Graph-Based Light Field Super-Resolution

Mattia Rossi and Pascal Frossard
École Polytechnique Fédérale de Lausanne
mattia.rossi@epfl.ch, pascal.frossard@epfl.ch

***Abstract*—Light field cameras can capture the 3D information in a scene with a single exposure. This special feature makes light field cameras very appealing for a variety of applications: from post capture refocus, to depth estimation and image-based rendering. However, light field cameras exhibit a very limited spatial resolution, which should therefore be increased by computational methods. Off-the-shelf single-frame and multi-frame super-resolution algorithms are not ideal for light field data, as they ignore its particular structure. A few super-resolution algorithms explicitly devised for light field data exist, but they exhibit significant limitations, such as the need to carry out an explicit disparity estimation step for one or several light field views. In this work we present a new light field super-resolution algorithm meant to address these limitations. We adopt a multi-frame alike super-resolution approach, where the information in the different light field views is used to augment the spatial resolution of the whole light field. In particular, we show that coupling the multi-frame paradigma with a graph regularizer that enforces the light field structure permits to avoid the costly and challenging disparity estimation step. Our experiments show that the proposed method compares favorably to the state-of-the-art for light field super-resolution algorithms, both in terms of PSNR and visual quality.**

I. Introduction

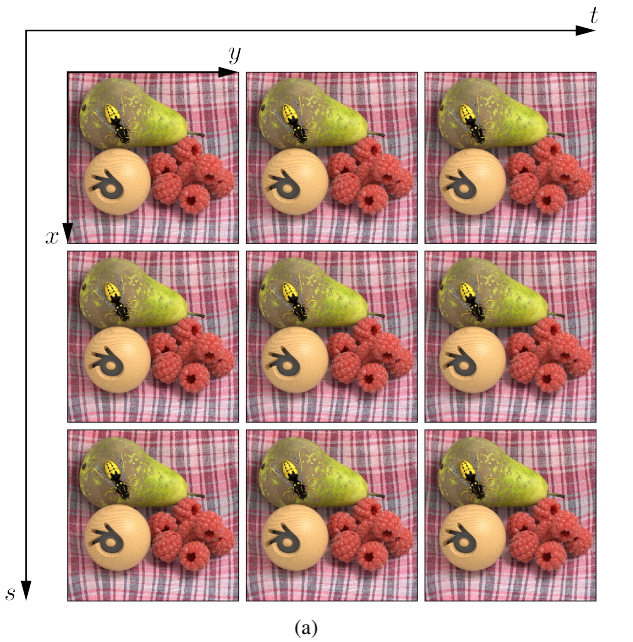
A light field camera behaves as a compact camera array, providing multiple simultaneous images of a 3D scene from slightly different points of view [1]. This very rich information, referred to as the *light field* [2], can potentially be used in a variety of applications, from post-capture refocus to depth estimation or virtual reality. However, the light field views exhibit a significantly lower resolution than images from traditional cameras, and many light field applications, such as depth estimation, happen to be very challenging on low spatial resolution data. The design of spatial super-resolution techniques, aiming at increasing the view resolution, is therefore crucial in order to fully exploit the potential of light field cameras.

Single-frame super-resolution algorithms [3] [4] can be applied to each light field view separately in order to augment the resolution of the whole light field. This is the approach of the method in [5], where a *Convolutional Neural Network* designed for single-frame super-resolution is applied to each light view separately. Ideally though, light field super-resolution should be able to exploit jointly the information in the multiple views. From this perspective, the multi-frame super-resolution scenario resembles more closely the light field one. However, the global image warping model that is typically adopted in multi-frame super-resolution to capture multi-image correlation, such as in [6] and [7], cannot capture the particular light field structure.

In [8] Wanner and Goldluecke propose a super-resolution algorithm targeting light field data. They compute a disparity map at each view of the light field and employ them to project all the views to the target one, within a global optimization formulation endowed with a *Total Variation* prior. However, disparity estimation is a very challenging task at low spatial resolution. As a result, disparity errors translate into significant artifacts in the textured areas and along object edges of the super-resolved light field views. Moreover, the algorithm has to be applied separately at each view in order to super-resolve the whole light field, which does not permit to fully exploit the inter-view dependencies.

In a different framework, Mitra and Veeraraghavan propose a light field super-resolution algorithm based on a learning procedure [9]. Each view in the low resolution light field is divided into possibly overlapping patches. All the patches at the same spatial coordinates in the different views form a light field with very small spatial resolution, i.e., a light field patch. The authors assign a constant disparity to each light field patch, i.e., all the objects within the light field patch are assumed to lie at the same depth in the scene. A different *Gaussian Mixture Model* prior for high resolution light field patches is learnt offline for each discrete disparity value, and then employed online within a *MAP* estimator to super-resolve each light field patch separately. However, first the learning-based approach ties the reconstruction quality to the chosen training set and requires a proper discretization of the disparity range; second, the simple assumption of constant disparity within each light field patch leads to severe artifacts at depth discontinuities in the super-resolved light field views.

In this work, we propose a new light field super-resolution algorithm that provides a global solution that augments the resolution of all the views together, without an explicit a priori disparity estimation step, and without relying on an offline learning procedure. In particular, we propose to cast *light field spatial super-resolution* into a global optimization problem, whose objective function comprises three terms. The first one enforces data fidelity, by constraining each high resolution view to be consistent with its low resolution counterpart. The second one is a warping term, which gathers for each view the complementary information encoded in the other ones. The third one is a graph-based prior, which regularizes the high resolution views by enforcing the geometric light field structure. These terms altogether form a quadratic objective function that we solve iteratively. The results show that our algorithm compares favorably to state-of-the-art light field super-resolution algorithms, both visually and in terms of



(a)

(b)

Fig. 1. Example of light field and EPI. Figure (a) shows an array of 3×3 views, extracted from the `stillLife` light field [10], which actually consists of an array of 9×9 views. Figure (b) shows an EPI from the same light field.

reconstruction error.

The article is organized as follows. Section II formalizes the light field structure. Section III presents our problem formulation. Section IV describes our super-resolution algorithm. Section V is dedicated to our experiments, and Section VI concludes the article.

II. Light field structure

In the following we consider the light field as the output of an $M \times M$ array of pinhole cameras, each one equipped with an $N \times N$ pixel sensor. A light field example is provided in Figure 1a. Each view is identified through the angular coordinate (s,t) with $s,t \in \{1,2,...,M\}$, while a pixel within the view is identified through the spatial coordinate (x,y) with $x,y \in \{1,2,...,N\}$. Within this setup, we can represent the light field as an $N \times N \times M \times M$ real matrix $\boldsymbol{U}$, with $\boldsymbol{U}(x,y,s,t)$ the intensity of a pixel with coordinates (x,y) in the view (s,t). In particular, we denote the view (s,t) as $\boldsymbol{U}_{s,t} \equiv \boldsymbol{U}(\cdot,\cdot,s,t) \in \mathbb{R}^{N \times N}$. Finally, without lack of generality, we assume that each pair of horizontally or vertically adjacent views in the light field are properly registered.

With reference to Figure 2, we now describe in more details the particular structure of the light field data. We consider a point P at depth z from the camera array, whose projection on one of the views is represented by the pixel $\boldsymbol{U}_{s,t}(x,y)$. We now look at the projection of P on the other views $\boldsymbol{U}_{s',t'}$. For the sake of simplicity, Figure 2 represents only the 8 views around $\boldsymbol{U}_{s,t}$. We observe that, in the absence of occlusions and under the Lambertian assumption (i.e., same color intensity regardless of the viewing angle), the projection of P obeys the following multi-stereo equation:

$$\begin{aligned} \boldsymbol{U}_{s,t}(x,y) &= \boldsymbol{U}_{s',t'}(x+(s-s')\,d_{x,y},\; y+(t-t')\,d_{x,y}) \\ &= \boldsymbol{U}_{s',t'}\,(x',y')\,. \end{aligned} \tag{1}$$

where $d_{x,y} \equiv \boldsymbol{D}_{s,t}(x,y)$, with $\boldsymbol{D}_{s,t} \in \mathbb{R}^{N \times N}$ the disparity map of view $\boldsymbol{U}_{s,t}$ with respect to its left view $\boldsymbol{U}_{s,t-1}$. A more visual interpretation of Eq. (1) is provided by the *Epipolar Plane Image (EPI)* in Figure 1b, which represents a slice $\boldsymbol{U}(x,\cdot,s,\cdot)^\top \in \mathbb{R}^{M \times N}$ of the light field. This exhibits a clear line pattern, as the projection $\boldsymbol{U}_{s,t}(x,y)$ of point P moves at a constant speed across the other views, with its speed determined by $d_{x,y}$. We stress out that, although $\boldsymbol{U}_{s,t}(x,y)$ is a pixel in the captured light field, all its projections $\boldsymbol{U}_{s',t'}(x',y')$ do not necessarily correspond to actual pixels in the light field views, as x' and y' may not be integer. We refer to the model described by Eq. (1) as the *light field structure*.

Later on, for the sake of clarity, we will denote a light field view either by its angular coordinate (s,t) or by its linear coordinate $k = ((t-1)M+s) \in \{1,2,\ldots,M^2\}$. In particular, we have $\boldsymbol{U}_{s,t} = \boldsymbol{U}_k$ where $\boldsymbol{U}_k$ is the k-th view encountered when visiting the camera array in column major order. Finally, we also handle the light field in a vectorized form, with the following notation:

- $\boldsymbol{u}_{s,t} = \boldsymbol{u}_k \in \mathbb{R}^{N^2}$ is the vectorized form of view $\boldsymbol{U}_{s,t}$,
- $\boldsymbol{u} = [\boldsymbol{u}_1^\top, \boldsymbol{u}_2^\top, \ldots, \boldsymbol{u}_{M^2}^\top]^\top \in \mathbb{R}^{N^2 M^2}$,

where the vectorized form of a matrix is simply obtained by visiting its entries in column major order.

III. Problem formulation

The light field super-resolution problem concerns the recovery of the high resolution light field $\boldsymbol{U}$ from its low resolution counterpart $\boldsymbol{V}$ at resolution $(N/\alpha) \times (N/\alpha) \times M \times M$, with $\alpha \in \mathbb{N}$. Equivalently, we aim at super-resolving each view $\boldsymbol{V}_{s,t} \in \mathbb{R}^{(N/\alpha)\times(N/\alpha)}$ to get its high resolution version $\boldsymbol{U}_{s,t} \in \mathbb{R}^{N \times N}$. We cast the super-resolution problem into the minimization of the following objective function:

$$\begin{aligned} \boldsymbol{u}^* \;&\in\; \underset{\boldsymbol{u}}{\operatorname{argmin}}\; \mathcal{F}\,(\boldsymbol{u}) \\ \text{with} \quad \mathcal{F}\,(\boldsymbol{u}) \;&\equiv\; \mathcal{F}_1\,(\boldsymbol{u}) \;+\; \lambda_2 \mathcal{F}_2\,(\boldsymbol{u}) \;+\; \lambda_3 \mathcal{F}_3\,(\boldsymbol{u}) \end{aligned} \tag{2}$$

where the multipliers λ_2 and λ_3 balance the different terms.

The first term in Eq. (2) enforces the consistency between the high and low resolution versions of the same view, and it is typically referred to as the *data fidelity term*:

$$\mathcal{F}_1\,(\boldsymbol{u}) \;\equiv\; \sum_k \|\boldsymbol{S}\boldsymbol{B}\,\boldsymbol{u}_k \;-\; \boldsymbol{v}_k\|_2^2. \tag{3}$$

where $\boldsymbol{B} \in \mathbb{R}^{N^2 \times N^2}$ and $\boldsymbol{S} \in \mathbb{R}^{(N/\alpha)^2 \times N^2}$ denote a blurring and a sampling matrix, respectively.

The various low resolution views in the light field capture the scene from slightly different perspectives, therefore, details

dropped by digital sensor sampling at one view may survive in another one. Gathering at one view all the complementary information from the other views can augment its resolution. This can be achieved by enforcing that the high resolution view $\boldsymbol{u}_k$ can generate all the other low resolution views $\boldsymbol{v}_{k'}$ in the light field, with $k' \neq k$. The second term in Eq. (2) enforces this constraint for every high resolution view:

$$\mathcal{F}_2(\boldsymbol{u}) \equiv \sum_k \sum_{k' \in \mathcal{N}_k} \| \boldsymbol{S}\boldsymbol{B}\boldsymbol{F}_k^{k'} \boldsymbol{u}_k - \boldsymbol{v}_{k'} \|_2^2 \qquad (4)$$

where the matrix $\boldsymbol{F}_k^{k'} \in \mathbb{R}^{N^2 \times N^2}$ is such that $\boldsymbol{F}_k^{k'} \boldsymbol{u}_k \simeq \boldsymbol{u}_{k'}$ and it is typically referred to as a *warping matrix*, while $\mathcal{N}_k$ denotes a subset of the views with $k \notin \mathcal{N}_k$.

Finally, a regularizer $\mathcal{F}_3$ happens to be necessary in the overall objective function of Eq. (2), as the overall super-resolution problem is ill-posed. We borrow the regularizer from *Graph Signal Processing* [11], and define $\mathcal{F}_3$ as follows:

$$\mathcal{F}_3(\boldsymbol{u}) \equiv \boldsymbol{u}^\top \boldsymbol{L}\, \boldsymbol{u} \qquad (5)$$

where the positive semi-definite matrix $\boldsymbol{L} \in \mathbb{R}^{M^2N^2 \times M^2N^2}$ is the *un-normalized Laplacian* of a graph designed to capture the light field structure. In particular, each pixel in the high resolution light field is modeled as a vertex in a graph, where the edges connect each pixel to its projections on the other views. The quadratic form in Eq. (5) enforces connected pixels to share similar intensity values, thus promoting the light field structure described in Eq. (1).

In particular, we consider an *undirected* weighted graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{W})$, with $\mathcal{V}$ the set of graph vertices, $\mathcal{E}$ the edge set, and $\mathcal{W}$ a function mapping each edge into a non negative real value, referred to as the *edge weight*. The vertex $i \in \mathcal{V}$ corresponds to the entry $\boldsymbol{u}(i)$ of the high resolution light field. The graph can be represented through its *adjacency matrix* $\boldsymbol{W} \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}|}$, with $|\mathcal{V}|$ the number of light field pixels:

$$\boldsymbol{W}(i,j) = \begin{cases} \mathcal{W}(i,j) & \text{if } (i,j) \in \mathcal{E} \\ 0 & \text{otherwise.} \end{cases}$$

Since the graph is assumed to be undirected, the adjacency matrix is symmetric. We can finally rewrite the term $\mathcal{F}_3$ in Eq. (5) as follows:

$$\mathcal{F}_3(\boldsymbol{u}) = \frac{1}{2} \sum_i \sum_j \boldsymbol{W}(i,j) \left(\boldsymbol{u}(i) - \boldsymbol{u}(j)\right)^2. \qquad (6)$$

Eq. (6) shows that the term $\mathcal{F}_3$ penalizes significant intensity variations along highly weighted edges. A weight typically captures the similarity between vertices, therefore the minimization of Eq. (6) leads to an adaptive smoothing, ideally along the EPI lines of Figure 1b in our light field framework.

Differently from the other light field super-resolution methods, the proposed formulation permits to address the recovery of the whole light field altogether, thanks to the global regularizer $\mathcal{F}_3$. The term $\mathcal{F}_2$ permits to augment the resolution of each view without recurring to external data and learning procedures. However, differently from the light field super-resolution approach in [8], the warping matrices in $\mathcal{F}_2$ do not rely on a precise estimation of the disparity at each view. This is possible mainly thanks to the addition of the graph regularizer $\mathcal{F}_3$, that acts on each view as a denoising term based on non local similarities [12] but at the same time constrains the reconstruction of all the views jointly, thus enforcing the full light field structure captured by the graph.

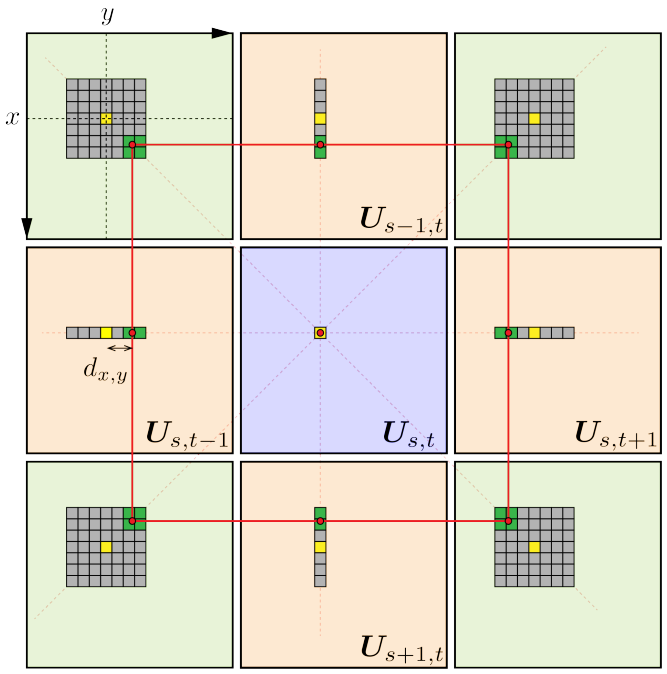


Fig. 2. The light field structure. All the squares indicate pixels, and all the yellow pixels lie at the spatial coordinate (x, y) in their view. The projection of pixel $U_{s,t}(x, y)$ on the eight neighboring views is indicated with a red dot. According to Eq. (1), all the projections are determined by the disparity $d_{x,y}$. The projection of pixel $U_{s,t}(x, y)$ lies between two green pixels in the orange views, and between four green pixels in the green views. The search windows, in gray, are shaped accordingly.

IV. Super-resolution algorithm

We now describe the algorithm that we use to solve the optimization problem in Eq. (2). We first discuss the construction of the graph employed in the regularizer $\mathcal{F}_3$ in Eq. (5), and then show how to extract the warping matrices of the term $\mathcal{F}_2$ in Eq. (4) directly from the constructed graph. Finally, we describe the complete super-resolution algorithm.

A. Regularization graph construction

The effectiveness of the term $\mathcal{F}_3$ depends on the graph capability to capture the light field structure. Ideally, we would like to connect each pixel $\boldsymbol{U}_{s,t}(x, y)$ in the light field to its projections on the other views, as they all share the same intensity value under the Lambertian assumption. However, since the projections do not lie at integer spatial coordinates in general, we rather aim at connecting the pixel $\boldsymbol{U}_{s,t}(x, y)$ to those pixels that are close to its projections on the other views. We thus propose a three-step approach to the computation of the adjacency matrix $\boldsymbol{W}$ of the graph in Eq. (6).

1) Edge weight computation: We consider a view $\boldsymbol{U}_{s,t} = \boldsymbol{U}_k$ and define its set of neighboring views $\mathcal{N}_k$ as the set containing the eight adjacent views, depicted in Figure 2:

$$\{\boldsymbol{U}_{k'} : k' \in \mathcal{N}_k\} \;=\; \{\boldsymbol{U}_{s,t\pm1}, \boldsymbol{U}_{s\pm1,t}, \boldsymbol{U}_{s-1,t\pm1}, \boldsymbol{U}_{s+1,t\pm1}\}\,.$$

We then concentrate on a pixel $\boldsymbol{u}(i) = \boldsymbol{U}_{s,t}(x,y)$ and define its edges toward one neighboring view $\boldsymbol{U}_{k'} = \boldsymbol{U}_{s',t'}$ with $k' \in \mathcal{N}_k$. We center a search window at the pixel $\boldsymbol{U}_{s',t'}(x,y)$ and compute the following weight between the pixel $\boldsymbol{U}_{s,t}(x,y) = \boldsymbol{u}(i)$ and each pixel $\boldsymbol{U}_{s',t'}(x',y') = \boldsymbol{u}(j)$ in the considered window:

$$\boldsymbol{W}_A\,(i,j) \;=\; \exp\left(-\frac{\|\mathcal{P}_{s,t}(x,y) - \mathcal{P}_{s',t'}(x',y')\|_F^2}{\sigma^2}\right), \quad (7)$$

where $\mathcal{P}_{s,t}(x,y)$ denotes a square patch centered at the pixel $\boldsymbol{U}_{s,t}(x,y)$, the operator $\|\cdot\|_F$ denotes the Frobenius norm, and σ is a tunable constant. We repeat the procedure for each one of the eight neighboring views in $\mathcal{N}_k$, but we employ different windows for different views:

- a $1 \times W$ pixel window for $(s',t') = (s, t \pm 1)$,
- a $W \times 1$ pixel window for $(s',t') = (s \pm 1, t)$,
- a $W \times W$ pixel window otherwise.

This is illustrated in Figure 2. The $W \times W$ pixel window is introduced for the diagonal views, in green in Figure 2, as the projection of the pixel $\boldsymbol{U}_{s,t}(x,y)$ on these views lies neither along row x, nor along column y. Iterating the outlined procedure over each pixel $\boldsymbol{u}(i)$ in the light field leads to the construction of the adjacency matrix $\boldsymbol{W}_A$. We regard $\boldsymbol{W}_A$ as the adjacency matrix of a *directed* graph, with $\boldsymbol{W}_A(i,j)$ the weight of the edge from $\boldsymbol{u}(i)$ to $\boldsymbol{u}(j)$.

2) Edge pruning: We want to keep only the most important connections in the graph. We thus perform a pruning of the edges leaving the pixel $\boldsymbol{U}_{s,t}(x,y)$ toward the eight neighboring views. In particular, we keep only

- the two largest weight edges, for $(s',t') = (s, t \pm 1)$,
- the two largest weight edges, for $(s',t') = (s \pm 1, t)$,
- the four largest weight edges, otherwise.

For the diagonal neighboring views $\boldsymbol{U}_{k'} = \boldsymbol{U}_{s',t'}$ we allow four weights rather than two as it is more difficult to detect those pixels that lie close to the projection of $\boldsymbol{U}_{s,t}(x,y)$. We define $\boldsymbol{W}_B$ as the adjacency matrix after the pruning.

3) Symmetric adjacency matrix: We finally carry out the symmetrization of the matrix $\boldsymbol{W}_B$, and set $\boldsymbol{W} \equiv \boldsymbol{W}_B$ in Eq. (6). We adopt a simple approach for obtaining a symmetric matrix: we choose to preserve an edge between two vertexes $\boldsymbol{u}(i)$ and $\boldsymbol{u}(j)$ if and only if both entries $\boldsymbol{W}_B(i,j)$ and $\boldsymbol{W}_B(j,i)$ are non zero. Note that if this is the case, $\boldsymbol{W}_B(i,j) = \boldsymbol{W}_B(j,i)$ necessarily holds true, and the weights are maintained. We observe that this procedure mimics the well-known *left-right disparity check* of stereo vision [13].

B. Warping matrix construction

We recall that the matrix $\boldsymbol{F}_k^{k'}$ is such that $\boldsymbol{F}_k^{k'}\boldsymbol{u}_k \simeq \boldsymbol{u}_{k'}$. In particular, the i-th row of this matrix is expected to compute the pixel $\boldsymbol{u}_{k'}(i) = \boldsymbol{U}_{s',t'}(x,y)$ as a convex combination of those pixels around its projection on $\boldsymbol{U}_k = \boldsymbol{U}_{s,t}$. We thus observe that the sub-matrix $\boldsymbol{W}_S$, obtained by extracting the rows $(k'-1)N^2+1, \ldots, k'N^2$ and the columns $(k-1)N^2, \ldots, kN^2$ from the adjacency matrix $\boldsymbol{W}$, represents a directed weighted graph with edges from the pixels of the view $\boldsymbol{U}_{k'} = \boldsymbol{U}_{s',t'}$ (rows of the matrix) to the pixels of the view $\boldsymbol{U}_k = \boldsymbol{U}_{s,t}$ (columns of the matrix). In this graph, the pixel $\boldsymbol{u}_{k'}(i) = \boldsymbol{U}_{s',t'}(x,y)$ is connected to a subset of pixels that lie close to its projections on $\boldsymbol{U}_k = \boldsymbol{U}_{s,t}$. We thus normalize the rows of $\boldsymbol{W}_S$ such that they sum up to one, in order to implement a convex combination, and set $\boldsymbol{F}_k^{k'} \equiv \widetilde{\boldsymbol{W}}_S$ in Eq. (4) with $\widetilde{\boldsymbol{W}}_S$ the normalized sub-matrix.

C. Optimization algorithm

We now have all the ingredients to solve the optimization problem in Eq. (2). We observe that it corresponds to a quadratic problem and it can be rewritten as follows:

$$\boldsymbol{u}^* \;\in\; \underset{\boldsymbol{u}}{\operatorname{argmin}} \;\underbrace{\frac{1}{2}\,\boldsymbol{u}^\top \boldsymbol{P}\,\boldsymbol{u} \;+\; \boldsymbol{q}^\top \boldsymbol{u} \;+\; r.}_{\mathcal{F}(\boldsymbol{u})} \quad (8)$$

In general the matrix $\boldsymbol{P}$ is positive semi-definite, therefore we choose to adopt the *Proximal Point Algorithm (PPA)*, which iteratively solves Eq. (8) using the following update rule:

$$\begin{aligned}\boldsymbol{u}^{(i+1)} \;&=\; \underset{\boldsymbol{u}}{\operatorname{argmin}} \;\; \mathcal{F}\,(\boldsymbol{u}) \;+\; \frac{1}{2\beta}\|\boldsymbol{u} - \boldsymbol{u}^{(i)}\|_2^2 \\ &=\; \underset{\boldsymbol{u}}{\operatorname{argmin}} \;\underbrace{\frac{1}{2}\,\boldsymbol{u}^\top \left(\boldsymbol{P} + \frac{\boldsymbol{I}}{\beta}\right) \boldsymbol{u} \;+\; \left(\boldsymbol{q} - \frac{\boldsymbol{u}^{(i)}}{\beta}\right)^\top \boldsymbol{u}}_{\mathcal{T}(\boldsymbol{u})}.\end{aligned}$$

The matrix $\boldsymbol{P} + (1/\beta)\boldsymbol{I}$ is positive definite for every $\beta > 0$, hence we can now use the CG method to solve the linear system $\nabla\mathcal{T}(\boldsymbol{u}) = 0$. The full Graph-Based super-resolution algorithm is summarized in Algorithm 1. We observe that the graph construction requires the high resolution light field. In order to bypass this causality problem, a fast and rough high resolution estimation of the light field is computed, e.g., via bilinear interpolation, at the bootstrap phase. Then, at each new iteration, the graph and the warping matrices can be reconstructed on the new available light field estimate.

V. Experiments

A. Experimental settings

We test our Graph-Based super-resolution algorithm (*GB* hereafter) on the *HCI light field dataset* [10], and we compare GB both to a state-of-the-art light field super-resolution algorithm [9] and to a simple bilinear interpolation of the single light field views. The light fields in the HCI dataset are characterized by a 9×9 array of views. Similarly to [9], we crop them to a 5×5 array of views, i.e., we choose $M = 5$.

In our experiments, the spatial resolution of each test light field $\boldsymbol{U}$ is first decreased by a factor $\alpha \in \mathbb{N}$ by applying the blurring and sampling matrix $\boldsymbol{SB}$ of Eq. (3) to each color channel of each view. Then, the low resolution light field $\boldsymbol{V}$ is brought back to the original spatial resolution by the super-resolution algorithms under study. In order to match

Algorithm 1 Graph-Based Light Field Super-Resolution

Input: $\boldsymbol{v} = [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{M^2}]$, $\alpha \in \mathbb{N}$, $\beta > 0$, $iter$.
Output: $\boldsymbol{u} = [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{M^2}]$.

$\boldsymbol{u} \leftarrow$ bilinear interp. of $\boldsymbol{v}_k$ by α, $\forall k = 1, \ldots, M^2$;
for $i = 1$ **to** $iter$ **do**
  build the graph adjacency matrix $\boldsymbol{W}$ on $\boldsymbol{u}$;
  build the matrices $\boldsymbol{F}_k$ on $\boldsymbol{u}$, $\forall k = 1, \ldots, M^2$;
  update the matrix $\boldsymbol{P}$;
  update the vector $\boldsymbol{q}$;
  $\boldsymbol{z} \leftarrow \boldsymbol{u}$; ▷ Initialize CG
  while convergence is reached **do**
    $\boldsymbol{z} \leftarrow \mathrm{CG}(\boldsymbol{P} + (\boldsymbol{I}/\beta), (\boldsymbol{z}/\beta) - \boldsymbol{q})$;
  end while
  $\boldsymbol{u} \leftarrow \boldsymbol{z}$; ▷ Update $\boldsymbol{u}$
end for
return $\boldsymbol{u}$;

the assumptions of the method in [9], but without loss of generality, the blur kernel implemented by the matrix $\boldsymbol{B}$ is set to an $\alpha \times \alpha$ box filter, and the matrix $\boldsymbol{S}$ performs a regular sampling.

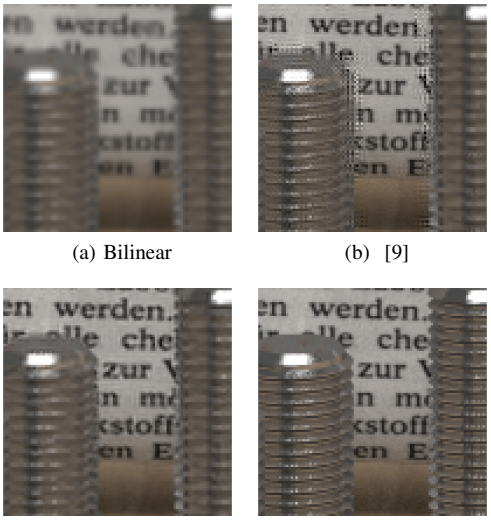
In the graph construction in Eq. (7), we empirically set the size of the patch $\mathcal{P}$ to 7×7 pixels and $\sigma = 0.7229$. For the search window size, we set $W = 13$ pixels. This choice is equivalent to consider a disparity range of $[-6, 6]$ pixels at high resolution. For the HCI dataset the disparity range is within $[-3, 3]$ pixels, but in practice the disparity range is not known a priori, therefore we found the $[-6, 6]$ pixel range to be a fair choice. Finally, we empirically set $\lambda_2 = 0.2$ and $\lambda_3 = 0.0055$ in the objective function in Eq. (2) and perform just two iterations of the full Algorithm 1, as this is experimentally found to be sufficient.

For our experiments on the algorithm in [9] we use the code provided by the authors. We discretize the $[-6, 6]$ pixel range using a 0.2 pixel step, and for each disparity value we train a different GMM prior. The procedure is carried out for $\alpha = 2$ and 3, and results in GMM priors defined on a $4\alpha \times 4\alpha \times M \times M$ light field patch. We perform the training on the data that comes together with the authors' code.

In the experiments, every considered method super-resolves only the luminance of the low resolution light field. The full color high resolution light field is obtained through bilinear interpolation of the two low resolution light field chrominances.

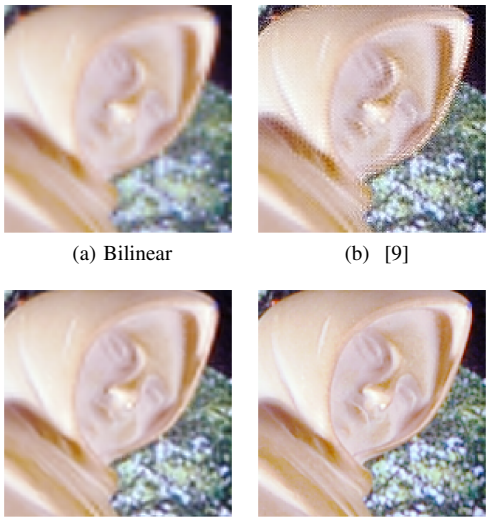
B. Light field reconstruction results

The numerical results from our super-resolution experiments on the HCI dataset are reported in Tables I and II for the super-resolution factors $\alpha = 2$ and 3, respectively. For each reconstructed light field we compute the PSNR (dB) at each view and report the average and variance of the computed PSNRs in the tables. The PSNRs are computed on the luminance of the light field views. Finally, for a fair comparison with the method in [9], which suffers from border effects, a 15-pixel border is removed from all the reconstructed views before the PSNR computation.



(a) Bilinear (b) [9] (c) GB (d) Original HR

Fig. 3. Detail from the bottom right-most view of the light field `horses`. The low resolution light field is super-resolved by a factor $\alpha = 2$ with bilinear interpolation in (a), the method [9] in (b), and GB in (c). The original high resolution light field is provided in (d).



(a) Bilinear (b) [9] (c) GB (d) Original HR

Fig. 4. Detail from the bottom right-most view of the light field `statue`. The low resolution light field is super-resolved by a factor $\alpha = 3$ with bilinear interpolation in (a), the method [9] in (b), and GB in (c). The original high resolution light field is provided in (d).

For a super-resolution factor $\alpha = 2$, GB provides the highest average PSNR on nine out of twelve light fields. The highest average PSNR in the remaining light fields `buddha`, `horses`, and `medieval` is achieved by [9], but the corresponding variances are non negligible. The large variance generally indicates that the quality of the central views is higher than the one of the lateral views. This is clearly non ideal, as our objective is to reconstruct all the views with high quality, as necessary in most light field applications. We also note that GB provides a better visual quality in these three light fields. An example is provided in Figure 3, where a detail from the light field `horses` is given for each method. In particular, the

TABLE I
HCI DATASET - PSNR MEAN AND VARIANCE FOR $\alpha = 2$

	Bilinear	[9]	GB
buddha	35.22 ± 0.00	**39.12** ± 0.62	38.59 ± 0.08
buddha2	30.97 ± 0.00	33.63 ± 0.22	**34.17** ± 0.01
couple	25.52 ± 0.00	31.83 ± 2.80	**32.79** ± 0.17
cube	26.06 ± 0.00	30.99 ± 3.02	**32.60** ± 0.23
horses	26.37 ± 0.00	**33.13** ± 0.72	30.99 ± 0.05
maria	32.84 ± 0.00	37.03 ± 0.44	**37.19** ± 0.03
medieval	30.07 ± 0.00	**33.34** ± 0.71	33.23 ± 0.03
mona	35.11 ± 0.00	38.32 ± 1.14	**39.30** ± 0.04
papillon	36.19 ± 0.00	40.59 ± 0.89	**40.94** ± 0.06
pyramide	26.49 ± 0.00	33.35 ± 4.06	**34.63** ± 0.34
statue	26.32 ± 0.00	32.95 ± 4.67	**34.81** ± 0.38
stillLife	25.28 ± 0.00	28.84 ± 0.82	**30.80** ± 0.07

TABLE II
HCI DATASET - PSNR MEAN AND VARIANCE FOR $\alpha = 3$

	Bilinear	[9]	GB
buddha	32.58 ± 0.01	35.36 ± 0.34	**35.42** ± 0.02
buddha2	28.14 ± 0.00	30.29 ± 0.10	**30.52** ± 0.00
couple	22.62 ± 0.00	**27.43** ± 1.16	26.65 ± 0.01
cube	23.25 ± 0.00	26.48 ± 1.16	**27.23** ± 0.01
horses	24.35 ± 0.00	**29.90** ± 0.55	25.53 ± 0.00
maria	30.02 ± 0.00	33.36 ± 0.37	**33.48** ± 0.01
medieval	28.29 ± 0.00	**29.78** ± 0.50	29.23 ± 0.00
mona	32.05 ± 0.00	33.31 ± 0.40	**34.66** ± 0.01
papillon	33.66 ± 0.00	36.13 ± 0.48	**36.44** ± 0.01
pyramide	23.39 ± 0.00	**29.13** ± 1.86	28.34 ± 0.01
statue	23.21 ± 0.00	**28.93** ± 2.03	28.21 ± 0.01
stillLife	23.28 ± 0.00	**27.23** ± 0.49	24.99 ± 0.00

reconstruction provided by [9] exhibits strong artifacts along object boundaries. This method assumes a constant disparity within each light field patch that it processes, but patches capturing object boundaries are characterized by an abrupt change of disparity that violates this assumption and causes unpleasant artifacts. The bilinear interpolation method provides the lowest PSNRs and the poor quality of its reconstruction is confirmed by the Figure 3a, which is significantly blurred.

For a larger super-resolution factor of $\alpha = 3$, GB provides the highest average PSNRs only on half of the light fields, while the other half is better with the method in [9]. However, the average PSNR happens to be a very misleading index here. In particular, the method in [9] provides the highest average PSNR on the light field `statue`, but the PSNR variance is larger than 2 dB, which indicates a very large difference in the quality of the reconstructed light field views. On the other hand, GB provides a slightly lower average PSNR on the same light field, but its PSNR variance is below 0.01 dB, which suggests a more homogenous quality of the reconstructed views. In particular, the lowest PNSR provided by GB among all the views of `statue` is equal to 27.93 dB, which is more than 2.5 dB higher than the worst case view reconstructed by [9]. Moreover, the light fields reconstructed by [9] again exhibit very strong artifacts along object boundaries. An example is provided in Figure 4, which represents a detail from the light field `statue`. The head of the statue reconstructed by [9] appears very noisy, especially at the depth discontinuity between the head and the background, while GB is not significantly affected. Finally, the worst numerical results are provided by the bilinear interpolation method, which does not exhibit strong artifacts, but provides very blurred images, as shown in Figure 4a.

For more extensive experiments we refer the reader to [14].

VI. Conclusions

We developed a new light field super-resolution algorithm that exploits the complementary information encoded in the different views to augment their spatial resolution, and that relies on a graph to regularize the target light field. In particular, we showed that the introduction of a graph enforcing the light field structure permits the use of coarse warping matrices, thus avoiding an explicit and costly disparity estimation step on each view. The proposed algorithm compares favorably to the state-of-the-art in light field super-resolution, both in terms of PSNR and visual quality. Moreover, it reduces to a simple quadratic optimization problem, which can be solved efficiently with standard convex optimization tools.

References

[1] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," *Computer Science Technical Report CSTR*, vol. 2, no. 11, pp. 1–11, 2005.

[2] M. Levoy and P. Hanrahan, "Light field rendering," in *Proceedings of the 23rd ACM annual conference on Computer graphics and interactive techniques*, 1996, pp. 31–42.

[3] Jianchao Yang, J. Wright, T. S. Huang, and Yi Ma, "Image Super-Resolution Via Sparse Representation," *IEEE Transactions on Image Processing*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.

[4] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. Alberi Morel, "Single-Image Super-Resolution via Linear Mapping of Interpolated Self-Examples," *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5334–5347, Dec. 2014.

[5] Y. Yoon, H.-G. Jeon, D. Yoo, J.-Y. Lee, and I. S. Kweon, "Light-Field Image Super-Resolution Using Convolutional Neural Network," *IEEE Signal Processing Letters*, vol. 24, no. 6, pp. 848–852, Jun. 2017.

[6] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graph. Models Image Process.*, vol. 53, no. 3, pp. 231–239, Apr. 1991.

[7] S. Farsiu, M. Robinson, M. Elad, and P. Milanfar, "Fast and Robust Multiframe Super Resolution," *IEEE Transactions on Image Processing*, vol. 13, no. 10, pp. 1327–1344, Oct. 2004.

[8] S. Wanner and B. Goldluecke, "Spatial and angular variational super-resolution of 4d light fields," in *European Conference on Computer Vision*. Springer, 2012, pp. 608–621.

[9] K. Mitra and A. Veeraraghavan, "Light field denoising, light field superresolution and stereo camera based refocussing using a GMM light field patch prior," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2012, pp. 22–28.

[10] S. Wanner, S. Meister, and B. Goldluecke, "Datasets and Benchmarks for Densely Sampled 4d Light Fields." in *Proceedings of the VMV*, 2013, pp. 225–226.

[11] 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, May 2013.

[12] A. Kheradmand and P. Milanfar, "A General Framework for Regularized, Similarity-Based Image Restoration," *IEEE Transactions on Image Processing*, vol. 23, no. 12, pp. 5136–5151, Dec. 2014.

[13] P. Fua, "A parallel stereo algorithm that produces dense depth maps and preserves image features," *Machine vision and applications*, vol. 6, no. 1, pp. 35–49, 1993.

[14] M. Rossi and P. Frossard, "Light Field Super-Resolution Via Graph-Based Regularization," *CoRR*, 2017. [Online]. Available: http://arxiv.org/abs/1701.02141