Temporally Coded Flash Illumination for Motion Deblurring

Scott McCloskey

Honeywell ACS Labs

1985 Douglas Drive North, Minneapolis, MN, USA

scott.mccloskey@honeywell.com

Abstract

We use temporally sequenced flash illumination to capture coded exposure images of fast-moving objects in low light environments. These coded flash images allow for accurate estimation of blur-free latent images in the presence of object motion. By distributing flashes over a window of time, we lessen eye safety concerns associated with powerful all-at-once flashes. We show how our flash-based coded exposure system has better robustness to increasing object velocity than shutter-based exposure coding, thereby obviating the need for pre-exposure velocity estimation. We also show that the quality of the estimated sharp image is robust to varying levels of ambient illumination. This and other benefits of our coded flash system are demonstrated with real images acquired using prototype hardware.

1. Introduction

We address sharp image acquisition of moving objects in low light environments using active illumination. Using a camera-based flash to stop motion is often considered a solution to this problem, though there are well-known limitations. Because camera-mounted flashes are point light sources, illumination falls off with the square of the distance to an object. The light reflected by the object falls off, as well, with the distance squared to the camera, significantly limiting the effective range of flash photography. Though it may seem straightforward to increase the power of the flash illumination, considerations such as power requirements and flash charging times limit this strategy's effectiveness. More fundamentally, flash power is limited by eye safety, in that very powerful flashes needed to illuminate distant subjects may blind nearby people.

We propose to use a bank of conventional flash modules and to temporally sequence their firings in order to achieve a target level of illumination *without* emitting light at the same time. We capture a single image with coded motion blur by firing the flashes according to a temporal sequence during the acquisition of a single image. An example is
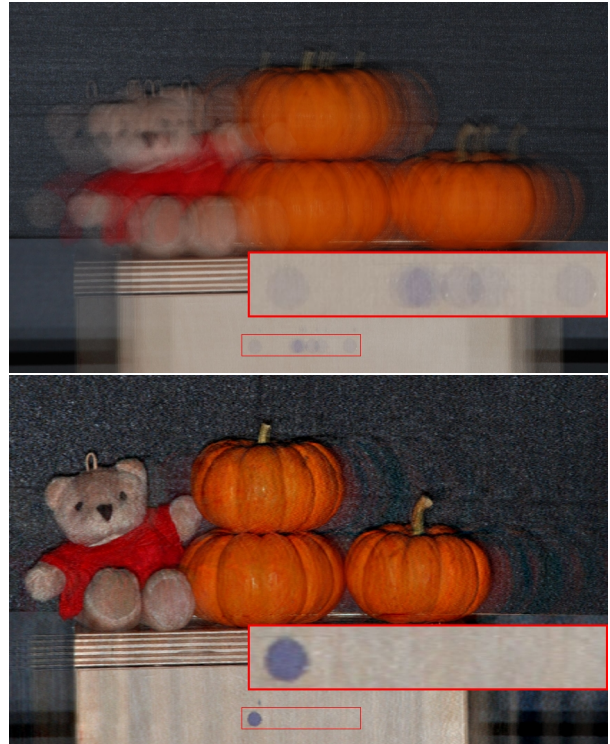

Figure 1. (Top) Image of a moving object taken with temporally coded flash illumination. (Bottom) De-blurred result. Inset regions show the track of a point feature in both images. Because they are dark, these and other images in the paper may not print well.

shown in Fig. 1. The flash sequence timing is determined in order to ensure that the resulting blur point spread function (PSF) can be removed by numerically stable deconvolution. This requires that the modulation transfer function (MTF), which is the absolute value of the Fourier transform of the PSF, does not vanish at any spatial frequency.

As our approach is similar to coded exposure using a fluttering shutter [14], we emphasize important differences between the two. Beyond demonstrating the feasability of flash-based coded exposure, our contributions are:

- We model flash- and shutter-based coded exposure in

order to point out important advantages of our flash system with respect to increased velocity, as described in Sec. 3.

- It has been pointed out in [11] that shutter-based exposure coding requires pre-capture velocity estimation to select an optimal shutter sequence. Since velocity estimation is impractical for flash-based coded exposure, out algorithm selects flash sequences that optimally preserve image content over a range of motions.

- Our flash-based coded exposure uses hardware that is compatible with any camera with a hot-shoe flash, and is demonstrated on the natural motion of human gait.

2. Related Work

No discussion of stop-motion flash photography would be complete without first mentioning the pioneering work of Harold Edgerton, who used strobe flash photography at MIT to capture high speed events illustrating the laws of physics. The approach emits high levels of illumination in a short period of time, unlike our use of stock flashes. More recently, Raskar *et al.* [15] captured multiple images with different flash positions to find depth edges and produced stylized renderings for illustrations. Koh *et al.* [7] extended this work to detect unoccluded objects using the same spatially-distributed flashes. Spatially-varying active illumination is also used in Photometric Stereo [18] for 3D shape reconstruction from multiple photographs.

Motion de-blurring from images acquired with ambient illumination has a long history in image processing and computer vision. The longest standing motion de-blurring problem is that of blind deconvolution [4], where algorithms take only a blurred image as input, from which blur is estimated and removed. Blind motion de-blurring in traditional images remains an active area of research, with recent work by Levin [8], Jia [5], and Shan *et al.* [16].

In addition to work on blind motion de-blurring, there has been a significant amount of work that uses information beyond the image in order to better constrain blur estimation. Yuan *et al.* [19] capture high ISO noisy images in addition to the blurred image. Ben-Ezra and Nayar [1] used a hybrid camera to simultaneously acquire high-resolution/low frame rate and low-resolution/high frame rate videos; the point spread function estimated from the low resolution video is then used to deblur the high resolution video. Joshi *et al.* [6] added inertial sensors to a DSLR to improve estimation of spatially-varying blur.

In recent computational photography work, others have advocated alternative capture techniques that simplify motion de-blurring in various ways. Levin *et al.* [10] capture images while the lens or sensor is translated along a line parallel to subject motion, producing a blur PSF that does not depend on the velocity of a moving subject and thereby obviating blur estimation. Cho *et al.* [2] propose a 2D extension of this method based on two exposures with orthogonal translations. Raskar *et al.* [14] have advocated coded exposure using a fluttering shutter, through which linear, constant-velocity motion produces an invertible PSF that can be removed through deconvolution. Narasimhan *et al.* [13] later demonstrated that coded exposure could be implemented by modulating a projector's output using a DLP device. Though our approach is similar to the flutter shutter, and though we use active illumination, there are critical differences described in the next section.

There are analogous approaches to the above-mentioned methods for handling defocus blur. Whereas coded temporal exposure was used for motion de-blurring, spatially coded apertures have been used by Levin *et al.* [9] and Veeraraghavan *et al.* [17] to simplify defocused image restoration. Whereas sensor motion orthogonal to the optical axis can be used to capture an image with invariant blur over different velocities, Nagahara *et al.* [12] have shown that sensor motion *along* the optical axis can be used to capture images with defocus blur that is invariant to the distance between the camera and objects in the scene.

3. Coded Exposure: Shutter vs. Flash

In order to avoid eye safety issues, our system acquires a single image while it fires a temporal sequence of flashes to distribute illumination over a safe time interval. Because objects in the scene are assumed to be in motion throughout the exposure time, they will appear blurred in the resulting image. As in other work in motion blur removal, we model the coded flash (blurred) image

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

where I is the sharp latent image that we wish to estimate, B is the blur PSF, and η represents noise. The use of this model has a well-known drawback, namely that it only applies within planar regions with the same motion. The model does not apply near occluding contours, or within regions with significant depth variations that lead to motion parallax. For scenes with multiple moving objects, multiple B can be estimated as in [8] by performing motion segmentation and applying de-blurring within segmented regions. The model also assumes that the light reflected by an object in the direction of the camera does not change based on the object's position, i.e. that there are no highly specular surfaces and that shadowing does not change.

When an object is moving laterally with constant velocity, and when image acquisition is done using a traditional shutter, the PSF B is a rectangle function whose size depends on the velocity. This results in reduced image sharpness in the motion direction and smoother intensity transi-
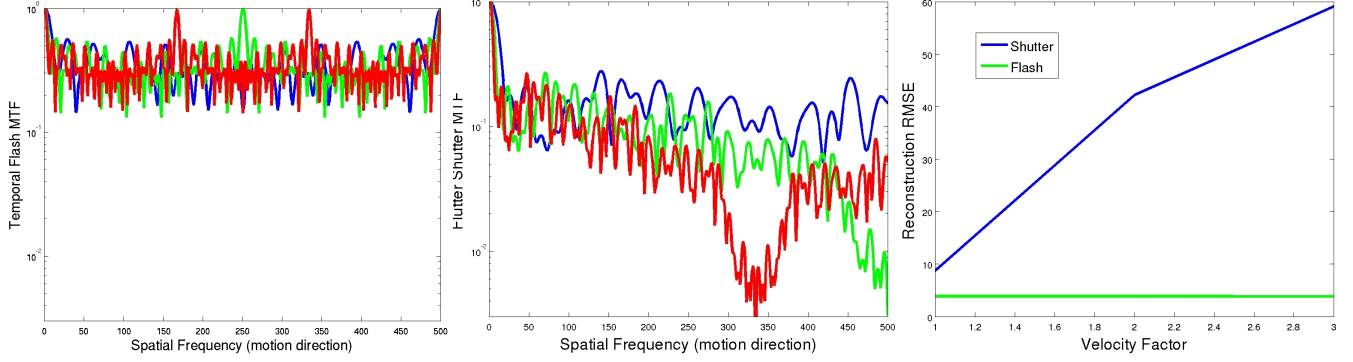
Figure 2. (Left) MTFs of B_N, B_{2N}, and B_{3N} for our flash sequence. (Center) MTFs of S_N, S_{2N}, and S_{3N} for the shutter sequence given in [14]. (Right) RMSE of reconstructed images with simulated blur corresponding to the MTF shown in this figure.

tions, as high frequencies in that direction are muted. Unfortunately, as pointed out in [14], such PSFs are problematic in that the latent image I can't be reliably estimated at certain lost spatial frequencies where the MTF vanishes. One way to avoid such lost frequencies is to flutter the shutter, opening and closing it during the capture of a single image. The sequence of shutter transitions is chosen in such a way as to produce a blur PSF B whose MTF retains high contrast at all spatial frequencies, and which has relatively uniform contrast at middle and high frequencies. In that our flash sequences are chosen to produce an invertible PSF B, our approach is similar to a flutter shutter camera. However, there are key differences between the two.

The first and most obvious difference is that flash-based exposure coding can produce well-exposed images in lower light situations. If the level of light and the camera's settings require Tms of exposure to produce a good image, and if the lens's field of view is d meters at a particular distance, we can't expect to capture a flutter shutter image for objects moving faster than $1000d/T$ meters per second. In fact, since the shutter is closed for as much as 50% of a flutter shutter image capture, the upper bound of speed is reduced by as much as half.

The second key difference has to do with PSF scaling relative to a moving object's velocity. We represent a flutter shutter sequence as a binary sequence of chops

$$C = [C(0)\ C(1)\ ...\ C(N-1)], C(x) \in \{0, 1\}\ \forall x, \quad (2)$$

where N is the number of chops. Whatever the duration of a single chop, an object moving with constant velocity over N pixels during exposure will produce a blur PSF

$$S_N(x) = \begin{cases} \frac{C(x)}{\sum_x C(x)} & \text{for } x = 0, 1, ...N-1, \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The shutter sequence is chosen in order to ensure that this is invertible. However, in [11], it was shown that, when the

object velocity is twice this nominal velocity, the blur PSF

$$S_{2N}(x) = \begin{cases} \frac{C(\lfloor \frac{x}{2} \rfloor)}{2\sum_x C(x)} & \text{for } x = 0, 1, ...2N-1, \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

always has a lost content at the Nyquist frequency, and that additional frequencies are lost for higher object velocities. In short, a given flutter shutter sequence will produce a PSF that is invertible over a small range of velocities. In order to assure that motion blur is invertible, then, it is necessary to estimate an object's velocity in order to select a flutter shutter sequence or chop duration that avoids lost frequencies.

Velocity-related scaling of PSFs from temporal flash sequences, however, is more forgiving. Consider firing N_f flashes in a temporal sequence and assume that the contribution of ambient illumination during exposure is negligible[1]. The PSF will be a superposition of delta functions and, for an object traveling over N pixels during exposure, will be of the form

$$B_N(x) = \begin{cases} \frac{F(x)}{N_f} & \text{for } x = 0, 1, ...N-1, \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where $F(x)$ is 1 if a flash was fired during the corresponding time, and 0 otherwise. For an object traveling over $2N$ pixels during exposure, on the other hand, the PSF is

$$B_{2N}(x) = \begin{cases} \frac{F(\frac{x}{2})}{N_f} & \text{for } x = 0, 2, 4, ...2N-2, \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Whereas values are repeated when a flutter shutter PSF scales in velocity, temporal flash PSFs have zeros interleaved. This is because, while the flutter shutter distributes light evenly when the shutter is opened, individual flashes still stop motion even if they don't provide sufficient exposure. The advantage is that the MTF $|\hat{B}_{2N}|$ corresponding

---

[1]We demonstrate the small effect of ambient illumination in Sec. 6.3.

to the PSF B_{2N} does not suffer from lost frequencies. At any spatial frequency k, it can be shown that

$$|\hat{B}_{2N}(k)| = |\hat{B}_N(2k)|. \qquad (7)$$

That is, if a flash sequence avoids lost frequencies for a given velocity, then it avoids lost frequencies over multiple scales, since $\hat{B}_{2N}(k) = 0 \iff \hat{B}_N(2k) = 0$.

Fig. 2 (left) shows the MTFs of our flash sequence over three scales, and we see that no new minima appear in the MTF as the velocity increases. By contrast, Fig. 2 (center) shows the MTFs of shutter-based coded exposure over three scales; the expected lost spatial frequency appears for S_{2N} and migrates to a lower spatial frequency for S_{3N}. Fig. 2 (right) shows the root mean squared error (RMSE) of a reconstructed image corresponding to these MTFs, including simulated white noise with standard deviation 1. Whereas the RMSE of flutter shutter images degrades with the velocity of a moving object, the RMSE of a reconstructed coded flash image stays nearly constant. It is important to note that, though the scaling of coded flash PSFs do not introduce lost spatial frequencies, the scale of the blur PSF must still be estimated for de-blurring.

Given that Narasimhan *et al.* [13] use a projector with a high-speed mirror to capture coded exposure images, one might expect that it has the same velocity scaling performance as our flash-based system. However, there is a very large difference in the instantaneous light ouput by each: high-end (>US\$100,000) DLP lamps output 30,000 lumens, whereas cheap (<US\$100) flash units output more than 1,000,000. In order to produce good exposure in the cases shown in our experiments, a projector-based system would need to illuminate the scene for about a second. Doing so using short bursts of illumination (needed to get the delta-type PSF) would have prohibitively long acquisition times: several seconds, during which our moving objects would have left the camera's view. With more manageable acquisition times, one would need longer periods of illumination, which give rise to the box-type PSFs that, as with the fluttering shutter, have poor velocity scaling properties.

4. Flash Timing Sequences

Though coded flash PSFs have better scaling properties than flutter shutter PSFs, it is still necessary to wisely select flash timing sequences that produce invertible PSFs. Our objective is to choose a timing sequence so that, in the presence of subject motion, the resulting blur is invertible with de-convolution. This implies not only that we wish to avoid lost spatial frequencies, but also that we maintain high contrast at all spatial frequencies in order to avoid unnecessarily amplifying noise in reconstruction. As such, we adopt the convention of [11] and measure the quality of a particular

MTF as

$$
\begin{aligned}
Q(|\hat{B}|) \quad = \quad & q_1 \, min_k(|\hat{B}(k)|) + q_2 \, var_k(|\hat{B}(k)|) \\
& + q_3 \, mean_k(|\hat{B}(k)|),
\end{aligned} \qquad (8)
$$

to prefer MTFs with the largest minimum contrast (the first term), low variation in contrast (the second term, when q_2 is negative), and high mean contrast (the third term). We use $q_1 = 0.10$, $q_2 = -0.25$, and $q_3 = 3.4$.

There is a disconnect between that which we control - the flash timing sequence - and that which we wish to optimize - the MTF of motion blur. The MTF depends on both the timing sequence and an object's motion, but we have no control over the motion. We represent a flash timing sequence $\mathcal{T}$ as a ordered list of N_f flash firing times $t_1, t_2, ... t_{N_f}$, where t_j represents the time between the beginning of exposure and the firing of the jth flash. Given a flash sequence, along with starting image velocity v_0 (in pixels per second) and acceleration a, we can write the PSF

$$B(x) = \frac{1}{N_f} \sum_{j=1}^{N_f} \delta\left(x - \left(v_0 t_j + \frac{a t_j^2}{2} \right) - 1 \right), \qquad (9)$$

where δ is a discrete delta function ($\delta(0) = 1, \delta(x) = 0 \ \forall x \neq 0$). Likewise, the MTF can be written as

$$|\hat{B}(k)| = |\frac{1}{N_f} \sum_{j=1}^{N_f} e^{-i \frac{2\pi}{N} k \left(v_0 t_j + \frac{a t_j^2}{2} + 1 \right)}|. \qquad (10)$$

Unlike the method of [11], we do not assume that a given sequence can be selected based on a pre-capture estimate of an object's velocity and acceleration, in which case it is necessary to choose a timing sequence that produces invertible PSFs over a range of different velocities and accelerations. Instead of producing a timing sequence that is optimal with respect to a particular constant velocity, then, we produce flash timing sequences that produce good MTFs over a range of motions. We define an object's motion $\mathcal{M}$ as its starting velocity v_0 and acceleration a, and a functional $\mathcal{B}$ mapping a flash sequence $\mathcal{T}$ and motion $\mathcal{M}$ to the resulting MTF, using eq. 10. Given an expected distribution of object motions, and corresponding probabilities for each, we evaluate the overall quality of a *flash sequence* as the probabilistically-weighted qualities of the MTFs produced under different motions, as

$$\mathcal{Q}(\mathcal{T}) = \sum_{\mathcal{M}} Q(\mathcal{B}(\mathcal{T}, \mathcal{M})) P(\mathcal{M}). \qquad (11)$$

If exposure time is quantized into T discrete intervals, there are $\binom{T}{N_f}$ potential flash sequences. Presuming that the first flash is fired at the beginning of exposure, the set of candidate sequences is reduced to $\binom{T-1}{N_f-1}$. Though this search space is still exponential in N_f, it is unlikely

Figure 3. The capture setup used for experiments in this paper. An array of 9 physical flashes is controlled by custom hardware and is triggered by the hot shoe output of a DSLR camera.

that very large numbers of physical flashes would be available, and we have found that this space can be exhaustively searched for $N_f \leq 10$. In searching for the best flash sequence, we restrict the maximum flash firing time so that the PSF is no more than 25% of the width w (in pixels) of the image, in order to ensure that at least 50% of the image can be reconstructed without edge effects. As such, every flash time in a sequence must be less than

$$t_{max} = \frac{\sqrt{v_o^2 + a\frac{w}{2}} - v_0}{a}. \qquad (12)$$

Using this algorithm, and quantizing time into $T = 100$ intervals, we find the optimal flash sequence for $N_f = 7$ with respect to a uniform probability over $v_0 \in \{1, 2, 3\}$ and $a \in \{0, 1\}$. That gives the sequence used in our experiments, $\mathcal{T} =$[0 28 30 32 40 46 67] (ms).

5. Flash Device and Calibration

Fig. 3 shows our flash device used for the experiments described in the next section. The setup consists of an array of 9 flashes connected to a controller that triggers the individual flashes in sequence. Though nearby, the different positions of the flashes will give rise to flash-dependent shadowing. We note that, when objects are far from the camera, such lighting effects will be minimized. The image in Fig. 1 is captured by an off-the-shelf DSLR camera, with

the sequence triggered by the camera's hot shoe flash output. For others, flashes are triggered by a computer which synchronizes sequence timing with a FireWire camera.

In early experimentation, we found that - though all nine flashes are the same model and age - different flashes had consistently different output. In order to calibrate these differences, we took several single-flash images of a diffuse, stationary surface in order to calibrate their power. We found that two of the flashes produced significantly less output, and that these deviations produced artifacts in the reconstructed images. As such, the experiments carried out in the next section used only 7 flashes in the sequence given above. In our experiments, the exposure time is 70ms.

6. Experiments

In this section, using real images, we demonstrate:

1. That de-blurred coded flash images provide better detail of an object's appearance than a single traditional flash.

2. That our coded flash system produces high quality images over a range of object velocities.

3. That changing levels of ambient illumination have little impact on the reconstructed image quality.

4. That, despite the assumptions made in our analysis, coded flash can be used to capture sharp images of people walking naturally.

Throughout this section, we will show de-blurred coded flash images. In order to produce these, we manually identified the direction and extent of motion blur, and applied de-convolution using the reference code given in [14].

6.1. Comparing to a Single Flash

Fig. 4 illustrates that, in low light situations for distant moving objects, a single flash may not put out enough light to produce a well exposed image. The use of seven simultaneous flashes may cause damage to nearby viewers' eyes, and was not tested. Instead, we capture a coded flash image and estimate the latent sharp image, from which we can make out that the objects are two pennies. Despite the fact that the pennies have specular reflectance, we can still make out image details that would not otherwise be evident.

6.2. Object Velocity Scaling

In Sec. 3 we show that, because of the scaling properties of coded flash PSFs, a given flash sequence can produce a high quality reconstruction over a wider range of object velocities than a flutter shutter sequence. To demonstrate this, we captured coded flash images of the same object moving laterally with three different speeds and de-blurred them using the scaled PSFs corresponding to our flash sequence.
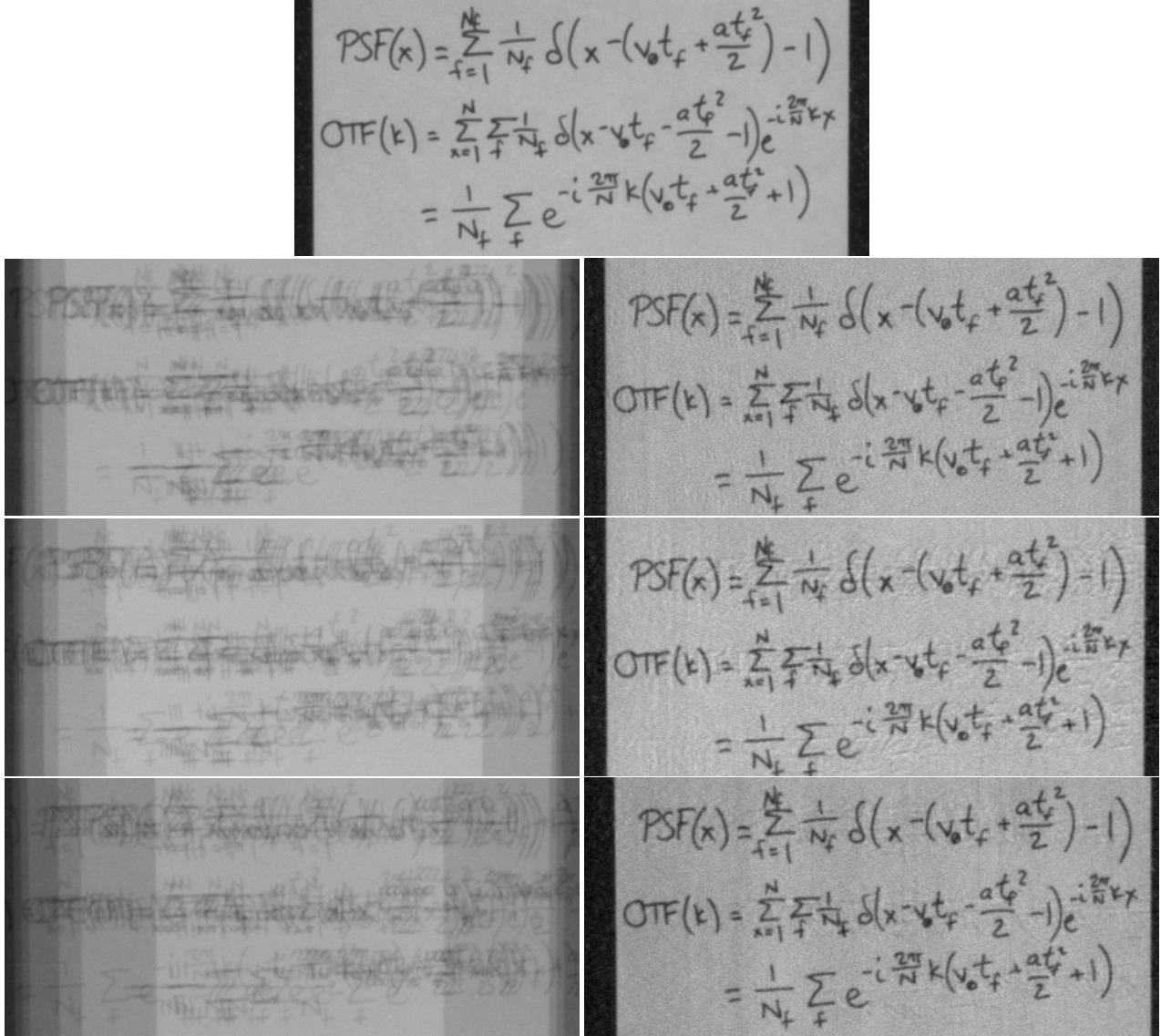
Figure 5. Reconstructed images of an object moving with increasing velocity. (Top row) Reference image of still object. Left column of lower rows show, from top to bottom, coded flash images with 73, 147, and 225 pixels of blur, corresponding to real world speeds of $\approx 0.4, 0.8, 1.2$ m/s for an object at ≈ 3 meters. Right column shows estimated latent images with consistent image quality.

Fig. 5 shows the latent sharp images estimated from these captured images, and a reference image taken without motion. We can see that, while there are slight reconstruction artifacts in the middle image (due to motion jitter), there is no systematic degradation in image quality with increased velocity. The quality of the reconstruction from the image with 225 pixels of blur is qualitatively similar to that of the object moving at a third of that speed, and both have good quality when compared to the stationary image. This confirms the synthetic results presented in Fig. 2.

6.3. Ambient Illumination

In both our analysis and implementation of deconvolution, we have ignored the effects of ambient illumination, and the PSF can be modeled as shown in eq. 9. However, when there is significant ambient illumination, the PSF has the slightly different form

$$B(x) = b + \frac{1}{N_f} \sum_{j=1}^{N_f} \delta \left(x - \left(v_0 t_j + \frac{a t_j^2}{2} \right) - 1 \right), \quad (13)$$

where $b \in [0, 1]$ denotes the relative power of the ambient illumination to the power of a flash.
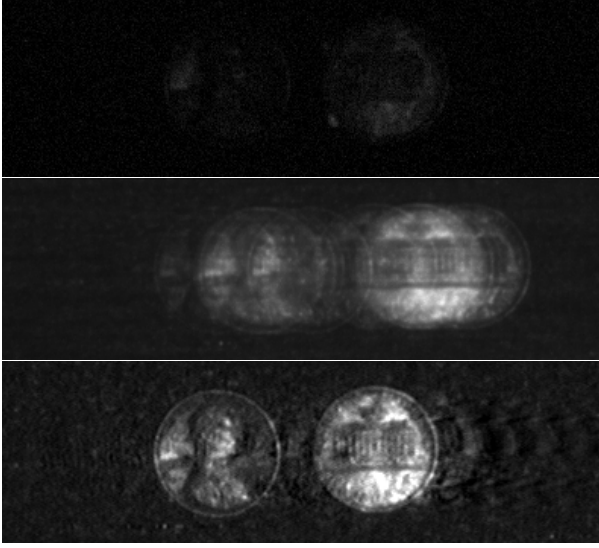
Figure 4. (Top) Image of a moving object taken with a single flash, from which it is difficult to make out the identity of the object. (Middle) Coded flash image of the moving objects. (Bottom) Estimated latent image from the coded flash image, from which we can see that the moving objects are two pennies.

Fig. 6 (top) shows the expected effects of ambient illumination on motion blur MTF assuming ambient power of 0, 10 and 20% of flash power. We see that this has an impact at lower spatial frequencies, in particular an increase in average intensity, but almost none at high frequencies. The images in Fig. 6 shows the effects of ambient illumination in estimated latent images. We can see the expected effects in the low spatial frequencies, particularly the lack of a sharp edge between light and dark blocks of pixels, but little effect in the fine text details.

6.4. Natural Human Motion

Our use of the blur model of eq. 1 implicitly makes a number of assumptions. To repeat, it assumes lateral motion and does not handle occlusions, specular objects, or self-shadowing of 3D objects. Our use of the deblurring code from [14] further restricts us to linear motion with constant velocity. Given these assumptions, readers may wonder if coded flash capture is useful for natural motions of real world objects. We point out that others have demonstrated good results on real world motions using the same assumptions in previous work [14, 10, 8]. We also demonstrate that, even with these assumptions, our coded flash system can estimate a good latent image of a person walking by the camera during exposure. This scenario violates the linearity assumption, in that the person's head moves up and down while moving forward. The person's clothing also casts shadows, and there are a number of specular surfaces (buttons, glasses, etc.) Nonetheless, Fig. 7 shows a good reconstruction of the side of the person's head.
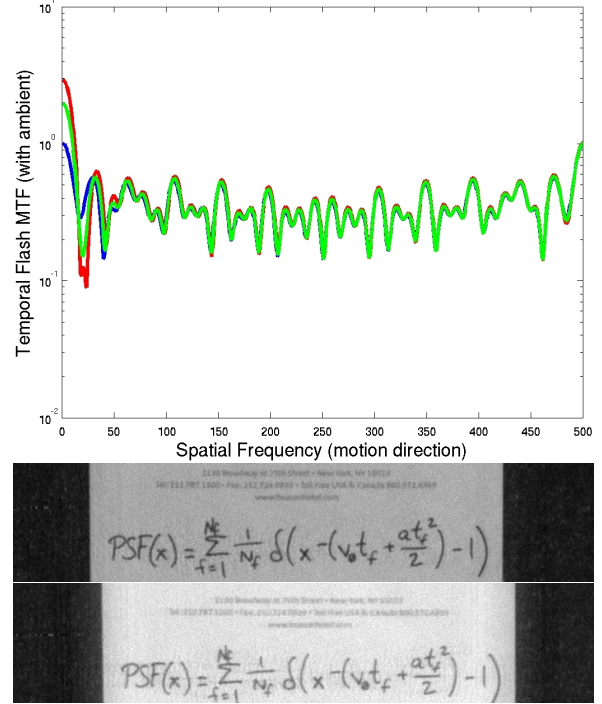

Figure 6. The effects of ambient light levels on coded flash. (Top) Blue/green/red curves show the MTF of our coded flash sequence for blur over 67 pixels with 0/10/20% ambient illumination. (Middle) Latent image region estimated from a coded flash image with negligible ambient illumination. (Bottom) Latent image region estimated from a coded flash image with ambient illumination $\approx 20\%$ of flash power, with slight low-frequency artifacts.

7. Conclusions and Future Work

We have demonstrated that coded flash illumination provides a way of capturing an image from which we can accurately estimate the sharp appearance of moving objects in low light. Though our exposure coding by temporal flashes is similar to coded exposure using a fluttering shutter, we have described how coded flash acquisition has better scaling properties with respect to object velocity. This obviates the need for pre-exposure velocity estimation, as we have demonstrated that the same flash sequence can be used to estimate good latent images over a range of velocities. Though not addressed in the paper, we point out that the spectral differences between ambient and flash illumination should also be considered when comparing these methods.

We have also demonstrated that our system can be used to capture sharp images in the presence of natural human motion, in addition to the linear, constant-velocity motion modeled throughout the paper. Our motivation has been to enable capture of distant moving objects in low light, where current capture methods are limited. By distributing flashes over a window of time we lessen eye safety concerns associated with powerful all-at-once flashes and could re-use
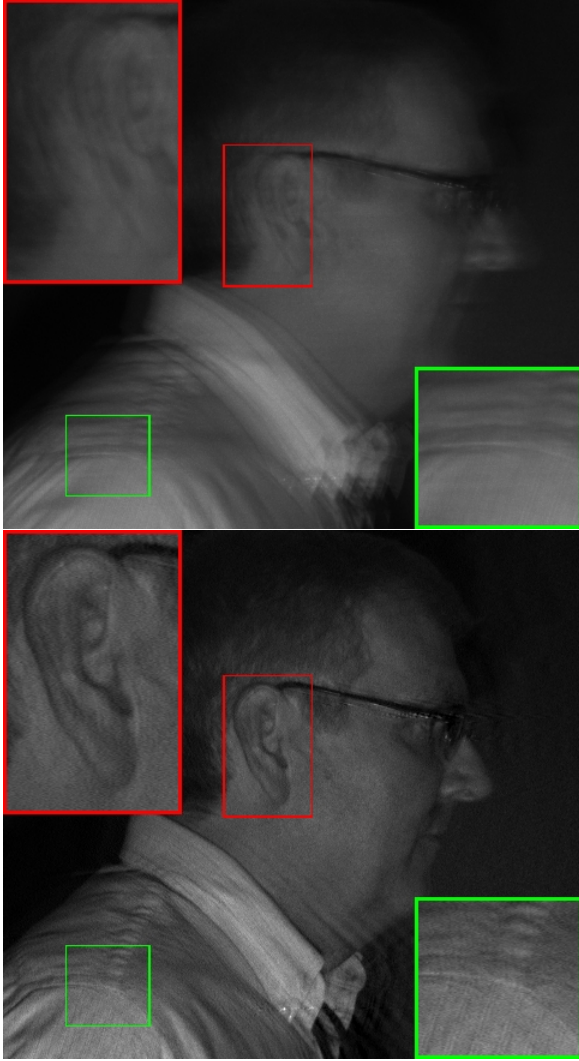
Figure 7. Despite assumptions of the blur model, coded flash can estimate a good quality latent image from an image of a person walking by the camera. Insets show ear and shirt seam regions.

physical flashes if they have short recharge times. Though we are not aware of a precise definition of safe flash illumination, such constraints can be added by rejection sampling during the search for an optimal flash sequence.

Throughout the paper we have used manual blur estimation and de-blurring. In the future we will experiment with a modified version of the coded exposure blur estimation algorithm proposed by Ding *et al.* [3]. It will be necessary to modify the PSF scaling used in blur estimation to reflect the way in which coded flash PSFs scale with velocity, but otherwise the method should apply.

Acknowledgements

References

[1] M. Ben-Ezra and S. K. Nayar. Motion deblurring using hybrid imaging. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 657–664, 2003.

[2] T. S. Cho, A. Levin, F. Durand, and W. T. Freeman. Motion blur removal with orthogonal parabolic exposures. In *IEEE International Conference in Computational Photography (ICCP)*, 2010.

[3] Y. Ding, S. McCloskey, and J. Yu. Analysis of motion blur with a flutter shutter camera for non-linear motion. In *European Conf. on Computer Vision (ECCV)*, 2010.

[4] S. Haykin. *Blind Deconvolution*. Prentice-Hall, 1994.

[5] J. Jia. Single image motion deblurring using transparency. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[6] N. Joshi, S. B. Kang, C. L. Zitnick, and R. Szeliski. Image deblurring using inertial measurement sensors. In *ACM SIGGRAPH*, pages 1–9, 2010.

[7] T. K. Koh, A. Agrawal, R. Raskar, S. Morgan, N. Miles, and B. Hayes-Gill. Detecting and segmenting un-occluded items by actively casting shadows. In *Asian Conference on Computer Vision (ACCV)*, pages 945–955, 2007.

[8] A. Levin. Blind motion deblurring using image statistics. In *NIPS*, pages 841–848, 2006.

[9] A. Levin, R. Fergus, R. Fergus, F. Durand, and W. T. Freeman. Image and depth from a conventional camera with a coded aperture. In *ACM SIGGRAPH*, 2007.

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

[11] S. McCloskey. Velocity-dependent shutter sequences for motion deblurring. In *European Conference on Computer Vision (ECCV)*, 2010.

[12] H. Nagahara, S. Kuthirummal, C. Zhou, and S. Nayar. Flexible Depth of Field Photography. In *European Conference on Computer Vision (ECCV)*, Oct 2008.

[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.

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

[15] R. Raskar, K. Tan, R. Feris, J. Yu, and M. Turk. Non-photorealistic camera: depth edge detection and stylized rendering using multi-flash imaging. In *ACM SIGGRAPH*, 2004.

[16] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. In *ACM SIGGRAPH*, 2008.

[17] A. Veeraraghavan, R. Raskar, A. Agrawal, A. Mohan, and J. Tumblin. Dappled photography: Mask enhanced cameras for heterodyned light fields and coded aperture refocusing. In *ACM SIGGRAPH*, 2007.

[18] R. Woodham. Photometric method for determining surface orientation from multiple images. *Optical Engineering*, 19(1):139–144, January 1980.

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