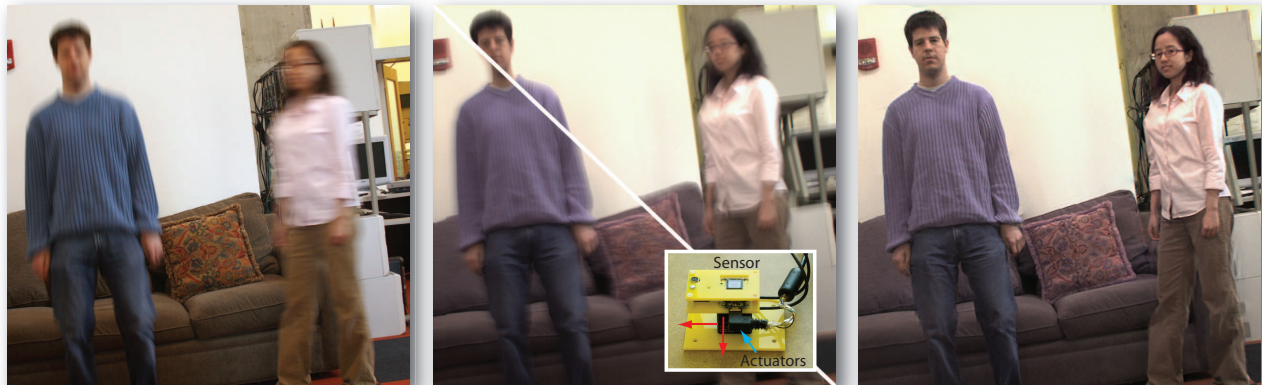# Motion blur removal with orthogonal parabolic exposures

Taeg Sang Cho[†], Anat Levin[‡], Frédo Durand[†], William T. Freeman[†]
[†] Massachusetts Institute of Technology, [‡] Weizmann Institute of Science

| Static camera image | Orthogonal parabolic camera: input | Orthogonal parabolic camera: deblurred output |

Abstract

Object movement during exposure generates blur. Removing blur is challenging because one has to estimate the motion blur, which can spatially vary over the image. Even if the motion is successfully identified, blur removal can be unstable because the blur kernel attenuates high frequency image contents. We address the problem of removing blur from objects moving at constant velocities in arbitrary 2D directions. Our solution captures two images of the scene with a parabolic motion in two orthogonal directions. We show that our strategy near-optimally preserves image content, and allows for stable blur inversion. Taking two images of a scene helps us estimate spatially varying object motions. We present a prototype camera and demonstrate successful motion deblurring on real motions.

1. Introduction

Motion blur can severely limit image quality, and while blur can be reduced using a shorter shutter speed, this comes with an unavoidable tradeoff of increased noise. One source of motion blur is camera shake. We can mitigate the camera shake blur by using a mechanical motion stabilization system or by placing the camera on a tripod. A second source of blur is an object movement in the scene. This type of blur is harder to control, and it is often desirable to remove it computationally using deconvolution.

Motion deblurring is challenging in two aspects. First, one needs to estimate the blur kernel, which depends on motion. Since objects in the scene can move independently, the blur kernel can vary over the image. While single-image based blur estimation techniques have been proposed [7, 8, 11–13, 19], they handle restricted motion types. Most recent blind deconvolution techniques [8, 19] rely on a strong assumption that the blur is spatially uniform over image. More robust motion estimation algorithms often involve multiple input images [5, 6, 18, 21], additional hardware [4, 14, 20], or user assistance [10].

A second challenge in deblurring is inverting the blur given the motion kernel. Typical motion blur kernels correspond to box filters in the motion direction. They attenuate high spatial frequencies and make the blur inversion ill-posed. One technique addressing this issue is the flutter-shutter camera [17]. By opening and closing the shutter during exposure, one can significantly reduce the high frequency image information loss. Levin *et al.* [14] propose a parabolic motion camera to minimize the information loss for 1D constant velocity motions, but the solution is invalid if 2D motion is present. Agrawal and Raskar [2] analyze the performance of the flutter-shutter camera and the parabolic camera and concludes that a flutter shutter camera performs better than a parabolic camera for a 2D constant velocity motion. Agrawal *et al.* [3] take multiple shots of a moving object, each with different exposures, and deconvolves the moving object using all the shots. This strategy is beneficial because the information lost in one of the shots is acquired by another. However, we show that their strategy does not offer guarantees on the worst-case performance.

We present an imaging technique that near optimally captures image information of objects moving at a constant velocity in 2D directions. We derive the optimal spectral bound for 2D constant velocity motions and introduce a new camera that captures two images using successive parabolic

motions in orthogonal directions. The joint spectrum of the two image captures approaches the 2D optimal spectral bound up to a constant multiplicative factor of $2^{-1.5}$, and it is the first known imaging technique to guarantee this bound. We recover a sharp image from the captured images using a multi-image deconvolution algorithm.

2. Sensor motion design and analysis

Consider an object moving at a constant velocity and let $s_{x,y} = [s_x, s_y]$ be its 2D velocity vector. Suppose we capture J images $B^1, ..B^J$ of this object using J translating cameras. Locally, the blur process is a convolution:

$$B^j = \phi^j_{s_{x,y}} \otimes I + n^j \qquad (1)$$

where I is an ideal sharp image, n^j imaging noise, and $\phi^j_{s_{x,y}}$ the blur kernel (point spread function, PSF). $\phi^j_{s_{x,y}}$ depends on the motion between the sensor and the scene. The convolution is a multiplication in the frequency domain:

$$\hat{B}^j(\omega_{x,y}) = \hat{\phi}^j_{s_{x,y}}(\omega_{x,y})\hat{I}(\omega_{x,y}) + \hat{n}^j(\omega_{x,y}) \qquad (2)$$

where $\omega_{x,y} = [\omega_x, \omega_y]$ is a 2D spatial frequency, and the ˆ denotes the Fourier transform. Eq 2 shows that when $\hat{\phi}(\omega_{x,y})$ is small, the signal-to-noise ratio drops. One can show that the success in deblurring depends on the spectral power of the blur kernels $\|\hat{\phi}^j_{s_{x,y}}(\omega_{x,y})\|^2$ by examining the expected reconstruction error, which can be computed in a closed form given a Gaussian prior on gradients [9]. The reconstruction quality is inversely related to the summed spectra:

$$\|\tilde{\hat{\phi}}_{s_{x,y}}(\omega_{x,y})\|^2 = \sum_j \|\hat{\phi}^j_{s_{x,y}}(\omega_{x,y})\|^2 \qquad (3)$$

The goal of the camera motion design is as follows:

Find a set of J camera motions that maximizes the summed power spectrum $\|\tilde{\hat{\phi}}_{s_{x,y}}(\omega_{x,y})\|^2$ for every spatial frequency $\omega_{x,y}$ and every motion vector $\|s_{x,y}\| < S_{obj}$.

2.1. Background on motion blur in the space-time volume

We represent light received by the sensor as a 3D x, y, t space-time volume $L(x, y, t)$. That is, $L(x, y, t)$ denotes the color of the light ray hitting the x, y coordinate of a static detector at time instance t. A static camera forms an image by integrating the light rays in the space-time volume over a finite exposure time T:

$$B(x, y) = \int_{-\frac{T}{2}}^{\frac{T}{2}} L(x, y, t)dt \qquad (4)$$

Assume the camera is translating during exposure on the xy plane, and let $\mathbf{f}$ be its displacement path:

$$\{\mathbf{f} : [x, y, t] = [f_x(t), f_y(t), t]\} \qquad (5)$$

Rays hitting the detector are shifted, and the recorded image is

$$B(x, y) = \int_{-\frac{T}{2}}^{\frac{T}{2}} L(x + f_x(t), y + f_y(t), t)dt + n \qquad (6)$$

where n is an imaging noise. We can represent the integration curve $\mathbf{f}$ as a 3D integration kernel k:

$$k(x, y, t) = \delta(x - f_x(t)) \cdot \delta(y - f_y(t)) \qquad (7)$$

where δ is a delta function. We show the integration kernel k for several camera motions in the first row of Figure 1.

If the object motion is locally constant, we can express the integrated image as a convolution of a sharp image at one time instance with a point spread function $\phi_{s_{x,y}}$. The PSF $\phi_{s_{x,y}}$ of a constant velocity motion $s_{x,y} = [s_x, s_y]$ is a sheared projection of the 3D integration kernel k [14]:

$$\phi_{s_{x,y}}(x, y) = \int_t k(x - s_x t, y - s_y t, t)dt \qquad (8)$$

Some PSFs of different integration kernels are shown in the second row of Figure 1.

The Fourier transform $\hat{\phi}_{s_{x,y}}$ of the PSF $\phi_{s_{x,y}}$ is a slice from the Fourier transform $\hat{k}$ of the integration kernel k [14, 15]:

$$\hat{\phi}_{s_{x,y}}(\omega_x, \omega_y) = \hat{k}(\omega_x, \omega_y, s_x\omega_x + s_y\omega_y) \qquad (9)$$

The Fourier transform $\hat{\phi}_{s_{x,y}}$ for different integration kernels k are shown in the bottom row of Figure 1. 2D Fourier slices corresponding to all motion directions $\|s_{x,y}\| < S_{obj}$ lie in the complementary volume of an inverted double cone. Therefore, $\hat{k}$ occupies this volume. We refer to this volume as *the wedge of revolution*, defined as the set:

$$C \equiv \{(\omega_x, \omega_y, \omega_t) | \omega_t < S_{obj}\|\omega_{x,y}\|\} \qquad (10)$$

This relationship holds since the Fourier transform of a PSF is a slice from $\hat{k}$ at $\omega_t = s_x\omega_x + s_y\omega_y$, and if $\|s_{x,y}\| \leq S_{obj}$, $s_x\omega_x + s_y\omega_y \leq S_{obj}\|\omega_{x,y}\|$.

The optimal spectral bound We extend the spectral bound for 1D linear motions in [14] to 2D linear motions, and show that spectral power in $\hat{k}$ cannot become arbitrarily high. Suppose we capture J images and let $\|\tilde{\hat{k}}\|^2$ be the joint motion spectrum $\|\tilde{\hat{k}}(\omega_x, \omega_y, \omega_t)\|^2 = \sum_j \|\hat{k}^j(\omega_x, \omega_y, \omega_t)\|^2$. We can derive an upper bound on the worst-case joint spectrum $\|\tilde{\hat{k}}\|^2$. The amount of energy collected by the camera within a fixed exposure time T is bounded. Levin *et al.* [14] use the Parseval theorem to show that the collected energy is preserved in the frequency domain and as a result, the norm of every ω_{x_0, y_0} slice of $\tilde{\hat{k}}$ (i.e. $\tilde{\hat{k}}(\omega_{x_0}, \omega_{y_0}, \omega_t)$) is bounded:

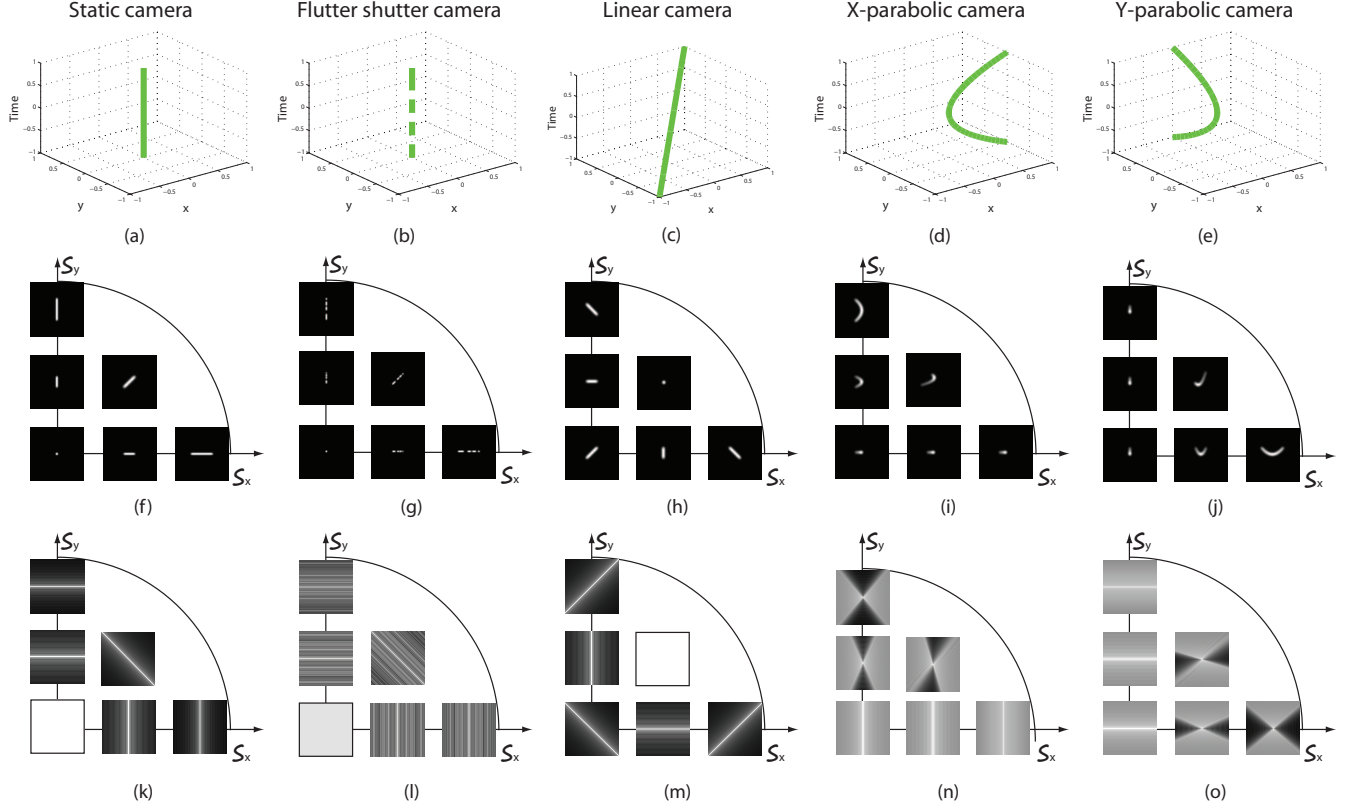$$\int \|\tilde{\hat{k}}(\omega_{x_0}, \omega_{y_0}, \omega_t)\|^2 d\omega_t \leq T \qquad (11)$$

Figure 1: *The integration curves k (a-e), the point spread functions $\phi_{s_{x,y}}$ (f-j) and their log-power spectra (k-o) for a few cameras. In (f-o), the outer axes correspond to x,y directional speed. In (f-j), the inner axes correspond to x,y, and in the spectra plots (k-o), the inner axes correspond to ω_x, ω_y. All spectra plots are normalized to the same scale.*

Every ω_{x_0,y_0}-slice intersects the wedge of revolution for a segment of length $2S_{obj}\|\omega_{x_0,y_0}\|$. An optimal camera should spread the captured energy equally in this intersection to maximize the worst-case spectral value. Therefore:

$$\min_{\omega_t} \|\hat{\tilde{k}}(\omega_{x_0}, \omega_{y_0}, \omega_t)\|^2 \le \frac{T}{2S_{obj}\|\omega_{x_0,y_0}\|} . \quad (12)$$

Since the PSFs spectra $\hat{\phi}_{s_{x,y}}^j$ are slices through $\hat{k}^j$, this bound also applies for the PSFs' spectral power:

$$\min_{s_{x,y}} \|\hat{\tilde{\phi}}_{s_{x,y}}(\omega_{x_0}, \omega_{y_0})\|^2 \le \frac{T}{2S_{obj}\|\omega_{x_0,y_0}\|} . \quad (13)$$

2.2. Orthogonal parabolic motions

We seek a motion path whose spectrum covers the wedge of revolution and approaches the bound in Eq 12. Our solution captures two images with two orthogonal parabolic motions. We show that the orthogonal parabolic camera captures the desired spectrum with the worst-case spectral power of at least a factor $2^{-1.5}$ of the upper bound.

2.2.1 Camera motion

Let k_1, k_2 be the 3D integration kernels of x and y parabolic camera motions. The kernels are defined by the integration curves f_1, f_2:

$$\begin{aligned} f_1(t) &= [a_x(t+T/4)^2, 0, t], \quad t = [-T/2...0] \\ f_2(t) &= [0, a_y(t-T/4)^2, t], \quad t = [0...T/2] \end{aligned} \quad (14)$$

At time t, the derivative of the x-parabolic camera's movement is $2a_x(t-T/4)$, and the camera is essentially tracking an object with velocity $2a_x(t-T/4)$ along the x axis. For a reason to be clarified below, we set

$$a_x = a_y = \frac{2\sqrt{2}S_{obj}}{T} \quad (15)$$

The maximal sensor velocity becomes $S_{sens} = \sqrt{2}S_{obj}$. Figure 1(i-j) show PSFs of different motions captured by the orthogonal parabolic camera. The PSFs are truncated and sheared parabolas that depend on the object speed.

2.2.2 Optimality

The spectrum of an x-parabolic motion is approximately a double wedge in the 2D ω_x, ω_t frequency space [14]. Since
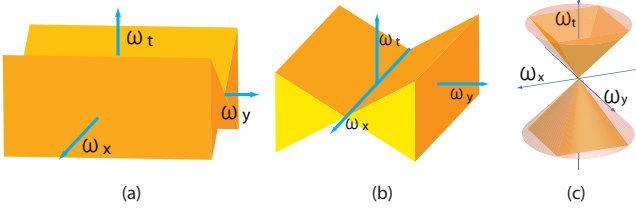
Figure 2: *(a) The spectrum $\hat{k}_1$ captured by a x-parabolic camera. (b) The spectrum $\hat{k}_2$ captured by a y-parabolic camera. (c) The sum of $\hat{k}_1$ and $\hat{k}_2$ approximates the wedge of revolution.*

an x-parabolic motion k_1 is Dirac delta along the y axis, the 3D spectrum $\|\hat{k}_1\|^2$ is constant along the ω_y axis and $\|\hat{k}_1\|^2$ spreads energy in a 3D double wedge (Figure 2(a)). The y-parabolic motion spreads energy on the orthogonal 3D double wedge (Figure 2(b)). Mathematically speaking,

$$\|\hat{k}_1(\omega_x, \omega_y, \omega_t)\|^2 \approx \frac{T}{4S_{sens}\|\omega_x\|}H(S_{sens}\|\omega_x\| - \|\omega_t\|)$$

$$\|\hat{k}_2(\omega_x, \omega_y, \omega_t)\|^2 \approx \frac{T}{4S_{sens}\|\omega_y\|}H(S_{sens}\|\omega_y\| - \|\omega_t\|)$$

(16)

where $H(\cdot)$ is a Heaviside step function.

The 2D PSF spectra are slices from the 3D double wedge spectra of $\|\hat{k}_j\|^2$. Figure 1 (n-o) show the log-spectrum of PSFs $\hat{\phi}_s^j$ for parabolic exposures as we sweep the object velocity. For x-directional motions ($s_y = 0$), the x-parabolic camera covers all spatial frequencies without zeros. On the other hand, as y-directional motion increases, the x-parabolic camera fails to capture frequencies near the ω_y axis. The y-parabolic camera, however, covers the frequencies missed by the x-parabolic camera, thus the *sum* of these two spectra eliminates zeros in all the spatial frequencies. Therefore, by taking two images of a scene using orthogonal parabolic cameras, we can stably invert the blur for all 2D object motions.

Figure 2(c) visualizes the joint spectrum covered by the orthogonal parabolic motions, suggesting that the sum of orthogonal 3D wedges is an approximation to the wedge of revolution that we aim to capture. We can show that if the maximal sensor speed S_{sens} is set to $\sqrt{2}S_{obj}$, the sum of double wedges subsumes the wedge of revolution.

Claim 1 *Let S_{sens} be the maximum sensor speed of the parabolic camera, and S_{obj} the maximum object speed in image space. If $S_{sens} \geq \sqrt{2}S_{obj}$, the joint motion spectrum $\|\tilde{k}\|^2$ of an orthogonal parabolic camera subsumes the wedge of revolution. When $S_{sens} = \sqrt{2}S_{obj}$, the worst-case spectral power of an orthogonal parabolic camera, at any frequency, is at least $\frac{1}{2\sqrt{2}}$ of the optimal bound.*

Proof: The joint motion spectrum of the orthogonal parabolic camera is non-zero in the set $\{(\omega_x, \omega_y, \omega_t)|\omega_t \leq S_{sens}\max(\|\omega_x\|, \|\omega_y\|)\}$. If $(\omega_x, \omega_y, \omega_t)$ lies in the wedge of revolution, then $\omega_t \leq S_{obj}\|\omega_{x,y}\|$. Since $\|\omega_{x,y}\|^2 \leq 2\max(\|\omega_x\|^2, \|\omega_y\|^2)$,

$$\begin{aligned} \omega_t &\leq S_{obj}\|\omega_{x,y}\| \\ &\leq \sqrt{2}S_{obj}\max(\|\omega_x\|, \|\omega_y\|) \\ &\leq S_{sens}\max(\|\omega_x\|, \|\omega_y\|) \end{aligned}$$

(17)

In other words, the joint motion spectrum of the orthogonal parabolic cameras subsumes the wedge of revolution.

In the joint motion spectrum, the spectral content at $(\omega_x, \omega_y, \omega_t)$ is at least $\min\left(\frac{T}{4S_{sens}\|\omega_x\|} + \frac{T}{4S_{sens}\|\omega_y\|}\right)$. Since $\|\omega_{x,y}\| \geq max(\|\omega_x\|, \|\omega_y\|)$,

$$\begin{aligned} \min\left(\frac{T}{4S_{sens}\|\omega_x\|} + \frac{T}{4S_{sens}\|\omega_y\|}\right) &\geq \frac{T}{4S_{sens}\|\omega_{x,y}\|} \\ &= \frac{T}{4\sqrt{2}S_{obj}\|\omega_{x,y}\|} \end{aligned}$$

(18)

The minimum spectral content of the orthogonal parabolic camera is at least $2^{-1.5}$ of the optimum. ∎

2.3. Discussion of other cameras

A static camera: The integration curve of a static camera (Figure 1 first column) is $k^s(t) = [0, 0, t]$, $t \in [-T/2...T/2]$. The power spectrum is constant along ω_x and ω_y:

$$\|\hat{k}^s(\omega_x, \omega_y, \omega_t)\|^2 = T^2 sinc^2(\omega_t T)$$

(19)

The Fourier transform of the PSF is a slice of the motion spectrum $\hat{k}$, and is a sinc whose width depends on the object velocity $\|\hat{\phi}_{s_{x,y}}^s\|^2 = T^2 sinc^2((s_x\omega_x + s_y\omega_y)T)$. For fast object motions, this sinc attenuates high frequencies. Similarly, by linearly moving the camera during exposure (Figure 1(c)), we can track the object that moves at the camera's speed, but objects whose velocity is different from the camera's velocity still suffer from the sinc fall-off.

A flutter shutter camera: In a flutter shutter camera [17] (Figure 1 second column), the motion spectrum k^f is constant along ω_x, ω_y and is modulated along ω_t: $\|\hat{k}^f(\omega_x, \omega_y, \omega_t)\|^2 = \|\hat{m}(\omega_t)\|^2$, where $\hat{m}$ is the Fourier transform of the shutter code. We can design the code to be more broadband than that of a static camera. Yet, the spectrum is constant along ω_x, ω_y, thus $\min_s \|\hat{\phi}_s^f(\omega_x, \omega_y)\|^2 \leq T/(2S_{obj}\Omega)$ for all (ω_x, ω_y) [14], where Ω is the spatial bandwidth of the camera. As a result, at low-to-mid frequencies the spectral power does not reach the upper bound.

Two shots: Taking two images with a static camera, a linearly moving camera, or a flutter shutter camera can im-

prove the kernel estimation accuracy, but it does not substantially change the spectral coverage. Optimizing the exposure lengths of each shot [3], and in the case of a flutter shutter camera also optimizing the random codes in each shot, do not eliminate their fundamental limitations: their power spectra are constant along $\omega_{x,y}$ and hence spend the energy budget outside the wedge of revolution.

Synthetic simulation: We compare the deblurring performance of a pair of static cameras, a pair of flutter shutter cameras, a single parabolic camera and an orthogonal parabolic camera through synthetic experiments. The orthogonal parabolic camera is designed to deblur 2D constant velocity motions with speed less than S_{obj}. The deblurring performance is compared for motions within the velocity range of interest. To be more in favor of previous solutions, we have optimized their parameters for each motion independently. For a pair of static camera, we use the optimal split of the exposure time T into two shots, optimized for *each* object motion independently. For a pair of flutter shutter camera, we use the optimal split of the exposure time T and the optimal combination of codes, optimized for *each* object motion independently. In a realistic scenario we cannot optimize the split of the exposure time T or the codes because the object motion is not known a priori.

We render images of a moving object seen by these cameras. We add zero-mean Gaussian noise with standard deviation $\eta = 0.01$ to the images. We deblur the images with the known blur kernels using Wiener deconvolution. In all experiments, we fix the total exposure time T.

Figure 3 shows the deconvolution results and its peak signal-to-noise ratio (PSNR) for different object velocities. Each row corresponds to a different object velocity. When the object is static, a pair of static camera generates visually the most pleasing image. For moving objects, however, a pair of orthogonal parabolic camera generates visually the most pleasing image. This visual result agrees with the theoretical prediction: the deconvolution quality is better when the spectral power of the PSF is higher.

We put the synthetic experiment results in the context of previous blur removal techniques. The performance of previous two-image motion deblurring techniques, such as [5, 6, 18, 21], can be approximated by the deconvolution result of the static camera pair in Figure 3. Even if these solutions correctly estimate the motion kernels, inverting the kernel is still hard since high frequencies are attenuated. Blind motion deblurring solutions, such as [8, 19], attempt to solve an even harder problem, since they try to estimate the blur kernel from a single image. Yet they only address the problem of kernel estimation and do not optimize the deconvolution quality given the correct kernel.

3. Image reconstruction

Subject motions cause spatially variant blur that should be estimated and removed locally. We adapt the Bayesian framework for image deconvolution and kernel estimation to locally estimate the motion blur within a small window. We employ a multi-scale technique to reduce the computational cost.

3.1. Non-blind deconvolution

Let $\bar{B}, \bar{\phi}$ be $\bar{B} = [B^1, B^2], \bar{\phi} = [\phi^1, \phi^2]$. We recover the blur-free image by maximizing the posterior probability $\tilde{I} = \arg\max p(I|\bar{B}, \bar{\phi})$. Using Bayes rule,

$$p(I|\bar{B}, \bar{\phi}) \quad \propto \quad p(I, \bar{B}|\bar{\phi}) = p(I) \prod_{j=1}^{2} p(B^j|\phi^j, I) \quad (20)$$

$$-\log p(B^j|\phi^j, I) \quad = \quad |B^j - \phi^j \otimes I|^2/\eta^2 + C_1 \quad (21)$$

$$-\log p(I) \quad = \quad \beta \sum_i \rho(|g_{x,i}(I)|) + \rho(|g_{y,i}(I)|) + C_2 \,(22)$$

where C_1, C_2 are constants, $g_{x,i}, g_{y,i}$ are x, y gradient operators at pixel i, $\beta = 0.002$ determines the variance of the gradient profile, and $\rho(z) = z^\alpha$ is a robust norm. When $\alpha = 2$, we impose a Gaussian prior on image gradients, and when $\alpha \leq 1$, we impose a sparse prior. When $\alpha = 2$, we can efficiently deconvolve the image in the frequency domain using the Wiener filter (e.g. [9]). We use a Gaussian prior for kernel estimation, and a sparse prior for deconvolution.

Eq 20 is a joint deconvolution model, stating that we seek an image $\tilde{I}$ *fitting the convolution constraints of both B^1 and B^2*. The deconvolved image $\tilde{I}$ should be able to regenerate the input images B^1 and B^2 using the kernel pair that generated $\tilde{I}$. Rav-Acha and Peleg [18] essentially deblurs two input images by maximizing the likelihood term (Eq 21), and Chen *et al*. [5] and Agrawal *et al*. [3] augment it with the prior term (Eq 22).

3.2. Kernel estimation

A critical step in motion deblurring is estimating the correct kernel pair $\bar{\phi}$. For that we seek:

$$\bar{\phi} = \arg\max p(\bar{\phi}|\bar{B}) = \arg\max p(\bar{B}|\bar{\phi}) p(\bar{\phi}) \quad (23)$$

where $p(\bar{\phi})$ is a prior on motion kernels (uniform in this work) and $p(\bar{B}|\bar{\phi})$ is obtained by marginalizing over all latent images I, $p(\bar{B}|\bar{\phi}) = \int p(\bar{B}, I|\bar{\phi}) dI$, where $p(\bar{B}, I|\bar{\phi})$ is given by Eq 21,22. If the prior $p(I)$ is Gaussian, $p(\bar{B}|\bar{\phi})$ is Gaussian as well and we can derive it in a closed form.

Alternatively, we can solve for the latent image I using the Wiener filter (Eq 20) and express $p(\bar{B}|\bar{\phi})$ as follows:

$$\log p(\bar{B}|\bar{\phi}) = \log p(\tilde{I}, \bar{B}|\bar{\phi}) + \tilde{\Psi} + C_4 \quad (24)$$
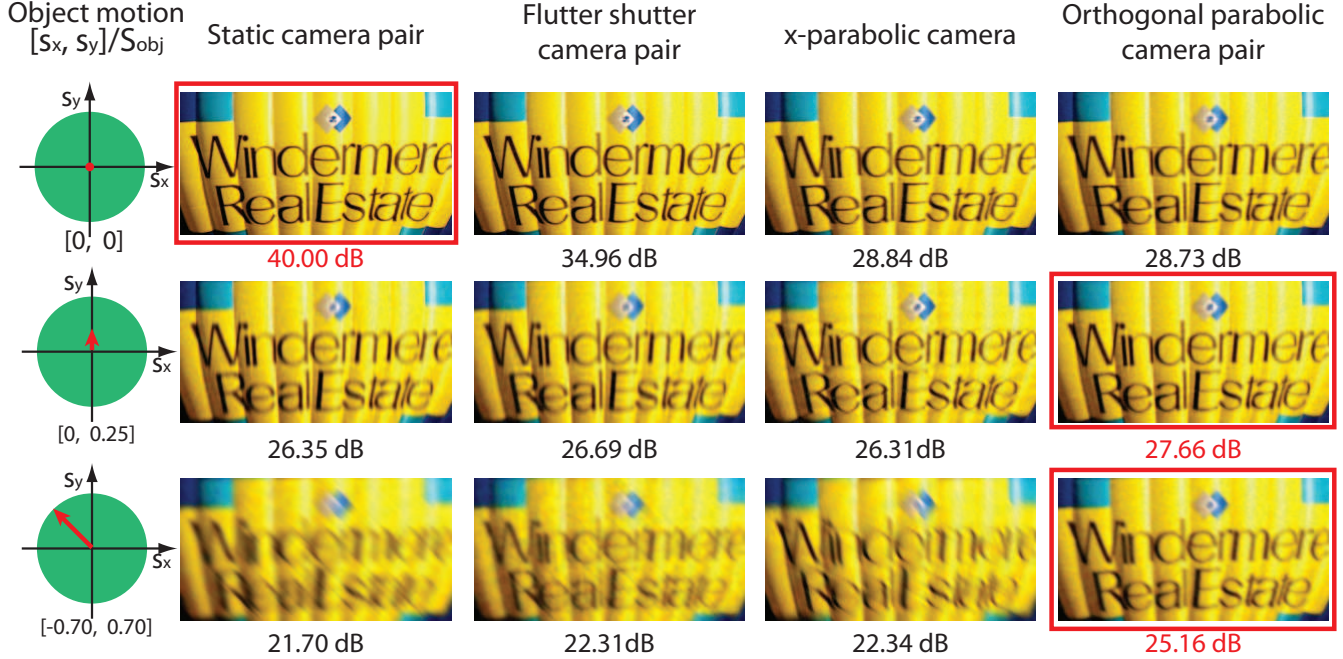
Figure 3: *Synthetic visualizations of the reconstruction quality. We optimized the exposure lengths of each camera. First column: The object motion during the exposure. The green disc denotes the velocity range covered by the orthogonal parabolic camera, and the red arrow denotes the object velocity. Other columns show the Wiener deconvolution results using the true PSFs. The orthogonal parabolic camera outperforms other optimized solutions in deblurring moving objects.*

where C_4 is a constant, $\tilde{\Psi} = \Sigma_\omega \log \Psi_\omega$, and $\Psi_\omega = \frac{1}{\eta^2} \Sigma_j \|\hat{\phi}_\omega^j\|^2 + \sigma_\omega^{-2}$ is the variance of $p(\hat{\bar{B}}_\omega | \hat{\bar{\phi}}_\omega)$. This expression is more useful because it allows us to compute $p(\bar{B}|\bar{\phi})$ in local image windows.

We estimate $\bar{\phi}$ by evaluating the log likelihood Eq 24 on a set of PSF pairs that correspond to discretized 2D linear motions, and choosing the pair with the highest value.

Local kernel estimation: If there are multiple motions in the scene, we need to locally estimate the blur. Let $\tilde{I}_s$ be images generated by deconvolving $\bar{B}$ with motion kernels $\bar{\phi}_s$, and let $\tilde{B}_s = \phi_s^j \otimes \tilde{I}_s$ be the reconvolved image. Using Eq 24 , we can approximate the score $p(\bar{B}|\bar{\phi}_s)$ locally:

$$\log p(\bar{B}(i)|\bar{\phi}_s) \approx -\frac{1}{\eta^2} \sum_j \sum_{k \in \mathbb{N}(i)} |B^j(k) - \tilde{B}_s^j(k)|^2 \quad (25)$$
$$- \rho(g_{x,i}(\tilde{I}_s)) - \rho(g_{y,i}(\tilde{I}_s)) + \frac{1}{N}\tilde{\Psi}$$

where $N = 15 \times 15$ is the window size and $\mathbb{N}(i)$ is the window around the pixel i.

Handling motion boundaries : There are regions next to motion boundaries that are visible in one image but not in the other. The observation model (Eq 21) is inconsistent in such regions and the joint deconvolution leads to artifacts. We use the image deblurred using only one of the two input images to fill in the motion boundary. We automatically

detect the motion boundary by also considering kernel candidates with a single image observation (i.e. $B^2 = 0, \phi^2 = 0$ in the log-likelihood (Eq 21)). We add a fixed penalty (set to 0.15 for all experiments) to using a single image solution. Otherwise, the log-likelihood (Eq 25) always favors a single image solution. Note that the high frequency content may not be well maintained in such regions.

Multi-scale PSF estimation : The blur-free image quality depends on how finely we sample the 2D linear motions. We resort to a coarse-to-fine strategy to search over linear motions. We discretize 2D linear motions into 4500 samples at the finest resolution. We down-sample input images $\bar{B}$ by a factor of 4 to reduce the number of pixels and the motion search space: blur kernels from adjacent velocity samples look identical in a down-sampled image. At the coarsest scale, we search $2 \times 4500/(4^2)$ velocity samples (single-image explanations incur the factor 2) for the kernel estimate. We propagate the estimates to a finer resolution to refine the estimates. At each spatial scale, we regularize the estimate using a Markov random field. Each node corresponds to a pixel, and the states at each node are the kernel pairs, with the local evidence Eq 25. The potential between nodes is designed to favor the same states in the neighboring node, and favor the state transition where the image gradient between the deblurred images is small [1, 13].

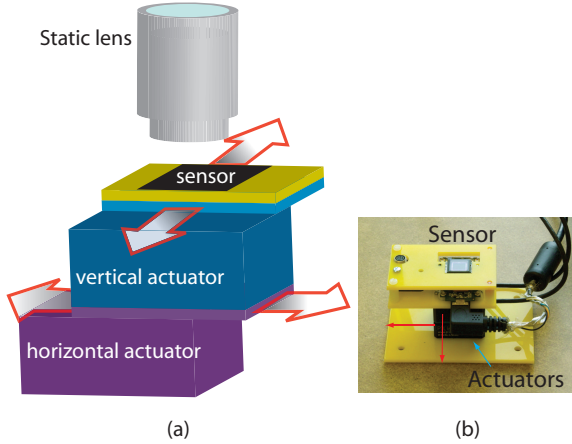We use the regularized kernel map to reconstruct the sharp

Figure 4: *(a) A diagram of our prototype. (b) A photograph of the actuators and the sensor.*

image $\tilde{I}$. We deconvolve input images $\bar{B}$ with all possible kernels $\bar{\phi}_{s_{x,y}}$ and generate a set of deconvolved images $\tilde{I}_{s_{x,y}}$. We reconstruct $\tilde{I}$ from $\tilde{I}_{s_{x,y}}$ by selecting the pixel value from image deblurred with the estimated kernel at each pixel. We blend different layers using the Poisson blending method [16] to reduce artifacts at abutting motion layers.

4. Experiments

4.1. Prototype camera

We built a prototype camera, different from Levin *et al.* [14], consisting of a sensor, two motion stages and their controllers. We mounted a light-weight camera sensor on two motion stages, where each can move the camera sensor along orthogonal axes (See Figure 4(a)). In each image capture, one of the motion stages undergoes parabolic motion, approximated with 19 segments of constant velocity due to control constraints. In practice, we could replace the motion stages with the image stabilization hardware. The camera lens is affixed to the camera lid, and does not move during exposure. The total exposure time for taking two images is 500ms: 200ms for each image, with a delay of 100ms between exposures. We incur a 100ms delay for switching the control from one motion stage to another, which can be reduced by using an improved hardware.

4.2. Results

Figure 5 illustrates the deblurring pipeline. First, we capture two images with the detector undergoing a parabolic motion in orthogonal directions. From the two images, we estimate a motion map, shown colored using the velocity coding scheme of the inset. We use the motion map to reconstruct the image. For reference we show an image taken with a static camera with 500ms exposure, synchronized to the first shot of the orthogonal parabolic camera. The refer-

Figure 6: *Images taken with a synchronized static camera and deblurred images from the orthogonal parabolic camera. Images from a static camera with 500ms exposure are shown for reference. Arrows on reference images show the direction and magnitude of motion.*

ence image reveals the object motion during the orthogonal parabolic camera's image capture.

We present more deblurring results on natural motions in Figure 6, using parabolic exposure to capture the motions in generic, non-horizontal directions. Images from the static camera (500ms exposure) reveal the motions, shown by red arrows. Some artifacts can be seen at motion boundaries but in general the reconstructions are visually plausible. In the second column of Figure 6, we show a deblurring result for a perspective motion blur. While the perspective motion does not conform to the constant object velocity motion model, our system still recovers a reasonably sharp image.

5. Discussions and conclusions

We present a two-exposure solution to removing spatially variant 2D constant velocity motion blur. We show that the union of PSFs corresponding to 2D linear motions occupy a wedge of revolution in Fourier domain, and that the orthogonal parabolic motion paths approach the optimal bound up to a multiplicative constant.

We assume that objects move at a constant velocity within the exposure time, which is a limitation shared by most previous work that deals with object motion. Camera shake, which typically exhibits complex kernels, needs to be handled separately. Our camera captures image information al-
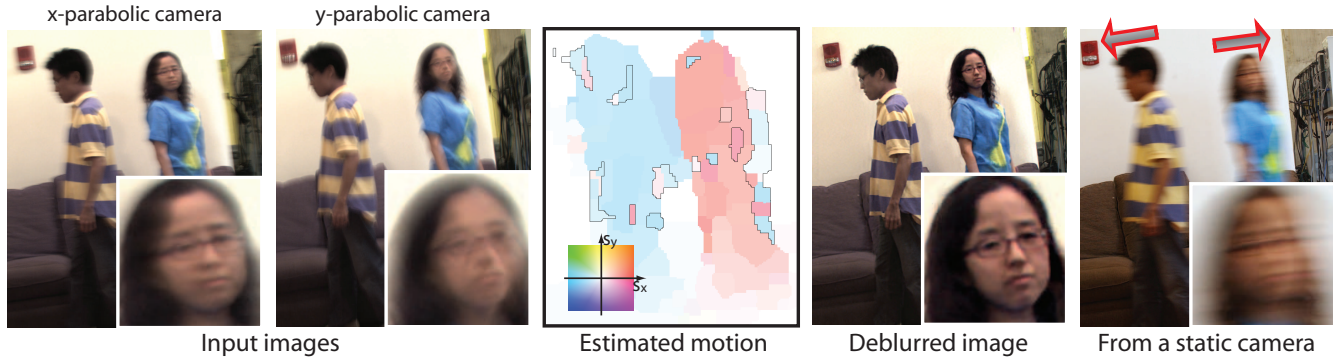
Figure 5: *The deblurring process pipeline: two images taken with the orthogonal parabolic cameras are used to locally estimate the motion. The motion estimate is presented with the color coding scheme in the inset, and pixels taken from images deblurred with a single input image are within black bounding boxes. The image pair is deconvolved using the estimated motion map. The image taken with a synchronized static camera with 500ms exposure is shown for reference.*

most optimally, but does not provide guarantees for the kernel estimation performance. While taking two images certainly helps the kernel estimation, designing a sensor motion that optimizes both kernel estimation and information capture is an open problem. Our image reconstruction takes into account occlusions by allowing some pixels to be reconstructed from a single image, but a full treatment of occlusion for deconvolution remains an open challenge. Our solution uses two exposures in order to cover the full velocity range while minimizing the number of shots to reduce the time overhead and additive noise penalty. The comprehensive study of solutions relying on an arbitrary number of exposures is, however, an important open question which requires careful modeling of the noise characteristics and the per-shot time overhead.

Acknowledgments

References

[1] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. *ACM TOG (SIGGRAPH)*, 2004.

[2] A. Agrawal and R. Raskar. Optimal single image capture for motion deblurring. In *IEEE CVPR*, 2009.

[3] A. Agrawal, Y. Xu, and R. Raskar. Invertible motion blur in video. *ACM TOG (SIGGRAPH)*, 2009.

[4] M. Ben-Ezra and S. K. Nayar. Motion-based motion deblurring. *IEEE TPAMI*, 26:689 – 698, 2004.

[5] J. Chen, L. Yuan, C.-K. Tang, and L. Quan. Robust dual motion deblurring. In *IEEE CVPR*, 2008.

[6] S. Cho, Y. Matsushita, and S. Lee. Removing non-uniform motion blur from images. In *IEEE ICCV*, 2007.

[7] S. Dai and Y. Wu. Motion from blur. In *IEEE CVPR*, 2008.

[8] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single image. *ACM TOG (SIGGRAPH)*, 2006.

[9] S. W. Hasinoff, K. N. Kutulakos, F. Durand, and W. T. Freeman. Time-constrained photography. In *IEEE ICCV*, 2009.

[10] J. Jia. Single image motion deblurring using transparency. In *IEEE CVPR*, 2007.

[11] N. Joshi, R. Szeliski, and D. J. Kriegman. PSF estimation using sharp edge prediction. In *IEEE CVPR*, 2008.

[12] D. Kundur and D. Hatzinakos. Blind image deconvolution. *IEEE signal processing magazine*, pages 43 – 64, 1996.

[13] A. Levin. Blind motion deblurring using image statistics. In *NIPS*, 2006.

[14] A. Levin, P. Sand, T. S. Cho, F. Durand, and W. T. Freeman. Motion-invariant photography. *ACM TOG (SIGGRAPH)*, 2008.

[15] R. Ng. Fourier slice photography. *ACM TOG (SIGGRAPH)*, 2005.

[16] P. Perez, M. Gangnet, and A. Blake. Poisson image editing. *ACM TOG (SIGGRAPH)*, 2003.

[17] R. Raskar, A. Agrawal, and J. Tumblin. Coded exposure photography: Motion deblurring using fluttered shutter. *ACM TOG (SIGGRAPH)*, 2006.

[18] A. Rav-Acha and S. Peleg. Two motion-blurred images are better than one. *PRL*, 26:311 – 317, 2005.

[19] Q. Shan, L. J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. *ACM TOG (SIGGRAPH)*, 2008.

[20] Y.-W. Tai, H. Du, M. S. Brown, and S. Lin. Image/video deblurring using a hybrid camera. In *IEEE CVPR*, 2008.

[21] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum. Image deblurring with blurred/noisy image pairs. *ACM TOG (SIGGRAPH)*, 2007.