

Analysis of Motion Blur With a Flutter Shutter Camera for Non-linear Motion

Yuanyuan Ding¹, Scott McCloskey², and Jingyi Yu¹

¹ University of Delaware, Newark, DE, USA

² Honeywell Labs, Golden Valley, MN, USA

Abstract. Motion blurs confound many computer vision problems. The fluttered shutter (FS) camera [1] tackles the motion deblurring problem by emulating invertible broadband blur kernels. However, existing FS methods assume known constant velocity motions, e.g., via user specifications. In this paper, we extend the FS technique to general 1D motions and develop an automatic motion-from-blur framework by analyzing the image statistics under the FS.

We first introduce a fluttered-shutter point-spread-function (FS-PSF) to uniformly model the blur kernel under general motions. We show that many commonly used motions have closed-form FS-PSFs. To recover the FS-PSF from the blurred image, we present a new method by analyzing image power spectrum statistics. We show that the Modulation Transfer Function of the 1D FS-PSF is statistically correlated to the blurred image power spectrum along the motion direction. We then recover the FS-PSF by finding the motion parameters that maximize the correlation. We demonstrate our techniques on a variety of motions including constant velocity, constant acceleration, and harmonic rotation. Experimental results show that our method can automatically and accurately recover the motion from the blurs captured under the fluttered shutter.

1 Introduction

Restoring motion blurred images is a challenging task as it relies on both accurate kernel estimation and robust deconvolution. Most existing approaches assume the blurs are caused by constant velocity motion and model the kernel as a box filter. Tremendous efforts have been focused on designing robust deconvolution methods, from the earlier approaches based on regularization [2] to the latest ones using image statistics [3] and edge priors [4]. However, since the box filter destroys high-frequency features that are difficult to recover post-capture, results using these deconvolution methods may still contain strong artifacts.

Several computational photography methods have recently been proposed to change the frequency profile of the blur kernel. The fluttered shutter (FS) camera developed by Raskar et al. [1] opens and closes the shutter during the exposure process according to a pre-determined sequence. The pseudo-random sequence creates a broad-band filter that preserves high frequency details and is robust to deconvolve. However, most existing fluttered shutter methods assume known constant velocity motions and rely on either user inputs [1] or alpha matting [5] to find the blur extent.

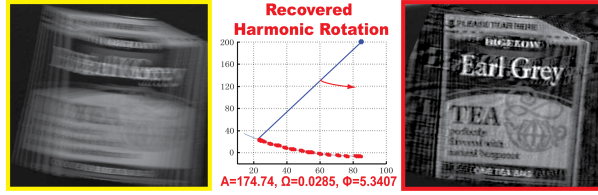


Fig. 1. Motion estimation and deblurring of harmonic rotation using our approach.

This paper addresses two fundamental problems when using the fluttered shutter: 1) how to apply the FS to handle a broader class of motions and 2) how to automatically recover the motion from the blurred image. For the first, we introduce a new fluttered-shutter point-spread-function (FS-PSF). FS-PSF uniformly models the blur kernel of arbitrary motions by computing how long each pixel gets exposed to the moving scene point throughout the shutter sequence. We show that many common motions such as constant velocity, acceleration, and harmonic rotation have closed-form FS-PSFs.

For the second, we present a new motion-from-blur method based on image power spectrum statistics. Schaaf and Hateren [6] have shown that circular power spectrum statistics of blur-free images follow the $1/\omega$ -exponent model. We extend their analysis to model the linear power spectrum of motion blurred images captured under the FS. We show that the Modulation Transfer Function (MTF) of the 1D FS-PSF should be strongly correlated to the linear statistics of the blurred image along the motion direction. We then develop a matching algorithm using a sign-of-derivative metric to find the motion parameters that yield the strongest correlation. We demonstrate our techniques on real images of various motion types. We show that our method can automatically and accurately recover the motion parameters from blurs under the fluttered shutter. Furthermore, the recovered motion can be used to modify the initial shutter sequence with improved invertibility in cases that have not previously been addressed in literature on coded exposure. Our specific contributions are:

1. A new motion-from-blur framework analyzing Fourier image statistics.
2. A closed-form formulation of the fluttered shutter point-spread-function (FS-PSF) to model general 1D motion blurs under the FS.
3. A new image statistics analysis that correlates the MTF of the FS-PSF with linear power spectrum statistics of the blurred image.
4. A sign-of-derivative matching algorithm to find the motion parameters that maximize the correlation. Our method also leads to the new design of the motion-aware fluttered shutters.

2 Related Work

Existing algorithms related to motion blur have focused on three main aspects: blur kernel (PSF) estimation, image deconvolution and, most recently, image acquisition.

PSF Estimation: PSF estimation from a single image is known to be ill-posed. Existing methods make it tractable in a number of ways. Yuan et al. [7], for example, use a blurred/noisy image pair of the same scene. Other approaches employ regularization, such as the classical Wiener filter [2]. Still other

approaches tackle the PSF estimation problem by constraining the space of potential PSFs. Assuming that blur arises from a traditional shutter with linear, constant-velocity motion constrains the potential PSFs to box filters. The cepstrum methods [8–10] have been proposed to characterize the motion by the number and position of zeros in the image power spectrum. However, these methods cannot be applied to fluttered shutter images that are acquired specifically to avoid such zeros. Recently, Dai and Wu [11] treat motion blurs as an alpha matte for estimating the PSF. Agrawal and Xu [5] apply a similar approach on the fluttered shutter. The implicit assumption in alpha-matte-based PSF estimation is the existence of high-contrast edges in the latent sharp image. Since the alpha matte only provides the blur extent, such methods cannot distinguish between the infinite number of velocity/acceleration combinations that might produce that extent.

Image Deconvolution: Numerous methods in the category of blind deconvolution [12] have been presented to mitigate the effects of motion or optical blur in images. Most motion deconvolution methods are based on the assumption that the object is moving along a straight line with constant velocity, in which case the PSF is a 1D box filter. Levin [4] examines the consequences of this type of blur on image statistics in order to perform blind deconvolution on blurred regions. It is also well understood that the magnitude of the Fourier transform of such a PSF has many zero points, where the frequency cannot be fully recovered. These missing frequencies lead to artifacts when using standard deconvolution. Though the scene’s content at these spatial frequencies is irrecoverable, outside information in the form of gradient or edge priors [13–15, 3, 16] can be used to produce visually pleasing images.

Acquisition: Sharp image acquisition of fast-moving can also be achieved using short exposure duration with high-powered flashes, which is impractical in most settings. Many modern digital cameras have adaptive optical elements controlled by inertial sensors to reduce the effects of moderate camera motion due to hand shakes. Using video with varying exposure durations, Agrawal et al. [17] capture multiple images with partial coverage of the spatial frequency spectrum, which are combined to produce a single sharp image with coverage of all spatial frequencies. Hybrid cameras [18–20] use additional images/video to obviate or simplify the kernel estimation step.

Our work is motivated by the Flutter Shutter (FS) method by Raskar et al. [1], in which a single image is acquired by randomly opening and closing the camera’s shutter during image capture. For constant velocity motion, the resulting blur kernel is invertible and standard image deconvolution can be directly used for deblurring. However, existing FS techniques assume known motion extent, e.g., via user specifications. In contrast, we set out to actively recover the motion from the blur. Our work is also related to Depth-from-Defocus (DfD) methods based on the coded apertures [21]. Although both DfD and motion estimation can be formulated as kernel estimation problems, motion blur kernels are usually complex yet spatially-invariant whereas defocus blur kernels are simple but spatially variant. As a result, motion estimation methods can uniformly treat

groups of pixels, e.g., via image statistics [3] while DfD techniques rely on other types of priors such as smoothness or edges [22, 23]. In this paper we analyze image statistics under the fluttered shutter for motion estimation.

3 Fluttered Shutter Point Spread Function (FS-PSF)

We start with defining the point-spread-function under the fluttered shutter that we call FS-PSF. We represent the shutter’s fluttering pattern as a sequence of chops with 1/0 values denoting the open/closed shutter states. We set every chop to have the same period w_{chop} and will use w_{chop} as the time unit t in the following analysis. Let $S(t)$ denote the flutter sequence, we have:

$$S(t) = \begin{cases} 0 & \text{shutter closed} \\ 1 & \text{shutter open} \end{cases}, t = 1, 2, 3, \dots, M_s \quad (1)$$

where t represents time, M_s is the number of chops in the sequence, and $E_s = w_{chop} \sum_{t=1}^{M_s} S(t)$ is the total exposure time.

The normalized FS-PSF $p(x)$ describes how much each pixel x gets exposed to a moving scene point Q . Therefore, it is a function of both the shutter sequence $S(t)$ and the motion of Q . To simplify our analysis, we adopt the same assumption in [1] that the moving object is frontal-planar and the FS-PSF is spatially-invariant. We measure the motion parameters such as displacement, velocity, and acceleration in unit of pixels, e.g., velocity as pixel/chop.

Recall that pixel x gets exposed to Q when Q ’s image passes through x . The exposure duration $w(x)$ is inverse proportional to Q ’s velocity $\nu(x)$ as:

$$w(x) = \frac{1}{\nu(x)} \quad (2)$$

Notice that, for general motions, it is natural to describe the velocity and displacement in terms of t . Thus, we can rewrite $w(x) = \frac{1}{\nu(t(x))}$, where $t(x)$ is the inverse of the displacement function $x(t)$. In this paper, we assume that $x(t)$ is monotonic throughout the shutter sequence, i.e., there is no back and forth motion, so that $x(t)$ is invertible.

Finally, we combine the shutter sequence and the exposure $w(x)$ to compute the un-normalized FS-PSF $p_0(x)$ ³ as:

$$p_0(x) = S(t(x))w(t(x)) = \frac{S(t(x))}{\nu(t(x))} \quad (3)$$

Eq. (3) indicates that the FS-PSF can be viewed as an envelope of $w(x)$ sampled by the shutter pattern $S(t)$ as shown in Fig. 2. To derive the FS-PSF for arbitrary motions, we simply need to derive $t(x)$.

3.1 Constant Velocity

For constant velocity motion at ν_c pixels/chop, we assume the first exposed pixel is the 0-th pixel, and we have $x(t) = \nu_c \cdot t$ and $t(x) = x/\nu_c$. The FS-PSF is thus:

$$p_0(x) = \frac{S(t(x))}{\nu_c} = \frac{S(\frac{x}{\nu_c})}{\nu_c}, x = 1, \dots, \nu_c M_s \quad (4)$$

³ $p_0(x)$ is later be normalized to FS-PSF $p(x)$

Since the last exposed pixel coordinate has $x(M_s) = \nu_c M_s$, we can compute the normalized FS-PSF as:

$$p(x) = \frac{p_0(x)}{\sum_{x=1}^{\nu_c M_s} p_0(x)} = \frac{S(\frac{x}{\nu_c})}{\nu_c E_s}, \quad x = 1, \dots, \nu_c M_s \quad (5)$$

Eq. (5) indicates that varying the velocity ν_c will result in spatial scaling in the PSF $p(x)$, while the envelope of $p(x)$ remains as a rectangle. An example is shown in Fig. 2(left); the discretization of x in pixels may result in non-integer values. Our FS-PSF approximates non-integer pixels as the closest integer pixels with partial exposure intensity.

Recall that existing methods [1, 5] directly treat the shutter sequence as the PSF, i.e., $p(x) = S(x)$. It is a special case of Eq. (4) where $\nu_c = 1$ pixel/chop, and was achieved by manually re-sampling the captured image.

3.2 Constant Acceleration

Let ν_s and ν_e denote the velocity of Q at the start and end of the shutter sequence. The acceleration a can be computed as: $a = \frac{\nu_e - \nu_s}{M_s}$. The velocity $\nu(t)$ and displacement $x(t)$ are:

$$\nu(t) = \nu_s + a \cdot t, \quad x(t) = \nu_s \cdot t + \frac{a}{2} \cdot t^2 \quad (6)$$

We can invert Eq. (6) to compute $t(x)$ as:

$$t(x) = \frac{-\nu_s + \sqrt{\nu_s^2 + 2a \cdot x}}{a} \quad (7)$$

Finally, we can derive its FS-PSF using Eq. (3):

$$p_0(x) = \frac{S(t(x))}{\nu(t(x))} = \frac{S(\frac{-\nu_s + \sqrt{\nu_s^2 + 2a \cdot x}}{a})}{\sqrt{\nu_s^2 + 2a \cdot x}} \quad (8)$$

The envelope of the constant acceleration FS-PSFs can be approximated as a trapezoid, as shown in Fig. 2 (middle); changing the starting velocity or the acceleration varies the slope and the shape of the trapezoid. We use ν_s and a as the parameters for constant acceleration motion.

3.3 Linear Harmonic Motion

Linear harmonic motion is a periodic motion, where an object oscillates about an equilibrium position in a sinusoidal pattern, such as the commonly studied spring-mass system and the pendulum (recall Fig. 1).

We parameterize the linear harmonic motion by the amplitude A , the angular speed Ω , and the initial phase Φ . We first compute x and ν as functions of t :

$$x(t) = A \sin(\Omega t + \Phi), \quad \nu(t) = A \Omega \cos(\Omega t + \Phi) \quad (9)$$

we solve t as an inverse function of x from Eq. (9):

$$t(x) = \frac{\arcsin(\frac{x}{A} - \Phi)}{\Omega} \quad (10)$$

Finally, we re-write Eq. (3) and compute the corresponding FS-PSF:

$$p_0(x) = \frac{S(t(x))}{\nu(t(x))} = \frac{S(\frac{\arcsin(\frac{x}{A} - \Phi)}{\Omega})}{A \Omega \cos(\Omega \cdot \frac{\arcsin(\frac{x}{A} - \Phi)}{\Omega} + \Phi)} \quad (11)$$

Fig. 2(right) illustrates harmonic motion with $A = 30$, $\Phi = \pi/2$, $\Omega = \pi/30$.

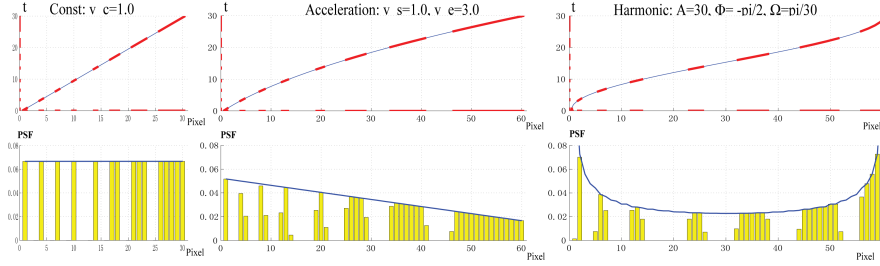


Fig. 2. FS-PSFs of common motions: constant velocity (left), constant acceleration (middle), and harmonic motion (right). Top row shows the time-velocity function sampled by the shutter (in red). Bottom row shows the corresponding FS-PSF.

4 Recovering Motion PSFs

We have shown many commonly observed motions have closed-form PSFs. Our goal is to recover the FS-PSF by analyzing blurred images. Recall that the process of motion blur can be modeled as standard convolution:

$$i(x, y) = j \otimes p(x, y) + n(x, y) \quad (12)$$

where \otimes is the convolution operator, j is the latent sharp image, i is the degraded image, p is the blur kernel, and n is noise.

If we ignore n , we can model the amplitude spectrum of Eq. (12) as:

$$|I| = |JP| = |J||P| \quad (13)$$

where i/I , j/J , and p/P are Fourier pairs, and $|\cdot|$ is the modulus operator. $|P|$ is also called the Modulation Transfer Function for 1D PSFs.

4.1 Power Spectrum Statistics

Our FS-PSF estimation algorithm is based on power spectrum statistics in natural images. van der Schaaf and van Hateren [6] have shown that, for a natural image j without motion blur, its circular power spectrum statistics follows the $1/\omega$ -exponent model: if we parameterize $|J|$ in polar coordinates (ω, ϕ) where ω is the radius (absolute frequency) and ϕ is the angle, we can average $|J|$ over ϕ for every ω and the resulting circular averaged power spectrum $\text{circ}_\omega(|J|) \approx \frac{C}{\omega^m}$, where m and C are constants. Statistically, if we assume every frequency is an independent and identically distributed random variable, circular statistics reveals that the expected value of $|J(u, v)|$ is:

$$E[|J(u, v)|] = \frac{C}{(u^2 + v^2)^{m/2}} \quad (14)$$

Fig. 3(a) shows example traces of the power spectra of five natural images.

Our goal is to use power spectrum statistics to recover the FS-PSF from the blurred image i . In this paper, we assume the motion type (constant velocity, acceleration, etc.) is known and we focus on recovering its corresponding motion parameters α . Given an candidate α , we can compute the closed-form FS-PSF p as shown in Sec. 3 and calculate its MTF $|P|$. The latent image power spectrum $|J|$ can then be computed as $|I|/|P|$ from Eq. (13). If α is the correct motion estimate, J should be motion blur free and its circular averaged power spectrum $\text{circ}_\omega(|J|)$ should follow $\frac{C}{\omega^m}$ distribution. A naive approach for testing if α is a good motion estimation, then, would be to check if $\text{circ}_\omega(|I|/|P|)$ has a

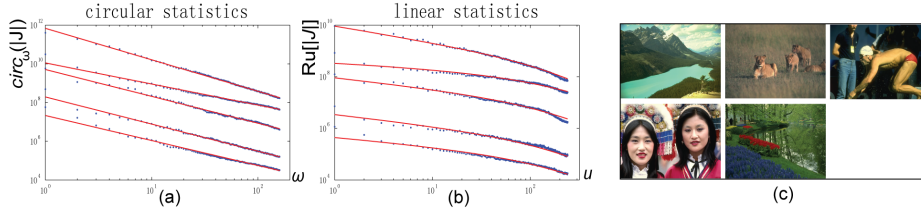


Fig. 3. Power Spectrum Statistics on Five Randomly Selected Images from the Berkeley Segmentation Database [24]. (a) The circular power spectrum vs. the spatial frequency ω in a log-log scale. The red lines show the fits of the $1/\omega$ -exponent model. The scaling of the vertical axis belongs to the top trace. (b) The linear statistics along v vs. u in a log-log scale. The red curves show our estimated linear statistics from the circular statistics. For clarity, traces in both plots are shifted -1, -2, -3, and -4 log-units.

$\frac{C}{\omega^m}$ distribution. However, since $circ_{\omega}(|J|)$ only represents the statistics of $|J|$, incorrectly estimated motion parameters may still produce such distributions. Therefore, we set out to match the statistics between $|P|$ and $|I|$ instead.

4.2 Linear Power Spectrum Statistics

We first replace circular power spectrum statistics with linear statistics. Specifically, we *project* the 2D power spectrum onto a line l that corresponds to the motion direction in the spectral domain. We rotate the Fourier plane so that l is aligned with the u axis and apply the projection by integrating over v . This process can be alternatively viewed as applying a Radon Transform [25] along the v direction. In the discrete case, we can compute the linear averaged power spectrum of an image $|J|$ as:

$$R_u[|J|] = \frac{1}{V} \sum_{v=0}^V |J(u, v)| \quad (15)$$

where V is the v -dimension resolution. $R_u[|J|]$ represents the horizontal power spectrum statistics and can be approximated using Eq. (14) as:

$$R_u[|J|] \approx E\left[\frac{1}{V} \sum_{v=0}^V |J(u, v)|\right] = \frac{1}{V} \sum_{v=0}^V \frac{C}{(u^2 + v^2)^{m/2}} \quad (16)$$

Fig. 3(b) illustrates that our $R_u[\cdot]$ estimation is accurate and robust.

We can further apply the R_u operator to both sides of Eq. (13):

$$R_u[|I|] = \sum_{v=0}^V |J(u, v)| |P(u)| = |P(u)| \cdot R_u[|J|] \quad (17)$$

Eq. (17) allows us to separate $R_u[|J|]$ and $|P|$. We can further take the log of Eq. (17) as:

$$\log(R_u[|I|]) = \log(|P|) + \log(R_u[|J|]) \quad (18)$$

4.3 Motion Estimation

Fig. 4 illustrates our motion estimation algorithm. We first determine the motion direction and align it with the u axis. For every candidate motion parameter α , we compute its FS-PSF p^α and MTF $|P^\alpha|$, and use it to estimate the latent image power spectrum $|J^\alpha| = |I|/|P^\alpha|$. We then compute the linear statistics $R_u[|J^\alpha|]$, and $R_u[|I|]$. Finally, we compute the match score μ between $\log(|P^\alpha|)$ and $\log(R_u[|I|]) - \log(R_u[|J^\alpha|])$. The optimal motion parameter α corresponds to the one that maximizes μ .

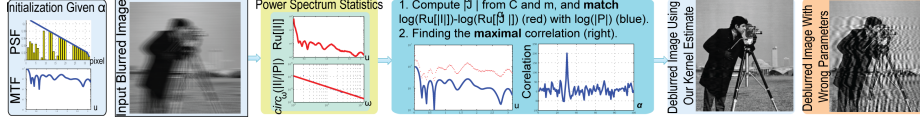


Fig. 4. Steps of Our Power-Spectrum-Based Motion Estimation Algorithm

Estimating the Motion Direction. We adopt a similar approach to [26] that finds the direction with most muted high frequencies. This assumes that the latent sharp image is not highly anisotropic, i.e., the power spectrum distribution along all directions have similar characteristics (variance, mean values). Since 1D motion blur attenuates the middle- and high-frequency information in the direction of motion, it amounts to a detection of a direction in which they are most muted. We do this by inspecting the Radon-power spectrum of the blurred image in all directions and choosing the one with the maximal variance.

Computing Linear Statistics of $|J^\alpha|$. A crucial step in our motion estimation algorithm is to derive the linear statistics of $|J^\alpha| = |I|/|P^\alpha|$ from the circular statistics. Since we assume J^α is motion blur free, its circular statistics should follow $1/\omega$ -exponent distribution. To estimate C and m , we compute the discrete circular averaged power spectrum and apply line fitting between $\log(\text{circ}_\omega[|J^\alpha|])$ and $\log(\omega)$. We then approximate the linear statistics $R_u[|J^\alpha|]$ using Eq. (16).

Matching Log-Linear Statistics. Recall that our ultimate goal is to match $f_1 = \log(|P^\alpha|)$ and $f_2 = \log(R_u[|I|]) - \log(R_u[|J^\alpha|])$ under some metric μ . A native μ is to measure the squared difference at sampled points on f_1 and f_2 . Since the power spectrums of images generally have much smaller values in high frequency, directly computing the correlation between the estimate f_1 and f_2 results in unequal contributions from different frequencies.

We employ a metric based on the signs of the function derivatives to equally treat all frequencies. Specifically, we use a derivative sign function $\Gamma(\cdot)$:

$$\Gamma(\chi(u)) = \begin{cases} 1, & \frac{d\chi}{du} \geq 0 \\ -1, & \frac{d\chi}{du} < 0 \end{cases} \quad (19)$$

where χ is a 1D function on u .

Finally, we sample f_1 and f_2 at discrete points u_1, u_2, \dots, u_n , and compute:

$$\mu(f_1, f_2) = \sum_{i=1}^n \Gamma(f_1(u_i)) \Gamma(f_2(u_i)) \quad (20)$$

4.4 Motion-Aware Fluttered Shutter

Although not the focus of the paper, we briefly discuss how to use our techniques to develop motion-aware flutter shutters. The standard flutter shutter method has been focused on deblurring a single image. For videos, the object’s motion may vary across the frames. Therefore, we aim to use the recovered motion to further update the initial shutter sequence to better match the motions.

Our strategy is to first determine the shutter sequence in the spatial domain and then map the sequence to the temporal domain. Recall that we have shown in Sec. 3 that the FS-PSF can be viewed as a motion envelope sampled by the shutter sequence: the envelope is a function of recovered motion parameters α

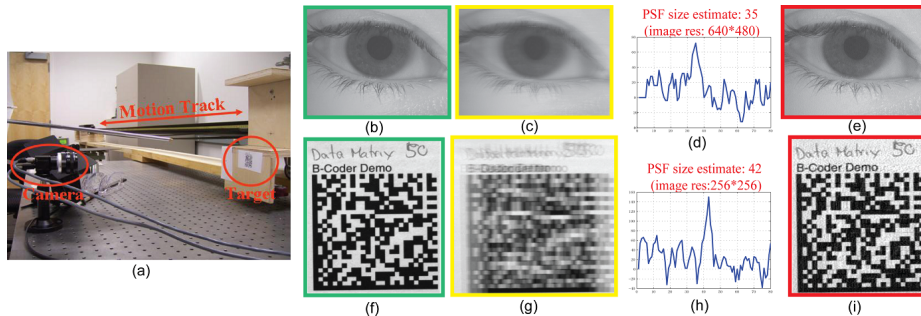


Fig. 5. Motion Estimation and Deblurring Results on an Iris Image and a Bar Code Image. (a) shows our motion stage and the fluttered shutter camera. Column 2: The ground truth blur-free images. Column 3: Blurred images caused by constant velocity motion under the FS. Column 4: The matching metric vs. the motion parameter (velocity). Column 5: The deblurred results using our recovered motion parameter.

and the sampling is determined by the shutter sequence. We can directly model the FS-PSFs as a dot product of the envelope $w(x)$ and a binary sequence $b(x)$ in the spatial domain and apply the same search scheme in [1] and [5] to locate an optimal $b(x)$ so that $w(x)b(x)$ is "most" invertible, i.e., the one that has the maximal minimum magnitude in its MTF. Finally, we determine the flutter pattern $s(t)$ from $b(x)$ by using the motion model:

$$s(t) = s(t(x)) = b(x(t)) \quad (21)$$

Fig. 8 compares the deblur results using the initial const velocity optimal sequence and using our motion aware sequence.

5 Results

We have applied our technique to all of the publicly available flutter shutter images [1], and find that our method produces estimates that are within 1 pixel of the ground truth values, giving high-quality reconstructions. In order to test the broader types of blur (acceleration, harmonic motion) handled by our method, we have acquired additional test images using a Point Grey Flea2 camera triggered via the serial port of the controlling computer. The camera supports an external shutter mode that accumulates exposure over several chops, after which a single readout produces the flutter shutter image. To deblur the image from our recovered FS-PSF, we use the linear system solution [1] for constant velocity motions and the Gaussian-derivative-prior method [21] for constant acceleration and harmonic rotation motions.

5.1 Constant Velocity

We first validate our algorithm on constant velocity motion. The only parameter here is the velocity. We captured the images from a fixed camera observing a motion stage to which textured objects are attached as shown in Fig. 5(a). Our motion stage can simulate different velocity motions via voltage controls. To measure the ground truth velocity, we use a step edge calibration target and measure its blurred width in an image with a known exposure time. We choose the shutter sequence as in [1] whose MTF has the maximum min magnitude and

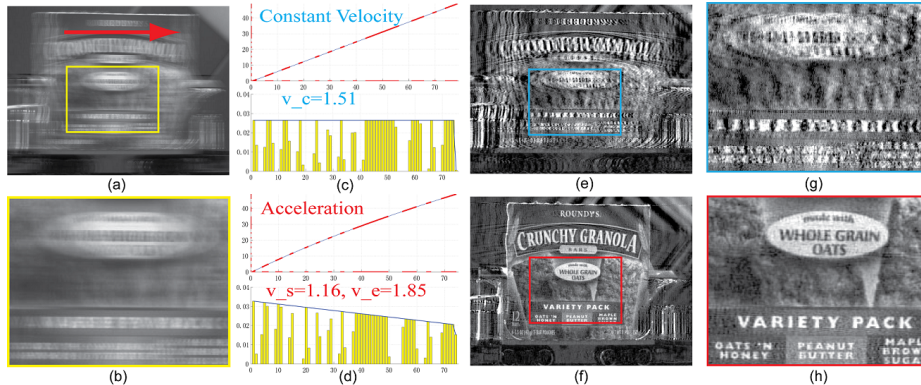


Fig. 6. FS-PSF Estimation and Deblurring Results on Constant Accelerations. A toy car sliding down a slanted track at 55° . The camera is rotated so that the motion appears horizontal. For clarity, a textured cardboard was attached to the car. (a): The motion blurred image under the fluttered shutter. (c) and (d) show the functions $t(x)$ and the corresponding PSFs using constant velocity assumption and our recovered constant acceleration motion. (e) The deblurred result using the FS-PSF of constant velocity ($v_c=1.51$ pixels/chop). (f) The deblurred result using our algorithm’s FS-PSF estimation for constant acceleration ($v_s=1.16$ pixel/chop, $v_e=1.85$ pixel/chop). (b), (g), and (h) are close-up views for (a), (e), and (f).

has a chop duration of $v_c = 1$ pixel/chop. With this setup, we obtain the ground truth FS-PSF using Eq. (5).

Fig. 5 shows two examples acquired using this setup, an iris image (b) with little texture and a bar code image (f) with repetitive texture. The iris and 2D barcode targets move from left to right with a constant velocity, giving the flutter shutter images (c) and (g). In both cases the motion is axis-aligned horizontal. Our estimated motion direction is within 1° degree of this ground truth. The plots (d) and (h) show the matching metric μ computed over a range of potential PSF sizes (proportional to velocities in this case), which have pronounced peaks exactly at the ground truth values (35 pixels for the iris image and 42 pixels for the barcode image). The resulting FS-PSF estimates are then used to deblur (c) and (g). Our deblurred results (e) and (i) contain sufficient detail to perform recognition on the de-blurred images. The iris template extracted from our deblurred image was successfully matched to a separate image of the same eye, and the barcode image can be decoded to extract its payload. Neither the iris recognition nor the barcode decoding were successful on Lucy-Richardson [27] de-blurred versions of traditional shutter images captured with the same setup.

5.2 Constant Acceleration

For constant acceleration, the motion parameters α are the starting velocity and acceleration. We capture accelerated motion images using a toy car on a slanted track, using a dead drop for which gravity provides the only acceleration. Because of the unknown timing between the release of the car and the image capture, we are unable to determine the ground truth FS-PSF for these images. Instead, we validate our motion estimation by the quality of the deblurred results. The

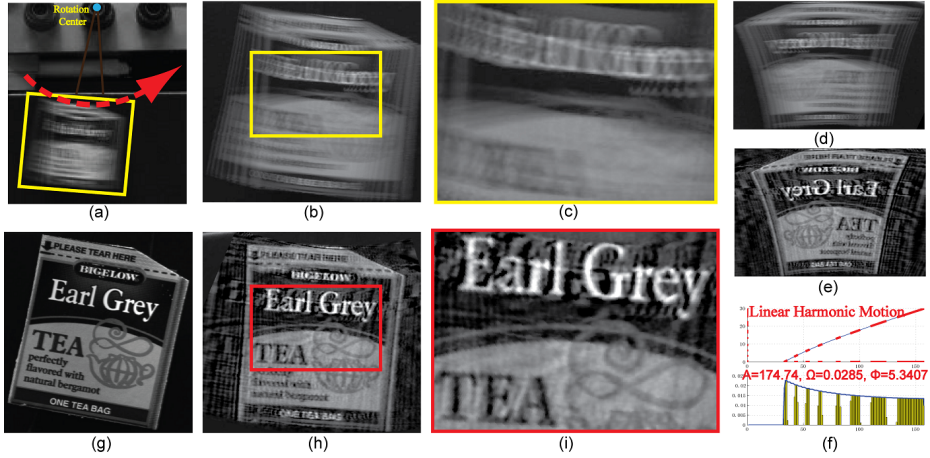


Fig. 7. FS-PSF Estimation and Deblurring Results on a Harmonic Rotational Tea Bag (g). (a) and (b): The captured blur image under the FS. (d): Warped (b) under polar coordinates. (e) and (h): Our motion deblurred result under the polar and the cartesian coordinates. (f): Our recovered FS-PSF. (c) and (i): Close-up views for (b) and (h).

shutter sequence used in these experiments is computed under the constant velocity assumption and does not account for acceleration. Because the velocity and acceleration are unknown a priori, these images are generated with what is essentially a random flutter shutter sequence.

Fig. 6 (a) shows the image captured as the toy train undergoes accelerated motion. Though the track is slanted at 55° , the camera is rotated so that the motion appears nearly horizontal. We first apply our motion direction estimation algorithm, which produces an estimate of 1° . Next, we apply our power spectrum statistics approach to determine the acceleration motion parameter, which gives $t(x)$ and the FS-PSF shown in (d). The deblurred result is shown in (f), and a close-up in (h). Given the severe blur in (a) and the fact that the fluttering sequence is not optimal under accelerated motion, the amount of detail present in the close-up is significant. Note that reconstruction artifacts in (f) are due to the stationary background’s intensity interacting with the moving foreground object. We also present the deblurred result assuming a constant velocity motion model. Our algorithm first estimates the motion velocity and plots the $t(x)$ and PSF in (c). As shown in (e) and (g), using incorrect motion model, the deblurred images contain severe artifacts.

5.3 Harmonic Rotation

Finally, we experiment our approach on planar harmonic rotation. The harmonic rotation consists of 3 parameters, i.e., A , Ω , and Φ . As shown in Fig. 7(a), we emulate harmonic rotation by hanging a heavy rigid object below a fixed stick using two approximately rigid, weightless cords. These two cords are connected to the same point. By swinging the object back and forth freely within a plane, we synthesize a periodic harmonic rotation. Notice that the rotation is 2-dimensional, with spatially varying blur kernels for different pixels (Fig. 7(b)).

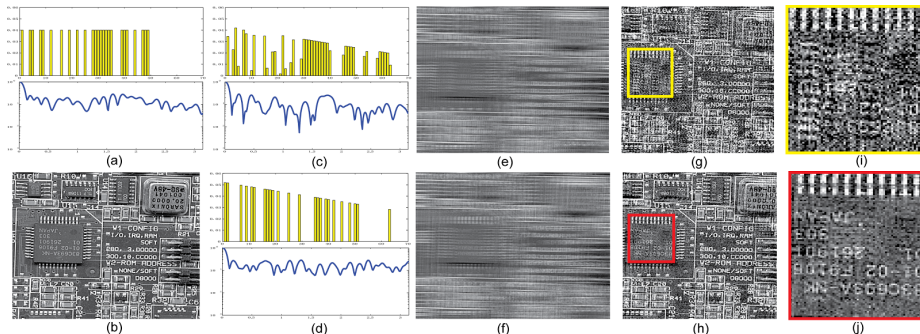


Fig. 8. Motion-aware Fluttered Shutter. (a) is the PSF and the MTF under constant velocity. (c) is the PSF/MTF under acceleration motion ($v_s=0.8$ pixels/chop, $v_e=1.8$ pixels/chop) using the same sequence as in (a). (d) is the PSF/MTF using our motion-aware sequence. (e) and (f) are synthetically blurred images of (b) using acceleration PSFs in (c) and (d). (g) and (h) are the corresponding deblurred images, (i) and (j) show the close-up views.

In order to simplify the analysis, we transform the harmonic rotation into a linear harmonic motion. Specifically, we track feature points and estimate the rotation center by solving a least squares problem [16]. We then warp the image along the radial directions to form a spatially invariant linear harmonic motion blur (d). Our algorithm recovers the harmonic motion parameters (f) and then deblurs the image (d) and obtain (e). Finally, we warp the image back to the original cartesian coordinate system (h).

5.4 Motion Aware Shutter Sequence

We pick a fluttered shutter sequence originally designed optimal for constant velocity motion ($v=1.0$ pixel/chop) as the initial shutter sequence. We then use this sequence to capture an accelerated motion with $v_s=0.8$ pixels/chop and $v_e=1.8$ pixels/chop. The resulting FS-PSF is shown in Fig. 8(c). Notice that it has small values at several frequencies. We synthetically blur a sharp image (b) using the FS-PSF with additive Gaussian white noise of $\sigma=0.01$. We then deblur it using our motion estimation algorithm, with constant acceleration. Although our method recovers highly accurate motion parameters ($v_s=0.790$, $v_e=1.805$), the resulting deblurred results contain strong ringing artifacts.

Using the recovered motion parameter, we apply the random search scheme as in [1] to find the optimal flutter shutter sequence. The new FS-PSF is shown in Fig. 8(d). Compared with the old FS-PSF, it maintains large values at all frequencies. We use the new FS-PSF to blur the sharp image and also add Gaussian white noise $\sigma=0.01$. Finally, we apply our power spectrum statistics method to recover the motion parameter and obtain a new deblurred image as shown in (h). (i) and (j) show the close-up views of the deblurred results under the old and new FS-PSF. The motion-aware FS-PSF yields much less artifacts.

6 Conclusion and Limitations

We have presented a new fluttered-shutter-based motion estimation and deblurring framework. Our method adopts the fluttered-shutter point-spread-function

(FS-PSF) model to uniformly describe blur kernels under general motions. We have developed an automatic motion-from-blur technique that recovers the FS-PSF by analyzing image power spectrum statistics. We have introduced a new linear statistics model that can be directly estimated from circular power spectrum statistics. We have shown that the MTF of 1D FS-PSF should be statistically correlated to the linear statistics of the blurred image’s power spectrum along the motion directions. To find the optimal FS-PSF, our method searches the space of motion parameters to find the one that yields the maximum correlation.

The use of fluttered shutters is crucial in our motion-from-blur algorithm. Recall that the first step in our linear statistics estimation is to compute the latent image power spectrum $|J| = |I|/|P|$. The implicit assumption there is that $|P|$ does not contain zeros, the most important property of the fluttered shutter. For conventional shutters where P is a sinc function and has many zeros, the resulting $|J|$ will contain points with large values and robustly fitting $1/\omega$ -exponent distribution to circular power spectrum statistics is difficult. Thus, our technique is not directly applicable to the box filters.

Another limitation of our framework is that it is restricted to 1D motions. 1D motions allows us to efficiently separate the FS-PSF from linear statistics of the latent image (Eq. (17)). Intuitively, our technique may be directly applied to 2D motions. For example, once we compute C and m of the $1/\omega$ -exponent model, we can approximate $|J(u, v)| \approx C/(u^2 + v^2)^{m/2}$ and directly match the 2D function $|I|/|J|$ with $|P|$. However, since the $1/\omega$ -exponent model is a statistical model, the actual $|J|$ values may significantly deviate from their expected values. Therefore, matching 2D $|I|/|J|$ with $|P|$ is not reliable. A possible solution for future work is to approximate the 2D FS-PSF as combinations of 1D FS-PSFs and then reapply our linear statistics method to fit along their corresponding directions. Another important future direction is to use our motion-aware flutter shutter for video deblurring. The challenge there is to determine the optimal shutter sequence from the estimated motions in real-time (e.g., 30fps). Recall that majority of our computations lie in the spectral space and computing image statistics is similar to texture filtering. Therefore, we plan to re-implement our algorithm on the GPU for real-time motion estimation and shutter selection.

Acknowledgement

The authors would like to thank the reviewers for their insightful comments. Part of this work was done while the first author was an intern at Honeywell Labs. Yuanyuan Ding and Jingyi Yu were partially supported by NSF grants MSPA-MCS-0625931 and IIS-CAREER-0845268.

References

1. Raskar, R., Agrawal, A., Tumblin, J.: Coded exposure photography: motion deblurring using fluttered shutter. *ACM Trans. Graph.* **25** (2006) 795–804
2. Y. Yitzhaky, I. Mor, A.L., Kopeika, N.S.: Direct method for restoration of motion-blurred images. In: *Journal of the Optical Society of America A: Optics, Image Science, and Vision*. Volume 15. (1998) 1512–1519

3. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-laplacian priors. In: NIPS. (2009) 1–8
4. Levin, A.: Blind motion deblurring using image statistics. In: NIPS. (2007) 841–848
5. Agrawal, A., Xu, Y.: Coded exposure deblurring: Optimized codes for psf estimation and invertibility. CVPR (2009)
6. Schaaf, A.v.d., Hateren, J.v.: Modelling the power spectra of natural images: Statistics and information. *Vision Research* **36** (1996) 2759–2770
7. Yuan, L., Sun, J., Quan, L., Shum, H.Y.: Image deblurring with blurred/noisy image pairs. *ACM Trans. Graph.* **26** (2007) 1
8. Ott, K.A., Krahmer, F., Lin, Y., McAdoo, B., Wang, J., Widemann, D., Wohlberg, B.: Blind image deconvolution: Motion blur estimation. In: Technical Report for the Mathematical Modeling in Industry X Workshop. (2006)
9. Moghaddam, M.E., Jamzad, M.: Fining point spread function of motion blur using radon transformation and modeling the motion length. In: ISSPIT. (2004)
10. Ji, H., Liu, C.: Motion blur identification from image gradients. In: CVPR. (2008)
11. Dai, S., Wu, Y.: Motion from blur. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on* **0** (2008) 1–8
12. Haykin, S.: *Blind Deconvolution*. Prentice-Hall (1994)
13. Jia, J.: Single image motion deblurring using transparency. In: *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*. (2007) 1–8
14. Shan, Q., Jia, J., Agarwala, A.: High-quality motion deblurring from a single image. In: *SIGGRAPH '08*. (2008) 1–10
15. Fergus, R., Singh, B., Hertzmann, A., Roweis, S.T., Freeman, W.T.: Removing camera shake from a single photograph. In: *SIGGRAPH '06*. (2006) 787–794
16. Shan, Q., Xiong, W., Jia, J.: Rotational motion deblurring of a rigid object from a single image. In: *ICCV 2007*. (2007) 1–8
17. Agrawal, A., Xu, Y., Raskar, R.: Invertible motion blur in video. *ACM Trans. Graph.* **28** (2009) 1–8
18. Ben-Ezra, M., Nayar, S.K.: Motion-based motion deblurring. *IEEE Trans. Pattern Anal. Mach. Intell.* **26** (2004) 689–698
19. Li, F., Yu, J., Chai, J.: A hybrid camera for motion deblurring and depth map super-resolution. In: CVPR. (2008) 1–8
20. Tai, Y.W., Du, H., Brown, M.S., Lin, S.: Image/video deblurring using a hybrid camera. In: CVPR. (2008) 1–8
21. Levin, A., Fergus, R., Durand, F., Freeman, W.T.: Image and depth from a conventional camera with a coded aperture. In: *ACM SIGGRAPH*. (2007)
22. Joshi, N., Zitnick, C., Szeliski, R., Kriegman, D.: Image deblurring and denoising using color priors. (2009) 1550–1557
23. Favaro, P., Soatto, S.: Learning shape from defocus. In: *ECCV (2)*. (2002) 735–745
24. Martin, D.R., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. Technical Report UCB/CSD-01-1133, EECS Department, University of California, Berkeley (2001)
25. Toft, P.: *The Radon Transform — Theory and Implementation*. PhD thesis, Electronics Institute, Technical University of Denmark, Lyngby, Denmark (1996)
26. Oliveira, J.P., Figueiredo, M.A., Bioucas-Dias, J.M.: Blind estimation of motion blur parameters for image deconvolution. In: *IbPRIA '07*. (2007) 604–611
27. Lucy, L.B.: An iterative technique for the rectification of observed distributions. *Astron. Journal.* **79** (1974) 745+