Divide and Conquer for Full-Resolution Light Field Deblurring

M. R. Mahesh Mohan, A. N. Rajagopalan
Indian Institute of Technology Madras
{ee14d023,raju}@ee.iitm.ac.in

Abstract

The increasing popularity of computational light field (LF) cameras has necessitated the need for tackling motion blur which is a ubiquitous phenomenon in hand-held photography. The state-of-the-art method for blind deblurring of LFs of general 3D scenes is limited to handling only downsampled LF, both in spatial and angular resolution. This is due to the computational overhead involved in processing data-hungry full-resolution 4D LF altogether. Moreover, the method warrants high-end GPUs for optimization and is ineffective for wide-angle settings and irregular camera motion. In this paper, we introduce a new blind motion deblurring strategy for LFs which alleviates these limitations significantly. Our model achieves this by isolating 4D LF motion blur across the 2D subaperture images, thus paving the way for independent deblurring of these subaperture images. Furthermore, our model accommodates common camera motion parameterization across the subaperture images. Consequently, blind deblurring of any single subaperture image elegantly paves the way for cost-effective non-blind deblurring of the other subaperture images. Our approach is CPU-efficient computationally and can effectively deblur full-resolution LFs.

1. Introduction

Handheld light field cameras (LFCs) are being used in a variety of tasks including wide-angle and hyperspectral imaging, shape recovery, segmentation, *etc*. [4, 25, 14, 22]. Refer to [24] for a recent survey on light field (LF) related works. The increase in popularity of LFCs can be attributed to their attractive features over conventional cameras (CCs), including post-capture refocusing, f-stopping, depth sensing [22, 1, 16], *etc*. LFCs achieve this by capturing multiple (subaperture) images instead of a single CC image by segregating the light reaching the CC-sensor into multiple angular components; and synthesize these images post-capture to form an image of desired CC setting [16, 1]. However, there is a downside too. It is well-known that motion blur is a common artifact in hand-held photography with CCs

[17, 26, 12]. This nuisance effect become exacerbated in LFCs. This is because the light-segregation principle in LFCs reduces the amount of photons that make up individual subaperture images, thereby necessitating higher exposure times relative to CC (under the same setting). This escalates the risk of motion blur in LFC. Moreover, a 4D LF comprising of 2D spatial and 2D angular resolutions can be interpreted as several CC images stacked together. Thus the numerical optimization involved in LF deblurring must deal with *very* large-sized data as compared to that of CC. This poses additional computational challenges [24, 20].

In this work, we address the problem of LF blind motion deblurring (LF-BMD). The problem of BMD deals with estimating both the clean image and underlying camera motion from a *single* motion blurred observation. BMD in CCs is a well-studied topic replete with efficient methodolgies. State-of-the-art CC-BMD methods [17, 26, 21] are based on the motion density function (MDF) [5] which allows both narrow- and wide-angle systems as well as nonparametric camera motion, have a homography-based filter flow framework for computational efficiency [8], and employ a scale-space approach to accommodate large blurs. Kohler *et al*. [11] have shown (for CCs) that general camera motion comprising of 3D translations and 3D rotations can be well approximated by full rotations, or inplane translations and inplane rotation. Inplane rotation common to both the approximations are necessary to capture wide angle settings [15, 21]. Whyte *et al*. [23], Xu *et al*. [26], and Pan *et al*. [17] follow full-rotations approximation and employ EFF to yield high quality results in CC-BMD.

In contrast, LF-BMD is an emerging research area and there exists very few works. Jin *et al*. [10] proposed the first LF-BMD approach, but restrict the scene to be frontoparallel and bilayer, and limit the camera motion to *only* inplane translations. A recent LF-BMD work by Srinivasan *et al*. [20] eliminates the planar scene assumption and even includes full 3D translations. However, the ego-motion is constrained to be parametric. This *reduces* its effectiveness under irregular ego-motions, which is common when imaging from moving vehicles, robotic platforms, *etc*. Moreover, since the translational pose cannot model inplane ro-
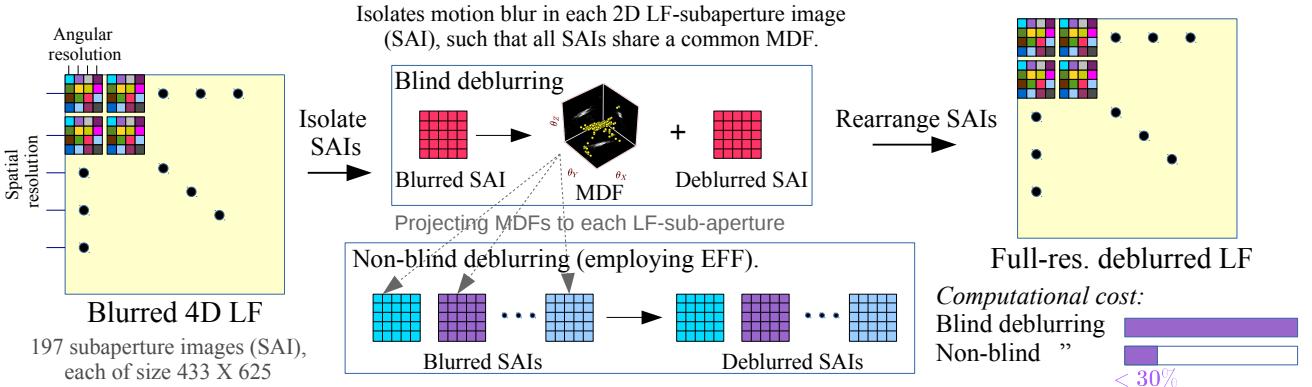
Figure 1. Outline of our proposed method. Our LF-BMD enables decomposing 4D LF deblurring problem into a set of *independent* 2D deblurring sub-problems, in which a blind deblurring of a *single* subaperture-image enables *low-cost* non-blind deblurring of individual subaperture images in parallel. Since all our sub-problems are 2D (akin to CC-case) and thus cost-effective (as it allows efficient filter flow or EFF [8] and is CPU-sufficient), our method is able to handle full-resolution LFs, with significantly less computational cost.

tation, both [10] and [20] are *ineffective* for wide angle systems. Dansereau *et al*. [2] introduce a hardware-assisted deblurring approach using a robotic arm to mount the camera and estimate camera motion (hence non-blind). To the best of our knowledge, [20] is the only work which deals with blind motion deblurring for general 3D scenes. Here, LFC is modeled as an array of pinhole cameras by discarding the effect of LFC-lens, and the motion blurred 4D LF is treated as a composition of shifted and sheared versions of a clean 4D LF. Deblurring with this model proceeds by optimizing for a clean 4D LF at 'one go', using a 4D prior [20].
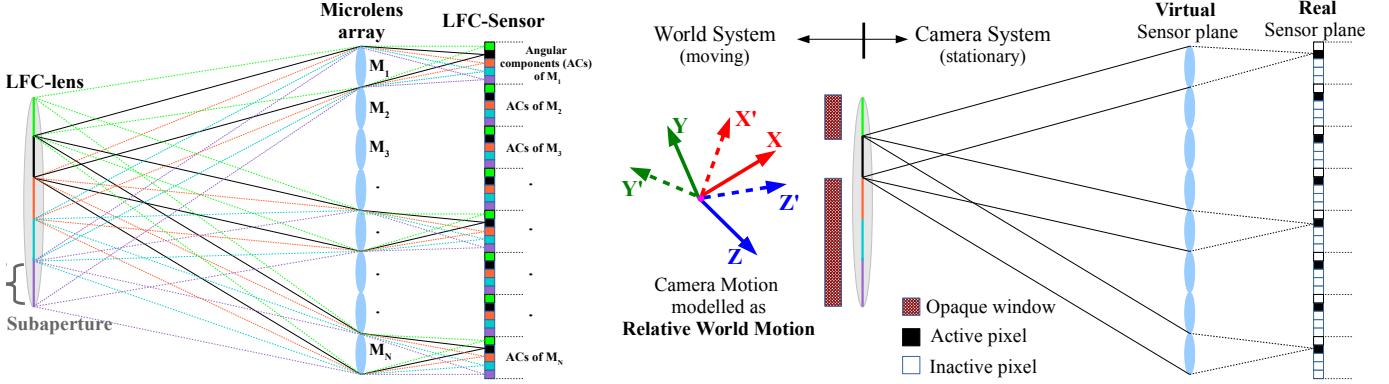
However, [20] has some major drawbacks. First, optimization of 4D LF *in toto* brings up new challenges. The computational requirement involved in this optimization restricts [20] to handle *only* downsampled LFs – both in spatial and angular resolutions (*e.g.*, a Lytro Illum LF file decoded using [3] has 197 subaperture images of size 433×625, whereas [20] requires downsampling it to 64 images of size 200×200 for computational feasibility). As LF postcapture rendering involves composition of multiple subaperture images (or angular components), angular downsampling can adversely affect rendering performance. On the other hand, spatial downsampling restrics LF-rendering software to produce *only* low-resolution RGB images. Furthermore, optimizing high-dimensional data elevates the computational complexity and the method warrants GPU-based processing. Also, such a high-dimensional optimization can distort the interrelations among subaperture images due to convergence issues, which is an important factor for consistent post-capture rendering of LFs [16].

In this paper, we introduce an MDF-based LF motion blur model which allows for decomposition of LF-BMD into low-dimensional subproblems. This admits an efficient filter flow framework [8] to remove the computational bot-

tlenecks and several other limitations of the state-of-the-art [20]. Specifically, our model *isolates* blur formation in *individual* subaperture images (unlike [20]), and imparts a *dependency* among *all* subaperture images through a *common* MDF. Our formulation performs LF-BMD in two efficient steps, as illustrated in Fig. 1. First, we estimate the common MDF from the center-subaperture image using BMD (akin to CC-BMD). Second, by invoking the blur-isolation and commonality of MDF properties inherent to LFCs, we perform *independent* (or parallelizable) non-blind deblurring of individual subaperture images using the estimated MDF while simultaneously accounting for the lens-effect and parallax arising from separation of subapertures from the lens-center. Since each of these subproblems is low-dimensional, our method overcomes the drawbacks associated with the high-dimensional optimization, and thus can deblur LFs at full-resolution [20]. In addition, unlike [20], our LF-MDF model captures the effect of camera lens, can cater to both wide- and narrow-angle camera settings, and can handle irregular camera motions.

Our main contributions are summarized below.

- By harnessing the physics behind LF, we decompose 4D LF-BMD to 2D subproblems, which enables the first ever attempt of full-resolution LF-BMD.

- Our work bridges the gap between the well-studied CC-BMD and emerging LFC-BMD, and facilitates mapping of analogous techniques (such as MDF formulation, efficient filter flow framework, and scale-space strategy) developed for the former to the later.

- Our work dispenses with some important limitations impeding the state-of-the-art [20], such as high computational cost, GPU requirement, and ineffectiveness in handling wide-angle systems & irregular ego-motions.

(a) Formation of 4D light field   (b) Formation of motion blur in a subaperture image

Figure 2. LF motion blur model: (a) Formation of 4D LF. (b) Interpreting camera motion as relative world motion, each motion blurred 2D subaperture image is obtained as a combination of the projections of moving world (parametrized by an MDF) through the *respective* subaperture onto a virtual sensor or microlens array. Also, all subapertures experience the *same* world motion (or share a *common* MDF).

2. MDF for Light Field

In this section, we discuss the working of LFC and limitations of the LF motion blur model of [20]. We then proceed to conceptualize (akin to conventional cameras) an MDF based interpretation for motion blur in LFs, that seeks to mitigate the drawbacks of [20].

A CC with a full-aperture setting spatially resolves light onto a 2D sensor array. An LFC further segregates the light in each CC sensor-element in accordance with the angular bin of the lens-aperture (or subaperture) through which the light arrives. This is achieved by means of a microlens placed in the position of each CC sensor element, which resolves each spatial component into multiple angular components, as shown in Fig. 2(a). The 2D spatially as well as 2D angularly resolved light is stored in a high-resolution LF-sensor (behind the microlens array) to form a 4D LF.

The LF motion blur model of [20] discards the effect of LF-lens and approximates LFC as an array of pinhole cameras positioned at the subapertures. The camera motion is interpreted as an associated movement of these pinhole cameras. In effect, a motion blurred LF is modeled as a composition of shifted and sheared versions of the clean 4D LF. Using flatland analysis (*i.e.*, considering a single angular dimension u and a single spatial dimension x), a motion blurred 4D LF (L_b) can be represented as

$$L_b(x,u) = \int_t L_c(x, u + p_x(t) - xp_z(t))dt, \quad (1)$$

where L_c is the clean 4D LF and $\{p_x(t), p_z(t)\}$ is the camera motion path during the exposure time. As only the angular term u is varying in Eq. (1), a motion blurred subaperture image can be interpreted as a composition of multiple clean subaperture images (where the amount depends on camera motion). In the above equation, considering a *single*

blurred 2D subaperture image as the observation amounts to solving for *multiple* clean 2D images as unknowns – a heavily ill-posed problem. Instead, [20] considers the entire blurred 4D LF as observation and solves for a clean 4D LF as unknown. This reduces the ill-posedness but incurs high-dimensional optimization issues, as discussed in Sec. 1.

We now briefly digress to discuss the MDF model employed in the state-of-the-art CC-BMD [17, 26, 23]. In this model, the CC is approximated as a pinhole at lens-center, and camera motion is interpreted as stationary camera but with relative world motion (see Fig. 3). Considering full-rotations approximation and a single pose change, the relative change in world coordinate is given as

$$\mathbf{X}' = R\mathbf{X}, \quad (2)$$

where R is rotation matrix, and $\mathbf{X} = [X, Y, Z]^T$ and $\mathbf{X}' = [X', Y', Z']^T$ are the 3D world coordinates with respect to initial and final camera positions, respectively. There exists a homography mapping h that relates the corresponding displacement in image coordinates, *i.e.*,

$$\mathbf{x}' = h(K, R, \mathbf{x}), \quad (3)$$

where K is the camera matrix, and $\mathbf{x}$ and $\mathbf{x}'$ are 2D image coordinates corresponding to the initial and final camera positions, respectively. Resultantly, the final image I' can be related to the initial image I as

$$I' = I(K, R), \quad (4)$$

where $I(K, R)$ performs warping of image I in accordance with Eq. (3). Thus a general motion blurred image B (*i.e.*, camera that experiences multiple pose-changes over its exposure time) can be expressed as

$$B = \sum_\lambda \omega_\lambda \cdot I(K, R_\lambda), \quad (5)$$

where R_λ spans the plausible camera pose-space and ω_{λ_0} is the motion density function (MDF) which gives the fraction of exposure time the camera stayed in the pose R_{λ_0} (Fig. 3-right). Note that the MDF completely characterizes the camera-shake. State-of-the-art CC-BMD methods proceed by alternating minimization of ω_λ and I in a scale-space manner to accommodate large blurs (*i.e.*, MDF estimation starts with a downsampled blurred image where the motion blur is less prominent, and proceeds to finer scale MDF-estimation using the previous estimate).

Along similar lines, it is possible to conceptualize camera shake in LFC to be uniquely characterised by an MDF, but having one-to-many mapping from world to LF-sensor due to LF-capture mechanism. To develop an analogous MDF framework for LF-BMD, we introduce an alternative interpretation of LF motion blur formation. The design of LFC-lens system is such that light from different subapertures do *not* interfere with each other [16]. As shown in Fig. 2(b)-right, a clean subaperture image is equivalent to a 2D image formed in a CC with full-aperture setting and with *only* the respective subaperture open, the sensor being placed at the position of the microlens-array.

Akin to CC, we too interpret motion to be stationary camera and a world moving (see Fig. 2(b)). Each motion blurred subaperture image is thus equivalent to an image formed in a full-aperture CC with *only* the respective subaperture open. Interestingly, note that all subapertures are subjected to same world-motion (*i.e.*, parameterized by a *single* MDF). Intuitively, *each motion blurred subaperture image is formed by a linear combination of the projections of the moving world through the respective subaperture onto a virtual sensor formed by the microlens array*. Thus, a blurred subaperture image B_i can be alternatively expressed as

$$B_i = \sum_\lambda \omega_\lambda \cdot I_i(K_i, R_\lambda, \gamma_i) \qquad 1 \le i \le N, \quad (6)$$

where I_i is the clean version of the ith subaperture image. Note that the camera matix K_i and parameter set γ_i vary with individual subapertures so as to capture one-to-many world-mapping. Their exact forms will be discussed in a later section. Our model isolates the blur in individual subaperture images, *i.e.*, a *single* 2D blurred subaperture image as observation amounts to solving for the *corresponding* clean subaperture image as unknown (unlike [20]). Also, a single MDF ω_λ is shared by all subaperture images.

The MDF-based LF motion blur model in Eq. (6) provides three distinct advantages. First, because it *isolates* motion blur in individual subaperture images, we can estimate the common MDF from a *single* subaperture image – a low-dimensional optimization (akin to CC-BMD). Second, since all subaperture images share a *common* MDF, we can use the estimated MDF to perform non-blind de-
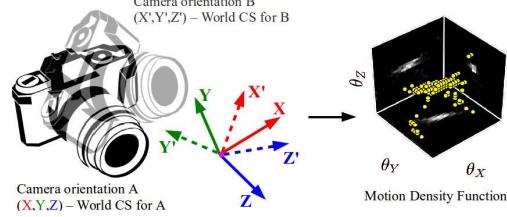
Figure 3. Motion Density Function (MDF): Change in camera orientation from A to B is equivalent to the relative change in world coordinate system (CS) from $\mathbf{X}$ to $\mathbf{X}'$. Thus, MDF, which gives the fraction of time the world CS stayed in different poses during the exposure time, *completely* characterizes the camera motion.

blurring of all the other subaperture images. As non-blind deblurring of individual subapertures can be done *independently*, this step is amenable to parallelization. Note that non-blind deblurring methods (which optimize for a clean 2D image *only* once) are quite cost-effective as compared to blind methods (which cumbersomely optimize for MDF and clean 2D image alternately over iterations). These factors drastically reduce the computational cost for LF-BMD and allow full-resolution LF-BMD. Third, since MDF captures both regular and irregular ego-motion, our method can handle unconstrained ego-motion; and consideration of full rotational camera motion accommodates both narrow- and wide-angle systems, unlike [20, 10].
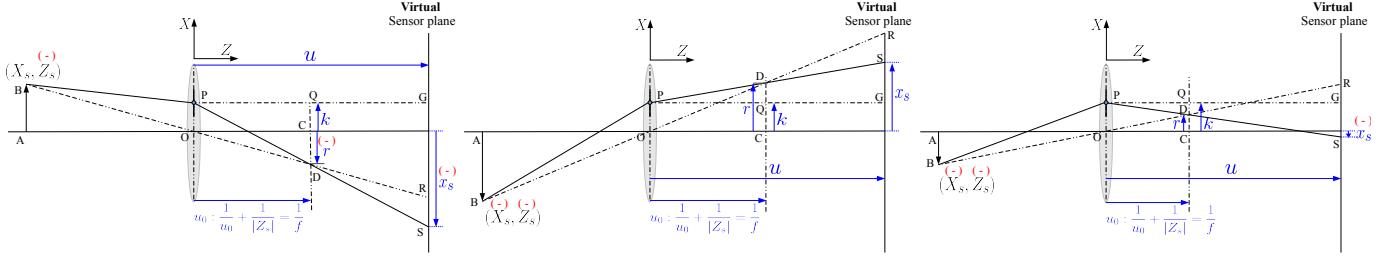
3. MDF-based LF Motion Blur Model

In this section, we formulate our MDF-based LF motion blur model. This requires world-to-sensor mapping in each subaperture, so as to derive individual LF homographies.

3.1. World-to-Sensor Mapping in a Subaperture

Conventional cameras with a small-aperture setting can be well-approximated by a pinhole centered at the aperture's center. This approximation is widely used in many practical applications (including CC-BMD) [6, 17, 26]. In LFCs, the characteristics of light refraction over different subapertures vary in accordance with their positions due to the effect of large-aperture lens. This effect cannot be captured with a pinhole array (as the main lens is not involved); *e.g.*, a beam of parallel rays through LFC-lens *converge* at the focal point, but will pass *parallel* through a pinhole camera array. To account for this effect, we approximate subapertures as pinholes over subaperture-centers, and yet configured to obey the refractions incurred at that portion.

Fig. 4(a) shows a flatland ray tracing model for a subaperture positioned above the optical-center and a world point with positive X coordinate. Following the thin-lens equation with focal length f, a light-ray from a world point $\{X_s, Y_s, Z_s\}$ through the subaperture has to pass through the point of intersection of the principal ray (*i.e.*, a ray

(a) Case 1: $X_s \to +$ and $x_s \to -$ (b) Case 2: $X_s \to -$ and $x_s \to +$ (c) Case 3: $X_s \to -$ and $x_s \to -$

Figure 4. Different cases of world-to-sensor mapping for a subaperture positioned at positive X axis. The same relations hold good even for subapertures positioned at negative X axis due to symmetry about optical axis. (Variables having negative values are marked with $(-)$)

through the optical center) and a fronto-parallel plane at a distance u_0 from the optical center, where u_0 is given by

$$\frac{1}{u_0} + \frac{1}{|Z_s|} = \frac{1}{f} \implies u_0 = \frac{f|Z_s|}{|Z_s| - f}. \quad (7)$$

Note that world coordinate Z_s is negative according to our convention (*i.e.*, $|Z_s| = -Z_s$). From Fig. 4(a), similarity of triangles ΔABO and ΔODC gives

$$\frac{r}{X_s} = \frac{u_0}{Z_s} \implies r = \frac{u_0 X_s}{Z_s}. \quad (8)$$

From similarity of ΔPQD and ΔPGS, we get

$$\frac{k - r}{k - x_s} = \frac{u_0}{u} \implies x_s = r \cdot \frac{u}{u_0} - k \cdot \left(\frac{u}{u_0} - 1 \right). \quad (9)$$

Figs. 4(b-c) illustrate cases of the same subaperture but with world point having negative X coordinate. It can be verified that Eqs. (7)-(9) are quite general and hold good for this situation as well, and also for the case where $u_0 > u$ (please refer to supplementary). Moreover, (due to symmetry about the optical axis) these equations are valid even for subapertures positioned below the lens-center. Substituting in Eq. (9), u_0 and r from Eqs. (7)-(8) yields

$$\begin{aligned} x_s &= \frac{uX_s}{Z_s} - k \cdot \left(\frac{u}{f|Z_s|} \cdot (|Z_s| - f) - 1 \right), \\ &= \frac{uX_s}{Z_s} - k \left(\frac{u}{f} - 1 \right) - \frac{ku}{Z_s}, \quad \because |Z_s| = -Z_s. \end{aligned} \quad (10)$$

The flatland analysis of Eq. (10) can be extended to 3D world coordinate system and a 2D sensor plane as

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} u & 0 & k_x(f-u)/f \\ 0 & u & k_y(f-u)/f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \frac{1}{Z} \begin{bmatrix} k_x u \\ k_y u \\ 0 \end{bmatrix}, \quad (11)$$

where we have dropped the subscript s for brevity, and k_x and k_y are the distances of subaperture from the optical-center in x and y directions, respectively. Representing

$[x, y, 1]^T$ as $\mathbf{x}$, $[X, Y, Z]$ as $\mathbf{X}$, the matrix as $K_{k_{xy}}$, and the vector as $\mathbf{b}_{k_{xy}}$, Eq. (11) can be concisely represented as

$$\mathbf{x} = \frac{1}{Z} \cdot (K_{k_{xy}} \mathbf{X} - \mathbf{b}_{k_{xy}}), \quad (12)$$

where subscript k_{xy} represents individual subapertures in accordance with their separations k_x and k_y. Note that the matrix $K_{k_{xy}}$ is of full rank or invertible.

3.2. Homographies for LFC blur

In this section, we develop homography transformation for a single camera pose-change and then extend it to our MDF-based blur model. As in Eq. (2), consider a single world coordinate change from $\mathbf{X}$ to $\mathbf{X}'$ as

$$\mathbf{X}' = R\mathbf{X}. \quad (13)$$

Using world-sensor mapping in Eq. (12), the mapping of sensor coordinate from $\mathbf{x}$ to $\mathbf{x}'$ (corresponding to the world coordinate mapping from $\mathbf{X}$ to $\mathbf{X}'$) is given by

$$\begin{aligned} &Z' \cdot K_{k_{xy}}^{-1} \mathbf{x}' + K_{k_{xy}}^{-1} \mathbf{b}_{k_{xy}} = Z \cdot RK_{k_{xy}}^{-1} \mathbf{x} + RK_{k_{xy}}^{-1} \mathbf{b}_{k_{xy}} \\ &\text{or } \mathbf{x}' = \frac{1}{Z'} \left(ZK_{k_{xy}} RK_{k_{xy}}^{-1} \mathbf{x} + (K_{k_{xy}} RK_{k_{xy}}^{-1} - I) \mathbf{b}_{k_{xy}} \right) \end{aligned} \quad (14)$$

which gives the homography mapping for subaperture k_{xy}. From Eqs. (6)&(14), the parameter set γ_i comprises of $\mathbf{b}_i$ and scene depth, to capture the parallax and lens effects.

Considering multiple pose-changes, we can represent the motion-blurred subaperture image $B_{k_{xy}}$ as

$$B_{k_{xy}} = \sum_\lambda \omega_\lambda \cdot I_{k_{xy}}(K_{k_{xy}}, R_\lambda, \gamma_{k_{xy}}), \quad (15)$$

where $I_{k_{xy}}(\cdot)$ performs the warping function according to Eq. (14) and MDF ω_{λ_0} represents the fraction of time the world stayed in rotational pose R_{λ_0}. Note that the MDF ω_λ is *shared* among all the subapertures.

We also throw light on the possibility of individually deblurring subaperture images using CC-BMD. Assuming $K_{k_{xy}} = K \; \forall k_{xy}$ and neglecting $\gamma_{k_{xy}}$ necessitates different MDFs for capturing the one-to-many mapping of LF. This distorts their mutual consistencies (*e.g.*, see epipolar image or EPI of Figs. 7(d-e)), mainly due to shift-ambiguity of latent image-MDF pair [15], and relative estimation-error of different MDFs. This adversely affects the refocusing and f-stopping functionality of LFs [16]. Furthermore, since blind deblurring is very costlier than non-blind, the computational cost climbs steeply with spatial and angular resolutions.

4. Optimization of LF-BMD

In this section, we discuss our LF-BMD approach that comprises of two steps: First, estimate the *common* MDF from a *single* subaperture image using BMD, and second, employ the estimated MDF to perform low-cost non-blind deblurring of *remaining* subaperture images (in parallel).

4.1. LF-MDF Estimation

The homography mapping of the center-subaperture image (*i.e.*, $k_x = k_y = 0$ in Eq. (14)) is equivalent to that of a CC-pinhole model [6], *i.e.*,

$$\mathbf{x}' = \lambda \cdot KRK^{-1}\mathbf{x} \quad \because K_{k_{xy}} = K \text{ and } \mathbf{b}_{k_{xy}} = 0, \quad (16)$$

where scalar $\lambda \, (= Z/Z')$ normalizes the third coordinate of $\mathbf{x}'$ to unity (Eq. (11)). Note that even though depths Z and Z' are present in Eq. (16), it is *not* required for homography mapping (and thus for MDF estimation) since it translates to a normalization of the third coordinate of $\mathbf{x}'$ to unity (see structure of x in Eq. (11)) through λ [6, 17, 26, 23]. Thus, any state-of-the-art CC-BMD method can be employed to find the LF-MDF using the center-subaperture image.

4.2. EFF for Non-Blind Deblurring of LFs

Since a common MDF is shared among all subaperture images, we utilize the estimated MDF to perform non-blind deblurring of individual subaperture images. For a non-centered subaperture, camera matrix $K_{k_{xy}}$ varies with subaperture positions and the additive term of Eq. (14) is nonzero (which makes it different from the CC-pinhole case). Eventhough $1/Z'$ in Eq. (14) can be obtained by normalization (as in CC-case), the depth information Z is required for homography mapping to capture parallax and lens effect. A direct approach for non-blind deblurring involves constructing a large matrix M_{kxy} using the MDF formulation of Eq. (15) for each subaperture k_{xy}, to solve the optimization problem

$$\hat{I_{k_{xy}}} = \min_{I_{k_{xy}}} \|M_{k_{xy}} I_{k_{xy}} - B_{k_{xy}}\|_2^2 + \text{prior}(I_{k_{xy}}), \quad (17)$$

where 'prior' is an image regularizer, such as total variation (TV), sparsity in image gradient [26], dark channel [26],

etc. As a full-resolution LF is composed of numerous subaperture images, construction of M_{kxy} and optimization of individual subaperture images with priors are computationally expensive. To this end, we elegantly extend the efficient filter flow (EFF) employed in CCs [8] to LFCs.

The EFF approximates space-variant blur in an image to be locally space-invariant over small image patches. Using this approximation, we can simplify the blurring process in a subaperture image as

$$B_{k_{xy}} = \sum_{i=1}^{R} \mathbf{C_i}^\dagger \cdot \left\{ h_{k_{xy}}^i * (\mathbf{C_i} \cdot I_{k_{xy}}) \right\}, \quad (18)$$

where i iterates over R overlapping patches in clean subaperture image $I_{k_{xy}}$, $\mathbf{C_i} \cdot I$ is a linear operation which extracts the ith patch from I, $(h * \mathbf{C_i} \cdot I)$ performs a convolution with kernel h on ith patch, and $\mathbf{C_i}^\dagger$ inserts the patch back to its original position with a Barlett windowing operation. The convolution kernel $h_{k_{xy}}^i$ corresponding to the ith patch center can be derived using Eq. (15) as

$$h_{k_{xy}}^i = \mathbf{C_i} \cdot \left(\sum_\lambda \omega_\lambda \cdot \delta^i(K_{k_{xy}}, R_\lambda, \gamma_{k_{xy}}) \right), \quad (19)$$

where δ^i is an image of the same size as that of the subaperture image with only an impulse located at the ith patch center. EFF requires MDF-based motion blur model to be calculated *only* at patch centers and eliminates the need for building large matrices for optimization, as in Eq. (17). The EFF allows for an efficient patch-based deblurring:

$$\hat{I_{k_{xy}}} = \sum_{i=1}^{R} \mathbf{C_i}^\dagger \cdot \text{deconv}\left(h_{k_{xy}}^i, (\mathbf{C_i} \cdot B_{k_{xy}}) \right), \quad (20)$$

where 'deconv' indicates non-blind deconvolution which is computationally efficient as compared to blind deblurring.

5. Analysis and Discussions

In this section, we elaborate on the validity of rotation-only approximation in LF-BMD, and depth estimation.
Rotation-only approximation: To reduce the number of unknowns in MDF, CC-BMD methods typically approximate full 6D motion to 3D [17, 23]. As the effect of translation is negligible for normal camera-shakes [23], 3D rotation-only approximation is commonly employed in CCs. We show that this is valid for LFCs too. Considering the worst-case (WC) *plausible* camera translation as $\widehat{T} = [|r|, |r|, |r|]$, [23] shows for CC that the corresponding WC pixel translation $t = [t_x, t_y, t_z] = K_0 \widehat{T}/Z'$ can be *ignored* ($K_0 = diag(u, u, 1)$). We claim that pixel translations in our LFC-model is *equivalent* to that of CC for inplane translations, and *approximates* t for 3D $\widehat{T}$. Considering $\mathbf{X}' = R\mathbf{X} + T$ in Eq. (13), T in SA i (using Eq. (14))

Clean and blurred LF using trajectory 1 of [11], defocus and correspondence cues [22]
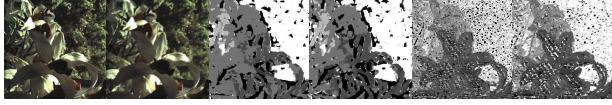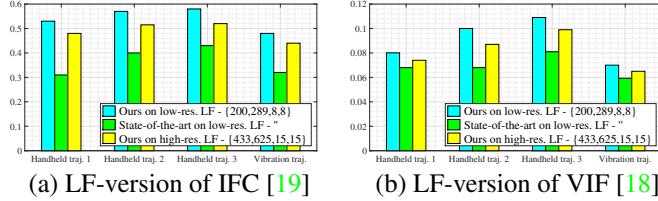
Figure 5. Evaluation of depth estimation cues.

(a) LF-version of IFC [19]   (b) LF-version of VIF [18]

Figure 6. Quantitative evaluation for wide-angle setting ($f = 29$ mm) using real hand-held trajectories (from [11]) and irregular camera motion using vibration trajectory (from [7]).

amounts to $\hat{t} = K_i T/Z' = (K_0 + M_i)T/Z'$, where $M_i = [0, 0, k_x(f-u)/f; 0, 0, k_y(f-u)/f; 0; 0, 0, 0]$. For $T_z = 0$, $M_i T/Z' = 0$ (*equal* effect in CC & LFC). Note that WC displacement happens for highest k_x or k_y ($= f/4$). For 3D $\widehat{T}$, LFC $\hat{t} = [\alpha t_x, \alpha t_y, t_z]$, where $\alpha = 1 + |(f - u)/4u|$ or $1 + |f|/(4|Z'|)$ (using Eq. (7)). As $|Z'|$ is in the order of m and $|f|$ in mm, $1 \approx \alpha < 2$ (*i.e.* $\hat{t} \approx t$). Hence proved.

Depth Estimation: To estimate the convolution kernels for different SAIs (for EFF), our algorithm requires depth information (γ in Eq. (19)). We use [22] to estimate depth for each patch by picking the most-confident depth estimate within that patch (without final depth refinement). Our consideration of uniform depth within a small image patch is analogous to the flatness and global smoothness priors commonly used for final depth-refinement [22, 9]. Depth estimation method in [22] is as follows. Refocusing LF translates to a skew in EPI, and the features of EPIs for a image point will be vertical (or horizontal depending on projection) when it is at focus [16]. [22] skews EPIs corresponding to different depths, and picks among them the depth which makes EPIs vertical. Motion blurred LFs also possess EPI characteristics required for depth estimation (compare EPIs in Figs. 7-8(a) and depth cues in Fig. 5).

6. Experimental Results

In this section, we provide quantitative and qualitative evaluations to highlight the computational gain of our approach and its ability to deal with full-resolution LFs with competitive performance. We also show that our method can deal with both wide-angle systems and irregular camera trajectories, unlike the state-of-the-art LF-BMD [20].

Datasets used: For real experiments on low-resolution LFs, we used the motion blurred LF dataset of [20]. Since there exist no full-resolution motion blur LF-datasets, we create one with LFs captured using `Lytro Illum`, and decoded raw-LFs to `MATLAB` format full-resolution LFs using [4].

LF-resolution $\{x, y, u, v\}$	State-of-the-art [20] **(GPU-based)**	Ours **(CPU-based)**
$\{200, 200, 8, 8\}$ (Dataset of [20])	2 hrs, 20 mins	8.21 mins (Gain **17.05×**)
$\{200, 289, 8, 8\}$ (Low-res. LF)	3 hrs, 17 mins	12.62 mins (Gain **15.61×**)
$\{433, 625, 15, 15\}$ **(Full-res. LF)**	Not feasible (Resource allocation error)	38 mins* **(Feasible)**

Table 1. Time comparisons. *Over 90% of the time is used for *low-cost* 197 non-blind deblurring parallelized in 8 cores of a CPU. Using more cores or GPU further improves the speed significantly.

For quantitative evaluation, we synthesized motion blur on clean full-resolution LFs using real handheld trajectories from [11] with 29 mm focal-length (wide-angle setting) and $1/50$ s exposure time. For irregular motion, we used real vibratory ego-motion trajectory from [7].

Comparison methods: We consider mainly the current state-of-the-art LF-BMD [20] for evaluation. To demonstrate the *ineffectiveness* of CC methods on LFs, we also use state-of-the-art CC-BMD methods [12] and [17] to perform *independent* deblurring on individual subaperture images. The codes for [20, 17] and [12] are downloaded from the authors' website, and used their default parameters for all methods. Additional analysis, comparisons, and implementation details are provided in our supplemental material.

Quantitative Evaluation: We introduce an LF-version of information fidelity criterion (IFC) [19] and visual information fidelity (VIF) [18], which are shown to be the best metrics for BMD evaluation in [13], by averaging these metric over subaperture images. As processing full-resolution LFs using [20] is not feasible, we use a downsampled version (by ≈ 0.5) of our dataset to perform comparisons with [20]. Using IFC/VIF, Figs. 6(a-b) compare with [20] for wide-angle scenario (using real trajectories of [11]) and irregular camera motion (using [7]). It is evident from Fig. 6 that our method performs better than [20] (performance degradation of [20] may be attributed to its inability to model these scenarios); ours can also deblur full-resolution LFs (unlike [20]). Table. 1 gives the timing comparisons with the state-of-the-art [20]. It is evident that, even though our method uses only CPU, we achieve a gain of atleast an order relative to the GPU-based [20]. Also, our method performs full-resolution LF-BMD within three-quarters of an hour, which can be further improved using more cores or GPUs.

Qualitative Evaluation: We qualitatively evaluate our deblurring performance on real LFs. Fig. 7 gives an example of low-resolution LF. Note that the EPIs of [20], [12] and [17] are not consistent with the input. Also, there exists ringing artifacts in Fig. 7(c) of [20] (especially in upper leaves). In contrast, our result in Fig. 7(b) reveals intricate

| (a) Input | (b) Ours | (c) Srinivasan *et al.* [20] | (d) Krishnan *et al.* [12] | (e) Pan *et al.* [17] |

Figure 7. Comparison using low-resolution LF ({200, 200, 8, 8}) from dataset of [20]. (a) Input, (b) Ours, (c) State-of-the-art LF-BMD [20], (d) State-of-the-art CC-BMD [12] (e) State-of-the-art CC-BMD [17]. Note the inconsistencies in EPI w.r.t input for c (possibly due to convergence issues) and d-e (possibly due to *lack* of dependency among BMD of subaperture images). Also, notice the ringing artifacts in the upper leaves in c. In contrast, ours reveals more details (like veins of lower leaf), has negligible ringing artifacts, and EPI is consistent.

| (a) Input | (b) Ours | (c) Srinivasan *et al.* [20] | (d) Pan *et al.* [17] |
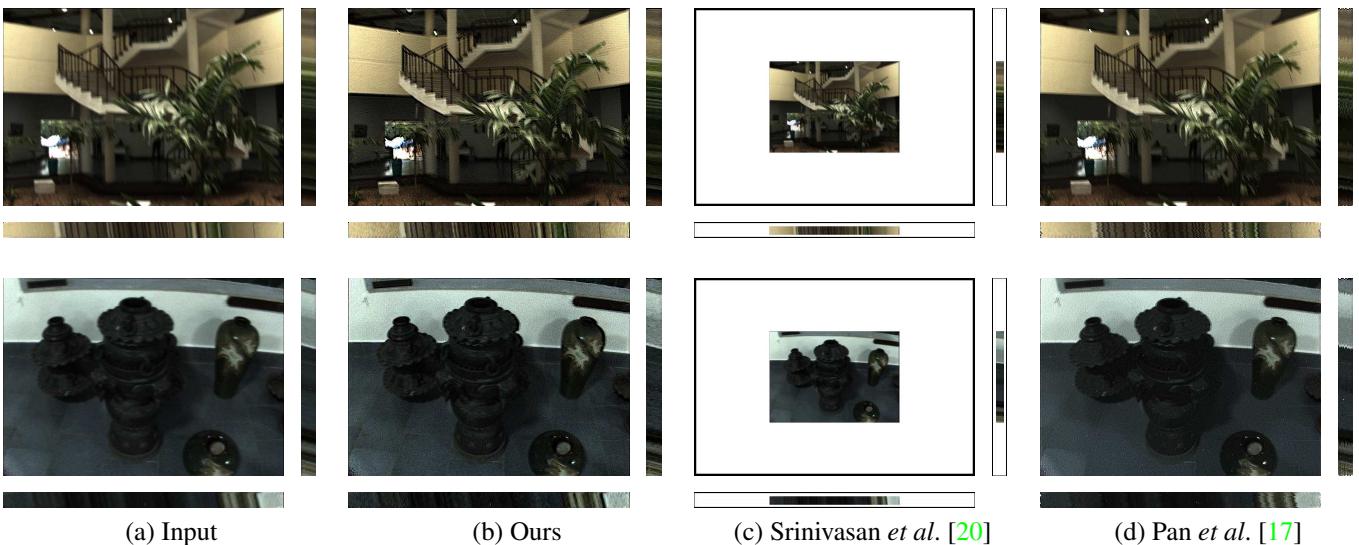
Figure 8. Comparisons using full-resolution LF ({433, 625, 15, 15}) of `Lytro Illum`. Top-row shows a well-lit case and bottom row shows a low-light scenario. (a) Input, (b) Ours, (c) State-of-the-art LF-BMD [20] and (d) State-of-the-art CC-BMD [17]. [20] can *only* deblur downsampled LF due to computational constraints. Ours produce a superior full-resolution LF with consistent EPIs in all cases.

details (see the veins in lower leaf), has negligible ringing artifacts and produces consistent EPIs. Since this example is captured using low-light setting, noise is present in the input as well as our output. A detailed analysis to tackle noise in LF-BMD is presented in our supplementary. Fig. 8 shows comparisons with real full-resolution LFs, where the top and bottom rows depict well-lit and low-lit scenarios, respectively. The LF-BMD of [20] processes *only* a down-sampled LF (both spatially and angularly) due to computational constraints. In contrast, the proposed method gives superior results in full-resolution and with consistent EPIs.

7. Conclusions

We introduced a novel interpretation of motion blur in 4D LF as *independent* blurring of multiple 2D images, yet all sharing a *common* motion parametrization. This paved the way for performing LF deblurring as a single 2D blind deblurring (to estimate the common motion) and paralleliz-able *low-cost* 2D non-blind deblurring of multiple images. Our approach overcomes several major drawbacks of the state-of-the-art, such as heavy computational cost, ability to deblur *only* low-resolution LFs, and GPU-processing. Unlike the state-of-the-art, our model realistically captures refraction effects of lens, and works for wide-angle scenarios and irregular ego-motion as well. As LF cameras continue to evolve with higher resolutions, our divide and conquer strategy will be invaluable for full-resolution deblurring.

References
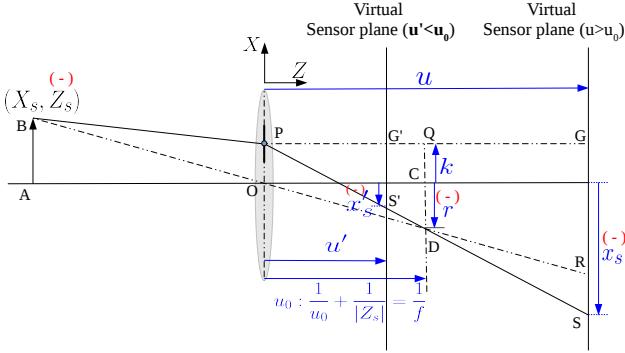
[1] E. H. Adelson and J. Y. Wang. Single lens stereo with a plenoptic camera. *IEEE transactions on pattern analysis and machine intelligence*, 14(2):99–106, 1992. 1

[2] D. G. Dansereau, A. Eriksson, and J. Leitner. Richardson-lucy deblurring for moving light field cameras. *CVPR workshop on Light Fields for Computer Vision (CVPR:LF4CV)*, 2017. 2

[3] D. G. Dansereau, O. Pizarro, and S. B. Williams. Decoding, calibration and rectification for lenselet-based plenoptic cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1027–1034, 2013. 2

[4] D. G. Dansereau, G. Schuster, J. Ford, and G. Wetzstein. A wide-field-of-view monocentric light field camera. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 7

[5] A. Gupta, N. Joshi, L. Zitnick, M. Cohen, and B. Curless. Single image deblurring using motion density functions. In *ECCV '10: Proceedings of the 10th European Conference on Computer Vision*, 2010. 1

[6] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004. 4, 6

[7] M. R. Hatch. *Vibration simulation using MATLAB and AN-SYS*. CRC Press, 2000. 7

[8] M. Hirsch, S. Sra, B. Schölkopf, and S. Harmeling. Efficient filter flow for space-variant multiframe blind deconvolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 607–614. IEEE, 2010. 1, 2, 6

[9] A. Janoch, S. Karayev, Y. Jia, J. T. Barron, M. Fritz, K. Saenko, and T. Darrell. A category-level 3d object dataset: Putting the kinect to work. In *Consumer Depth Cameras for Computer Vision*, pages 141–165. Springer, 2013. 7

[10] M. Jin, P. Chandramouli, and P. Favaro. Bilayer blind deconvolution with the light field camera. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 10–18, 2015. 1, 2, 4

[11] R. Köhler, M. Hirsch, B. Mohler, B. Schölkopf, and S. Harmeling. Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In *European Conference on Computer Vision (ECCV)*, pages 27–40. Springer, 2012. 1, 7

[12] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 233–240. IEEE, 2011. 1, 7, 8

[13] W.-S. Lai, J.-B. Huang, Z. Hu, N. Ahuja, and M.-H. Yang. A comparative study for single image blind deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1701–1709. IEEE, 2016. 7

[14] Z. Li, Z. Xu, R. Ramamoorthi, and M. Chandraker. Robust energy minimization for brdf-invariant shape from light fields. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1

[15] M. M. Mohan, A. Rajagopalan, and G. Seetharaman. Going unconstrained with rolling shutter deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4010–4018, 2017. 1, 6

[16] 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*, 2(11):1–11, 2005. 1, 2, 4, 6, 7

[17] J. Pan, D. Sun, H. Pfister, and M.-H. Yang. Blind image deblurring using dark channel prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1628–1636. IEEE, 2016. 1, 3, 4, 6, 7, 8

[18] H. R. Sheikh and A. C. Bovik. Image information and visual quality. *IEEE Transactions on image processing (TIP)*, 15(2):430–444, 2006. 7

[19] H. R. Sheikh, A. C. Bovik, and G. De Veciana. An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Transactions on image processing (TIP)*, 14(12):2117–2128, 2005. 7

[20] P. P. Srinivasan, R. Ng, and R. Ramamoorthi. Light field blind motion deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1, 2, 3, 4, 7, 8

[21] S. Su and W. Heidrich. Rolling shutter motion deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1529–1537. IEEE, 2015. 1

[22] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi. Depth from combining defocus and correspondence using light-field cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 673–680, 2013. 1, 7

[23] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *International journal of computer vision (IJCV)*, 98(2):168–186, 2012. 1, 3, 6

[24] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu. Light field image processing: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 2017. 1

[25] Z. Xiong, L. Wang, H. Li, D. Liu, and F. Wu. Snapshot hyperspectral light field imaging. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1

[26] L. Xu, S. Zheng, and J. Jia. Unnatural l0 sparse representation for natural image deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1107–1114. IEEE, 2013. 1, 3, 4, 6

Divide and Conquer for Full-Resolution Light Field Deblurring
(Supplementary Material)
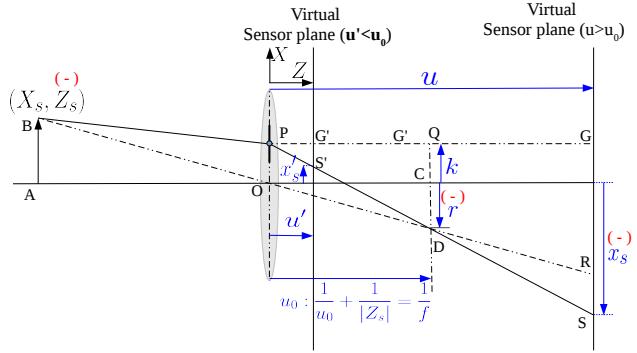
M. R. Mahesh Mohan, A. N. Rajagopalan

Indian Institute of Technology Madras

{ee14d023,raju}@ee.iitm.ac.in
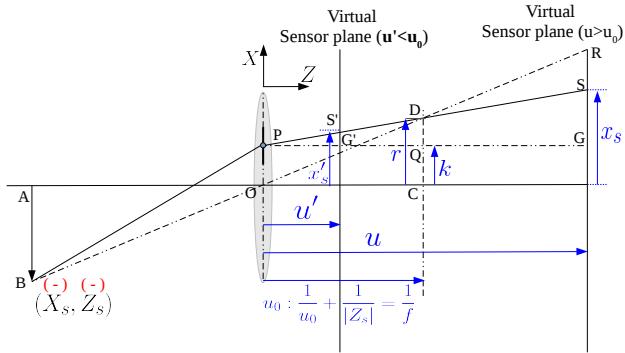
We begin by proving that the world-to-sensor mapping derived in Sec. 3.1, and thus our LF-MDF model, holds good in general. This is followed by a discussion on the choice of our deconvolution method in EFF (Sec. S2), and analysis of various aspects of our LF-BMD (Sec. S3). We then provide our implementation details in Sec. S4, and additional evaluations in Sec. S5. Note that the sections, equations, and figures in the supplementary are numbered with a prefix 'S'.
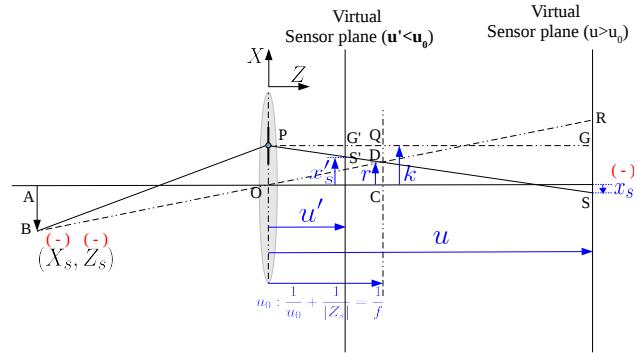
(a) In Case 1 of Fig. 4: $u' < u_0$, $X_s \to +$ and $x'_s \to -$

(b) In Case 1 of Fig. 4: $u' < u_0$, $X_s \to +$ and $x'_s \to +$

(c) In Case 2 of Fig. 4: $u' < u_0$, $X_s \to -$ and $x'_s \to +$

(d) In Case 3 of Fig. 4: $u' < u_0$, $X_s \to -$ and $x'_s \to +$

Figure S1. Different cases of world-to-sensor mapping for $u \le u_0$ for a subaperture positioned at positive X axis. Note that these cases are superimposed on the $u > u_0$ cases shown in Fig. 4. (A symbol ' is added to those variables representing $u \le u_0$.)

| (a) Input | (b) [9] (2X bicubic interpolated) | (c) Direct approach (Gaussian prior) |
| (d) [5] (Hyper-Laplacian prior) | (e) [6] (0.8 norm on gradient) | (f) RL deconvolution [8] |
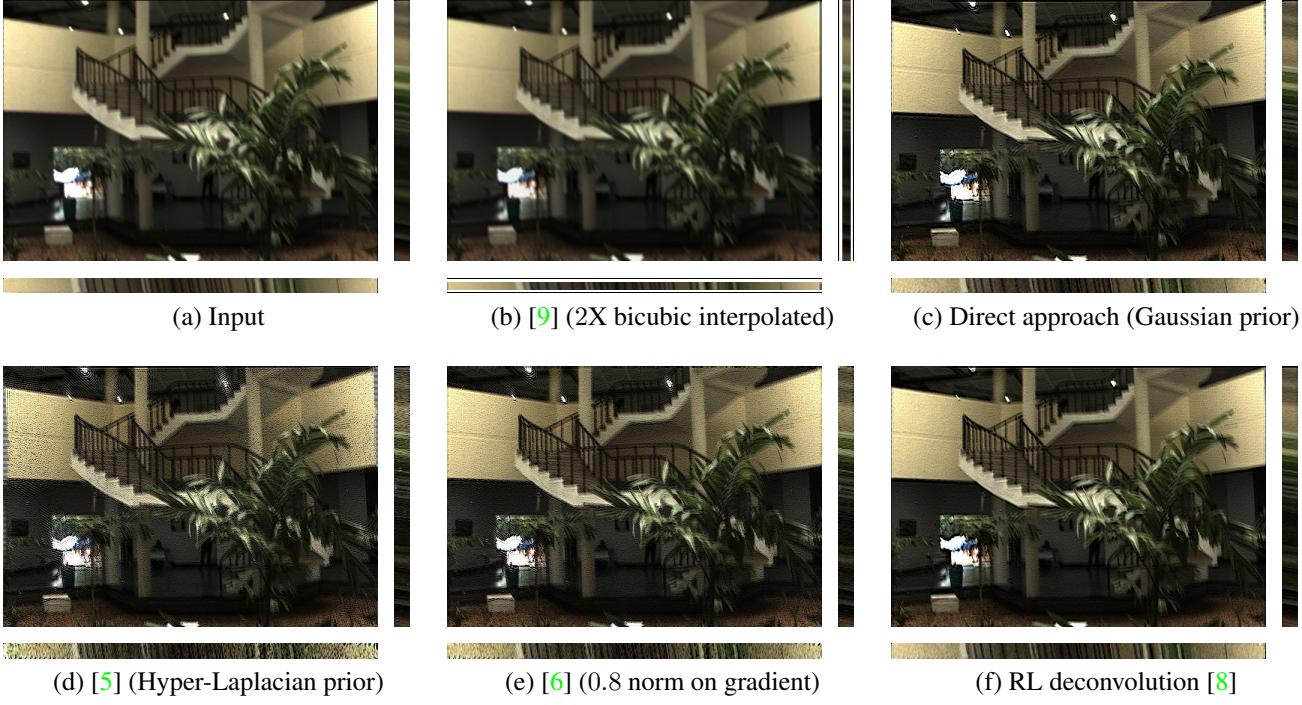
Figure S2. Qualitative evaluation of different LF-EFF deconvolutions using a full-resolution LF. (a) Input, (b) LF-BMD result of [9] for reference (2X bicubic-interpolated). (c) Direct approach using Gaussian prior, (d) Fast MAP estimation with hyper-Laplacian prior using lookup table [5], (e) MAP estimation with heavy-tailed prior ($\alpha = 0.8$) [6], and (f) Richarson Lucy deconvolution [8]. Note the ringing artifacts in c in the saturated regions (*e.g.*, in lights and door exit). RL deconvolution in f produces the best result with negligible artifacts.

Deconvolution method	Direct (Gaussian)	[5] (Fast hyper-Laplacian)	[6] (0.8 norm on gradients)	[8] RL deconv.
Time/SA image (**Full-res. LF**)	1.1 second (closed-form)	6.2 seconds (lookup table)	55 seconds (50 iters.)	80 seconds (50 iters.)

Table S1. Time per subaperture (SA) image for different LF-EFF deconvolution methods for full-resolution LFs.

S1. World-to-Sensor Mapping for the case $u \leq u_0$

We showed in Sec. 3.1 that the three fundamental equations (Eqs. (7)-(9)) that we employed to derive world-to-sensor mapping for a subaperture (and further for the MDF formulation) hold good for different cases of $u > u_0$ (*i.e.*, sensor plane in front of the focal-plane of the lens). Here we show that those relations are also valid for $u \leq u_0$. In Fig. S1, we depict various cases of $u \leq u_0$, superimposed on the $u > u_0$ cases in Fig. 4. For distinguishing both cases, we have used a symbol '′' to indicate variables of $u \leq u_0$ case (*e.g.*, u' indicates u). Eqs. (7) and (8) can be verified using the lens equation and similarity of triangles ΔABO and ΔODC, respectively (as in Sec. 3.1). Similarity of $\Delta PG'S'$ and ΔPQD gives

$$\frac{k-r}{k-x'_s} = \frac{u_0}{u'} \implies x'_s = r \cdot \frac{u'}{u_0} - k \cdot \left(\frac{u'}{u_0} - 1\right),\tag{S1}$$

which is same as Eq. (9). This shows that Eqs. (7)-(9) hold true *in general* for a subaperture positioned at positive X axis as well; *i.e.*, valid irrespective of the scene-point location and the sensor-plane placement ($u > u_0$ or $u \leq u_0$). Due to symmetry about the optical axis of ray diagrams, these relations are *equally valid for subapertures positioned at negative X axis*.

S2. Choice of LF-Deconvolution

In this section, we discuss our choice of deconvolution method employed to perform LF-EFF patch-wise deblurring in Eq. (20). A nonblind LF-EFF deconvolution problem, *i.e.* estimation of a clean image patch given the blur kernel and blurred

| (a) Input | (b) Ours (Prior 0.005) ≈ 17X gain with CPU | (c) Ours (Prior 0.009) ≈ 17X gain with CPU | (d) Ours (Prior 0.05) ≈ 17X gain with CPU | (e) Srinivasan *et al.* [9] **GPU-based** |
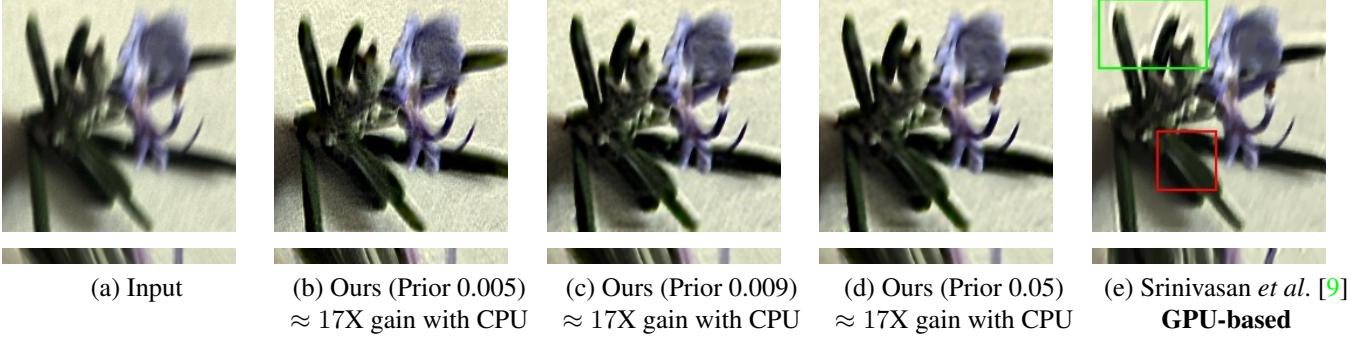
Figure S3. Effect of prior in our LF-BMD (using dataset of [9]). (a) Input, (b) Ours with default smoothness regularization (SR) 0.005, (c) Ours with SR 0.009, (d) Ours with SR 0.05, and (e) State-of-the-art [9] for reference. In e, notice the ringing artifacts in the upper leaves and the suppressed veins of lower leaf (shown boxed). Our result with 0.05 prior is comparable to that of [9], yet with negligible ringing artifacts. Moreover, ours is CPU-based and yet achieves a speed-up of atleast an order ($\approx$ 17X) as compared to [9] which is GPU-based.

image patch, possesses multiple solutions due to zero crossings of filter response, saturation or noise effects, *etc*. Maximum a posteriori (MAP) estimation which imposes prior(s) on clean image patch is typically employed to obtain a single solution from the multiple solution space. A MAP estimation for nonblind deconvolution is given as

$$\hat{I} = \text{deconv}(h, B) = \min_I \|HI - B\|_2^2 + \|\nabla I\|_\alpha \tag{S2}$$

where H captures the blur-kernel information, ∇ is the gradient operator, and B and I are blurred and latent image patches, respectively. We considered four different deconvolution approaches: (a) A direct approach which considers Gaussian prior ($\alpha = 2$) and thus has a closed form solution, (b) A fast deconvolution using hyper-Laplacian prior ($0.5 \leq \alpha \leq 0.8$) which is solved using a lookup table [5], (c) A heavy-tailed prior ($\alpha = 0.8$) which is solved using iterative reweighted least squares process [6], and (d) RL deconvolution with smoothness prior which is solved using iterative process [8]. Figure S2 provides a representative example of LF deblurring quality (using Fig. 8) with different approaches, and Table. S1 gives the average time per subaperture image; it is evident that there exists a trade-off between visual quality and computational speed. In terms of visual quality, we empirically found out that RL [8] is the best, and the direct method comes second but with ringing artifacts (*e.g.*, see Fig. S2(c)). In terms of computational time, the direct method is the most efficient, whereas RL (due to its iterative approach) is less efficient. We have selected RL method due to its superior deblurring quality. However, direct deblurring can be selected for computational efficiency, provided one can tolerate minor ringing artifacts.

S3. Analysis

We showed that our method produces comparable results with respect to the state-of-the-art (GPU-based) [9], yet with significant computational gain even on CPU. Moreover, our method deblurs full-resolution LFs, unlike [9] which can process *only* downsampled LFs. In this section, we consider the effect of noise in our LF-BMD and propound ways to suppress it, and analyse the effect of adding more subaperture images (SAIs) to estimate the MDF (instead of one SAI that we followed). **Noise in LF-BMD:** LF images captured in low-light scenarios possess higher level of shot noise as compared to that of an analogous CC-camera (due to segregation of photons for angular resolution) [12]. As deblurring can be interpreted as enhancing the high-frequency content of the scene, LF-BMD also enhances the high-frequency noise (if present). As discussed in Sec. 4.1, we consider the center subaperture image to estimate the common LF-MDF using [11]. State-of-the-art CC-BMDs frame the objective function in image's gradient space so as to reduce the ill-conditionness [3, 11]. Unlike the gradient of scene features which form contiguous segments, the gradients of shot noise form isolated spikes. Harnessing this, we remove the less-contiguous segments from image-gradient to form the objective function, which reduces the ill-effects of noise in MDF-estimation. For nonblind deblurring (Sec. 4.2), we use the estimated MDF to obtain patch-wise kernels for individual subaperture images (Eq. (19)), and perform deconvolution using [8]. In case of noisy images, we use a higher smoothness prior (regularization of 0.05) for deconvolution to reduce the noise-effect in deblurred images. Our default regularization value is 0.005. To show how noise can be handled as well as give comparison with [9] on Fig. 7 (which uses our default setting), Fig. S3 provides the effect of varying regularization that clearly shows suppression of noise as the prior increases. **More SAIs to find MDF:** Incorporating more SAIs does *not* produce any significant improvement in MDF, while *accentuating* the computational cost. MDF is estimated as $\hat{\omega}_\lambda = \min_{\omega_\lambda} \|H_{I_{k_{xy}}} \omega_\lambda - B_{k_{xy}}\|_2 + \|\omega_\lambda\|_0$. For a maximum 30 pixel blur,

3D rotation space binned by 1 pixel is $\mathbf{29}^3$. Considering a single SAI ($< 1\%$ data), the number of equations (or the number of SAI pixels) will be $\mathbf{10X}$ as that of the number of unknowns, which is already an *overdetermined system (ODS)* and sufficient for MDF estimation [16,23]. Incorporating n more SAIs scales the number of equations by order of n (*but the effect of* more $ODS \approx ODS$), while incurring additional cost for creating *individual H_is* and *handling* large matrix (n H_is stacked).

S4. Implementation Details

System Specifications: We used a PC with an Intel Xeon processor and a 16 GB RAM for all CPU-based experiments, and implemented our algorithm in MATLAB. The repeatedly used EFF routine is implemented in C for computational efficiency. We perform nonblind deblurring of eight subaperture images in parallel. For executing the code of [9], we used a GPU-server and employed a Pascal Titan X GPU. Running time reported in Table. 1 is obtained using these specifications. The camera we used for obtaining full-resolution light field examples is LYTRO ILLUM 40 Megaray.

Parameters: We employed Lytro Desktop App to download LF raw images and [1] to decode raw images into LF Matlab file. The camera parameters focal length f and lens-sensor separation u are obtained from Lytro metadata. As Lytro camera has constant aperture setting as $f/2$, we periodically sampled 197 subapertures in a circular disk of the aperture dimension to obtain k_x and k_y. We used camera metadata and a modified source code of [10] to produce discrete depth with respect to the center subaperture image in individual patches (as discussed in Sec. 4.2).

The sensor coordinate $\mathbf{x}$ corresponding to a scene point varies with subaperture k_{xy} due to parallax and lens effect (*e.g.*, in Fig. S1, for the case of $u > u_0$ the depth Z_s of a scene point maps to sensor coordinate at R through the centre pinhole, whereas shifted by RS through the shifted pinhole). As the depth estimate Z obtained using [10] is with respect to the center subaperture image, it is required to map this to other non-centered subaperture images for retaining one-to-one correspondence between x and Z (in Eq. 14). This we accomplished by warping the estimated depth (with coordinate x) to subaperture k_{xy} (with coordinate x') as $x' = x - \delta x_{k_{x,y}}$, where $\delta x_{k_{x,y}}$ is derived using similarity of ΔDOP and ΔDRS in Fig. 4 or S1:

$$\delta x_{k_{x,y}} = k \cdot \frac{u - u_0}{u_0}. \tag{S3}$$

where u_0 is a function of Z. This relation even holds true for the case of $u < u_0$ (which is verifiable using Fig. S1).

Development: Our algorithm comprises of two steps: blind deblurring of center subaperture image to estimate the common MDF and project the estimated MDF to other subaperture images to perform nonblind deblurring (in parallel) employing EFF. For the first step, as the MDF-based source code of the best CC-BMD [7] is not available and [13] provides only an executable code, we used a modified code of [11] to incorporate LF parameters. For the scale-space based alternative minimization for MDF and latent image, we used 5 scales with 6 iterations each. For all experiments, we used MDF regularization as 0.01 and total variation regularization as 0.005. For the second step, we implemented a C-based EFF code to obtain kernels corresponding to the patch centers using Eq. (19), and employ RL as the deconvolution method in Eq. (20). A pseudo-code is provided in algorithm 1.

Algorithm 1 Light field blind motion deblurring

Require: Decoded motion blurred LF file (LF)	(using [1])
Estimate patch-wise depth using [10]	(following Sec. 5)
$centerSAI \leftarrow LF(0,0)$	
Estimate MDF using the $centerSAI$	(employing [11])
for all SAIs (in parallel) **do**	
Project blur in SAI patches using the estimated MDF	(using Eq. (19))
Patch-wise deconvolution using the projected blur	(using Eq. (20))
Merge individual patches using windowing operation	(Sec. 4.2)
end for	

S5. Additional Evaluations

We provide additional comparisons against the state-of-the-art [9] in Figs. S4-S5. Fig. S4-top provides synthetic experiment results on datatset [1] using real handheld trajectory from [4]. Fig. S4(c)-top shows the result of [9] which is bicubic interpolated to match the full-resolution LF. It is evident from the figure that the interpolation of low-resolution deblurred

(a) Handheld traj. #1 [4] (b) 'Lorikeet' from dataset [1] (c) Ours (d) [9] (2X bicubic interp.)

(e) Vibration traj. [2] (f) 'Jarkanda' from dataset [1] (g) Ours (h) [9] (2X bicubic interp.)
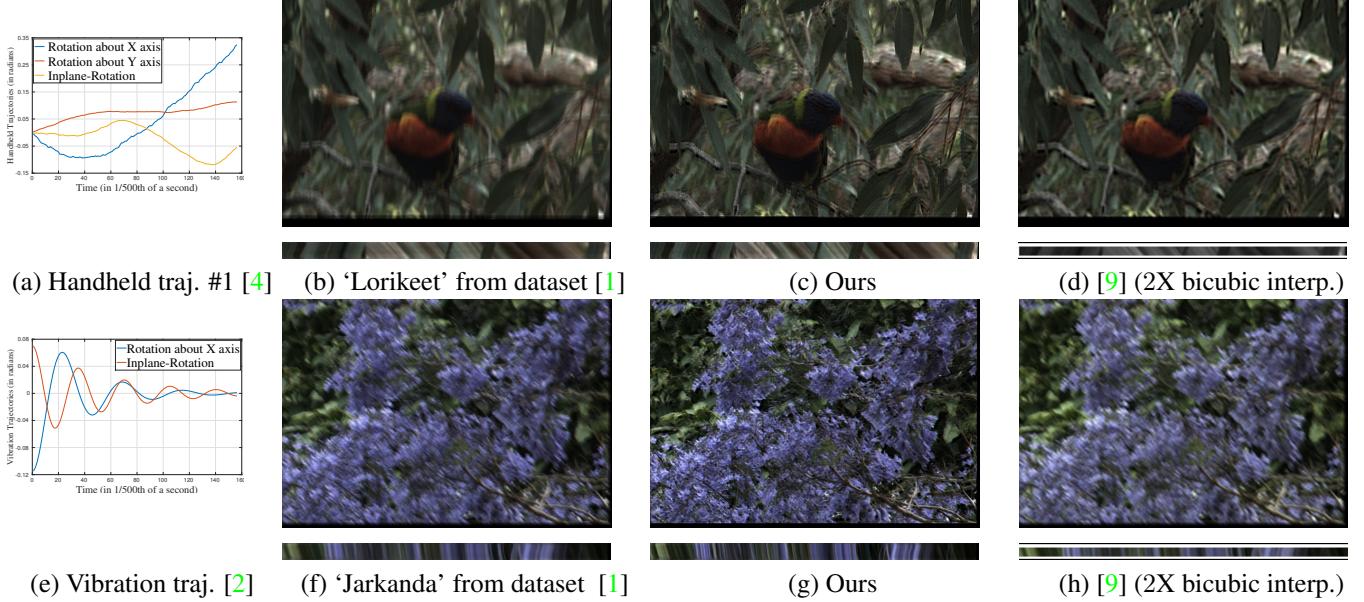
Figure S4. Synthetic experiments in dataset [1] using real handheld [4] and vibration [2] trajectories. (a,e) Trajectories, (b,f) Inputs, (c,g) Ours, and (d,h) Bicubic interpolated result of [9]. Top-row gives a case of handheld trajectory. In d, note that the low-resolution result of [9] after interpolation fails to recover intricate details (*e.g.*, feathers in lorikeet's face). Bottom-row gives a case of irregular motion. Deblurring performance of [9] in h is quite low, possibly due to the inability of its parametric motion model in capturing vibratory motion.

(a) Input (c) Ours (e) Srinivasan *et al*. [9]
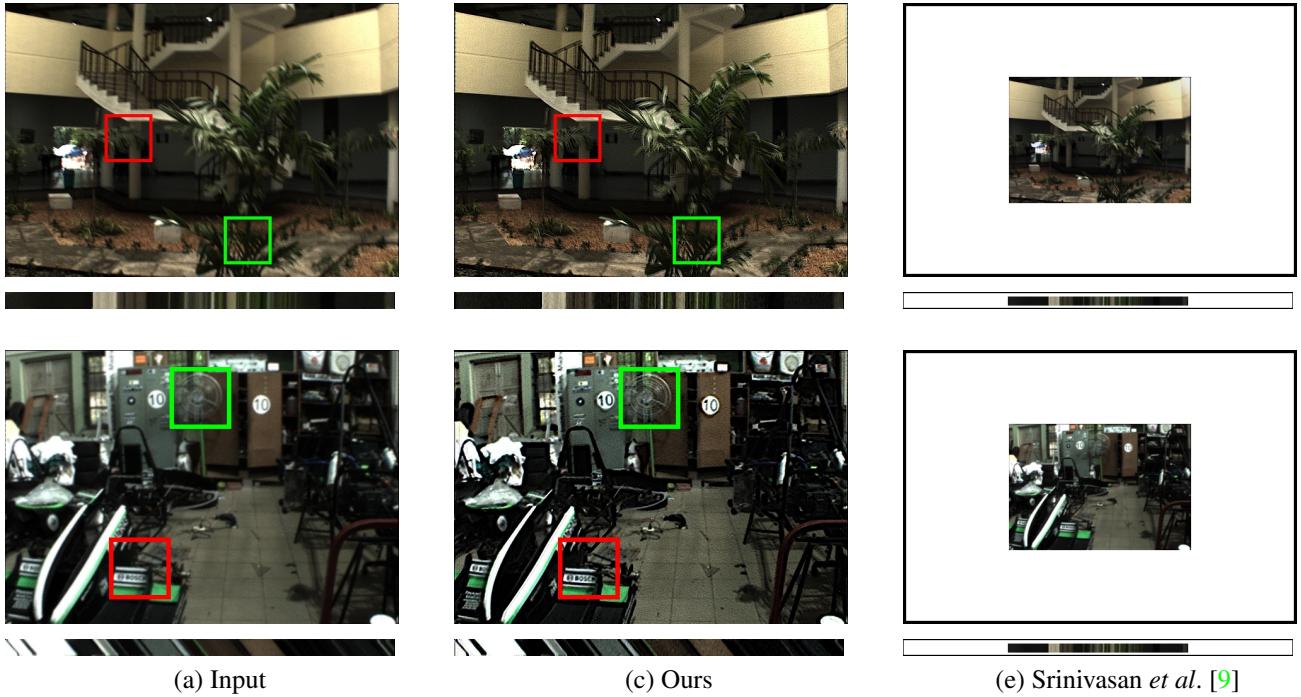
Figure S5. Additional real experiments. (a) Input, (b) Ours, and (c) State-of-the-art [9]. Top row gives a wide-angle scenario (of Fig. 8). Bottom row shows an image of garage. Note that the state-of-the-art [9] *cannot* process a full-resolution LF due to computational considerations, whereas ours performs full-resolution LF-BMD. Also, our method is CPU-based, unlike [9] which requires GPU.

image fails to recover intricate details (*e.g.*, the feathers in lorikeet's face), which further underscores the importance of performing LF deblurring at its full-resolution. Fig. S4-bottom shows a case of irregular motion using vibration trajectory from
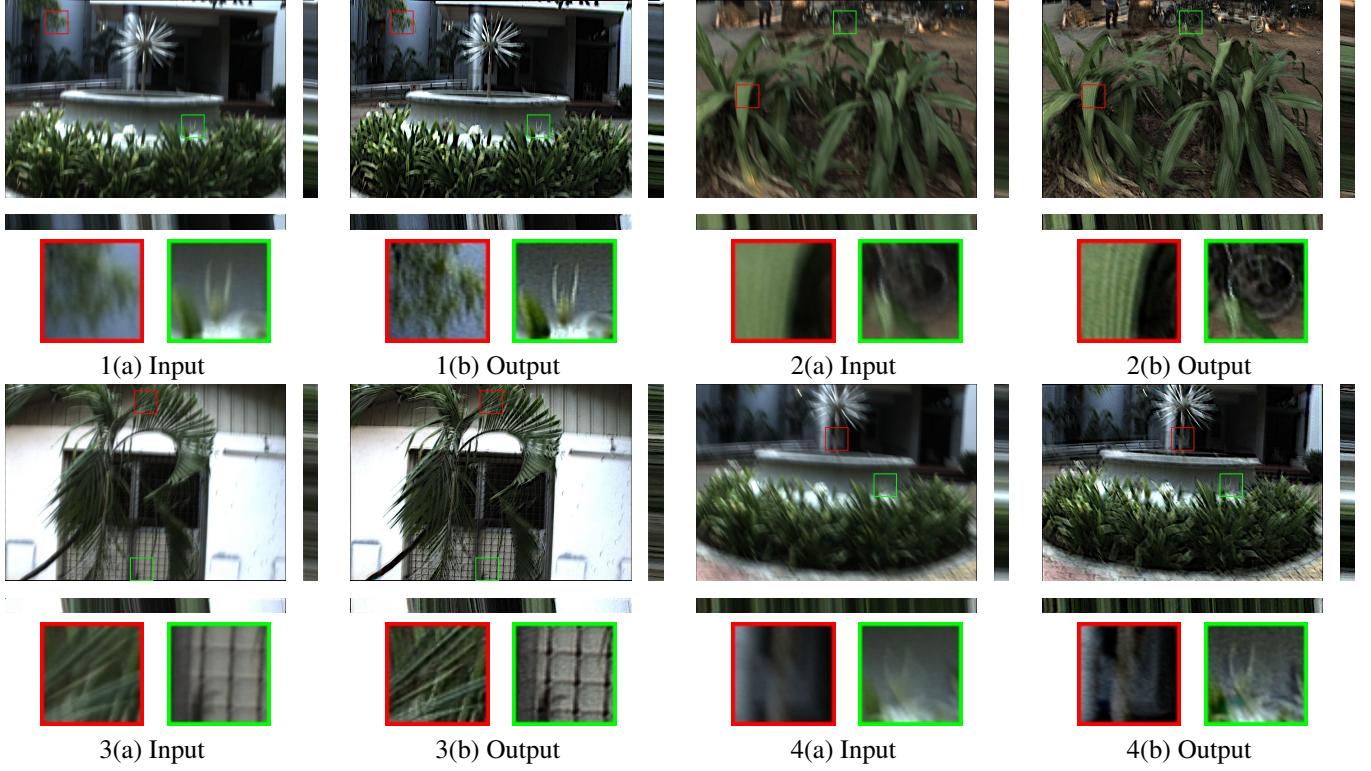
Figure S6. Four additional examples on full-resolution LFs ($\{433, 625, 15, 15\}$) captured using `Lytro Illum`. The first three examples (1-3) depict normal hand-shake blurs, whereas the fourth example gives an example of heavy motion blur. Notice the consistent EPIs in all examples. Also, patches are shown to highlight the deblurring performance.

[2]. Fig. S4(c)-bottom shows the result of [9], where the deblurring performance is inferior (possibly due to the inefficacy of its parametric motion in capturing irregular trajectory). Fig. S5 shows two additional evaluations on real full-resolution LF examples (top-row shows an outdoor wide-angle LF-image and bottom-row shows an indoor narrow-angle LF-image); and Fig. S6 shows four more additional examples, which yet again proves the effectiveness of our proposed method.

References

[1] D. G. Dansereau, O. Pizarro, and S. B. Williams. Decoding, calibration and rectification for lenselet-based plenoptic cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1027–1034, 2013. 4, 5

[2] M. R. Hatch. *Vibration simulation using MATLAB and ANSYS*. CRC Press, 2000. 5, 6

[3] M. Hirsch, S. Sra, B. Schölkopf, and S. Harmeling. Efficient filter flow for space-variant multiframe blind deconvolution. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 607–614. IEEE, 2010. 3

[4] R. Köhler, M. Hirsch, B. Mohler, B. Schölkopf, and S. Harmeling. Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database. In *European Conference on Computer Vision (ECCV)*, pages 27–40. Springer, 2012. 4, 5

[5] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. In *Advances in Neural Information Processing Systems*, pages 1033–1041, 2009. 2, 3

[6] A. Levin, R. Fergus, F. Durand, and W. T. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM transactions on graphics (TOG)*, 26(3):70, 2007. 2, 3

[7] J. Pan, D. Sun, H. Pfister, and M.-H. Yang. Blind image deblurring using dark channel prior. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1628–1636. IEEE, 2016. 4

[8] W. H. Richardson. Bayesian-based iterative method of image restoration. *JOSA*, 62(1):55–59, 1972. 2, 3

[9] P. P. Srinivasan, R. Ng, and R. Ramamoorthi. Light field blind motion deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 3, 4, 5, 6

[10] M. W. Tao, S. Hadap, J. Malik, and R. Ramamoorthi. Depth from combining defocus and correspondence using light-field cameras. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 673–680, 2013. 4

[11] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *International journal of computer vision (IJCV)*, 98(2):168–186, 2012. 3, 4

[12] G. Wu, B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu. Light field image processing: An overview. *IEEE Journal of Selected Topics in Signal Processing*, 2017. 3

[13] L. Xu, S. Zheng, and J. Jia. Unnatural l0 sparse representation for natural image deblurring. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1107–1114. IEEE, 2013. 4