Kernel-Free Image Deblurring with a Pair of Blurred/Noisy Images

Chunzhi Gu
University of Fukui
gu-cz@monju.fuis.u-fukui.ac.jp

Xuequan Lu
Deakin University
xuequan.lu@deakin.edu.au

Ying He
Nanyang Technological University
yhe@ntu.edu.sg

Chao Zhang
University of Fukui
zhang@u-fukui.ac.jp

Abstract

Complex blur like the mixup of space-variant and space-invariant blur, which is hard to be modeled mathematically, widely exists in real images. In the real world, a common type of blur occurs when capturing images in low-light environments. In this paper, we propose a novel image deblurring method that does not need to estimate blur kernels. We utilize a pair of images which can be easily acquired in low-light situations: (1) a blurred image taken with low shutter speed and low ISO noise, and (2) a noisy image captured with high shutter speed and high ISO noise. Specifically, the blurred image is first sliced into patches, and we extend the Gaussian mixture model (GMM) to model the underlying intensity distribution of each patch using the corresponding patches in the noisy image. We compute patch correspondences by analyzing the optical flow between the two images. The Expectation-Maximization (EM) algorithm is utilized to estimate the involved parameters in the GMM. To preserve sharp features, we add an additional bilateral term to the objective function in the M-step. We eventually add a detail layer to the deblurred image for refinement. Extensive experiments on both synthetic and real-world data demonstrate that our method outperforms state-of-the-art techniques, in terms of robustness, visual quality and quantitative metrics. We will make our dataset and source code publicly available.

1. Introduction

It is prevalent to adopt image deblurring techniques to recover quality images from blurry images. A common situation is capturing photos in dimly-lit environments (e.g., photographing moving objects in a night scene), where one can hardly get sharp and bright photos. Most likely, the taken photos are dark or blurry, depending on the camera settings and object conditions. Though a lower shutter speed can effectively increase brightness, it almost inevitably leads to blur. On the other hand, increasing the shutter speed makes the camera sensor or film exposed to limited light, resulting in dark photos. Setting a high ISO for increasing brightness is a trade-off way to obtain bright photos. Nevertheless, a higher gain setting amplifies noise which may even worsen the photo quality. Recovering quality photos from such captured blurry photos remains chal-
Removing blur from blurred images to achieve latent sharp images has been widely studied \[32, 22\]. Many approaches \[16, 7\] estimate the blur kernels using salient features. Such methods may fail when images are not bright enough to get sufficient features such as edges. In fact, it is difficult to model blur in real photos in some cases, because of the mix of different types of blur (i.e., complex blur). As a result, deblurring methods based on blur kernel estimation have limited performance in handling complex blur.

The main contributions of this paper are threefold. First, we propose a novel deblurring approach called optical flow guided GMM (OGMM) with a pair of blurred/noisy images. Second, we formulate deblurring as a parameter estimation problem, and derive an EM algorithm to optimize the involved parameters. Eventually, a bilateral term is added to the objective function of the M-step in EM to better preserve sharp features, and a detail layer is extracted to enhance the details in the deblurred image. Instead of kernel estimation or deconvolution, we make full use of the noisy image taken in a different view for deblurring.

2. Related work

Blind deblurring for single image. Blind deblurring aims to accurately estimate the unknown blur kernel, based on which deconvolution is performed to recover the corresponding sharp image. There are several types of methods for blind deblurring, such as maximum a posterior (MAP) \[17, 35\], variational Bayes \[11, 41, 33\] and edge prediction \[13, 16\]. For the MAP based methods, various strategies are presented to cope with the problem revealed by Levin et al. \[18\] that the failure of naïve MAP may occur because it favors no-blur explanations. Marginal distributions are considered to be maximized over all possible images \[18, 11\]. Image regularizations are introduced into the MAP framework \[17, 36, 24\] to retain salient image structures. The state-of-the-art methods for blind deblurring also depend on rich information hidden in the blur. Yan et al. \[37\] proposed an image prior named Extreme Channels Prior (ECP) to help the uniform kernel estimation based on the observation that the values of bright channel pixels are likely to decrease. Hu et al. \[15\] utilized light streaks in the images taken in low light situations as constraints for estimating the blur model, but it only succeeds when the light streak is large. Single image blind deblurring usually encounters the bottleneck that the useful information for kernel estimation is insufficient, and can hardly output a proper blur model in real cases.

Multiple images deblurring. Efforts have been made to multiple images deblurring \[4, 12, 38, 40, 25, 6, 42, 19\]. The superiority of deblurring with multiple images lies in the complementary information provided in those images. Hee et al. \[12\] introduced a Gyro-Based method to cope with handshake blur caused by camera motion. Multiple blurred images can provide necessary frequency components which are missing due to blur. However, it can hardly handle object movement. Cai et al. \[4\] aligned multiple motion blurred frames accurately and show promising results with their tight framelet system. Li et al. \[19\] used two well-aligned blurred images to better estimate the blur kernel. Zhang et al. \[40\] estimate the latent sharp image with given multiple blurry and/or noisy images by designing a penalty function which can balance the effects of observations with varying quality and avoid local minimal. However, they assume a single type of linear motion blur or uni-
Adding detail
layer
(OGMM+DL)

Deblurring
(OGMM)

Figure 2: Overview of our image deblurring approach. The Deblurring stage is an iterative procedure. The deblurred image \( I_1^{t+1} \) at the \((t+1)\)-th iteration is updated based on the deblurred result \( I_1^t \) at the \(t\)-th iteration. In the Adding detail layer stage, the detail layer can be extracted using the Laplacian mask image. The spatial inconsistency between \( I_1^t \) and the detail layer is solved by the updated optical flow.

form blur. In fact, none of the above approaches can handle complex blur.

**Patch based GMM framework.** Gaussian mixture model has been widely exploited in image restoration tasks [39, 28, 43, 34, 27, 26] and point cloud processing tasks [21, 20]. In [43], Gaussian mixture priors are learned from a set of natural images. By maximizing the expected patch log likelihood, an image without distortion can be reconstructed with priors. The learned patch group Gaussian mixture model (PG-GMM) by Xu et al. [34], providing dictionaries and regularization parameters, achieves a high denoising performance. The study by Zoran et al. [44] gives a comprehensive analysis that modeling natural images by GMM is effective in log likelihood scores, denoising performance and sample quality. However, GMM based learning methods commonly suffer from huge computational time and a massive dataset. We exploit GMM in a different way, which relates the patches in the noisy image with the patches in the latent image of the blurred image according to dense optical flow. In other words, we attempt to model the intensity distribution in each patch instead of learning patch based image priors to restore images.

### 3. Method

Figure 2 illustrates the overview of our method, which consists of two stages: deblurring and adding detail layer. The latter can be viewed as post-processing or refinement. We first adopt optical flow [10] to find the corresponding patches between the blurred image and the noisy image. We then formulate the image deblurring problem under the framework of GMM, and adopt the EM algorithm [29] to optimize the involved parameters. We further add a bilateral term to the objective function in the M-step, to prevent smoothing out the sharp features. Optical flow update and the EM algorithm are alternately called, to achieve the best deblurring results. Finally, we extract a detail layer from the noisy image and add it to the deblurred image, to better preserve the details.

#### 3.1. Patch Correspondence

The blurred image \( I_1 \) is decomposed into a set of overlapping square patches \( C = \{ c_1, ..., c_P \} \), where \( c_i \in \mathbb{R}^M \) and \( M = s_1 \times s_1 \). \( P \) is the number of the patches, and \( s_1 \) denotes the patch size in \( I_1 \), and \( M \) is the number of pixels in each patch. The set of pixel intensities in an arbitrary patch from \( I_1 \) is denoted as \( X (X \in \mathbb{R}^M) \), and \( x_{xy} \) denotes a pixel intensity in \( X \). We extend the dense optical flow (DOF) [10] to find \( c_i \)'s corresponding patch \( d_j \) in the noisy image \( I_2 \). Note that for brighter and clearer visualization purposes, in the case of real images, brightness and contrast of \( I_2 \) are obtained by adjusting gain, bias, and gamma correction parameters. Here, patch \( c_i \) has correspondence to patch \( d_j \) if the two center pixels of \( c_i \) and \( d_j \) are connected with respect to the DOF field. The set of corresponding patches in \( I_2 \) can then be denoted as \( D = \{ d_1, ..., d_j, ..., d_P \} \), where \( d_j \in \mathbb{R}^K \), \( K = s_2 \times s_2 \), \( s_2 \) is the patch size in \( I_2 \) and \( s_2 \geq s_1 \). The pixel intensity set of an arbitrary \( d_j \) is indicated as \( Y, Y \in \mathbb{R}^K \), and \( y_k \) is a pixel intensity in \( Y \).
3.2. The Probabilistic Model

Our key idea is to model the underlying distribution of pixel intensities with the noisy observation $Y$. We use $X = \{x_m\}$ to denote the corresponding latent pixel intensities, for slight notation misuse. To relate $X$ with $Y$, we assume that $y_k$ follows a GMM whose centroids are $\{x_m\}$. That is, the GMM with those centroids can generate the noisy observations. Thus, we formulate the deblurring problem under the GMM probabilistic framework. The probability density function of $y_k$ is defined as

$$p(y_k) = (1 - \omega) \frac{1}{M} \sum_{m=1}^{M} p(y_k|x_m) + \omega \frac{1}{K},$$  

(1)

where $p(y_k|x_m) = \frac{1}{(2\pi \sigma^2)^{d/2}} e^{-\frac{\|y_k - x_m\|^2}{2\sigma^2}}$ denotes the $m$-th Gaussian component, and $d$ is the dimension of $x_m$ and $y_k$ ($d = 1$ for gray image). An additional uniform distribution $\frac{1}{K}$ accounts for the noise, with a weight $\omega$. $\sigma^2$ is the isotropic covariance and $\frac{1}{\sigma}$ represents the equal membership probability for all the Gaussian components. The centroids of the GMM model is initialized by $X$. We next need to find the centroids and covariances that can best explain the distribution of $Y$.

3.3. EM optimization

The centroids and covariances of the GMM can be estimated by minimizing the negative log-likelihood function [2].

$$E(X, \sigma^2) = -\sum_{k=1}^{K} \log \left(\frac{1}{M} \sum_{m=1}^{M} p(y_k|x_m) + \omega \frac{1}{K}\right).$$  

(2)

We use the expectation-maximization (EM) algorithm [9] to solve Eq. (2). The EM algorithm consists of two steps: E-step and M-step. E-step and M-step are alternately called for multiple iterations to achieve decent estimations.

**E-step.** The posterior probability $p^{old}(x_m|y_k)$ is calculated based on Bayes’ theorem and the parameters in the previous iteration. $p^{old}_{mk}$ represents $p^{old}(x_m|y_k)$ for simplicity.

$$p^{old}_{mk} = \frac{e^{-\frac{\|y_k - x_m\|^2}{2\sigma^2}}}{\sum_{m=1}^{M} e^{-\frac{\|y_k - x_m\|^2}{2\sigma^2}} + \omega M (2\pi \sigma^2)^{d/2} + \omega \frac{1}{K}}.$$  

(3)

**M-step.** The M-step is to update the involved parameters ($X$ and $\sigma^2$) based on the computed posteriors. This is equivalent to minimizing the upper bound of Eq. (2). “new” means calculating the posterior probability with the parameters to be estimated in the current iteration.

$$Q(X, \sigma^2) =$$

$$-\sum_{k=1}^{K} \sum_{m=1}^{M} p^{old}_{mk} \log \left(\frac{1-M \omega}{M} p^{new}(y_k|x_m) + \frac{\omega}{M} \right) p^{old}_{mk}$$

$$\propto \sum_{k=1}^{K} \sum_{m=1}^{M} p^{old}_{mk} \|y_k - x_m\|^2 - \sum_{k=1}^{K} \sum_{m=1}^{M} \sum_{m'=N(m)} p^{old}_{mk} d \log \sigma^2.$$  

(4)

3.4. Bilateral Term

Eq. (4) can be treated as a data term, which in this work is to numerically approximate $Y$ with $X$. However, this data term only takes the pixel intensity distribution into account, without considering the spatial information. As illustrated in Fig. 3(a), sharp edges would become coarse (e.g., discontinuity) in the iteration of EM. To overcome this problem, we propose to add a bilateral term to the objective function in M-step. Inspired by the bilateral filter [29], we define the bilateral term $B$ as

$$B(X) = \sum_{m' \in N(m)} (x_m - x_{m'}) e^{-\frac{d_{m,m'}^2}{2\sigma^d}} e^{-\frac{l_{m,m'}^2}{2\sigma_l^2}},$$  

(5)

where $m' \in N(m)$ denotes a neighbour pixel with its intensity equals to $x_{m'}$. $d_{m,m'}$ and $l_{m,m'}$ are the spatial distance and the difference of intensity value between the neighbour pixel $m'$ and the center pixel $m$, respectively. $\sigma_d$ and $\sigma_l$ are constants to control the degree of smoothness.

Redefining Eq. (4) as $D(X, \sigma^2)$ and weighing it with $\lambda$, the final objective function can be written as

$$Q(X, \sigma^2) = \lambda D(X, \sigma^2) + (1 - \lambda) B(X).$$  

(6)

We next need to minimize Eq. (6), to solve the involved parameters.

3.5. Minimization

We take the partial derivative of Eq. (6) with respect to $x_m$ and $\sigma^2$, respectively. By solving $\partial Q/\partial x_m = 0$ and $\partial Q/\partial \sigma^2 = 0$, the new $x_m$ is updated as

$$x_m = \lambda \sum_{k=1}^{K} p^{old}_{mk} y_k + (1-\lambda) \sum_{m' \in N(m)} \frac{d_{m,m'}^2}{2\sigma_d^2} e^{-\frac{l_{m,m'}^2}{2\sigma_l^2}} x_{m'},$$

$$\sum_{m' \in N(m)} e^{-\frac{d_{m,m'}^2}{2\sigma_d^2}} e^{-\frac{l_{m,m'}^2}{2\sigma_l^2}}$$  

(7)
and the new $\sigma^{2'}$ is updated as

\[
\sigma^{2'} = \left[ \sum_{k=1}^{K} \sum_{m=1}^{M} p^{old}(x'_m | y_k) y_k y_k^T \right. \\
+ \sum_{k=1}^{K} \sum_{m=1}^{M} p^{old}(x'_m | y_k) x'_m x'_m T \\
- 2 \sum_{d=1}^{D} \sum_{k=1}^{K} \sum_{m=1}^{M} p^{old}(x'_m | y_k) x'_m y_k y_k^T / \\
\left. \sum_{k=1}^{K} \sum_{m=1}^{M} p^{old}(x'_m | y_k). \right]
\]

Notice that the step size for slicing $I_1$ into patches should be small so that a certain pixel can be updated in different GMM models due to overlapped patches. The final output value of a certain pixel $I_{u,v}$ is calculated by simply averaging all the updated values located at $(u,v)$,

\[
I(u,v) = \frac{\sum_{i,m} \mathbb{I}(\text{pos}^i_m = (u,v)) x^i_m}{\sum_{i,m} \mathbb{I}(\text{pos}^i_m = (u,v))},
\]

where $\text{pos}^i_m$ denotes the position of pixel $m$ in $i$-th patch, $x^i_m$ denotes the pixel intensity of $m$ in $i$-th patch. $\mathbb{I}$ is an indicator function. See Fig. 4 for an example.

3.6. Optical Flow Update

Blur hinders accurate estimation of optical flow, which can possibly lead to inaccurate matches in finding patch correspondences. To mitigate this issue, we alternate optical flow and the EM algorithm for multiple iterations. $I^T_1$ denotes the deblurred result in the $t$-th iteration ($T$ times in total), and is used to compute the optical flow in the $(t+1)$-th iteration. Updating optical flow increases the confidence of the patch correspondences.

Figure 3: Deblurred results with or without the bilateral term: (a) without the bilateral term (i.e., $\lambda = 1.0$ in Eq. (6)), (b) with the bilateral term ($\lambda = 0.75$ in Eq. (6)).

3.7. Detail Layer

We extract the sharp features from $I_2$ and add it back to $I^T_1$ to further preserve the details. A similar idea has been used in [38]. Since the noise in $I_2$ can negatively affect the quality of the detail layer, we apply the bilateral filter [29] to $I_2$ at first. We then obtain a mask $I_m$ by applying the Laplacian filter [3] to $I_2$, to select the retained details. Since the $I_2$ and $I^T_1$ are in different views, we use the DOF field between them to find the spatial correspondence (see Sec. 3.1). We can observe from Fig. 5 that the details are better recovered by adding the detail layer.

The algorithm of adding details is listed in Alg. 1.
Algorithm 1 Adding detail layer (DL)

**Input:** Deblurred image $I_T^t$, enhanced noisy image $I_2$, constant threshold $\tau \in [10, 50]$
**Output:** Deblurred image with sharp features added

1. Apply bilateral filter to $I_2$
2. Apply Laplacian filter to $I_2$ to obtain mask image $I_m$
3. for every pixel located at $(u, v)$ in $I_T^t$ do
   4. Find the correspondence according to vector in DOF field from $(u, v)$ in $I_T^t$ to $(u', v')$ in $I_2$
   5. if $I_m(u', v') > \tau$ then
      6. $I_T^t(u, v) \leftarrow (I_2(u', v') + I_T^t(u, v))/2$
   end if
4. end for
5. return Updated $I_T^t$

Algorithm 2 Image deblurring (OGMM+DL)

**Input:** Blurred image $I_1$, enhanced noisy image $I_2$, iteration times $T$, termination parameter $\gamma$
**Output:** Deblurred image $I_T^t$

1. Parameters setting: $\sigma^2 \in [100, 500]$, $\omega = 0.01$, $\lambda \in [0.75, 0.8]$, $\gamma = 0.05$
2. for $t = 1$ to $T$ do
   3. Update optical flow and find corresponding patches with respect to $I_{t-1}$ and $I_2$ ($I_0 = I_1$)
   4. for Each patch in $I_{t-1}$ do
      5. Initialize centroids by $X$
   6. repeat
      7. E-step: update each $p_{i,m}^{\text{old}}$ by Eq. (5)
      8. M-step: update each $x_m$ and $\sigma^2$ by Eq. (7) and Eq. (8)
   9. until The decrease ratio of log-likelihood by Eq. (9) is smaller than $\gamma$
   10. end for
   11. Obtain $I_1$ via Eq. (2)
12. end for
13. Add details to $I_T^t$ by Alg. 2
14. return $I_T^t$

4. Experimental Results

We evaluate our approach on both synthetic data and real-world data. Quantitative comparisons on synthetic data with ground truth are also carried out.

4.1. Synthetic Data

We first assess the performance of some current deblurring methods and our approach on ten image pairs from the publicly available dataset [1]. The dataset consists of multiple pairs of images taken from two different views in various scenes. To demonstrate the robustness of our method to different blur models, we synthetically generate six types of blur: (1) linear motion blur, (2) circular motion blur, (3) the mixture of circular motion blur, linear motion blur and Gaussian blur, (4) the mixture of two types of linear motion blur and circular motion blur, (5) the mixture of circular motion blur, zoom motion blur and two types of linear motion blur, and (6) the mixture of two types of linear motion blur and circular motion blur. The visualization of each type of blur is shown in Fig. 6. The first image in each pair is blurred with these six types of blur, respectively. Gaussian noise is added to the second image for generating the noisy image.

We compare our approach with the deblurring methods [40, 38] which can also take a pair of such images as input. To our knowledge, deblurring using a pair of blurred/noisy images has been sparsely treated so far, and the method [38] is the closest to ours. We also compare our method with three single image deblurring methods [33, 23, 14], including two baseline methods [23, 14].

As suggested by previous works [30, 31], we compute three metrics (in an average sense) for quantitative comparisons: peak signal-to-noise ratio (PSNR), structural similarity (SSIM) and mean square error (MSE). Tab. 1 displays the PSNR, SSIM and MSE, which are calculated between the deblurred images and the corresponding ground-truth image. We can see from Tab. 1 that our approach is more accurate than the two baseline methods [23, 14] and the state-of-the-art techniques [40, 38, 33], which proves the strong robustness of our method to various blur. Also, it is worth pointing out that our method with the adding details step achieves an even higher accuracy, and is more visually pleasing (Fig. 7).
Table 1: Comparisons of average PSNR/SSIM/MSE on test images which are corrupted with different types of synthetic blur. *BlurType1* is linear motion blur, *BlurType2* is circular motion blur, and *BlurType3* to *BlurType6* are complex blur mixed with multiple types of blur. See Fig. 6 for illustration.

| BlurType | RL [23] | Deconvblind [14] | Whyte [33] | SBD (single) [40] | SBD (multiple) [40] | GCRL+DL [38] | OGM +DL | RL [23] | Deconvblind [14] | Whyte [33] | SBD (single) [40] | SBD (multiple) [40] | GCRL+DL [38] | OGM +DL |
|----------|---------|------------------|-----------|------------------|-------------------|-----------|------|---------|----------------|-----------|------------------|-------------------|-----------|-------|
| BlurType1 | 17.787 / 0.804 / 1244.060 | 21.910 / 0.913 / 522.051 | 23.739 / 0.901 / 846.817 | 25.354 / 0.956 / 253.032 | 12.354 / 0.471 / 4356.241 | 20.939 / 0.883 / 614.729 | 27.716 / 0.967 / 139.489 | 19.863 / 0.856 / 826.026 | 23.606 / 0.931 / 354.657 | 24.927 / 0.947 / 241.660 | 11.537 / 0.446 / 4748.240 | 21.798 / 0.884 / 504.584 | 26.316 / 0.958 / 188.936 | 26.494 / 0.959 / 181.357 |
| BlurType2 | 18.766 / 0.822 / 1013.496 | 22.736 / 0.916 / 416.793 | 22.147 / 0.879 / 603.632 | 24.313 / 0.932 / 276.382 | 12.728 / 0.499 / 3822.419 | 21.578 / 0.877 / 524.715 | 25.289 / 0.948 / 228.670 | 19.499 / 0.845 / 891.206 | 23.153 / 0.924 / 381.380 | 24.629 / 0.942 / 255.831 | 23.644 / 0.920 / 340.248 | 21.722 / 0.881 / 515.390 | 25.965 / 0.954 / 202.588 | 26.119 / 0.955 / 196.155 |
| BlurType3 | 19.413 / 0.838 / 908.450 | 22.900 / 0.917 / 405.950 | 21.687 / 0.866 / 684.806 | 24.199 / 0.931 / 286.646 | 11.833 / 0.452 / 4521.26 | 21.576 / 0.877 / 526.811 | 25.287 / 0.948 / 225.174 | 19.357 / 0.833 / 886.731 | 22.765 / 0.920 / 340.248 | 23.644 / 0.920 / 340.248 | 21.494 / 0.874 / 539.748 | 21.494 / 0.874 / 539.748 | 24.699 / 0.940 / 258.984 | 24.794 / 0.941 / 253.556 |

Fig. 7 shows the results by different deblurring methods. Since the method in [40] can handle both single image and multiple images, we show two versions of this method for comparisons. Despite the fact that the method [40] puts emphasis on automatically distinguishing blurred images from noisy images, it tends to mistake the noisy image for the blurred image and conduct deblurring to the noisy image, shown in Fig. 7(g). This is because the pair of images are from two different views, a limitation for both methods [40, 38]. As can be observed from Fig. 7(h), information of the blurred image is almost overlapped and ignored by [38] when intense blur occurs and the noisy image dominates the final result. This may further result in “ghost area” when the difference of views gets large. As a result, it has low PSNR...
Figure 8: Visual comparison on real-world data. (a) Blurred image taken with the shutter speed of 0.5 second and ISO of 100. (b) Noisy image taken with the shutter speed of 0.01 second and ISO of 3200, and is further enhanced by gamma correction ($\gamma = 1.5$). The blown-up windows in (a) and (b) show different appearances because the two images are taken in different views.

and SSIM values, reflected in Tab. 1. The single image deblurring version of [40] performs well on the linear motion blur case, but fails to deal with the remaining five kinds of non-uniform blur.

4.2. Real-World Data

We test our approach on various kinds of blurred/noisy image pairs which are captured in low light environments using an off-the-shelf camera. Also, we compare our method with the state-of-the-art techniques [40, 38].

We adopt the following procedure to take a real-world photo pair. First, we set a low ISO and a low shutter speed to obtain the blurred image. In the process of capturing, we add a camera shake, or move the object on purpose to produce stronger blur. Secondly, we use a high ISO and a high shutter speed to obtain the noisy image. Different from the synthetic data, the captured noisy images are too dark to use directly. Before deblurring, the noisy image is enhanced by synchronizing its brightness with the blurred image. The enhancement is achieved via gain/bias change and gamma correction, which also amplifies noise.

Fig. 8 shows an example of visual comparisons. The blur kernels estimated by [40], using single or multiple images, have difficulty in recovering the sharp image. The method [38] requires the same capturing view for the blurred/noisy image pair, which limits their applicability. As presented in the close-up view, it can be easily observed that heavy misalignment occurs when adding their generated detail layer back. The result by our method, without the need of kernel estimation, enjoys significantly better visual quality than those by the state-of-the-art methods [40, 38]. It should be noted that directly denoising the noisy image in the pair can hardly achieve a desired outcome, due to differing views and own artifacts. We show a visual comparison in Fig. 9. Moreover, the transformation from the noisy image view to the blurred image view may lead to further artifacts.

Figure 9: Comparison with the denoised results of Fig. 8(b). (b) and (c) are overly smoothed to some extent. (d) involves artifacts in sharp edges of the characters. These are typical problems for image denoising methods.

Figure 10: A failure example. A large view difference deteriorates the accuracy of optical flow, thus leading to a distorted result.

5. Conclusion

We proposed a novel, robust image deblurring method with the use of a pair of blurred/noisy images. Our approach
first builds patch correspondences between the blurred and noisy images, and then relates the latent pixel intensities with the noisy pixel intensities under the GMM framework. We introduced a bilateral term for better features preservation. To refine the deblurred result, we extract and add a detail layer to it. Our approach is free of blur kernel estimation and robust to various types of blur. Extensive experiments over the synthetic and real-world data demonstrated that our method outperforms state-of-the-art techniques, in terms of both visual quality and quantity.

The major limitation of our method is its dependency on optical flow. If the motion gap between the two images is large, the accuracy of optical flow deteriorates. As a result, this would alter object appearance or reshape some sharp features, as demonstrated in Fig. 10. In the future, we would like to exploit more effective relationship among patches to address the issue of undesired optical flow.

References

[1] S. Baker, D. Scharstein, J. Lewis, S. Roth, M. J. Black, and R. Szeliski. A database and evaluation methodology for optical flow. International Journal of Computer Vision (IJCV), 92(1):1–31, 2011.

[2] C. M. Bishop et al. Neural networks for pattern recognition. Oxford university press, 1995.

[3] P. Burt and E. Adelson. The laplacian pyramid as a compact image code. IEEE Transactions on communications, 31(4):532–540, 1983.

[4] J.-F. Cai, H. Ji, C. Liu, and Z. Shen. Blind motion deblurring using multiple images. Journal of computational physics, 228(14):5057–5071, 2009.

[5] F. Chen, L. Zhang, and H. Yu. External patch prior guided internal clustering for image denoising. In Proceedings of the IEEE international conference on computer vision (CVPR), pages 603–611. IEEE, 2015.

[6] J. Chen, L. Yuan, C.-K. Tang, and L. Quan. Robust dual motion deblurring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1–8. IEEE, 2008.

[7] S. Cho and S. Lee. Fast motion deblurring. ACM Transactions on Graphics (TOG), 28(5):145, 2009.

[8] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image restoration by sparse 3d transform-domain collaborative filtering. In Image Processing: Algorithms and Systems VI, volume 6812, page 681207. International Society for Optics and Photonics, 2008.

[9] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. Journal of the royal statistical society. Series B (methodological), pages 1–38, 1977.

[10] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In Scandinavian conference on Image analysis, pages 363–370. Springer, 2003.

[11] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single photograph. ACM Transactions on Graphics (TOG), 25(3):787–794, 2006.

[12] S. Hee Park and M. Levoy. Gyro-based multi-image deconvolution for removing handshake blur. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3366–3373. IEEE, 2014.

[13] M. Hirsch, S. Sra, B. Schölkopf, and S. Harmeling. Efficient filter flow for space-variant multiframe blind deconvolution. In Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR), pages 607–614. IEEE, 2010.

[14] T. J. Holmes, S. Bhattacharyya, J. A. Cooper, D. Hansel, V. Krishnamurthi, W.-c. Lin, B. Roysam, D. H. Szarowski, and J. N. Turner. Light microscopic images reconstructed by maximum likelihood deconvolution. In Handbook of biological confocal microscopy, pages 389–402. Springer, 1995.

[15] Z. Hu, S. Cho, J. Wang, and M.-H. Yang. Deblurring low-light images with light streaks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3382–3389. IEEE, 2014.

[16] N. Joshi, R. Szeliski, and D. J. Kriegman. Psf estimation using sharp edge prediction. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1–8. IEEE, 2008.

[17] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 233–240. IEEE, 2011.

[18] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1964–1971. IEEE, 2009.

[19] W. Li, J. Zhang, and Q. Dai. Exploring aligned complementary image pair for blind motion deblurring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 273–280. IEEE, 2011.

[20] X. Lu, H. Chen, S.-K. Yeung, Z. Deng, and W. Chen. Unsupervised articulated skeleton extraction from point set sequences captured by a single depth camera. pages 7226–7234, 2018.

[21] X. Lu, S. Wu, H. Chen, S. Yeung, W. Chen, and M. Zwicker. Gpf: Gmm-inspired feature-preserving point set filtering. IEEE Transactions on Visualization and Computer Graphics (TVCG), 24(8):2315–2326, Aug 2018.

[22] A. Mahalakshmi and B. Shanthini. A survey on image deblurring. In Proceedings of the International Conference on Computer Communication and Informatics (ICCCI), pages 1–5, Jan 2016.

[23] W. H. Richardson. Bayesian-based iterative method of image restoration. JOSA, 62(1):55–59, 1972.

[24] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. ACM Transactions on Graphics (TOG), 27(3):73, 2008.

[25] F. Sroubek and P. Milanfar. Robust multichannel blind deconvolution via fast alternating minimization. IEEE Transactions on Image processing, 21(4):1687–1700, 2012.
[26] L. Sun, S. Cho, J. Wang, and J. Hays. Good image priors for non-blind deconvolution. In European Conference on Computer Vision (ECCV), pages 231–246. Springer, 2014.

[27] A. M. Teodoro, J. M. Bioucas-Dias, and M. A. Figueiredo. Image restoration and reconstruction using variable splitting and class-adapted image priors. In Proceedings of the IEEE Conference on International Conference on Image Processing (ICCV), pages 3518–3522. IEEE, 2016.

[28] A. M. Teodoro, J. M. Bioucas-Dias, and M. A. Figueiredo. Image restoration with locally selected class-adapted models. In Machine Learning for Signal Processing (MLSP), 2016 IEEE 26th International Workshop on, pages 1–6. IEEE, 2016.

[29] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In Computer Vision, 1998. Sixth International Conference on, pages 839–846. IEEE, 1998.

[30] F. Vankawala, A. Ganatra, and A. Patel. A survey on different image deblurring techniques. International Journal of Computer Applications (IJCA), 116(13):15–18, 2015.

[31] S. Vasu, V. Reddy Maligireddy, and A. Rajagopalan. Non-blind deblurring: Handling kernel uncertainty with cnns. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3272–3281. IEEE, 2018.

[32] R. Wang and D. Tao. Recent progress in image deblurring. arXiv preprint arXiv:1409.6838, 2014.

[33] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. International journal of computer vision (IJCV), 98(2):168–186, 2012.

[34] J. Xu, L. Zhang, W. Zuo, D. Zhang, and X. Feng. Patch group based nonlocal self-similarity prior learning for image denoising. In Proceedings of the IEEE international conference on computer vision (ICCV), pages 244–252. IEEE, 2015.

[35] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In Proceedings of the European conference on computer vision (ECCV), pages 157–170. Springer, 2010.

[36] L. Xu, S. Zheng, and J. Jia. Unnatural lp sparse representation for natural image deblurring. In Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR), pages 1107–1114. IEEE, 2013.

[37] Y. Yan, W. Ren, Y. Guo, R. Wang, and X. Cao. Image deblurring via extreme channels prior. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4003–4011. IEEE, 2017.

[38] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum. Image deblurring with blurred/noisy image pairs. ACM Transactions on Graphics (TOG), 26(3):1, 2007.

[39] Z. Zha, X. Zhang, Q. Wang, Y. Bai, and L. Tang. Image denoising using group sparsity residual and external nonlocal self-similarity prior. arXiv preprint arXiv:1701.00723, 2017.

[40] H. Zhang, D. Wipf, and Y. Zhang. Multi-image blind deblurring using a coupled adaptive sparse prior. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 1051–1058. IEEE, 2013.

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

[42] X. Zhu, F. Šroubek, and P. Milanfar. Deconvolving psfs for a better motion deblurring using multiple images. In Proceedings of the European Conference on Computer Vision (ECCV), pages 636–647. Springer, 2012.

[43] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In Proceedings of the IEEE Conference on Computer Vision (ICCV), pages 479–486. IEEE, 2011.

[44] D. Zoran and Y. Weiss. Natural images, gaussian mixtures and dead leaves. In Advances in Neural Information Processing Systems (NIPS), pages 1736–1744, 2012.