

# Using Optical Defocus to Denoise

Qi Shan\*   Jiaya Jia†   Sing Bing Kang‡   Zenglu Qin†

\*University of Washington   †The Chinese University of Hong Kong   ‡Microsoft Research

## Abstract

*Effective reduction of noise is generally difficult because of the possible tight coupling of noise with high-frequency image structure. The problem is worse under low-light conditions. In this paper, we propose slightly optically defocusing the image in order to loosen this noise-image structure coupling. This allows us to more effectively reduce noise and subsequently restore the small defocus. We analytically show how this is possible, and demonstrate our technique on a number of examples that include low-light images.*

## 1. Introduction

Despite the advances in camera technology, sensor noise remains a major problem in computational photography, especially when pictures are taken in low-light conditions or with short exposure. The characterization of noise is non-trivial (see, for example, [28, 21]), as noise is a function of exposure level, photon flux, and the electron-photon conversion process. In addition, noise contains high-frequency components that are quantitatively and visually indistinguishable from the inherent fine structures in natural images. Denoising these images using current techniques either excessively smooths out image structure or retains noise with texture details.

In this paper, we present a novel denoising technique that uses optical defocus to decouple noise from a single input image. An illustration is shown in Figure 1. This is based on the observation that many image details are hard to separate from noise; we simplify the denoising algorithm by reducing these image components optically. The ISO setting of the camera can then be set high in our system to allow better shutter speed.

This paper has three main contributions. First, our imaging system uses optical defocus (illustrated in Figure 1) to hide image details, which simplifies noise reduction. Extensive experiments indicate that blurry images conceal many structural details but do not necessarily lose all of them. The majority of the image structures can be recovered in a blur removal process. Moreover, optical defocus and noise production are two separate processes in image formation,

which allows us to manipulate the former in order to simplify the reduction of the latter.

Second, we describe a new and effective method for single image noise estimation that is based on the presence of optical defocus. We analyze the relationship between defocus and noise, and propose a novel metric for defining the noise likelihood. We use effective *variable splitting* optimization (based on local closed-form solutions in iterations) to remove noise from the observed image. One result is shown in Figure 2; here, the underexposed image was captured in a dark room using a Nikon DSLR camera slightly defocused. If we merely enhance brightness, noise is amplified as well.

Third, we theoretically analyze the performance of the proposed algorithm in choosing a key parameter and quantitatively study the information gain with our new imaging technique. Noise and blur can be regarded as two types of artifacts that degrade image quality. But we show that the information loss introduced by defocus is several orders of magnitude smaller than the gain by removing noise. Compared to conventional single-image denoising methods, the total image structure destruction in the new denoise-plus-deconvolution framework can generally be smaller.

**Assumptions.** Focusing on foreground objects is a widely used photo-taking strategy. We assume that the foreground objects are in focus after removing the defocus blur. Our goal is not to deblur the entire scene (especially if it has significant depth variation). Our method estimates the defocus blur PSF through a calibration process by measuring the foreground depth using cameras.

## 2. Related Work

There has been a significant amount of work to remove image noise in both computer vision and image processing fields. Techniques primarily fall into multi-image and single-image approaches.

**Two-image/video denoising.** The temporal information in video has been extensively used for denoising. For example, Chen and Tang [4] used a Spatio-Temporal Markov Random Field (MRF) approach to denoise video sequences

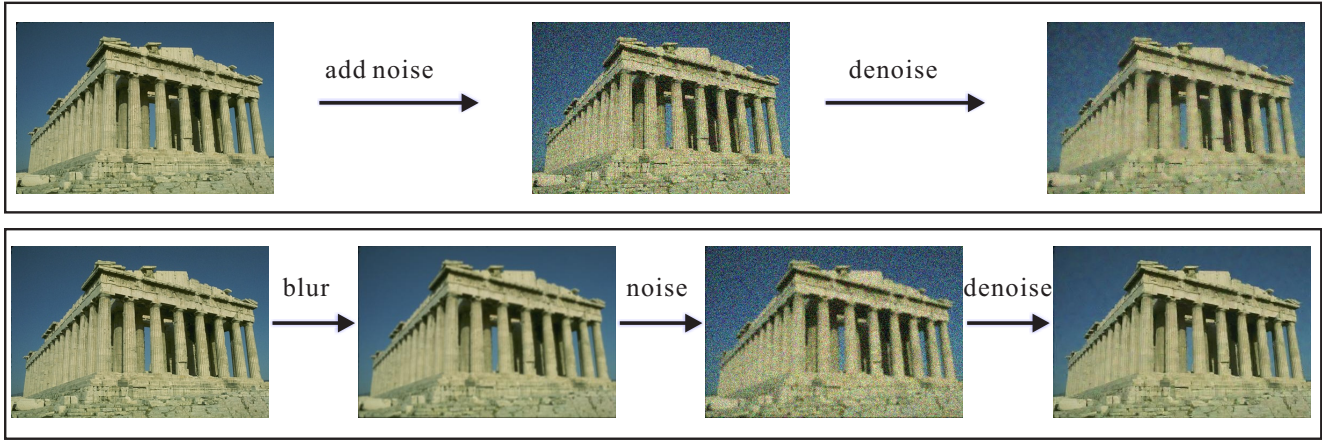


Figure 1. Traditional noise generation and removal model (upper row) versus our new setup (lower row).



Figure 2. Effective denoising through slight optical defocus. (a) Input low-light image, (b) enhanced input (4X brightness), (c) restored image after noise and blur reduction.

with motion estimation between frames. Without considering the displacement information, Bennett and McMillan [2] enhanced videos by locally switching between spatial or temporal filtering.

There are other techniques for enhancing pictures by processing two images taken with different camera settings, for example, with flash/no-flash (e.g., [26, 9]) or long/short exposure (e.g., [16, 37]). In [26], the noise in the non-flashed image is removed by bilateral filtering, where the detail is obtained from the flashed image. Eisemann and Durand [9] extracted a coarse structure layer from the non-flash image and enhanced it by adding the detail and color information from the flashed image. Jia *et al.* [16] captured blur/underexposed image pairs and then transferred color from the blur image to the underexposed counterpart. Yuan *et al.* [37] used a blur/noisy image pair. The result is computed by deconvolution with an estimated PSF.

Methods that use multiple input images typically require pixels to be accurately aligned over time. This is difficult to achieve in the presence of significant noise in a dynamic

scene.

**Single image denoising.** The most direct solution is to use a filter. Popular methods involve bilateral filtering [33] and anisotropic diffusion, either implemented in the form of partial differential equations (PDEs) [35, 34, 12] or derived from optimization using variational methods [30, 31]. Anisotropic diffusion methods control the diffusion direction and magnitude such that small smoothness is imposed on points near edges and in the direction of edge crossing.

Effective single image denoising is very challenging because the problem is under-constrained and requires to make additional assumptions about the image or noise. For example, Roth and Black [30] modified the simple smoothness prior and introduced a high-order learning-based image prior model, which is potentially capable of modeling natural scenes. Liu *et al.* [21] estimated the noise level function and constructed a Gaussian conditional random field to infer the clean image with the piecewise smoothness assumption. [29] presents a method using total variation regularization.

An interactive approach is proposed to remove noise in [5].

Wavelet-based methods make use of the observation that multiscale subbands satisfy a highly kurtotic marginal distribution (e.g., [11]). The method of Portilla *et al.* [27] models the wavelet coefficients at adjacent positions and scales as the product of two independent random variables and uses the Gaussian scale mixture (GSM) model for denoising. Another common assumption is the existence of regular texture or repeated local appearance. In [34], various “geometry tensors” were proposed for noise removal and texture preservation. Non-local spatial domain denoising methods [3, 1, 7] rely on repeated local appearance to restore the latent image.

All these single image methods work best for denoising images with little or no fine texture. Unfortunately, it is very difficult to separate camera sensor noise from subtle image structures. In this paper, we partially resolve this ambiguity by taking into account both noise and defocus blur.

**Single image deblurring.** Another type of image artifacts caused by low lighting is motion blur. Single image deblurring methods [10, 15, 32] are capable of restoring images to a certain extent. These methods involve kernel estimation and deconvolution. Non-blind deconvolution methods [22, 20, 38] assume that noise is relatively small, so that general smoothness constraints are adequate. With the exception of [18], single image deblurring techniques usually do not work well if noise is significant.

### 3. Noise Analysis

An image with noise can be expressed as

$$\mathbf{B}' = \mathbf{x} + \mathbf{n}, \quad (1)$$

where  $\mathbf{B}'$  is the observed noisy image,  $\mathbf{x}$  is the latent image, and  $\mathbf{n}$  is noise.  $\mathbf{n}$  is typically assumed to be signal independent when caused by dark current, amplifier noise, and the quantizer in the camera circuitry [14, 21]. However, recent camera noise estimation work found that this simple formula does not sufficiently model the mechanism of an image sensor where photon flux and the uncertainty of the electron-photon conversion process produce signal (or luminance) dependent noise. The total noise variance  $\sigma^2$  is dependent on the gray-level variance  $\kappa_{gray}^2$ . It can be written as  $\sigma^2 = \kappa_{gray}^2 + C\eta^2$  [28], where  $\eta^2$  is the photon noise variance and  $C$  is a weight. The existence of fine and high-frequency structures in  $\mathbf{x}$  (typical in real images) compounds the difficulty in accurately estimating  $\mathbf{n}$  from  $\mathbf{B}'$ .

Our solution is to slightly defocus the image. During image capture, optical defocus is relatively independent of pixel noise generation [13, 36]. Thus, the image formation process can be expressed as

$$\mathbf{B} = (\mathbf{x} \otimes \mathbf{f}) + \mathbf{n}, \quad (2)$$

where  $\mathbf{B}$  and  $\mathbf{x}$  are the observed and latent images.  $\mathbf{f}$  and  $\mathbf{n}$  are respectively the defocus PSF and noise.  $(\mathbf{x} \otimes \mathbf{f})$  makes the image structure less correlated with noise.

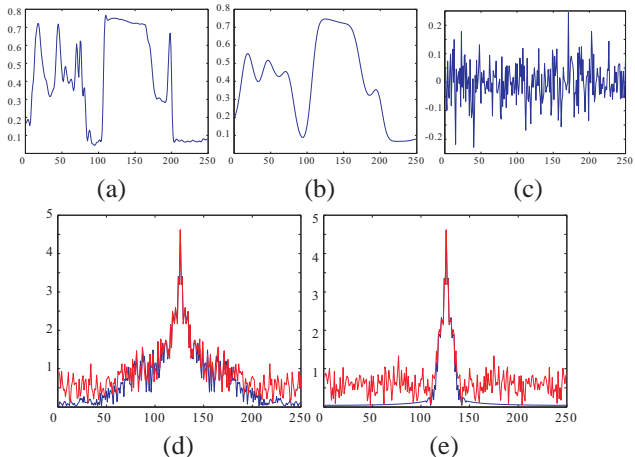


Figure 3. Reducing signal-noise coupling. (a) Input clean signal, (b) Gaussian filtered version of (a), (c) noise, (d) log-magnitudes of the unfiltered signals with (red curve) and without (blue curve) noise in the frequency domain. Both of them contain high frequency components. (e) Log-magnitudes of the filtered signals with (red curve) and without (blue curve) noise in the frequency domain. Their distributions are different.

Note that equations similar to (2) were proposed in non-blind deconvolution [22, 32] as the image degradation model. However, these methods cannot handle large noise. We have experimented with these methods to directly deblur our captured images with large regularization weights to overcome noise, and found either excessive removal of texture details or amplification of noise.

Figure 3 shows (in 1D) how blur loosens the signal-noise coupling. The red and blue curves in Figure 3(d) are plots of the log-magnitudes of the clean and noise-corrupted signals respectively in the frequency domain. They share many high frequency components that are not easily separable. However, if the signal is pre-filtered, as shown in (b), adding noise to it significantly changes its frequency distribution (shown in Figure 3(e)). The distribution difference implies a way to remove substantial noise from signal.

If noise can primarily be removed, we then digitally remove the slight optical blur through deconvolution. Recent deblurring work (e.g., [38, 19]) and our analysis (Section 5) show that deconvolving an image can recovered many details even though they are barely noticeable.

### 4. Noise Estimation with Focal Blur

In this section, we describe our novel method to estimate noise from a single defocused image (after which we can get the latent image by subtraction). Conventional noise estimation methods typically do at least one of the following:

use multiple images as input, assume spatial smoothness, or assume replicable patterns. In contrast, our method avoids these by exploring the benefit of image defocus in noise estimation.

#### 4.1. Noise Estimation with a Convolution Model

Following Eq. (2), a simple method to remove defocus blur while assuming small noise contamination would be to solve

$$\bar{\mathbf{x}} = F^{-1}\bar{\mathbf{B}} = \frac{F^T}{F^T F}\bar{\mathbf{B}}, \quad (3)$$

where  $F$  is the matrix form of the PSF  $\mathbf{f}$ .  $\bar{\cdot}$  defines the vectorization operator which stacks all values in a rasters canning order. Here  $\bar{\mathbf{x}}$  and  $\bar{\mathbf{B}}$  are respectively the vectorized  $\mathbf{x}$  and  $\mathbf{B}$ . For the rest of this paper,  $F^{-1}$  and  $\frac{1}{F}$  are used interchangeably to denote the inverse of  $F$ .

When used as is, Eq. (3) is very sensitive to noise because of magnification by the denominator  $F^T F$ . To deal with this problem, a common practice is to increase the diagonal values of  $F^T F$  to stabilize the matrix inverse. Thus Eq. (3) is modified to

$$\bar{\mathbf{x}}(\lambda) = \frac{F^T}{F^T F + \lambda I}\bar{\mathbf{B}}, \quad (4)$$

where  $I$  is the identity matrix with the same dimension of  $\bar{\mathbf{B}}$ , and parameter  $\lambda$  controls the regularization strength.

With this modification, we now analyze how the estimated  $\bar{\mathbf{x}}(\lambda)$  using Eq. (4) deviates from the ground truth latent image  $\bar{\mathbf{x}}^*$ . This theoretical analysis will provide important insights on how image noise influences the defocused image formation, which in turn leads to a novel formula for accurate noise estimation. We write

$$\begin{aligned} \bar{\mathbf{x}}(\lambda) - \bar{\mathbf{x}}^* &= \frac{F^T}{F^T F + \lambda I}\bar{\mathbf{B}} - \frac{F^T}{F^T F}(\bar{\mathbf{B}} - \bar{\mathbf{n}}) \\ &= \frac{F^T \bar{\mathbf{n}}}{F^T F + \lambda I} + \frac{-\lambda I \bar{\mathbf{x}}^*}{F^T F + \lambda I}. \end{aligned} \quad (5)$$

$\bar{\mathbf{x}}^*$  can be expressed as  $F^{-1}(\bar{\mathbf{B}} - \bar{\mathbf{n}})$  based on Eq. (2). Here the commutativity of multiplication in the denominator is not a concern because both  $F^T F + \lambda I$  and  $F^T F$  are symmetric. Eq. (5) contains two terms. The first denotes the influence of image noise, while the second represents the effect of structure smoothing. Note that the ground truth  $\bar{\mathbf{x}}^*$  is unknown. So introducing Eq. (5) is only to establish a metric to properly *measure* the influence of noise.

Our proposed metric is the partial derivative of  $\bar{\mathbf{x}}$  with respect to  $\lambda$ :

$$\begin{aligned} \frac{\partial \bar{\mathbf{x}}}{\partial \lambda} &= \lim_{\delta \lambda \rightarrow 0} \frac{1}{\delta \lambda} (\bar{\mathbf{x}}(\lambda + \delta \lambda) - \bar{\mathbf{x}}^*) - (\bar{\mathbf{x}}(\lambda) - \bar{\mathbf{x}}^*) \\ &= \lim_{\delta \lambda \rightarrow 0} \frac{1}{\delta \lambda} \left( \left( \frac{F^T \bar{\mathbf{n}}}{F^T F + \lambda + \delta \lambda} + \frac{-(\lambda + \delta \lambda) \bar{\mathbf{x}}^*}{F^T F + \lambda + \delta \lambda} \right) \right. \\ &\quad \left. - \left( \frac{F^T \bar{\mathbf{n}}}{F^T F + \lambda} + \frac{-\lambda \bar{\mathbf{x}}^*}{F^T F + \lambda} \right) \right) \\ &= -h(F, \lambda) \bar{\mathbf{n}} - h(F, \lambda) F \bar{\mathbf{x}}^*, \end{aligned} \quad (6)$$

where

$$h(F, \lambda) = \frac{F^T}{(F^T F + \lambda)^T (F^T F + \lambda)}. \quad (7)$$

We further compute the squared  $L_2$  norm of  $\frac{\partial \bar{\mathbf{x}}}{\partial \lambda}$  as

$$\left\| \frac{\partial \bar{\mathbf{x}}}{\partial \lambda} \right\|_2^2 = \|h(F, \lambda) \bar{\mathbf{n}} + h(F, \lambda) F \bar{\mathbf{x}}^*\|_2^2. \quad (8)$$

We now explain that  $\left\| \frac{\partial \bar{\mathbf{x}}}{\partial \lambda} \right\|_2^2$  is a new noise likelihood because it monotonically increases with noise variance.

**New Noise Metric Analysis** First off, it can be derived (see Appendix) that

$$\left\| \frac{\partial \bar{\mathbf{x}}}{\partial \lambda} \right\|_2^2 = \|\mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{n}) + \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\mathbf{x}^*)\|_2^2,$$

where

$$\mathcal{H}(\mathbf{f}, \lambda) = \frac{\overline{\mathcal{F}(\mathbf{f})}}{(\overline{\mathcal{F}(\mathbf{f})} \circ \mathcal{F}(\mathbf{f}) + \lambda) \circ (\overline{\mathcal{F}(\mathbf{f})} \circ \mathcal{F}(\mathbf{f}) + \lambda)}. \quad (9)$$

$\mathcal{F}$  denotes the Fourier transform, and  $\overline{\cdot}$  is the conjugate. Let

$$\begin{aligned} \xi_1 &= \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{n}), \\ \xi_2 &= \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\mathbf{x}^*), \end{aligned}$$

and denote  $\xi_{1,i}$  (or  $\xi_{2,i}$ ) an element of  $\xi_1$  (or  $\xi_2$ ) with the index  $i$ . It yields

$$\begin{aligned} \left\| \frac{\partial \bar{\mathbf{x}}}{\partial \lambda} \right\|_2^2 &= \|\xi_1 + \xi_2\|_2^2 = \sum_i |\xi_{1,i} + \xi_{2,i}|^2 \\ &= \sum_i \left( \overline{(\xi_{1,i} + \xi_{2,i})} (\xi_{1,i} + \xi_{2,i}) \right) \\ &= \sum_i \overline{\xi_{1,i}} \xi_{1,i} + \sum_i \overline{\xi_{2,i}} \xi_{2,i} + \sum_i \overline{\xi_{1,i}} \xi_{2,i} + \sum_i \xi_{1,i} \overline{\xi_{2,i}} \\ &= \|\xi_{1,i}\|_2^2 + \|\xi_{2,i}\|_2^2 + \sum_i \overline{\xi_{1,i}} \xi_{2,i} + \sum_i \xi_{1,i} \overline{\xi_{2,i}}, \end{aligned} \quad (10)$$

The first term  $\|\xi_1\|_2^2$  on the right hand side can be expressed as

$$\begin{aligned} \|\xi_1\|_2^2 &= \|\mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{n})\|_2^2 \\ &\approx \sigma^2 \|\mathcal{H}(\mathbf{f}, \lambda)\|_2^2, \end{aligned} \quad (11)$$

assuming zero mean noise (see [28, 24]).  $\|\xi_1\|_2^2$  and  $\sigma^2 \|\mathcal{H}(\mathbf{f}, \lambda)\|_2^2$  in our experiments have very similar values. Note that  $\|\mathcal{F}(\mathbf{n})\|_2^2 = \|\bar{\mathbf{n}}\|_2^2$ , representing the variance of image noise, denoted by  $\sigma^2$ .

The other terms  $\|\xi_2\|_2^2$ ,  $\sum_i \overline{\xi_{1,i}} \xi_{2,i}$  and  $\sum_i \xi_{1,i} \overline{\xi_{2,i}}$  can also be simplified as follows. First,  $\xi_{1,i} \overline{\xi_{2,i}}$  is one element of

$\overline{\xi_1} \circ \xi_2$  with index  $i$ . We then study the representation of  $\overline{\xi_1} \circ \xi_2$  for transforming  $\overline{\xi_{1,i}} \xi_{2,i}$ . Note that

$$\begin{aligned} \overline{\xi_1} \circ \xi_2 &= \overline{\mathcal{H}(\mathbf{f}, \lambda)} \circ \mathcal{F}(\mathbf{n}) \circ \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\mathbf{x}^*), \\ &= \overline{\mathcal{H}(\mathbf{f}, \lambda)} \circ \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{f}) \circ \overline{\mathcal{F}(\mathbf{n})} \circ \mathcal{F}(\mathbf{x}^*), \\ &= \overline{\mathcal{H}(\mathbf{f}, \lambda)} \circ \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\mathbf{x}^* \oplus \mathbf{n}), \end{aligned}$$

where  $\oplus$  is a correlation operator. For simplicity's sake, in the last term  $\mathcal{F}(\mathbf{x}^* \oplus \mathbf{n})$ , we denote  $\alpha = \mathbf{x}^* \oplus \mathbf{n}$ . Each element  $\alpha(x, y) \in \alpha$  with coordinate  $(x, y)$  is expressed as

$$\alpha(x, y) = \sum_{(x', y')} \mathbf{x}^*(x', y') \mathbf{n}(x' + x, y' + y),$$

where  $(x', y')$  enumerates all elements in  $\mathbf{n}$  for correlation. Then we group  $\mathbf{n}$  in  $\alpha(x, y)$  based on the gray level of  $\mathbf{x}^*$  as

$$\alpha_{x,y} = \sum_j \mathbf{x}_j^* \sum_{k \in \Omega_{\mathbf{x}_j^*}} \mathbf{n}_k,$$

where  $\mathbf{x}_j^*$  denotes pixels  $\mathbf{x}^*(x', y')$  with the gray level  $j$ .  $\Omega_{\mathbf{x}_j^*}$  denotes the set of pixels in  $\mathbf{n}$  that correlate with  $\mathbf{x}_j^*$ . Given the sufficient number of pixels for each gray level and zero mean noise [28, 24], we can reasonably assume  $\sum_{k \in \Omega_{\mathbf{x}_j^*}} \mathbf{n}_k \approx 0$ , which further yields  $\alpha_{x,y} \approx 0$ . We thus obtain  $\sum_i \overline{\xi_{1,i}} \xi_{2,i} \approx 0$  and  $\sum_i \xi_{1,i} \overline{\xi_{2,i}} \approx 0$  in theory.

We also validated this with extensive experiments and found that the value of  $\|\xi_1\|_2^2$ , expressed in Eq. (11), is about 4 orders of magnitude larger than  $\sum_i \xi_{1,i} \xi_{2,i}$  in natural images. This justifies the practice of ignoring  $\sum_i \overline{\xi_{1,i}} \xi_{2,i}$  and  $\sum_i \xi_{1,i} \overline{\xi_{2,i}}$ . Finally, since  $\|\xi_2\|_2^2$  is independent of noise  $\mathbf{n}$ , we denote

$$\|\xi_2\|_2^2 = C,$$

where  $C$  is a constant for a given noisy/blur image. Thus, Eq. (10) transforms to

$$\left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2 \approx \sigma^2 \|\mathcal{H}(\mathbf{f}, \lambda)\|_2^2 + C. \quad (12)$$

We do not know the values of  $C$  and  $\sigma^2$  (which is also unnecessary in our method); but Eq. (12) has an important property:  $\left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2$  increases monotonically with the noise variance. This makes it a good metric for evaluating how noisy a blurred image is. We empirically validate this property by generating 10 blurred images. They are added with different degrees of noise. Three of the images are shown in Figure 4(a)-(c) with noise standard deviations  $\sigma = 14\%$ ,  $18\%$ , and  $20\%$  respectively. We then compute  $\left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2$  for each image and plot their values in Figure 4(d). As expected,  $\left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2$  monotonically increases with  $\sigma^2$ .

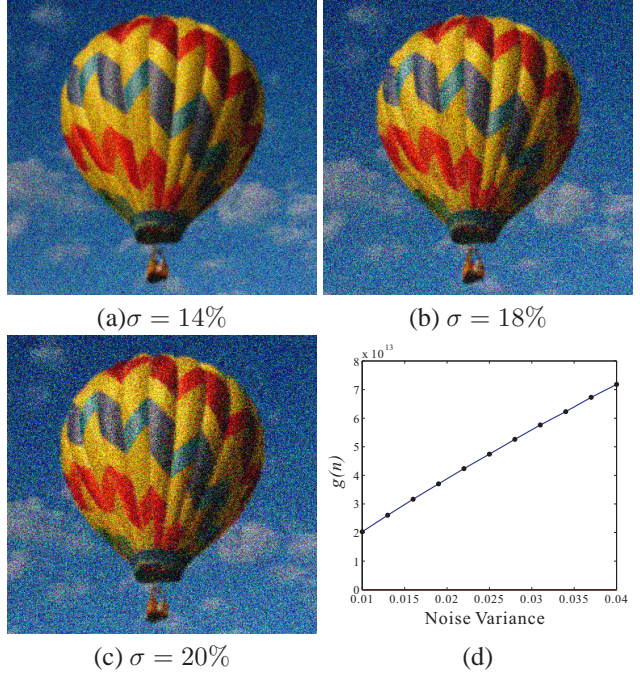


Figure 4. Change of  $\left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2$  with respect to different noise variance  $\sigma^2$ . (a)-(c) Three blurred images out of ten that contain different amount of noise. (d)  $\left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2$  versus  $\sigma^2$ .

Thus, the function  $\left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2$  is a proper measure of how well noise is removed and of how the remaining image satisfies the convolution model defined with the defocus blur PSF. Let  $\mathbf{n}^*$  and  $\mathbf{n}'$  denote the ground truth noise and the noise (somehow) estimated from  $\mathbf{B}$  respectively. The remaining noise  $\mathbf{n}$  (with its vectorized form denoted by  $\bar{\mathbf{n}}$ ) in the image is thus

$$\bar{\mathbf{n}} = \bar{\mathbf{n}}^* - \bar{\mathbf{n}}' = \bar{\mathbf{B}} - F\bar{\mathbf{x}}^* - \bar{\mathbf{n}}'. \quad (13)$$

Substituting Eq. (13) into Eq. (8) and denoting  $E_0(\mathbf{n}') = \left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2$  yield

$$E_0(\mathbf{n}') = \left\| h(F, \lambda)(\bar{\mathbf{B}} - \bar{\mathbf{n}}') \right\|_2^2. \quad (14)$$

$E_0(\mathbf{n}')$  indicates how good the noise estimate  $\mathbf{n}'$  is by evaluating the strength of the remaining noise  $\mathbf{n}^* - \mathbf{n}'$  in the image  $\mathbf{B}$ .  $\mathbf{x}^*$  cancels out in Eq. (14) and thus does not need to be known beforehand. With the monotone property of  $\left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2$  with respect to the level of noise, we use  $E_0(\mathbf{n}')$  as the likelihood in defining a new objective function.

**Energy Function** Likelihood (14) is combined with a regularization term  $\|\mathbf{n}'\|$  to avoid the trivial solution  $\mathbf{n}' = \mathbf{B}$ . The total energy  $E_1(\mathbf{n}')$  is written as

$$\begin{aligned} E_1(\mathbf{n}') &= E_0(\mathbf{n}') + w\|\mathbf{n}'\| \\ &= \left\| h(F, \lambda)(\bar{\mathbf{B}} - \bar{\mathbf{n}}') \right\|_2^2 + w\|\mathbf{n}'\|, \end{aligned} \quad (15)$$

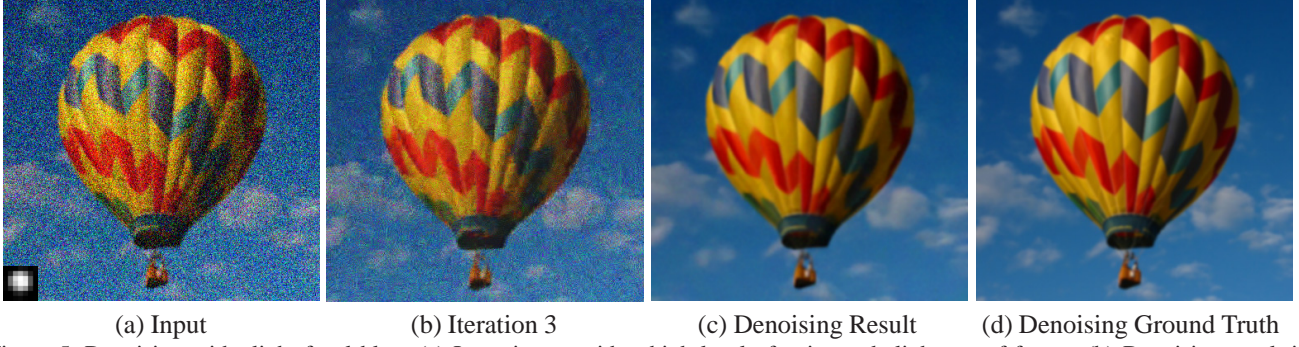


Figure 5. Denoising with slight focal blur. (a) Input image with a high level of noise and slight out-of-focus. (b) Denoising result in iteration 3. (c) The denoising result after convergence in 8 iterations. (d) Ground truth blurred image without noise.

where weight  $w$  is set to 1.0 in our experiments. To optimize  $E_1(\mathbf{n}')$ , we use the *variable splitting* optimization technique. The idea is to split the variable  $\mathbf{n}'$  into a pair of variables ( $\mathbf{n}'$  and  $\mathbf{n}^s$ ), used respectively in the two terms in Eq. (15), such that minimizing the sum of the two terms under the constraint that  $\mathbf{n}' = \mathbf{n}^s$  is essentially equivalent to solving the original problem.

**Optimization** Eq. (15) is thus reformulated as

$$E(\mathbf{n}', \mathbf{n}^s) = \|h(F, \lambda)(\bar{\mathbf{B}} - \bar{\mathbf{n}}^s)\|_2^2 + w\|\mathbf{n}'\| + \kappa\|\mathbf{n}' - \mathbf{n}^s\|_2^2, \quad (16)$$

where the weight  $\kappa$  determines how similar  $\mathbf{n}'$  and  $\mathbf{n}^s$  are. An iterative approach is used to update  $\mathbf{n}^s$  and  $\mathbf{n}'$  separately.

#### [Updating $\mathbf{n}'$ ]

By removing all terms independent of  $\mathbf{n}'$ , we get

$$E'(\mathbf{n}') = w\|\mathbf{n}'\| + \kappa\|\bar{\mathbf{n}}' - \bar{\mathbf{n}}^s\|_2^2. \quad (17)$$

Its closed-form solution is

$$\mathbf{n}'_i = \begin{cases} \max\left(\frac{2\kappa\bar{\mathbf{n}}^s_i - w}{2\kappa}, 0\right); & \bar{\mathbf{n}}^s_i \geq 0 \\ \min\left(\frac{2\kappa\bar{\mathbf{n}}^s_i + w}{2\kappa}, 0\right); & \bar{\mathbf{n}}^s_i < 0 \end{cases} \quad (18)$$

where  $i$  indexes the pixels.

#### [Updating $\mathbf{n}^s$ ]

Similarly, removing all terms independent of  $\mathbf{n}^s$  yields

$$E^s(\mathbf{n}^s) = \|h(F, \lambda)(\bar{\mathbf{B}} - \bar{\mathbf{n}}^s)\|_2^2 + \kappa\|\bar{\mathbf{n}}' - \bar{\mathbf{n}}^s\|_2^2.$$

It can be rewritten in the frequency domain as

$$E^s(\mathbf{n}^s) = \|\mathcal{H}(\mathbf{f}, \lambda) \circ (\mathcal{F}(\mathbf{B}) - \mathcal{F}(\mathbf{n}^s))\|_2^2 + \kappa\|\mathcal{F}(\bar{\mathbf{n}}') - \mathcal{F}(\mathbf{n}^s)\|_2^2. \quad (19)$$

A closed-form solution also exists by computing the partial derivative with respect to  $\mathbf{n}^s$  and setting it to zero:

$$\mathbf{n}^s = \mathcal{F}^{-1} \left( \frac{\kappa\mathcal{F}(\bar{\mathbf{n}}') + \overline{\mathcal{H}(\mathbf{f}, \lambda)} \circ \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{B})}{\kappa + \overline{\mathcal{H}(\mathbf{f}, \lambda)} \circ \mathcal{H}(\mathbf{f}, \lambda)} \right). \quad (20)$$

Updating  $\mathbf{n}'$  and  $\mathbf{n}^s$  iterates until convergence (no more than 15 iterations generally). Because each step has a closed-form solution, the computation is much more efficient compared to conventional gradient descent. The energy is guaranteed to monotonically decrease.

Figure 5 shows one example. The input image is shown in (a). It is blurred with a defocus PSF (see bottom left). Significant noise ( $\sigma = 20\%$ ) is also added. The intermediate denoising result (in iteration 3) and the final result are shown in Figures 5(b) and (c) respectively. Most noise is removed. The PSNR is as high as 33.83 (or MSE 5.19) compared to the ground truth blurred image shown in Figure 5(d). The denoised image will be further deconvolved to remove the slight blur (details in Sections 5 and 6). The final result (shown in Figure 6) contains many details. It is notable that the original PSNR of the input image is only 14 before denoising.

## 4.2. Determining $\lambda$

The value of  $\lambda$  in the above formulas has a significant impact in denoising. In this section, we show that  $\lambda$  needs to satisfy a small-value requirement.

To effectively separate noise from the input image  $\mathbf{B}$ , the total energy produced using the ground truth noise  $\mathbf{n}^*$  should be smaller than that with an incorrect noise estimate  $\mathbf{n}'$ , such that the optimization has a good chance to converge correctly (i.e., making towards the small-energy state). It therefore requires

$$\|h(F, \lambda)(\bar{\mathbf{B}} - \bar{\mathbf{n}}')\|_2^2 + w\|\mathbf{n}'\| > \|h(F, \lambda)(\bar{\mathbf{B}} - \bar{\mathbf{n}}^*)\|_2^2 + w\|\mathbf{n}^*\|.$$

In the frequency domain, an equivalent inequality is expressed as

$$\|\mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{B} - \mathbf{n}')\|_2^2 + w\|\mathbf{n}'\| > \|\mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{B} - \mathbf{n}^*)\|_2^2 + w\|\mathbf{n}^*\|. \quad (21)$$

Denoting

$$\begin{aligned}\Psi_1 &= \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{B} - \mathbf{n}') \quad \text{and} \\ \Psi_2 &= \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{B} - \mathbf{n}^*),\end{aligned}$$

Eq. (21) can be written as

$$\|\Psi_1\|_2^2 - \|\Psi_2\|_2^2 > w\|\mathbf{n}^*\| - w\|\mathbf{n}'\|. \quad (22)$$

We further define

$$\|\Psi\|_1 = \|\Psi_1\|_2^2 - \|\Psi_2\|_2^2, \quad (23)$$

where  $\|\Psi\|_1 = \sum_i \psi_i$  and  $\psi_i$  is the element in matrix  $\Psi$  with index  $i$ . Through a few algebraic operations, we can express

$$\psi_i = \mathcal{H}(\mathbf{f}, \lambda) \circ \overline{\mathcal{H}(\mathbf{f}, \lambda)} \circ \Lambda, \quad (24)$$

where

$$\Lambda = \overline{\mathcal{F}(\Delta\mathbf{n})} \circ \mathcal{F}(\mathbf{B}^*) + \overline{\mathcal{F}(\mathbf{B}^*)} \circ \mathcal{F}(\Delta\mathbf{n}) + \overline{\mathcal{F}(\Delta\mathbf{n})} \circ \mathcal{F}(\Delta\mathbf{n}).$$

$\mathbf{B}^*$  denotes the ground truth noise-free image, i.e.,  $\mathbf{B}^* = \mathbf{B} - \mathbf{n}^*$ .  $\Delta\mathbf{n}$  is the difference between  $\mathbf{n}^*$  and  $\mathbf{n}$ , i.e.,  $\Delta\mathbf{n} = \mathbf{n}^* - \mathbf{n}'$ . Assuming that  $\Delta\mathbf{n}$  and  $\mathbf{B}^*$  are independent, we derive

$$\begin{aligned}\overline{\mathcal{F}(\Delta\mathbf{n})} \circ \mathcal{F}(\mathbf{B}^*) &= \mathcal{F}(\mathbf{B}^* \oplus \mathbf{n}) \approx \mathbf{0}, \\ \mathcal{F}(\Delta\mathbf{n}) \circ \overline{\mathcal{F}(\mathbf{B}^*)} &= \mathcal{F}(\mathbf{n} \oplus \mathbf{B}^*) \approx \mathbf{0}.\end{aligned} \quad (25)$$

$\Lambda$  can therefore be simplified to

$$\Lambda = \overline{\mathcal{F}(\Delta\mathbf{n})} \circ \mathcal{F}(\Delta\mathbf{n}) = \|\Delta\mathbf{n}\|_2^2. \quad (26)$$

Because

$$w\|\Delta\mathbf{n}\|_1 > w\|\mathbf{n}^*\|_1 - w\|\mathbf{n}'\|_1,$$

we establish a stronger inequality

$$\|\Psi\|_1 = \sum_i \psi_i > w\|\Delta\mathbf{n}\|_1. \quad (27)$$

If (27) holds, so does (22). Substituting Eq. (24) into (27) yields

$$\|\mathcal{H}(\mathbf{f}, \lambda)\|_2^2 > w\|\Delta\mathbf{n}\|_1 / \|\Delta\mathbf{n}\|_2^2.$$

This inequality shows that a large value of  $\lambda$  adversely affects convergence. The *maximum* value of  $\lambda$  depends on  $w$  and noise estimation error.

In our experiments, we assign  $\lambda$  a fixed small value  $10^{-4}$  to inhibit its negative influence. It is also allowed alternatively to approximate a reasonable upper bound of  $\lambda$  individually for each example in the following way.



(a) Denoised image (b) Final deconvolved image

Figure 6. Denoised and final deconvolved results. The PSNR of (b) is high (31.44). This indicates that the error introduced in deconvolution is small compared to intensive image noise artifacts. The deliberately produced blur as a matter of fact profits noise removal.

With the observation that residual noise computed in optimization is usually weaker than the actual image noise, we express

$$\frac{\|\Delta\mathbf{n}\|_1}{\|\Delta\mathbf{n}\|_2^2} < \frac{1}{\|\mathbf{n}^*\|_2}.$$

A coarse estimation of image noise strength  $\|\mathbf{n}^*\|_2$  can be accomplished using the approach of Healey and Kondepudy [13]. So an upper bound for  $\lambda$  is approximately  $w/\|\mathbf{n}^*\|_2$ .

## 5. Final Deconvolution and Error Analysis

After noise removal, we deconvolve the image. In this section, we show that this process introduces error that is insignificant compared to image noise.

We analyze a naïve algorithm characterized by Eq. (4), which simply inverts the blur process through blur matrix division. Its error estimate can generally be regarded as a loose *upper bound* of errors produced by various deconvolution methods because almost all deconvolution methods, such as [38, 20, 32, 19], use more advanced techniques to regularize deblurring, and are thus capable of producing much higher quality results. We show, compared to the scale of image noise, even the upper bound of the deconvolution error is sufficiently small.

The deconvolution error is the difference between the recovered and ground truth latent images. For Eq. (4), the error can be expressed as

$$\bar{\mathbf{x}}(\lambda) - \bar{\mathbf{x}}^* = \frac{F^T \bar{\mathbf{n}}}{F^T F + \lambda I} + \frac{-\lambda I \bar{\mathbf{x}}^*}{F^T F + \lambda I}, \quad (28)$$

where  $\mathbf{n}$  primarily comprises quantization errors after noise removal. Note that  $\lambda$  is used to stabilize deconvolution, and is usually with a very small value ( $10^{-4}$  in our experiments). We can therefore ignore the second term in Eq.

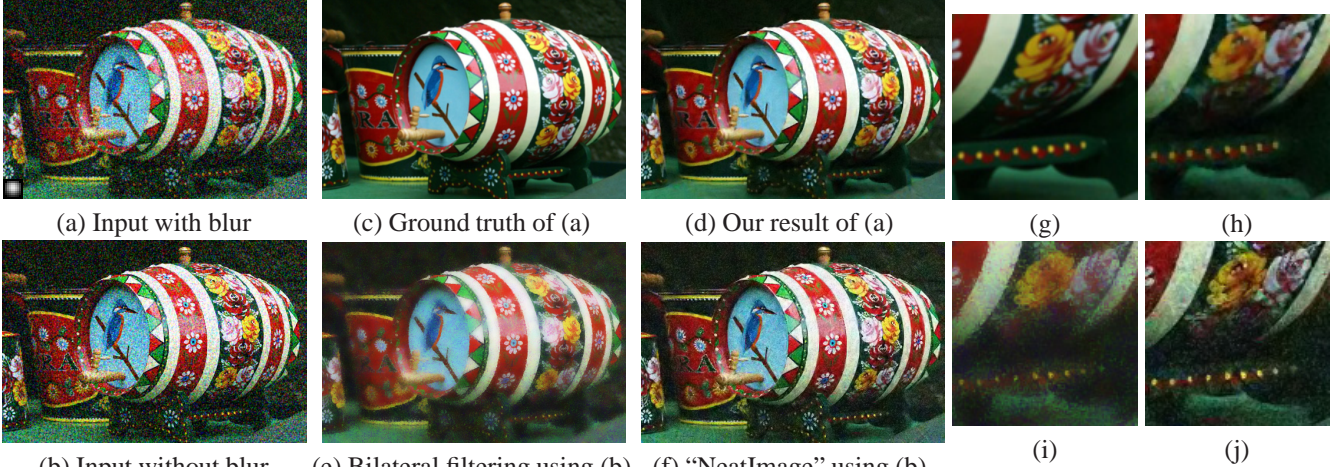


Figure 7. Quantitative evaluation. The input image (shown in (a)) is blurred with the PSF shown on bottom left. Significant CCD noise ( $\sigma = 20\%$ ) is also added. (b) Another input noisy image that is not focal blurred. (c) The ground truth sharp image without noise. (d) Our restoration result of (a), with PSNR 28.8. (e)-(f) Results of bilateral filtering [9] and “NeatImage” [25] with PSNRs 21.3 and 22.5, respectively. (g)-(j) Close-ups of (c)-(f).

(28), yielding

$$\|\bar{\mathbf{x}}(\lambda) - \bar{\mathbf{x}}^*\|_2^2 \approx \left\| \frac{F^T \bar{\mathbf{n}}}{F^T F + \lambda I} \right\|_2^2.$$

Expressing it in the frequency domain gives

$$\left\| \frac{F^T \bar{\mathbf{n}}}{F^T F + \lambda I} \right\|_2^2 = \left\| \frac{\overline{\mathcal{F}(\mathbf{f})} \circ \mathcal{F}(\mathbf{n})}{\overline{\mathcal{F}(\mathbf{f})} \mathcal{F}(\mathbf{f}) + \lambda} \right\|_2^2 = \left\| \frac{\overline{\mathcal{F}(\mathbf{f})}}{\overline{\mathcal{F}(\mathbf{f})} \mathcal{F}(\mathbf{f}) + \lambda} \right\|_2^2 \sigma^2,$$

where  $\sigma$  is the standard deviation of noise  $\mathbf{n}$ . This shows that  $\mathbf{n}$  is magnified by a factor

$$\mathcal{J} = \sqrt{\left\| \frac{\overline{\mathcal{F}(\mathbf{f})}}{\overline{\mathcal{F}(\mathbf{f})} \mathcal{F}(\mathbf{f}) + \lambda} \right\|_2^2}. \quad (29)$$

We now estimate the magnitude of  $\mathcal{J}$  using different out-of-focus PSFs. In our experiments, the value varies from 17.7 to 28.6. The quantization error  $\mathbf{n}$  is generally modeled as a uniform distribution in the range  $[-0.5, 0.5]$ , with a standard deviation of 0.29. With these quantities, it can be estimated that the reconstruction PSNR of the naïve deconvolution algorithm easily exceeds 33. The MSE ranges from 24 to 66. These quantities indicate that the error introduced using this simple deconvolution algorithm is very small if the convolution model is satisfied. In comparison, the input image has significant noise where the PSNR is 14 and the MSE is  $2.2 \times 10^3$ . It is two orders of magnitude larger than the deconvolution error. In our experiments, the error introduced only from deconvolution is small enough compared to the contribution of defocus blur to intensive noise removal.

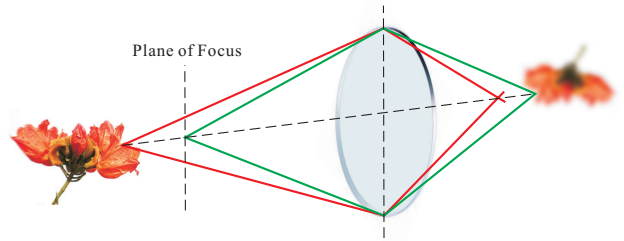


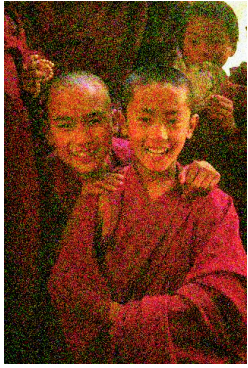
Figure 8. Optical defocusing. We adjust the lens to focus on a plane slightly *in front* of the object of interest.

One example is shown in Figure 6. Figure 6(a) is the same image shown in Figure 5(c). It is slightly defocus blurred after noise removal using our method. We perform deconvolution (implementation details in Section 6) and obtain the final result shown in Figure 6(b), which has a high PSNR. Note that the initial PSNR is only 14.

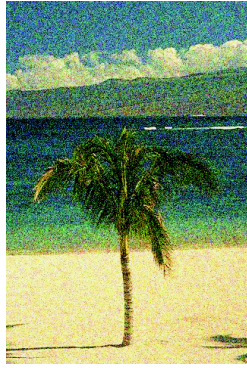
## 6. Implementation

In our experiments, we first perform photometric calibration [8] and then produce the defocus blur using the camera manual focusing function to a slightly near point instead of the ideal object plane, as depicted in Figure 8. The corresponding PSF is lens-specific and is reproducible. In contrast, image sensor noise is stochastic and image-dependent, which means any two shots cannot produce identical noise maps even with the same camera setting. To estimate the PSF, we first record the manual focusing distance  $u_1$  from the camera lens and the ideal object distance  $u_2$  using the rangefinder attached to the camera.  $u_2$  can also be obtained using the camera lens auto-focusing function. We after-





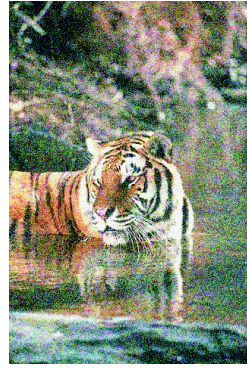
246053



46076



10080



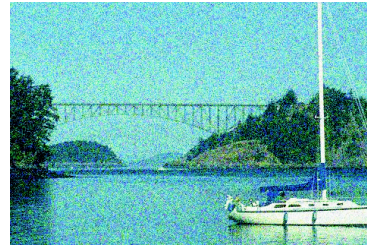
108041



209070



134008



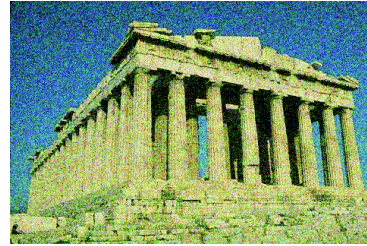
22090



247085



41004



67079



353013



166081



43070



103041



161062



176039

Figure 9. Sixteen images are selected from the Berkeley image segmentation dataset [23]. Defocus blur and noise are added for quantitatively evaluating our method.



Figure 10. Visual comparison of the denoising results. The first column shows the input to our method where both noise and blur present. Other methods use unblurred input images but with the same level of noise. Close-ups are shown in Figure 11.

wards apply the calibration technique of [20] on  $u_1$  and  $u_2$  to estimate the PSF. Note that for small PSFs, our method even allows simple Gaussian approximation without needing careful calibration, as demonstrated in a following example.

We constrain the size of the defocus blur kernel to be at most  $11 \times 11$  (pixels). We initialize the noise layer using the method of Dabov et al. [6]. This initialization does not take the blur model into consideration and thus remains considerable errors. We then apply our method, as described in Section 4, to estimate the noise map. After denoising, we use the non-blind deconvolution method [32] to remove the slight defocus blur introduced optically. Ringing artifacts are seldom produced because the noise likelihood (defined in Eq. (14)) already takes into account the blur model.

This makes the noise-removed result satisfy the convolution constraint subject to the input PSF. The denoising and deblurring steps alternate. Typically at most 10 iterations are enough to produce a visually compelling result. The running time is about 3 minutes for an image with  $800 \times 600$  pixels on a desktop PC with a Core2Duo 2.8GHz CPU.

## 7. Quantitative Evaluation

The first example shown in Figure 7 is to quantitatively evaluate the effectiveness of our method given significant image noise. The input image (shown in Figure 7(a)) is blurred (PSF shown on bottom left) followed by adding large CCD noise [21] ( $\sigma = 20\%$ ). Our image restoration result is shown in Figure 7(d). In (e)-(f), we show

PSNR	$\sigma = 10\%$				$\sigma = 15\%$			
	bilat	PDE	wavelet	Ours	bilat	PDE	wavelet	Ours
100080.jpg	26.22	32.44	32.65	34.55	28.51	31.15	31.24	34.39
<b>103041.jpg</b>	25.35	29.90	29.91	31.93	26.66	28.00	28.39	31.07
108041.jpg	24.51	28.20	28.29	29.38	24.99	25.90	26.99	28.52
134008.jpg	25.30	29.33	29.73	31.25	26.26	27.55	28.27	30.89
<b>161062.jpg</b>	25.73	29.58	30.36	30.82	27.04	28.11	28.86	30.85
166081.jpg	24.57	27.79	28.24	29.53	25.74	26.34	27.03	29.31
176039.jpg	25.69	28.67	29.06	27.70	25.36	26.46	27.22	29.23
209070.jpg	24.74	29.06	29.64	30.24	25.69	27.32	27.99	29.77
<b>22090.jpg</b>	25.59	29.46	29.97	30.22	26.22	27.51	28.30	30.25
246053.jpg	26.77	32.45	31.53	34.34	27.85	30.07	29.67	33.19
<b>247085.jpg</b>	24.76	28.51	28.92	29.62	25.98	26.80	27.50	29.69
<b>353013.jpg</b>	25.29	28.35	27.29	30.38	24.90	25.97	25.77	29.13
41004.jpg	26.25	30.93	31.09	32.72	27.64	28.86	29.78	31.32
<b>43070.jpg</b>	25.87	28.99	29.24	30.95	25.97	26.75	27.74	30.55
46076.jpg	25.85	28.62	28.69	30.77	26.17	26.71	27.60	30.33
67079.jpg	24.40	27.01	28.06	29.01	24.58	25.38	26.42	28.17
mean	25.43	29.33	<b>29.54</b>	<b>30.96</b>	26.22	27.43	<b>28.05</b>	<b>30.42</b>

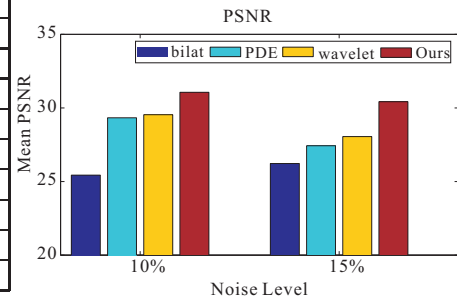


Table 1. Left: PSNRs of a set of the processed images for comparison. Images with red titles are shown in Figure 10. Our PSNRs are calculated based on the final deconvolution results. Right: histogram of the mean PSNRs.

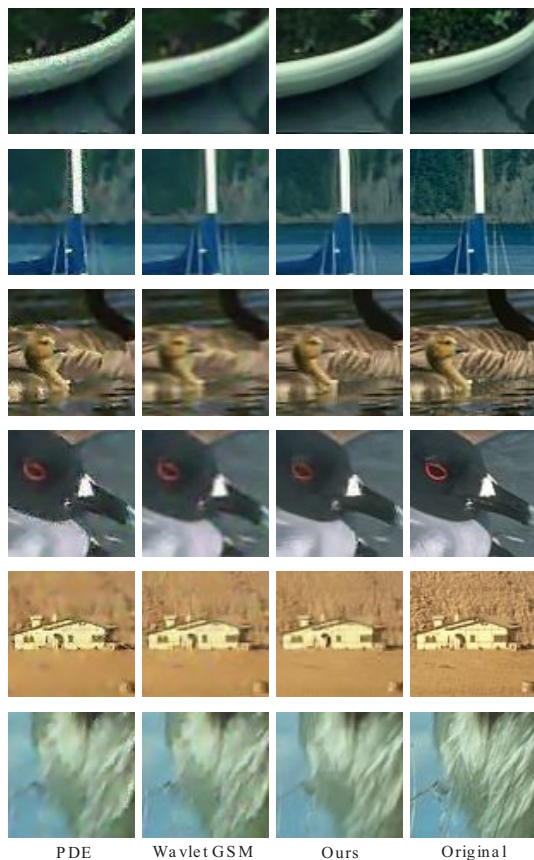


Figure 11. Close-ups of the denoising results in Figure 10.

the results from two other denoising algorithms with an un-

blurred noisy image (shown in Figure 7(b)) as input. Note that “NeatImage” [25] is one of the most effective denoising software.

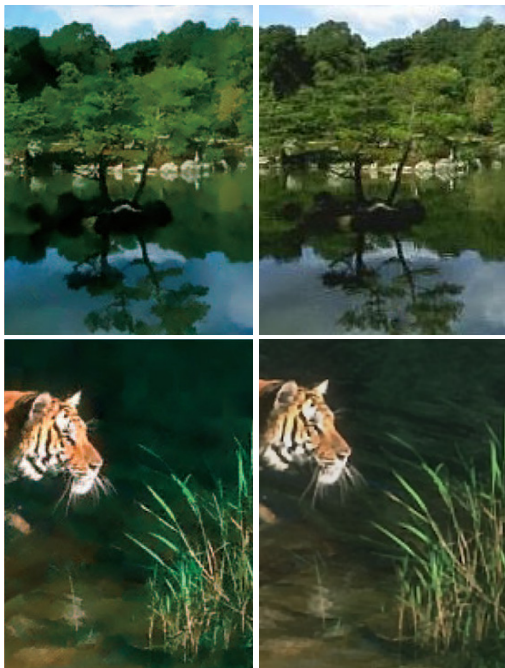
We then collect the statistics of our denoising method using a set of image examples. In this experiment, 16 images containing different types of objects and scenes were selected from the Berkeley segmentation data set [23]. We blurred these input images by convolving them with small defocus kernels estimated by Joshi et al. [17]. This process is followed by adding additive white Gaussian noise (AWGN) respectively in 10% and 15% levels. For final result comparison, several other denoising methods were also tested on the images with the same amount of noise (but without defocus blur).

The blurred images with 10% noise level are shown in Figure 9. For comparison, we also tested bilateral filtering [33], curvature preserving PDE [34], wavelet, and GSM [27]. We use default parameters for the downloaded executables. For PDE, we tuned the parameters case-by-case and found that the best results are generally obtained with  $\alpha = 1$  and  $iter = 4$  for  $\sigma = 10\%$ ,  $\alpha = 1.5$  and  $iter = 4$  for  $\sigma = 15\%$ . The visual comparisons are shown in Figures 10. For better result illustration, the noisy unblurred input for conventional denoising methods is not included.

The peak signal to noise rates (PSNRs) are given in Table 1. Thanks to the novel denoising framework and method, our results are generally with small errors. It is also notable that high PSNRs (exceeding 30) are hardly accomplished by conventional methods using non-blurry input with 15% noise. Figure 11 provides close-ups.

PSNR	$\sigma = 10\%$		
	0th	1th	Ours
File name			
100075.jpg	28.14	28.96	30.34
105053.jpg	30.63	31.95	33.27
106025.jpg	32.03	34.22	35.47
108073.jpg	28.33	29.21	29.35
113009.jpg	28.89	30.19	31.51
134052.jpg	28.61	29.55	30.20
145053.jpg	27.46	28.71	28.85
15004.jpg	25.58	27.50	27.71
15088.jpg	27.71	28.76	30.80
22013.jpg	27.14	28.84	29.06
23084.jpg	27.05	29.23	28.72
302003.jpg	27.91	30.84	31.56
314016.jpg	26.81	27.83	29.89
35008.jpg	30.24	33.27	34.00
65010.jpg	27.18	28.41	28.69
66075.jpg	29.79	31.80	32.56
mean	28.34	29.95	30.75

Table 2. PSNRs of the images compared to those in Table 2 of [21]. Columns “0th” and “1st” show the statistics obtained using orders zero and one models (described in [21]), respectively.



Liu et al. (1st order model)

Our results

Figure 12. Visual comparison with the results of Liu *et al.* [21]. Left images are published in [21]. We show our results on the right.

In addition, we compare our method with that of Liu *et al.* [21]. The PSNRs extracted from [21] are used for comparison. Again, our method uses blurred images with additive 10% AWGN while the noisy images in [21] do not

undergo the blurring process. The PSNRs are presented in Table 2. Two image close-ups are showed in Figure 12 for visual quality comparison.

## 8. More Experimental Results

We show a number of experimental results using camera captured defocused noisy images and present comparisons in this section. In Figures 13 and 14, we show images captured by a Nikon D200 camera in low-light conditions. The input images are severely underexposed with significant noise. For comparison, we also took corresponding in-focus images (shown in Figures 13(b) and 14(b)) and enhanced them as input to the other denoising methods. Their results are shown in (e) and (f), with close-ups in (i) and (j).

In Figure 16, we show that we can denoise images even if the small ground truth defocus PSF is unknown. Here, we merely guess the PSF. The example shown in this figure was used in [21]. The input image shown in Figure 16(a) is slightly out-of-focus; we empirically use a  $7 \times 7$  Gaussian kernel with standard deviation 1.2 to approximate the defocus PSF. We then follow the exact same steps used to process the other images. Our result is shown in Figure 16(d) with close-up in (f).

Two more examples are shown in Figures 15 and 17 with the input images captured by Nikon DSLR cameras using slight manual defocusing in a dark room.

Finally, we have tested cases where the foreground and background are far apart (one example is shown in Figure 18), and found that the results are visually plausible even along sharp object boundaries. The main reason is that our denoising technique makes the image result apt to satisfy a convolution model, which can almost eliminate visual artifacts in deconvolution. Similar empirical discovery was demonstrated in [20].

## 9. Concluding Remarks

We have presented a new denoising technique based on optical defocus to reduce the signal-noise coupling. We showed that the gains in noise reduction more than offset the degradation in signal due to the defocus. We also introduced a new metric for evaluating how noisy a blurred image is. Experimental results were shown to validate our technique and analysis.

**Limitations** Our technique is less effective in cases where depth cannot be quantified (e.g., in macro photography). Our method tends to work best when both foreground and background are in the field of view. Our technique also assumes a certain style of photography where the nearest object is originally in focus (which is common); it removes only a small amount of defocus. Finally, we assume that the blur PSF is spatially-invariant.

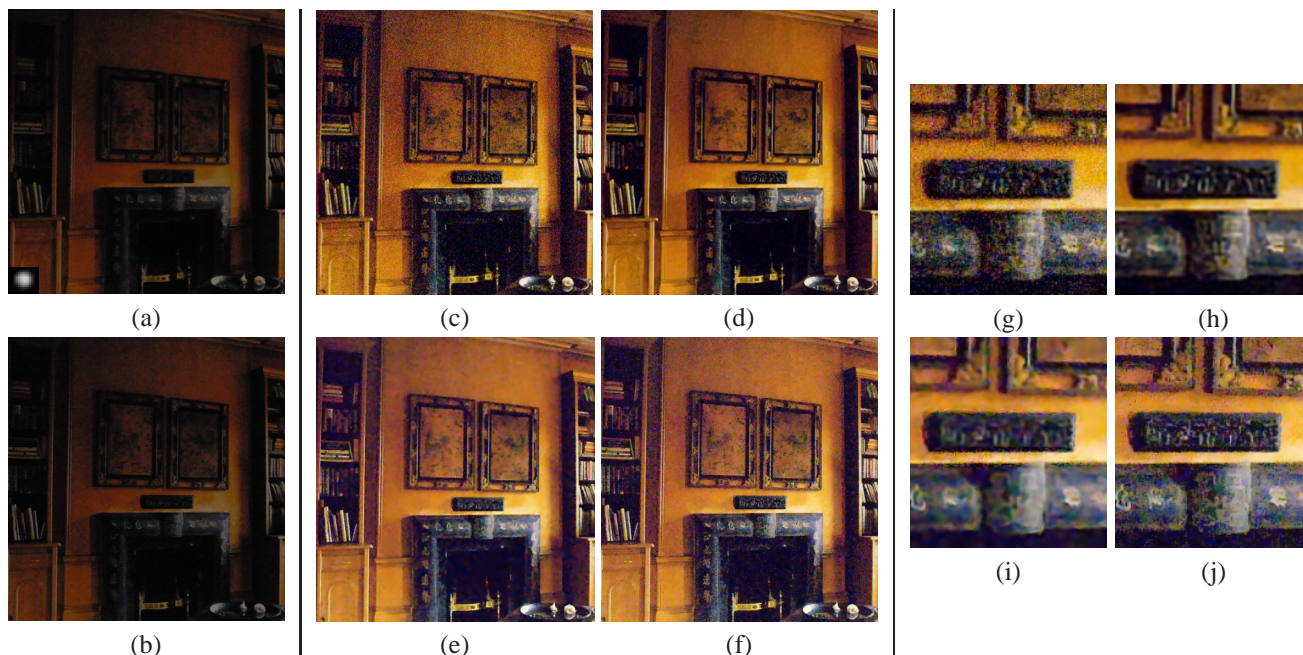


Figure 13. “Wall” example. (a) The captured under-exposed image with slight defocus blur. (b) Another captured in-focus noisy image with the same exposure setting. (c) The intensity enhanced input. (4X brightness) (d) Our final result from (c) after denoising and removing the blurriness. (e)-(f) The denoising results of PDE [34] and bilateral filtering [9] with an intensity enhanced (b) as input. (g)-(j) Close-ups of (c)-(f).

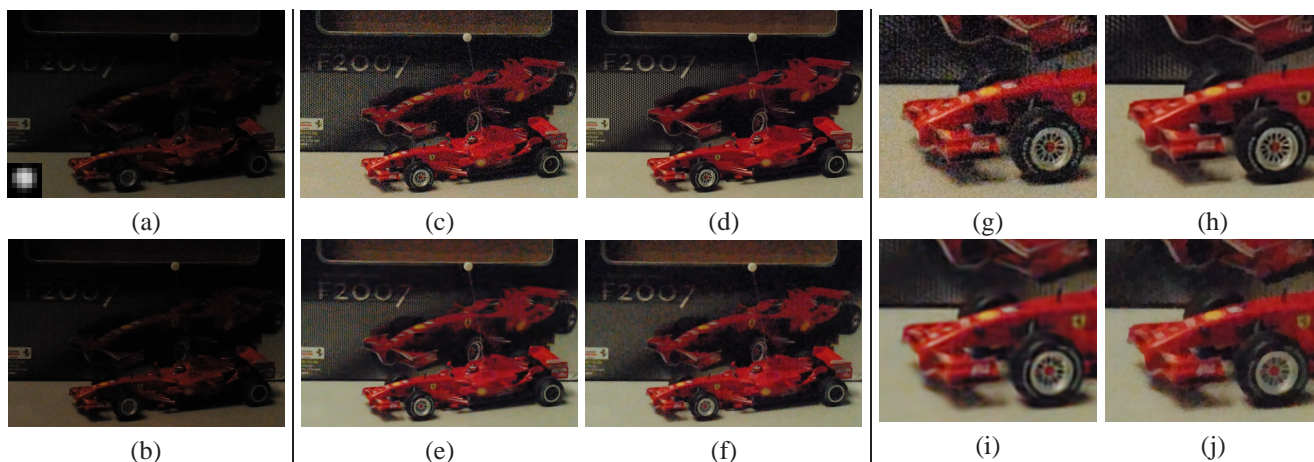


Figure 14. “Model” example. (a) The captured under-exposed image with slight defocus blur. (b) Another captured in-focus noisy image with the same exposure setting. (c) The intensity enhanced input (4X brightness). It contains significant noise. (d) Our final result from (c) after denoising and removing the blurriness. (e) The denoising result of the Gaussian mixture wavelet method [27] with (b) as input. (f) The denoising result of Rodriguez and Wohlberg [29] from an intensity enhanced version of (b). (g)-(j) Close-ups of (c)-(f).



Figure 15. Coke example. (a) Image captured under low light. (b) Image after intensity enhancement, with noise proportionally amplified. (c) Result of noise removal. (d) Final result after removing defocus blur.

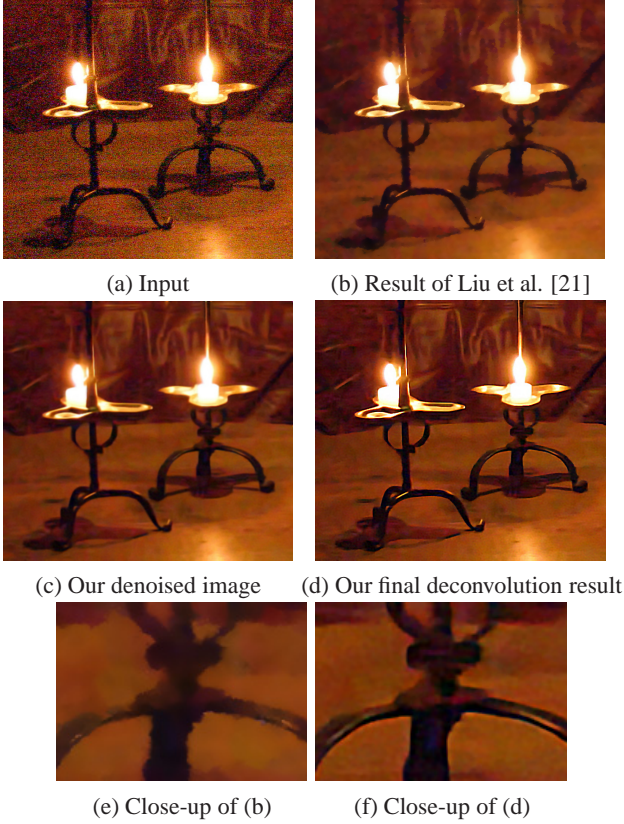


Figure 16. Candle example. (a) The input noisy image with small out-of-focus blur. (b) The result of Liu et al. [21]. (c) Our denoised image. (d) Our final result by removing the slight blur with a Gaussian kernel approximation. (e)-(f) Close-ups of (b) and (d).

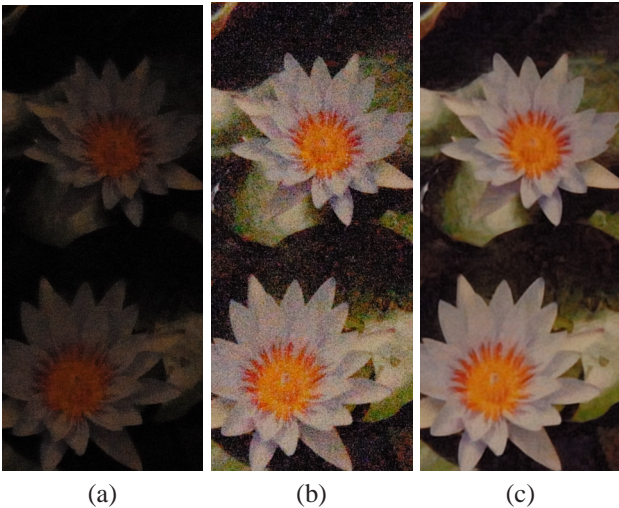


Figure 17. One more denoising example. (a) is the captured image, (b) is the image after intensity enhancement, with noise proportionally amplified, (c) shows our final result.

Our technique is tolerant of moderate errors in the PSF estimates. Figure 19 shows the results when the kernels are

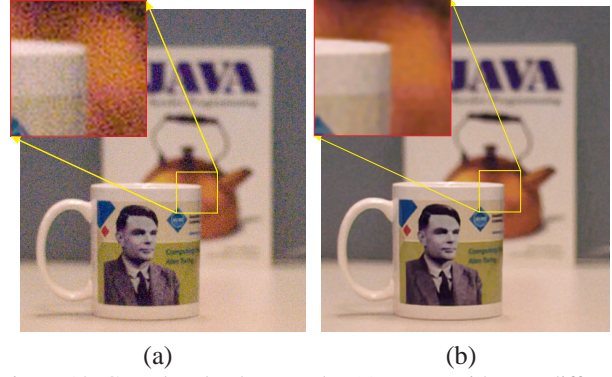


Figure 18. Complex depth example. (a) Image with very different foreground and background depths. (b) Our restored image.

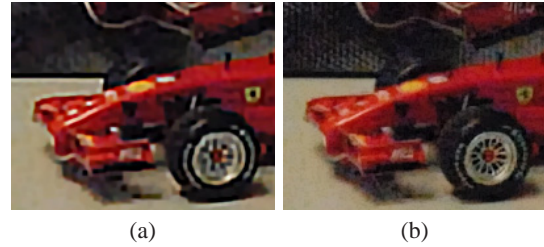


Figure 19. Image reconstructed with erroneous kernels. The input image is shown in Fig. 14(c) with ground truth kernel size  $7 \times 7$ . (a) Using  $11 \times 11$  Gaussian kernel (close-up). (b) Using  $3 \times 3$  Gaussian kernel (close-up).

largely mis-estimated. We found using an excessively large kernel over-sharpens images while a too small kernel makes the result still look blurred.

## Appendix

$$\begin{aligned} \left\| \frac{\partial \mathbf{x}}{\partial \lambda} \right\|_2^2 &= \|h(F, \lambda) \bar{\mathbf{n}} + h(F, \lambda) F \bar{\mathbf{x}}^*\|_2^2 \\ &= \|\mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\bar{\mathbf{n}}) + \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\mathbf{x}^*)\|_2^2. \end{aligned}$$

**[Proof:]**

Product  $h(F, \lambda) \bar{\mathbf{n}}$  results in a vector with the same length as  $\bar{\mathbf{n}}$ :

$$h(F, \lambda) \bar{\mathbf{n}} = \bar{\mathbf{A}}, \quad (30)$$

where  $\mathbf{A}$  is the corresponding 2D signal with similar structure as  $\mathbf{n}$ . Using Eq. (7) to replace  $h(F, \lambda)$ , Eq. (30) becomes

$$F^T \bar{\mathbf{n}} = (F^T F + \lambda)^T (F^T F + \lambda) \bar{\mathbf{A}}.$$

Taking the Fourier transform in both sides yields

$$\begin{aligned} \overline{\mathcal{F}(\bar{\mathbf{n}})} \circ \mathcal{F}(\mathbf{n}) &= \overline{(\mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\mathbf{f}) + \lambda)} \circ \\ &\quad \overline{(\mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\mathbf{f}) + \lambda)} \circ \mathcal{F}(\mathbf{A}). \end{aligned}$$

We further obtain

$$\begin{aligned}\mathcal{F}(h(F, \lambda)\bar{\mathbf{n}}) &= \mathcal{F}(\bar{\mathbf{A}}) \\ &= \mathcal{F}(\mathbf{A}) \\ &= \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{n}),\end{aligned}$$

where

$$\mathcal{H}(\mathbf{f}, \lambda) = \frac{\overline{\mathcal{F}(\mathbf{f})}}{(\overline{\mathcal{F}(\mathbf{f})} \circ \mathcal{F}(\mathbf{f}) + \lambda) \circ (\overline{\mathcal{F}(\mathbf{f})} \circ \mathcal{F}(\mathbf{f}) + \lambda)}.$$

With similar derivation, we can also obtain

$$\mathcal{F}(h(F, \lambda)F\nu(\mathbf{x}^*)) = \mathcal{H}(\mathbf{f}, \lambda) \circ \mathcal{F}(\mathbf{f}) \circ \mathcal{F}(\mathbf{x}^*).$$

□

## References

- [1] N. Azzabou, N. Paragios, F. Guichard, and F. Cao. Variable bandwidth image denoising using image-based noise models. In *CVPR*, 2007.
- [2] E. P. Bennett and L. McMillan. Video enhancement using per-pixel virtual exposures. *ACM Trans. Graph.*, 24(3):845–852, 2005.
- [3] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *CVPR*, pages 60–65, 2005.
- [4] J. Chen and C.-K. Tang. Spatio-temporal markov random field for video denoising. In *CVPR*, 2007.
- [5] J. Chen, C.-K. Tang, and J. Wang. Noise brush: Interactive high quality image-noise separation. *ACM Trans. Graph.*, 2009.
- [6] K. Dabov, A. Foi, and K. Egiazarian. Image restoration by sparse 3D transform-domain collaborative filtering. In *SPIE Electronic Imaging*, 2008.
- [7] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Color image denoising via sparse 3D collaborative filtering with grouping constraint in luminance-chrominance space. In *ICIP*, 2007.
- [8] P. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. *ACM Trans. Graph.*, 1997.
- [9] E. Eisemann and F. Durand. Flash photography enhancement via intrinsic relighting. *ACM Trans. Graph.*, 23(3):673–678, 2004.
- [10] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single photograph. *ACM Trans. Graph.*, 25:787–794, 2006.
- [11] D. J. Field. Relations between the statistics of natural images and the response properties of cortical cells. *Journal of the Optical Society of America A*, 4:2379–2394, 1987.
- [12] G. Gilboa, N. A. Sochen, and Y. Y. Zeevi. Image enhancement and denoising by complex diffusion processes. *TPAMI*, 26(8):1020–1036, 2004.
- [13] G. Healey and R. Kondepudy. Radiometric CCD camera calibration and noise estimation. *TPAMI*, 16(3):267–276, 1994.
- [14] S. Ioué and K. R. Spring. *Video Microscopy, 2nd ed.* Plenum Press, 1997.
- [15] J. Jia. Single image motion deblurring using transparency. In *CVPR*, 2007.
- [16] J. Jia, J. Sun, C.-K. Tang, and H.-Y. Shum. Bayesian correction of image intensity with spatial consideration. In *ECCV*, pages 342–354, 2004.
- [17] N. Joshi, R. Szeliski, and D. J. Kriegman. PSF estimation using sharp edge prediction. In *CVPR*, 2008.
- [18] N. Joshi, C. L. Zitnick, R. Szeliski, and D. J. Kriegman. Image deblurring and denoising using color priors. In *CVPR*, 2009.
- [19] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. In *NIPS*, 2009.
- [20] A. Levin, R. Fergus, F. Durand, and B. Freeman. Image and depth from a conventional camera with a coded aperture. *ACM Trans. Graph.*, 2007.
- [21] C. Liu, R. Szeliski, S. B. Kang, C. L. Zitnick, and W. T. Freeman. Automatic estimation and removal of noise from a single image. *TPAMI*, 30(2):299–314, 2008.
- [22] L. Lucy. Bayesian-based iterative method of image restoration. *Journal of Astronomy*, 79:745–754, 1974.
- [23] D. R. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. Technical report, University of California, Berkeley, 2001.
- [24] J. Nakamura. *Image Sensors and Signal Processing for Digital Still Cameras*. CRC Press, 2005.
- [25] NeatImage<sup>©</sup>. <http://www.neatimage.com/>. 2009.
- [26] G. Petschnigg, R. Szeliski, M. Agrawala, M. F. Cohen, H. Hoppe, and K. Toyama. Digital photography with flash and no-flash image pairs. *ACM Trans. Graph.*, 23(3):664–672, 2004.
- [27] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli. Image denoising using scale mixtures of gaussians in the wavelet domain. *TIP*, 12(11):133–1351, 2003.
- [28] N. Ratner and Y. Y. Schechner. Illumination multiplexing within fundamental limits. In *CVPR*, 2007.
- [29] P. Rodriguez and B. Wohlberg. Efficient minimization method for a generalized total variation functional. *TIP*, 18(2):322–332, 2009.
- [30] S. Roth and M. J. Black. Fields of experts: A framework for learning image priors. In *CVPR*, pages 860–867, 2005.
- [31] H. Scharr, M. J. Black, and H. W. Haussecker. Image statistics and anisotropic diffusion. In *ICCV*, pages 840–847, 2003.
- [32] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. *ACM Trans. Graph.*, 27(3), 2008.
- [33] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *ICCV*, pages 839–846, 1998.
- [34] D. Tschumperlé. Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE’s. *IJCV*, 68(1):65–82, 2006.
- [35] D. Tschumperlé and R. Deriche. Vector-valued image regularization with PDEs: A common framework for different applications. *TPAMI*, 27(4):506–517, 2005.

- [36] Y. Tsin, V. Ramesh, and T. Kanade. Statistical calibration of the CCD imaging process. In *ICCV*, pages 480–487, 2001.
- [37] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum. Image deblurring with blurred/noisy image pairs. *ACM Trans. Graph.*, 26(3):1, 2007.
- [38] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum. Progressive inter-scale and intra-scale non-blind image deconvolution. *ACM Trans. Graph.*, 27(3), 2008.