Motion Deblurring Using Hybrid Imaging

Moshe Ben-Ezra and Shree K. Nayar
Computer Science Department, Columbia University
New York, NY, USA
E-mail: {moshe, nayar}@cs.columbia.edu

Abstract

Motion blur due to camera motion can significantly degrade the quality of an image. Since the path of the camera motion can be arbitrary, deblurring of motion blurred images is a hard problem. Previous methods to deal with this problem have included blind restoration of motion blurred images, optical correction using stabilized lenses, and special CMOS sensors that limit the exposure time in the presence of motion.

In this paper, we exploit the fundamental tradeoff between spatial resolution and temporal resolution to construct a hybrid camera that can measure its own motion during image integration. The acquired motion information is used to compute a point spread function (PSF) that represents the path of the camera during integration. This PSF is then used to deblur the image. To verify the feasibility of hybrid imaging for motion deblurring, we have implemented a prototype hybrid camera. This prototype system was evaluated in different indoor and outdoor scenes using long exposures and complex camera motion paths. The results show that, with minimal resources, hybrid imaging outperforms previous approaches to the motion blur problem.

1. Introduction

Motion blur is the result of the relative motion between the camera and the scene during the integration time of the image. Motion blur can be used for aesthetic purposes, such as emphasizing the dynamic nature of a scene. However, it is usually an undesired effect that has been accompanied photography since its early days and is still considered an open problem that can significantly degrade image quality. Fig. 1 shows examples of images that are blurred by simple, yet different motion paths. In reality, due to the diversity of possible motion paths, every motion blurred image is uniquely blurred. This diversity makes the problem of motion deblurring hard.

Motion blurred images can be restored (up to lost spatial frequencies) by image deconvolution [8], provided that the motion is shift-invariant, at least locally, and that the blur function (*point spread function*, or PSF) that caused the blur is known. As the PSF is not usually known, a considerable amount of research has been dedicated to the estimation of

(a) Scene   (b) Horizontally blurred

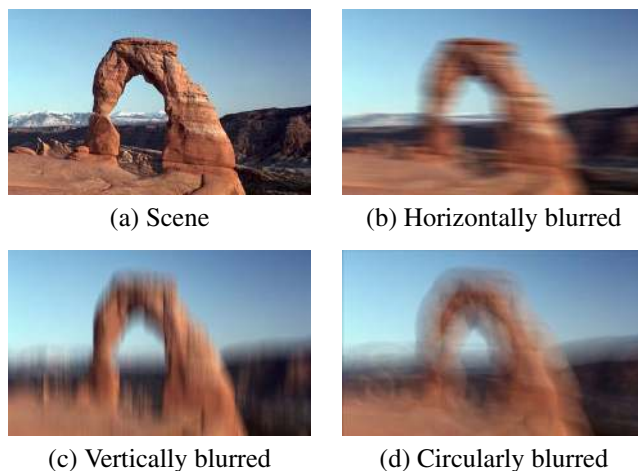(c) Vertically blurred   (d) Circularly blurred

Figure 1: Different camera motions leads to different motion blur. In this example, the unblurred scene shown in (a) is blurred by three different camera motion paths. In (b) and (c) the scene is blurred by linear horizontal and vertical motions respectively; while in (d) the scene is blurred by a circular motion. In reality, the space of possible motion paths is much more diverse, which makes the problem of motion deblurring hard.

the PSF from the image itself [3, 18, 19], or from a sequence of images [2, 9, 16]. This approach, which is called *blind image deconvolution*, assumes that the motion that caused the blur can be parameterized by a specific and very simple motion model, such as constant velocity motion or linear harmonic motion. Since, in practice, camera motion paths are more complex, the applicability of the above approach to real-world photography is very limited.

Two hardware approaches to the motion blur problem, which are more general than the above methods, have been recently put forward. The first approach uses optically stabilized lenses for camera shake compensation [5, 6]. These lenses have an adaptive optical element, which is controlled by gyroscopes, that compensates for camera motion. This method is effective only for relatively small exposures; images that are integrated over durations that are as small as $1/15$ of a second exhibit noticeable motion blur [15, 14]. The second approach uses specially designed CMOS sensors [4, 10]. These sensors prevent motion blur by selectively

stopping the image integration in areas where motion is detected. It does not, however, solve the problem of motion blur due to camera shake during long exposures.

In this paper, we present a novel approach to motion deblurring of an image. Our method estimates the *continuous* PSF that caused the blur, from *sparse* real motion measurements that are taken during the integration time of the image, using energy constraints.

This PSF is used to deblur the image by deconvolution.

In order to obtain the required motion information, we exploit the fundamental tradeoff between spatial resolution and temporal resolution by combining a high resolution imaging device (the *primary detector*) together with a simple, low cost, and low resolution imaging device (the *secondary detector*) to form a novel hybrid imaging system. While the primary detector captures an image, the secondary detector obtains the required motion information for the PSF estimation.

We also address the question of motion analysis of motion blurred imaged, where the constant brightness constraint does not normally hold. This discussion appears, for clarity, in the appendix.

We have conducted several simulations to verify the feasibility of hybrid imaging for motion deblurring. These simulations show that, with minimal resources, a secondary detector can provide motion (PSF) estimates with sub-pixel accuracy. Motivated by these results, we have implemented a prototype hybrid imaging system. We have conducted experiments with various indoor and outdoor scenes and complex motions of the camera during integration. The results show that hybrid imaging outperforms previous approaches to the motion blur problem.

2. Fundamental Resolution Tradeoff

An image is formed when light energy is integrated by an image detector over a time interval. Let us assume that the total light energy received by a pixel during integration must be above a minimum level for the light to be detected. This minimum level is determined by the signal-to-noise characteristics of the detector. Therefore, given such a minimum level and an incident flux level, the exposure time required to ensure detection of the incident light is inversely proportional to the area of the pixel. In other words, exposure time is proportional to spatial resolution. When the detector is linear in its response, the above relationship between exposure and resolution is also linear. This is the fundamental tradeoff between the spatial resolution (number of pixels) and the temporal resolution (number of images per second).

This tradeoff is illustrated by the solid line in Fig. 2. The parameters of this line are determined by the characteristics of the materials used by the detector and the incident flux. Different points on the line represent cameras with different
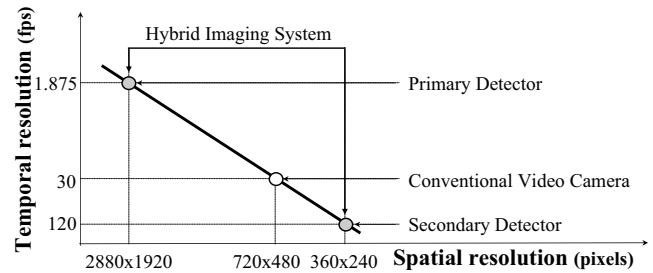

Figure 2: The fundamental tradeoff between spatial resolution and temporal resolution of an imaging system. While a conventional video camera (white dot) is a single operating point on the tradeoff line, a hybrid imaging system uses two different operating points (gray dots) on the line, *simultaneously*. This feature enables a hybrid system to obtain the additional information needed to deblur images.

---

spatio-temporal characteristics. For instance, a conventional video camera (shown as a white dot) has a typical temporal resolution $30 \ fps$ and a spatial resolution of 720×480 pixels.

Now, instead of relying on a single point on this tradeoff line, we could use two very different operating points on the line to *simultaneously* obtain very high spatial resolution with low temporal resolution and very high temporal resolution with low spatial resolution. This type of a hybrid imaging system is illustrated by the two gray dots in Fig. 2. As we shall see, this type of hybrid imaging gives us the missing information needed to deblur images with minimal additional resources.

3. Hybrid Imaging Systems

We now describe three conceptual designs for the hybrid imaging system. The simplest design, which is illustrated in Fig. 3(a), uses a rigid rig of two cameras: a high-resolution still camera as the primary detector and a low-resolution video camera as the secondary detector. Note that this type of a hybrid camera was exploited in a different way in [17] to generate high resolution stereo pairs using an image based rendering approach. In our case, the secondary detector is used for obtaining motion information. Note that it is advantageous to make the secondary detector black and white, since such a detector collects more light energy (broader spectrum) and therefore can have higher temporal resolution. Also note that the secondary detector is used only as a motion sensor; it has low resolution, and high gain and is not suitable for super resolution purposes [1].

The second design uses the same lens for both detectors by splitting the image with a beam splitter. This design, which is shown in Fig. 3(b), requires less calibration than the previous one since the lens is shared, and hence, the image projection models are identical. An asymmetric beam splitter that passes most of the visible light to the primary detector and
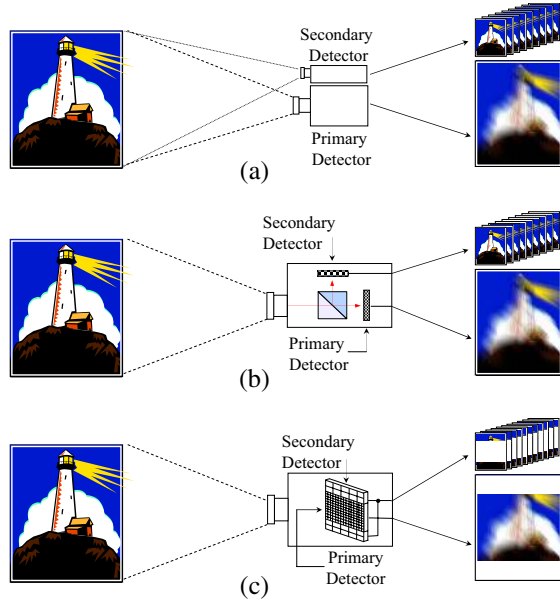
2

Figure 3: Three conceptual designs of a hybrid camera. (a) The primary and secondary detectors are essentially two separate cameras. (b) The primary and secondary detectors share the same lens by using a beam splitter. (c) The primary and secondary detectors are located on the same chip with different resolutions (pixel sizes).

---

reflects non-visible wavelengths toward the secondary detector, for example a "hot mirror" [13], would be preferred.

A third conceptual design, which is illustrated in Fig. 3(c), uses a special chip layout that includes the primary and the secondary detectors on the same chip. This chip has a high resolution central area (the primary detector) and a low resolution periphery (the secondary detector). Note that this chip can be implemented using binning technology now commonly found in CMOS (and CCD) sensors [7]. Binning allows the charge of a group of adjacent pixels to be combined before digitization. This enables the chip to switch between a normal full-resolution mode (when binning is off) and a hybrid primary-secondary detector mode (when binning is activated).

4. Computing Motion

The secondary detector provides a sequence of images (frames) that are taken at fixed intervals during the exposure time. By computing the global motion between these frames, we obtain samples of the continuous motion path during the integration time. The motion between successive frames is limited to a global rigid transformation model. However, the path, which is the concatenation of the motions between successive frames, is not restricted and can be very complex. We compute the motion between successive frames using a multi-resolution iterative algorithm that min-

imizes the following optical flow based error function [11]:

$$\arg\min_{(u,v)} \sum \left(u\frac{\partial I}{\partial x} + v\frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} \right)^2 \qquad (1)$$

where, $\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}, \frac{\partial I}{\partial t}$ are the spatial and temporal partial derivatives of the image, and (u, v) is the instantaneous motion at time t. This motion between the two frames is defined by the following global rigid motion model:

$$\left[\begin{array}{c} u \\ v \end{array} \right] = \left[\begin{array}{ccc} \cos\theta & \sin\theta & \Delta x \\ -\sin\theta & \cos\theta & \Delta y \end{array} \right] \left[\begin{array}{c} x \\ y \\ 1 \end{array} \right] \qquad (2)$$

where $(\Delta x, \Delta y)$ is the translation vector and θ is the rotation angle about the optical axis.

Note that the secondary detector, which has a short but non-zero integration time, may also experience some motion blur. This motion blur can violate the constant brightness assumption, which is used in the motion computation. In appendix A we show that, under certain symmetry conditions, the computed motion between two motion blurred frames is the center of gravity of the instantaneous displacements between these frames during their integration time. We refer to this as the *motion centroid assumption* when estimating the PSF.

5. Continuous PSF Estimation

The discrete motion samples that are obtained by the motion computation need to be converted into a continuous point spread function. To do that, we define the constraints that a motion blur PSF must satisfy, and then use these constraints in the PSF estimation.

Any PSF is an energy distribution function, which can be represented by a convolution kernel $k : (x, y) \mapsto w$, where (x, y) is a location and w is the energy level at that location. The kernel k must satisfy the following energy conservation constraint:

$$\iint k(x, y)\, dx\, dy = 1, \qquad (3)$$

which states that energy is neither lost nor gained by the blurring operation (k is a normalized kernel). In order to define additional constraints that apply to motion blur PSFs, we use a time parameterization of the PSF as a path function $f : t \mapsto (x, y)$ and an energy function $h : t \mapsto w$. Due to physical speed and acceleration constraints, $f(t)$ should be continuous and at least twice differentiable, where $f'(t)$ is the speed and $f''(t)$ is the acceleration at time t. By assuming that the scene radiance does not change during image integration, we get the additional constraint:

$$\int_t^{t+\delta t} h(t)\, dt = \frac{\delta t}{t_{end} - t_{start}},\ \delta t > 0, t_{start} \le t \le t_{end} - \delta t, \qquad (4)$$
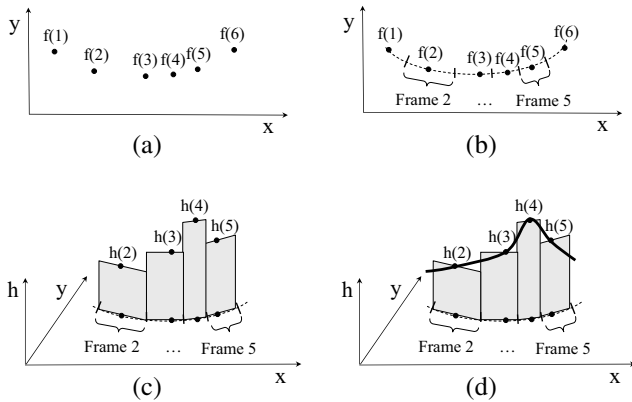
3

Figure 4: The computation of the continuous PSF **from the discrete motion vectors. (a) The discrete motion vectors which are samples of the function** $f : t \mapsto (x,y)$**. (b) Interpolated path** $f(t)$ **and its division into frames by Voronoi tessellation. (c) Energy estimation for each frame. (d) The computed** PSF, $h(t)$**.**

where $[t_{start}, t_{end}]$ is the image integration interval. This constraint states that the amount of energy which is integrated at any time interval is proportional to the length of the interval.

Given these constraints, and the motion centroid assumption from the previous section, we can estimate a continuous motion blur PSF from the discrete motion samples, as illustrated in Fig. 4. First, we estimate the path $f(t)$ by spline interpolation as shown in Fig. 4(a,b); spline curves are used because of their smoothness and twice differentiability properties, which satisfy the speed and acceleration constraints. In order to estimate the energy function $h(t)$ we need to find the extent of each frame along the interpolated path. This is done using the motion centroid assumption by splitting the path $f(t)$ into frames with a 1D Voronoi tessellation, as shown in Fig 4(b). Since the constant radiance assumption implies that frames with equal exposure times integrate equal amount of energy, we can compute $h(t)$ (up to scale) for each frame as shown in Fig. 4(c). Note that all the rectangles in this figure have equal areas. Finally, we normalize (scale) $h(t)$ to satisfy the energy conservation constraint and smooth it. The resulting PSF is shown in Fig. 4(d). The end result of the above procedure is a continuous motion blur PSF that can now be used for motion deblurring.

6. Image Deconvolution

Given the estimated PSF we can deblur the high resolution image that was captured by the primary detector using existing image deconvolution algorithms [8, 12]. Since this is the only step that involves high-resolution images, it dominates the time complexity of the method, which is usually the complexity of FFT. The results reported in this paper

Figure 5: The set of diverse natural images that were used in the simulation tests.

were produced using the Richardson-Lucy iterative deconvolution algorithm [8], which is a non-linear ratio-based method that always produces non-negative gray level values, and hence gives results that make better physical sense than linear methods [8].

7. Simulation Results

Prior to prototype implementation, two sets of simulation tests were done in order to validate the accuracy of the PSF estimation algorithm.

The first set addresses the accuracy of the motion estimation as a function of frame resolution and gray level noise. The second set illustrates the accuracy of the computed path $f(t)$ in the presence of motion blur. Both our tests were conducted using a large set of images that were synthesized from the 16 images shown in Fig. 5.

7.1. Motion Estimation Accuracy Test

In this test, we computed the motion between an image and a displaced version of the same image (representing two frames) using four different resolutions and four different levels of Gaussian noise for each resolution. The displacement used in the test was $(17, 17)$ pixels, and the noise level was varied between standard deviations of 3 gray levels to 81 gray levels. The computed displacements of the downscaled images were scaled back to the original scale and compared with the actual (ground truth) values. Table 1 shows the test results. We can see that sub-pixel motion accuracy was obtained for all tests except the test with the lowest image quality of 80×80 pixels and noise standard deviation of 81 gray levels. This test confirms the feasibility of using a low resolution detector to obtain accurate motion estimates.

Table 1: Scaled motion estimation error between two frames (in pixels) as a function of resolution and noise level. This table shows that it is possible to obtain sub-pixel motion accuracy from significantly low resolution and noisy inputs.

Noise	$\sigma = 3$		$\sigma = 9$		$\sigma = 27$		$\sigma = 81$	
	Error		**Error**		**Error**		**Error**	
Res.	Avg	stdv	Avg	stdv	Avg	stdv	Avg	stdv
640×640	0.01,	0.00	0.01,	0.00	0.02,	0.00	0.04,	0.00
320×320	0.03,	0.00	0.04,	0.00	0.05,	0.00	0.10,	0.01
160×160	0.03,	0.00	0.04,	0.00	0.07,	0.00	0.40,	0.14
80×80	0.13,	0.00	0.21,	0.01	0.39,	0.10	2.60,	4.49

4

7.2. Path Accuracy Test

Here, we first generated a dense sequence of 360 images by using small displacements of each image in the set shown in Fig. 5, along a predefined path. We then created a motion blurred sequence by averaging groups of successive frames together. Finally we recovered the path from this sequence and compared it to the ground truth path. Table 2 shows the results computed over a set of 16 synthesized sequences, for different blur levels and different paths. We can see that sub-pixel accuracy was obtained for all paths. Moreover, the small standard deviation obtained for the different test sequences shows that the different textures of the test images have little effect on the accuracy of the path estimation.

Table 2: Path estimation error, in pixels, as a function of path type and motion blur. We can see that sub-pixel accuracy was obtained for all tests with very little deviation between different test images.

$f(t) =$	$r(\sin t, \cos t)$		$(t, \sin t)$		$(\alpha t^2, \sin t)$	
	Error		**Error**		**Error**	
Blur	**Avg**	**stdv**	**Avg**	**stdv**	**Avg**	**stdv**
8 frames	0.092, 0.0000		0.099, 0.0000		0.299, 0.0001	
16 frames	0.278, 0.0000		0.311, 0.0001		0.610, 0.0004	

8. Prototype Hybrid Camera Results

Fig. 6 shows the prototype hybrid imaging system we have implemented. The primary detector of the system is a 3M pixel (2048×1536) Nikon digital camera equipped with a $\times 6$ Kenko zoom lens. The secondary detector is a Sony DV camcorder. The original resolution of the camcorder (720×480) was reduced to 360×240 to simulate a low resolution detector.

Fig. 7 and Fig. 8 show results obtained from experiments conducted using the prototype system. Note that the exposure times (up to 4.0 seconds) and the focal lengths (up to 884mm) we have used in our experiments far exceed the capabilities of other approaches to the motion blur problem.
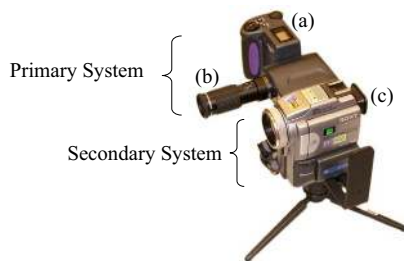

Figure 6: The hybrid camera prototype used in the experiments is a rig of two cameras. The primary system consists of a 3M pixel Nikon CoolPix camera (a) equipped with a $\times 6$ Kenko zoom lens (b). The secondary system is a Sony DV camcorder (c). The Sony images were reduced in size to simulate a low-resolution camera.

In Fig. 7(a) and Fig. 8(a) we see the inputs for the deblurring algorithm, which consists of the primary detector's blurred image, and a sequence of low resolution frames captured by the secondary detector. Figures 7(b) and 8(b) show the computed PSFs for these images. Notice the complex motion paths and the sparse energy distributions in these PSFs. Figures 7(c) and 8(c) show the deblurring results. Notice the details that appear in the magnified rectangles and compare them to the original blurred images and the ground truth images shown in figures 7(d) and 8(d) (that were taken without motion blur using a tripod).

Notice the text on the building shown in the left column of Fig. 8, which is completely unreadable in the blurred image shown in Fig. 8(a), and clearly readable in the deblurred image show in Fig. 8(c). Some increase of noise level and small deconvolution artifacts are observed and are expected side effects of the deconvolution algorithm. Overall, however, in all the experiments the deblurred images show significant improvement in image quality and are very close to the ground truth images.

9. Conclusion

In this paper, we have presented a method for motion deblurring by using hybrid imaging. This method exploits the fundamental tradeoff between spatial and temporal resolution to obtain ego-motion information. We use this information to deblur the image by estimating the PSF that causes the blur. Simulation and real test results show that, with minimal resources, hybrid imaging outperforms previous approaches to the motion blur problem.

Our approach has several application. It can be applied to aerial surveillance systems where vehicle translation, which cannot be corrected by gyro-based stabilization systems, can greatly reduce the quality of acquired images. The method also provides a motion deblurring solution for consumer level digital cameras. These cameras often have small yet powerful zoom lenses, which makes them prone to severe motion-blur, especially in the hands of a non-professional photographer. Since the method is passive, it can be implemented by incorporating a low-cost chip into the camera such as the one used in optical mice. This chip has low spatial resolution and high temporal resolution, which can be used to obtain the egomotion information. The image deblurring process can be performed automatically, or upon user request, by the host computer that is usually used to download the images from the camera. Alternatively, the deblurring function can be incorporated into the camera itself, so that the user always sees images of the highest (motion deblurred) quality. We believe that our proposed method can be applied to various domains of imaging, including, remote sensing, aerial imaging, and digital photography.
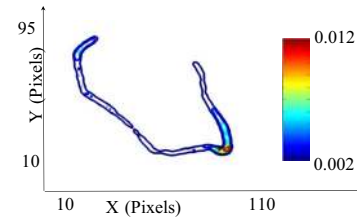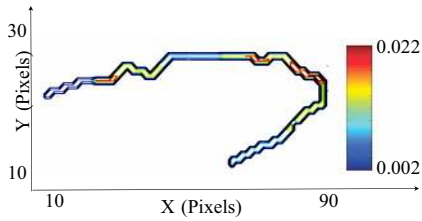
5

Indoor Scene: 3D Objects
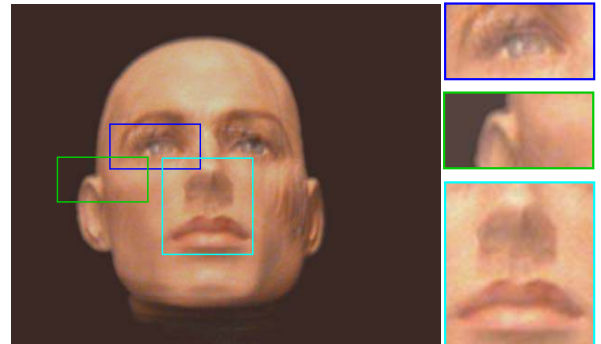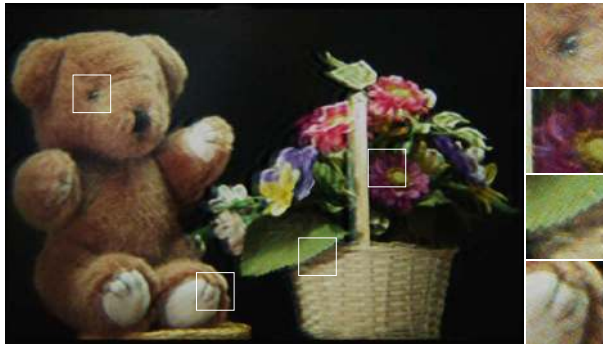**(Focal length = 604mm, Exposure time = 0.5 sec.)**
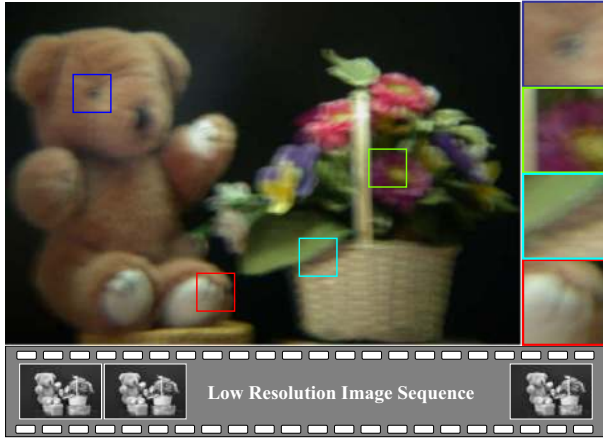
Indoor Scene: Face
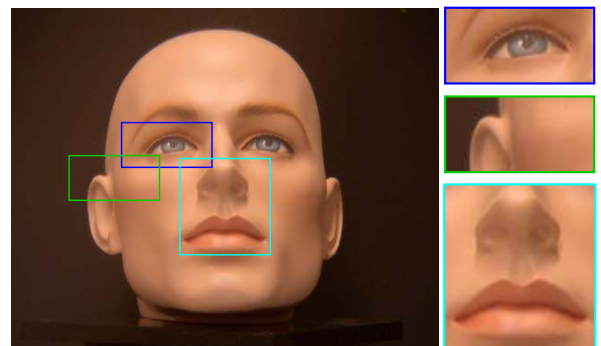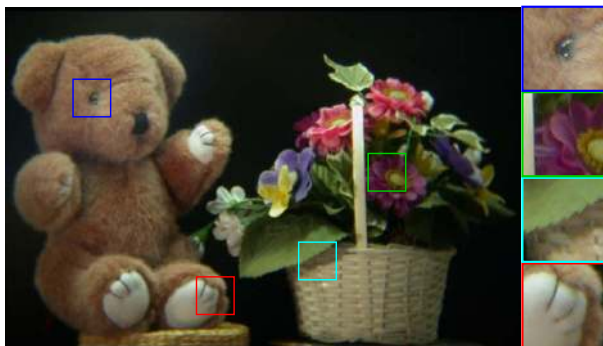**(Focal length = 593mm, Exposure time = 0.5 sec.)**
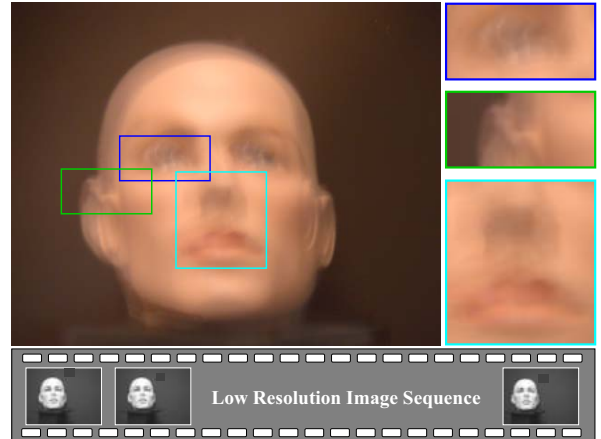
(a) Input images from the primary detector and the secondary detector.

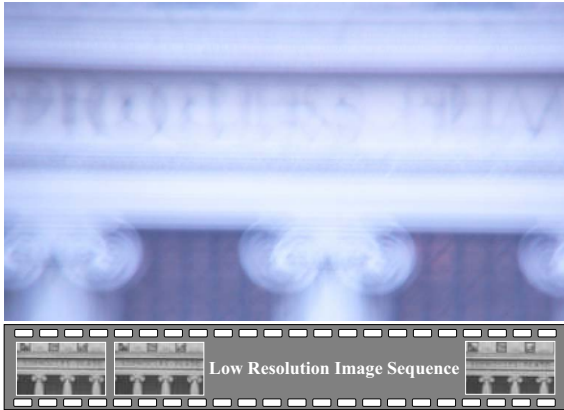**(b) Computed PSFs. Color indicates the energy density.**

(c) Computed deblurred images. Windows show details.

(d) Ground truth images that were taken using a tripod (no motion blur).

Figure 7: Experimental results for indoor scenes. (a) Input images, including the motion blurred image from the primary detector and a sequence of low resolution frames from the secondary detector. (b) The computed PSFs. Notice the complexities of their paths and their energy distributions. (c) The deblurring results. The magnified windows show details. (d) Ground truth images that were captured without motion blur using a tripod.

6

Outdoor Scene: Building
(Focal length = 633mm, Exposure time = 1.0 sec.)

Outdoor Night Scene: Tower
(Focal length = 884mm, Exposure time = 4.0 secs.)

(a) Input images from the primary detector and the secondary detector.

(b) Computed PSFs. Color indicates the energy density.

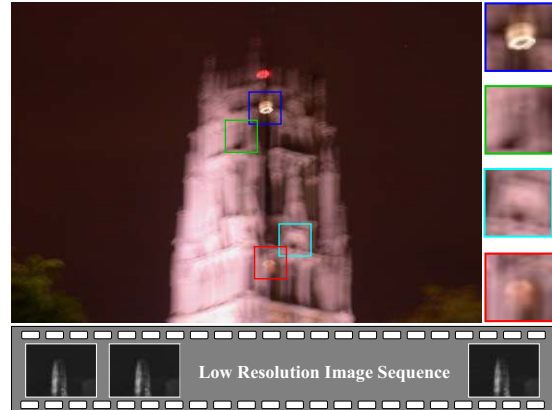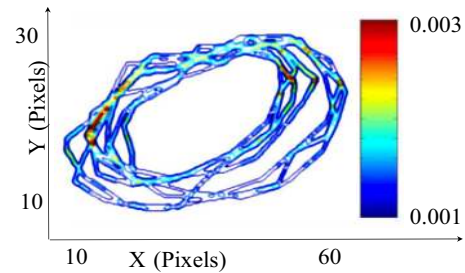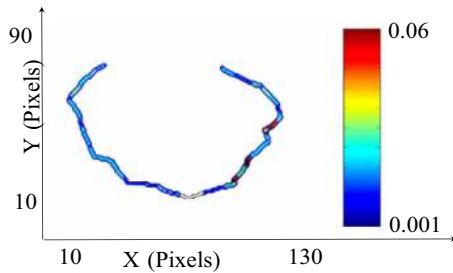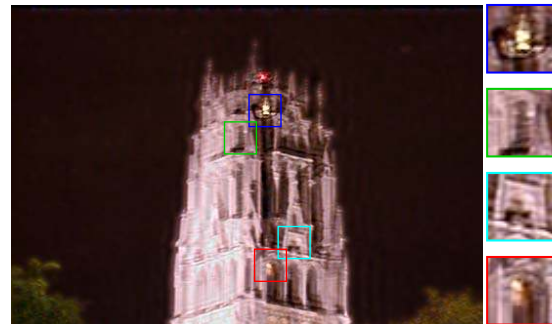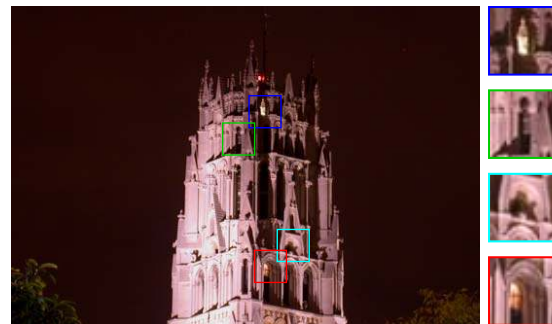(c) Computed deblurred images. Windows show details.

(d) Ground truth images that were taken using a tripod (no motion blur).

Figure 8: Experimental results for outdoor scenes. (a) Input images, including the motion blurred image from the primary detector and a sequence of low resolution frames from the secondary detector. (b) The computed PSFs. Notice the complexities of their paths and their energy distributions. (c) The deblurring results. Notice the clarity of the text and the windows in the left and right deblurred images, respectively. (d) Ground truth images that were captured without motion blur using a tripod.

7

Acknowledgments

References

[1] S. Baker and T. Kanade. Limits on super-resolution and how to break them. *PAMI*, 24(9):1167–1183, September 2002.

[2] B. Bascle, A. Blake, and A. Zisserman. Motion deblurring and super-resolution from an image sequence. *Proceedings of Fourth European Conference on Computer Vision. ECCV '96*, page 573, 1996.

[3] R. Fabian and D. Malah. Robust identification of motion and out-of-focus blur parameters from blurred and noisy images. *CVGIP: Graphical Models and Image Processing*, 53:403, 1991.

[4] T. Hamamoto and K. Aizawa. A computational image sensor with adaptive pixel-based integration time. *IEEE Journal of Solid-State Circuits*, 36:580, 2001.

[5] Canon Inc. www.canon.com/technology/optics/shakecorrect_shift/index.html.

[6] Canon Inc. www.canon.com/technology/optics/vap/content.html.

[7] Canon Inc. www.roper.co.jp/html/tefbin.htm.

[8] Peter A. Jansson. *Deconvolution of Image and Spectra*. Academic – Press, second edition, 1997.

[9] Sang Hwa Lee, Nam Su Moon, and Choong Woong Lee. Recovery of blurred video signals using iterative image restoration combined with motion estimation. *Proceedings of International Conference on Image Processing*, page 755, 1997.

[10] Xinqiao Liu and A. El Gamal. Simultaneous image formation and motion blur restoration via multiple capture. *2001 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings*, page 1841, 2001.

[11] B.D. Lucas and T. Kanade. An iterative image registration technique with anapplication to stereo vision. In *DARPA81*, pages 121–130, 1981.

[12] D.P. MacAdam. Digital image restoration by constrained deconvolution. *JOSA*, 60(12):1617–1627, December 1970.

[13] Edmund Industrial Optics. www.edmundoptics.com/iod/displayproduct.cfm?productid=1492.

[14] Digital Photo Outback. www.outbackphoto.com/reviews/equipment/canon_is_100_400/-canon_is_100_400.html.

[15] Popular Photograpy. www.popularphotography.com/camera/articledisplay.asp?-articleid=59.

[16] A. Rav-Acha and S. Peleg. Restoration of multiple images with motion blur in different directions. *Proceedings Fifth IEEE Workshop on Applications of Computer Vision. WACV 2000*, page 22, 2000.

[17] H.S. Sawhney, Yanlin Guo, K. Hanna, R. Kumar, S. Adkins, and S. Zhou. Hybrid stereo camera: an ibr approach for synthesis of very high resolution stereoscopic image sequences. *Proceedings of SIGGRAPH 2001*, page 451, 2001.

[18] Y. Yitzhaky, G. Boshusha, Y. Levy, and N.S. Kopeika. Restoration of an image degraded by vibrations using only a single frame. *Optical Engineering*, 39:2083, 2000.

[19] Y. Yitzhaky, I. Mor, A. Lantzman, and N.S. Kopeika. Direct method for restoration of motion-blurred images. *Journal of the Optical Society of America A (Optics, Image Science and Vision)*, 15:1512, 1998.

A. Optical Flow for Motion Blurred Images

Consider two frames F and G, which are both motion blurred. We assume for the purpose of analysis only that we have n instantaneous snapshots of the scene that were taken during the integration time of frames F and G. We refer to these as F_i, G_i $1 \leq i \leq n$ and to the optical flow vectors between two corresponding snapshots F_i and G_i as (u_i, v_i). Each point $F_i(x, y)$ satisfies the optical flow constraint equation:

$$u_i \frac{\partial F_i(x,y)}{\partial x} + v_i \frac{\partial F_i(x,y)}{\partial y} + \frac{\partial F_i(x,y)}{\partial t} = 0. \quad (5)$$

For clarity, we shall omit the spatial index (x, y). Adding the equations of all the instantaneous snapshots yields:

$$\sum \left(u_i \frac{\partial F_i}{\partial x} + v_i \frac{\partial F_i}{\partial y} + \frac{\partial F_i}{\partial t} \right) = 0. \quad (6)$$

Approximating the spatial derivatives $(\frac{\partial F_i}{\partial x}, \frac{\partial F_i}{\partial y})$ by convolutions with the derivative kernels $(\Delta x, \Delta y)$, and approximating the temporal derivative $\frac{\partial F_i}{\partial t}$ by $G_i - F_i$ yields:

$$\Delta x \otimes \sum u_i F_i + \Delta y \otimes \sum v_i F_i + \sum G_i - \sum F_i = 0. \quad (7)$$

Without loss of generality, we can assume that $\sum F_i \neq 0$ since adding a constant to F and G does not change any of the derivatives. By multiplying and dividing by $\sum F_i$ we get:

$$\Delta x \otimes \frac{\sum u_i F_i}{\sum F_i} F + \Delta y \otimes \frac{\sum v_i F_i}{\sum F_i} F + (G - F) = 0, \quad (8)$$

which can also be written as:

$$\left(\frac{\sum u_i F_i}{\sum F_i} \right) \frac{\partial F}{\partial x} + \left(\frac{\sum v_i F_i}{\sum F_i} \right) \frac{\partial F}{\partial y} + \frac{\partial F}{\partial t} = 0. \quad (9)$$

Equation (9) is the optical flow constraint for frames with motion blur, where $(u, v) = \left(\frac{\sum u_i F_i}{\sum F_i}, \frac{\sum v_i F_i}{\sum F_i} \right)$. This constraint has little practical use since we do not know the values of the instantaneous motion vectors (u_i, v_i). However, if the distribution of F_i is *symmetrical* with respect to u_i and v_i, meaning that $\sum u_i \cdot (F_i - \bar{F}_i) = 0$, $\sum v_i \cdot (F_i - \bar{F}_i) = 0$, equation (9) can be reduced to:

$$\bar{u} \frac{\partial F}{\partial x} + \bar{v} \frac{\partial F}{\partial y} + \frac{\partial F}{\partial t} = 0, \quad (10)$$

where $\bar{u}_i = \sum u_i / n$, $\bar{v}_i = \sum v_i / n$ are the averages of the instantaneous motion vectors (u_i, v_i).

Note that integration is invariant to the order, and since the instantaneous motions (u_i, v_i) implicitly assumes chronological order, we need to show that the above equations are true for different orderings of the snapshots. This is true since the *sum* of displacements between snapshots does not change by order permutation, and therefore neither does their average.

8