Light Field Blind Motion Deblurring

Pratul P. Srinivasan[1], Ren Ng[1], Ravi Ramamoorthi[2]
[1]University of California, Berkeley [2]University of California, San Diego
[1]{pratul,ren}@eecs.berkeley.edu, [2]ravir@cs.ucsd.edu

Abstract

We study the problem of deblurring light fields of general 3D scenes captured under 3D camera motion and present both theoretical and practical contributions. By analyzing the motion-blurred light field in the primal and Fourier domains, we develop intuition into the effects of camera motion on the light field, show the advantages of capturing a 4D light field instead of a conventional 2D image for motion deblurring, and derive simple methods of motion deblurring in certain cases. We then present an algorithm to blindly deblur light fields of general scenes without any estimation of scene geometry, and demonstrate that we can recover both the sharp light field and the 3D camera motion path of real and synthetically-blurred light fields.

1. Introduction

Motion blur is the result of relative motion between the scene and camera, where photons from a single incoming ray of light are spread over multiple sensor pixels during the exposure. In this work, we make both theoretical and practical contributions by studying the effects of camera motion on light fields and presenting a method to restore motion-blurred light fields. Light field cameras are typically used in situations with optically significant scene depth ranges and out-of-plane camera motion, so it is important to consider how motion blur varies both spatially within each sub-aperture image and angularly between sub-aperture images.

Theory We derive a forward model that describes a motion-blurred light field as an integration over transformations of the sharp light field along the camera motion path. By analyzing the motion-blurred light field in the primal and Fourier domains (Sec. 3 and Figs. 3, 4, 5), we show that capturing a light field enables novel methods of motion deblurring that are not possible with just a conventional image. First, we show that a light field blurred with in-plane camera motion is a simple convolution of the sharp light field with the camera motion path kernel, regardless of the depth contents of the scene (Sec. 3.2.1). This allows us to use simple deconvolution to restore the sharp light field, which cannot be done with conventional images because the magnitude of the motion blur is depth-dependent (Figs. 4, 6). Additionally, we show that a light field blurred with out-of-plane camera motion is an integral over shears of the sharp light field (Sec. 3.2.2). Therefore, we can blindly deblur a light field of a textured plane captured with out-of-plane camera motion by modulating a slice of the Fourier spectrum of the motion-blurred light field (Figs. 4, 7). This is not possible for conventional images due to the spatially-varying blur caused by out-of-plane camera motion.

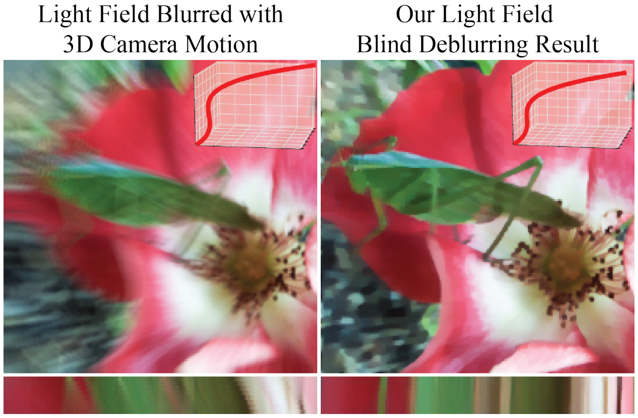

Figure 1. We theoretically study the effects of motion blur on a captured light field and present a practical algorithm to deblur light fields of general scenes captured with 3D camera motion. Left: a 4D light field (visualized as a 2D sub-aperture image and a 2D epipolar slice) is blurred by the synthetic camera motion shown in the inset. Right: absent knowledge of the synthetic motion path, our algorithm is able to accurately recover the sharp light field and the motion path. See Fig. 9 for examples with real handheld camera motion.

Practical Algorithm General light fields of 3D scenes captured with 3D camera motion are integrals over compositions of shears and shifts of the sharp light field. The general light field blind motion deblurring problem lacks a simple analytic approach and is severely ill-posed because there is an infinite set of pairs of sharp light fields and motion paths that explain any observed motion-blurred light field. We propose a practical light field blind motion deblurring algorithm to correct the complex blurring that occurs in situations where light field cameras are useful (Sec. 4). Our forward model is differentiable with respect to the camera motion path parameterization and the estimated light field, allowing us to simultaneously solve for both using first-order optimization methods. Furthermore, by treating motion blur as an integration of transformations of the sharp light field, we can simplify the problem by bypassing any estimation of scene geometry. Instead of solving for a dense matrix that represents spatially and angularly varying motion blurs or separately deblurring each sub-aperture image by solving for a 2D blur kernel and 2D depth map, we directly solve for a parameterization of the continuous camera motion curve in $\mathbb{R}^3$. This is a much lower-dimensional optimization problem, and it allows us to utilize the structure of the light field to efficiently recover the motion curve and sharp light field. Finally, we demonstrate the performance of our algorithm on real (Fig. 9) and synthetically-blurred (Figs. 1, 8) light fields.

2. Related Work

Light Fields The 4D light field [14, 22, 24] is the total spatio-angular distribution of light rays passing through free space, and light field cameras capture the light field that exists inside the camera body [27]. A conventional 2D full-aperture image is produced by integrating the rays entering the entire aperture for each spatial location. Therefore, a captured light field will be interesting and more useful than a conventional image when the equivalent full-aperture image contains significant depth-of-field effects, because this indicates that rays from different regions of the aperture have different values. Common photography situations where capturing a 4D light field would be useful include macro and portrait photography.

Previous work has demonstrated the benefits of lifting problems in computer vision, computer graphics, and computational photography into the 4D light field space. These include rendering 2D pinhole images as slices of the 4D light field [22], stereo reconstruction from a single capture [2], changing the focus and depth-of-field of photographs after capture [27], correcting lens aberrations [26], passive depth estimation [33], glare artifact reduction [31], and scene flow estimation [32].

Previous works have also examined the Fourier spectrum of light fields for various purposes. Chai *et al*. [6] analyzed the spectral support of light fields for sampling in light field rendering and showed that Lambertian objects at specific depths correspond to angles in the Fourier domain. Durand *et al*. [11] analyzed the effects of shading, occlusion, and propagation on the light field spectrum. Ng [25] showed that refocusing a 2D full-aperture image is equivalent to taking 2D slices of the 4D light field spectrum and analyzed the performance of light field refocusing. Liang and Ramamoorthi [23] developed a light transport framework to investigate the fundamental limits of light field camera resolution. Dansereau *et al*. [9] derived the 4D spectral support of light fields for rendering, denoising, and refocusing. Additionally, Egan *et al*. [12] analyzed the spectrum of motion-blurred 3D space-time images to derive filters for efficient rendering of motion-blurred images. In this work, we analyze the Fourier spectrum of motion-blurred light fields to provide intuition for the effects of camera motion on the captured light field and methods to deblur light fields.

Motion Deblurring Blind motion deblurring, removing the motion blur given just a noisy blurred image, is a very challenging problem that has been extensively studied (see [19] for a recent review and comparison of various algorithms). Representative methods for single image blind deblurring include the variational Bayes approaches of Fergus *et al*. [13] and Levin *et al*. [21], and algorithms using novel image priors such as normalized sparsity [18], an evolving approximation to the L_0 norm [34], and L_0 norms on both image gradients and intensities [28].

Previous multi-image blind deblurring works have also presented algorithms that recover a single 2D image, given multiple observations that have been blurred differently [10, 35, 36]. Jin *et al*. [15] present a method that uses a motion-blurred light field of a scene with two depth layers to recover a 2D image and bilayer depth map. Our method also takes a motion-blurred light field as input, but we recover a full 4D deblurred light field as opposed to a 2D texture. Moreover, our method does not need to estimate a depth map.

Many computational photography works have modified the imaging process to make motion deblurring easier. Raskar *et al*. [30] used coded exposures to preserve high frequency details that would be attenuated due to object motion. Another line of work focused on modified imaging methods to engineer point spread functions that would be invariant to object motion. This includes focal sweeps [4, 17], parabolic camera motions [7, 20], and circular sensor motions [3]. In contrast, we focus on the problem of deblurring light fields that have already been captured, and we do not modify the imaging process.

Concurrently with our work, Dansereau *et al*. [8] introduced a non-blind algorithm to deblur light fields captured with known camera motion.

3. A Theory of Light Field Motion Blur

In our analysis below, we perform a flatland analysis of motion-blurred light fields with a single angular dimension u and a single spatial dimension x, and note that it is straightforward to extend this to the full 4D light field with spatial dimensions (x, y) and angular dimensions (u, v). We focus on 3D as opposed to 6D camera motion, so the camera motion path is a general 3D curve and the optical axis does not rotate.

3.1. Forward Model

The observed blurred light field is the integration over the light fields captured at each time t during the exposure:

$$f(x,u) = \int_t l_t(x_t, u_t)dt, \tag{1}$$

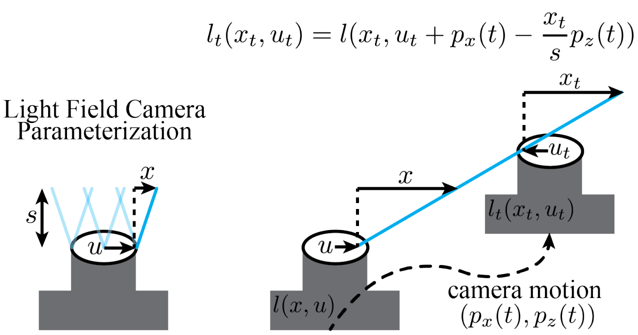
$l_t(x_t, u_t) = l(x_t, u_t + p_x(t) - \frac{x_t}{s} p_z(t))$
Light Field Camera Parameterization
x
s
u
x_t
u_t
$l_t(x_t, u_t)$
x
u
$l(x, u)$
camera motion
$(p_x(t), p_z(t))$

Figure 2. Left: we use a 2-plane parameterization for light fields, where each ray (x, u) is defined by its intercept with the u and x planes separated by distance s. Note that the x coordinate is relative to the u coordinate, which is convenient for later derivations. Right: consider a camera translating along a path $\mathbf{p}(t) = (p_x(t), p_y(t), p_z(t))$ during its exposure (in flatland we consider x and z only). The local camera coordinate frame for each time t has its origin located at the center of the camera aperture. The light field $l_t(x_t, u_t)$ is the sharp light field that would have been recorded by the camera at time t, in the local camera coordinates at time t. The diagram shows that ray (x_t, u_t) in the local coordinate frame at time t is equal to ray $(x_t, u_t + p_x(t) - \frac{x_t}{s} p_z(t))$ in the local coordinate frame at time $t = 0$.

where f is the observed light field and $l_t(x_t, u_t)$ is the sharp light field at time t during the exposure.

Figure 2 illustrates that the light field at time t is a transformation of the sharp light field at time $t = 0$, $l(x, u)$, based on the camera motion path $\mathbf{p}(t) = (p_x(t), p_z(t))$ ($p_y(t)$ is not included in the flatland analysis but is included in the full 3D model). Our light field parameterization is equivalent to considering the light field as a collection of pinhole cameras with centers of projection u and sensor pixels x, and we set the separation between the x and u planes $s = 1$ so x is a ray's spatial intercept 1 unit above u in the z direction. The observed motion-blurred light field is then

$$f(x, u) = \int_t l(x, u + p_x(t) - x p_z(t)) dt. \tag{2}$$

Since the light field contains all rays that intersect the two parameterization planes, this forward model accounts for occluded points, as long as the parameterization planes lie outside the convex hull of the visible scene geometry. Certain rare scenarios, such as a macro photography shot where the camera moves between blades of grass during the exposure, may violate this assumption, but it generally holds for typical photography situations. This model also assumes that the light field parameterization planes are infinite, because camera motion can cause the sharp light field at time t to contain rays outside the field-of-view of the light field at a previous time.

3.2. Space-Angle and Fourier Analysis

We examine the motion-blurred light field in the primal space-angle and Fourier domains to better understand the effects of camera motion on the captured light field. We denote signals in the Fourier domain with capital letters, and use Ω_x and Ω_u to denote spatial and angular frequencies.

It is useful to utilize the Affine Theorem for Fourier transforms [5, 29]: if $h(\mathbf{a}) = g(M\mathbf{b} + \mathbf{c})$, where M is a matrix, $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$ are vectors, and h and g are functions, the relevant Fourier transforms are related as follows:

$$H(\mathbf{\Omega}) = |\det(M)|^{-1} G(M^{-T} \mathbf{\Omega}) \exp(2\pi i \mathbf{\Omega}^T M^{-1} \mathbf{c}), \tag{3}$$

where $\det(M)$ is the determinant of M and $i = \sqrt{-1}$.

We use this to take the Fourier transform of the observed motion-blurred light field in Eq. 2, with transformation matrices $M = \begin{pmatrix} 1 & 0 \\ -p_z(t) & 1 \end{pmatrix}$ and $\mathbf{c} = \begin{pmatrix} 0 \\ p_x(t) \end{pmatrix}$:

$$F(\Omega_x, \Omega_u) = \int_t L\left(\Omega_x + p_z(t)\Omega_u, \Omega_u\right) \exp\left[2\pi i \Omega_u p_x(t)\right] dt. \tag{4}$$

As visualized in Fig. 5, the Fourier spectrum is an integration over shears based on the out-of-plane motion $p_z(t)$ and there is also a phase in the complex exponential corresponding to in-plane motion. This complex exponential is the Fourier transform of $\delta(x)\delta(u+p_x(t))$, so we can rewrite the flatland primal domain motion-blurred light field as

$$f(x, u) = \int_t [l(x, u - x p_z(t)) \otimes \delta(x)\delta(u + p_x(t))] dt. \tag{5}$$

The spatial and frequency domain expressions now separate in-plane motion, which is a convolution with a kernel corresponding to the in-plane camera motion path, and out-of-plane motion, which is an integration over shears in both the spatial and frequency domains. Note that this convolution kernel is restricted to a subspace of the light field space (1D subspace of 2D for flatland light fields, and 2D subspace of 4D for full light fields).

To gain greater insight into these expressions, we consider two special cases for purely in-plane camera motion, and purely out-of-plane camera motion, with general motion being an integral over compositions of these two cases.

3.2.1 In-Plane Camera Motion

For camera motion paths that are parallel to the x and u parameterization planes, $p_z(t) = 0$, and the expression for the primal domain motion-blurred light field simplifies to

$$\begin{aligned} f(x, u) &= l(x, u) \otimes \int_t \delta(x)\delta(u + p_x(t)) dt \\ &= l(x, u) \otimes \delta(x) k(u), \end{aligned} \tag{6}$$

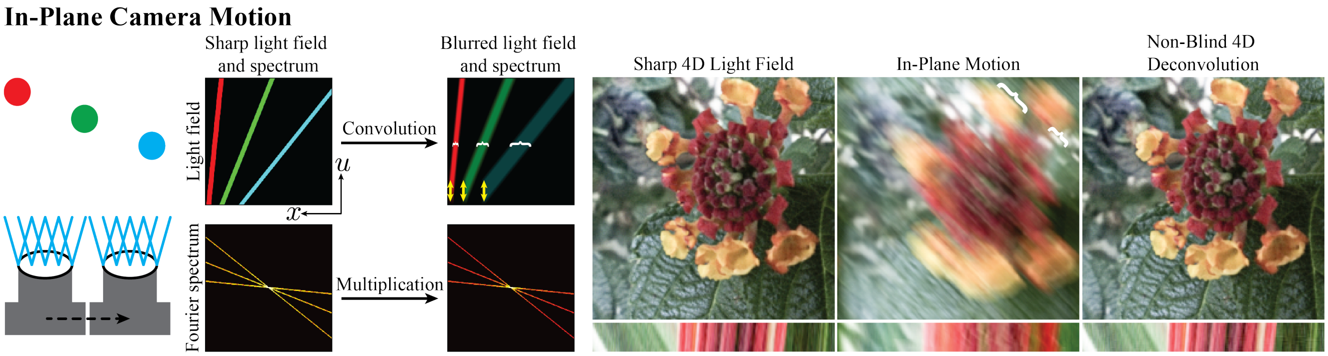

Figure 3. In-plane camera motion is equivalent to a convolution of the light field and the corresponding multiplication of the Fourier spectrum. We are able to easily recover a light field blurred with known in-plane camera motion using 4D deconvolution. Note that in-plane camera motion causes spatially-varying (with x) blur due to varying scene depths, as shown by the white brackets, while the blur magnitude does not vary angularly (with u), as shown by the yellow vertical arrows.

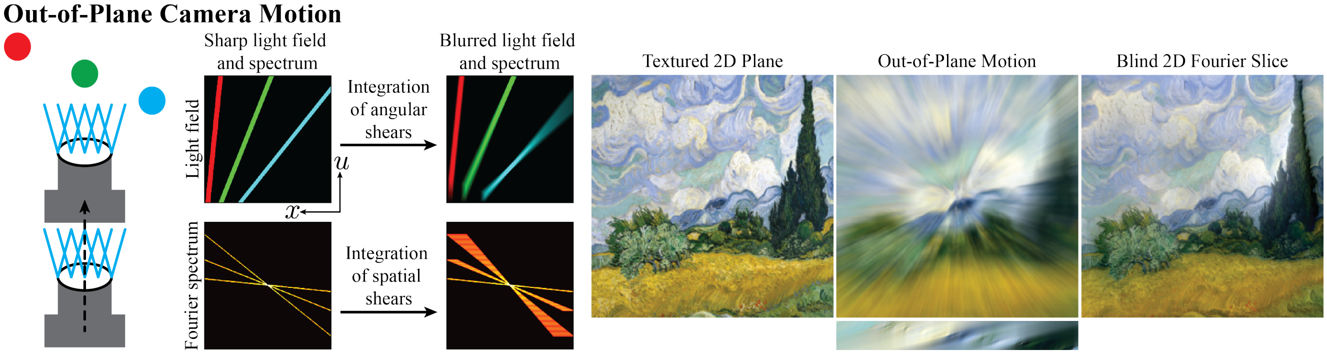
Out-of-Plane Camera Motion
Sharp light field and spectrum
Blurred light field and spectrum
Light field
Fourier spectrum
Integration of angular shears
Integration of spatial shears
u
x
Textured 2D Plane
Out-of-Plane Motion
Blind 2D Fourier Slice

Figure 4. Out-of-plane camera motion is equivalent to an integration over shears in both the primal and Fourier domains. Note that out-of-plane camera motion causes both spatially and angularly varying blur. Given a light field of a single fronto-parallel textured plane (Vincent van Gogh's "Wheat Field with Cypresses") with out-of-plane camera motion, we can blindly recover the texture, with slight artifacts due to finite aperture and edge effects, by modulating a 2D slice of the 4D Fourier spectrum.

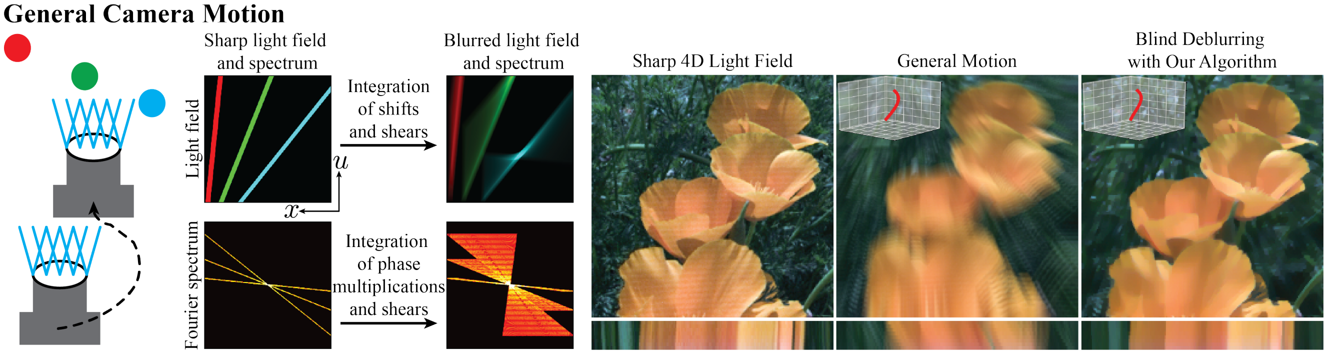
General Camera Motion
Sharp light field and spectrum
Blurred light field and spectrum
Light field
Fourier spectrum
Integration of shifts and shears
Integration of phase multiplications and shears
u
x
Sharp 4D Light Field
General Motion
Blind Deblurring with Our Algorithm

Figure 5. General 3D camera motion is an integration over shears and shifts of the light field and an integration over shears and phase multiplications of the Fourier spectrum. Blindly deblurring light fields captured with general camera motion lacks a simple analytic approach and is severely ill-posed, so we solve this as a regularized inverse problem.

where $k(u) = \int_t \delta(u + p_x(t))dt$ is the integrated in-plane camera motion path.

In the Fourier domain,

$$\begin{aligned} F(\Omega_x, \Omega_u) =& L(\Omega_x, \Omega_u) \int_t \exp[2\pi i \Omega_u p_x(t)]dt \\ =& L(\Omega_x, \Omega_u) K(\Omega_u), \end{aligned} \tag{7}$$

where $K(\Omega_u) = \int_t \exp[2\pi i \Omega_u p_x(t)]dt$ is the integrated in-plane blur kernel spectrum.

An important insight is that for in-plane camera motion, it is possible to take the original light field out of the integral. *This clearly identifies the motion-blurred light field as a simple convolution of the sharp light field with the in-plane blur kernel, regardless of the content and range of depths present in the scene, as illustrated in Fig. 3. No such*

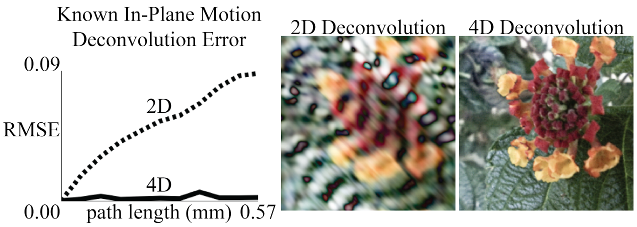

Figure 6. Light fields of general 3D scenes blurred with known in-plane camera motion can be recovered by simple 4D deconvolution. This is not possible with conventional 2D images because the motion blur magnitude is depth-dependent. We synthetically blur a light field with increasing linear in-plane motion, and note that the root mean square error (RMSE) of the central sub-aperture image obtained by 2D deconvolution consistently increases, while the RMSE of the central sub-aperture image obtained by 4D deconvolution of the full light field stays relatively constant.

simple result holds for conventional 2D images, as quantified in Fig. 6, because the motion blur magnitude is depth-dependent. Intuitively, in-plane motion is a convolution of the sharp light field because light field cameras at points along the motion path observe the same set of rays shifted, while conventional cameras at points along the motion path observe disjoint sets of rays. If we know the blur kernel, we can recover the sharp light field by simple deconvolution, as shown in Figs. 3, 6. However, if both the blur kernel and light field are unknown, we need to use priors to estimate the blur kernel and sharp light field, as discussed in Sec. 4.

3.2.2 Out-of-Plane Camera Motion

For purely out-of-plane camera motion, $p_x(t) = 0$, and the expression for the primal domain motion-blurred light field simplifies to

$$f(x,u) = \int_t l(x, u - xp_z(t))dt \tag{8}$$

In the Fourier domain,

$$F(\Omega_x, \Omega_u) = \int_t L(\Omega_x + p_z(t)\Omega_u, \Omega_u)dt. \tag{9}$$

These are simply integrations over different shears of the light field, as illustrated in Fig. 4. It is particularly interesting to consider the light field of a textured fronto-parallel plane $w(x)$ at depth z'. The geometry of our light field parameterization indicates that $l(x,u) = w(xz' + u)$. In the primal domain, the out-of-plane motion-blurred light field of this textured plane is

$$f(x,u) = \int_t w(x(z' - p_z(t)) + u)dt = \int_t w(xz(t) + u)dt, \tag{10}$$

where we define $z(t) = z' - p_z(t)$.

Using the Affine Theorem for Fourier transforms with transformation matrices $M = \left(\begin{smallmatrix} z(t) & 1 \\ 0 & 1 \end{smallmatrix}\right)$ and $\mathbf{c} = \left(\begin{smallmatrix} 0 \\ 0 \end{smallmatrix}\right)$, after noting that the original Fourier transform of the textured plane is $W(\Omega_x)\delta(\Omega_u)$, the Fourier transform of the out-of-plane motion-blurred light field is

$$F(\Omega_x, \Omega_u) = \int_t \frac{1}{|z(t)|} W\left(\frac{\Omega_x}{z(t)}\right) \delta\left(\Omega_u - \frac{\Omega_x}{z(t)}\right) dt. \tag{11}$$

This is also an integration over various shears, each a line with slope given by $\Omega_u = \Omega_x / z(t)$. The motion-blurred light field takes the original texture frequencies (in a line) and shears them to lines of different slopes, followed by integration. Using the sifting property of the delta function, we can simplify the above expression,

$$F(\Omega_x, \Omega_u) = W(\Omega_u) \int_{z_{\min}}^{z_{\max}} \delta\left(\Omega_u - \frac{\Omega_x}{z}\right) \gamma(z)\, dz, \tag{12}$$

where we have switched to integration over z directly (effectively substituting z for t), and for simplicity, we assume $z(t)$ monotonically increases with time. The term $\gamma(z) = (|z|dz/dt)^{-1}$ accounts for the $1/|z|$ factor and change of variables.

Intuitively, the motion-blurred light-field spectrum is a double wedge [6, 9, 11], bounded by slopes $z_{\min}$ and $z_{\max}$ and containing an infinite number of lines in the frequency-domain. The magnitudes along each line are the same, determined by the original texture $W(\Omega_u)$, but every value is uniformly scaled by a factor $\gamma(z)$, based on the amount of time the camera lingered at that depth (other than $W(0)$, which is constant for all lines).

The delta function in Eq. 12 can then be evaluated, setting $z = \Omega_x / \Omega_u$, leading to the simple expression,

$$F(\Omega_x, \Omega_u) = W(\Omega_u)\beta\left(\frac{\Omega_x}{\Omega_u}\right), \tag{13}$$

where the function β includes γ, as well as the change of variables from the delta function, and is given by

$$\beta(z) = \frac{|z|}{\Omega_x dz/dt} \qquad \beta(\Omega_x/\Omega_u) = \left(|\Omega_u| \left.\frac{dz}{dt}\right|_{\Omega_x/\Omega_u}\right)^{-1}. \tag{14}$$

Texture Recovery The structure of the out-of-plane motion-blurred light field enables blind deblurring by a very simple factorization (essentially a rank-1 decomposition of the 2D light field matrix into 1D factors for W and γ or β). One simple approach is to estimate W from any line, then fix the scaling by comparing the overall magnitude of W across lines to estimate the motion blur kernel (β or γ), and finally divide $W(0)$ by the total exposure time.

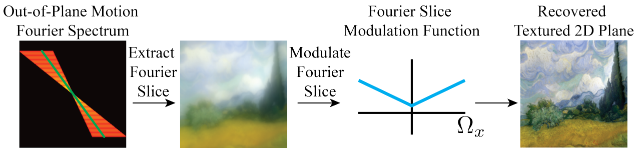

Figure 7. Visualization of process to blindly recover a textured plane from a light field captured with out-of-plane camera motion.

Taking a slice of the light field in the Fourier domain can be implemented in the primal domain by a sheared integral projection, and this is equivalent to refocusing the full-aperture image to a specific depth [25]. Intuitively, this means that blind deblurring of the texture can be performed in the primal domain by computing the full-aperture image refocused to a single depth during the exposure. In summary, *we can separately estimate the blur kernel and the original texture for out-of-plane motion of a light field camera, assuming a single fronto-parallel textured plane, by extracting a slice in the Fourier domain or equivalently refocusing the full-aperture image in the primal domain.*

Figure 4 shows an example of a light field of a textured plane blurred by linear out-of-plane motion. We are able to blindly recover the texture, with slight artifacts due to finite aperture and edge effects, by computing a sheared integral projection (equivalent to taking a 2D slice of the 4D Fourier spectrum). As a practical note, when computing this in the discrete case, we must linearly scale frequencies in the extracted slice by $|\zeta\Omega_x| + 1$ to correct for the value of each discrete frequency being spread across the shear length during the exposure, where ζ is a constant corresponding to the relative time the camera lingers at the depth corresponding to that slice. ζ can be automatically determined by sampling Fourier slices and comparing their magnitudes to calculate the relative time spent at each depth along the motion path. This process is visualized in Fig. 7. As detailed in [25], the resolution of the recovered Fourier slice is limited by the angular resolution of the light field camera.

Comparison with a Conventional Image It is also insightful to compare this to information available from a conventional 2D image (1D in flatland), corresponding to the view from the central pinhole of a light field camera. In this case, we set $u = 0$ in Eq. 10, defining $l(x) = w(xz')$. Since we are now working in 1D, from the Fourier scale theorem,

$$F(\Omega_x) = \int_t \frac{1}{|z(t)|} W\left(\frac{\Omega_x}{z(t)}\right) dt = \int_{z_{\min}}^{z_{\max}} W\left(\frac{\Omega_x}{z}\right) \gamma(z) dz. \tag{15}$$

This is similar to the light field case, except that we no longer have the delta function for multiple sheared lines in 2D; indeed we only have a single 1D line, with a frequency spectrum scaled according to z. It is clear that from the perspective of analysis and recovery, the conventional image case provides far less insight than in the light field case. We cannot separate out the texture W, and the methods for recovery discussed in the light field case do not apply, since there are not multiple lines in a 2D spectrum we can study. In fact, it is not even straightforward to recover texture and motion blur kernel even when one of the factors is known.

3.2.3 General 3D Motion and Scenes

General 3D camera motion is an integral over compositions of shears and shifts of the light field, as shown in Fig. 5. Blindly deblurring a light field of a general scene captured with 3D camera motion lacks a simple analytic approach and is a severely ill-posed problem because there is an infinite set of pairs of light fields and motion paths that explain any observed motion-blurred light field. Below, we present an algorithm to estimate the sharp light field and camera motion path by solving a regularized inverse problem.

4. Blind Light Field Deblurring Algorithm

For blind light field motion deblurring, we estimate both the camera motion curve $\mathbf{p}(t)$ and the sharp light field $\mathbf{l}$. We utilize our forward model derived in Eq. 2 to formulate a regularized inverse problem, and our approach is particularly efficient due to our direct representation of the camera motion curve, as discussed below. We solve a discrete optimization problem, since light field cameras record samples and not continuous functions:

$$\min_{\mathbf{l},\mathbf{p}(t)} ||\hat{\mathbf{f}}(\mathbf{l}, \mathbf{p}(t)) - \mathbf{f}||_2^2 + \lambda\psi(\mathbf{l}), \tag{16}$$

where the first term minimizes the L_2 norm of the difference between the observed motion-blurred light field $\mathbf{f}$ and that predicted by the forward model $\hat{\mathbf{f}}$, and the second term, $\psi(\mathbf{l})$, is a prior on the sharp light field. To address finite aperture and sensor planes, we assume replicating boundaries for the sharp light field. We use bilinear interpolation to transform the sharp light field along the camera motion path, so our forward model is differentiable with respect to the camera path and the sharp light field.

Camera Motion Path Representation We model the camera motion path $\mathbf{p}(t)$ as a Bézier curve made up of n control points in $\mathbb{R}^3$. This approach is much more efficient than the alternative approaches of solving for a dense matrix to represent spatially and angularly varying blurs, or separately deblurring each sub-aperture image. A dense motion blur ray transfer matrix would have size $r \times r$, where r is the number of rays sampled by the light field camera (this matrix would have size 2560000×2560000 for the light fields used in this work). Separately deblurring each sub-aperture image involves estimating a 2D depth map and a 2D convolution kernel, each of size $s \times s$, where s is the number of samples along each spatial dimension (this equates to solving for two matrices of size 200×200 for the light fields used in this work). Instead, we solve for a much lower-dimensional vector of control points with $3n$ elements. In practice, we find that typical camera motion paths can be represented by $n = 3$ or $n = 4$ control points.

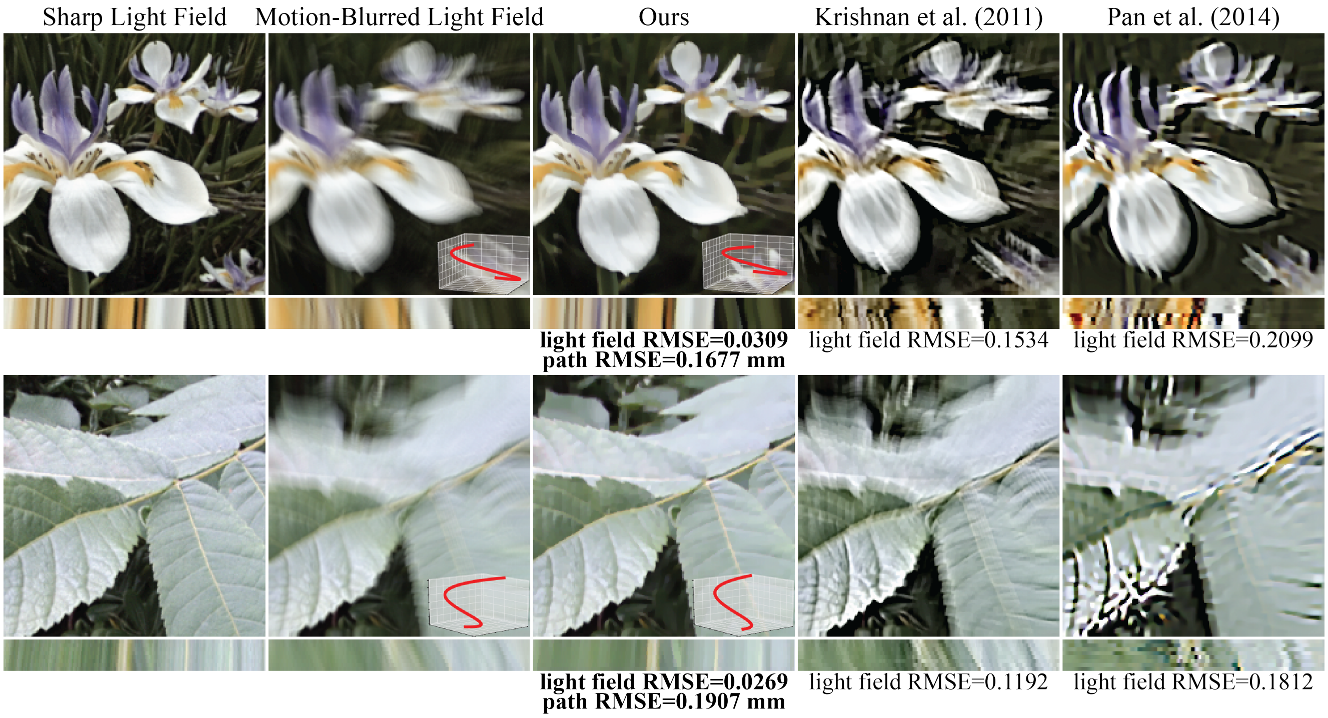

Figure 8. Blind deblurring results on synthetically motion-blurred light fields. Our algorithm is able to correctly recover the sharp light field and estimate the 3D camera motion path, while alternative methods perform poorly due to the large spatial variance in the blur. Additionally, as demonstrated by the epipolar images, other algorithms do not recover a light field that is consistent across angular dimensions. The root mean square error (RMSE) of our deblurred results are consistently lower than those of the alternative methods.

Light Field Prior To regularize the inverse problem above, we use a 4D version of the sparse gradient prior proposed in [34]:

$$\psi(\mathbf{l}) = \sum_{x,y,u,v} \begin{cases} \frac{1}{\epsilon^2}|\nabla \mathbf{l}|^2 & \text{if } |\nabla \mathbf{l}| \leq \epsilon, \\ 1 & \text{otherwise.} \end{cases} \tag{17}$$

This function gradually approximates the L_0 norm of gradients by thresholding a quadratic penalty function parameterized by ϵ, and approaches the L_0 norm as $\epsilon \to 0$.

Implementation Details We utilize the automatic differentiation of Tensorflow [1] to differentiate the loss of the blind deblurring problem in Eq. 16 with respect to the camera motion path control points and sharp light field, and use the first-order Adam solver [16] for optimization.

While the prior in Eq. 17 is effective for estimating the camera motion path, the sharp light fields estimated using this prior typically appear unnatural and over-regularized. We hold the camera motion path constant and solve Eq. 16 for the sharp light field using a 4D total variation (L_1 norm of gradients) prior to obtain the final sharp light field.

4.1. Results

We validate our algorithm using light fields captured with the Lytro Illum camera, that have been blurred by both synthetic camera motion within a 2 mm cube using our forward model in Eq. 2 and real handheld camera motion using a shutter speed of 1/20 second. We compare our results to the alternative of applying state-of-the-art blind image motion deblurring algorithms to each sub-aperture image. As shown in a recent review and comparison paper [19], the algorithms of Krishnan *et al.* [18] and Pan *et al.* [28] are two of the top performers for blind deblurring of both real and synthetic images with spatially-varying blur, so we compare our algorithm to these two methods. As demonstrated by both the synthetically motion-blurred results in Fig. 8 and the real motion-blurred results in Fig. 9, our algorithm is able to accurately estimate both the sharp light field and the camera motion path. The state-of-the-art blind image motion deblurring algorithms are not as successful due to the significant spatial variance of the blur. Furthermore, they are not designed to take advantage of the light field structure and do not estimate a 3D camera motion path, so their results are inconsistent between sub-aperture images, as demonstrated by the epipolar image results.

In the synthetically-blurred examples in Fig. 8, note that our algorithm correctly estimates the ground truth complex camera motion paths and corrects the large spatially-varying blurs in the flowers and leaves. In the real handheld blurred examples in Fig. 9, note that our algorithm corrects the blur in the specularities and edges of the circuit component, the leaves and flower of the rosemary plant, and the hair, eyebrows, teeth, and background plants.

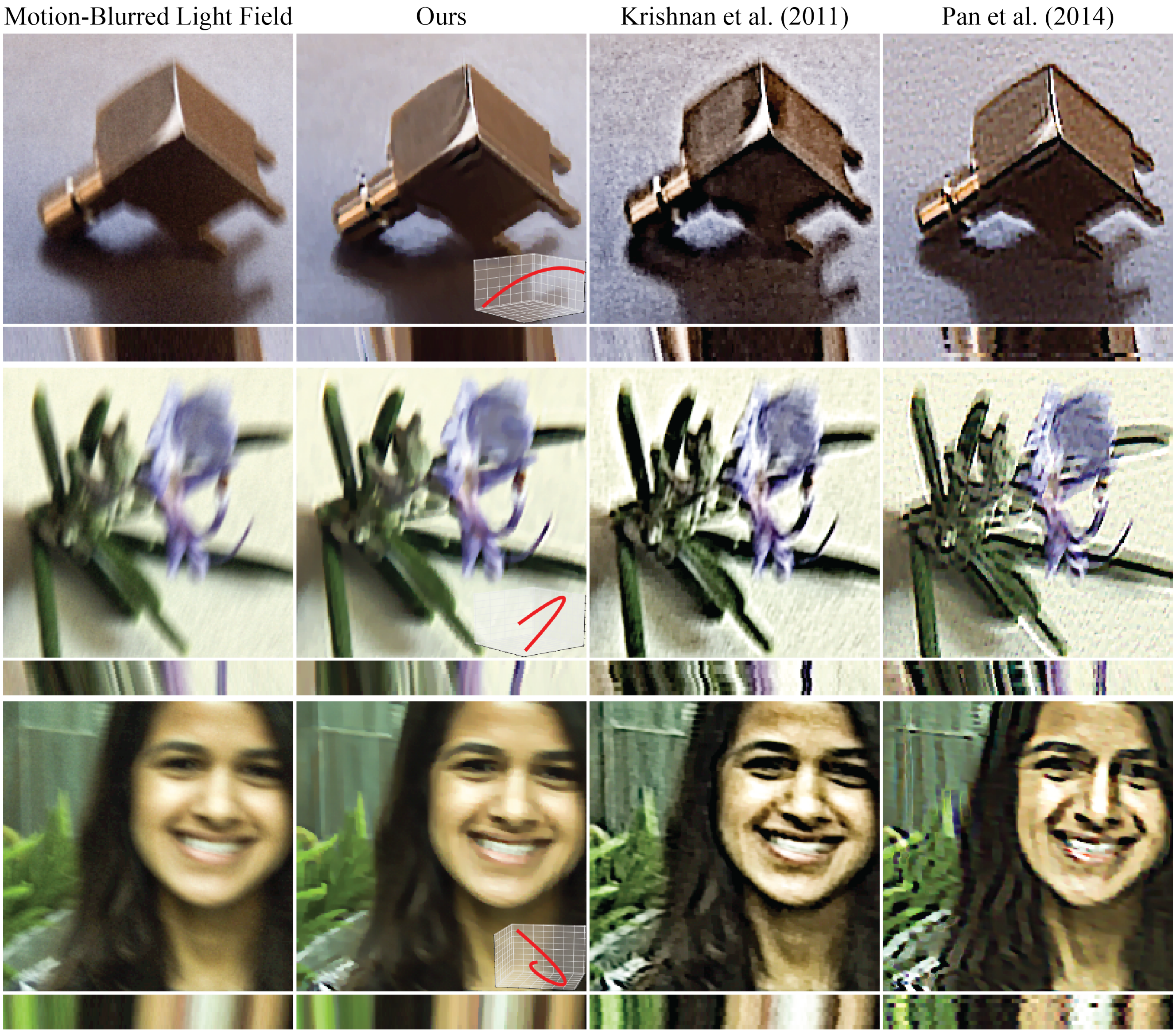

Figure 9. Blind deblurring results on real handheld motion-blurred light fields. Our algorithm is able to correctly recover the sharp light field and estimate the 3D camera motion path. Note that we correct the motion seen in the specular reflections and object edges in the circuit component example, the motion seen in the leaves and flower in the rosemary plant example, and the motion seen in the hair, eyebrows, teeth, and background plants of the portrait example. Furthermore, our method produces angularly-consistent results, as demonstrated by the epipolar slices of all 3 examples. Please view our supplementary video and project webpage for animated visualizations of our results.

5. Conclusion

In this work, we studied the problem of deblurring light fields of general scenes captured with 3D camera motion. We analyzed the effects of motion blur on the light field in the primal and Fourier domains, derived simple methods to deblur light fields in specific cases, and presented an algorithm to infer the sharp light field and camera motion path from real and synthetically-blurred light fields. It would be interesting to extend our forward model to account for 3D rotations of the optical axis, and theoretically analyze the effects of camera rotation on the motion-blurred light field. Since the forward model would be differentiable with respect to the rotation parameters, our blind deblurring optimization algorithm can easily generalize to account for camera rotation.

We think that the insights of this work enable future investigations of light field priors that more explicitly consider the effects of motion blur on the light field, as well as novel interpretations of single and multi-image motion deblurring as subsets of the general light field motion deblurring problem.

Acknowledgments This work was supported in part by ONR grant N00014152013, NSF grant 1617234, NSF graduate research fellowship DGE 1106400, a Google Research Award, the UC San Diego Center for Visual Computing, and a GPU donation from NVIDIA.

References

[1] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. 7

[2] E. H. Adelson and J. Y. A. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1992. 2

[3] Y. Bando, B.-Y. Chen, and T. Nishita. Motion deblurring from a single image using circular sensor motion. In *Computer Graphics Forum*, 2011. 2

[4] Y. Bando, H. Holtzman, and R. Raskar. Near-invariant blur for depth and 2D motion via time-varying light field analysis. In *ACM Transactions on Graphics*, 2013. 2

[5] R. N. Bracewell, K.-Y. Chang, A. K. Jha, and Y. Wang. Affine theorem for two-dimensional fourier transform. *Electronics Letters*, 1993. 3

[6] J.-X. Chai, X. Tong, S.-C. Chan, and H.-Y. Shum. Plenoptic sampling. In *SIGGRAPH*, 2000. 2, 5

[7] T. S. Cho, A. Levin, F. Durand, and W. T. Freeman. Motion blur removal with orthogonal parabolic exposures. In *ICCP*, 2010. 2

[8] D. G. Dansereau, A. Eriksson, and J. Leitner. Motion deblurring for light fields. In *arXiv:1606.04308*, 2016. 2

[9] D. G. Dansereau, O. Pizarro, and S. Williams. Linear volumetric focus for light field cameras. In *ACM Transactions on Graphics*, 2015. 2, 5

[10] M. Delbracio and G. Sapiro. Burst deblurring: removing camera shake through fourier burst accumulation. In *CVPR*, 2015. 2

[11] F. Durand, N. Holzschuch, C. Soler, E. Chan, and F. X. Sillion. A frequency analysis of light transport. In *ACM Transactions on Graphics*, 2005. 2, 5

[12] K. Egan, Y.-T. Tseng, N. Holzschuch, F. Durand, and R. Ramamoorthi. Frequency analysis and sheared reconstruction for rendering motion blur. In *ACM Transactions on Graphics*, 2009. 2

[13] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single image. In *ACM Transactions on Graphics*, 2006. 2

[14] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. Cohen. The lumigraph. In *SIGGRAPH*, 1996. 2

[15] M. Jin, P. Chandramouli, and P. Favaro. Bilayer blind deconvolution with the light field camera. In *ICCV Workshops*, 2015. 2

[16] D. Kingma and J. Ba. Adam: a method for stochastic optimization. In *ICLR*, 2015. 7

[17] T. Kobayashi, F. Sakaue, and J. Sato. Depth and arbitrary motion deblurring using integrated PSF. *ECCV Workshops*, 2014. 2

[18] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. *CVPR*, 2011. 2, 7

[19] W.-S. Lai, J.-B. Huang, Z. Hu, N. Ahuja, and M.-H. Yang. A comparative study for single image blind deblurring. *CVPR*, 2016. 2, 7

[20] A. Levin, P. Sand, T. S. Cho, F. Durand, and W. T. Freeman. Motion-invariant photography. In *ACM Transactions on Graphics*, 2008. 2

[21] A. Levin, Y. Wiess, F. Durand, and W. T. Freeman. Efficient marginal likelihood optimization in blind deconvolution. In *CVPR*, 2011. 2

[22] M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH*, 1996. 2

[23] C.-K. Liang and R. Ramamoorthi. A light transport framework for lenslet light field cameras. In *ACM Transactions on Graphics*, 2015. 2

[24] G. Lippmann. La photographie intégrale. In *Comptes-Rendus, Académie des Sciences*, 1908. 2

[25] R. Ng. Fourier slice photography. In *ACM Transactions on Graphics*, 2005. 2, 6

[26] R. Ng and P. Hanrahan. Digital correction of lens aberrations in light field photography. In *SPIE International Optical Design*, 2006. 2

[27] R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, and P. Hanrahan. Light field photographhy with a hand-held plenoptic camera. *CSTR 2005-02*, 2005. 2

[28] J. Pan, Z. Hu, and M.-H. Su, Z. Yang. Deblurring text images via L0-regularized intensity and gradient prior. *CVPR*, 2014. 2, 7

[29] R. Ramamoorthi, D. Mahajan, and P. Belhumeur. A first-order analysis of lighting, shading, and shadows. *ACM Transactions on Graphics*, 2007. 3

[30] R. Raskar, A. Agrawal, and J. Tumblin. Coded exposure photography: Motion deblurring using fluttered shutter. In *ACM Transactions on Graphics*, 2006. 2

[31] R. Raskar, A. Agrawal, C. Wilson, and A. Veeraraghavan. Glare aware photography: 4D ray sampling for reducing glare effects of camera lenses. In *ACM Transactions on Graphics*, 2008. 2

[32] P. P. Srinivasan, M. W. Tao, R. Ng, and R. Ramamoorthi. Oriented light-field windows for scene flow. In *ICCV*, 2015. 2

[33] M. W. Tao, P. P. Srinivasan, J. Malik, S. Rusinkiewicz, and R. Ramamoorthi. Depth from shading, defocus, and correspondence using light-field angular coherence. In *CVPR*, 2015. 2

[34] L. Xu, S. Zheng, and J. Jia. Unnatural L0 sparse representation for natural image deblurring. *CVPR*, 2013. 2, 7

[35] H. Zhang, D. Wipf, and Y. Zhang. Multi-image blind deblurring using a coupled adaptive sparse prior. In *CVPR*, 2013. 2

[36] X. Zhu, F. Sroubek, and P. Milanfar. Deconvolving PSFs for a better motion deblurring using multiple images. In *ECCV*, 2012. 2