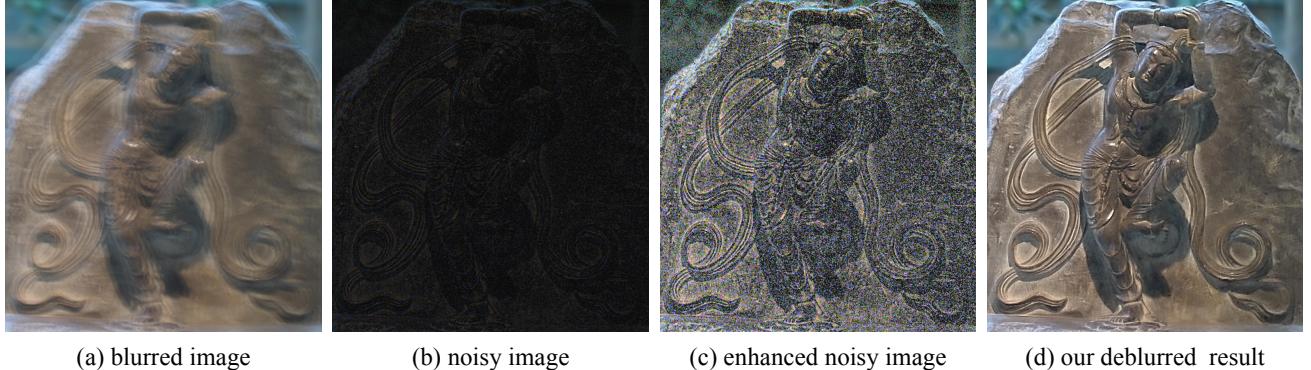


Image Deblurring with Blurred/Noisy Image Pairs

Lu Yuan¹ Jian Sun² Long Quan¹ Heung-Yeung Shum²

¹The Hong Kong University of Science and Technology

²Microsoft Research Asia



(a) blurred image

(b) noisy image

(c) enhanced noisy image

(d) our deblurred result

Figure 1: Photographs in a low light environment. (a) Blurred image (with shutter speed of 1 second, and ISO 100) due to camera shake. (b) Noisy image (with shutter speed of 1/100 second, and ISO 1600) due to insufficient light. (c) Noisy image enhanced by adjusting level and gamma. (d) Our deblurred image.

Abstract

Taking satisfactory photos under dim lighting conditions using a hand-held camera is challenging. If the camera is set to a long exposure time, the image is blurred due to camera shake. On the other hand, the image is dark and noisy if it is taken with a short exposure time but with a high camera gain. By combining information extracted from both blurred and noisy images, however, we show in this paper how to produce a high quality image that cannot be obtained by simply denoising the noisy image, or deblurring the blurred image alone.

Our approach is image deblurring with the help of the noisy image. First, both images are used to estimate an accurate blur kernel, which otherwise is difficult to obtain from a single blurred image. Second, and again using both images, a residual deconvolution is proposed to significantly reduce ringing artifacts inherent to image deconvolution. Third, the remaining ringing artifacts in smooth image regions are further suppressed by a gain-controlled deconvolution process. We demonstrate the effectiveness of our approach using a number of indoor and outdoor images taken by off-the-shelf hand-held cameras in poor lighting environments.

1 Introduction

Capturing satisfactory photos under low light conditions using a hand-held camera can be a frustrating experience. Often the photos

taken are blurred or noisy. The brightness of the image can be increased in three ways. First, to reduce the shutter speed. But with a shutter speed below a *safe shutter speed* (the reciprocal of the focal length of the lens, in the unit of seconds), camera shake will result in a blurred image. Second, to use a large aperture. A large aperture will however reduce the depth of field. Moreover, the range of apertures in many cameras is very limited. Third, to set a high ISO. However, the high ISO image is very noisy because the noise is amplified as the camera's gain increases. To take a sharp image in a dim lighting environment, the best settings are: safe shutter speed, the largest aperture, and the highest ISO. Even with this combination, the captured image may still be dark and very noisy.

Typically, two kinds of degraded image can be taken in the low light conditions. One is a blurred image which is taken with a slow shutter speed and a low ISO setting, as shown in Figure 1(a). With enough light, it has the correct color, intensity and a high Signal-to-Noise Ratio (SNR). But it is blurry due to camera shake. The other is an underexposed and noisy image with a fast shutter speed and a high ISO setting, as shown in Figure 1(b). It is sharp but very noisy due to insufficient exposure and high camera gain. The colors of this image are also partially lost due to low contrast.

Recovering a high quality image from a very noisy image is no easy task as fine image details and textures are concealed in noise. Denoising [Portilla et al. 2003] cannot completely separate signals from noise. On the other hand, deblurring from a single blurred image is a challenging blind deconvolution problem - both blur kernel (or Point Spread Function) estimation and image deconvolution are highly under-constrained. Moreover, unpleasant artifacts (e.g., ringing) from image deconvolution, even when using a perfect kernel, also appear in the reconstructed image.

Deblurring with blurred/noisy image pair has been proposed by Lim and Silverstein [2006]¹. In this paper, we also use a blurred/noisy image pair, but describe an approach that estimates a much more accurate blur kernel and produces a deblurred image with almost no ringing. Like most previous image deblurring approaches, we

¹We thank the reviewers for pointing out Lim and Silverstein [2006]'s work during the rebuttal phase.

assume that the image blur can be well described by a single blur kernel caused by camera shake and the scene is static. Inspired by [Fergus et al. 2006], we convert the blind deconvolution problem into two non-blind deconvolution problems - non-blind kernel estimation and non-blind image deconvolution. In kernel estimation, we show that a very accurate initial kernel can be recovered from the blurred image by exploiting the large scale, sharp image structures in the noisy image. Our approach is also able to handle larger kernels than those recovered by [Fergus et al. 2006] using a single blurred image.

To greatly reduce the “ringing” artifacts that commonly result from the image deconvolution, we propose a residual deconvolution approach. We also propose a gain-controlled deconvolution to further suppress the ringing artifacts in smooth image regions. All three steps - kernel estimation, residual deconvolution, and gain-controlled deconvolution - take advantage of both images. The final reconstructed image is sharper than the blurred image and clearer than the noisy image, as shown in Figure 1(d).

Our approach is practical despite that we require two images. We have found that the motion between two blurred/noisy images, when taken in a quick succession, is mainly a translation. This is significant because the kernel estimation is independent of the translation, which only results in an offset of the kernel. We will describe how to acquire and align such image pairs in Section 7.

2 Previous Work

Single image deblurring. Image deblurring can be categorized into two types: blind deconvolution and non-blind deconvolution. The former is more difficult since the blur kernel is unknown. A comprehensive literature review can be found in [Kundur and Hatzinakos 1996]. As demonstrated in [Fergus et al. 2006], the real kernel caused by camera shake is complex, beyond a simple parametric form (e.g., single one-direction motion or a gaussian) assumed in previous approaches [Reeves and Mersereau 1992; Y. Yitzhaky and Kopeika. 1998; Caron et al. 2002; Jalobeanu et al. 2002]. In [Fergus et al. 2006], natural image statistics together with a sophisticated variational Bayes inference algorithm are used to estimate the kernel. The image is then reconstructed using a standard non-blind deconvolution algorithm. Very nice results are obtained when the kernel is small (e.g. 30×30 pixels or fewer) [Fergus et al. 2006]. Kernel estimation for a large blur is, however, inaccurate and unreliable using a single image.

Even with a known kernel, non-blind deconvolution [Geman and Reynolds 1992; Zarowin 1994; Neelamani et al. 2004] is still under-constrained. Reconstruction artifacts, e.g., “ringing” effects or color speckles, are inevitable because of high frequency loss in the blurred image. The errors due to sensor noise and quantizations of the image/kernel are also amplified in the deconvolution process. For example, more iterations in the Richardson-Lucy (RL) algorithm [H. Richardson 1972] will result in more “ringing” artifacts. In our approach, we significantly reduce the artifacts in a non-blind deconvolution by taking advantage of the noisy image.

Recently, spatially variant kernel estimation has also been proposed in [Bardsley et al. 2006]. In [Levin 2006], the image is segmented into several layers with different kernels. The kernel in each layer is uni-directional and the layer motion velocity is constant.

Hardware based solutions [Nikon 2005] to reduce image blur include lens stabilization and sensor stabilization. Both techniques physically move an element of the lens, or the sensor, to counter-balance the camera shake. Typically, the captured image can be as sharp as if it were taken with a shutter speed 2-3 stops faster.

Single image denoising. Image denoising is a classic problem extensively studied. The challenge of image denoising is how to compromise between removing noise and preserving edge or texture. Commercial softwares, e.g., “NeatImage” (www.neatimage.com) and “Imagenomic” (www.imagenomic.com), use wavelet-based approaches [Simoncelli and Adelson 1996; Portilla et al. 2003]. Bilateral filtering [Tomasi and Manduchi 1998; Durand and Dorsey 2002] has also been a simple and effective method widely used in computer graphics. Other approaches include anisotropic diffusion [Perona and Malik 1990], PDE-based methods [Rudin et al. 1992], fields of experts [Roth and Black 2005], and nonlocal methods [Buades et al. 2005].

Multiple images deblurring and denoising. Deblurring and denoising can benefit from multiple images. Images with different blurring directions [Bascle et al. 1996; Rav-Acha and Peleg 2000; Rav-Acha and Peleg 2005] can be used for kernel estimation. In [Liu and Gamal 2001], a CMOS sensor can capture multiple high-speed frames within a normal exposure time. The pixel with motion replaced with the pixel in one of the high-speed frames. Raskar et al. [2006] proposed a “fluttered shutter” camera which opens and closes the shutter during a normal exposure time with a pseudo-random sequence. This approach preserves high frequency spatial details in the blurred image and produces impressive results, assuming the blur kernel is known. Denoising can be performed by a joint/cross bilateral filter using flash/no-flash images [Petschnigg et al. 2004; Eisemann and Durand 2004], or by a spatio-temporal filter for video sequences [Bennett and McMillan 2005].

Hybrid imaging system [Ben-Ezra and Nayar 2003] consists of a primary sensor (high spatial resolution) and a secondary sensor (high temporal resolution). The secondary sensor captures a number of low resolution, sharp images for kernel estimation. Our approach estimates the kernel only from two images, without the need for special hardware. Another related work [Jia et al. 2004] also uses a pair of images, where the colors of the blurred image are transferred into the noisy image without kernel estimation. But this approach is limited to the case that the noisy image has a high SNR and fine details.

The work most related to ours is [Lim and Silverstein 2006] which also makes use of a short exposure image to help estimate the kernel and deconvolution. The kernel is estimated in the linear least-squares sense using two images. Their works has also suggested an application for defocus using large/small aperture images. However, their work does not show any results or analysis. In this paper, we demonstrate that our proposed techniques can obtain much accurate kernel compared with Lim and Silverstein’s approach, and produce almost artifact-free image by a proposed de-ringing approach in deconvolution.

3 Problem Formulation

We take a pair of images: a blurred image B with a slow shutter speed and low ISO, and a noisy image N with high shutter speed and high ISO. The noisy image is usually underexposed and has a very low SNR since camera noise is dependent on the image intensity level [Liu et al. 2006]. Moreover, the noise in the high ISO image is also larger than that in the low ISO image since the noise is amplified by camera gain. But the noisy image is sharp because we use a fast shutter speed that is above the safe shutter speed.

We pre-multiply the noisy image by a ratio $\frac{ISO_R \Delta t_B}{ISO_N \Delta t_N}$ to compensate for the exposure difference between the blurred and noisy images, where Δt is the exposure time. We perform the multiplication in irradiance space then go back to image space if the camera response curve [Debevec and Malik 1997] is known. Otherwise, a gamma

$(\gamma = 2.0)$ curve is used as an approximation.

3.1 Our approach

Our goal is to reconstruct a high quality image I using the input images B and N

$$B = I \otimes K, \quad (1)$$

where K is the blur kernel and \otimes is the convolution operator. For the noisy image N , we compute a denoised image N_D [Portilla et al. 2003] (See Section 7 for details). N_D loses some fine details in the denoising process, but preserves the large scale, sharp image structures. We represent the lost detail layer as a *residual image* ΔI :

$$I = N_D + \Delta I. \quad (2)$$

Our first important observation is that the denoised image N_D is a very good initial approximation to I for the purpose of kernel estimation from Equation (1). The residual image ΔI is relatively small with respect to N_D . The power spectrum of the image I mainly lies in the denoised image N_D . Moreover, the large scale, sharp image structures in N_D make important contributions for the kernel estimation. As will be shown in our experiments on synthetic and real images, accurate kernels can be obtained using B and N_D in non-blind convolution.

Once K is estimated, we can again use Equation (1) to non-blindly deconvolute I , which unfortunately will have significant artifacts, e.g., ringing effects. Instead of recovering I directly, we propose to first recover the residual image ΔI from the blurred image B . By combining Equations (1) and (2), the residual image can be reconstructed from a *residual deconvolution*:

$$\Delta B = \Delta I \otimes K, \quad (3)$$

where $\Delta B = B - N_D \otimes K$ is a *residual blurred image*.

Our second observation is that the ringing artifacts from residual deconvolution of ΔI (Equation (3)) are smaller than those from deconvolution of I (Equation (1)) because ΔB has a much smaller magnitude than B after being offset by $N_D \otimes K$.

The denoised image N_D also provides a crucial gain signal to control the deconvolution process so that we can suppress ringing artifacts, especially in smooth image regions. We propose a de-ringing approach using a *gain-controlled* deconvolution algorithm to further reduce ringing artifacts.

The above three steps - kernel estimation (Section 4), residual deconvolution (Section 5), and de-ringing (Section 6) - are iterated to refine the estimated blur kernel K and the deconvoluted image I .

4 Kernel Estimation

In this section, we show that a simple constrained least-squares optimization is able to produce a very good initial kernel.

Iterative kernel estimation. The goal of kernel estimation is to find the blur kernel K from $B = I \otimes K$ with the initialization $I = N_D$. In vector-matrix form, it is $\mathbf{b} = \mathbf{A}\mathbf{k}$, where \mathbf{b} and \mathbf{k} are the vector forms of B and K , and \mathbf{A} is the matrix form of I . Lim and Silverstein [2006] compute the kernel \mathbf{k} by solving $\mathbf{b} = \mathbf{A}\mathbf{k}$ in the linear least-squares. However, the estimated kernel by this simple approach may be poor, as shown in Figure 2 (f). To obtain a better kernel, we use Tikhonov regularization and hysteresis thresholding in scale space.

Regularization. To stabilize the solution, we use Tikhonov regularization method with a positive scalar λ by solving $\min_{\mathbf{k}} \|\mathbf{A}\mathbf{k} -$

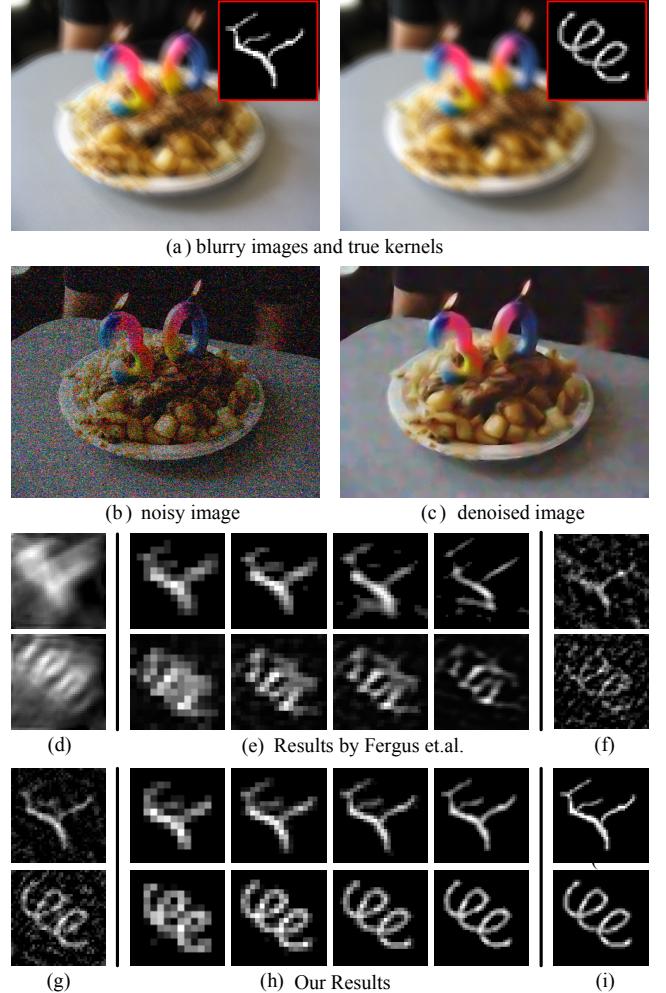


Figure 2: Kernel Estimation. Two blurred images are synthesized from a true image (also shown in Figure 4(e)). (d) Matlab’s deconvblind routine results. (e) Fergus’s result at finest 4 levels. (f) Lim and Silverstein’s result. (g) estimated kernels without hysteresis thresholding. (h) our result at the finest 4 levels. (i) true kernels.

$\mathbf{b}\|^2 + \lambda^2 \|\mathbf{k}\|^2$. The default value of λ is set at 5. The solution is given by $(\mathbf{A}^T \mathbf{A} + \lambda^2 \mathbf{I})\mathbf{k} = \mathbf{A}^T \mathbf{b}$ in closed-form if there are no other constraints on the kernel \mathbf{k} . But a real blur kernel has to be non-negative and preserve energy, so the optimal kernel is obtained from the following optimization system:

$$\min_{\mathbf{k}} \|\mathbf{A}\mathbf{k} - \mathbf{b}\|^2 + \lambda^2 \|\mathbf{k}\|^2, \text{ subject to } k_i \geq 0, \text{ and } \sum_i k_i = 1. \quad (4)$$

We adopt the Landweber method [Engl et al. 2000] to iteratively update as follows.

1. Initialize $\mathbf{k}^0 = \delta$, the delta function.
2. Update $\mathbf{k}^{n+1} = \mathbf{k}^n + \beta(\mathbf{A}^T \mathbf{b} - (\mathbf{A}^T \mathbf{A} + \lambda^2 \mathbf{I})\mathbf{k}^n)$.
3. Set $k_i^{n+1} = 0$ if $k_i^{n+1} < 0$, and normalize $k_i^{n+1} = k_i^{n+1} / \sum_i k_i^{n+1}$.

β is a scalar that controls the convergence. The iteration stops when the change between two steps is sufficiently small. We typically run about 20 to 30 iterations by setting $\beta = 1.0$. The algorithm is fast using FFT, taking about 8 to 12 seconds for a 64×64 kernel and a 800×600 image.

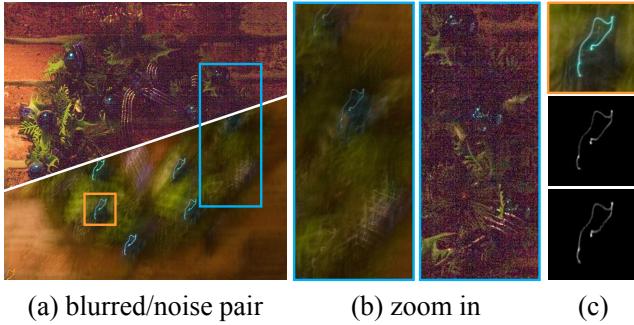


Figure 3: Blurred and noisy images from the light-blue box in (a) are zoomed-in in (b). The top image in (c) is a zoomed-in view of the light-orange box in (a), revealing the true kernel. The middle image in (c) is the estimated kernel using only image patches in (b). The bottom image in (c) is the estimated kernel using the whole image.

Hysteresis thresholding in scale space. The above iterative algorithm can be implemented in scale space to make the solution to overcome the local minimal. A straightforward method is to use the kernel estimated at the current level to initialize the next finer level. However, we have found that such initialization is insufficient to control noise in the kernel estimation. The noise or errors at coarse levels may be propagated and amplified to fine levels. To suppress noise in the estimate of the kernel, we prefer the global shape of the kernel at a fine level to be similar to the shape at its coarser level. To achieve this, we propose a **hysteresis thresholding** [Canny 1986] in scale space.



At each level, a kernel mask M is defined by thresholding the kernel values, $M_i = 1$ if $k_i > tk_{max}$, where t is a threshold and k_{max} is the maximum of all kernel values. We compute two masks M_{low} and M_{high} by setting two thresholds t_{low} and t_{high} . M_{low} is larger and contains M_{high} . After kernel estimation, we set all elements of K^l outside the mask M_{high} to zero to reduce the noise at level l . Then, at the next finer level $l+1$, we set all elements of K^{l+1} outside the up-sampled mask of M_{low} to zero to further reduce noise. This hysteresis thresholding is performed from coarse to fine. The pyramids are constructed using a downsampling factor of $1/\sqrt{2}$ until the kernel size at the coarsest level reaches 9×9 . We typically choose $t_{low} = 0.03$, and $t_{high} = 0.05$.

Results and discussion. We first compare our estimated kernel with the true kernel using a synthetic example. Figures 2(a-c) show two blurred images, a noisy image, and a denoised image. The blurred images are synthesized with two 41×41 known kernels. Figure 2(d) shows kernels estimated by Matlab’s deconvblind routine (a blind deconvolution) using the denoised image N_D as initialization. Figure 2(e) shows coarse-to-fine kernels (the finest 4 levels) estimated by Fergus’s algorithm only using the blurred image [Fergus et al. 2006]. The Matlab code is released by Fergus (<http://people.csail.mit.edu/fergus/>). We exhaustively tune all options in Fergus’s algorithm and select different regions in the image to produce the best results. Fergus’s algorithm recovers much better kernels than those using Matlab’s blind deconvolution. Figure 2(f) is result from [Lim and Silverstein 2006]. In comparison, our estimated kernels in Figure 2(h) are very close to the true kernels in Figure 2(i) because we solve a non-blind kernel estimation problem. The fine details and thin structures of the kernels are recovered. Figure 2(g) also shows our kernel estimation without hysteresis thresholding, which is very noisy.

Figure 3 shows our result on real images. Light-blue trajectories caused by highlights in the scene clearly reveal the accurate shape

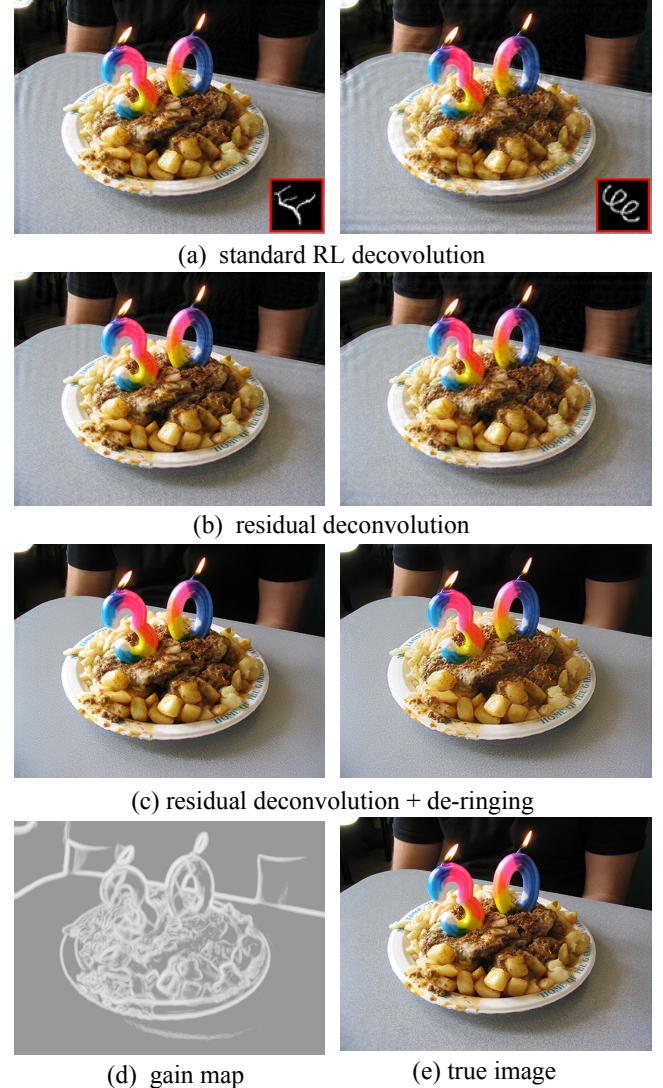


Figure 4: Deconvolution using true kernels. All results are generated after 20 iterations. Note that standard RL results contain unpleasant “ringing” artifacts - dark and light ripples around strong image features.

of the kernel. One such trajectory is shown in Figure 3(c). We also compare two kernels using selected image patches and the whole image. The recovered kernels have very similar shape to the light-blue trajectory, as shown in Figure 3(c). Kernel estimation is insensitive to the selected regions. The kernel size is very large, with 92×92 pixels.

5 Residual Deconvolution

Given the blur kernel K , the true image can be reconstructed from $B = K \otimes I$. Figure 4(a) shows the deconvolution results using a standard Richardson-Lucy (RL) algorithm after 20 iterations with the true kernels. The resulting images contain visible “ringing” artifacts, with dark and light ripples around bright features in the image. The ringing artifacts often occur with iterative methods, such as the RL algorithm. More iterations introduce not only more image details but also more ringing. Fergus et al. [2006] also observed this issue from their results.

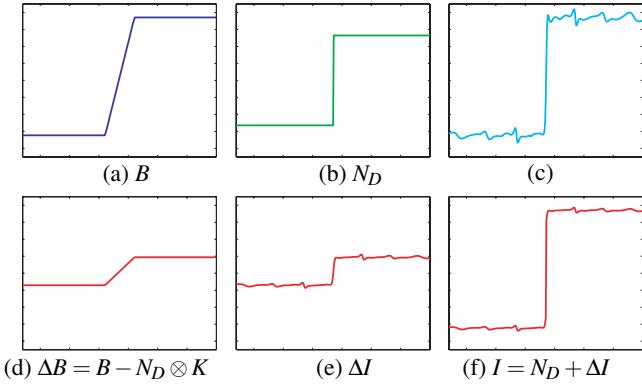


Figure 5: Residual deconvolution. (a-b) are the blurred signal and denoised signal. The blur kernel is a box filter. (c) is the standard deconvolution result from (a). (d-e) are the blurred residual signal and its deconvolution result. (f) is the residual deconvolution result. Notice that ringing artifact in (f) is smaller than that in (c).

The ringing effects are due to the well-known Gibbs phenomena in Fourier analysis at discontinuous points. The discontinuities could be at image edge points, boundaries or are artificially introduced by the inadequate spatial sampling of the images or the kernels. The larger the blur kernel, the stronger the ringing artifacts are.

The Gibbs oscillations have an amplitude independent of the cut-off frequencies of the filter, but are always proportional to the signal jump at the discontinuous points. The key to our approach is that we perform the deconvolution on relative image quantities to reduce the absolute amplitude of the signals. Instead of doing the deconvolution directly on the image B , we perform deconvolution on the residual blurred image $\Delta B = \Delta I \otimes K$ to recover the residual image ΔI . The final reconstructed image is $I = N_D + \Delta I$.

The standard RL algorithm is one of ratio-based iterative approaches. It enforces the non-negativity of pixel values. When using RL algorithms, the residual images should be offset by adding the constant 1, $\Delta I \mapsto \Delta I + 1$ and $\Delta B \mapsto \Delta B + 1$, as all images are normalized to range $[0,1]$. After each iteration, the residual image is offset back by subtracting the constant 1:

$$\Delta I_{n+1} = (K * \frac{\Delta B + 1}{(\Delta I_n + 1) \otimes K}) \cdot (\Delta I_n + 1) - 1, \quad (5)$$

where '*' is the correlation operator. Figure 4(b) shows the deconvolution results using the residual RL algorithm with the same number of iterations. Compared with the standard RL results (Figure 4(a)), the ringing effects are reduced.

Figure 5 shows a 1D example of the residual deconvolution. The ringing artifacts from ΔI are significantly weaker than those in I because the magnitude of ΔB (after subtracting $N_D \otimes K$ from B) is much smaller than that of B .

6 De-ringing with Gain-controlled RL

The residual deconvolution lessened the ringing effects, but cannot fully eliminate them, as shown in Figure 4(b). Another example is shown in Figure 7(b). We observe that the ringing effects are most distracting in smooth regions because human perception can tolerate small scale ringing in highly textured regions. We have also found that the mid-scale ringing effects are more noticeable compared with the fine details and large scale sharp structures in the image. Note that the strong ringing is mainly caused by high contrast edges and the magnitude of ringings is proportional to the

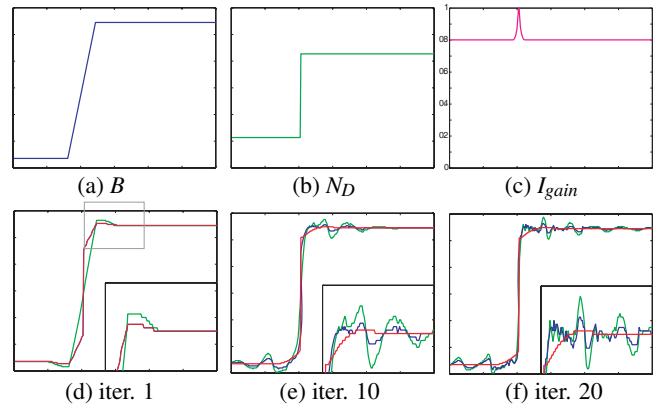


Figure 6: Gain-controlled RL. (a-c) blurred signal, denoised signal, and gain map. The kernel is estimated using B and N_D . (d-f) deconvolution results by standard RL (green), residual RL (blue), and gain-controlled RL (red), after iteration 1, 10, and 20. The plot at the bottom-right are blown-up views. Notice that the ringing effects are amplified and propagated in standard RL and residual RL, but suppressed in gain-controlled RL.

magnitude of image gradient. Based on these observations, we propose a de-ringing approach with a gain-controlled RL algorithm as follows.

Gain-controlled Richardson-Lucy (RL). We modify the residual RL algorithm by introducing a gain map I_{Gain} :

$$\Delta I_{n+1} = I_{Gain} \cdot \left\{ (K * \frac{\Delta B + 1}{(\Delta I_n + 1) \otimes K}) \cdot (\Delta I_n + 1) - 1 \right\}, \quad (6)$$

where I_{Gain} is a multiplier (≤ 1) to suppress the contrast of the recovered residual image ΔI . Since RL is a ratio-based algorithm, the ringing effects are amplified at each iteration by the ratio $K * \frac{\Delta B + 1}{(\Delta I_n + 1) \otimes K}$ in (6). Multiplying a factor less than one at each iteration will suppress the propagation of the ringing effects. Notice that multiplying a factor will not decrease the overall magnitude of the signal but decrease the contrast of the signal because the ratio $K * \frac{\Delta B + 1}{(\Delta I_n + 1) \otimes K}$ will increase the magnitude of the signal in each iteration. At the last iteration, we do not multiply the gain map I_{Gain} . We denote the image reconstructed by gain-controlled RL as I_g .

Since we want to suppress the contrast of ringing in the smooth regions while avoiding suppression of sharp edges, the gain map should be small in smooth regions and large in others. Hence, we define the gain map using the gradient of the denoised image as:

$$I_{Gain} = (1 - \alpha) + \alpha \cdot \sum_l ||\nabla N_D^l||, \quad (7)$$

where α controls the influence of the gain map, and ∇N_D^l is the gradient of the denoised image at the l th level of the Gaussian pyramid with standard deviation 0.5. The parameter α controls the degree of suppression. In all the results shown in this paper, we set the value of α to 0.2. Aggregated image gradients at multiple scales have also been used in HDR compression [Fattal et al. 2002; Li et al. 2005]. Here, the gradients of denoised image provide a gain signal to adaptively suppress the ringing effects in different regions.

Figure 6 shows a 1D example of gain-controlled RL. As we can see, the residual RL can reduce the magnitude of ringing compared with the standard RL. In both standard RL and residual RL, the magnitude of ringing increases and the spatial range of ringing spreads gradually, after each iteration. With the control from the gain map, the ringing effects are suppressed at each iteration

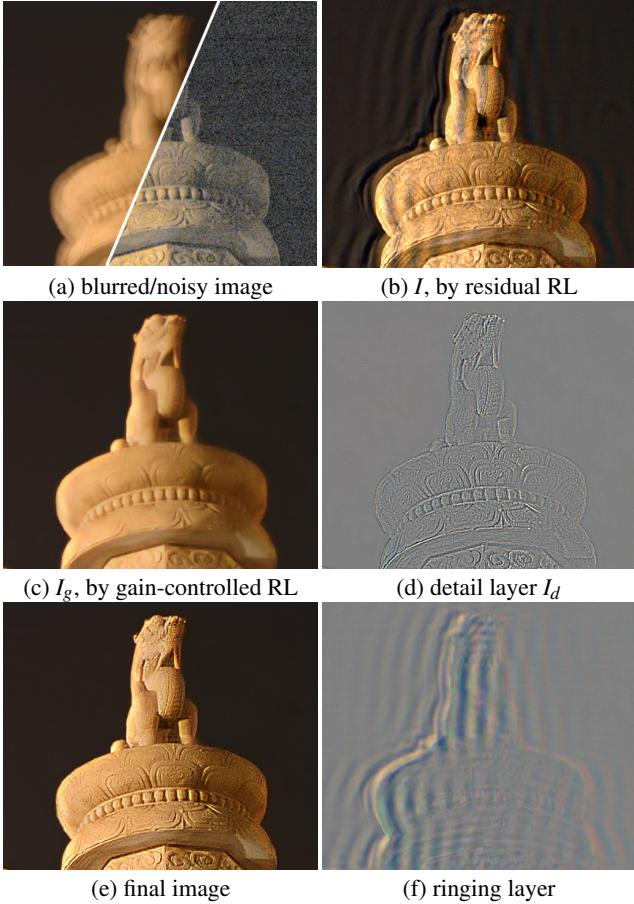


Figure 7: De-ringing. The gain-controlled RL effectively suppresses the ringing artifacts and produces de-ringing image I_g in (c). The detail layer I_d in (d) is extracted from the residual RL result in (b) with the guidance of the I_g using a joint/cross bilateral filter. Our final image in (e) is obtained by adding (c) and (d) together.

(e.g., $I_{Gain} = 0.8$ in flat region). Most importantly, the propagation of ringing is greatly prevented so that the ringing is significantly reduced.

Figure 7(c) shows a gain-controlled RL result I_g . It is a clean deconvolution result with large scale sharp edges, compared with the residual RL result I in Figure 7(c). However, some fine details are inevitably suppressed by gain-controlled RL. Fortunately, we are able to add fine scale image details for the residual RL result I using the following approach.

Adding details. We extract the fine scale detail layer $I_d = I - \bar{I}$ from the residual RL result I , where $\bar{I}(\mathbf{x}) = F(I(\mathbf{x}))$ is a filtered image and $F(\cdot)$ is a low-pass filter. In other words, the details layer is obtained by a high-pass filtering. We use joint/cross bilateral filtering [Petschnigg et al. 2004; Eisemann and Durand 2004] as it preserves large scale edges in I_g :

$$F(I(\mathbf{x}); I_g) = \frac{1}{Z_{\mathbf{x}}} \sum_{\mathbf{x}' \in W(\mathbf{x})} G_d(\mathbf{x} - \mathbf{x}') G_r(I(\mathbf{x}) - I_g(\mathbf{x}')) \cdot I_{\mathbf{x}'},$$

where σ_d and σ_r are spatial and signal deviations of Gaussian kernels G_d and G_r . $W(\mathbf{x})$ is a neighboring window and $Z_{\mathbf{x}}$ is a normalization term. The default values of σ_d and σ_r are 1.6 and 0.08. Figure 7(d) shows the extracted detail layer.

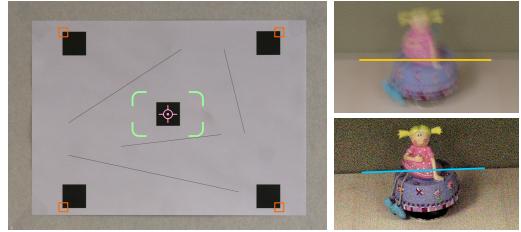


Figure 8: Top left: image pattern. Four corners in red boxes are extracted in two shots as corresponding point pairs. Top right: in-plane rotation correction using two manually specified lines. Bottom: The experiment was repeated by four users (A,B,C,D). In each cell (a 4x4 grid), one color dot represents a difference vector between one of corresponding point pairs in two shots. The grid unit is 0.5 pixel and cell center is the coordinate origin.

Composing the gain-controlled RL result I_g and the detail layer I_d produces our final image, as shown in Figure 7(e). The ringing layer (Figure 7(f)) can also be obtained by subtracting I_g from the filtered image \bar{I} . As we expected, the ringing layer mainly contains the ripple-like ringing effects. In the final result, the ringing artifacts are significantly reduced while the recovered image details from deconvolution are well preserved. Figures 4 (c-d) show another example of results after de-ringing and the computed gain map.

To summarize, our iterative image deblurring algorithm consists of the following steps: estimate the kernel K , compute the residual deconvolution image I , compute the gain-controlled deconvolution image I_g , and construct the final image by adding the detail layer I_d . The iterations stop when the change is sufficiently small.

7 Implementation Details

Image acquisition In practice, we require one image be taken soon after another, to minimize misalignment between two images. We have two options to capture such image pairs very quickly. First, two successive shots with different camera settings are triggered by a laptop computer connected to the camera. This frees the user from changing camera settings between two shots. Second, we use *exposure bracketing* built in many DSLR cameras. In this mode, two successive shots can be taken with different shutter speeds by pressing the shutter only once. Using these two options, the time interval between two shots can be very small, typically only 1/5 second which is a small fraction of typical shutter speed (> 1 second) of the blurred image. The motion between two such shots is mainly a small translation if we assume that the blurred image can be modeled by a single blur kernel, i.e., the dominant motion is translation. Because the translation only results in an offset of the kernel, it is unnecessary to align two images.

We can also manually change the camera settings between two shots. In this case, we have found that the dominant motions between two shots are translation and in-plane rotation. To correct in-plane rotation, we simply draw two corresponding lines in the blurred/noisy images. In the blurred image, the line can be specified along a straight object boundary or by connecting two corner features. The noisy image is rotated around its image center such

that two lines are virtually parallel. If an advanced exposure bracketing allowing more controls is built to future cameras, this manual alignment will become unnecessary.

To quantitatively measure relative motion between two shots, we have performed a usability study. We asked four users to continuously take two shots of a pattern on the wall (as shown in the top right of Figure 8), using laptop control and manual control, with a compact camera and a DSLR camera. Two shots have no blur and are taken with the same camera settings. Then, four corresponding points nearby the image corners in two shots are extracted. We correct the transformation (only translation for laptop control, but in-plane rotation after translation for manual control) between two shots. The bottom row of Figure 8 shows registration errors after the correction. In each cell, a dot represents a difference vector between a pair of corresponding points. The overall pixel error is less than 2 pixels at the full image resolution. Not surprisingly, the best aligned image is obtained using laptop control and a DSLR camera.

Image denoising For the noisy image N , we apply a wavelet-based denoising algorithm [Portilla et al. 2003] with Matlab code from <http://decsai.ugr.es/~javier/denoise/>. The algorithm is one of the state-of-art techniques and comparable to several commercial denoising softwares. We have also experimented with bilateral filtering but found that it is hard to achieve a good balance between removing noise and preserving details, even with careful parameter tuning.

8 Experimental Results

We apply our approach to a variety of blurred/noisy image pairs in low lighting environments using a compact camera (Canon S60, 5M pixels) and a DSLR camera (Canon 20D, 8M pixels).

Comparison. We compare our approach with denoising [Portilla et al. 2003], and a standard RL algorithm. Figure 9, from left to right, shows a blurred image, noisy image (enhanced), denoised image, standard RL result (using our estimated kernel), and our result. The kernel sizes are 31×31 , 33×33 , and 40×40 for the three examples.

We manually tune the noise parameter (standard deviation) in the denoising algorithm to achieve a best visual balance between noise removal and detail preservation. Compared with denoised results shown in Figure 9(c), our results in Figure 9(e) contain much more fine details, such as tiny textures on the fabric in the first example, thin grid structures on the crown in the second example, and clear text on the camera in the last example. Because the noise image is scaled up from a very dark, low contrast image, partial color information is also lost. Our approach recovers correct colors through image deblurring. Figure 9(d) shows standard RL deconvolution results which exhibit unpleasant ringing artifacts.

Large noise. Figure 10 shows a blurred/noisy pair containing thin hairs and a sweater with detailed structures. The images are captured by the compact camera and the noisy image has very strong noises. Most fabric textures on the sweater are faithfully recovered in our result.

The last column in the second row of Figure 10 shows the estimated initial kernel and the refined kernel by the iterative optimization. The iteration number is typically 2 or 3 in our experiments. The refined kernel has a sharper and sparser shape than the initial one.

Large kernel. Figure 11 shows an example with a large blur by the compact camera. The kernel size is 87×87 at the original resolution 1200×1600 . The image shown here is cropped to 975×1146 . Compared with the state-of-art single image kernel estimation approach [Fergus et al. 2006] in which the largest kernel is 30 pixels,

	blurred image	noisy image
art (Fig. 8)	1.0s, ISO 100	1/200s, ISO 1600
crown (Fig. 8)	1.0s, ISO 100	1/90s, ISO 1600
camera (Fig. 8)	0.8s, ISO 100	1/320s, ISO 1600
sweater (Fig. 9)	1.3s, ISO 100	1/80s, ISO 400
dragon (Fig. 10)	1.3s, ISO 100	1/80s, ISO 400
budda (Fig. 11)	1.0s, ISO 100	1/200s, ISO 1600

Table 1: Shutter speeds and ISO settings in Figure 9, 10, 11, and 12.

our approach using an image pair significantly extends the degree of blur that can be handled.

Small noise and kernel. In a moderately dim lighting environment, we may capture input images with small noise and blur, as shown in Figure 12. This is a typical case assumed in Jia’s approach [2004] which is a color transfer based algorithm. The third and fourth columns in Figure 12 are color transferred result [Jia et al. 2004] and histogram equalization result from the blurred image to the denoised image. Note that the colors cannot be accurately transferred (e.g., Buddha’s golden hat) because both approaches use global mappings. Our result not only recovers more details (e.g., horizontal lines on background) but also has similar colors to the blurred image for all details.

Table 1 shows the shutter speeds and ISO settings of examples in Figure 9-12. We are able to reduce exposure time (shutter speed \times ISO) by about 10 stops.

9 Discussion and Conclusion

We have proposed an image deblurring approach using a pair of blurred/noisy images. Our approach takes advantage of both images to produce a high quality reconstructed image. By formulating the image deblurring problem using two images, we have developed an iterative deconvolution algorithm which can estimate a very good initial kernel and significantly reduce deconvolution artifacts. No special hardware is required. Our proposed approach uses off-the-shelf, hand-held cameras.

Limitations remain in our approach, however. Our approach shares the common limitation of most image deblurring techniques: assuming a single, spatial-invariant blur kernel. For spatial-variant kernel, it is possible to locally estimate kernels for different parts of the image and blend deconvolution results. Most significantly, our approach requires two images. We envision that the ability to capture such pairs will eventually move into the camera firmware, thereby making two-shots capture easier and faster.

In the future, we plan to extend our approach to other image deblurring applications, such as deblurring video sequences, or out-of-focus deblurring. Our techniques can also be applied in a hybrid image system [Ben-Ezra and Nayar 2003] or combined with coded exposure photography [Raskar et al. 2006].

Acknowledgements

We thank the anonymous reviewers for helping us to improve this paper, Stephen Lin for his help in video production and proofreading. This work is performed when Lu Yuan visited Microsoft Research Asia. Lu Yuan and Long Quan were supported in part by Hong Kong RGC projects 619005 and 619006.

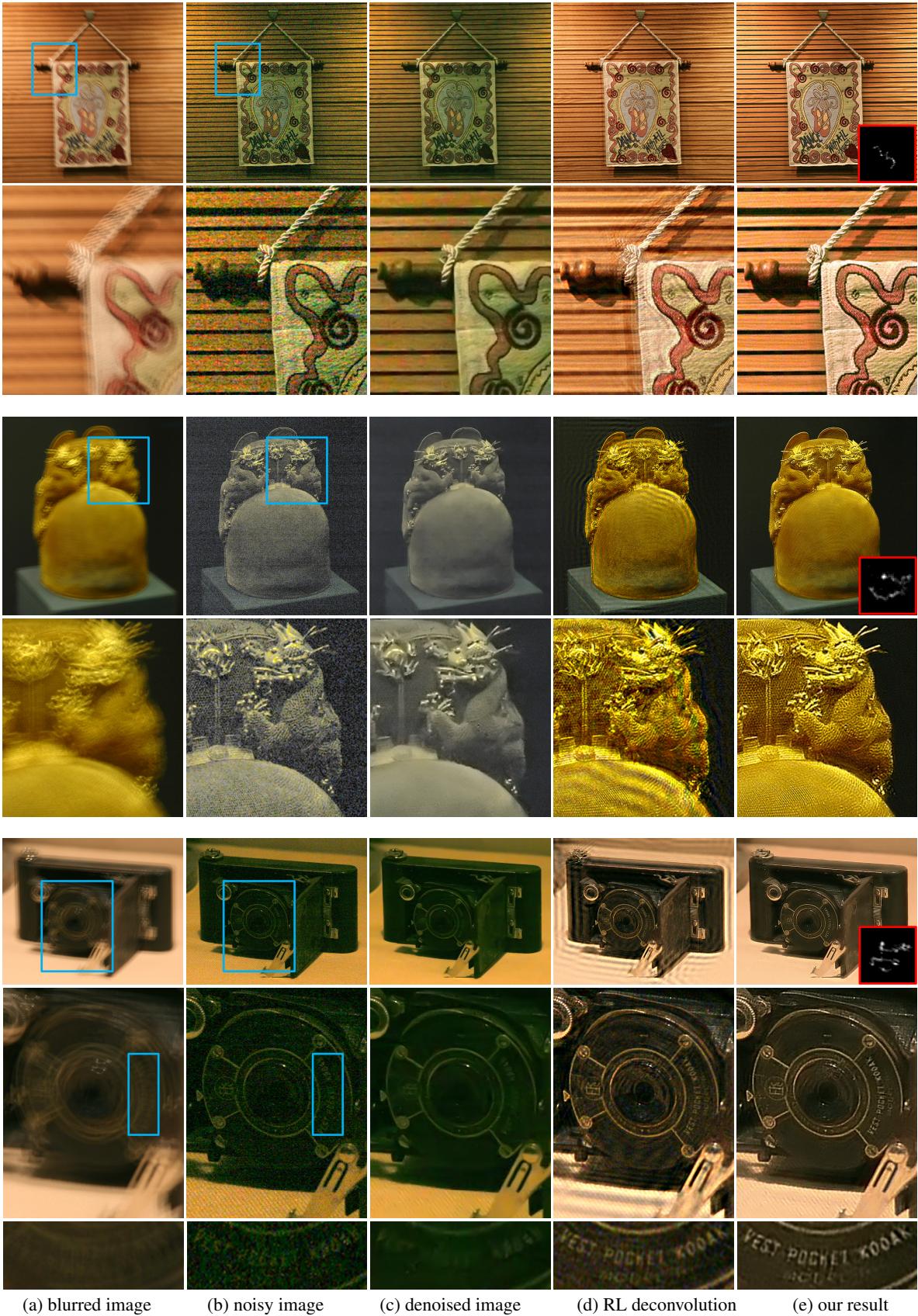


Figure 9: Comparison. The noisy image is enhanced for display. The estimated blur kernel is shown at the bottom-right corner in the last column. The second example is taken by the compact camera and the other two by the DSLR camera. Note that our result contains finer details than the denoised image and less ringing artifacts than the RL deconvolution result. In the last example, "VEST POCKET KODAK" on the camera can be seen from our result but it is hard, if not impossible, to be recognized from the blurred image or the noisy image. We encourage the reader to see a close-up view in the electronic version.

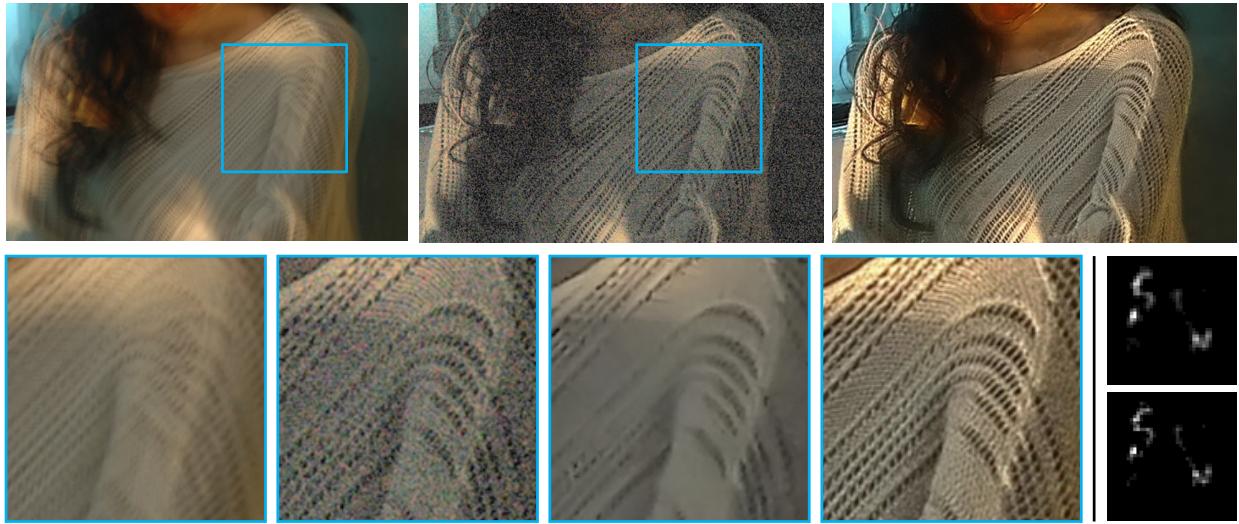


Figure 10: Large noise. Top three images: blurred, noisy, and our result. Bottom left four images: zoomed-in views of blurred, noisy, denoised and our result. Bottom right two images are initial kernel (top) and refined kernel (bottom) using our iterative algorithm. The kernel size is 32×32 .



Figure 11: Large kernel. Left: blurred image, noisy image, denoised image, and our result. Top right: two image patches in the light-orange boxes in blurred/noisy images reveal the kernel shape. Note that the highlight point in the noisy patch is an ellipse-like shape. Bottom right: estimated 87×87 kernel.



Figure 12: Small noise and kernel. This examples is taken by the DSLR camera. The kernel size is 21×21 . From left to right: blurred image, noisy image, color transferred denoised image, histogram-equalization denoised image, and our result. Our deblurred result has more details and vivid colors.

References

- BARDSELEY, J., JEFFERIES, S., NAGY, J., AND PLEMMONS, R. 2006. Blind iterative restoration of images with spatially-varying blur. In *Optics Express*, 1767–1782.
- BASCLE, B., BLAKE, A., AND ZISSERMAN, A. 1996. Motion deblurring and super-resolution from an image sequence. In *Proceedings of ECCV*, vol. II, 573–582.
- BEN-EZRA, M., AND NAYAR, S. K. 2003. Motion deblurring using hybrid imaging. In *Proceedings of CVPR*, vol. I, 657–664.
- BENNETT, E. P., AND McMILLAN, L. 2005. Video enhancement using per-pixel virtual exposures. *ACM Trans. Graph.* 24, 3, 845–852.
- BUADES, A., COLL, B., AND MOREL, J. M. 2005. A non-local algorithm for image denoising. In *Proceedings of CVPR*, vol. II, 60–65.
- CANNY, J. 1986. A computational approach to edge detection. *IEEE Trans. on PAMI*. 8, 6, 679–698.
- CARON, J. N., M., N. N., AND J., R. C. 2002. Noniterative blind data restoration by use of an extracted filter function. *Applied optics (Appl. opt.)* 41, 32, 68–84.
- DEBEVEC, P. E., AND MALIK, J. 1997. Recovering high dynamic range radiance maps from photographs. In *Proceedings of SIGGRAPH*, 369–378.
- DURAND, F., AND DORSEY, J. 2002. Fast bilateral filtering for the display of high-dynamic-range images. In *Proceedings of SIGGRAPH*, 257–266.
- EISEMANN, E., AND DURAND, F. 2004. Flash photography enhancement via intrinsic relighting. *ACM Trans. Graph.* 23, 3, 673–678.
- ENGL, H. W., HANKE, M., AND NEUBAUER, A. 2000. *Regularization of Inverse Problems*. Kluwer Academic.
- FATTAL, R., LISCHINSKI, D., AND WERMAN, M. 2002. Gradient domain high dynamic range compression. In *Proceedings of SIGGRAPH*, 249–256.
- FERGUS, R., SINGH, B., HERTZMANN, A., ROWEIS, S. T., AND FREEMAN, W. T. 2006. Removing camera shake from a single photograph. In *ACM Trans. Graph.*, vol. 25, 787–794.
- GEMAN, D., AND REYNOLDS, G. 1992. Constrained restoration and the recovery of discontinuities. *IEEE Trans. on PAMI*. 14, 3, 367–383.
- H. RICHARDSON, W. 1972. Bayesian-based iterative method of image restoration. *JOSA, A* 62, 1, 55–59.
- JALOBEANU, A., BLANC-FERAUD, L., AND ZERUBIA, J. 2002. Estimation of blur and noise parameters in remote sensing. In *Proceedings of ICASSP*, 249–256.
- JIA, J., SUN, J., TANG, C.-K., , AND SHUM, H.-Y. 2004. Bayesian correction of image intensity with spatial consideration. In *Proceedings of ECCV*, 342–354.
- KUNDUR, D., AND HATZINAKOS, D. 1996. Blind image deconvolution. *IEEE Signal Processing Magazine*. 13, 3, 43–64.
- LEVIN, A. 2006. Blind motion deblurring using image statistics. In *Advances in Neural Information Processing Systems (NIPS)*.
- LI, Y., SHARAN, L., AND ADELSON, E. H. 2005. Compressing and companding high dynamic range images with subband architectures. *ACM Trans. Graph.* 24, 3, 836–844.
- LIM, S. H., AND SILVERSTEIN, D. A. 2006. Method for deblurring an image. *US Patent Application, Pub. No. US2006/0187308 A1*, Aug 24, 2006.
- LIU, X., AND GAMAL, A. 2001. Simultaneous image formation and motion blur restoration via multiple capture. *Proceedings of ICASSP..*
- LIU, C., FREEMAN, W., SZELISKI, R., AND KANG, S. 2006. Noise estimation from a single image. In *Proceedings of CVPR*, vol. I, 901–908.
- NEELAMANI, R., CHOI, H., AND BARANIUK, R. 2004. ForWaRd: Fourier-wavelet regularized deconvolution for ill-conditioned systems. *IEEE Trans. on Signal Processing* 52, 2, 418–433.
- NIKON. 2005. http://www.nikon.co.jp/main/eng/portfolio/about/technology/nikon_technology/vr_e/index.htm.
- PERONA, P., AND MALIK, J. 1990. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. on PAMI* 12, 7, 629–639.
- PETSCHNIGG, G., AGRAWALA, M., HOPPE, H., SZELISKI, R., COHEN, M., AND TOYAMA, K. 2004. Digital photography with flash and no-flash image pairs. *ACM Trans. Graph.* 23, 3, 664–672.
- PORTILLA, J., STRELA, V., WAINWRIGHT, M., AND SIMONCELLI, E. P. 2003. Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Trans. on Image Processing* 12, 11, 1338–1351.
- RASKAR, R., AGRAWAL, A., AND TUMBLIN, J. 2006. Coded exposure photography: motion deblurring using fluttered shutter. *ACM Trans. Graph.* 25, 3, 795–804.
- RAV-ACHA, A., AND PELEG, S. 2000. Restoration of multiple images with motion blur in different directions. *IEEE Workshop on Applications of Computer Vision*.
- RAV-ACHA, A., AND PELEG, S. 2005. Two motion-blurred images are better than one. *Pattern Recogn. Lett.* 26, 3, 311–317.
- REEVES, S. J., AND MERSEREAU, R. M. 1992. Blur identification by the method of generalized cross-validation. *IEEE Trans. on Image Processing*. 1, 3, 301–311.
- ROTH, S., AND BLACK, M. J. 2005. Fields of experts: A framework for learning image priors. In *Proceedings of CVPR*, vol. II, 860–867.
- RUDIN, L., OSHER, S., AND FATEMI, E. 1992. Nonlinear total variation based noise removal algorithms. *Phys. D*. 60, 259–268.
- SIMONCELLI, E. P., AND ADELSON, E. H. 1996. Noise removal via bayesian wavelet coring. In *Proceedings of ICIP*, vol. I, 379–382.
- TOMASI, C., AND MANDUCHI, R. 1998. Bilateral filtering for gray and color images. In *Proceedings of ICCV*, 839–846.
- Y. YITZHAKY, I. MOR, A. L., AND KOPEIKA, N. 1998. Direct method for restoration of motion blurred images. *J. Opt. Soc. Am., A* 15, 6, 1512–1519.
- ZAROWIN, C. B. 1994. Robust, noniterative, and computationally efficient modification of vab cittert deconvolution optical figuring. *JOSA, A* 11, 10, 2571–2583.