016.

[36] J. Berent and P. L. Dragotti, “Unsupervised extraction of coherent regions for image based rendering.” in *BMVC*, 2007, pp. 1–10.

[37] H. Mihara, T. Funatomi, K. Tanaka, H. Kubo, Y. Mukaigawa, and H. Nagahara, “4d light field segmentation with spatial and angular consistencies,” in *ICCP*, 2016, pp. 1–8.

[38] S. Wanner, C. Straehle, and B. Goldluecke, “Globally consistent multi-label assignment on the ray space of 4d light fields,” in *CVPR*, 2013, pp. 1011–1018.

[39] H. Zhu, Q. Zhang, and Q. Wang, “4D light field superpixel and segmentation,” in *IEEE International Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6384–6392.

[40] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, “Slic superpixels compared to state-of-the-art superpixel methods,” *IEEE transactions on pattern analysis and machine intelligence*, vol. 34, no. 11, pp. 2274–2282, 2012.

[41] M. Hog, N. Sabater, and C. Guillemot, “Superrays for efficient light field processing,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 7, pp. 1187–1199, 2017.

[42] A. Kovnatsky, M. M. Bronstein, A. M. Bronstein, K. Glashoff, and R. Kimmel, “Coupled quasi-harmonic bases,” in *Computer Graphics Forum*, vol. 32, no. 2pt4. Wiley Online Library, 2013, pp. 439–448.

[43] I. Daribo, G. Cheung, and D. Florencio, “Arithmetic edge coding for arbitrarily shaped sub-block motion prediction in depth video compression,” in *2012 19th IEEE International Conference on Image Processing*, Sept 2012, pp. 1541–1544.

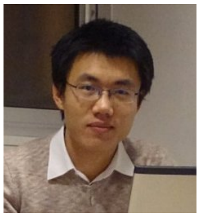
[44] H. Freeman, “On the encoding of arbitrary geometric configurations,” *IRE Transactions on Electronic Computers*, vol. EC-10, no. 2, pp. 260–268, June 1961.

[45] I. Viola, H. P. Maretic, P. Frossard, and T. Ebrahimi, “A graph learning approach for light field image compression,” in *Applications of Digital Image Processing XLI*, vol. 10752. International Society for Optics and Photonics, 2018, p. 107520E.

[46] X. Jiang, M. Le Pendu, and C. Guillemot, “Depth estimation with occlusion handling from a sparse set of light field views,” in *2018 IEEE International Conference on Image Processing (ICIP)*, 2018.

[47] K. B. Petersen, M. S. Pedersen *et al.*, “The matrix cookbook,” *Technical University of Denmark*, vol. 7, no. 15, p. 510, 2008.

Mira Rizkallah is a Post Doctoral researcher in INRIA Rennes. She received her Ph.D. degree from University of Rennes 1 - INRIA (Rennes, France) in 2019 with a full grant from The French Ministry of Education and Research. She received the M.Sc. and B.E degree in Telecommunications engineering with highest honors from the Holy Spirit University of Kaslik (USEK), Lebanon, in late 2015. She has also received an Excellence Scholarship for her first 3 years in USEK, then a merit-based scholarship for her master studies. During her last year of Master studies, she did her professional internship in collaboration with INRIA, Rennes under a full scholarship from AUF (Agence Universitaire de La Francophonie). Her current research interests span the areas of Light Fields and Omnidirectional image Processing, Image and Video coding and Graph Signal Processing.



Xin Su received the B.S. degree in electronic engineering from Wuhan University, Wuhan, China, in 2008, and the Ph.D. degree in image and signal processing from Telecom ParisTech, Paris, France, in 2015. His research interests include multiview coding, 3-D video communication, image rendering, image processing, and computer vision.

Thomas Maugey received the Engineering degree from Ecole Superieure d Electricite, Supelec, Gif-sur-Yvette, France, in 2007, the M.Sc. degree in fundamental and applied mathematics from Supelec and Universite Paul Verlaine Metz, Metz, France, in 2007, and the Ph.D. degree in image and signal processing from Telecom ParisTech, Paris, France, in 2010.

He was a Post-Doctoral Researcher with the Signal Processing Laboratory, Swiss Federal Institute of Technology, Lausanne, Switzerland, from 2010 to 2014. He is currently a Research Scientist with the Team SIROCCO, Institut National de Recherche en Informatique et en Automatique, Rennes, France. His research interests include monoview and multiview distributed videocoding, 3-D video communication, data representation, video compression, network coding, and view synthesis.

Christine Guillemot IEEE fellow, is Director of Research at INRIA, head of a research team dealing with image and video modeling, processing, coding and communication. She holds a Ph.D. degree from ENST (Ecole Nationale Superieure des Telecommunications) Paris, and an Habilitation for Research Direction from the University of Rennes. From 1985 to Oct. 1997, she has been with FRANCE TELECOM, where she has been involved in various projects in the area of image and video coding for TV, HDTV and multimedia. From Jan. 1990 to mid 1991, she has worked at Bellcore, NJ, USA, as a visiting scientist. Her research interests are signal and image processing, and in particular 2D and 3D image and video processing for various problems (compression, super-resolution, inpainting, classification).

She has served as Associate Editor for IEEE Trans. on Image Processing (from 2000 to 2003, and from 2014-2016), for IEEE Trans. on Circuits and Systems for Video Technology (from 2004 to 2006), and for IEEE Trans. on Signal Processing (2007-2009). She has served as senior member of the editorial board of the IEEE journal on selected topics in signal processing (2013-2015) and is currently senior area editor of IEEE Trans. on Image Processing.