Geometry-Consistent Light Field Super-Resolution via Graph-Based Regularization

Mattia Rossi and Pascal Frossard

***Abstract*— Light field cameras 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 suffer by design from strong limitations in their spatial resolution. Off-the-shelf super-resolution algorithms are not ideal for light field data, as they do not consider its structure. On the other hand, the few super-resolution algorithms explicitly tailored for light field data exhibit significant limitations, such as the need to carry out a costly disparity estimation procedure with sub-pixel precision. We propose a new light field super-resolution algorithm meant to address these limitations. We use the complementary information in the different light field views to augment the spatial resolution of the whole light field at once. In particular, we show that coupling the multi-view approach with a graph-based regularizer, which enforces the light field geometric structure, permits to avoid the need of a precise and costly disparity estimation step. Extensive experiments show that the new algorithm compares favorably to the state-of-the-art methods for light field super-resolution, both in terms of visual quality and in terms of reconstruction error.**

***Index Terms*— Light field, super-resolution, graph, regularization, multi-view system, camera array.**

I. Introduction

We LIVE in a 3D world but the pictures taken with traditional cameras can capture just 2D projections of this reality. The *light field* is a model that has been originally introduced in the context of image-based rendering with the purpose of capturing richer information in a 3D scene [1]. The light emitted by the scene is modeled in terms of rays, each one characterized by a direction and a radiance value. The light field function provides, at each point in space, the radiance from a given direction. The rich information captured by the light field function could be used in many applications, from post-capture refocus to depth estimation or virtual reality.

However, the light field is a theoretical model: in practice the light field function has to be properly sampled, which is a challenging task. A straightforward but hardware-intensive approach relies on camera arrays [2]. In this setup, each camera records an image of the same scene from a particular position and the light field takes the form of an array of views. More recently, the development of the first commercial light field cameras [3], [4] has made light field sampling more accessible. In light field cameras, a micro lens array placed between the main lens and the sensor permits to virtually partition the main lens into sub-apertures, whose images are recorded altogether in a single exposure [5], [6]. As a consequence, a light field camera behaves as a compact camera array, providing multiple images of a 3D scene from slightly different points of view.

Even if light field cameras become very appealing, they still face the so called *spatio-angular resolution tradeoff*. Since the whole array of views is captured by a single sensor, a dense sampling of the light field in the angular domain (i.e., a large number of views) necessarily translates into a sparse sampling in the spatial domain (i.e., low resolution views) and vice versa. A dense angular sampling is at the basis of any light field application, as the 3D information provided by light field data comes from the availability of different views. It follows that the angular sampling cannot be excessively penalized to favor the spatial resolution. Moreover, even in the limit scenario of a light field with just two views, the spatial resolution of each one may be reduced to half of the sensor [5]. Consequently, 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.

In this work, we propose a new light field super-resolution algorithm that augments the resolution of all the views together, while relying only on an approximate disparity estimation procedure. In particular, we propose to cast *light field spatial super-resolution* into a global optimization problem, whose objective function is designed to capture the relations between the light field views. The 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 smoothness along the light field epipolar lines that define the light field geometric structure. These terms altogether form a quadratic objective function that we minimize iteratively with the Proximal Point Algorithm. The results show that our algorithm compares favorably to state-of-the-art light field

Manuscript received August 15, 2017; revised February 20, 2018; accepted April 10, 2018. Date of publication April 20, 2018; date of current version June 1, 2018. This work was supported by the Swiss National Science Foundation under Grant NURIS 200021_149800. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jean-Francois Aujol. *(Corresponding author: Mattia Rossi.)*

The authors are with the Signal Processing Laboratory, École Polytechnique Fédérale de Lausanne, 1015 Lausanne, Switzerland (e-mail: mattia.rossi@epfl.ch; pascal.frossard@epfl.ch).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2018.2828983

super-resolution algorithms, both in terms of visual quality and in terms of reconstruction error. This article extends our previous work in [7]. In particular, we present a new approach to the construction of the warping term, where we consider the light field geometric structure explicitly, we also provide experiments on two different types of datasets and comparisons with a larger number of super-resolution algorithms.

The article is organized as follows. Section II presents an overview of the related literature. Section III formalizes the light field structure. Section IV presents our problem formulation and carefully analyzes each of its terms. Section V provides a detailed description of our super-resolution algorithm. Section VI is dedicated to our experiments. Finally, Section VII concludes the article.

II. Related Work

The super-resolution literature is quite vast, but it can be divided mainly into two areas: single-frame and multi-frame super-resolution methods. In single-frame super-resolution, only one image from a scene is provided and its resolution has to be increased. This goal is typically achieved by learning a mapping from the low resolution data to the high resolution one, either on an external training set [8]–[10] or on the image itself [11], [12]. Single-frame algorithms can be applied to each light field view separately in order to augment the resolution of the whole light field, but this approach would neither exploit the high correlation among the views, nor enforce the consistency among them.

In the multi-frame scenario, multiple images of the same scene are used to increase the resolution of a target image. To this purpose, a global image warping model is assumed, where all the available images are treated as translated and rotated versions of the target one [13], [14]. The multi-frame super-resolution scenario resembles the light field one, but its global image warping model does not fit the light field structure. In particular, the different moving speeds of the objects in the scene across the light field views, which encode their different depths, cannot be captured by a global warping model. Multi-frame algorithms employing more complex warping models exist, for example in video super-resolution [15], [16], yet the warping models do not exactly fit the geometry of light field data and their construction is computationally demanding. In particular, multi-frame video super-resolution involves two main steps, namely optical flow estimation, which finds correspondences between temporally successive frames, and eventually a super-resolution step that is built on the optical flow.

In the light field representation, the views lie on a two-dimensional grid with adjacent views sharing a constant baseline under the assumption of both vertical and horizontal registration. As a consequence, not only the optical flow computation reduces to disparity estimation, but also the disparity map at one view determines its warping to every other view in the light field, in the absence of occlusions. Wanner and Goldluecke [17] build over these observations to extract the disparity map at each view directly from the epipolar line slopes with the help of a structure tensor operator. The estimated low resolution disparity maps are upsampled and, similarly to multi-frame super-resolution, all the views are projected to the target one within a global optimization formulation endowed with a *Total Variation (TV)* prior. Although the structure tensor operator permits to carry out disparity estimation with continuous precision, this task remains very challenging at low spatial resolution. As a result, the disparity errors translate into significant artifacts in the textured areas and along the object edges of the super-resolved target view. Finally, each light field view is super-resolved separately in [17], which does not permit to fully exploit the inter-view dependencies.

In another work, Heber and Pock [18] consider the matrix obtained by warping all the views to a reference one, and they propose to model it as the sum of a low rank matrix and a noise one, where the later describes the noise and occlusions. This model, that resembles *Robust PCA* [19], is primarily meant for disparity estimation at the reference view. However, the authors show that a slight modification of the objective function can provide the corresponding high resolution view, in addition to the low resolution disparity map at the reference view. The algorithm could ideally be applied separately to each view in order to super-resolve the whole light field, but again this would not permit to fully exploit the inter-view dependencies.

In a different framework, Mitra and Veeraraghavan [20] propose a light field super-resolution algorithm based on a learning procedure. Each view in the low resolution light field is divided into patches that are possibly overlapping. 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 (GMM)* prior for high resolution light field patches is learnt offline for each discrete disparity value, and it is then employed within a *MAP* estimator to super-resolve each light field patch with the corresponding disparity. However, the learning strategy has some drawbacks: the dependency of the reconstruction on the chosen training set, the need for a new training for each super-resolution factor, and finally the need for a proper discretization of the disparity range, which introduces a tradeoff between the reconstruction quality and the time required by both the training and the reconstruction steps. Moreover, 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.

The light field super-resolution problem has been addressed within the framework of *Convolutional Neural Networks (CNNs)* too. In particular, Yoon *et al.* [21] consider the cascade of two CNNs, the first meant to super-resolve the given light field views, and the second to synthesize new high resolution views based on the previously super-resolved ones. However, the first CNN (whose design is borrowed from [10]) is meant for single-frame super-resolution, therefore the views are super-resolved independently, without considering the light field structure.

Finally, we note that some authors, e.g., Bishop and Favaro [22], consider the recovery of an all-in-focus image with full sensor resolution from the light field camera output.

They refer to this task as light field super-resolution although it is different from the problem considered in this work. In this article, no light field application is considered a priori: the light field views are all super-resolved, thus enabling any light field application to be performed later at a resolution higher than the original one. Differently from the other light field super-resolution algorithms, our new method does not rely on a precise and costly disparity estimation step, and it does not rely on a learning procedure. Moreover, our algorithm reconstructs all the views jointly, provides homogeneous quality across the reconstructed views, and it preserves the light field structure.

III. Light Field Structure

In the light field literature, it is common to parametrize the light rays from a 3D scene by the coordinates of their intersection with two parallel planes, typically referred to as the *spatial plane* Ω and the *angular plane* Π. Each light ray is associated to a radiance value, and a pinhole camera with its aperture on the plane Π and its sensor on the plane Ω can record the radiance of all those rays accommodated by its aperture. This is represented in Figure 1, where each pinhole camera is represented as a pyramid, with its vertex and basis representing the camera aperture and sensor, respectively. In general, an array of pinhole cameras can perform a regular sampling of the angular plane Π, therefore the sampled light field takes the form of a set of images captured from different points of view. This is the sampling scheme approximated by both camera arrays and light field cameras.

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. Each camera is identified through the angular coordinates (s, t) with $s, t \in \{1, 2, \dots, M\}$, while a pixel within the camera sensor is identified through the spatial coordinates (x, y) with $x, y \in \{1, 2, \dots, N\}$. The distance between the apertures of horizontally or vertically adjacent cameras is b, referred to as the *baseline*. The distance between the planes Π and Ω is f, referred to as the camera *focal length*. Figure 1 sketches two cameras of the $M \times M$ array. Within this setup, we can represent the light field as an $N \times N \times M \times M$ real tensor $\boldsymbol{U}$, with $\boldsymbol{U}(x, y, s, t)$ the intensity of the pixel with coordinates (x, y) in the view of the camera at (s, t). In particular, we denote the view at (s, t) as $\boldsymbol{U}_{s,t} = \boldsymbol{U}(\cdot, \cdot, s, t) \in \mathbb{R}^{N \times N}$. Finally, without loss of generality, we assume that each pair of horizontally or vertically adjacent views in the light field are registered.

With reference to Figure 1, we now describe in more details the particular structure of light field data. We consider a point P at depth z from Π, whose projection on one of the cameras is represented by the pixel $\boldsymbol{U}_{s,t}(x, y)$, in the right view of Figure 1. We now look at the projection of P on the other views $\boldsymbol{U}_{s,t'}$ in the same row of the camera array, such as the left view in Figure 1. We observe that, in the absence of occlusions and under the Lambertian assumption (i.e., all the rays emitted by the point P exhibit the same radiance), the projection of P obeys the following stereo equation:

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

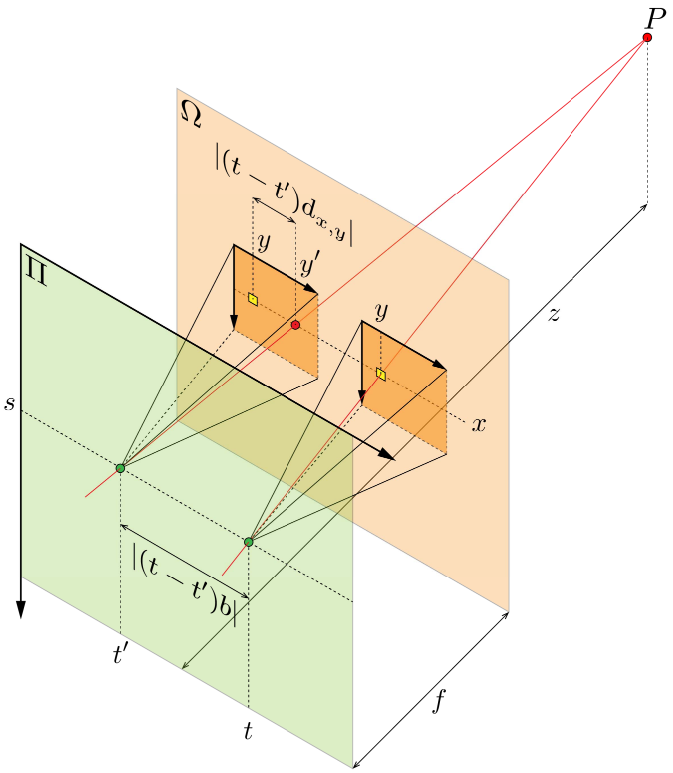


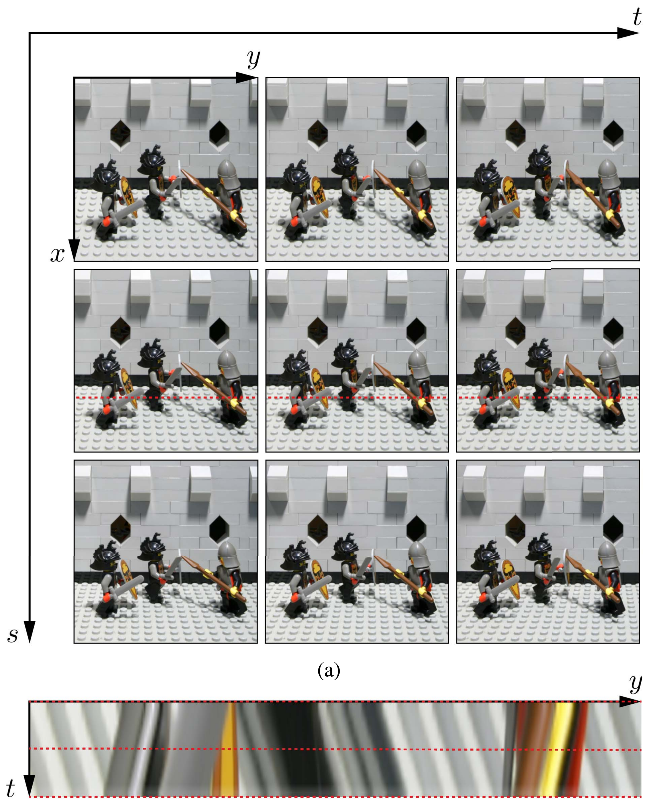
Fig. 1. Light field sampling in the two plane parametrization. The light field is ideally sampled through an $M \times M$ array of pinhole cameras. The pinhole cameras at coordinates (s, t) and (s, t') in the camera array are represented as two pyramids, with their apertures denoted by the two green dots on the plane Π, and their $N \times N$ sensors represented by the two orange squares on the plane Ω. The distance between the two planes is the *focal length* f. The distance between the apertures of horizontally or vertically adjacent views in the $M \times M$ array is the *baseline* b, therefore the distance between the two green dots on the plane Π is $|(t - t')b|$. The small yellow squares in the two sensors denote the pixel (x, y). The pixel (x, y) of the camera at (s, t) captures one of the light rays (in red) emitted by a point P at depth z in the scene. The disparity associated to the pixel (x, y) of the camera at (s, t) is $d_{x,y}$, therefore the projection of P on the sensor of the camera at (s, t') lies at $(x, y') = (x, y + (t - t')d_{x,y})$. The intersection point (x, y') is denoted by a red spot, as it does not necessarily correspond to a pixel, since $d_{x,y}$ is not necessarily integer.

where $d_{x,y} = \boldsymbol{D}_{s,t}(x, y) = fb/z$, with $\boldsymbol{D}_{s,t} \in \mathbb{R}^{N \times N}$ the disparity map of the view $\boldsymbol{U}_{s,t}$ with respect to its left view $\boldsymbol{U}_{s,t-1}$. A visual interpretation of Eq. (1) is provided by the *Epipolar Plane Image (EPI)* [23] in Figure 2b, 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 the point P moves at a constant speed across the other views $\boldsymbol{U}_{s,t'}$, 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 y' may not be integer. We finally observe that Eq. (1) can be extended to the whole light field:

$$\begin{aligned} \boldsymbol{U}_{s,t}(x, y) &= \boldsymbol{U}_{s',t'}\left(x + \left(s - s'\right) d_{x,y}, y + \left(t - t'\right) d_{x,y}\right) \\ &= \boldsymbol{U}_{s',t'}\left(x', y'\right). \end{aligned} \tag{2}$$

We refer to the model in Eq. (2) as the *light field structure*.

Later on, for the sake of clarity, we will denote a light field view either by its angular coordinates (s, t) or by its linear coordinate $k = ((t - 1)M + s) \in \{1, 2, \dots, M^2\}$. In particular, we have $\boldsymbol{U}_{s,t} = \boldsymbol{U}_k$ where $\boldsymbol{U}_k$ is the k-th view encountered



(a)

(b)

Fig. 2. Example of light field and Epipolar Plane Image (EPI). Figure (a) shows an array of 3 × 3 views, extracted from the `knights` light field (Stanford dataset [24]) which actually consists of an array of 17 × 17 views. Figure (b) shows an EPI from the original 17 × 17 `knights` light field. In particular, the t-th row in the EPI corresponds to the row $\boldsymbol{U}_{9,t}(730, \cdot)$. The top, central, and bottom red dashed rows in (b) correspond to the left, central, and right dashed rows in red in the sample views in (a), respectively.

when visiting the camera array in column major order. 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 the view $\boldsymbol{U}_{s,t}$,
- $\boldsymbol{u} = [\boldsymbol{u}_1^\top, \boldsymbol{u}_2^\top, \dots, \boldsymbol{u}_{M^2}^\top]^\top \in \mathbb{R}^{N^2 M^2}$,

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

IV. Problem Formulation

The light field (spatial) 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 $\alpha \in \mathbb{N}$ the super-resolution factor. 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 $\boldsymbol{U}_{s,t} \in \mathbb{R}^{N \times N}$. In order to recover the high resolution light field from the low resolution data, we propose to minimize the following objective function:

$$\boldsymbol{u}^* \in \underset{\boldsymbol{u}}{\operatorname{argmin}} \underbrace{\mathcal{F}_1(\boldsymbol{u}) + \lambda_2 \mathcal{F}_2(\boldsymbol{u}) + \lambda_3 \mathcal{F}_3(\boldsymbol{u})}_{\mathcal{F}(\boldsymbol{u})} \tag{3}$$

where each term describes one of the constraints about the light field structure and the multipliers λ_2 and λ_3 balance the different terms. We now analyze each one of them separately.

Each pair of high and low resolution views have to be consistent, and we model their relationship as follows:

$$\boldsymbol{v}_k = \boldsymbol{S}\boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{n}_k \tag{4}$$

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, and the vector $\boldsymbol{n}_k \in \mathbb{R}^{(N/\alpha)^2}$ captures possible inaccuracies of the assumed model. The first term in Eq. (3) enforces the constraint in Eq. (4) for each high resolution and low resolution view pair, and it is typically referred to as the *data fidelity term*:

$$\mathcal{F}_1(\boldsymbol{u}) = \sum_k \|\boldsymbol{S}\boldsymbol{B}\boldsymbol{u}_k - \boldsymbol{v}_k\|_2^2. \tag{5}$$

Then, 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 others 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 $k' \neq k$. For every view $\boldsymbol{u}_k$ we thus have the following model:

$$\boldsymbol{v}_{k'} = \boldsymbol{S}\boldsymbol{B}\boldsymbol{F}_k^{k'}\boldsymbol{u}_k + \boldsymbol{n}_k^{k'}, \quad \forall k' \neq k \tag{6}$$

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*. The vector $\boldsymbol{n}_k^{k'}$ captures possible inaccuracies of the model, such as the presence of pixels of $\boldsymbol{v}_{k'}$ that cannot be generated because they correspond to occluded areas in $\boldsymbol{u}_k$. The second term in Eq. (3) enforces the constraint in Eq. (6) for every high resolution view:

$$\mathcal{F}_2(\boldsymbol{u}) = \sum_k \sum_{k' \in \mathcal{N}_k^+} \|\boldsymbol{H}_k^{k'}\left(\boldsymbol{S}\boldsymbol{B}\boldsymbol{F}_k^{k'}\boldsymbol{u}_k - \boldsymbol{v}_{k'}\right)\|_2^2 \tag{7}$$

where the matrix $\boldsymbol{H}_k^{k'} \in \mathbb{R}^{(N/\alpha)^2 \times (N/\alpha)^2}$ is diagonal and binary, and it masks those pixels of $\boldsymbol{v}_{k'}$ that cannot be generated due to occlusions in $\boldsymbol{u}_k$, while $\mathcal{N}_k^+$ denotes a subset of the views (potentially all) with $k \notin \mathcal{N}_k^+$.

Finally, a regularizer $\mathcal{F}_3$ happens to be necessary in the overall objective function of Eq. (3), as the original problem in Eq. (4), and encoded in the term $\mathcal{F}_1$, is ill-posed due to the fat matrix $\boldsymbol{S}$. The second term $\mathcal{F}_2$ can help, but the warping matrices $\boldsymbol{F}_k^{k'}$ in Eq. (7) are not known exactly, such that the third term $\mathcal{F}_3$ is necessary. We borrow the regularizer from *Graph Signal Processing (GSP)* [25] and define $\mathcal{F}_3$ as follows:

$$\mathcal{F}_3(\boldsymbol{u}) = \boldsymbol{u}^\top \boldsymbol{L}\boldsymbol{u} \tag{8}$$

where the positive semi-definite matrix $\boldsymbol{L} \in \mathbb{R}^{(NM)^2 \times (NM)^2}$ 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. (8) enforces connected pixels to share similar intensity values, thus promoting the light field structure described in Eq. (2).

More precisely, 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*:

$$\mathcal{W} : \mathcal{E} \subseteq (\mathcal{V} \times \mathcal{V}) \rightarrow \mathbb{R}, \quad (i, j) \mapsto \mathcal{W}(i, j).$$

The vertex $i \in \mathcal{V}$ corresponds to the entry $\boldsymbol{u}(i)$ of the high resolution light field, therefore 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 pixels in the light field:

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

Since the graph is assumed to be undirected, the adjacency matrix is symmetric: $\boldsymbol{W}(i, j) = \boldsymbol{W}(j, i)$. Finally, we can rewrite the term $\mathcal{F}_3$ in Eq. (8) as follows:

$$\mathcal{F}_3(\boldsymbol{u}) = \frac{1}{2} \sum_i \sum_{j \sim i} \boldsymbol{W}(i, j) (\boldsymbol{u}(i) - \boldsymbol{u}(j))^2 \tag{9}$$

where $j \sim i$ denotes the set of the vertices j directly connected to the vertex i, and we recall that the scalar $\boldsymbol{u}(i)$ is the i-th entry of the vectorized light field $\boldsymbol{u}$. In Eq. (9) the term $\mathcal{F}_3$ penalizes significant intensity variations along highly weighted edges. A weight typically captures the similarity between two vertices, therefore the minimization of Eq. (9) leads to an adaptive smoothing [25], ideally along the EPI lines of Figure 2b 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 video super-resolution or the light field super-resolution approach in [17], the warping matrices in $\mathcal{F}_2$ rely only on a rough estimation of the disparity at each view. This is possible thanks to the graph regularizer $\mathcal{F}_3$, which acts on each view as a denoising term based on nonlocal similarities [26] but at the same time enforces the full light field structure captured by the graph.

V. Super-Resolution Algorithm

We now describe the algorithm that we use to solve the optimization problem in Eq. (3). We first discuss the construction of the warping matrices of the term $\mathcal{F}_2$ in Eq. (7) and then the construction of the graph in the regularizer $\mathcal{F}_3$ in Eq. (8). Finally, we describe the complete super-resolution algorithm.

A. Warping Matrix Construction

We define the set of the neighboring views $\mathcal{N}_k^+$ in the term $\mathcal{F}_2$ in Eq. (7) as containing only the four views $\boldsymbol{U}_{k'}$ adjacent to $\boldsymbol{U}_k$ in the light field:

$$\left\{\boldsymbol{U}_{k'} : k' \in \mathcal{N}_k^+\right\} = \left\{\boldsymbol{U}_{s,t\pm1}, \boldsymbol{U}_{s\pm1,t}\right\}.$$

This choice reduces the number of the warping matrices but at the same time does not limit our problem formulation, as the interlaced structure of the term $\mathcal{F}_2$ constrains together also those pairs of views that are not explicitly constrained in $\mathcal{F}_2$.

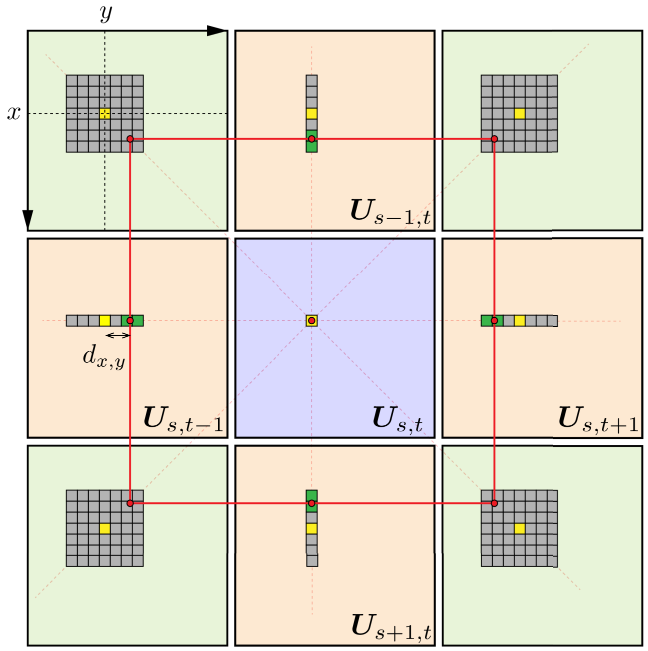


Fig. 3. The neighboring views and the square constraint. The small squares indicate pixels, and each rectangular cluster of pixels within a view represents a search window. All the yellow pixels lie at the spatial coordinates (x, y) in their own view, therefore every search window is centered at (x, y). The projection of the pixel $U_{s,t}(x, y)$ on the eight neighboring views is indicated with a red dot. According to Eq. (2), all the projections lie on a square, highlighted in red. The four orange views represent the set $\mathcal{N}_k^+$, used in the warping matrix construction. In each one of these four views, the two adjacent pixels employed in the convex combination targeting the pixel $U_{s,t}(x, y)$ are indicated in green, they enclose the projection of the pixel $U_{s,t}(x, y)$, and they belong to a 1D search window. The four green views represent the set $\mathcal{N}_k^\times$, used in the graph construction, and they all host a 2D search window.

The inner summation in Eq. (7) considers the set of the four warping matrices $\{\boldsymbol{F}_k^{k'} : k' \in \mathcal{N}_k^+\}$ that warp the view $\boldsymbol{U}_k$ to each one of the four views $\boldsymbol{U}_{k'}$. Conversely, but without loss of generality, in this section we consider the set of the four warping matrices $\{\boldsymbol{F}_{k'}^k : k' \in \mathcal{N}_k^+\}$ that warp each one of the four views $\boldsymbol{U}_{k'}$ to the view $\boldsymbol{U}_k$. The warping 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 the matrix $\boldsymbol{F}_{k'}^k$ computes the pixel $\boldsymbol{u}_k(i) = \boldsymbol{U}_k(x, y) = \boldsymbol{U}_{s,t}(x, y)$ as a convex combination of those pixels around its projection on $\boldsymbol{U}_{k'} = \boldsymbol{U}_{s',t'}$. Note that the convex combination is necessary, as the projection does not lie at integer spatial coordinates in general. The exact position of the projection on $\boldsymbol{U}_{s',t'}$ is determined by the disparity value $d_{x,y}$ associated to the pixel $\boldsymbol{U}_{s,t}(x, y)$. This is represented in Figure 3, which shows that the projections of $\boldsymbol{U}_{s,t}(x, y)$ on the four neighboring views lie on the edges of a virtual square (in red) centered on the pixel $\boldsymbol{U}_{s,t}(x, y)$ and whose size depends on the disparity value $d_{x,y}$.

We estimate roughly the disparity value $d_{x,y}$ by finding a $\delta \in \mathbb{Z}$ such that $d_{x,y} \in [\delta, \delta + 1]$. In details, we first define a similarity score between the target pixel $\boldsymbol{U}_{s,t}(x, y)$ and a generic pixel $\boldsymbol{U}_{s',t'}(x', y')$ as follows:

$$w_{s',t'}\left(x', y'\right) = \exp\left(-\frac{\|\mathcal{P}_{s,t}(x, y) - \mathcal{P}_{s',t'}(x', y')\|_F^2}{\sigma^2}\right) \tag{10}$$

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. Then, we center a search window at $\boldsymbol{U}_{s',t'}(x, y) = \boldsymbol{U}_{k'}(x, y)$ in each one of the four views with $k' \in \mathcal{N}_k^+$, as represented in Figure 3. In particular, we consider

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

with $W \in \mathbb{N}$ defining the disparity range assumed for the whole light field, i.e., $d_{x,y} \in [-\lfloor W/2 \rfloor, \lfloor W/2 \rfloor]$. We introduce the following function, which assigns a score to each integer disparity value $d \in \{-\lfloor W/2 \rfloor, \ldots, \lfloor W/2 \rfloor\}$:

$$\begin{aligned}\mathcal{S}(d) = {} & w_{s,t-1}(x, y+d) + w_{s,t+1}(x, y-d) \\ & + w_{s-1,t}(x+d, y) + w_{s+1,t}(x-d, y).\end{aligned}$$

In order to estimate $\delta \in \mathbb{Z}$ such that $d_{x,y} \in [\delta, \delta+1]$, we first compute the disparity value d_1^* with the highest score:

$$d_1^* = \underset{d \in \{-\lfloor W/2 \rfloor, \ldots, \lfloor W/2 \rfloor\}}{\operatorname{argmax}} \mathcal{S}(d).$$

We select d_1^* to be one of the two extrema of $[\delta, \delta+1]$. We select the other extremum, denoted with d_2^*, as the disparity value with the highest score between $d_1^* - 1$ and $d_1^* + 1$:

$$d_2^* = \underset{d \in \{d_1^*-1, d_1^*+1\}}{\operatorname{argmax}} \mathcal{S}(d),$$

where we assume $\mathcal{S}(d) = 0$ for $d \notin \{-\lfloor W/2 \rfloor, \ldots, \lfloor W/2 \rfloor\}$. Finally, since d_1^* and d_2^* are the two extrema of $[\delta, \delta+1]$, we define δ as follows:

$$\delta = \min\left(d_1^*, d_2^*\right).$$

We can now fill the i-th row of the matrix $\boldsymbol{F}_{k'}^k$ such that the pixel $\boldsymbol{u}_k(i) = \boldsymbol{U}_{s,t}(x, y)$ is computed as the convex combination of the two closest pixels to its projection on $\boldsymbol{U}_{k'} = \boldsymbol{U}_{s',t'}$, namely the following two pixels:

$$\begin{aligned}
&\{\boldsymbol{U}_{k'}(x, y+\delta), \boldsymbol{U}_{k'}(x, y+\delta+1)\} \text{ for } (s', t') = (s, t-1),\\
&\{\boldsymbol{U}_{k'}(x, y-\delta-1), \boldsymbol{U}_{k'}(x, y-\delta)\} \text{ for } (s', t') = (s, t+1),\\
&\{\boldsymbol{U}_{k'}(x+\delta, y), \boldsymbol{U}_{k'}(x+\delta+1, y)\} \text{ for } (s', t') = (s-1, t),\\
&\{\boldsymbol{U}_{k'}(x-\delta-1, y), \boldsymbol{U}_{k'}(x-\delta, y)\} \text{ for } (s', t') = (s+1, t),
\end{aligned}$$

which are indicated in green in Figure 3. Once the two pixels involved in the convex combination at the i-th row of the matrix $\boldsymbol{F}_{k'}^k$ are determined, the i-th row can be constructed. As an example, let us focus on the left neighboring view $\boldsymbol{U}_{k'} = \boldsymbol{U}_{s,t-1}$. The two pixels involved in the convex combination at the i-th row of the matrix $\boldsymbol{F}_{k'}^k$ are the following:

$$\{\boldsymbol{u}_{k'}(j_1) = \boldsymbol{U}_{k'}(x, y+\delta), \boldsymbol{u}_{k'}(j_2) = \boldsymbol{U}_{k'}(x, y+\delta+1)\}.$$

We thus define the i-th row of the matrix $\boldsymbol{F}_{k'}^k$ as follows:

$$\boldsymbol{F}_{k'}^k(i, j) = \begin{cases} w_{s,t-1}(x, y+\delta) / w & \text{if } j = j_1 \\ w_{s,t-1}(x, y+\delta+1) / w & \text{if } j = j_2 \\ 0 & \text{otherwise} \end{cases}$$

with $w = w_{s,t-1}(x, y+\delta) + w_{s,t-1}(x, y+\delta+1)$. In particular, each one of the two pixels in the convex combination has a weight that is proportional to its similarity to the target pixel $\boldsymbol{U}_{s,t}(x, y)$. For the remaining three neighboring views we proceed similarly.

We stress out that, for each pixel $\boldsymbol{u}_k(i) = \boldsymbol{U}_{s,t}(x, y)$, the outlined procedure fills the i-th row of each one of the four matrices $\boldsymbol{F}_{k'}^k$ with $k' \in \mathcal{N}_k^+$. As illustrated in Figure 3, the pair of pixels selected in each one of the four neighboring views encloses one edge of the red square hosting the projections of the pixel $\boldsymbol{U}_{s,t}(x, y)$, therefore this procedure contributes to enforce the light field structure in Eq. (2). Later on, we will refer to this particular structure as the *square constraint*.

Finally, since occlusions are mostly handled by the regularizer $\mathcal{F}_3$, we use the masking matrix $\boldsymbol{H}_{k'}^k$ in Eq. (7) to handle only the trivial occlusions due to the image borders.

B. Regularization Graph Construction

The effectiveness of the term $\mathcal{F}_3$ depends on the graph capability to capture the underlying structure of the light field. 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 these all share the same intensity value under the Lambertian assumption. However, since the projections do not lie at integer spatial coordinates in general, we adopt a procedure similar to the warping matrix construction and we 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 graph adjacency matrix $\boldsymbol{W}$ in Eq. (9).

1) Edge Weight Computation: We consider a view $\boldsymbol{U}_{s,t}$ and define its set of neighboring views $\mathcal{N}_k$ as follows:

$$\mathcal{N}_k = \mathcal{N}_k^+ \cup \mathcal{N}_k^\times$$

where we extend the neighborhood considered in the warping matrix construction with the four diagonal views. In particular, $\mathcal{N}_k^\times$ is defined as follows:

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

The full set of neighboring views is represented in Figure 3, with the views in $\mathcal{N}_k^+$ in orange, and those in $\mathcal{N}_k^\times$ in green. We then consider 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 similarity score 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:

$$\boldsymbol{W}_A(i, j) = \exp\left(-\frac{\|\mathcal{P}_{s,t}(x, y) - \mathcal{P}_{s',t'}(x', y')\|_F^2}{\sigma^2}\right), \quad (11)$$

with the notation already introduced in Section V-A. We repeat the procedure for each one of the eight neighboring views in $\mathcal{N}_k$, but we use differently shaped windows at 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 3. The $W \times W$ pixel window is introduced for the diagonal views $\boldsymbol{U}_k = \boldsymbol{U}_{s',t'}$, with $k' \in \mathcal{N}_k^\times$, as the projection of the pixel $\boldsymbol{U}_{s,t}(x, y)$ on these views lies neither along the row x, nor along the 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 $\boldsymbol{U}_{k'} = \boldsymbol{U}_{s',t'}$, with $k' \in \mathcal{N}_k^{\times}$, 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 $\boldsymbol{W} = \boldsymbol{W}_B$ in Eq. (9). We preserve an edge between two vertexes $\boldsymbol{u}(i)$ and $\boldsymbol{u}(j)$ if and only if the entries $\boldsymbol{W}_B(i, j)$ and $\boldsymbol{W}_B(j, i)$ are both non zero. If this is the case, then $\boldsymbol{W}_B(i, j) = \boldsymbol{W}_B(j, i)$ necessarily holds true, and the weights are maintained. This procedure mimics the well-known *left-right disparity check* of stereo vision [27].

We finally note that, as proposed in [7], the constructed graph can be used to build an alternative warping matrix to the one in Section V-A. 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 $(k-1)N^2 + 1, \ldots, kN^2$ and the columns $(k'-1)N^2, \ldots, k'N^2$ 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 $\boldsymbol{F}_{k'}^{k} = \widetilde{\boldsymbol{W}}_S$ with $\widetilde{\boldsymbol{W}}_S$ the normalized sub-matrix. This alternative approach to the warping matrix construction does not take the light field structure explicitly into account, but it represents a valid alternative when computational resources are limited.

C. Optimization Algorithm

We now have all the ingredients to solve the optimization problem in Eq. (3). We observe that it corresponds to a quadratic problem. We rewrite the first term, in Eq. (5), as follows:

$$\mathcal{F}_1(\boldsymbol{u}) = \|\boldsymbol{A}\,\boldsymbol{u} - \boldsymbol{v}\|_2^2 \tag{12}$$

with $\boldsymbol{A} = \boldsymbol{I} \otimes \boldsymbol{SB}$, $\boldsymbol{I} \in \mathbb{R}^{M^2 \times M^2}$ the identity matrix, and $\otimes$ the Kronecker product. For the second term, in Eq. (7), we introduce the following matrices:

- $\boldsymbol{H}_k = \operatorname{diag}(\boldsymbol{H}_k^1, \boldsymbol{H}_k^2, \ldots, \boldsymbol{H}_k^{M^2})$,
- $\boldsymbol{F}_k = \boldsymbol{e}_k^\top \otimes \left[(\boldsymbol{F}_k^1)^\top (\boldsymbol{F}_k^2)^\top \ldots (\boldsymbol{F}_k^{M^2})^\top\right]^\top$,

where diag denotes a block diagonal matrix, and $\boldsymbol{e}_k \in \mathbb{R}^{M^2}$ denotes the k-th vector of the canonical basis, with $\boldsymbol{e}_k(k) = 1$ and zero elsewhere. The matrices $\boldsymbol{H}_k^{k'}$ and $\boldsymbol{F}_k^{k'}$, originally defined only for $k' \in \mathcal{N}_k^+$, have been extended to the whole light field by assuming them to be zero for $k' \notin \mathcal{N}_k^+$. Finally, it is possible to remove the inner sum in Eq. (7):

$$\mathcal{F}_2(\boldsymbol{u}) = \sum_k \|\boldsymbol{H}_k \boldsymbol{A} \boldsymbol{F}_k\,\boldsymbol{u} - \boldsymbol{H}_k \boldsymbol{v}\|_2^2. \tag{13}$$

Replacing Eq. (12) and Eq. (13) in Eq. (3) permits to rewrite the objective function $\mathcal{F}(\boldsymbol{u})$ in a quadratic form:

$$\boldsymbol{u}^* \in \operatorname{argmin}_{\boldsymbol{u}} \underbrace{\frac{1}{2}\boldsymbol{u}^\top \boldsymbol{P}\,\boldsymbol{u} + \boldsymbol{q}^\top \boldsymbol{u} + r}_{\mathcal{F}(\boldsymbol{u})} \tag{14}$$

with

$$\boldsymbol{P} = 2\left(\boldsymbol{A}^\top \boldsymbol{A} + \lambda_2 \sum_k (\boldsymbol{H}_k \boldsymbol{A} \boldsymbol{F}_k)^\top (\boldsymbol{H}_k \boldsymbol{A} \boldsymbol{F}_k) + \lambda_3 \boldsymbol{L}\right),$$

$$\boldsymbol{q} = -2\left(\boldsymbol{A}^\top + \lambda_2 \sum_k \left((\boldsymbol{H}_k \boldsymbol{A} \boldsymbol{F}_k)^\top \boldsymbol{H}_k\right)\right)\boldsymbol{v},$$

$$r = \boldsymbol{v}^\top \left(\boldsymbol{I} + \lambda_2 \sum_k \left(\boldsymbol{H}_k^\top \boldsymbol{H}_k\right)\right)\boldsymbol{v}.$$

We observe that, in general, the matrix $\boldsymbol{P}$ is positive semi-definite, therefore it is not possible to solve Eq. (14) just by employing the *Conjugate Gradient (CG)* method on the linear system $\nabla\mathcal{F}(\boldsymbol{u}) = \boldsymbol{P}\boldsymbol{u} - \boldsymbol{q} = 0$. We thus choose to adopt the *Proximal Point Algorithm (PPA)*, which iteratively solves Eq. (14) using the following update rule:

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

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 Light Field Super-Resolution* algorithm is summarized in Algorithm 1. We observe that the construction of the warping matrices and the graph requires the high resolution light field to be available. In order to bypass this causality problem, a fast and rough high resolution estimation of the light field is computed via bilinear interpolation at the bootstrap phase. Then, at each new iteration, the warping matrices and the graph can be re-constructed on the new available light field estimate.

The problem in Eq. (14) could be solved also with a *Gradient Descent (GD)* algorithm, which is characterized by a less computationally demanding update rule. However, in our experiments, PPA leads to a faster convergence than GD not only in terms of the number of iterations but also in terms of computation time. For an analysis of the computational complexity of our algorithm, we refer the reader to our technical report [28].

VI. Experiments

A. Experimental Settings

We now provide extensive experiments to analyze the performance of our algorithm. We compare it to the two

Algorithm 1 Graph-Based Light Field Super-Resolution

```
Input: $\boldsymbol{v} = [\boldsymbol{v}_1, \dots, \boldsymbol{v}_{M^2}]$, $\alpha \in \mathbb{N}$, $\beta > 0$, $iter$.
Output: $\boldsymbol{u} = [\boldsymbol{u}_1, \dots, \boldsymbol{u}_{M^2}]$.
$\boldsymbol{u} \leftarrow$ bilinear interp. of $\boldsymbol{v}_k$ by $\alpha$, $\forall k = 1, \dots, M^2$;
for $i = 1$ to $iter$ do
    build the graph adjacency matrix $\boldsymbol{W}$ on $\boldsymbol{u}$;
    build the matrices $\boldsymbol{F}_k$ on $\boldsymbol{u}$, $\forall k = 1, \dots, M^2$;
    update the matrix $\boldsymbol{P}$ and 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 $\boldsymbol{u}$;
```

super-resolution algorithms that, up to our knowledge, are the only ones developed explicitly for light field data, and that we already introduced in Section II: Wanner and Goldluecke [17], and Mitra and Veeraraghavan [20]. We also compare our algorithm to the CNN-based super-resolution algorithm in [10], which represents the state-of-the-art for single-frame super-resolution. Up to the authors knowledge, a multi-frame super-resolution algorithm based on CNNs has not been presented yet.

We test our algorithm on two public datasets: the *HCI light field dataset* [29] and the *(New) Stanford light field dataset* [24]. The HCI dataset comprises twelve light fields, each one characterized by a 9×9 array of views. Seven light fields have been artificially generated with a 3D creation suite, while the remaining five have been acquired with a traditional SLR camera mounted on a motorized gantry, that permits to move the camera precisely and emulate a camera array with an arbitrary baseline. The HCI dataset is meant to represent the data from a light field camera, where both the baseline distance b between adjacent views and the disparity range are typically very small. In particular, in the HCI dataset the disparity range is within $[-3, 3]$ pixels. Differently, the Stanford dataset contains light fields whose view baseline and disparity range can be much larger. For this reason, the Stanford dataset does not closely resemble the typical data from a light field camera. However, we include the Stanford dataset in our experiments in order to evaluate the robustness of light field super-resolution methods to larger disparity ranges, possibly exceeding the assumed one. The Stanford light fields have all been acquired with a gantry, and they are characterized by a 17×17 array of views. Similarly to [17] and [20], we crop the light fields to a 5×5 array of views, i.e., we choose $M = 5$.

In our experiments, the spatial resolution of each 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. (4) 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 the assumptions of the methods [17] and [20], the matrices $\boldsymbol{B}$ and $\boldsymbol{S}$ implement an $\alpha \times \alpha$ box filter and a decimator, respectively.

We consider two variants of our Graph-Based Light Field Super-Resolution algorithm (GB). The first is *GB-SQ*, which employs the warping matrix construction based on the square constraint (SQ) and is presented in Section V-A. The second is *GB-DR*, which employs the warping matrices extracted directly (DR) from the graph and introduced at the end of Section V-B. In the warping matrix construction in Eq. (10) and in the graph construction in Eq. (11), 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. This range happens to be loose for the HCI dataset, whose disparity range is within $[-3, 3]$. Choosing exactly the correct disparity range for each light field could both improve the reconstruction, by avoiding possible wrong correspondences in the graph and warping matrices, and decrease the computation time. On the other hand, for some light fields in the Stanford dataset, the chosen disparity range may become too small, thus preventing the possibility to capture the correct correspondences. In practice, the disparity range is not always available, hence the value $[-6, 6]$ happens to be a fair choice considering that our super-resolution framework targets data from light field cameras, i.e., data with a typically small disparity range. Finally, without any fine tuning on the considered datasets, we empirically set $\lambda_2 = 0.2$ and $\lambda_3 = 0.0055$ in the objective function in Eq. (3) and perform just two iterations of the full Algorithm 1, as we experimentally found them to be sufficient[1].

We carry out our experiments on the light field super-resolution algorithms in [17] using the original code by the authors, available online within the library *cocolib*. For a fair comparison, we provide the $[-6, 6]$ pixel range at the library input, in order to permit the removal of outliers in the estimated disparity maps. For our experiments on the algorithm in [20] 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 repeated for every α 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. For the comparison with the CNN-based super-resolution algorithm in [10] we employ the original code, available online. We perform the CNN training on the data provided together with the code. We learn a different CNN for every α and perform $8 * 10^8$ back-propagations, as described in [10]. We then super-resolve each light field by applying the CNN on the single views. Finally, as a baseline reconstruction, we consider also the high resolution light field obtained from the bilinear interpolation of the single views.

Our super-resolution method GB, the methods in [20] and [10], and the bilinear interpolation one, super-resolve 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. Instead, the method in [17] reconstructs

[1] The MATLAB code of our algorithm is available at the following link: https://github.com/rossimattia/light-field-super-resolution

TABLE I

HCI DATASET [29] - PSNR MEAN (AND VARIANCE) FOR THE SUPER-RESOLUTION FACTOR $\alpha = 2$

	Bilinear	[17]	[20]	[10]	GB-DR	GB-SQ
buddha	35.22 (0.00)	38.22 (0.11)	**39.12** (0.62)	37.73 (0.03)	38.59 (0.08)	39.00 (0.14)
buddha2	30.97 (0.00)	33.01 (0.11)	33.63 (0.22)	33.67 (0.00)	34.17 (0.01)	**34.41** (0.02)
couple	25.52 (0.00)	26.22 (1.61)	31.83 (2.80)	28.56 (0.00)	32.79 (0.17)	**33.51** (0.25)
cube	26.06 (0.00)	26.40 (1.90)	30.99 (3.02)	28.81 (0.00)	32.60 (0.23)	**33.28** (0.35)
horses	26.37 (0.00)	29.14 (0.63)	**33.13** (0.72)	27.80 (0.00)	30.99 (0.05)	32.62 (0.12)
maria	32.84 (0.00)	35.60 (0.33)	37.03 (0.44)	35.50 (0.00)	37.19 (0.03)	**37.25** (0.02)
medieval	30.07 (0.00)	30.53 (0.67)	33.34 (0.71)	31.23 (0.00)	33.23 (0.03)	**33.45** (0.02)
mona	35.11 (0.00)	37.54 (0.64)	38.32 (1.14)	39.07 (0.00)	39.30 (0.04)	**39.37** (0.05)
papillon	36.19 (0.00)	39.91 (0.15)	40.59 (0.89)	39.88 (0.00)	**40.94** (0.06)	40.70 (0.04)
pyramide	26.49 (0.00)	26.73 (1.42)	33.35 (4.06)	29.69 (0.00)	34.63 (0.34)	**35.41** (0.67)
statue	26.32 (0.00)	26.15 (2.15)	32.95 (4.67)	29.65 (0.00)	34.81 (0.38)	**35.61** (0.73)
stillLife	25.28 (0.00)	25.58 (1.41)	28.84 (0.82)	27.27 (0.00)	30.80 (0.07)	**30.98** (0.05)

TABLE II

STANFORD DATASET [24] - PSNR MEAN (AND VARIANCE) FOR THE SUPER-RESOLUTION FACTOR $\alpha = 2$

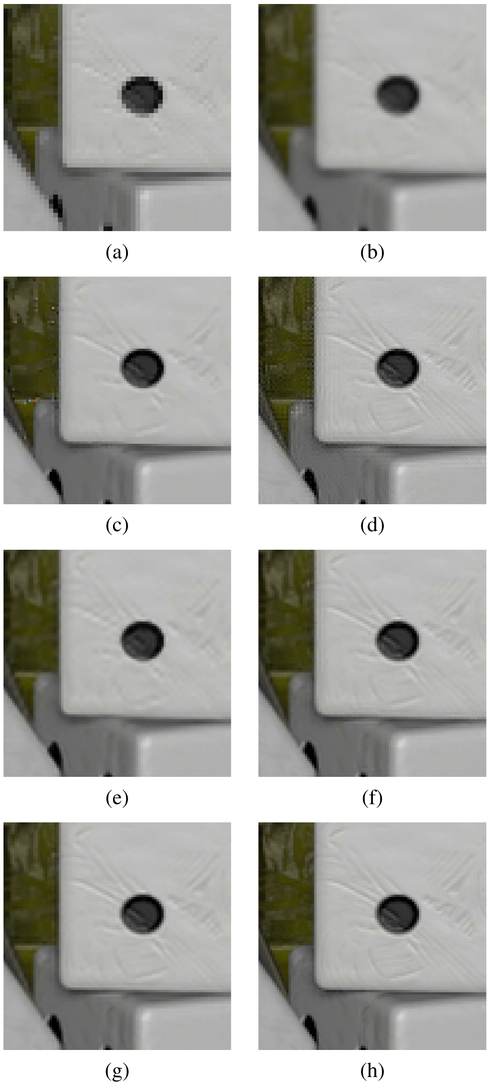
	Bilinear	[17]	[20]	[10]	GB-DR	GB-SQ
amethyst	35.59 (0.01)	24.18 (0.20)	36.08 (4.12)	38.81 (0.01)	**40.30** (0.11)	39.19 (0.25)
beans	47.92 (0.01)	23.28 (0.53)	36.02 (7.38)	**52.01** (0.01)	50.20 (0.16)	48.41 (1.18)
bracelet	33.02 (0.00)	18.98 (0.22)	19.91 (3.86)	38.05 (0.00)	**39.10** (0.43)	28.27 (2.84)
bulldozer	34.94 (0.01)	22.82 (0.09)	32.05 (3.73)	**39.76** (0.03)	37.27 (0.33)	35.96 (0.43)
bunny	42.44 (0.01)	26.22 (1.15)	40.66 (6.69)	**47.16** (0.01)	46.96 (0.06)	47.01 (0.06)
cards	29.50 (0.00)	19.38 (0.22)	28.46 (5.68)	33.77 (0.00)	**36.54** (0.20)	36.52 (0.20)
chess	36.36 (0.00)	21.77 (0.27)	34.74 (5.85)	40.75 (0.00)	**42.04** (0.08)	41.86 (0.07)
eucalyptus	34.09 (0.00)	25.04 (0.28)	34.90 (3.50)	36.69 (0.00)	39.07 (0.12)	**39.09** (0.08)
knights	34.31 (0.04)	21.14 (0.24)	29.33 (4.77)	38.37 (0.06)	**39.68** (0.27)	37.23 (0.40)
treasure	30.83 (0.00)	22.81 (0.15)	30.52 (2.93)	34.16 (0.00)	**37.68** (0.26)	37.51 (0.15)
truck	36.26 (0.04)	25.77 (0.08)	37.52 (4.60)	39.11 (0.09)	41.09 (0.14)	**41.57** (0.15)

directly a full color light field. Since most of the considered methods super-resolve only the luminance of the low resolution light field, we compute the reconstruction error only on the luminance channel. For the method in [17], whose light field luminance is not directly available, we compute it from the reconstructed full color light field.

B. Light Field Reconstruction Results

The numerical results from our super-resolution experiments on the HCI and Stanford datasets are reported in Tables I and II for a super-resolution factor $\alpha = 2$. Due to space constraints, for the results of our experiments for $\alpha = 3$, we refer the reader to our technical report [28]. For each reconstructed light field we compute the *Peak Signal-to-Noise Ratio (PSNR [dB])* at each view and report the average and variance of the computed PSNRs in the tables. Finally, for a fair comparison with the method in [20], which suffers from border effects, a 15-pixel border is removed from all the reconstructed views before the PSNR computation.

In the HCI dataset and for a super-resolution factor $\alpha = 2$, GB provides the highest average PSNR on ten out of twelve light fields. In particular, nine out of ten of the highest average PSNRs are due to GB-SQ. The highest average PSNR in the two remaining light fields `buddha` and `horses` is achieved by [20], 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 two light fields. This is shown in Figures 4 and 5, where two details from the bottom right-most views of the light fields `buddha` and `horses`, respectively, are given for each method. The reconstructions provided by [20], in Figures 4d and 5d, exhibit 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. Figures 4c and 5c show that also the reconstructions provided by the method in [17] exhibit strong artifacts along edges, although the disparity is estimated at each pixel in this case. This is due to the presence of errors in the estimated disparity at object boundaries. These errors are caused both by the poor performance of the structure tensor operator in the presence of occlusions and, more in general, by the challenges posed by disparity estimation at low resolution. We also observe that the TV term in [17] tends to over-smooth the fine details, as evident in the dice of Figure 4c. The method in [10], meant for single-frame super-resolution and therefore agnostic of the light field structure, provides PSNR values that are significantly lower than those provided by GB and [20], which instead take the light field structure into account. The quality of the views reconstructed by the method in [10] depends exclusively on the training data, as it does not employ the complementary information available at the other views. This is clear in Figure 4e, where [10] does not manage to recover the fine structure around the black spot in the dice, which remains pixelated as in the original low resolution view. Similarly, the method in [10] does not manage to reconstruct

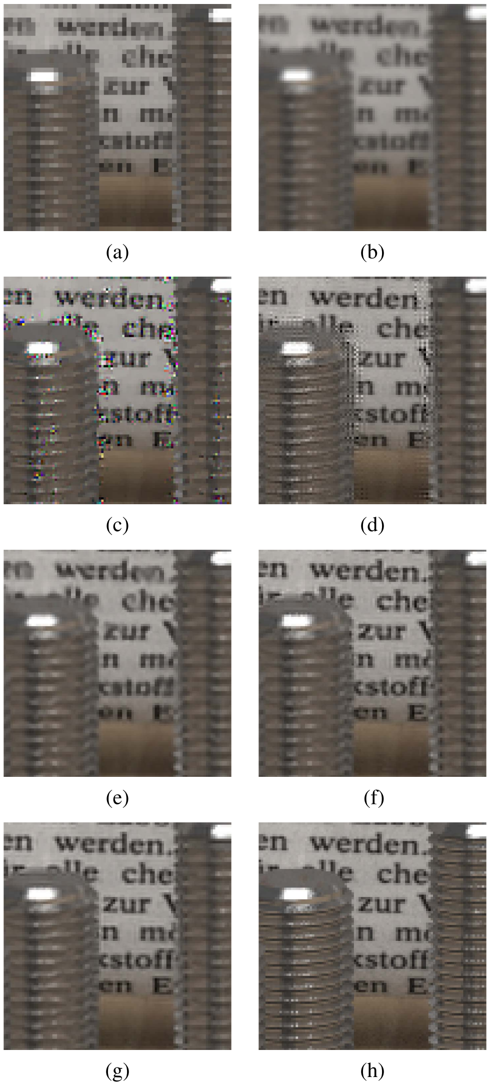

(a) (b)

(c) (d)

(e) (f)

(g) (h)

Fig. 4. Detail from the bottom right-most view of the light field buddha, in the HCI dataset [29]. The low resolution light field in (a) is super-resolved by a factor $\alpha = 2$ with bilinear interpolation in (b), the method [17] in (c), the method [20] in (d), the method [10] in (e), GB-DR in (f) and GB-SQ in (g). The original high resolution light field is provided in (h).


(a) (b)

(c) (d)

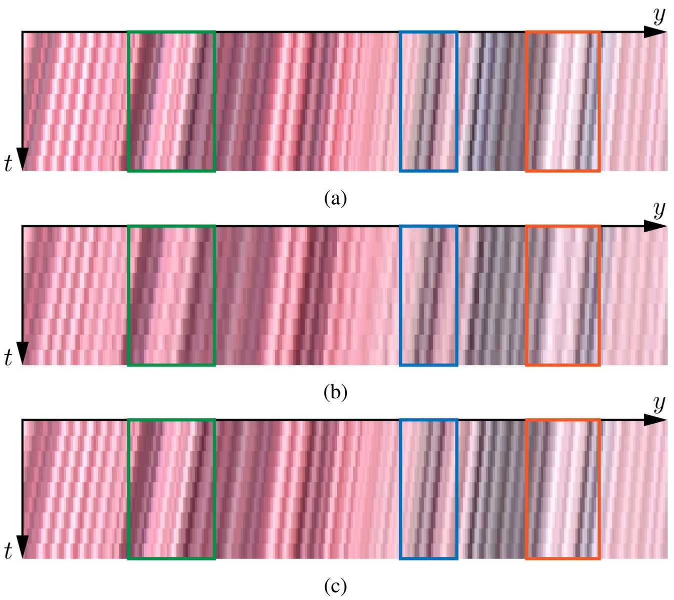
(e) (f)

(g) (h)

Fig. 5. Detail from the bottom right-most view of the light field horses, in the HCI dataset [29]. The low resolution light field in (a) is super-resolved by a factor $\alpha = 2$ with bilinear interpolation in (b), the method [17] in (c), the method [20] in (d), the method [10] in (e), GB-DR in (f) and GB-SQ in (g). The original high resolution light field is provided in (h).

effectively the letters in Figure 5e, which remain blurred and in some cases cannot be discerned. Moreover, since the method in [10] does not consider the light field geometric structure, it does not necessarily preserve it. An example is provided in Figure 6, where an EPI is extracted from the reconstructions of the stillLife light field computed by GB-SQ and the method in [10]. While GB-SQ preserves the line pattern, the method in [10] does not. The bilinear interpolation method provides the lowest PSNR values and the poor quality of its reconstruction is confirmed by the Figures 4b and 5b, which appear significantly blurred. Finally, the numerical results suggest that our GB-SQ methods is more effective in capturing the correct correspondences between adjacent views in the light field. A visual example is provided in Figure 7, where the letters in the view reconstructed by GB-SQ are sharper than those in the view reconstructed by GB-DR.

In the Stanford dataset and for the same super-resolution factor $\alpha = 2$, GB provides the highest average PSNRs on eight light fields out of eleven, the method in [10] provides the highest average PSNRs in the three remaining light fields, while the algorithms in [17] and [20] perform even worse than bilinear interpolation in most of the cases. The very poor performance of [17] and [20], and the generally higher PSNR provided by GB-DR compared to GB-SQ, are mainly consequences of the Stanford dataset disparity range, which exceeds the $[-6, 6]$ pixel range assumed in our tests. In particular, objects with a disparity outside the assumed disparity range are not properly reconstructed in general. An example is provided in Figure 8, where two details from the bottom right-most view of the light field bulldozer are shown. The detail at the bottom captures the bulldozer blade, placed very close to the camera and characterized by large disparity values outside the assumed disparity range, while the detail at the top captures a cylinder behind the blade and characterized by disparity values within the assumed range. As expected, in Figures 8f and 8g, GB manages to correctly reconstruct the cylinder, while it introduces some artifacts on the blade. However, it can be observed that GB-DR introduces milder artifacts

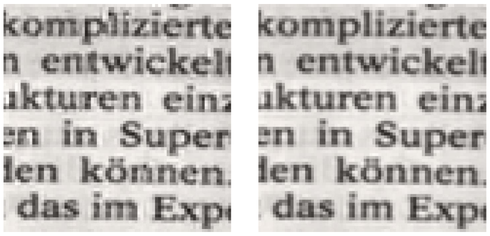


(a)

(b)

(c)

Fig. 6. Epipolar Plane Image (EPI) from the light field `stillLife`, in the HCI dataset [29]. The 9×9 light field is super-resolved by a factor $\alpha = 2$ using the single-frame super-resolution method in [10] and GB-SQ. The same EPI is extracted from the original high resolution light field, in (a), and from the reconstructions provided by [10], in (b), and GB-SQ, in (c). Since the method in [10] super-resolves the views independently, the original line pattern appears compromised, therefore the light field structure is not preserved. On the contrary, GB-SQ preserves the original line pattern, hence the light field structure.



(a) (b)

Fig. 7. Detail from the central view of the super-resolved light field `horses`, in the HCI dataset [29], for the super-resolution factor $\alpha = 2$. The reconstruction provided by GB-SQ, in (b), exhibits sharper letters than the reconstruction by GB-DR, in (a), as the square constraint captures better the light field structure.

than GB-SQ on the blade, as GB-SQ forces the warping matrices to fulfill the square constraint of Section V-A on a wrong disparity range, while GB-DR is more accommodating in the warping matrix construction and therefore more robust to a wrong disparity range assumption. Figure 8h provides the reconstruction computed by GB-SQ when the assumed disparity range is extended to $[-12, 12]$ pixels, and it shows that the artifacts disappear when the correct disparity range is within the assumed one. On the other hand, in Figure 8d the method in [20] fails to reconstruct also the cylinder, as the top of the image exhibits depth discontinuities that do not fit its assumption of constant disparity within each light field patch. The method in [17] fails in both areas as well, and in general on the whole Stanford light field dataset, as the structure tensor operator cannot detect large disparity values [30]. Differently from the light field super-resolution methods, the one in [10] processes each view independently and it does not introduce any visible artifact in Figure 8e, neither in the top nor in the bottom detail. However, the absence of visible artifacts does not guarantee that the light field structure is preserved, as [10] does not take it into account. For the sake of completeness, we observe that not all the light fields in the Stanford dataset meet the Lambertian assumption. Some areas of the captured scenes violate it. This contributes to the low PSNR values exhibited by the methods [17], [20], and GB-SQ on certain light fields (e.g., `bracelet`) in Table II, as in non Lambertian areas the light field structure in Eq. (2) does not hold true. On the other hand, GB-DR is more accommodating in the warping matrix construction, and this makes the method more robust not only to the adoption of incorrect disparity ranges but also to the violation of the Lambertian assumption, as confirmed numerically in Table II.

To conclude, our experiments show that the proposed super-resolution algorithm GB has some remarkable reconstruction properties that make it preferable over its considered competitors. First, its reconstructed light fields exhibit a better visual quality, often confirmed numerically by the PSNR measure. Second, it provides an homogeneous and consistent reconstruction of all the views in the light field. Third, it is more robust than the other considered methods in those scenarios where some objects in the scene exceed the assumed disparity range, as it may be the case in practice.

For additional tests and details, we refer the reader to [28].

VII. 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 coupling an approximate warping matrix construction strategy with a graph regularizer, which enforces the light field geometric structure, avoids to carry out a costly disparity estimation step at sub-pixel precision. In addition, the proposed algorithm reduces to a quadratic problem, that can be solved efficiently with standard convex optimization tools.

The proposed algorithm compares favorably to the state-of-the-art light field super-resolution frameworks, both in terms of visual quality and in terms of PSNR. It provides an homogeneous reconstruction of all the views in the light field, which is a property that is not present in the other light field super-resolution frameworks [17], [20]. Also, although the proposed algorithm is meant mainly for light field camera data, where the disparity range is typically small, it is flexible enough to handle light fields with larger disparity ranges too. We also compared our algorithm to a state-of-the-art single-frame super-resolution method based on CNNs [10], and we showed that taking the light field structure into account allows our algorithm to recover finer details, and most importantly it avoids the reconstruction of a set of geometrically inconsistent high resolution views.

Acknowledgements

The authors would like to thank Prof. Christine Guillemot (INRIA Rennes) for the very valuable discussions in the early stages of this work.

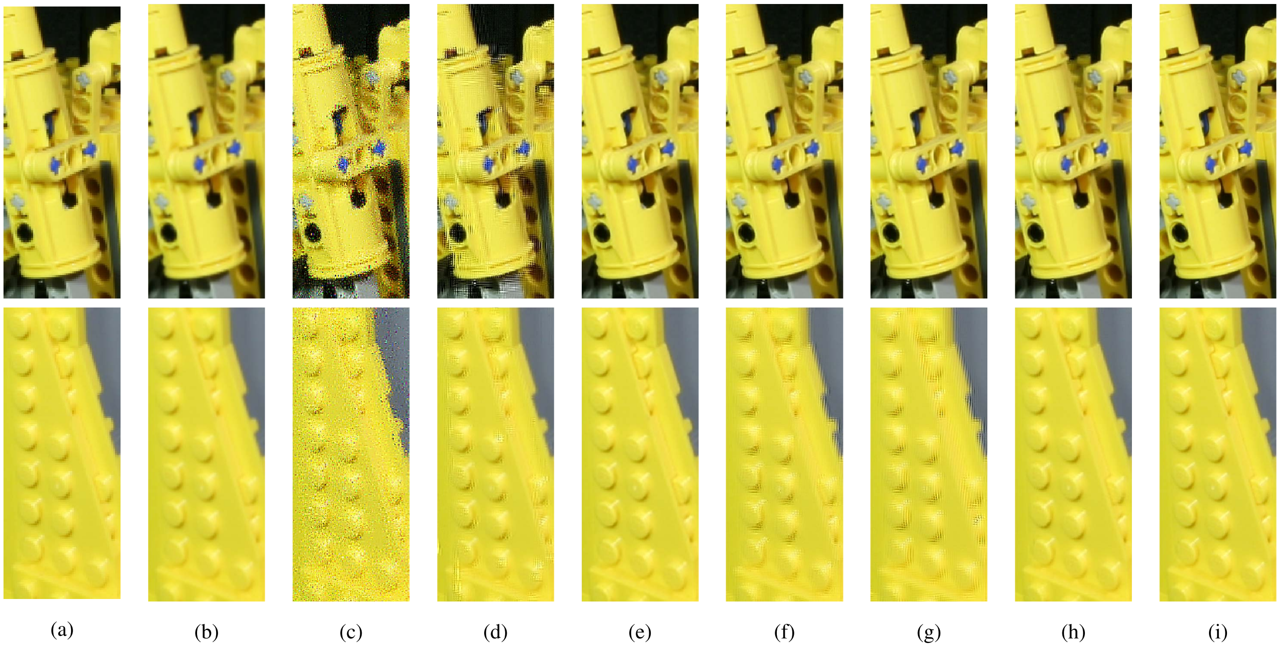


Fig. 8. Details from the bottom right-most view of the light field `bulldozer`, in the Stanford dataset [24]. The low resolution light field in (a) is super-resolved by a factor $\alpha = 2$ with bilinear interpolation in (b), the method [17] in (c), the method [20] in (d), the method [10] in (e), GB-DR in (f) and GB-SQ in (g). The reconstruction of GB-SQ with the extended disparity range $[-12, 12]$ pixels is provided in (h), and the original high resolution light field is in (i).

REFERENCES

[1] M. Levoy and P. Hanrahan, "Light field rendering," in *Proc. 23rd ACM Annu. Conf. Comput. Graph. Interact. Techn.*, 1996, pp. 31–42.

[2] B. Wilburn *et al.*, "High performance imaging using large camera arrays," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 765–776, Jul. 2005.

[3] *Lytro Inc.* Accessed: Aug. 10, 2017. [Online]. Available: https://www.lytro.com/

[4] *Raytrix GmbH*. Accessed: Aug. 10, 2017. [Online]. Available: https://www.raytrix.de/

[5] R. Ng, M. Levoy, M. Brédif, G. Duval, M. Horowitz, and P. Hanrahan, "Light field photography with a hand-held plenoptic camera," Dept. Comput. Sci., Stanford Univ., Stanford, CA, USA, Tech. Rep. CSTR 2005-02, 2005, pp. 1–11.

[6] C. Perwass and L. Wietzke, "Single lens 3D-camera with extended depth-of-field," in *Proc. SPIE*, vol. 8291, p. 829108, Feb. 2012.

[7] M. Rossi and P. Frossard, "Graph-based light field super-resolution," in *Proc. IEEE 19th Int. Workshop Multimedia Signal Process. (MMSP)*, Oct. 2017, pp. 1–6.

[8] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.

[9] X. Gao, K. Zhang, D. Tao, and X. Li, "Joint learning for single-image super-resolution via a coupled constraint," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 469–480, Feb. 2012.

[10] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 184–199.

[11] D. Glasner, S. Bagon, and M. Irani, "Super-resolution from a single image," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep./Oct. 2009, pp. 349–356.

[12] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel, "Single-image super-resolution via linear mapping of interpolated self-examples," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5334–5347, Dec. 2014.

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

[14] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1327–1344, Oct. 2004.

[15] D. Mitzel, T. Pock, T. Schoenemann, and D. Cremers, "Video super resolution using duality based TV-L^1 optical flow," in *Proc. Joint Pattern Recognit. Symp.*, 2009, pp. 432–441.

[16] M. Unger, T. Pock, M. Werlberger, and H. Bischof, "A convex approach for variational super-resolution," in *Proc. Joint Pattern Recognit. Symp.*, 2010, pp. 313–322.

[17] S. Wanner and B. Goldluecke, "Spatial and angular variational super-resolution of 4D light fields," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 608–621.

[18] S. Heber and T. Pock, "Shape from light field meets robust PCA," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 751–767.

[19] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *J. ACM*, vol. 58, no. 3, pp. 1–37, May 2011.

[20] K. Mitra and A. Veeraraghavan, "Light field denoising, light field superresolution and stereo camera based refocussing using a GMM light field patch prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2012, pp. 22–28.

[21] 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 Process. Lett.*, vol. 24, no. 6, pp. 848–852, Jun. 2017.

[22] T. E. Bishop and P. Favaro, "The light field camera: Extended depth of field, aliasing, and superresolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 972–986, May 2012.

[23] R. C. Bolles, H. H. Baker, and D. H. Marimont, "Epipolar-plane image analysis: An approach to determining structure from motion," *Int. J. Comput. Vis.*, vol. 1, no. 1, pp. 7–55, 1987.

[24] *The (New) Stanford Light Field Archive*. Accessed: Aug. 10, 2017. [Online]. Available: http://lightfield.stanford.edu/

[25] 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 Process. Mag.*, vol. 30, no. 3, pp. 83–98, May 2013.

[26] A. Kheradmand and P. Milanfar, "A general framework for regularized, similarity-based image restoration," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5136–5151, Dec. 2014.

[27] P. Fua, "A parallel stereo algorithm that produces dense depth maps and preserves image features," *Mach. Vis. Appl.*, vol. 6, no. 1, pp. 35–49, Dec. 1993.

[28] M. Rossi and P. Frossard, "Light field super-resolution via graph-based regularization," *CoRR*, 2017. [Online]. Available: http://arxiv.org/abs/1701.02141

[29] S. Wanner, S. Meister, and B. Goldluecke, "Datasets and benchmarks for densely sampled 4D light fields," in *Proc. VMV*, 2013, pp. 225–226.

[30] M. Diebold and B. Goldluecke, "Epipolar plane image refocusing for improved depth estimation and occlusion handling," in *Proc. VMV*, 2013, pp. 1–8.

Mattia Rossi, photograph and biography not available at the time of publication.

Pascal Frossard, photograph and biography not available at the time of publication.