A Low-noise Fluttering Shutter Camera Handling Accelerated Motion

Scott McCloskey, Sharath Venkatesha, Kelly Muldoon
Honeywell ACS Labs
1985 Douglas Drive N., Golden Valley, MN
{firstname.lastname}@honeywell.com

Ryan Eckman
Promet International, Inc.
4611 Chatsworth St. N., Shoreview, MN
ryan@promet.net

Abstract

We address the problem of motion blur removal using a computational camera with a fluttering shutter. While there are several prototype flutter shutter cameras, and many scenarios in which motion blur is problematic, there are few real-world uses of flutter shutter cameras due to two important limitations. The first is that the shutter mechanisms used to date - primarily Liquid Crystal Display (LCD) elements or electronic shutters - increase noise due to reduced light efficiency or multiple readouts, respectively. Secondly, the class of motions to which the flutter shutter is applicable has been limited to linear, constant velocity motion. We address the first limitation by developing a prototype flutter shutter camera with a reflective element providing high light efficiency and a single-read imaging system. In addition to improved noise performance, this method of exposure modulation imposes fewer limitations on the shutter sequence, allowing us to extend the flutter shutter technique to cases with constant (non-zero) acceleration. We demonstrate both the noise reduction and improved reconstructions in the case of an accelerating camera.

1. Introduction

Motion blur is limiting to both the aesthetic quality of images and their utility for recognition and other tasks. When either the camera or parts of the scene move during exposure, light reflected from an object is spread over multiple sensor pixels. Processing images degraded in this way is a long-standing problem in computer vision, with the majority of the work manipulating imagery captured using a traditional camera. However, these methods are limited by the content of imagery captured by a traditional camera. In particular, it is well known that linear motion with constant velocity gives rise to a rectangular Point Spread Function (PSF) whose de-convolution is ill-conditioned because the captured image has no contrast at certain spatial frequencies. This follows from the convolution theorem and the fact that the rectangular PSF's Fourier equivalent is a sinc function, meaning that the Modulation Transfer Function (MTF) goes to zero at several spatial frequencies. Since no image texture from the scene is captured in the image, such spatial frequencies are said to be *lost*.

In order to make motion deblurring well-conditioned for linear, constant velocity motion, Raskar *et al*. [14] introduced the flutter shutter camera, which captures an image of a moving object while exposure is temporally modulated. Conceptually, the shutter opens and closes several times during the exposure of the image, so the rectangular PSF of a traditional camera is replaced by one whose deconvolution is numerically stable. While random shutter timing sequences avoid lost spatial frequencies, it has been shown [2, 11] that carefully choosing the timing sequence leads to further improvement in reconstruction quality or easier blur estimation.

The flutter shutter technique has been shown to improve the performance of both iris-based biometric recognition [10] and barcode scanning [18]. Despite this, few such cameras are commercially available, due to two important limitations. First, shutter mechanisms which have been used in previous work lead to increased noise in the captured image, either by reducing the light throughput of the optical system or by using a multiple exposure mechanism that compounds noise from the sensor. Second, the flutter shutter work to date has focused on constant velocity, and does not address more general motions. As we describe below, this limitation follows, to some extent, from hardware constraints.

In order to reduce the prevalence of noise in the captured image, and to enable coded exposure capture in cases of constant (non-zero) acceleration, we present a mirror-based lens based on a Digital Micro-mirror Device (DMD). By using a reflective element for exposure modulation, we maintain high light throughput while eliminating the need to read multiple exposures. The switching time is also short enough to shape the PSF in the presence of significant object acceleration. We demonstrate these improvements with our reflective optical system and a Point Grey machine vision camera.

2. Related Work

The flutter shutter method was first proposed by Raskar *et al*. [14], who demonstrated the approach using a Digital Single Lens Reflex (DSLR) camera with an LCD shutter added at the entry aperture of the lens. While this sufficed to demonstrate the utility of exposure coding, the LCD itself greatly reduces the light flux into the main lens, as LCDs involve polarization which reduces the amount of light by about two stops. In subsequent work [1, 2, 8], others have used machine vision cameras from Point Grey, which enable software-based coded exposure capture. As pointed out in [8], however, this implementation gives noise levels in the captured image which are proportional to the number of open shutter periods in the shutter sequence. This is due to the fact that the light is accumulated off of the sensor array so, at the end of each open shutter sequence, exposure from the unshielded pixels are transferred and various noise penalties are incurred. This increased noise in the captured image is then exacerbated by the de-convolution process. This noise can be avoided by modulating exposure with active illumination while an image is continuously exposed [13, 9], but these approaches are ineffective in high levels of ambient illumination and for subjects at long distances.

Besides the flutter shutter, other computational cameras have been proposed to make motion de-blurring well-posed, including the motion invariant camera introduced by Levin *et al*. [7]. Because the shutter is open during the entire exposure, motion invariant cameras based on moving optical elements [12] achieve higher light throughput than coded exposure. The implementation of this technique with a Liquid Crystal on Silicon (LCoS) device by Sonoda *et al*. [15] could also be used for coded exposure, but incurs a light efficiency penalty due to polarization used by the LCoS device.

Because of the prevalence of motion-blurred imagery taken with them, many methods address both blur estimation and de-blurring of imagery from traditional cameras. Much of the recent work in this area has generalized the type of motion considered, specifically spatially variant blur arising from camera rotations [16, 3]. While these methods show a marked improvement on images with out-of-plane motion, experiments with a dataset of hand-held camera motions [5] have shown that they don't necessarily outperform those based on a shift-invariant model of motion blur. Kohler's evaluation actually shows that the work of Xu and Jia [17] performed best overall despite an underlying assumption of uniform blur.

3. Review of Flutter Shutter Theory

Like the method of Xu and Jia, flutter shutter cameras are based on the classic shift invariant model of motion blur, where a convolution of a latent image I with a PSF B, gives

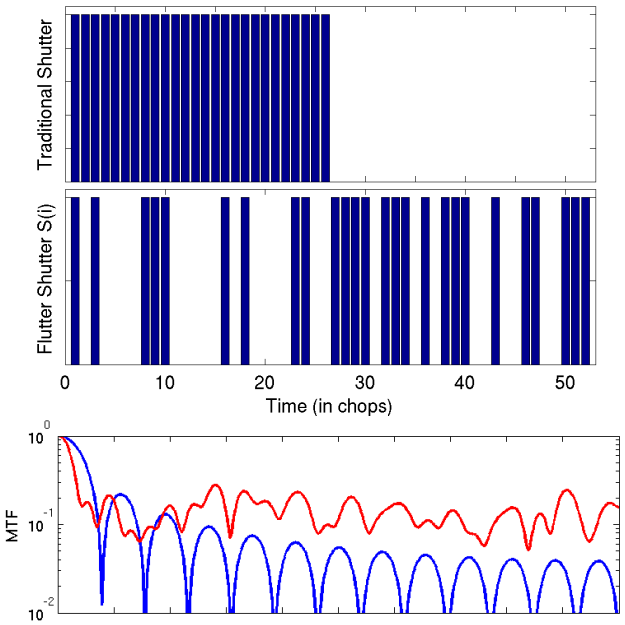

Figure 1. (Top) The timing of a traditional shutter, which gives rise to a rectangular PSF. (Middle Row) The timing of coded exposure using the sequence from [14]. (Bottom) The MTF of a traditional shutter (blue) vanishes at several lost spatial frequencies, whereas that of a flutter shutter camera (red) keeps higher contrast.

the blurred image

$$I_b = I * B + \eta, \tag{1}$$

where η represents noise. Conceptually, the key of the flutter shutter approach is to engineer a point spread function B such that I_b contains information about I at all spatial frequencies. This is not the case for a traditional camera under linear, constant velocity motion, where B is a 1D rectangle function in the motion direction. Flutter shutter cameras use a timing sequence $S = (s(0), s(1), ...s(N-1))$ with each entry $s(i) \in \{0, 1\}$ representing whether the shutter is open or closed during the corresponding time span, called a *chop*, and each chop has the same duration. Figure 1 illustrates the shutter sequence from [14], and shows the difference between traditional imaging and coded exposure MTFs.

In addition to the shutter sequence, the motion of the object/camera determines the blur PSF. If a moving object with constant velocity transits 2 pixels per t_{chop}, for instance, the PSF B has length $2N$, and

$$B_{2N}(i) = \frac{s(\lfloor i/2 \rfloor)}{2\sum s}. \tag{2}$$

Previous work with the flutter shutter has designed shutter sequences to maximize certain criteria on the MTF, targeting a particular (constant) velocity.

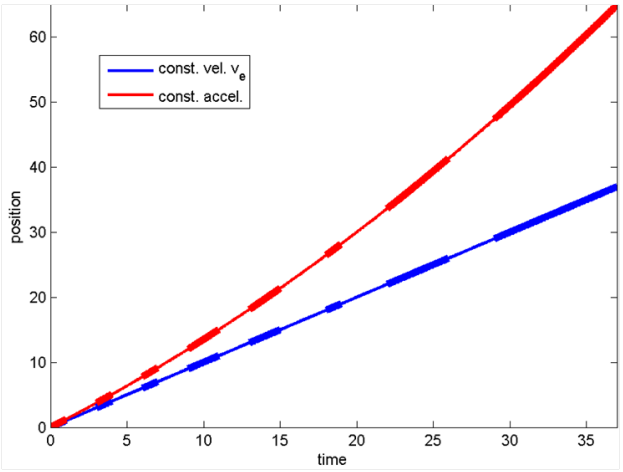

Figure 2. Motion curves for constant velocity (blue) and constant acceleration (red). Relative to a particular shutter timing sequence, the open shutter periods (thicker segments) happen at different positions (given in pixels relative to the location at the start of exposure). This leads to different PSFs, and the optimality of the intended PSF is not achieved.

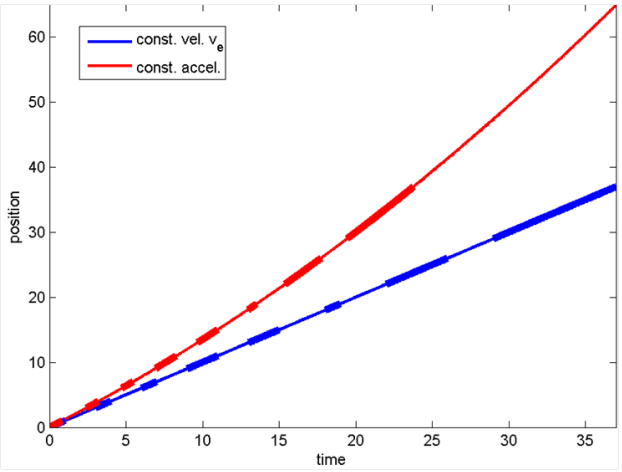

Figure 3. Motion curves for constant velocity (blue; $v_e = 1.2$ pixels/ms) and constant acceleration (red; $v_0 = 1.2$ pixels/ms, $a = 0.03$ pixels/ms). By modifying the times associated with each change in the shutter state, the open shutter periods appear at the same positions of the PSF despite the presence of acceleration. However, the total exposure time is reduced by approximately 30%.

4. Compensating for Acceleration

Constant velocity motion is a special case, so it is natural to investigate what happens when a moving object's motion does not fit this model. In [11], the authors demonstrate that motion estimation from coded exposure images is possible in the case of constant acceleration, but do not address the shutter sequence timing that is needed to ensure invertible deconvolution. When either the camera or object are accelerating, the timing sequence designed to provide invertible deconvolution relative to an expected constant velocity v_e will produce a different PSF that, even if properly estimated, might not be numerically stable for de-convolution. Figure 2 illustrates the differences arising from non-zero acceleration, relative to a fixed shutter sequence.

With respect to the PSF, a positive acceleration from an expected starting velocity v_e will result in both elongation and a decrease - with respect to time - in the PSF value for open shutter periods. Negative acceleration will result in a shorter PSF, and one where the PSF value increases with time. Due to the change in the location of non-zero pixels in the PSF, the de-convolution performance will differ.

Since flutter shutter timing sequences have already been designed to produce a PSF with zero and non-zero entries at particular locations, our approach is reductionist: we start with a constant velocity shutter sequence and compensate for non-zero accelerations in order to produce approximately the same PSF. In particular, we change the shutter timing so that changes from zero to non-zero (and vice versa) take place at the same locations within the PSF. Existing sequences are given as a list of times when the shutter state changes (i.e., opens or closes), and have an expected velocity v_e for which they will produce an optimal PSF. For cases with non-zero constant acceleration, we describe the motion with the initial velocity v_0 and acceleration a. We start from the shutter sequence designed for the value of v_e closest to v_0. For each entry t in the chosen shutter sequence, the time t' corresponding to the same location satisfies

$$v_e \cdot t = v_0 \cdot t' + \frac{1}{2} a t'^2. \tag{3}$$

Solving the quadratic for t', we get

$$t' = \frac{-v_0 + \sqrt{v_0^2 - 2 a v_e t}}{a}. \tag{4}$$

In cases of positive acceleration, applying eq. 4 will result in $t' < t$, which has two effects. The first is that the exposure time of the image is reduced in order to compensate for acceleration, which can be seen in both the position-vs-time plot of Fig. 3 or the PSF comparison in Fig. 4. The extent of exposure reduction depends on the shutter sequence, v_0, and a, but when $v_0 = v_e$ and acceleration results in a 50% increase in velocity during capture, exposure is reduced by 50% or one stop. Given that the changing speed imposes a sloped envelope on the PSF, as seen in Fig. 4 (left), it is natural to wonder whether this effect introduces any lost spatial frequencies. For our running example, Fig. 4 (right) shows that this is not the case, with the primary change being a slight loss of power in low spatial frequencies as a result of decreased exposure.

Another result of compensating for acceleration in this way is that - again, for positive a - the time between shutter

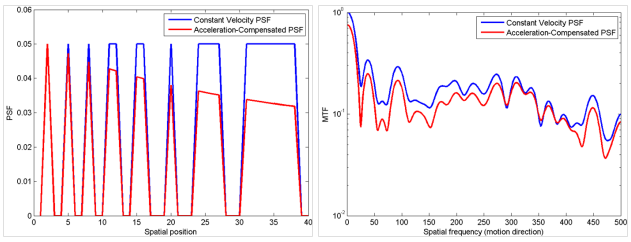

Figure 4. Due to the increased velocity during later parts of coded exposure with acceleration, a decreasing envelope is imposed on the PSF (left). While this reduces exposure somewhat (note the difference in the area under the two PSFs), this change does not give rise to lost spatial frequencies, as seen from the MTF comparison (right).

state changes which take place later in the sequence is reduced. This is important to note because several of the cameras previously used for flutter shutter - namely, the Point Grey cameras using multiple exposure - impose a restriction on the minimum time between open shutter periods, during which charges are shifted and accumulated. This minimum time depends on the camera clock rate and the resolution mode, but the least restrictive mode on a Flea®2 camera (reduced to 800x600 pixel frames) is 1ms. So, in addition to the increased noise level due to multiple reads, the electronic shutter implementation is a significant limitation on the amount of acceleration that can be handled.

5. Reflective Lens Implementation

In order to provide good noise performance, an implementation of the flutter shutter should physically modulate the light striking the sensor (to avoid the need for multiple read electronic shuttering) and do so without reducing the light throughput of the camera. This second condition precludes LCDs, LCoS devices, and the like. Given the availability of physical shutters in DSLR cameras, we investigated the use of that mechanism first. Curiously, despite myriad other improvements, the multiple exposure features common to film-based SLRs have largely been removed from DSLR cameras. More fundamentally, though, we found that the physical shutter mechanisms in DSLR cameras require tens of milliseconds to re-cock, meaning that their use for coded exposure would seriously limit the choice of shutter sequence.

Based on this, our prototype implementation uses a Digital Micro-mirror Device, namely the DLP® LightCrafter™ Evaluation Module from Texas Instruments Incorporated. The typical use of these DMDs, as illustrated in Figure 5, is for projection, where they spatio-temporally modulate light from a constant LED source. In our case, the DMD modulates light coming into a camera, temporally selecting whether the light reaches the sensor. The DMD is a 608-by-684 array of aluminum micro-mirrors with $7.637\mu m$ pitch,

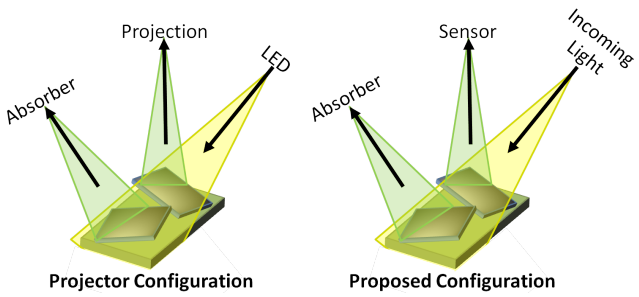

Figure 5. (Left) DMDs are commonly used in projection systems, where the mirrors spatially and temporally modulate light from a constant LED light source, with light either exiting the projector optics or absorbed. (right) In our configuration, we use the DMD to temporally modulate light coming from the scene, directing the light to either the sensor or the absorptive element.

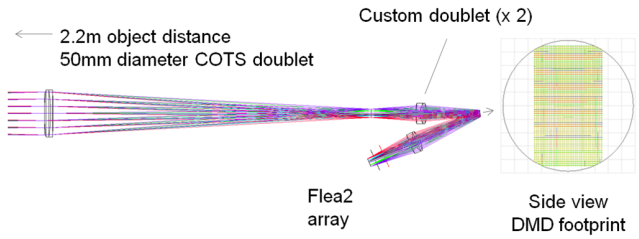

Figure 6. ZMax optical design program output showing our reflected coded exposure lens, incorporating the DMD mirror array in the aperture plane, and with threading for a Point Grey Fles2 camera to act as the sensor.

with each mirror switchable to an angle of ± 12 degrees. As a result, the optical path of our lens is folded with a 24 degree interior angle. In the mode we use the DMD, mirrors can be modulated at 4000Hz.

An optical layout of the lens is shown in Fig. 6, and is based on a modified terrestrial telescope. The lens was designed using a combination of commercial off-the-shelf and custom optical components. The layout was chosen so that the DMD is positioned at the exit pupil in collimated space to allow all field angles to experience the same aperture perturbations from the DMD. The DMD is effectively located at the limiting aperture stop of the system.

Beyond demonstrating the improvements emphasized in this paper, the lens was designed to capture iris imagery for biometric identification. As such, the design was carried out for the near IR waveband used for iris recognition, and the field of view and magnification were chosen to provide sufficient resolution on an iris at 2.2 meters. This optical layout geometrically achieves:

- Diffraction-limited performance
- f/8 and f/14 vertically and horizontally in image space limited by DMD format
- Chromatic correction over 750-860nm waveband

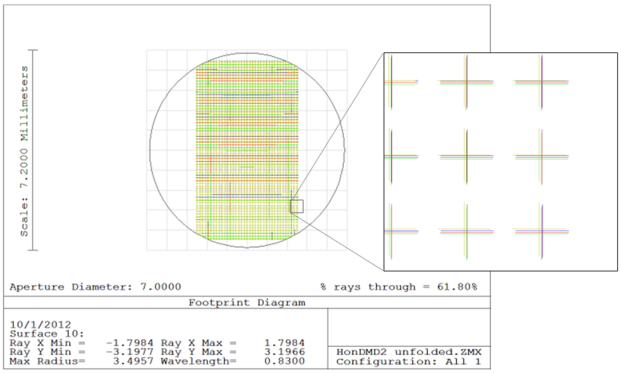

Figure 7. Each field overlaps at the DMD surface to experience the same aperture within a few DMD pixels with low pupil distortion. About $10\mu m$ pupil distortion is shown over a 24mm field of view.

- Object FOV 45.1 x 34mm
- Magnification 12 pix/mm (with 3x3 binning)
- $>60\%$ contrast at 2 line pairs/mm in object space
- $< f/55$ in object space

The objective lens has a 300mm effective focal length (EFL) and is set to focus at 2.2m object distance. A baffle is placed at the internal image plane for stray light reduction. Two equal custom doublets are used for the relay lenses and achieve $\pm 5^o$ field of view and provide enough room to acutely fold the optical axis without mechanical interference or rays tracing back on themselves. The doublets are quasi-telecentric in design and virtually distortion free. The 53mm EFL slightly overfills the vertical extent of the DMD and provides the additional space needed for mechanical mounting and electronics. Light is reflected off the DMD at an angle of 24 degrees toward the Flea®2 camera which is a 7.1456 x 5.3856 mm sensor array with 1624x1224 pixels at 4.4 micron pitch. As shown in Figure 7, the design achieves low distortion across the field of view.

Control of the DMD, and synchronization with the image exposure, is done through a host PC. The PC's RS-232 (serial) ports are used to synchronize the start of the sensor's exposure (through the GPIO pins on the camera) with the shutter sequence, which is implemented by switching the DMD between black (mirrors in the off position) and white (on position) images in the Pattern Sequence Mode of operation. Images are communicated to the host PC via a Firewire connection.

6. Limitations and Assumptions

Due to limited funding for our prototype lens, it is not chromatically corrected into the visible spectrum. As such, experiments were carried out with a laser illuminator and diffuser, but some speckle remains. Adding chromatic correction to the lens does not fundamentally change the contribution of our work, but would add significantly to the cost of the initial lens design.

While our work improves the performance of flutter shutter cameras in important ways, it is not fully general with respect to motion. Though we improve the flutter shutter to handle accelerated motion, we are still limited to linear motion trajectories; as noted elsewhere [7], the linear motion assumption holds up reasonably well for short exposure times. In order to compensate for accelerated motion, we must estimate it before the start of exposure. We use hardware-based estimation of acceleration in the experiments which addresses a fairly narrow case of camera motion, and does not handle the out-of-plane camera rotations handled by recent blind deconvolution work [16, 3]. Image-based estimation of the acceleration of a moving object, by measuring speed from two frames preceding the flutter shutter exposure, could be used with our method.

7. Experimental Demonstration

In this section, we show the improved utility of our coded exposure camera by demonstrating that we can compensate for accelerated motion, and by quantifying the reduced effects of noise with our reflective lens. Due to the need for monochromatic light, as described above, we carry out experiments in a lab. In order to evaluate our performance in the presence of acceleration, we use a robot arm on which we can control motion, and mount the camera on the arm.

7.1. Accelerated Motion Compensation

Beyond the analytical approach explained in section 4, there are two important implementation details around the compensation for non-zero acceleration. The first is related to the estimation of acceleration a used to modify the shutter sequence. In order to demonstrate our method, we use hardware-based estimation, via a three-axis accelerometer. In this sense, our work is related to Joshi *et al*. [4], who use similar inertial measurements to constrain an image-based blur estimation. Our use of these sensors is different in that we use real-time estimates of acceleration *to modify the image capture parameters,* namely by modifying the shutter sequence. For our work, we use the MEMS accelerometers built into the Canon 100mm Macro IS lens, namely the STMicroelectronics LIS344ALH, which retails for $2.60. This sensor has two different settings: one for a $\pm 6g$ range of acceleration values, and another for a $\pm 2g$ range; in our experiments, we use the $\pm 2g$ setting. The accelerometer is attached to our Point Grey camera and its outputs are low-pass filtered and communicated to the host PC via a USB connection. In addition to providing a continuous stream of instantaneous acceleration values for a, we also integrate the accelerometer outputs in order to determine the current

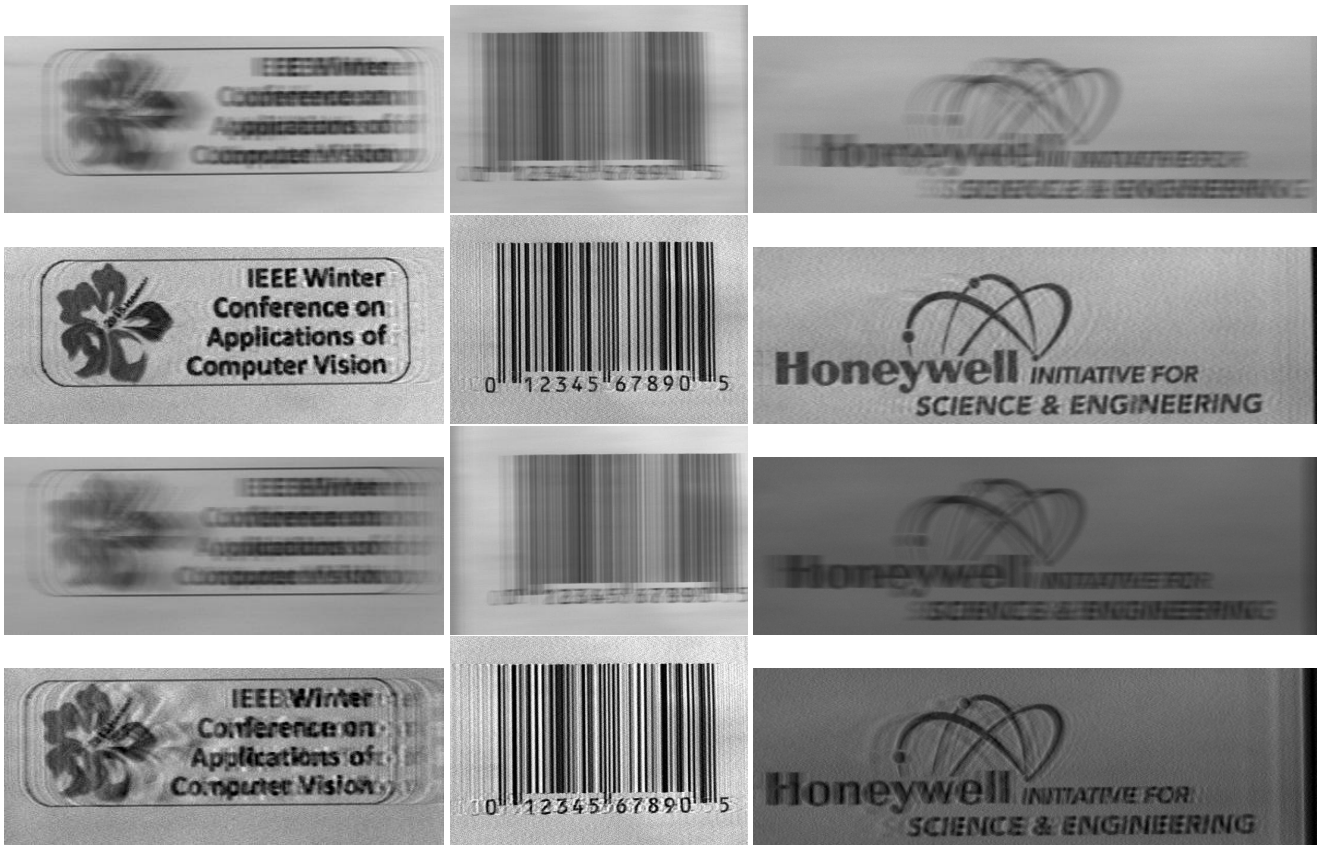

Figure 8. Results of our acceleration compensation, compared to the state of the art. Top row shows the captured coded exposure image using acceleration compensation, and the second row shows the corresponding de-blurred image. The third and fourth rows show the coded exposure image captured without acceleration compensation and the de-blurred result, respectively. The de-blurred images without acceleration compensation have significant artifacts which we reduce. The left column shows camera acceleration with $v_0 = 0.65$ meters per second (left to right) and $a = 1.46m/s^2$. The center column also shows acceleration, with $v_0 = 1.1m/s$ and $a = 0.96m/s^2$. The right column shows deceleration, with $v_0 = -0.29m/s$ (right to left) and $a = 0.15m/s^2$. Overall differences in brightness are due to changes in net exposure time arising from the acceleration compensation; the shutter sequences for each column had a nominal (i.e., before compensation for acceleration) exposure time of 20, 15, and 15ms, respectively.

camera velocity v. This requires dead reckoning to zero the velocity periodically, and to remove the offset due to gravity.

A second key detail is that applying eq. 4 in order to modify the shutter timing sequence takes an amount of time proportional to the number of shutter state changes in the sequence. And, because there is non-zero acceleration during this computation, the value of v_0 must be extrapolated from the instantaneous velocity v at the time exposure is triggered. Empirically, we characterize

$$v_0 = v + kN_\Delta, \tag{5}$$

where N_Δ represents the number of shutter state changes during the exposure sequence.

For the sake of comparison, we evaluate the advantage of acceleration compensation separate from the noise reduction effect, using the same capture hardware for the images shown in Figure 8. In both cases, the shutter sequence was chosen based on the velocity at the time of exposure, from a set of sequences generated using the method of [11]. Images are de-blurred using the reference code provided in [14]. Uncompensated images are de-blurred using several different PSF length values, and the one giving the best-looking image is shown in the figure. For the acceleration compensated images, de-blurring uses a PSF length which is calculated from v_0 and a. The results in Figure 8 show that our method significantly reduces artifacts in the reconstructed images.

As a further point of reference, Figure 9 shows the performance of traditional imaging corresponding to the rightmost column of Figure 8. Because this case has the lowest velocity magnitude, and because the camera is decelerating during image capture, this is the best case (of the three) for this approach. However, the horizontal blur still corresponds to 15 pixels in this case, and the de-blurred image (using the non-blind method of Krishnan and Fergus [6])

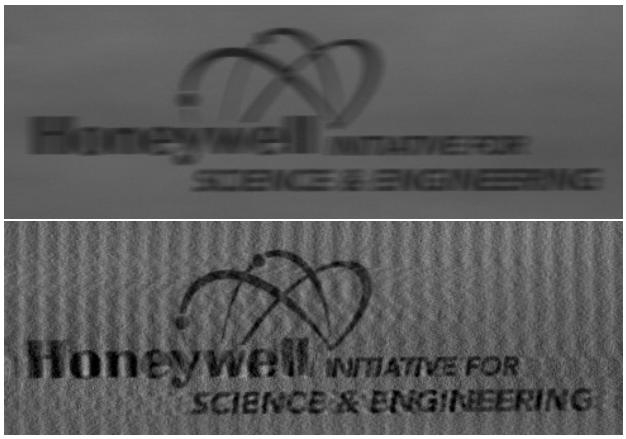

Figure 9. The captured (top) and de-blurred (bottom) traditionally-captured images from the same motion condition as presented in the right-most column of Figure 8. Despite the use of a state-of-the-art non-blind deconvolution method, severe artifacts arise from the ill-conditioned deconvolution.

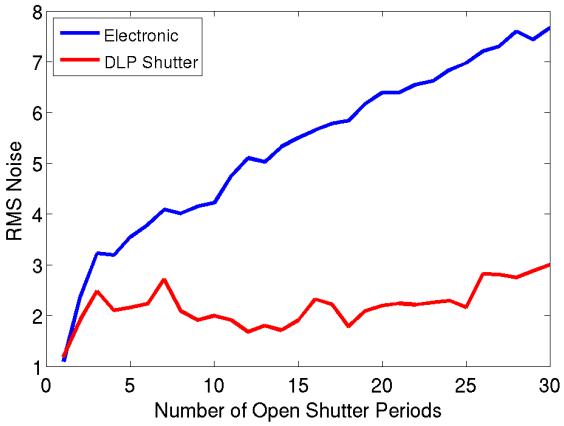

Figure 10. Quantitative comparison between electronic and DMD shuttering implementations. For the electronic shutter implementation, RMS noise in the captured image increases linearly with the number of open shutter periods. For the DMD implementation, the noise level is unrelated to the number of open shutter periods, due to the optical modulation of light.

has severe artifacts.

7.2. Noise Reduction vs. Electronic Shutter

Quantitatively, we measure the noise reduction by capturing several images of a stationary scene. A fixed exposure time (200ms) is divided into an increasing number of open shutter periods, and five images are captured for each number. The noise-free image of the stationary scene is approximated by averaging the 5 images taken with a single open shutter period, and the RMS error is measured between this image and the 5 images taken with 2 to 30 open shutter periods. Results of this are shown in Figure 10, which shows that the electronic shutter implementation has a linear increase in RMS noise as a function of the number of open shutter periods (as previously noted in [8]). Our DMD implementation, however, has no such relationship, with an RMS noise level in the captured image which does not depend on the number of open shutter periods.

Figure 11 shows a comparison of captured and de-blurred images, on the same target moving at the same speed, captured with the same sequence using both the electronic and mirror-based shuttering mechanisms. When viewed electronically at high magnification, the noise reduction is apparent. Numerically, the variance of pixel intensities in a 100-by-100 pixel square of uniform intensity pixels is reduced from 468 to 180 - more than 60%.

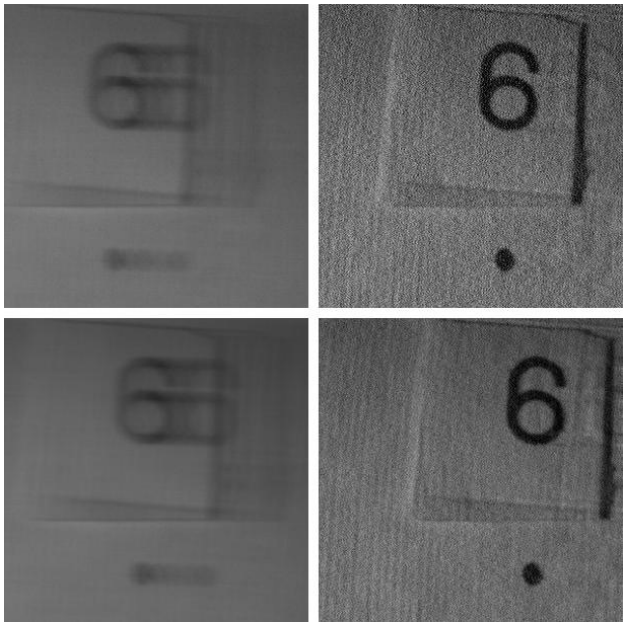

Figure 11. Illustration of noise reduction due to the use of our mirror-based lens. Left and right columns show the captured and de-blurred images, respectively. The top row shows the images from an electronic shutter implementation of coded exposure. The bottom row shows the mirror-based system, using the same shutter sequence on the same object moving with the same velocity. **Best viewed electronically at high magnification.**

8. Conclusions

In order to reduce the severity of noise in coded exposure images, and to expand the space of motions to which it applies, we have prototyped a mirror-based lens which modulates light with MEMS micro-mirrors. We have shown that this improves the noise performance of image capture relative to the most popular (electronic shutter) implementation of coded exposure; the implementation is also significantly more light efficient than LCD-based implementations, due to the removal of polarizing elements from the optical path. We handle motion with non-zero acceleration by modifying the timing sequence of coded exposure in order to pro-

duce a PSF with non-zero elements at the same positions as the constant velocity sequence for which it was designed. These timing sequences, combined with the increased timing flexibility provided by our mirror-based lens, allow us to produce sharp images in cases where previous flutter shutter cameras would fail. In so doing, we believe that flutter shutter cameras are now more likely to be employed in commercial devices where motion-tolerant capture is required.

Acknowledgement

This material is based upon work supported by the U.S. Army RDECOM under Contract Number W911NF-11-C-0253. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the U.S. Government. Flea®2 is a trademark of Point Grey Research, Inc. DLP® LightCrafter is a trademark of Texas Instruments Incorporated. All brand names and trademarks used herein are for descriptive purposes only and are the property of their respective owners.

References

[1] A. Agrawal and R. Raskar. Resolving objects at higher resolution from a single motion-blurred image. In *Computer Vision and Pattern Recognition*, pages 1–8, 2007. 2

[2] A. Agrawal and Y. Xu. Coded exposure deblurring: Optimized codes for PSF estimation and invertibility. In *Computer Vision and Pattern Recognition*, 2009. 1, 2

[3] A. Gupta, N. Joshi, L. Zitnick, M. Cohen, and B. Curless. Single image deblurring using motion density functions. In *Euro. Conf. on Computer Vision*, 2010. 2, 5

[4] N. Joshi, S. B. Kang, C. L. Zitnick, and R. Szeliski. Image deblurring using inertial measurement sensors. In *SIGGRAPH*, 2010. 5

[5] R. Köhler, M. Hirsch, B. Mohler, B. Schölkopf, and S. Harmeling. Recording and playback of camera shake: benchmarking blind deconvolution with a real-world database. In *Euro. Conf. on Computer Vision*, pages 27–40, 2012. 2

[6] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. In *NIPS*, 2009. 6

[7] A. Levin, P. Sand, T. S. Cho, F. Durand, and W. T. Freeman. Motion-invariant photography. In *SIGGRAPH*, 2008. 2, 5

[8] S. McCloskey. Velocity-dependent shutter sequences for motion deblurring. In *Euro. Conf. on Computer Vision*, 2010. 2, 7

[9] S. McCloskey. Temporally coded flash illumination for motion deblurring. In *Int'l Conf. on Computer Vision*, 2011. 2

[10] S. McCloskey, W. Au, and J. Jelinek. Iris capture from moving subjects using a fluttering shutter. In *Proc. of the Fourth IEEE Conference on Biometrics: Theory, Applications, and Systems*, 2010. 1

[11] S. McCloskey, Y. Ding, and J. Yu. Design and estimation of coded exposure point spread functions. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 34(10):2071–2077, Oct. 2012. 1, 3, 6

[12] S. McCloskey, K. Muldoon, and S. Venkatesha. Motion invariance and custom blur from lens motion. In *Int'l Conf. on Computational Photography*, 2011. 2

[13] S. G. Narasimhan, S. J. Koppal, and S. Yamazaki. Temporal dithering of illumination for fast active vision. In *European Conference on Computer Vision (ECCV)*, pages 830–844, 2008. 2

[14] R. Raskar, A. Agrawal, and J. Tumblin. Coded exposure photography: motion deblurring using fluttered shutter. *ACM Trans. on Graphics*, 25(3):795–804, 2006. 1, 2, 6

[15] T. Sonoda, H. Nagahara, and R. Taniguchi. Motion-invariant coding using a programmable aperture camera. In *Asian Conf. on Computer Vision*, 11 2012. 2

[16] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *Int'l Journal of Computer Vision*, 98(2):168–186, 2012. 2, 5

[17] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *Euro. Conf. on Computer Vision*, 2010. 2

[18] W. Xu and S. McCloskey. 2d barcode localization and motion deblurring using a flutter shutter camera. In *Workshop on Applications of Computer Vision*, 2011. 1