SOFT DECODING OF LIGHT FIELD IMAGES USING POCS AND FAST GRAPH SPECTRAL FILTERS

Shuai Yang[1], Gene Cheung[2], Jiaying Liu[1], Zongming Guo[1]*

[1]Institute of Computer Science and Technology, Peking University, China
[2]National Institute of Informatics, Japan

ABSTRACT

Light field data captured by a lenslet-based image sensor is typically demosaicked, aligned and rearranged into a series of sub-aperture (viewpoint) images, before a disparity-compensated coding scheme is employed for compression. In this paper, we focus on the problem of soft decoding of block-based compressed sub-aperture images at the decoder: given quantization bin indices of DCT coefficients of non-overlapping code blocks, we select appropriate coefficient values that are low-pass filtered using graph spectral filters and view-consistent across sub-aperture images via projection on convex sets (POCS). Specifically, after an initial pixel estimate, we low-pass filter each pixel block using accelerated graph filters based on the Lanczos method. We then map filtered pixels to a neighborhood of sub-aperture images based on estimated disparity to enforce indexed quantization bin constraints of multiple images. Experimental results show that our algorithm achieves PSNR gain of 2.34dB over JPEG hard decoding.

Index Terms— light field imaging, graph signal processing

1. INTRODUCTION

A lenslet light field sensor places a microlens array behind a conventional camera lens, so that the image sensor can capture intensity of light from different directions per pixel. The captured color pixels then undergo demosaicking, alignment, and finally mapping to a 2D array of sub-aperture images, that are slight viewpoint shifts of each other [1]. Because of the large data volume of light field images compared to conventional RGB images, recently there are extensive research on the compression of sub-aperture images [2–8]. With some exceptions, traditional coding tools like block-based intra-prediction (intra-coding) or disparity compensation (inter-coding) plus transform coding are applied to exploit inherent spatial and inter-view redundancy in the sub-aperture images for coding gain. Transform coefficients are then scalar-quantized and entropy-coded.

In this paper, we address instead the decoder-side problem: given encoded quantization bin indices of a code block in a sub-aperture image, how to choose transform coefficients within the indexed bins to optimize reconstruction quality. A *hard decoding* approach would simply choose the center of each indexed quantization bin as the reconstructed coefficient, which typically leads to annoying coding artifacts in the decoded image. In contrast, a *soft decoding* approach would choose coefficients within indexed bins with the help of appropriate signal priors. For JPEG images, priors such as *total variation* (TV) [9], sparsity with respect to a learned dictionary [10] and graph-signal smoothness [11] have been proposed.

Unlike previous JPEG soft decoding works [9–12], we design a soft decoding algorithm specifically for sub-aperture images using fast graph spectral filters and *projection on convex sets* (POCS) [12] that consider in addition inter-view consistency. Specifically, we first use [11] to soft decode an initial set of sub-aperture images. We then design two graph spectral filters—implemented as acceleration filters based on Lanczos method [13]—that low-pass filter two different types of blocks depending on the block content. Finally, using disparity estimated in [14], we project filtered pixels of a target image onto neighboring sub-aperture images to enforce quantization bin constraints of multiple views. Experimental results show that our algorithm achieves PSNR gain of 2.34dB over JPEG hard decoding[1] and 1.40dB over [11].

2. RELATED WORK

Soft decoding of JPEG images—selecting DCT coefficients within indexed quantization bins in each code block at the decoder with signal priors—has been studied extensively [9–12]. Similarly, we also select coefficients in indexed quantization bins for sub-aperture images, but in addition consider inter-view consistency as an additional constraint. Further, our solution to the MAP optimization is implemented using accelerated graph filters [13], which are significantly faster than alternative straightforward filter implementations.

Most recent light field compression work focus on the encoding component [2–8], *i.e.,* how to exploit spatial and inter-view redundancy among sub-aperture images to maximally reduce encoding bitrate during lossy compression. Assuming a generic block-based image compression scheme with scalar quantization, to the best of our knowledge, how to best select transform coefficients within indexed quantization bins to decode compressed sub-aperture images has not been studied before. The most related work we are aware of is [15], which addresses the joint decoding problem of multiview depth images considering quantization bins of multiple images. Our work differs from [15] in that we in addition introduce fast graph spectral filtering to obtain a reasonable starting set of pixels.

3. SYSTEM OVERVIEW

3.1. Sub-aperture Array

We assume that the light field data have been organized into a $M \times N$ 2D array of sub-aperture images $\mathbf{I}_{i,j}$, $1 \leq i \leq M$, $1 \leq j \leq N$. Each sub-aperture image is of the same spatial resolution, and a pair of neighboring sub-aperture images are a slight viewpoint shift from each other. We first assume that a recent JPEG soft decoding algorithm [11] has been applied to each sub-aperture image individually, so that a set of reasonable quality sub-aperture images have

∗ Corresponding author
This work was supported by the National Natural Science Foundation of China under contract No. 61772043.

[1]Although the experiments are conducted using JPEG coded sub-aperture images, in general any block-based transform coding scheme with scalar quantization can be candidates for our optimization.

been decoded at the decoder. We next assume that a known disparity estimation algorithm [14] has been executed. Thus, for a target patch $\mathbf{x}$ of size $\mathbb{R}^{\sqrt{K}\times\sqrt{K}}$ in a sub-aperture image, disparity-compensated patches $\mathbf{y}_i$ also of size $\mathbb{R}^{\sqrt{K}\times\sqrt{K}}$ in a neighborhood of sub-aperture images have been identified. We denote the average of these disparity-compensated patches as $\mathbf{y}$.

If target patch $\mathbf{x}$ covers only smooth surface of the same object in a 3D scene, then disparity estimation is likely correct, and the computed average patch $\mathbf{y}$ is typically well matched with $\mathbf{x}$. On the other hand, if $\mathbf{x}$ straddles across a disparity boundary (*i.e.*, it covers pixels from both foreground objects and background), then disparity estimation among sub-aperture images is likely inaccurate, and $\mathbf{y}$ would blur foreground and background pixels from different $\mathbf{y}_i$. In this case, we model the average image patch formation using an image blurring model:

$$\mathbf{y} = \mathbf{B}\mathbf{x} + \mathbf{n}, \tag{1}$$

where $\mathbf{B}$ is a blur kernel and $\mathbf{n}$ is a zero-mean noise term. We discuss how to estimate $\mathbf{B}$ from data in Section 4.3. Based on the gradient of $\mathbf{y}$, we will employ different image formation models and apply different graph spectral filters (to be discussed later).

3.2. Quantization Constraints

We assume that each sub-aperture image $\mathbf{I}_{i,j}$ is block-transform-coded at the encoder using a transform matrix $\mathbf{T}$. Specifically, the l-th n-pixel block $\mathbf{b}_{i,j}(l) \in \mathbf{I}_{i,j}$ is DCT-transformed and scalar-quantized with quantization parameters (QP) $\mathbf{Q} = [q_1, \ldots, q_n]^T$:

$$\mathbf{q}_{i,j}(l) \odot \mathbf{Q} \leq \mathbf{T}\mathbf{b}_{i,j}(l) < (\mathbf{q}_{i,j}(l) + \mathbf{1}) \odot \mathbf{Q}, \tag{2}$$

where $\mathbf{q}_{i,j}(l)$ is the vector of quantization indices for $\mathbf{b}_{i,j}(l)$ and $\odot$ denotes element-by-element multiplication. In words, (2) states that reconstructed transform coefficients $\mathbf{T}\mathbf{b}_{i,j}(l)$ must fall inside the indexed quantization bins. At the decoder, given indices $\mathbf{q}_{i,j}(l)$ one must choose transform coefficients $\mathbf{T}\mathbf{b}_{i,j}(l)$ that satisfy quantization bin constraints in (2) with the help of additional signal priors.

4. ALGORITHM DEVELOPMENT

4.1. Graph Signal Processing (GSP) Definitions

We begin with a brief review of key GSP definitions. We interpret a K-pixel patch $\mathbf{x} \in \mathbb{R}^K$ as a *graph-signal* on a sparse graph $\mathcal{G}(\mathcal{N}, \mathcal{E})$ with $|\mathcal{N}| = K$ nodes and sparse edges $\mathcal{E}$, $|\mathcal{E}| \ll K^2$; each pixel is represented by one node. We assume a 4-connected graph, meaning each pixel is connected to its four horizontal and vertical neighbors. An *adjacency matrix* $\mathbf{W}$ contains entries $w_{i,j}$ that are edge weights connecting nodes i and j. Conventionally, edge weight $w_{i,j}$ is a function of inter-pixel difference computed using a Gaussian kernel [16]. In this paper, we argue that compression artifacts make computation of single pixel-pair difference unreliable. Thus we propose a robust neighborhood-based measurement as follows:

$$w_{i,j} = \max(0, 1 - (g_i + g_j)/2), \tag{3}$$

$$g_i = \sum_{k\in R(i)} G_{i,k}|\nabla x_k| \;/\; \sum_{k\in R(i)} G_{i,k}, \tag{4}$$

where $R(i)$ is a small neighborhood around pixel i, $|\nabla x_k|$ is the gradient magnitude at x_k, $G_{i,k} = \exp\left(-|i-k|^2/\sigma^2\right)$ is the Gaussian distance weight and g_i is the sum of weighted $|\nabla x_k|$ to estimate local variation. A diagonal *degree matrix* $\mathbf{D}$ contains entries $d_{i,i} = \sum_j w_{i,j}$. An *unnormalized Graph Laplacian matrix* $\mathbf{L}$ is simply defined as $\mathbf{D} - \mathbf{W}$. Given that the edge weights $w_{i,j}$ computed using (4) are non-negative, $\mathbf{L}$ is a positive semi-definite (PSD) matrix.

4.2. MAP Formulation with Graph Smoothness Prior

We first formulate a *maximum a posteriori* (MAP) problem using the image model in (1) and *blur kernel* $\mathbf{B}$, along with graph-signal smoothness prior $\mathbf{x}^T\mathbf{L}\mathbf{x}$ [17]:

$$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{B}\mathbf{x}\|_2^2 + \mu\, \mathbf{x}^T\mathbf{L}\mathbf{x}, \tag{5}$$

where μ is a parameter that trades off the fidelity term and the smoothness prior. $\mathbf{x}^T\mathbf{L}\mathbf{x}$ can be rewritten as [16]:

$$\mathbf{x}^T\mathbf{L}\mathbf{x} = \sum_{i,j} w_{i,j}(x_i - x_j)^2 = \sum_k \lambda_k \alpha_k^2, \tag{6}$$

where λ_k are the eigenvalues (graph frequencies) of $\mathbf{L}$ and α_k are the signal $\mathbf{x}$'s corresponding frequency coefficients. A small smoothness prior $\mathbf{x}^T\mathbf{L}\mathbf{x}$ thus means that nodes i and j connected by large weights $w_{i,j}$ have similar values x_i and x_j, or the signal $\mathbf{x}$ has energy α_k^2 resides mostly in the low frequencies λ_k's.

(5) has a closed-form solution (assuming $\mathbf{B}^T\mathbf{B} + \mu\mathbf{L}$ is positive definite (PD)):

$$\mathbf{x}^* = \left(\mathbf{B}^T\mathbf{B} + \mu\mathbf{L}\right)^{-1}\mathbf{B}\mathbf{y}. \tag{7}$$

We argue $\mathbf{B}^T\mathbf{B} + \mu\mathbf{L}$ is PD in general as follows. By definition constant vector $\mathbf{1}$ is a unique eigenvector corresponding to eigenvalue 0 for $\mathbf{L}$. On the other hand, blur kernel $\mathbf{B}$ should not change a constant signal, *i.e.*, $\mathbf{B}\mathbf{1} = \mathbf{1}$. Since both $\mathbf{B}^T\mathbf{B}$ and $\mathbf{L}$ are positive semi-definite (PSD), and $\nexists \mathbf{v} \neq \mathbf{0}$ such that $\mathbf{v}^T\mathbf{B}^T\mathbf{B}\mathbf{v} = 0$ and $\mathbf{v}^T\mathbf{L}\mathbf{v} = 0$ at the same time, $\mathbf{B}^T\mathbf{B} + \mu\mathbf{L}$ must be PD for $\mu > 0$.

Let $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, where $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$. Suppose $\mathbf{B} = \mathbf{U}g(\mathbf{\Lambda})\mathbf{U}^T$, where $g(\mathbf{\Lambda}) = \mathrm{diag}\left(g(\lambda_1), \ldots, g(\lambda_N)\right)$, is a polynomial of some finite order H. Then,

$$\begin{aligned}\mathbf{B}^T\mathbf{B} + \mu\mathbf{L} &= \mathbf{U}g^2(\mathbf{\Lambda})\mathbf{U}^T + \mu\mathbf{L}, \\ &= \mathbf{U}\left(g^2(\mathbf{\Lambda}) + \mu\mathbf{\Lambda}\right)\mathbf{U}^T. \end{aligned} \tag{8}$$

$$\mathbf{x}^* = \underbrace{\left(\frac{g(\mathbf{L})}{g^2(\mathbf{L}) + \mu\mathbf{L}}\right)}_{\mathbf{\Theta}_1}\mathbf{y}. \tag{9}$$

4.3. Optimal $g(\mathbf{L})$ Estimation

To estimate blur kernel $\mathbf{B}$, for simplicity we first assume $\mathbf{B}$ is a linear function of $\mathbf{L}$, *i.e.*, $\mathbf{B} = a_0\mathbf{I} + a_1\mathbf{L}$. Then given a non-smooth patch $\mathbf{y}_i$ (averaged gradient magnitude $|\nabla\mathbf{y}|$ is greater than a predefined parameter τ) and its corresponding ground truth patch $\mathbf{x}_i$ from the uncompressed image, we compute the optimal polynomial coefficients as follows:

$$\min_{a_0,a_1} \|(a_0\mathbf{I} + a_1\mathbf{L})\mathbf{x}_i - \mathbf{y}_i\|_F^2. \tag{10}$$

Taking the derivative with respect to each of the coefficients and setting them to 0, we get the following linear system of equations:

$$\begin{bmatrix} \mathbf{x}_i^T\mathbf{x}_i & \mathbf{x}_i^T\mathbf{L}\mathbf{x}_i \\ \mathbf{x}_i^T\mathbf{L}\mathbf{x}_i & \mathbf{x}_i^T\mathbf{L}^2\mathbf{x}_i \end{bmatrix}\begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} \mathbf{y}_i^T\mathbf{x}_i \\ \mathbf{y}_i^T\mathbf{L}\mathbf{x}_i \end{bmatrix}. \tag{11}$$

Thus the solution for a pair $(\mathbf{x}_i, \mathbf{y}_i)$ is a tuple (a_0, a_1) in $\mathbb{R}^2$. If we plot these sets of solutions in 2D space as in Fig. 1, we observe that the distribution of the values of (a_0, a_1) is highly concentrated at the same particular value for two different images. In fact, the variance of a_0 and a_1 is less than 10^{-3} and 10^{-2}, respectively, meaning that

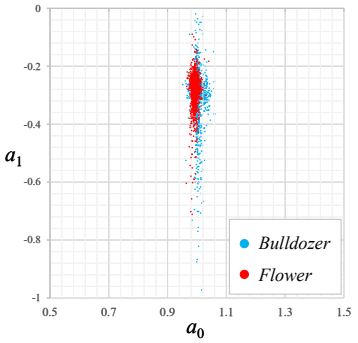


Fig. 1. Distribution of the optimal polynomial coefficients (a_0, a_1) over 1678 patches for image *Flower* (Fig. 2) and 1416 patches for *Bulldozer* (Fig. 3) at QF $= 40$.

we can find one linear polynomial that well models the blur process of most patches. In this paper, we fix $a_0 = 1$ and empirically set a_1 to $-0.39, -0.34, -0.28$ and -0.22 for images coded by a JPEG coder with quality factors (QF) $5, 10, 20$ and 40, respectively. One can show that for this approximated blur kernel $\mathbf{B}$, $\mathbf{B}^T\mathbf{B} + \mu\mathbf{L}$ in (7) is still PD and thus invertible, since $\mathbf{B1} = (\mathbf{I} + a_1\mathbf{L})\mathbf{1} \neq \mathbf{0}$.

4.4. Total Generalized Variation for Smooth Patches

For generally smooth patches (where $|\nabla \mathbf{y}| \leq \tau$) that suffer little blur across sub-aperture images, instead of the blur model (1) we adopt a simplier additive noise model $\mathbf{y} = \mathbf{x} + \mathbf{n}$. Further, we employ a graph variant of *total generalized variation* (TGV) [18] to avoid well-known staircase artifacts of TV, rather than the graph filter in (9). We first define a new graph Laplacian $\mathcal{L}$ for a 4-connected graph with all edge weights equal to 1. We then remove rows in $\mathcal{L}$ that correspond to t boundary pixels in the patch, resulting in a $(N-t) \times N$ sub-matrix $\mathcal{L}_s$. Using $\mathcal{L}_s$ to construct a higher-order smoothness prior analogous to TGV, we get:

$$\min_{\mathbf{x}} \|\mathbf{y} - \mathbf{x}\|_2^2 + \gamma \mathbf{x}^T \mathcal{L}_s^T \mathcal{L}_s \mathbf{x}. \tag{12}$$

Taking the derivative and solving for $\mathbf{x}$, we get:

$$\mathbf{x}^* = (\mathbf{I} + \gamma \mathcal{L}_s^T \mathcal{L}_s)^{-1} \mathbf{y}. \tag{13}$$

Clearly $\mathbf{I} + \gamma \mathcal{L}_s^T \mathcal{L}_s$ is PD. We can now define $\mathcal{L}^* = \mathcal{L}_s^T \mathcal{L}_s$ as a new graph Laplacian, and since it is symmetric and real, it also can be eigen-decomposed into $\mathbf{V}\mathbf{\Sigma}\mathbf{V}^T$, where $\mathbf{\Sigma}$ is a diagonal matrix with eigenvalues $\sigma_1, \ldots, \sigma_N$. We can thus write the filter (13) as $\mathbf{x}^* = \underbrace{(I + \gamma \mathcal{L}^*)^{-1}}_{\mathbf{\Theta}_2} \mathbf{y}$.

4.5. Accelerated Graph Filter using Lanczos Method

We follow [13] to construct accelerated graph filter implementations of (9) and (13). Denote by $V_Z = [v_1, \ldots, v_Z]$ the Z orthonormal vectors (we set $Z = 5$) of the Krylov subspace $K_M(\mathbf{L}, \mathbf{y}) = \text{span}\{\mathbf{y}, \mathbf{L}\mathbf{y}, \ldots, \mathbf{L}^{Z-1}\mathbf{y}\}$ computed using the Lanczos method. The computation cost is $O(Z|\mathcal{E}|)$. H_Z is a symmetric tridiagonal matrix relating $\mathbf{L}$ and V_Z:

$$V_Z^* \mathbf{L} V_Z = H_Z = \begin{bmatrix} \alpha_1 & \beta_2 & & & \\ \beta_2 & \alpha_2 & \beta_3 & & \\ & \beta_3 & \alpha_3 & \ddots & \\ & & \ddots & \ddots & \beta_Z \\ & & & \beta_Z & \alpha_Z \end{bmatrix}. \tag{14}$$

It is shown that a graph filter $f(\mathbf{L})\mathbf{y}$ that is a function of graph Laplacian $\mathbf{L}$ can be approximated as:

$$f(\mathbf{L})\mathbf{y} \approx \|\mathbf{y}\|_2 V_Z f(H_Z) e_1 := g_Z, \tag{15}$$

where e_1 is the first unit vector. $f(H_Z)$ is a length-Z vector, where λ_z^h is the z-th eigenvalue of H_Z. By the eigenvalue interleaving properties, $\lambda_1 \leq \lambda_z^h \leq \lambda_N$, so $f(H_Z)$ can be properly evaluated given $f(\mathbf{L})$.

4.6. Quantization Constraints across Multiple Views

After graph spectral filtering, we project pixels from the target sub-aperture image to neighboring images to enforce quantization bin constraints in multiple images. Let $N_8(u, v)$ denote the eight neighboring sub-aperture images of $\mathbf{I}_{u,v}$. To enforce the quantization bin constraints of $\mathbf{I}_{i,j}$, $(i, j) \in N_8(u, v)$, on pixels in $\mathbf{I}_{u,v}$, we map pixels from $\mathbf{I}_{u,v}$ to $\mathbf{I}_{i,j}$ via estimated disparities. Then for each block in $\mathbf{I}_{i,j}$ we simply clip its coefficients by the corresponding quantization bin boundaries—this is equivalent to POCS:

$$\min\Big((\mathbf{q}_{i,j}(l) + \mathbf{1}) \odot \mathbf{Q}, \max\Big(\mathbf{q}_{i,j}(l) \odot \mathbf{Q}, \mathbf{T}\mathbf{b}_{i,j}(l)\Big)\Big), \tag{16}$$

to satisfy (2). Finally, we transform the clipped coefficients back to pixel domain and map each pixel back to $\mathbf{I}_{u,v}$.

4.7. Summary of Soft Decoding Algorithm

To summarize, we first soft decode sub-aperture images using [11], align and take average over them based on the estimated disparities [14] to obtain an initial $\mathbf{I}_{u,v}$. Then we classify patches $\mathbf{y}$ in $\mathbf{I}_{u,v}$ by comparing their averaged gradient magnitude $|\nabla \mathbf{y}|$ with a predefined parameter τ, and select graph filters based on the classification results to low-pass filter $\mathbf{y}$. Finally we project pixels of $\mathbf{I}_{u,v}$ onto neighboring sub-aperture images to enforce quantization bin constraints of multiple views. The fast graph filter procedure and POCS procedure iterate alternately until a predefined maximum iteration number $\mathcal{K}$ is reached. We include the pseudocode of the light field soft decoding in Algorithm 1.

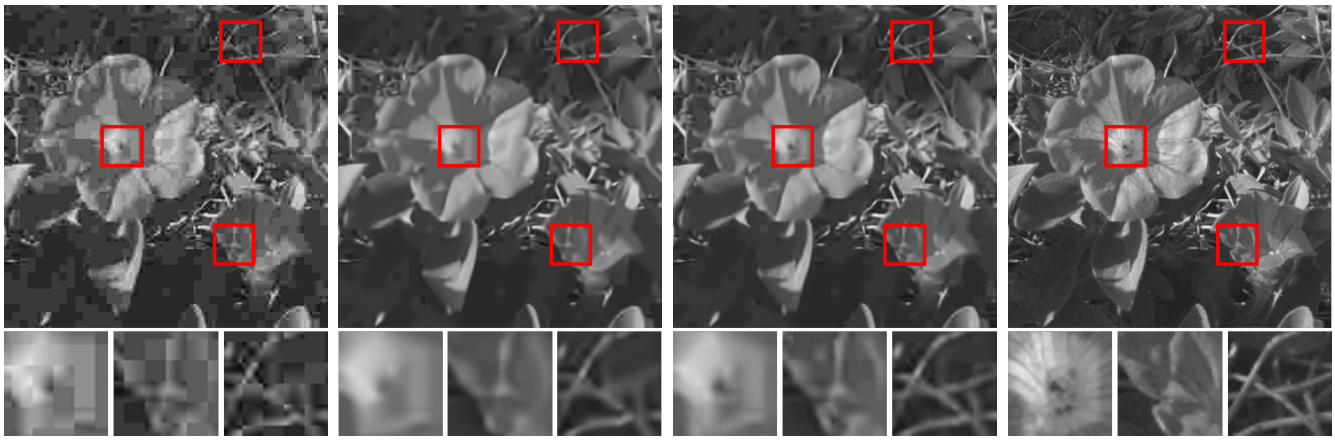
Algorithm 1 Light Field Soft Decoding

Input: Initial $\mathbf{I}_{u,v}$, parameters $\tau, \mathcal{K}$, disparity D
Output: Soft decoded image $\mathbf{I}_{u,v}^*$

Initialize $k = 1, \mathbf{I}^{(k)} = \mathbf{I}_{u,v}$
for $k = 1 \rightarrow \mathcal{K}$ **do**
  $\triangle$ *Fast graph spectral filters*:
  for all $\mathbf{y} \in \mathbf{I}^{(k)}$ **do**
    if $|\nabla \mathbf{y}|$ is greater than τ **then**
      Update $\mathbf{y}$ as $\mathbf{x}^* = \mathbf{F}_1(\mathbf{y})$ (Eq. (9)(15))
    else
      Update $\mathbf{y}$ as $\mathbf{x}^* = \mathbf{F}_2(\mathbf{y})$ (Eq. (13)(15))
  $\triangle$ *Multi-view quantization bin constraints*:
  for all $(i, j) \in N_8(u, v) \cup \{(u, v)\}$ **do**
    Map pixels from $\mathbf{I}_{u,v}$ to $\mathbf{I}_{i,j}$ via D
    for all $\mathbf{b}_{i,j}(l) \in \mathbf{I}_{i,j}$ **do**
      Clip coefficients via (16) and update $\mathbf{b}_{i,j}(l)$
    Map updated pixels back to $\mathbf{I}_{u,v}$ via D
  $\mathbf{I}^{(k+1)} = \mathbf{I}^{(k)}, k = k + 1$
$\mathbf{I}_{u,v}^* = \mathbf{I}^{(\mathcal{K})}$

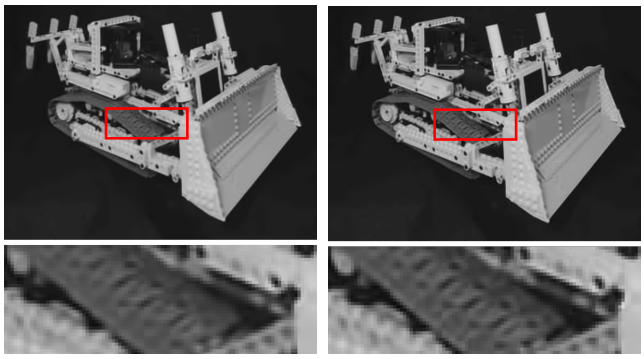
Table 1. Quality comparison with respective to PSNR (in dB) at different QF

Methods	*Flower*				*Bracelet*				*Bulldozer*				Average
	5	10	20	40	5	10	20	40	5	10	20	40	
JPEG	26.31	29.15	31.65	33.89	25.02	27.92	30.28	32.71	26.24	28.91	31.34	33.67	29.76
LERaG [11]	27.48	29.94	32.37	34.48	25.79	28.58	31.03	33.55	27.55	30.12	32.46	34.99	30.70
Ours	**29.35**	**31.95**	**34.11**	**36.00**	**26.76**	**29.78**	**32.28**	**34.88**	**28.62**	**31.40**	**33.81**	**36.27**	**32.10**



(a) JPEG (26.31dB) (b) LERaG [11] (27.48dB) (c) Ours (29.35dB) (d) Ground truth

Fig. 2. Comparison of tested methods in visual quality on *Flower* at QF $= 5$. For visual inspection, regions highlighted by red rectangles are enlarged and shown below the results. The corresponding PSNR values are also given as references.



(a) LERaG [11] (32.46dB) (b) Ours (33.81dB)

Fig. 3. Comparison of tested methods in visual quality on *Bulldozer* at QF $= 20$. For visual inspection, regions highlighted by red rectangles are enlarged and shown below the results. The corresponding PSNR values are also given as references.

5. EXPERIMENTS

We now present experimental results to demonstrate the superior performance of our soft decoding algorithm. In the experiments, we use 10×10 patches with an overlap of two pixel between adjacent patches. Weights for smoothness prior are set as $\mu = \gamma = 0.05$. We run for $\mathcal{K} = 1$ iteration and empirically set parameter τ to $0.03, 0.02, 0.005$ and 0.005 at QF $= 5, 10, 20$ and 40, respectively. To the best of our knowledge, there does not exist such a system that focuses on soft decoding of sub-aperture images. For comparison, we thus use LERaG [11], a JPEG soft decoding algorithm using an improved graph Laplacian based smoothness prior.

For quantitative comparison, soft decoding results in terms of PSNR are reported in Table 1. We observe that our method consistently performs better than other methods. The average PSNR gain over LERaG [11] is 1.40dB.

For visual comparison, soft decoding results for *Flower* at QF $= 5$ and for *Bulldozer* at QF $= 20$ are shown in Fig. 2 and Fig. 3, respectively. At very low QF $= 5$, the quantization noise is severe in the compressed JPEG image. Without reliable pixel information, LERaG [11] produces over-smoothed texture and edge regions. By carefully integrating the pixel information as well as the quantization information from nearby sub-aperture images, our method can restore image details like the dark stems and flower branches. At medium QF $= 20$, our result enjoys richer details as shown in Fig. 3.

6. CONCLUSION

The large volume of light field imaging data has attracted a lot of research in efficient compression of 2D arrays of sub-aperture images. In this paper, we focus instead on the soft decoding problem at the decoder-side: given indexed quantization bins of code blocks in a compressed sub-aperture image, how to best select transform coefficients within the indexed bins with appropriate signal priors. After an initial soft decoding of individual sub-aperture images using a recent work [11], we first estimate disparity among images using [14] to find matching patches across images to compute an average $\mathbf{y}$ in a target image. We then classify patch $\mathbf{y}$ into two classes depending on patch gradient, and apply one of two designed graph spectral low-pass filters, implemented as accelerated filters using the Lanczos method [13]. Finally, we project the low-pass filtered pixels to neighboring sub-aperture images for projection on convex sets (POCS), so that the quantization bin constraints of multiple images are satisfied. Experimental results show that our method outperforms JPEG hard decoding and a state-of-the-art JPEG soft decoding method [11] noticeably in both PSNR and subjective quality.

7. REFERENCES

[1] D. Dansereau, O. Pizarro, and S. Williams, "Decoding, calibration and rectification for lenselet-based plenoptic cameras," in *IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, June 2013.

[2] C. Conti, P. Nunes, and L. D. Soares, "HEVC-based light field image coding with bi-predicted self-similarity compensation," in *IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, Seattle, WA, July 2016.

[3] R. Monteiro et al., "Light field HEVC-based image coding using locally linear embedding and self-similarity compensated prediction," in *IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, Seattle, WA, July 2016.

[4] D. Liu et al., "Pseudo-sequence-based light field image compression," in *IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, Seattle, WA, July 2016.

[5] C. Perra and P. Assuncao, "High efficiency coding of light field images based on tiling and pseudo-temporal data arrangement," in *IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, Seattle, WA, July 2016.

[6] Y.-H. Chao, G. Cheung, and A. Ortega, "Pre-demosiac light field compression using graph lifting transform," in *IEEE International Conference on Image Processing*, Beijing, China, September 2017.

[7] T.-H. Tran, Y. Baroud, Z. Wang, S. Simon, and D. Taubman, "Light-field image compression based on variational disparity estimation and motion-compensated wavelet decomposition," in *IEEE International Conference on Image Processing*, Beijing, China, September 2017.

[8] J. Chen, J. Hou, and L. P. Chau, "Light field compression with disparity guided sparse coding based on structural key views," *IEEE Transactions on Image Processing*, vol. PP, no. 99, pp. 1–1, 2017.

[9] K. Bredies and M. Holler, "A total variation-based jpeg decompression model," in *SIAM J. Img. Sci.*, January 2012, vol. 5, no.1, pp. 366–393.

[10] H. Chang, M. Ng, and T. Zeng, "Reducing artifacts in JPEG decompression via a learned dictionary," in *IEEE Transactions on Signal Processing*, February 2014, vol. 62, no. 3, pp. 718–728.

[11] X. Liu, G. Cheung, X. Wu, and D. Zhao, "Random walk graph laplacian based smoothness prior for soft decoding of JPEG images," in *IEEE Transactions on Image Processing*, February 2017, vol. 26, no.2, pp. 509–524.

[12] R. Rosenholtz and A. Zakhor, "Iterative procedures for reduction of blocking effects in transform image coding," in *IEEE Transactions on Circuits and Systems for Video Technology*, March 1992, vol. 2, no.1, pp. 91–95.

[13] A. Susnjara1, N. Perraudin, D. Kressner1, and P. Vandergheynst, "Accelerated filtering on graphs using Lanczos method," in *unpublished, arXiv:1509.04537*, 2015.

[14] Hae Gon Jeon, Jaesik Park, Gyeongmin Choe, Jinsun Park, Yunsu Bok, Yu Wing Tai, and In So Kweon, "Accurate depth map estimation from a lenslet light field camera," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1547–1555.

[15] P. Wan, G. Cheung, P. Chou, D. Florencio, C. Zhang, and O. Au, "Precision enhancement of 3D surfaces from compressed multiview depth maps," in *IEEE Signal Processing Letters*, October 2015, vol. 22, no.10, pp. 1676–1680.

[16] 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," in *IEEE Signal Processing Magazine*, May 2013, vol. 30, no.3, pp. 83–98.

[17] J. Pang and G. Cheung, "Graph Laplacian regularization for inverse imaging: Analysis in the continuous domain," in *IEEE Transactions on Image Processing*, April 2017, vol. 26, no.4, pp. 1770–1785.

[18] K. Bredies and M. Holler, "A TGV-based framework for variational image decompression, zooming and reconstruction. Part I: Analytics," in *SIAM Jour*, 2015, vol. 8, no.4, pp. 2814–2850.