Gyro-Based Multi-Image Deconvolution for Removing Handshake Blur

Sung Hee Park Marc Levoy
Stanford University
shpark7@stanford.edu levoy@cs.stanford.edu

Abstract

Image deblurring to remove blur caused by camera shake has been intensively studied. Nevertheless, most methods are brittle and computationally expensive. In this paper we analyze multi-image approaches, which capture and combine multiple frames in order to make deblurring more robust and tractable. In particular, we compare the performance of two approaches: align-and-average and multi-image deconvolution. Our deconvolution is non-blind, using a blur model obtained from real camera motion as measured by a gyroscope. We show that in most situations such deconvolution outperforms align-and-average. We also show, perhaps surprisingly, that deconvolution does not benefit from increasing exposure time beyond a certain threshold. To demonstrate the effectiveness and efficiency of our method, we apply it to still-resolution imagery of natural scenes captured using a mobile camera with flexible camera control and an attached gyroscope.

1. Introduction

Image blur due to camera shake is one of the main reasons people discard their photos. Camera shake becomes critical in low-light situations, where long exposure times are required, or when the camera motion is amplified by telephoto optics. To reduce the amount of blur caused by camera shake, two approaches are commonly used: *align-and-average* and *image deblurring*. Although they aim to solve the same problem, they are built on different capture strategies. Align-and-average captures multiple blur-free but noisy images using a short exposure time, and merges them after alignment. On the other hand, image deblurring uses a long exposure time to capture a clean but blurry image and recovers the sharp image using deblurring. However, there exists a middle ground between two extreme ends of capture strategy. A set of images can be captured with intermediate exposure time, hence some small amount of blur, and jointly deconvolved to recover the latent image.

In this paper, we propose a multi-image deblurring system that combines two existing techniques: gyroscope-based camera motion estimation and non-blind multi-image deconvolution. A burst of images is captured while the gyroscope data is recorded simultaneously. We use a drift correction method to remove bias from our gyroscope data. By measuring camera motion and the scene as accurately as we can, we improve the robustness of deconvolution.

A key question about any multi-image deblurring method is: for a given total capture time, how many images should be captured, and what should their exposure times be? Shorter exposures suffer from worse read noise and photon shot noise, while longer exposures suffer from increased handshake blur. To model this tradeoff, we define a noise amplification factor γ, which quantifies the increase in noise after multi-image deconvolution is performed. We show that for long exposures, γ grows linearly with exposure time. As a result, increasing exposure time does not help because additional blur cancels the increased signal to noise ratio (SNR) due to less photon shot noise in the longer exposure.

Based on this observation, we compare the performance of align-and-average and multi-image deblurring to find the best capture strategy in various photographic situations. With the help of simulation, we show how performance is affected by factors such as scene brightness, focal length of optics (hence field of view), sensor read noise, exposure time, analog gain, number of shots, and so on. We then formulate guidelines to help users to determine how to capture and process images to get the best output in real-world situations.

2. Related Work

Blind image deblurring methods suffer from lack of information to recover the latent image. Even with the help of multiple images or image priors, blind methods are considered brittle. Joshi *et al*. [3] proposed an effective solution by using an inertial measurement unit to directly measure camera motion. We adopt their method for motion estimation, but with two modifications. First, we only use gyroscope data, not accelerometer data, because rotation is the dominant factor that causes blur [7]. Second, our method performs the drift correction on gyroscope data; otherwise,

accumulated bias causes non-negligible drift when long exposure times are used.

Comparing the capture strategies for camera motion deblurring has been studied by Zhang *et al*. [9]. Their analysis concludes that align-and-average works better than single-image deconvolution. Boracchi and Foi [1] made the similar observation that restoration error of single-image deconvolution stabilizes after reaching a minimum. However, these results are limited to the single-image approach, while we focus on the case when multiple images are used.

3. Image Formation with Camera Motion

The principal drawback of using a short exposure time in low light is image noise. Lengthening the exposure decreases noise, but increases handshake blur. It is therefore important that we define a noise model that allows us to handle handshake blur. We begin by describing an image noise model for when the camera is stable, then we extend it to include the effect of camera motion.

3.1. Image Noise Model

Given analog gain g and exposure time t, with scene brightness Φ, we use a linear imaging model to describe raw pixel values of image Y as the sum of noise-free image X and the Gaussian noise N with standard deviation σ_N. Specifically, the SNR of a captured image Y is written as

$$\text{SNR}(Y) = \frac{X}{\sigma_N} = \frac{c_1 g t \Phi}{\sqrt{c_2 g^2 t \Phi + g^2 \sigma_{r_0}^2 + \sigma_{r_1}^2}}. \quad (1)$$

where c_1 and c_2 are conversion factors, with $\sigma_{r_0}^2$ and $\sigma_{r_1}^2$ representing the variance of readout noise applied before and after analog amplification, respectively [2].

3.2. From Camera Motion to Image Blur

Let's extend this image noise model to incorporate camera motion while assuming no motion of objects in the scene. With camera motion, a point in the scene contributes to multiple pixels along the motion path. We assume also that camera motion is dominated by rotation − a common assumption in image stabilization systems, and valid for sufficiently distant scenes. This allows us to represent the off-axis angular rotation at point $\boldsymbol{r}$ as

$$R_{\boldsymbol{o}}^{\Delta\boldsymbol{\theta}}\{\boldsymbol{r}\} = R^{\Delta\boldsymbol{\theta}}(\boldsymbol{r} - \boldsymbol{o}) + \boldsymbol{o} \quad (2)$$

where $R^{\Delta\boldsymbol{\theta}}$ is a 3-axis rotation matrix for angular rotation $\Delta\boldsymbol{\theta}$, and $\boldsymbol{o} = (o_x, o_y, o_z)^T$ is the rotation center. If a rotation $\Delta\boldsymbol{\theta}(t_s, t_t)$ is applied during time window $[t_s, t_t]$, then jth pixel $\boldsymbol{p}^j = (x^j, y^j)^T$ in the image plane is shifted to $(x^{j\prime}, y^{j\prime})^T$ by the following relation:

$$\alpha(x^{j\prime}, y^{j\prime}, 1)^T = M_f R_{\boldsymbol{o}}^{\Delta\boldsymbol{\theta}(t_s,t_t)}\{M_f^{-1}(x^j, y^j, 1)^T\} \quad (3)$$

where α is an unknown scaling factor and M_f is the camera intrinsic matrix, which depends mainly on focal length f.

Given this model for rotation, the rotation-blurred image Y can be formulated as the sum of the time integration of all geometrically transformed latent images during the exposure window $[t_s, t_t]$ plus noise. We assume a local window of blurry image Y_i around $\boldsymbol{p}^j$ contains uniform blur, which is represented as

$$Y_i = K_i^j \otimes X + N_i \quad (4)$$

where K_i^j is the 2-D blur kernel, $\otimes$ is the 2-D convolution operator and i denotes the input image index when multiple images are available. The kernel K_i^j is specified by the exposure window $[t_s^i, t_t^i]$, which can be expressed in terms of capture parameters. The two ends of the window are defined as $t_s^i = t_s^1 + (i-1)t_{ft}$ and $t_t^i = t_s^i + t_{exp}$, where t_{exp} is the exposure time of a single shot and the frame time t_{ft} is the gap between frames. Note that all entries in K_i^j sum to one. Also, t_s^1, the time when the first exposure starts, is set as the reference time so that relative rotation between input images are preserved in the kernel.

4. Modeling the Performance of Non-Blind Multi-Image Deconvolution

When the blur kernel is available, the latent image can be recovered via non-blind multi-image deconvolution. Modeling the performance of this deconvolution is not trivial because two factors are mixed together: camera shake blur and the scene. Although it is hard to estimate the influence of the scene on deconvolution, modeling the effect of blur is tractable in the non-blind case.

In this section, we derive a noise amplification factor that quantifies the performance of deconvolution based on the characteristics of this blur. Wiener deconvolution [8] is a standard frequency domain method for estimating a latent image X from a blurry image Y_i where the image formation model is given as Equation 4. By minimizing the expectation of squared estimation error $\text{E}\left|\bar{\mathcal{X}} - \mathcal{X}\right|^2$, we get

$$\bar{\mathcal{X}} = \frac{\sum_{i=1}^{n} \frac{1}{\sigma_{\mathcal{N}_i}^2} \mathcal{K}_i^* \mathcal{Y}_i}{\sum_{k=1}^{n} \frac{1}{\sigma_{\mathcal{N}_k}^2} \left|\mathcal{K}_k\right|^2 + 1/\left|\mathcal{X}\right|^2} \quad (5)$$

where $\bar{X}$ is the optimum estimate of X, K_i is the known blur kernel, n is the number of images, $\mathcal{X}$, $\bar{\mathcal{X}}$, $\mathcal{K}_i$, and $\mathcal{Y}_i$ denote the Fourier transforms of X, $\bar{X}$, K_i, and Y_i, respectively, $\left|\mathcal{X}\right|^2$ is the power spectrum of X, $\mathcal{N}_i$ is the white Gaussian noise in the frequency domain with variance $\sigma_{\mathcal{N}_i}^2 = n_u n_v \sigma_{N_i}^2$, where n_u and n_v are the number of pixels in each dimension in the frequency domain. By Parseval's theorem, we can calculate total mean square error (MSE) between X and $\bar{X}$ over the whole image in the

frequency domain as

$$\text{MSE}(\bar{X}) = \frac{1}{n_u^2 n_v^2} \sum_{u,v} \frac{1}{\sum_{k=1}^{n} \frac{1}{\sigma_{N_k}^2} |\mathcal{K}_k|^2 + 1/|\mathcal{X}|^2}. \quad (6)$$

To get some insight into this deconvolution procedure, let's consider the case when all input images are captured with the same exposure time t and analog gain g. In this case, assume that the noise statistics for all images are identical, $\sigma_{N_i}^2 = \sigma_N^2$. If we assume no prior on the image statistics, as is the case for an unbiased optimal estimator, then Equation 6 reduces to

$$\text{MSE}(\bar{X}) = \gamma \frac{\sigma_N^2}{n} \quad (7)$$

where γ is the noise amplification factor defined as

$$\gamma = \frac{1}{n_u n_v} \sum_{u,v} \frac{1}{\frac{1}{n}\sum_{i=1}^{n} |\mathcal{K}_i(u,v)|^2} \quad (8)$$

and corresponding SNR is given by

$$\text{SNR}(\bar{X}) = \frac{\sqrt{n}}{\sqrt{\gamma}} \frac{c_1 g t \Phi}{\sqrt{c_2 g^2 t \Phi + g^2 \sigma_{r_0}^2 + \sigma_{r_1}^2}}. \quad (9)$$

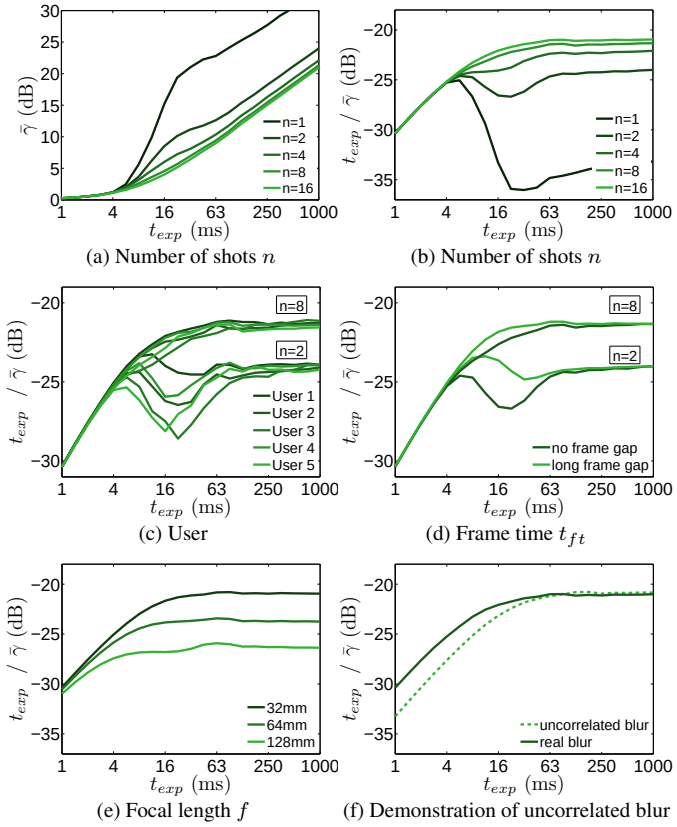
Note that γ quantifies the increase in high frequency noise when average power spectrum of the blur kernels is inverted in the deconvolution, and this factor is only dependent on the characteristics of the blur applied to the system.

When image priors are used in deconvolution, the restoration error depends on the statistics of the scene, and a frequency domain analysis is not possible. When such priors are used, we instead empirically measure performance by estimating MSE in the spatial domain, assuming the ground truth scene is available.

5. Characterization of Real Blur Caused by Camera Motion

In this section we focus on understanding handshake blur in real-world situations. Our main tool for this analysis will be the noise amplification factor defined in Section 4. The main challenge is that the blur is affected by various factors: exposure settings (exposure time, frame-to-frame time, and number of shots), user, type of camera (weight, grip type, optics, and pixel size), and so on. To explore this high-dimensional parameter space, we use simulation to build an empirical model based on real measurements of camera shake.

As a start, we built a database of camera shake by collecting 100 sequences of real camera motion. Each user was asked to hold a tablet steady for a 20-second exposure, while we recorded images and gyroscope streams. This database allows us to create independent samples of camera motion by randomly selecting a small slice from a gyroscope stream. Then, we can simulate various image capture

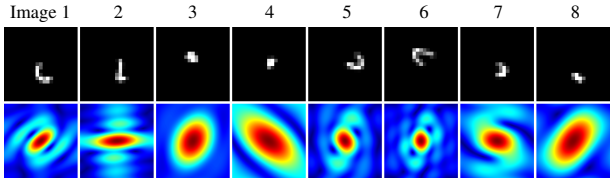


(a) Number of shots n (b) Number of shots n
(c) User (d) Frame time t_{ft}
(e) Focal length f (f) Demonstration of uncorrelated blur

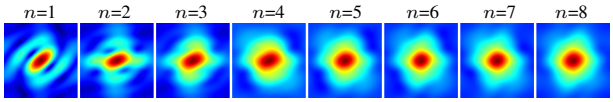
Figure 1: Average noise amplification factor $\bar{\gamma}$ measured from simulation based on real camera motion. Each plot shows how $\bar{\gamma}$ varies when a different parameter is changed as denoted by the plot captions. (a) shows $\bar{\gamma}$ while (b)-(f) plot the ratio between $\bar{\gamma}$ and per-image exposure time t_{exp}. Note that the ratio becomes constant for long exposures in all cases, which means that $\bar{\gamma}$ increases linearly with t_{exp}. The relation comes from the fact that the blurs become uncorrelated. (f) shows that consecutive blur kernels are uncorrelated when t_{exp} is longer than ∼50ms.

scenarios, and evaluate $\bar{\gamma}$, the noise amplification factor averaged over different camera shakes.

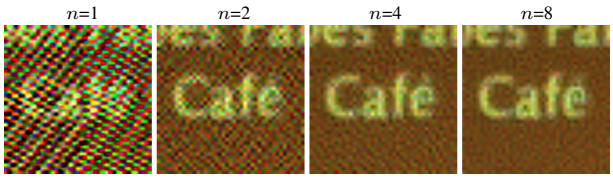
Figure 1 shows how average noise amplification factor $\bar{\gamma}$ is related to exposure time under various combination of parameter settings. When t_{exp} is very small, $\bar{\gamma}$ is close to 1 because no blur exists as shown in Figure 1a. Note that $\bar{\gamma}$ increases with exposure time as image becomes more blurry. However, more images reduces $\bar{\gamma}$ because missing frequency components in an image can be preserved in other images, which is visualized in Figure 2. The important observation is that $\bar{\gamma}$ grows *linearly* with t_{exp} as shown in Figure 1b: the ratio between $\bar{\gamma}$ and t_{exp} remains constant if the exposure time per shot is long enough. This relation is more obvious when n is large, and was consistently observed among different users and independent of the cam-



(a) Blur kernels and their power spectra



(b) Average power spectrum of n images



(c) Multi-image deconvolution applied to n images

Figure 2: An example of n-image deconvolution. (a) The top row shows a set of eight consecutive blur kernels from a single instrumented handheld capture, and the bottom row visualizes their power spectrum. Note that different shapes of blur capture different frequency information. (b) Multi-image deconvolution is applied using n images without any priors, to show the improvement when more images are used. As these power spectra show, adding more images fill in missing frequency components. Although no prior is used, the output images in (c) are not severely degraded by noise when at least two images are used.

era's focal length.

We can explain this trend mathematically for large n and t as follows. Let's define the average power spectrum of blur kernel as

$$\bar{P}_t = \frac{1}{n_i} \sum_i |\mathcal{K}_{i,t}|^2 \tag{10}$$

where $K_{i,t}$ denotes ith sample of blur kernel for exposure time t, $\mathcal{K}_{i,t}$ is the Fourier transform of $K_{i,t}$, n_i is the number of samples, and assume $\bar{P}_t$ converges for large n_i. Let's think of an image as the sum of first and second-half images with the same exposure time of $\frac{1}{2}t$. Then, we can split the blur kernel $K_{i,t}$ into two as

$$K_{i,t} = \frac{1}{2}\left(K_{i_1,\frac{t}{2}} + K_{i_2,\frac{t}{2}}\right) \tag{11}$$

where i_1 and i_2 denotes the index corresponds to first and second-half of i, respectively. Then, the power spectrum of $K_{i,t}$ can be approximated as

$$|\mathcal{K}_{i,t}|^2 \approx \frac{1}{4}\left(\left|\mathcal{K}_{i_1,\frac{t}{2}}\right|^2 + \left|\mathcal{K}_{i_2,\frac{t}{2}}\right|^2\right) \tag{12}$$

if we assume the crosscorrelation term is negligible because consecutive blur kernels become uncorrelated for relatively long t. Then, we get the recursive relation for average power spectrum $\bar{P}_t = \frac{1}{2}\bar{P}_{\frac{t}{2}}$, which leads to $\bar{\gamma}_{n,t} = 2\bar{\gamma}_{n,\frac{t}{2}}$ for large n. This shows the linear relation for large n and t:

$$\bar{\gamma}_{n,t} \propto t. \tag{13}$$

To verify the assumption that blurs are uncorrelated for large t, we evaluate $\bar{\gamma}$ by using the approximation in Equation 12, and compare it with the curve shown in Figure 1b for $n = 16$. Figure 1f shows that they agree well when exposure time is longer than ∼50ms, and proves the assumption. Figure 1d shows another effect that uncorrelated blurs have on $\bar{\gamma}$: uncorrelated blurs reduce $\bar{\gamma}$ compared to correlated ones because they preserve frequency components that are complementary to each other. Longer frame-to-frame time makes the blurs less correlated, which results in smaller $\bar{\gamma}$ shown in Figure 1d. Note that we can generally assume blurs are uncorrelated after a certain time for any camera or any user. Thus, the linear relation in Equation 13 holds for *any* camera shake for large n and t.

	no blur ($\bar{\gamma} = 1$)	large blur ($\bar{\gamma} \propto t$)
read noise dominant	SNR $\propto \sqrt{n}gt$	SNR $\propto \sqrt{n}g\sqrt{t}$
photon noise dominant	SNR $\propto \sqrt{nt}$	SNR $\propto \sqrt{n}$

Table 1: Summary of the performance of multi-image deconvolution in various capture situations.

6. Analysis of Capture Strategies

6.1. Performance of Capture Strategies

With the blur model obtained in Section 5, we evaluate the image quality after multi-image deconvolution is applied. To begin, let's consider the case in which the performance is modeled by the blur applied to system, without assuming any image priors. The relationships between output SNR and various capture parameters can be obtained from Equation 9 and Equation 13, and are summarized in Table 1. The first row in Table 1 assumes that read noise after amplification is dominant. Note that if no blur exists (first column of Table 1), then the analysis reduces to the case of align-and-average. When photon noise is dominant and large blur is involved (lower right cell of Table 1), then the SNR does not improve with exposure time, because additional blur introduced during the exposure cancels the improvement in SNR due to reduced shot noise in input images. In other words, longer exposures are preferable, at least to the point where handshake blur becomes intolerable, and if read noise is dominant, then this switchover point happens at a longer exposure time.

Let us now consider what effect image priors might have

in the analysis. We compare two approaches in removing camera shake: align-and-average (AA) and multi-image deconvolution (MD). We test both cases with and without assuming image priors. To make a fair comparison between AA and MD when image priors are used, we apply an additional denoising step on the output of AA by applying Equation 5 with $\mathcal{K}_i = 1$ and the Gaussian prior to suppress the same amount of noise as in MD. The simulation is done on various scenes, and the performance is evaluated by the peak signal-to-noise ratio (PSNR) which is averaged over many trials with varying camera motions. Two examples are shown in Figure 3, in which we assume eight images are captured with a unity analog gain. We observe that when the same parameters and image prior are used, then MD performs better than AA as shown in Figure 3d and 3f for the Gaussian prior. The improvement in PSNR of MD approaches a horizontal asymptote as exposure time becomes sufficiently long, as discussed in Section 5. Introducing a prior improves the PSNR but it does not significantly change the relation to exposure time as shown in Table 1. As a result, an analysis based on the noise amplification factor can still help estimate the performance of MD even when image priors are used.

6.2. Choosing Capture Strategies

Based on our analysis, a list of guidelines can be obtained, which helps determine the best capture strategy and reconstruction method for various environments. We assume the user has control of n, g, and t_{exp}, and faces variability in scene statistics and dynamic range, and camera noise model, optics, and so on. To understand the effect of each factor, we performed simulation on the *Cafe* scene, changing one axis of the parameter space at a time. Figure 4a shows that focal length strongly affects the range of exposure times in which AA is effective because handshake blur, which remains after AA, is worse for longer focal lengths. This observation gives an upper bound on exposure times appropriate for AA. On the other hand, a lower bound on exposure exists because accurate alignment is predicated on having low noise. AA is expected to work only when these two conditions are satisfied. On the other hand, MD prefers longer exposures to the point at which read noise is negligible and handshake blur is intolerable. This point comes at shorter exposure times if scenes are brighter or more images are captured, as shown in Figure 4c and 4b. In addition, when long exposures are used, t_{exp} should be controlled to avoid over-exposing bright regions.

In many situations, AA is preferred to MD because the former has lower computational cost. A good strategy is to find a parameter set that gives desired image quality with AA, although it may not be possible for certain scenes. Using higher analog gain or capturing more images helps improve the performance of AA, but the choice of g and n is

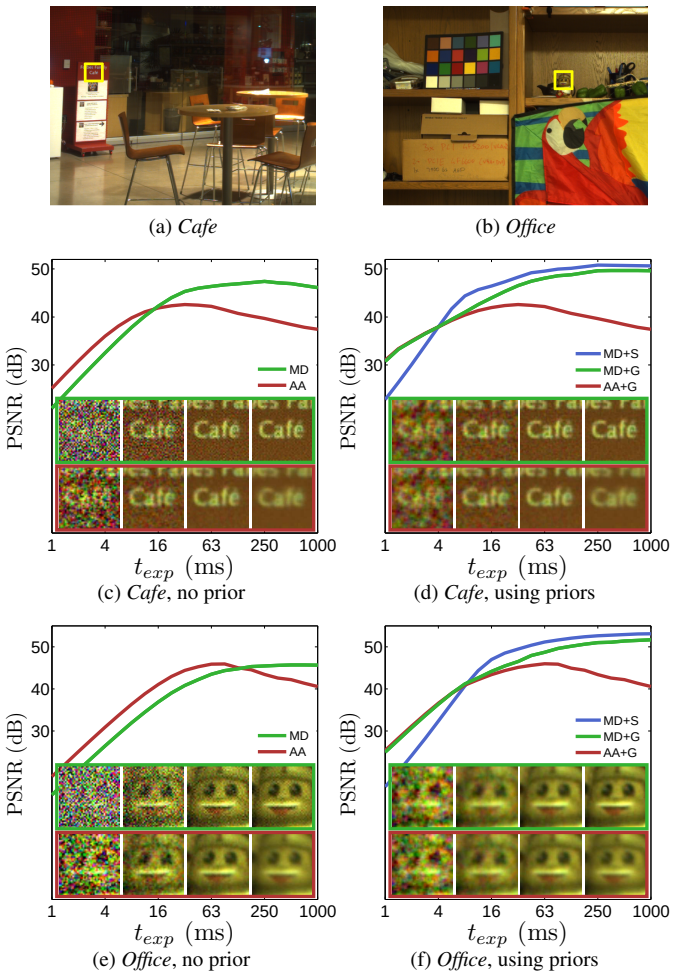
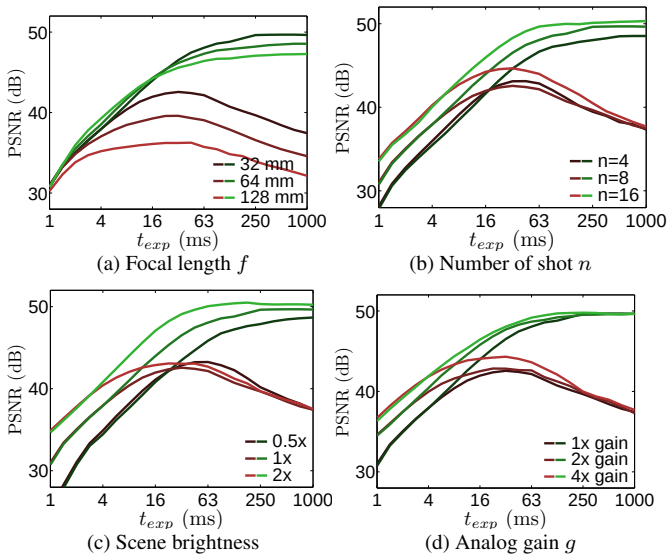


Figure 3: The performance of align-and-average (AA) and multi-image deconvolution (MD) with image priors. Two example scenes are shown in (a)(b). The curves in (c)-(f) are similar to Figure 1, but this time with priors. (c)(e) are obtained with no prior applied, while (d)(f) show the improvement when image priors are employed. The sparse prior (S) shown with the blue curve gives slightly more improvement than the Gaussian prior (G) in green. A small region of output images for different exposure times is shown. The insets in red boxes correspond to the output of AA while green boxes correspond to the output from MD. The performance follows the analysis in Table 1 even when image priors are used.

also bounded by various factors. For example, the effective range of analog gain is restricted by the presence of read noise added before analog amplification. Also, the memory buffer in the camera system, which is used to store captured images before they are merged, has a limited size, restricting n from above. Finally, n is bounded by the total capture time during which the photographer can expect the scene to



(a) Focal length f (b) Number of shot n (c) Scene brightness (d) Analog gain g

Figure 4: The effect of changing each parameter on the performance of align-and-average (red) and multi-image deconvolution (green). Results are from simulation, and all cases assume a Gaussian prior. Focal length strongly affects the range where align-and-average is effective as shown in (a). Other parameters shown in (b)-(d) change the ratio between read noise and photon noise.

hold still. These various limitations restrict the use of AA in some environments, where MD can provide an alternative with better performance.

7. Image Deblurring System

In addition to the foregoing analysis, which was based largely on simulations, we have built a system for capturing image and gyroscope data at high speed and applying our deblurring methods. Our system supports a flexible capture configuration and records images and gyroscope data simultaneously, and our software pipeline performs camera motion estimation, drift correction, image deblurring, and image post-processing.

7.1. Hardware Platform

We implemented our capture application on an NVIDIA Tegra 3 Android developer tablet. We modified the tablet to rigidly attach an Atmel UC3-A3 Xplained and Invensense MPU-6050 sensor board to obtain unfiltered raw sensor data. The sensor data is sent to the tablet through a USB connection at a maximum rate of 750 Hz. The tablet captures 5M-pixel raw data at a maximum rate of 4 fps. We employ standard photometric and geometric calibration procedures to find the parameters of the image formation model.

(a) Blur kernels (b) Without correction (c) With correction

Figure 5: The effect of applying gyroscope drift correction on image deblurring. (a) First four blur kernels of eight blurry images are shown. The gap between the red and yellow kernels is due to drift. (b)(c) The output of image deblurring significantly improves when drift is corrected.

7.2. Camera Motion Estimation using Gyroscope

The sensor data from our gyroscope requires additional processing to estimate correct rotational motion described in Section 3.2. In particular, the gyroscope suffers from unstable bias, which appears as drift in the rotational data. Moreover, this drift accumulates with longer exposures. The standard deviation of the samples is $(0.21, 0.12, 0.17)^\circ/s$, which produces more than a pixel of deviation if the total capture time is longer than $\frac{1}{6}s$ for our system. Also, there exists an unknown time delay between image and gyroscope data.

We model the angular rotation by accumulating the incremental changes given as

$$\Delta\boldsymbol{\theta}(t, t+\Delta t) = (\boldsymbol{\omega}(t+t_d) + \boldsymbol{\omega}_d)\Delta t \qquad (14)$$

where $\boldsymbol{\omega}(t)$ is the rate of rotation measured with gyroscope, t_d is the time delay, $\boldsymbol{\omega}_d$ is the bias which causes drift, and Δt denotes the time interval between gyroscope samples. Based on the model, we propose a drift correction algorithm that is applied whenever capture takes place. The algorithm estimates the time delay t_d, the bias $\boldsymbol{\omega}_d$ and the rotation center $\boldsymbol{o}$ by comparing measured camera motion with image data, where $\boldsymbol{o}$ is assumed to be constant during capture and o_z is zero. To begin, we find the input image $Y_{i_{ref}}$ that is observed to be sharpest. This is done by estimating the blur kernel of ith image at the image center and by picking the one that has maximum spatial variance. Then, we find n_c image blocks that contain the region with strong corner response in $Y_{i_{ref}}$ and denote the center of the blocks as $\boldsymbol{c}_j$. We formulate the kernel estimation as the minimization problem in which the objective function is defined as

$$\underset{\boldsymbol{\omega}_d, o_x, o_y, t_d}{\operatorname{argmin}} \sum_{i \neq i_{ref}} \sum_j \|W_i^j (K_{i_{ref}}^j \otimes Y_i - K_i^j \otimes Y_{i_{ref}})\|^2 \qquad (15)$$

where $W_i^j = w_j w_i W^j W_i$ is the weight applied to each block. The weight W^j is the binary mask that selects the local window around $\boldsymbol{c}_j$, W_i reflects the magnitude of gra-

dient in Y_i, w_j is the distance from $\boldsymbol{c}_j$ to the image center and $w_i = |i - i_{ref}|$ is the temporal distance from the sharpest input image. The role of W_i is to avoid flat regions that only contain noise. Also, w_j and w_i are introduced to weight the region that is more affected by the drift and give consistent estimation throughout the camera motion.

The optimization is done by using the coordinate descent. First, we search for $(\boldsymbol{\omega}_d, o_x, o_y)$ in the multi-scale pyramid. Then, t_d is estimated at the finest scale, followed by refining other parameters based on new t_d. Each step usually converges fast within 30 iterations. Figure 5 shows the effect of the correction when eight images are captured in 2s total. The deviation between the red and yellow kernels show that the drift is actually quite significant.

7.3. Image Deblurring

Based on the camera motion estimated with gyroscope, we perform non-blind multi-image deblurring, which is implemented by applying multi-image deconvolution to small image blocks. The scene is divided into 36×24 blocks where each block covers about 2° of field of view, which is assumed to contain uniform blur. Additional margin is added based on the blur size to avoid boundary artifacts.

Multi-image deconvolution is done by minimizing the objective function:

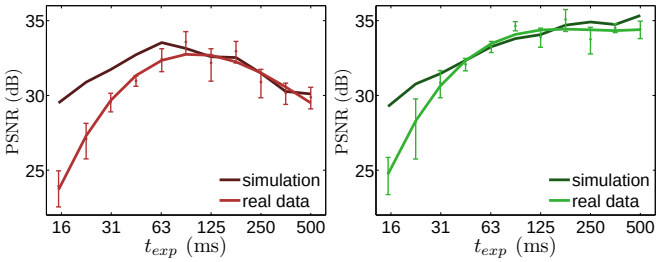
$$\sum_{i=1}^{n} \|Y_i - K_i \otimes X\|^2 + \lambda \|\nabla X\|^\alpha \qquad (16)$$

where λ is the regularization weight and ∇X is the gradients of X. The minimization is done in two ways: frequency domain division using the Gaussian prior and iterative minimization with the sparse prior [5]. Align-and-average is a special case of multi-image deconvolution when no blur is assumed. Our implementation utilizes gyroscope data for image alignment, which allows a fair comparison to deconvolution.

7.4. Handling Practical Issues

Determining the prior weight All deconvolution methods exhibit a trade-off between reducing noise level and recovering a sharp latent image, and this tradeoff is largely controlled by the weight assigned to the prior. When we have a good observation of scene that can be considered as ground truth, for example, when we simulate or have a tripod shot available, we search for the parameter λ in the range $[0.001, 1]$ which gives the smallest MSE with the ground truth. Otherwise, we manually select one.

Rolling shutter correction Most image sensors embedded in mobile devices adopt the electronic rolling shutter, which means that each row in the image is actually captured at slightly different times. Since we capture more gyroscope samples than image frames, we can use a different time slice from our gyroscope stream for each scanline.



(a) Align-and-average (b) Multi-image deconvolution

Figure 6: Verification of the simulation results in Section 6.1 using real images processed by our image deblurring system. The performance matches the simulation for long exposures closely, but when exposure time is short, the low SNR of input images degrades the accuracy of camera motion estimation and hence of deconvolution.

When we generate blur kernels, the exposure time window defined in Section 3.2 is shifted as

$$[t_s^i, t_t^i] + \frac{j_y}{n_y} t_{rs} \qquad (17)$$

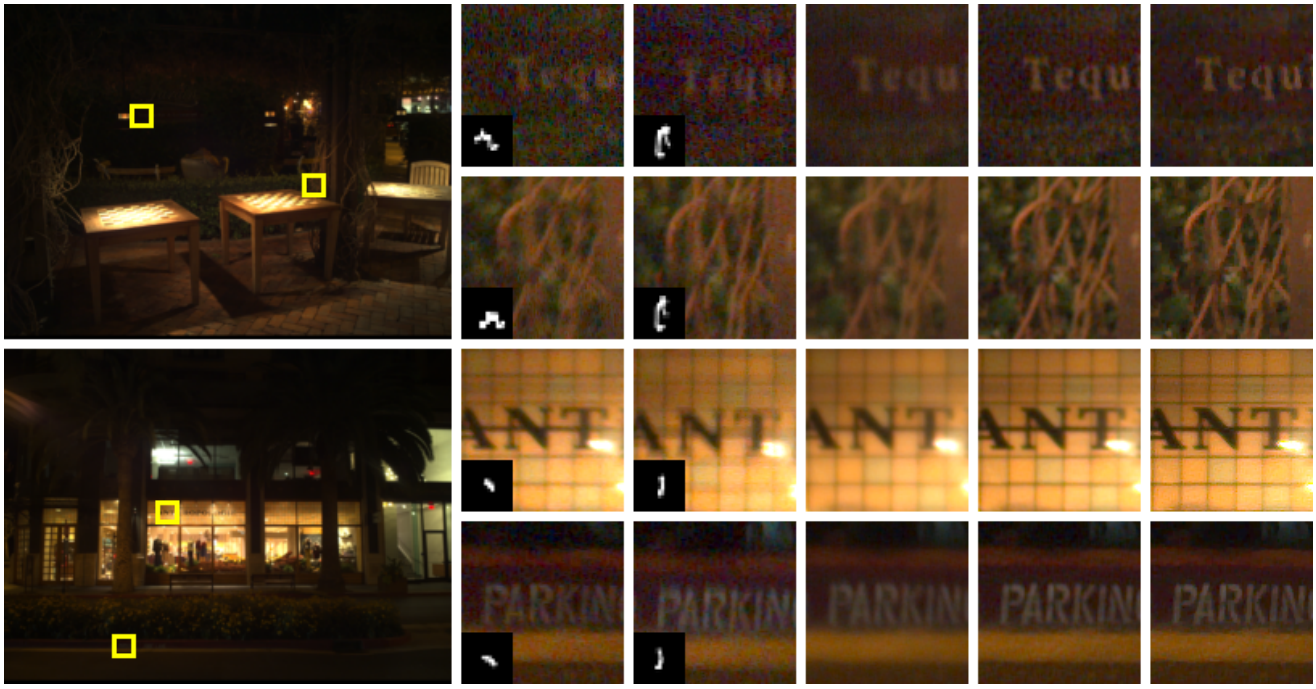
where t_{rs} is the time required to readout whole image, j_y is the image row of pixel j and n_y is the image height [4].

Handling moving objects and over-exposed regions Moving objects and over-exposed regions do not follow our image formation model. After deconvolution, moving objects may suffer from excessive blur and bright regions often cause severe artifacts. Because multiple images are available, these regions can be effectively detected. We introduce an additional image blending operation [6] that merges the denoised reference input image and deconvolved image to hide possible artifacts and give more natural look.

8. Experimental Results

We performed an additional experiment to verify the simulation results in Section 6.1 with real images. With our tablet, eight images are captured from a single burst at 5M-pixel resolution. Each burst is deblurred, and its error is averaged over 20 trials. Figure 6 shows the case in which the Gaussian prior is applied. The results obtained with real data agree with simulation for long exposures, while short exposures show some gap because the low SNR of input images degraded the accuracy of camera motion estimation.

Now we show deblurred images obtained from our system in Figure 7. Two examples are captured with per-shot exposure time of 353ms and 177ms respectively. Our results show clear improvement over any input images, with less blur and noise, and we recover more details compared to align-and-average, with negligible artifacts. The running time for generating a deblurred image in Figure 7 is about 24.5s for the Gaussian prior with our unoptimized CPU im-



(a) Two typical input images (b) (c) Align-and-average (d) Gaussian prior (e) Sparse prior

Figure 7: Deblurring results from our system. Eight images are captured from a single burst at 5M-pixel resolution, while gyroscope data is recorded simultaneously. Each column in (a) and (b) shows two insets of an input image with their blur kernels shown at 2× size. Note the difference in shape of blur in two insets, which demonstrates spatially-varying blur. Input images are jointly deblurred with two different priors. The deblurred images in (d) and (e) removed handshake blur more effectively than in (c), where the input images are aligned and averaged.

plementation. Drift correction takes 10.5s while rest of the time is used for deconvolution. When the sparse prior is used, running time is about 20 minutes.

9. Conclusion

In this paper, we presented an analysis of modeling the performance of multi-image approaches for removing camera shake, in which the model is based on the characteristics of real blur. Guidelines are provided to help users capture better images in real world situations. We also built a multi-image deblurring system that utilizes gyroscope data, and showed its effectiveness with real examples. Our key conclusion is that gyro-based deconvolution improves image quality in very low light, and performs better than align-and-average when read noise is dominant or analog gain has already been maximized.

A limitation of our method is that moving objects in the scene are not deblurred. Such objects can be restored separately using motion deblurring methods. Another interesting direction would be to deliberately vary exposure times among captures. This would allow us to capture scenes of higher dynamic range at the same time we remove handshake blur.

References

[1] G. Boracchi and A. Foi. Modeling the performance of image restoration from motion blur. *TIP*, 21(8):3502–3517, 2012.

[2] S. W. Hasinoff, F. Durand, and W. T. Freeman. Noise-optimal capture for high dynamic range photography. In *CVPR*, 2010.

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

[4] A. Karpenko, D. Jacobs, J. Baek, and M. Levoy. Digital video stabilization and rolling shutter correction using gyroscopes. *CSTR 2011-03, Stanford University, CS*, 2011.

[5] A. Levin, R. Fergus, F. Durand, and W. T. Freeman. Deconvolution using natural image priors. *MIT CSAIL*, 2007.

[6] S. H. Park and M. Levoy. Handling moving objects and over-exposed regions in non-blind multi-image deconvolution. *Stanford Computer Graphics Laboratory Technical Report*, 2014.

[7] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *IJCV*, 98(2):168–186, 2012.

[8] N. Wiener. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. MIT press, 1964.

[9] L. Zhang, A. Deshpande, and X. Chen. Denoising vs. deblurring: Hdr imaging techniques using moving cameras. In *CVPR*, 2010.