

Discriminative Learning of Iteration-wise Priors for Blind Deconvolution

Wangmeng Zuo¹, Dongwei Ren¹, Shuhang Gu², Liang Lin³, and Lei Zhang²

¹School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China

²Dept. of Computing, The Hong Kong Polytechnic University, Hong Kong, China

³Sun Yat-Sen University, Guangzhou, China

{cswmzuo, rendongweihit}@gmail.com, {cssgu, cslzhang}@comp.polyu.edu.hk, linliang@ieee.org

Abstract

The maximum a posterior (MAP)-based blind deconvolution framework generally involves two stages: blur kernel estimation and non-blind restoration. For blur kernel estimation, sharp edge prediction and carefully designed image priors are vital to the success of MAP. In this paper, we propose a blind deconvolution framework together with iteration specific priors for better blur kernel estimation. The family of hyper-Laplacian ($\Pr(\mathbf{d}) \propto e^{-\|\mathbf{d}\|_p^q/\lambda}$) is adopted for modeling iteration-wise priors of image gradients, where each iteration has its own model parameters $\{\lambda^{(t)}, p^{(t)}\}$. To avoid heavy parameter tuning, all iteration-wise model parameters can be learned using our principled discriminative learning model from a training set, and can be directly applied to other dataset and real blurry images. Interestingly, with the generalized shrinkage / thresholding operator, negative p value ($p < 0$) is allowable and we find that it contributes more in estimating the coarse shape of blur kernel. Experimental results on synthetic and real world images demonstrate that our method achieves better deblurring results than the existing gradient prior-based methods. Compared with the state-of-the-art patch prior-based method, our method is competitive in restoration results but is much more efficient.

1. Introduction

Blind deconvolution is a severely ill-posed problem, in which both latent image \mathbf{x} and blur kernel \mathbf{k} are required to be estimated from an input blurred image \mathbf{y} . One class of popular solutions to blind deconvolution is the maximum a posterior (MAP) approaches [3, 22, 24], in which the pair (\mathbf{x}, \mathbf{k}) can be jointly estimated by maximizing a posterior probability,

$$\min_{\mathbf{x}, \mathbf{k}} \frac{\lambda}{2\sigma_n^2} \|\mathbf{k} \otimes \mathbf{x} - \mathbf{y}\|^2 + \phi(\mathbf{x}) + \mu\varphi(\mathbf{k}), \quad (1)$$

where σ_n is the standard deviation (*std.*) of additive white Gaussian noise, $\phi(\mathbf{x})$ and $\varphi(\mathbf{k})$ are some regularizers for \mathbf{x} and \mathbf{k} , respectively, λ and μ are the trade-off parameters. Several regularizers on \mathbf{k} had been proposed to enforce the sparsity and spatial smoothness of the estimated kernel. As to \mathbf{x} , one natural choice is based on image prior or natural image statistics, e.g., hyper-Laplacian of gradients [14].

However, naive MAP prefers the trivial delta kernel solution [16]. To make MAP work, salient structure selection [4, 11, 32] and carefully designed image priors [15, 34] should be adopted to enhance strong edges while suppressing harmful small-scale textures. So far, a number of edge prediction methods and image priors had been proposed, and the recent MAP approaches can achieve high quality results for blind deconvolution and even spatially variant blind deblurring [8, 34]. Edge prediction based approaches usually involve some heuristic and engineered methods, e.g., shock and bilateral filters in [4] and relative total variation (RTV) [10], to restore salient edges explicitly. Image prior-based approaches generally deploy novel regularizers, e.g., l_1/l_2 [15] and generalized l_0 [34], for implicit edge selection. These edge based methods may fail to handle small scale textures, in which, on the contrary, power-spectrum based methods can work well [35, 9, 18, 7, 36].

For better estimation of \mathbf{k} , researchers had noted that parameters for edge prediction and image regularization should be dynamically tuned during the iterations. For edge prediction-based approaches, strong edges are selected for coarse kernel estimation in the first a few iterations, and subsequently more details are added to further refine the estimated kernel [15, 27]. Under the MAP framework [3, 15, 27, 23], the regularization parameter λ is set small to preserve strong edges while suppressing detailed textures in the first a few iterations, and gradually tuned along with the iteration to produce accurate kernel. In summary, most MAP approaches involve carefully designed regularizers and handcrafted parameter tuning to guide the algorithms

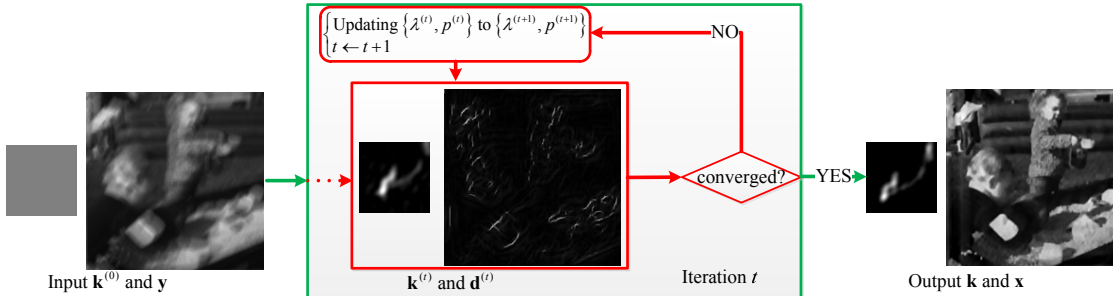


Figure 1: Illustration of iteration-wise priors for blind deconvolution.

converge to desired solution.

In this paper, we present a principled formulation on MAP-based deconvolution, where we adopt the hyper-Laplacian prior $\Pr(\mathbf{d}) \propto e^{-\|\mathbf{d}\|_p^p/\lambda}$ on gradients \mathbf{d} but allow the parameters $\{\lambda^{(t)}, p^{(t)}\}$ to be iteration-wisely tuned. As illustrated in Fig. 1, our algorithm iteratively performs three steps: (i) updating the latent image gradient $\mathbf{d}^{(t)}$ via variable splitting and generalized shrinkage / thresholding (GST) operator [40], (ii) updating the kernel $\mathbf{k}^{(t)}$ via one-step augmented Lagrangian method, and (iii) updating the parameters $\{\lambda^{(t)}, p^{(t)}\}$ to $\{\lambda^{(t+1)}, p^{(t+1)}\}$. Moreover, we extend GST for solving l_p -norm minimization problem with $p < 0$, and find that it is effective in enhancing strong edges while suppressing weak textures.

Instead of handcrafted tuning of parameters, a discriminative learning method is developed for estimating iteration-wise parameters $\{\lambda^{(t)}, p^{(t)}\}$ from a training set. We adopt the weighted mean squared error (MSE) on the estimated \mathbf{k} and \mathbf{d} as the loss function $L(\lambda, p)$, and propose a gradient descent algorithm for determining $\{\lambda^{(t)}, p^{(t)}\}$ in each iteration. Experimental results show that the parameters learned from the training dataset can be directly applied to other synthetic and real world images. The proposed method are more visually plausible than the existing gradient prior-based methods, including both MAP [4, 32, 15] and variational Bayes [17], and is comparable with the state-of-the-art patch-based method [29]. Our contribution is three-fold:

(1) A novel MAP-based blind convolution method is developed by incorporating with iteration-wise image priors, which provide a natural way to employ salient edges for robust estimation of coarse kernel, and gradually converge to desired solution.

(2) A principled discriminative learning approach is proposed to learn iteration-wise image priors from the training dataset, and experiments validate that the learned parameters can be directly applied to test images.

(3) The GST operator for l_p -norm minimization is extended to the case with $p < 0$, where GST can magnify the strong edges whereas suppressing detailed textures, making it very encouraging in estimating the coarse kernel in the

first a few iterations.

2. Related Work

Under the MAP framework, regularizers are critical to facilitate the estimate of blur kernel. In this section, we provide a brief survey on the representative regularizers on both latent image and blur kernel.

2.1. Regularization on gradients

Total variation (TV) is an interesting regularizer for blind deconvolution. Chan and Wong [3] proposed an alternating minimization algorithm that works well on TV-based blind deconvolution (TVBD), following which several modifications were developed [27, 28, 33]. However, theoretical analysis by Levin et al. [16] showed that, instead of the desired kernel, TVBD would converge to the trivial delta kernel [22, 16, 27], and variational Bayes (VB) was then suggested to avoid the local minima traps of MAP, [31, 2, 6, 20, 1]. To explain the incongruence between the success of TV in [3] and the failure of TV in [16], Perrone and Favaro [22] revealed that Chan and Wong actually adopted a projected alternating minimization algorithm for TVBD, and the success of the algorithm can be explained by the delayed normalization of blur kernel.

Other gradient-based regularizers had been proposed to enhance strong edges for better kernel estimation. In [12], Paragios et al. proposed a discrete MRF prior to constrain the latent image only with main structure, which essentially produced highly sparse gradients. In [34], Xu et al. proposed a generalized l_0 -norm regularizer, in which the intermediate recovered gradient is much more sparse than natural image statistics. In [29], edge-based patch prior was introduced for robust restoration of strong edges. Besides gradient sparsity, the dictionary based sparsity has also been employed in blind deconvolution [37].

In this work, instead of exploiting one specific gradient-based regularizer, we suggest that the gradient priors and regularization parameters are set iteration-wisely, and propose a principled method to learn the parameters from the training set.

2.2. Regularization on blur kernel

One conventional regularizer on blur kernel \mathbf{k} is the non-negative constraint $k_i \geq 0, \forall i$ and the equality constraint $\sum_i k_i = 1$. To enforce sparsity on \mathbf{k} , in the implementations [4, 15, 29, 21] the hard-thresholding operator was adopted to make $k_i \geq \varepsilon, \forall i$, where ε is some small positive value. Other sparsity priors, e.g., TV [3] and hyper-Laplacian [14, 13], were also suggested to avoid the non-blur solution and converge to desired solution [24].

3. Blind deconvolution with iteration-wise priors

In this section, we incorporate the iteration-wise image priors with the sparsity regularizer on blur kernel, and formulate our MAP-based model in gradient space. An alternating minimization algorithm is then proposed for blind deconvolution. Moreover, we carefully design the updating rules on blur kernel \mathbf{k} and latent gradient image \mathbf{d} , making it possible to learn iteration-wise priors from training data.

3.1. Problem Formulation

As suggested in [4, 32, 15, 21], we formulate the proposed MAP-based blur kernel estimation model in the gradient space,

$$\min_{\mathbf{d}, \mathbf{k}} \frac{\lambda}{2\sigma_n^2} \|\mathbf{k} \otimes \mathbf{d} - \nabla \mathbf{y}\|^2 + \phi(\mathbf{d}) + \mu \varphi(\mathbf{k}), \quad (2)$$

where $\mathbf{d} = \nabla \mathbf{x}$ ($\mathbf{d}_h = \nabla_h \mathbf{x}$ and $\mathbf{d}_v = \nabla_v \mathbf{x}$ with gradient operator $\nabla = \{\nabla_h, \nabla_v\}$) denotes the gradient images along both horizontal and vertical directions. In this work, we impose the hyper-Laplacian priors on both \mathbf{k} and \mathbf{d} ,

$$\Pr(\mathbf{k}) \propto e^{-\mu \|\mathbf{k}\|_{0.5}^{0.5}}, \quad (3)$$

$$\Pr(\mathbf{d}) \propto e^{-\|\mathbf{d}\|_p^p / \lambda}, \quad (4)$$

but allow the parameters λ, p to be iteration-wise, i.e., $\lambda = \lambda^{(t)}$ and $p = p^{(t)}$ in the t -th iteration. Moreover, the image \mathbf{d} should be the gradient of some image \mathbf{x} and thus satisfies the constraint $\mathbf{d} = \nabla \mathbf{x}$, which was generally omitted in blind deconvolution. Actually, the constraint $\mathbf{d} = \nabla \mathbf{x}$ can be explicitly expressed as [19],

$$\nabla_h \mathbf{d}_v = \nabla_v \mathbf{d}_h. \quad (5)$$

Furthermore, we also adopt the non-negative constraint and the equality constraint on \mathbf{k}

$$\sum_i k_i = 1, k_i \geq 0, \forall i, \quad (6)$$

By incorporating the constraints in Eqns. (5) and (6) and the priors in Eqns. (4) and (3) into Eq. (2), the proposed MAP-based model is then formulated as

$$\min_{\mathbf{d}, \mathbf{k}} \frac{\lambda^{(t)}}{2\sigma_n^2} \|\mathbf{k} \otimes \mathbf{d} - \nabla \mathbf{y}\|^2 + \|\mathbf{d}\|_{p^{(t)}}^{p^{(t)}} + \mu \|\mathbf{k}\|_{0.5}^{0.5} \quad (7)$$

$$\text{s.t. } \nabla_h \mathbf{d}_v = \nabla_v \mathbf{d}_h, \sum_i k_i = 1, k_i \geq 0, \forall i.$$

where $\lambda^{(t)}$ and $p^{(t)}$ are iteration-wise parameters to be learned.

3.2. Alternating minimization

To solve the model in Eq. (7), we employ the alternating minimization algorithm by iteratively performing the following two steps: (i) updating \mathbf{d} given \mathbf{k} , and (ii) updating \mathbf{k} given \mathbf{d} .

3.2.1 Updating \mathbf{d}

Given the estimate of blur kernel $\mathbf{k}^{(t-1)}$, the subproblem on \mathbf{d} can be formulated as

$$\min_{\mathbf{d}} \frac{\lambda^{(t)}}{2\sigma_n^2} \|\mathbf{k}^{(t-1)} \otimes \mathbf{d} - \nabla \mathbf{y}\|^2 + \|\mathbf{d}\|_{p^{(t)}}^{p^{(t)}} \text{ s.t. } \nabla_h \mathbf{d}_v = \nabla_v \mathbf{d}_h, \quad (8)$$

This model is non-convex and non-smooth, and we employ the half-quadratic strategy to solve the model above, in which an auxiliary variable \mathbf{w} is introduced, making the \mathbf{d} -step decomposed into two subproblems,

$$\begin{cases} \mathbf{w}^{(t)} = \arg \min_{\mathbf{w}} \frac{\beta^{(t)}}{2} \|\mathbf{w} - \mathbf{d}^{(t-1)}\|^2 + \|\mathbf{w}\|_{p^{(t)}}^{p^{(t)}}, \\ \mathbf{d}^{(t)} = \arg \min_{\mathbf{d}} \frac{\lambda^{(t)}}{2\sigma_n^2} \|\mathbf{k}^{(t-1)} \otimes \mathbf{d} - \nabla \mathbf{y}\|^2 + \frac{\beta^{(t)}}{2} \|\mathbf{w}^{(t)} - \mathbf{d}\|^2 \\ \text{s.t. } \nabla_c^T \mathbf{d} = 0, \end{cases} \quad (9)$$

where $\nabla_c = [\nabla_v, -\nabla_h]$. When the penalty parameter β approaches infinity, the alternating optimization of \mathbf{d} and \mathbf{w} is exactly equivalent to optimize \mathbf{d} -step in Eq. (8). The \mathbf{w} -subproblem and \mathbf{d} -subproblem usually require to be solved alternatively with several iterations, but this will make the relationship between $\mathbf{d}^{(t)}$ with $(\lambda^{(t)}, p^{(t)})$ hard to analyze. Therefore, we set the number of inner-iteration be 1.

In Eq. (9), the subproblem on \mathbf{w} can be efficiently solved using the GST operator with one-iteration [40],

$$w_i^{(t)} = \begin{cases} 0, & \text{if } d_i^{(t)} \leq \tau_i^{(t)}, \\ \text{sgn}(d_i^{(t)}) \left(|d_i^{(t)}| - \frac{1}{\beta^{(t)}} p^{(t)} \left(|d_i^{(t)}| \right)^{p^{(t)}-1} \right), & \text{else,} \end{cases} \quad (10)$$

where the threshold $\tau_i^{(t)}$ is determined by

$$\tau_i^{(t)} = \left(\frac{2}{\beta^{(t)}} (1-p^{(t)}) \right)^{\frac{1}{2-p^{(t)}}} + \frac{1}{\beta^{(t)}} p^{(t)} \left(\frac{2}{\beta^{(t)}} (1-p^{(t)}) \right)^{\frac{p^{(t)}-1}{2-p^{(t)}}}. \quad (11)$$

The \mathbf{d} -subproblem is convex, whose closed-form solution can be obtained by adopting hybrid alternating direction method of multipliers,

$$\mathbf{d}^{(t)} = \mathbf{\Omega}^{-1} (\boldsymbol{\eta} + \beta^{(t)} \mathbf{w}^{(t)}), \quad (12)$$

where $\mathbf{\Omega} = \mathbf{A}_k^T \mathbf{A}_k + \beta^{(t)} \mathbf{I}$, $\boldsymbol{\eta} = \frac{\lambda^{(t)}}{\sigma_n^2} \mathbf{A}_k^T \nabla \mathbf{y} - \nabla_c \boldsymbol{\nu}^{(t)}$, and \mathbf{A}_k , with periodic boundary condition of blurred image, is the block circulant with circulant blocks (BCCB) matrix of corresponding PSF \mathbf{k} , and it can be diagonalized via fast Fourier transform (FFT). $\boldsymbol{\nu}^{(t)}$ is the Lagrangian vector,

$$\boldsymbol{\nu}^{(t)} = (\nabla_c^T \mathbf{\Omega}^{-1} \nabla_c)^{-1} \nabla_c^T \mathbf{\Omega}^{-1} (\mathbf{A}_k^T \nabla \mathbf{y} + \beta^{(t)} \mathbf{w}^{(t+1)}). \quad (13)$$

3.2.2 Updating \mathbf{k}

Given the estimate of latent gradient $\mathbf{d}^{(t)}$, the subproblem on \mathbf{k} can be formulated as

$$\min_{\mathbf{k}} \frac{\lambda^{(t)}}{2\sigma_n^2} \left\| \mathbf{k} \otimes \mathbf{d}^{(t)} - \nabla \mathbf{y} \right\|^2 + \mu \|\mathbf{k}\|_{0.5}^{0.5} \text{ s.t. } \sum_i k_i = 1, k_i \geq 0, \forall i, \quad (14)$$

We further introduce two auxiliary variables $\mathbf{h} = \mathbf{k}$ and $\mathbf{g} = \mathbf{k}$, and reformulate the model in Eq. (14) as

$$\min_{\mathbf{k}, \mathbf{h}, \mathbf{g}} \frac{\lambda^{(t)}}{2\sigma_n^2} \left\| \mathbf{k} \otimes \mathbf{d}^{(t)} - \nabla \mathbf{y} \right\|^2 + B(\mathbf{h}) + \frac{\delta_1^{(t)}}{2} (\mathbf{1}^T \mathbf{k} - 1)^2 + \mu^{(t)} \|\mathbf{g}\|_{0.5}^{0.5} \quad (15)$$

s.t. $\mathbf{k} = \mathbf{h}, \mathbf{k} = \mathbf{g}$,

where $\mathbf{1}$ is a vector with all entries being 1, and $B(\mathbf{h})$ is defined as

$$B(h_i) = \begin{cases} +\infty, & \text{if } h_i < 0, \\ 0, & \text{else,} \end{cases} \quad (16)$$

The problem in Eq. (15) can be solved using the augmented Lagrangian method by iteratively solving the following subproblems

$$\begin{cases} \mathbf{h}^{(t)} = \min_{\mathbf{h}} \frac{\delta_2^{(t)}}{2} \left\| \mathbf{k}^{(t-1)} - \mathbf{h} \right\|^2 + B(\mathbf{h}), \\ \mathbf{g}^{(t)} = \min_{\mathbf{g}} \frac{\delta_2^{(t)}}{2} \left\| \mathbf{k}^{(t-1)} - \mathbf{g} \right\|^2 + \mu^{(t)} \|\mathbf{g}\|_{0.5}^{0.5}, \\ \mathbf{k}^{(t)} = \min_{\mathbf{k}} \frac{\lambda^{(t)}}{2\sigma_n^2} \left\| \mathbf{k} \otimes \mathbf{d}^{(t)} - \nabla \mathbf{y} \right\|^2 + \frac{\delta_3^{(t)}}{2} (\mathbf{1}^T \mathbf{k} - 1)^2 \\ \quad + \frac{\delta_2^{(t)}}{2} \left\| \mathbf{k} - \mathbf{h}^{(t)} \right\|^2 + \frac{\delta_1^{(t)}}{2} \left\| \mathbf{k} - \mathbf{g}^{(t)} \right\|^2. \end{cases} \quad (17)$$

Of course we can repeat the updating of $\mathbf{h}^{(t)}$, $\mathbf{g}^{(t)}$, $\mathbf{k}^{(t)}$ for a number of iterations. But, to explicitly analyze the relationship between $\mathbf{k}^{(t)}$ and $\lambda^{(t)}$, we only update $\mathbf{h}^{(t)}$, $\mathbf{g}^{(t)}$, $\mathbf{k}^{(t)}$ one time. The solution to the \mathbf{h} -subproblem is simply a projection,

$$h_i^{(t)} = \begin{cases} k_i^{(t-1)}, & \text{if } k_i^{(t-1)} > 0, \\ 0, & \text{else.} \end{cases} \quad (18)$$

The \mathbf{g} -subproblem can be simply solved by the GST algorithm [40]. The \mathbf{k} -subproblem is a quadratic optimization problem with the closed-form solution as

$$\mathbf{k}^{(t)} = \Phi^{-1} \zeta, \quad (19)$$

where $\zeta = \frac{\lambda^{(t)}}{\sigma_n^2} \mathbf{A}_d \nabla \mathbf{y} + \delta_1^{(t)} \mathbf{g}^{(t)} + \delta_2^{(t)} \mathbf{h}^{(t)} + \delta_3^{(t)} \mathbf{1}$ and $\Phi = \frac{\lambda^{(t)}}{\sigma_n^2} \mathbf{A}_d^T \mathbf{A}_d + \delta_1^{(t)} \mathbf{I} + \delta_2^{(t)} \mathbf{I} + \delta_3^{(t)} \mathbf{11}^T$. $\mathbf{11}^T$ is a matrix whose entries are all 1, and clearly, it is also a BCCB matrix that can be efficiently diagonalized via FFT.

Finally, Algorithm 1 summarizes the main steps of our alternating minimization algorithm. Given an $n \times n$ image, both the updating of \mathbf{d} and \mathbf{k} have the complexity of $O(n^2 \log n)$, which make our algorithm very efficient for blind deconvolution.

3.3. Extension of GST to $p < 0$

In [40], the GST operator was proposed to solve the following ℓ_p -minimization problem ($0 \leq p \leq 1$)

$$\hat{x} = \min_x \frac{1}{2} (y - x)^2 + \lambda |x|^p. \quad (20)$$

Algorithm 1 Blind deconvolution

Input: Blurry image \mathbf{y} , and $\{\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(t)}, \dots, \boldsymbol{\theta}^{(MaxIter)}\}$

Output: Blur kernel \mathbf{k} and latent image \mathbf{x}

- 1: Initializing $\mathbf{d}^{(0)}, \mathbf{k}^{(0)}, \beta^{(1)}, \delta_1^{(1)}, \delta_2^{(1)}, \delta_3^{(1)}$
 - 2: **for** $t = 1$ to $MaxIter$ **do**
 - 3: //Lines 4-5 solve \mathbf{d} -step Eq. (9)
 - 4: $\mathbf{w}^{(t)} = \text{GST}(\mathbf{d}^{(t-1)}, p^{(t)}, 1/\beta^{(t)})$
 - 5: $\mathbf{d}^{(t)} = \Omega^{-1}(\boldsymbol{\eta} + \beta^{(t)} \mathbf{w}^{(t)})$
 - 6: //Lines 7-9 solve \mathbf{k} -step Eq. (17)
 - 7: $\mathbf{h}^{(t)} = \arg \min_{\mathbf{h}} \frac{\delta_2^{(t)}}{2} \left\| \mathbf{k}^{(t-1)} - \mathbf{h} \right\|^2 + B(\mathbf{h})$
 - 8: $\mathbf{g}^{(t)} = \text{GST}(\mathbf{k}^{(t-1)}, 0.5, \mu^{(t)}/\delta_1^{(t)})$
 - 9: $\mathbf{k}^{(t)} = \Phi^{-1} \zeta$
 - 10: **if** $\|\mathbf{d}^{(t)} - \mathbf{d}^{(t-1)}\| / \|\mathbf{d}^{(t)}\| \leq \varepsilon_0$ & $\|\mathbf{k}^{(t)} - \mathbf{k}^{(t-1)}\| / \|\mathbf{k}^{(t)}\| \leq \varepsilon_0$
 - 11: **break**
 - 12: **endif**
 - 13: Updating $\beta^{(t+1)}, \delta_1^{(t+1)}, \delta_2^{(t+1)}, \delta_3^{(t+1)}$
 - 14: **end for**
 - 15: $\mathbf{k} = \mathbf{k}^{(t)}$
 - 16: Given \mathbf{k} , recovering \mathbf{x} using non-blind deconvolution methods
-

Let

$$\tau_p^{GST}(\lambda) = (2\lambda(1-p))^{\frac{1}{2-p}} + \lambda p (2\lambda(1-p))^{\frac{p-1}{2-p}}, \quad (21)$$

when $|y| \leq \tau_p^{GST}(\lambda)$, we have $\hat{x} = 0$, and when $|y| > \tau_p^{GST}(\lambda)$, \hat{x} can be obtained by repeating the following step:

$$\hat{x} = \text{sgn}(y) \left(|y| - \lambda p (|x|)^{p-1} \right) \quad (22)$$

Although GST was proposed for $0 \leq p \leq 1$, from Eqns. (21) and (22), one can easily see that the threshold rule and the shrinkage rule can be directly extended to the case $p < 0$. It is interesting to point out that, when $p < 0$, Eq. (22) actually is an expansion rule. Thus, GST with $p < 0$ can be adopted to suppress detailed textures while enhancing strong edges, which in spirit is similar with the bilateral and shock filters but is computationally more efficient. Considering this fact, $p < 0$ is allowed in our blind deconvolution method.

4. Discriminative learning of iteration-wise priors

Given the maximum number of iterations $MaxIter$ in Algorithm 1, we should set $2 \times MaxIter$ image prior parameters $\{(\lambda^{(t)}, p^{(t)})\}_{t=1}^{MaxIter}$. Hand-crafted tuning of these parameters is too challenging and impractical. Therefore, we adopt a discriminative learning method to learn the parameters directly from a training set.

Denote \mathcal{D} by a set of synthetic images $\{(\mathbf{d}_i^{gt}, \mathbf{k}_i^{gt}, \nabla \mathbf{y}_i)\}_{i=1}^N$, where \mathbf{d}_i^{gt} denotes the gradient of the i -th clear image, \mathbf{k}_i^{gt} denotes the i -th blur kernel, and $\nabla \mathbf{y}_i$ denotes the gradient of the i -th blurry image. For estimating $\boldsymbol{\theta}^{(t)} = \{\lambda^{(t)}, p^{(t)}\}$, we adopt the following loss

function defined on \mathcal{D} ,

$$L^{(t)}(\boldsymbol{\theta}) = \sum_{i=1}^N L_i^{(t)}(\boldsymbol{\theta}) = \sum_{i=1}^N \alpha^{(t)} L_{\mathbf{d}_i}^{(t)}(\boldsymbol{\theta}) + L_{\mathbf{k}_i}^{(t)}(\boldsymbol{\theta}) \\ = \sum_{i=1}^N \frac{\alpha^{(t)}}{2} \left(\|\mathbf{d}_i^{(t)} - \mathbf{d}_i^{gt}\|^2 / |\mathbf{d}_i^{gt}| + \frac{1}{2} \|\mathbf{k}_i^{(t)} - \mathbf{k}_i^{gt}\|^2 / |\mathbf{k}_i^{gt}| \right), \quad (23)$$

where $|\bullet|$ counts the entries of the vector for the normalization of image and kernel sizes, and α denotes the trade-off parameter. Note that our aim is to estimate the desired blur kernel. Therefore, we set α to be small at first a few iterations, which thus can make the learned prior contribute to rough kernel estimation. Then the larger α value will be set to learn the parameters $\boldsymbol{\theta}^{(t)}$ for better estimation of the clear latent image and refinement of the blur kernel.

4.1. Learning algorithm

To learn the parameters, we adopt the gradient descent algorithm. From Eqns. (10), (12), (19), and (23), we have the following observations:

- (1) $L_i^{(t)}$ is a function of $\mathbf{d}_i^{(t)}$ and $\mathbf{k}_i^{(t)}$;
- (2) $\mathbf{d}_i^{(t)}$ is a function of $\mathbf{w}_i^{(t)}$ and $\lambda^{(t)}$;
- (3) $\mathbf{w}_i^{(t)}$ is a function of $p^{(t)}$;
- (4) $\mathbf{k}_i^{(t)}$ is a function of $\lambda^{(t)}$.

By combining these observations, the gradient of loss function $L_i^{(t)}(\boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$ can be written as,

$$\frac{\partial L_i^{(t)}}{\partial \boldsymbol{\theta}} = \left(\alpha^{(t)} \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial p}, \alpha^{(t)} \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial \lambda} + \frac{\partial L_{\mathbf{k}_i}^{(t)}}{\partial \lambda} \right). \quad (24)$$

Based on Eq. (12), the derivatives of $L_{\mathbf{d}_i}^{(t)}$ w.r.t. λ and p can be written as

$$\frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial \lambda} = \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial \mathbf{d}_i^{(t)}} \frac{\partial \mathbf{d}_i^{(t)}}{\partial \lambda} \\ = 1/(\sigma_n^2 |\mathbf{d}_i^{gt}|) (\mathbf{d}_i^{(t)} - \mathbf{d}_i^{gt})^T \boldsymbol{\Omega}^{-1} (\mathbf{A}_k^T \nabla y_i - \mathbf{A}_k^T \mathbf{A}_k). \quad (25)$$

$$\frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial p} = \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial \mathbf{d}_i^{(t)}} \frac{\partial \mathbf{d}_i^{(t)}}{\partial \mathbf{w}_i^{(t)}} \frac{\partial \mathbf{w}_i^{(t)}}{\partial p} \\ = (\beta^{(t)} / |\mathbf{d}_i^{gt}|) (\mathbf{d}_i^{(t)} - \mathbf{d}_i^{gt})^T \boldsymbol{\Omega}^{-1} \frac{\partial \mathbf{w}_i^{(t)}}{\partial p}. \quad (26)$$

Based on Eq. (10), the derivative of $\mathbf{w}_i^{(t)}$ w.r.t. p can be obtained by,

$$\frac{\partial \mathbf{w}_i^{(t)}}{\partial p} = \begin{cases} 0, & \text{if } d_i^{(t)} \leq \tau_i^{(t)}, \\ -\frac{\text{sgn}(d_i^{(t)})}{\beta^{(t)}} \left(\left(|d_i^{(t)}| \right)^{p-1} + p \left(|d_i^{(t)}| \right)^{p-1} \ln \left(|d_i^{(t)}| \right) \right), & \text{else,} \end{cases} \quad (27)$$

where the threshold $\tau_i^{(t)}$ in Eq. (11) is determined by p value, and we approximate it based on $p^{(t-1)}$.

Based on Eq. (19), the derivative of $L_{\mathbf{k}_i}^{(t)}$ w.r.t. λ can be written as

$$\frac{\partial L_{\mathbf{k}_i}^{(t)}}{\partial \lambda} = \frac{\partial L_{\mathbf{k}_i}^{(t)}}{\partial \mathbf{k}_i^{(t)}} \frac{\partial \mathbf{k}_i^{(t)}}{\partial \lambda} \\ = 1/(\sigma_n^2 |\mathbf{k}_i^{gt}|) (\mathbf{k}_i^{(t)} - \mathbf{k}_i^{gt})^T \boldsymbol{\Phi}^{-1} (\mathbf{A}_d^T \nabla y_i - \mathbf{A}_d^T \mathbf{A}_d). \quad (28)$$

Given the derivative of $L^{(t)}(\boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$, we use simple gradient descent method to search the optimal $\boldsymbol{\theta}^{(t)}$. The procedure of learning priors is summarized in Algorithm 2.

Algorithm 2 Learning iteration-wise priors

Input: Training set \mathcal{D} , $MaxIter$

Output: $\{\boldsymbol{\theta}^{(1)}, \dots, \boldsymbol{\theta}^{(t)}, \dots, \boldsymbol{\theta}^{(MaxIter)}\}$

- 1: Initializing $\alpha^{(1)}, \mu^{(1)}, \beta^{(1)}, \delta_1^{(1)}, \delta_2^{(1)}, \delta_3^{(1)}$
 - 2: **for** $t = 1$ to $MaxIter$ **do**
 - 3: grad = 0
 - 4: **for** $i = 1$ to N **do**
 - 5: Updating $\mathbf{d}_i^{(t)}$ Eq. (9) and $\mathbf{k}_i^{(t)}$ Eq. (17)
 - 6: grad = grad + $\partial L_i^{(t)}(\boldsymbol{\theta}) / \partial \boldsymbol{\theta}$
 - 7: **end for**
 - 8: Using gradient descent method to search optimal $\boldsymbol{\theta}^{(t)}$
 - 9: Updating $\alpha^{(t+1)}, \mu^{(t+1)}$ and penalty parameters $\beta^{(t+1)}, \delta_1^{(t+1)}, \delta_2^{(t+1)}, \delta_3^{(t+1)}$
 - 10: **end for**
-

4.2. Discussion

Our MAP-based blind deconvolution method in Section 3 is specially designed for learning iteration-wise priors. Unlike the conventional alternating minimization and ALM algorithms, we only perform one time for the inner-loop iterations to solve the problems in Eqns. (9) and (17), and adopt the one-step GST operator and the approximation of the threshold $\tau_i^{(t)}$. As shown in Section 4, all these make it feasible to compute the derivative of $L_i^{(t)}(\boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$, and allow us to learn iteration-wise priors from training data \mathcal{D} .

Our work is conceptually similar to the models in [25, 26] which discriminatively learn a cascade of image priors [25, 5, 26]. Our model is distinctly different with them from two aspects. First our model is designed for blind deconvolution rather than non-blind deconvolution. Second, instead of learning shrinkage function for filter responses, we learn the iteration-wise parametric priors for image gradients.

Besides, some extra constraints are considered to improve the algorithm robustness and stability. In [3, 15, 27, 23], the regularization parameter λ begins with small value and gradually increases along with the iteration. Thus the search range of λ is constrained in [0.5, 5], and we further require the sequence $\{\lambda^{(t)}\}$ should be non-decreasing, i.e., $\lambda^{(t+1)} \geq \lambda^{(t)}$. As to the p value, Xu et al. [34] first adopt $p = 0$ for estimating blur kernel, and then set $p = 0.5$ for final restoration. Therefore, we also incorporate the non-decreasing constraint on the the sequence $\{p^{(t)}\}$, i.e., $p^{(t+1)} \geq p^{(t)}$, and its search range is constrained in $[-1, 0.2]$.

4.3. Implementations

In our implementation, we set $MaxIter = 200$. The weighting parameter α is initialized as 1×10^{-3} , and then increases along with the iteration by $\alpha^{(t+1)} =$

$\min(1.1 \times \alpha^{(t)}, 1)$. The regularization weight μ on blur kernel \mathbf{k} is initialized as 1×10^{-6} , and updated by $\mu^{(t+1)} = \min(1.1 \times \mu^{(t)}, 1 \times 10^{-3})$. The penalty parameter β is initialized as 1×10^{-4} , and updated by $\beta^{(t+1)} = \min(1.5 \times \beta^{(t)}, 1 \times 10^{-2})$. The three δ s are initialized as 1×10^{-3} , and updated by $\delta^{(t+1)} = \min(1.1 \times \delta^{(t)}, 1 \times 10^{-1})$.

5. Experimental Results

Two synthetic datasets and real blurry photographs are used to evaluate the proposed method. We first learn the iteration-wise priors using Levin et al.’s dataset [16] which contains 8 kernels and 4 images with the size of 255×255 . Then, Sun et al.’s dataset [29] is applied to evaluate the generalization of the learned iteration-wise priors. Finally, we apply our method to real blurry photographs and compare with the state-of-the-art gradient prior based and patch prior based methods.

We assume the *std.* σ_n of the additive Gaussian white noise should be in the range of $[1 \times 10^{-3}, 1 \times 10^{-2}]$, and estimate it using [38]. To evaluate the restoration quality, we use three quantitative metrics, i.e., PSNR, SSIM [30] and error ratio [16]. We also evaluate the efficiency of each competing method, and all the programs are ran on processors with Intel(R) Xeon(R) CPU 3.30GHz.

5.1. Training on Levin et al.’s dataset

Fig. 2 shows the plot of the learned iteration-wise p and λ values. One can see that, smaller p (≈ -1) and λ (≈ 0.5) values benefit the rough estimation of blur kernel at the first a few iterations, and larger p (≈ 0.2) and λ (≈ 5) values are required at the last a few iterations, which are consistent with the handcrafted parameter setting [15, 27, 33].

Table 1 lists the three quantitative metrics and running time of the competing methods and Fig. 4 shows the success ratio of error ratios. To be fair, the method in [17] is adopted for final non-blind deconvolution after the kernels were estimated. One can see that, our method is comparable to the state-of-the-art patch prior based method by Sun et al. [29], and significantly outperforms all the gradient

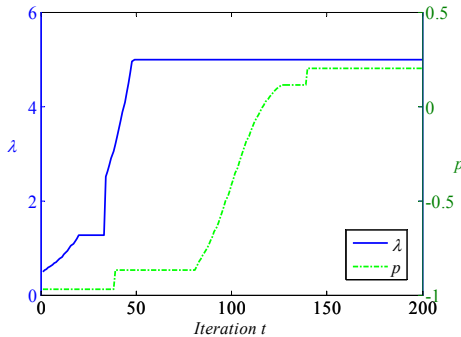


Figure 2: Learned iteration-wise p and λ for each iteration.

Table 1: Comparisons on Levin et al.’s dataset [16] using mean PSNR, mean SSIM, mean error ratio and mean running time (seconds)

	PSNR	SSIM	Error Ratio	Time
Known \mathbf{k}	32.31	0.9385	1.0000	—
Krishnan et al. [15]	28.26	0.8547	2.3746	8.9400
Cho & Lee [4]	28.83	0.8801	1.5402	1.3951
Levin et al. [17]	28.79	0.8922	1.5592	78.263
Xu & Jia [32]	29.45	0.9000	1.4071	1.1840
Sun et al. [29]	30.85	0.9191	1.2244	191.03
Ours	30.33	0.9192	1.2537	25.184

prior based methods [17, 15, 4, 32] in terms of all the three quantitative metrics. Fig. 3 provides the deblurring results of one image. The estimated blur kernel by our method is more accurate, making our deblurring image more visually plausible. As to running time, the methods of Cho & Lee [4] and Xu & Jia [32] are significantly faster than the others [17, 15, 29], partially for their optimized coarse-to-fine implementations in C/C++. Compared with Sun et al. [29], our method is more than 6 times faster even they also adopted the coarse-to-fine scheme.

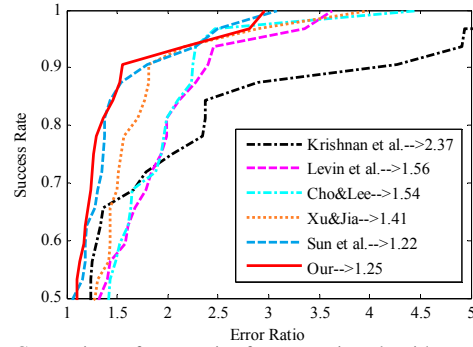


Figure 4: Comparison of error ratios for competing algorithms on Levin et al.’s dataset, and the values in the legend indicate the mean error ratio.

5.2. Evaluation on Sun et al.’s dataset

We further applied the learned iteration-wise priors to Sun et al.’s dataset [29] which consists of 640 synthetic blurry images generated with 80 images and 8 blur kernels. The EPLL method [39] is adopted to non-blind deconvolution for all the competing methods.

Table 2: Comparisons on Sun et al.’s dataset [29] using mean PSNR, mean SSIM, mean error ratio and mean running time (seconds)

	PSNR	SSIM	Error Ratio	Time
Known \mathbf{k}	32.35	0.9536	1.0000	—
Krishnan et al. [15]	22.76	0.8136	6.8351	159.29
Cho & Lee [4]	26.13	0.8624	5.0731	10.518
Levin et al. [17]	24.64	0.8606	4.5798	518.59
Xu & Jia [32]	28.11	0.9016	3.2843	6.2940
Sun et al. [29]	29.32	0.9200	2.4036	3911.1
Ours (-1)	27.96	0.9019	3.2188	311.77
Ours (0.2)	28.35	0.9111	2.9877	312.11
Ours	29.10	0.9220	2.4054	311.61



Figure 3: Example of deblurring results on Levin et al.'s dataset.

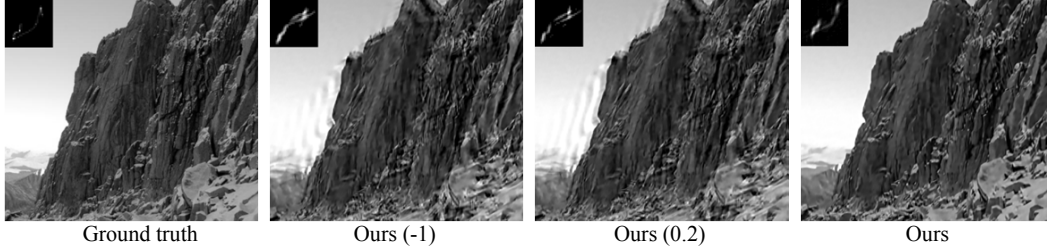


Figure 5: Comparison of fixed priors and iteration-wise priors.

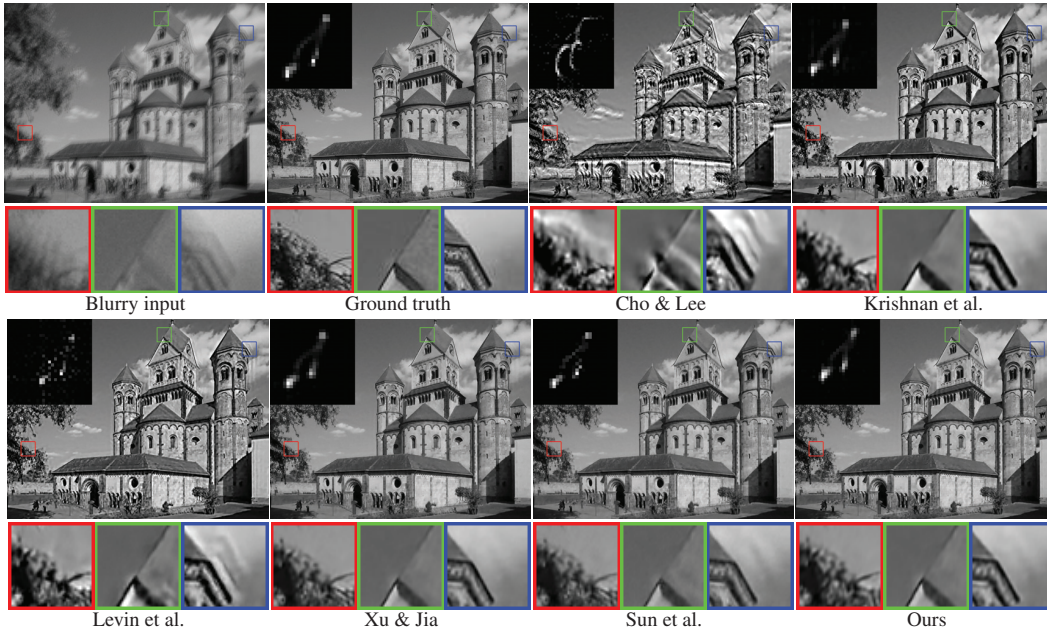


Figure 7: Example of deblurring results on Sun et al.'s dataset.

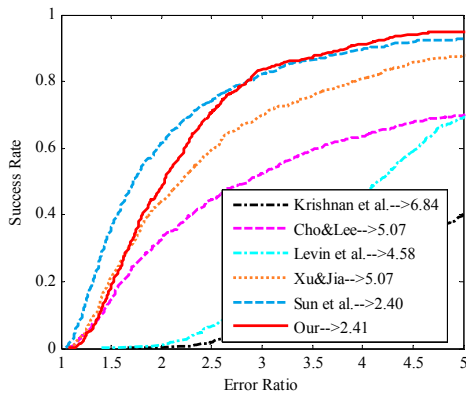


Figure 6: Comparison of error ratios for competing algorithms on Sun et al.'s dataset, and the values in the legend indicate the mean error ratio.

Table 2 lists the performance metrics of the competing methods. To validate the effectiveness of iteration-wise priors, we also consider two variants of our method by fixing the p value as -1 and 0.2 . From Table 2 and Fig. 5, one can see that our method with iteration-wise p values outperforms the two variants by all performance metrics, and can obtain more accurate estimation on blur kernel and the deblurring result is visually more pleasing.

In terms of quantitative metrics, our method significantly outperforms the other gradient prior based methods [17, 15, 4, 32], which indicate that the learned iteration-wise priors have good generalization ability and can be directly applied to other dataset. The mean SSIM value of our method is

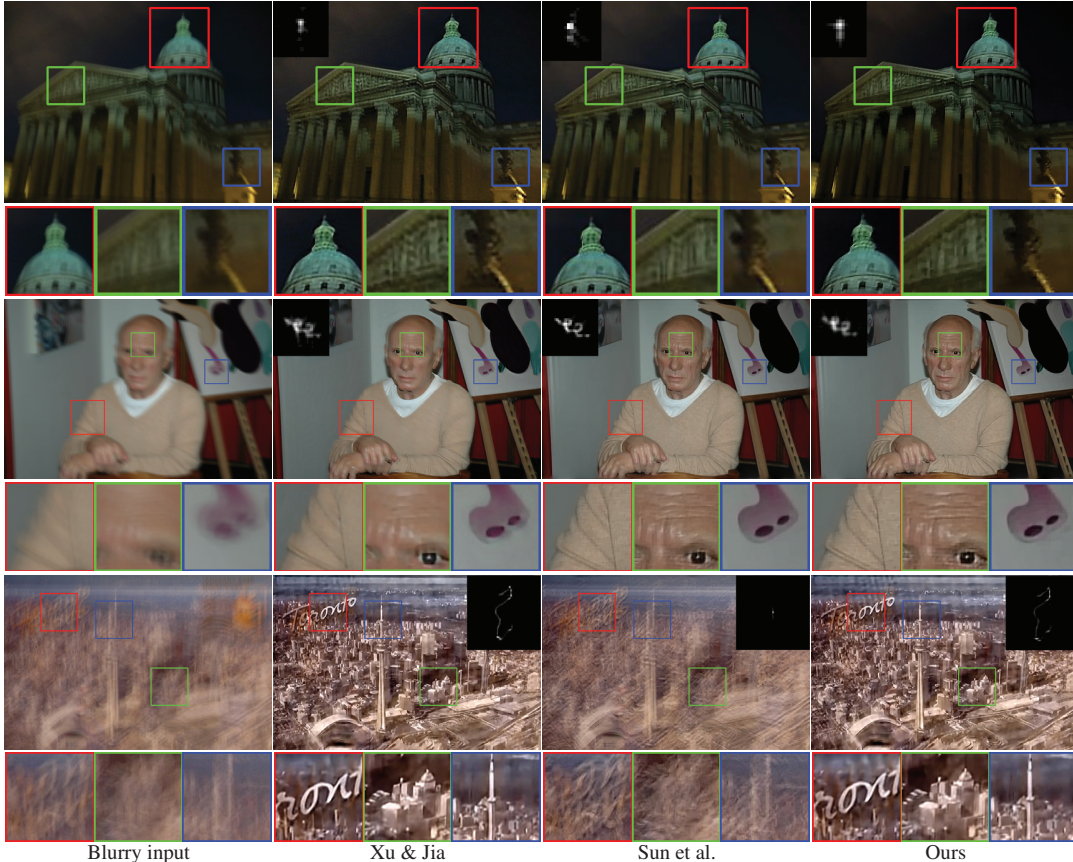


Figure 8: Comparison of deblurring results for real camera shaken photographs.

a little higher than Sun et al. [29]. When the error ratio is higher than 2.6, our method achieves the highest success rate. In Fig. 7, the deblurring results by our method is more visually plausible. As to the computational efficiency, our method is more than 12 times faster than Sun et al. [29].

5.3. Blur removal for real blurry images

We further validate the performance of our method on real blurry photographs, and compared with Xu & Jia [34] and Sun et al. [29], which are the top two methods on the two synthetic datasets. Fig. 8 presents the deblurred results, in which kernel sizes are fixed as 51×51 for the first two images, and large kernel size 95×95 is set for the third image. For the first two images, all the three methods can achieve satisfactory deblurring results, and our method can better preserve the detailed textures, e.g., the wrinkles in the second image of our method are more clear. For the third image, Sun et al. [29] fails to estimate the large kernel, while our method can achieve comparable if not superior restoration quality compared with Xu & Jia [34]. More results and evaluation on the effect of $p < 0$ can be found in the supplementary materials.

6. Conclusions

In this paper, we propose an MAP-based blind deconvolution model with iteration-wise image priors discriminatively learned from the training set of synthetic images. Experimental results demonstrated that the learned iteration-wise image priors can be well extended to the other synthetic dataset and real blurry photographs. Our method can achieve more visually plausible deblurring results than the competing gradient prior based methods, and is comparable but more efficient while compared with state-of-the-art patch prior based method.

In this work, hyper-Laplacian is adopted for modeling image gradients which is limited in modeling patch-level structures in images. Our future work will extend the discriminative learning framework to learn priors on patches or filter responses, and multi-scale scheme will be also considered.

Acknowledgements

This work is supported by NSFC grant (61271093), the program of ministry of education for new century excellent talents (NCET-12-0150), and the Hong Kong RGC General Research Fund (PolyU 5313/13E).

References

- [1] S. D. Babacan, R. Molina, M. N. Do, and A. K. Katsaggelos. Bayesian blind deconvolution with general sparse image priors. In *ECCV*, 2012. 2
- [2] S. D. Babacan, R. Molina, and A. K. Katsaggelos. Variational bayesian blind deconvolution using a total variation prior. *IEEE Transactions on Image Processing*, 18(1):12–26, 2009. 2
- [3] T. F. Chan and C. Wong. Total variation blind deconvolution. *IEEE Transactions on Image Processing*, 7(3):370–375, 1998. 1, 2, 3, 5
- [4] S. Cho and S. Lee. Fast motion deblurring. *ACM Transactions on Graphics (TOG)*, 28(5):145, 2009. 1, 2, 3, 6, 7
- [5] S. Fanello, C. Keskin, P. Kohli, S. Izadi, J. Shotton, A. Criminisi, U. Pattacini, and T. Paek. Filter forests for learning data-dependent convolutional kernels. 5
- [6] 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. 2
- [7] A. Goldstein and R. Fattal. Blur-kernel estimation from spectral irregularities. In *ECCV*, 2012. 1
- [8] M. Hirsch, C. J. Schuler, S. Harmeling, and B. Scholkopf. Fast removal of non-uniform camera shake. In *ICCV*, 2011. 1
- [9] W. Hu, J. Xue, and N. Zheng. Psf estimation via gradient domain correlation. *IEEE Transactions on Image Processing*, 21(1):386–392, 2012. 1
- [10] J. Jia and L. Xu. Structure extraction from texture via relative total variation. *ACM Transactions on Graphics (TOG)*, 31(6):139, 2012. 1
- [11] N. Joshi, R. Szeliski, and D. Kriegman. Psf estimation using sharp edge prediction. In *CVPR*, 2008. 1
- [12] N. Komodakis and N. Paragios. Mrf-based blind image deconvolution. In *ACCV*, 2013. 2
- [13] J. Kotera, F. Šroubek, and P. Milanfar. Blind deconvolution using alternating maximum a posteriori estimation with heavy-tailed priors. In *International Conference on Computer Analysis of Images and Patterns*, 2013. 3
- [14] D. Krishnan and R. Fergus. Fast image deconvolution using hyper-laplacian priors. In *NIPS*, 2009. 1, 3
- [15] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR*, 2011. 1, 2, 3, 5, 6, 7
- [16] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *CVPR*, 2009. 1, 2, 6
- [17] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Efficient marginal likelihood optimization in blind deconvolution. In *CVPR*, 2011. 2, 6, 7
- [18] G. Liu, S. Chang, and Y. Ma. Blind image deblurring using spectral properties of convolution operators. *IEEE Transactions on Image Processing*, 23:5047–5056, 2014. 1
- [19] O. V. Michailovich. An iterative shrinkage approach to total-variation image restoration. *IEEE Transactions on Image Processing*, 20(5):1281–1299, 2011. 3
- [20] J. Miskin and D. J. MacKay. Ensemble learning for blind image separation and deconvolution. In *Advances in independent component analysis*, 2000. 2
- [21] J. Pan, Z. Hu, Z. Su, and M.-H. Yang. Deblurring face images with exemplars. In *ECCV*, 2014. 3
- [22] D. Perrone and P. Favaro. Total variation blind deconvolution - the devil is in the details. In *CVPR*, 2014. 1, 2
- [23] A. N. Rajagopalan and R. Chellappa. *Motion deblurring algorithms and systems*. Cambridge Press, 2014. 1, 5
- [24] R. M. Rameshan, S. Chaudhuri, and R. Velmurugan. Joint map estimation for blind deconvolution: when does it work? In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2012. 1, 3
- [25] U. Schmidt and S. Roth. Shrinkage fields for effective image restoration. In *CVPR*, 2014. 5
- [26] U. Schmidt, C. Rother, S. Nowozin, J. Jancsary, and S. Roth. Discriminative non-blind deblurring. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, 2013. 5
- [27] Q. Shan, J. Jia, and A. Agarwala. High-quality motion deblurring from a single image. *ACM Transactions on Graphics (TOG)*, 27(3):73, 2008. 1, 2, 5, 6
- [28] F. Sroubek and P. Milanfar. Robust multichannel blind deconvolution via fast alternating minimization. *IEEE Transactions on Image Processing*, 21(4):1687–1700, 2012. 2
- [29] L. Sun, S. Cho, J. Wang, and J. Hays. Edge-based blur kernel estimation using patch priors. In *ICCP*, 2013. 2, 3, 6, 8
- [30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 6
- [31] D. Wipf and H. Zhang. Revisiting bayesian blind deconvolution. *MSRA Tech. Report*, 2013. 2
- [32] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *ECCV*, 2010. 1, 2, 3, 6, 7
- [33] L. Xu, X. Tao, and J. Jia. Inverse kernels for fast spatial deconvolution. In *ECCV*, 2014. 2, 6
- [34] L. Xu, S. Zheng, and J. Jia. Unnatural l0 sparse representation for natural image deblurring. In *CVPR*, 2013. 1, 2, 5, 8
- [35] Y. Yitzhaky, I. Mor, A. Lantzman, and N. Kopeika. Direct method for restoration of motion-blurred images. *JOSA A*, 15(6):1512–1519, 1998. 1
- [36] T. Yue, S. Cho, J. Wang, and Q. Dai. Hybrid image deblurring by fusing edge and power spectrum information. In *ECCV*, 2014. 1
- [37] H. Zhang, J. Yang, Y. Zhang, and T. S. Huang. Sparse representation based blind image deblurring. In *IEEE International Conference on Multimedia and Expo (ICME)*, 2011. 2
- [38] D. Zoran and Y. Weiss. Scale invariance and noise in natural images. In *ICCV*, 2009. 6
- [39] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *ICCV*, 2011. 6
- [40] W. Zuo, D. Meng, L. Zhang, X. Feng, and D. Zhang. A generalized iterated shrinkage algorithm for non-convex sparse coding. In *ICCV*, 2013. 2, 3, 4