Learning Iteration-wise Generalized Shrinkage-Thresholding Operators for Blind Deconvolution

Wangmeng Zuo, Dongwei Ren, David Zhang, Shuhang Gu and Lei Zhang

***Abstract*—Salient edge selection and time-varying regularization are two crucial techniques to guarantee the success of maximum a posterior (MAP)-based blind deconvolution. However, the existing approaches usually rely on carefully designed regularizers and handcrafted parameter tuning to obtain satisfactory estimation of the blur kernel. Many regularizers exhibit the structure-preserving smoothing capability, but fail to enhance salient edges. In this paper, under the MAP framework, we propose the iteration-wise ℓ_p-norm regularizers together with data-driven strategy to address these issues. First, we extend the generalized shrinkage-thresholding (GST) operator for ℓ_p-norm minimization with negative p value, which can sharpen salient edges while suppressing trivial details. Then, the iteration-wise GST parameters are specified to allow dynamical salient edge selection and time-varying regularization. Finally, instead of handcrafted tuning, a principled discriminative learning approach is proposed to learn the iteration-wise GST operators from the training dataset. Furthermore, the multi-scale scheme is developed to improve the efficiency of the algorithm. Experimental results show that, negative p value is more effective in estimating the coarse shape of blur kernel at the early stage, and the learned GST operators can be well generalized to other dataset and real world blurry images. Compared with the state-of-the-art methods, our method achieves better deblurring results in terms of both quantitative metrics and visual quality, and it is much faster than the state-of-the-art patch-based blind deconvolution method.**

***Index Terms*—blind deconvolution, kernel estimation, image deblurring, hyper-Laplacian, discriminative learning.**

I. Introduction

Blind image deconvolution aims to recover the latent sharp image $\mathbf{x}$ and blur kernel $\mathbf{k}$ from the blurry observation

$$\mathbf{y} = \mathbf{k} \otimes \mathbf{x} + \mathbf{n}, \tag{1}$$

where $\otimes$ denotes 2D convolution and $\mathbf{n}$ is additive Gaussian white noise. Blind image deconvolution generally involves two stages, i.e., blur kernel estimation and non-blind deconvolution, where the former is crucial to the success of the algorithm. There are two classes of popular blur kernel estimation strategies: variational Bayes (VB)-based and maximum a posterior (MAP)-based ones. Levin et al. [1] showed that naive MAP prefers trivial delta kernel solution while the VB-based approaches are more robust in estimating the blur kernel. This observation has motivated several VB-based methods for image deconvolution [2–7]. However, the approximation of integration is required, making VB-based methods computationally inefficient.

Recently, with the introduction of salient edge selection and time-varying regularization, interest in improved MAP has been revived for efficient blind deconvolution with many representative methods [8–11]. MAP jointly estimates the pair $(\mathbf{k}, \mathbf{x})$ by maximizing a posterior probability,

$$\max_{\mathbf{x},\mathbf{k}} \Pr(\mathbf{x}, \mathbf{k}|\mathbf{y}) \propto \max_{\mathbf{x},\mathbf{k}} \Pr(\mathbf{y}|\mathbf{x}, \mathbf{k}) \Pr(\mathbf{x}) \Pr(\mathbf{k}), \tag{2}$$

where $\Pr(\mathbf{x}, \mathbf{k}|\mathbf{y})$ denotes a posterior on $(\mathbf{x}, \mathbf{k})$, $\Pr(\mathbf{x})$ and $\Pr(\mathbf{k})$ are the priors of the latent sharp image and the blur kernel, and $\Pr(\mathbf{y}|\mathbf{x}, \mathbf{k})$ denotes the likelihood of the observation $\mathbf{y}$. The MAP model can be equivalently rewritten as,

$$\min_{\mathbf{k},\mathbf{x}} \frac{\lambda}{2\sigma_n^2} \|\mathbf{k} \otimes \mathbf{x} - \mathbf{y}\|^2 + \phi(\mathbf{x}) + \mu\varphi(\mathbf{k}), \tag{3}$$

where σ_n is the standard deviation (*std.*) of the additive Gaussian white noise, and $\phi(\mathbf{x})$ and $\varphi(\mathbf{k})$ are the regularizers on $\mathbf{x}$ and $\mathbf{k}$, respectively. Widpf and Zhang [6] recasted VB as MAP with spatially-adaptive sparse prior to explain the connections between the existing VB [2–4] and MAP approaches [12, 13].

Our work is motivated by the two key techniques, i.e., salient edge selection [12] and time-varying regularization, which have been widely adopted in MAP-based blind deconvolution. However, we re-analyze these techniques by raising three questions:

1) Salient edge selection is widely used to explicitly or implicitly recover salient edges to facilitate kernel estimation [12–17]. In [12], shock and bilateral filters are employed in each iteration to enhance strong edges while suppressing harmful small-scale textures. Actually, bilateral filter is a smoothing operator and shock filter is a sharpening operator, while the regularizers like ℓ_0-norm [16] result in a structure-preserving smoothing operator. Thus, our first question is: *is it possible to extend the existing regularizers, e.g., ℓ_p-norm [18], to achieve both smoothing and sharpening capability?*

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 5313/12E).

W. Zuo is with School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China (email: cswmzuo@gmail.com).

D. Ren is with School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China, and Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong (email: rendongweihit@gmail.com).

D. Zhang, S. Gu and L. Zhang are with Department of Computing, The Hong Kong Polytechnic University, Kowloon, Hong Kong (email: csdzhang@comp.polyu.edu.hk, cssgu@comp.polyu.edu.hk and cslzhang@comp.polyu.edu.hk).

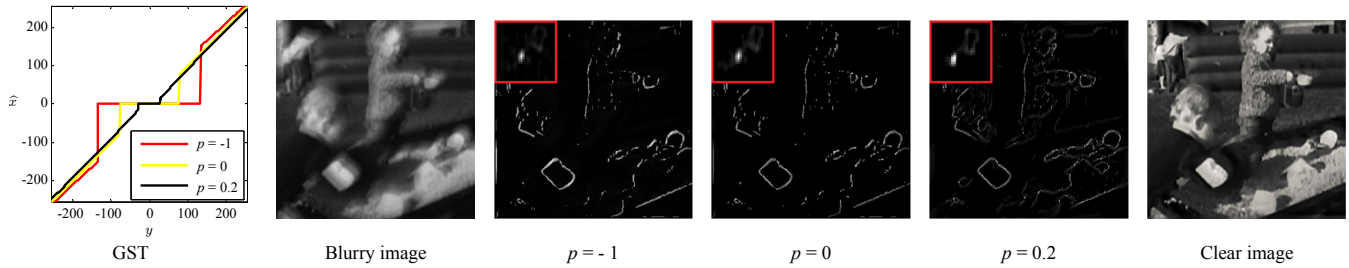


Figure 1: Illustration of the GST operators with different p values, intermediate edge images and estimated blur kernels. The GST operator is the solution to the ℓ_p-norm minimization problem $\hat{x} = \arg\min_x \frac{1}{2}(x-y)^2 + \lambda|x|^p$, and in the left plot the horizontal and vertical axes are denoted by y and $\hat{x}$, respectively. In the initial stage, the GST operator with negative p value can impose a larger threshold for suppressing small-scale textures and magnifying the salient edges. Then, by gradually increasing the p values along with iteration, we can include more gradient details for refining the kernel estimation.

2) Time-varying regularization is also widely adopted in blind deconvolution. To better estimate the blur kernel $\mathbf{k}$, the salient edges should be dynamically recovered to guide the algorithm gradually converge to the desired solution. For example, in [12] parameters of the shock and bilateral filters are tuned to select the strongest edges at first, and subsequently the estimated kernel is refined by the gradually added details. In [8, 15, 19, 20], the regularization parameter λ is set small in the first a few iterations to preserve strong edges, and then gradually increases along with iteration to produce accurate blur kernel. So, our second question is: *is there a family of priors (each iteration has its own parameters) for blind deconvolution?*
3) Most existing approaches involve carefully designed regularizers and handcrafted parameter tuning to guide the algorithms to converge to the desired solution. It is interesting to ask the question: *can we learn the iteration-wise regularization parameters using the data-driven strategy?*

In this paper, we address above problems based on the MAP framework with iteration-wise regularizers. For the *second* problem, we adopt a family of hyper-Laplacian distribution $\Pr(\mathbf{d}) \propto e^{-\|\mathbf{d}\|_p^p/\lambda}$ on gradient $\mathbf{d} = \nabla\mathbf{x}$, where each iteration has its own parameters (p_i, λ_i). The resulting ℓ_p-norm minimization problem can be readily solved using the generalized shrinkage-thresholding (GST) operator [18]. Thus, iteration-wise GST operators are exploited in our MAP framework. For the *first* problem, we extend the GST operator for ℓ_p-norm minimization with negative p value. GST is computationally efficient and can well imitate the shock and bilateral filters in enhancing salient edges while smoothing small-scale textures. As shown in Fig. 1, GST with $p = -1$ magnifies the strongest edges so that the coarse shape of blur kernel can be rapidly estimated. With the increase of p value, e.g., $p = 0$ and $p = 0.2$, GST gradually adds more gradient details to refine the estimated blur kernel.

For the *third* problem, we propose a discriminative learning method to learn the iteration-wise regularization parameters, i.e. GST operators, from the training dataset. Furthermore, multi-scale scheme is employed to improve the efficiency and effectiveness by learning regularization parameters over scales $(S-1, ..., s, ..., 0)$ and inner iterations $(1, ..., t, ..., T)$. As illustrated in Fig. 2, in the training stage a discriminative learning framework is used to greedily learn the iteration-wise regularization parameters by minimizing the weighted mean square error (MSE) between the estimated gradient images / blur kernels and the ground truth ones, i.e. we greedily learn $(\lambda^{(s,t)}, p^{(s,t)})$ after learning $\{(\lambda^{(S-1,1)}, p^{(S-1,1)}), ..., (\lambda^{(s,t-1)}, p^{(s,t-1)})\}$. In the deblurring stage, kernel estimation is performed over scales $(S-1, ..., s, ..., 0)$. One-step solutions by the augmented Lagrangian method (ALM) are adopted to update the blur kernel $\mathbf{k}$ and the latent image gradient $\mathbf{d}$ so that the parameters $(\lambda^{(s,t)}, p^{(s,t)})$ can be optimally determined via the gradient descent method. When the inner iteration t reaches the maximum value T, the estimated blur kernel $\mathbf{k}^{(s,T)}$ and latent gradient $\mathbf{d}^{(s,T)}$ would be upsampled to be used as the initialization of the finer scale $s-1$, i.e., $\mathbf{k}^{(s-1,1)}$ and $\mathbf{d}^{(s-1,1)}$. Experimental results show that the learned iteration-wise GST operators can be directly applied to other dataset and real world blurry images. The proposed method can achieve visually more plausible results than the existing gradient prior based methods, including both MAP [12, 13, 15] and VB [2], and also outperforms the state-of-the-art patch based method [21] in terms of both deblurring quality and efficiency.

This paper is a substantial extension of our pioneer work [22]. Compared with [22], the coarse-to-fine framework is adopted to improve the kernel estimation performance, more experiments are conducted to evaluate the proposed method, and more analyses and discussions are presented. We summarize our contributions from three aspects:

1) We generalize the GST operator for ℓ_p-norm minimization with negative p values, which can magnify the salient edges while suppressing trivial textures, making it very promising in coarse estimation of the blur kernel.
2) By specifying iteration-wise GST operators, a novel MAP-based blind deconvolution method is developed to naturally select proper salient edges for robust blur kernel estimation, which makes the estimated blur kernel free from the trivial delta kernel solution and gradually converge to the desired solution.
3) To avoid heavy parameters tuning, we propose a principled discriminative learning approach to learn iteration-wise GST operators from a training dataset, and the

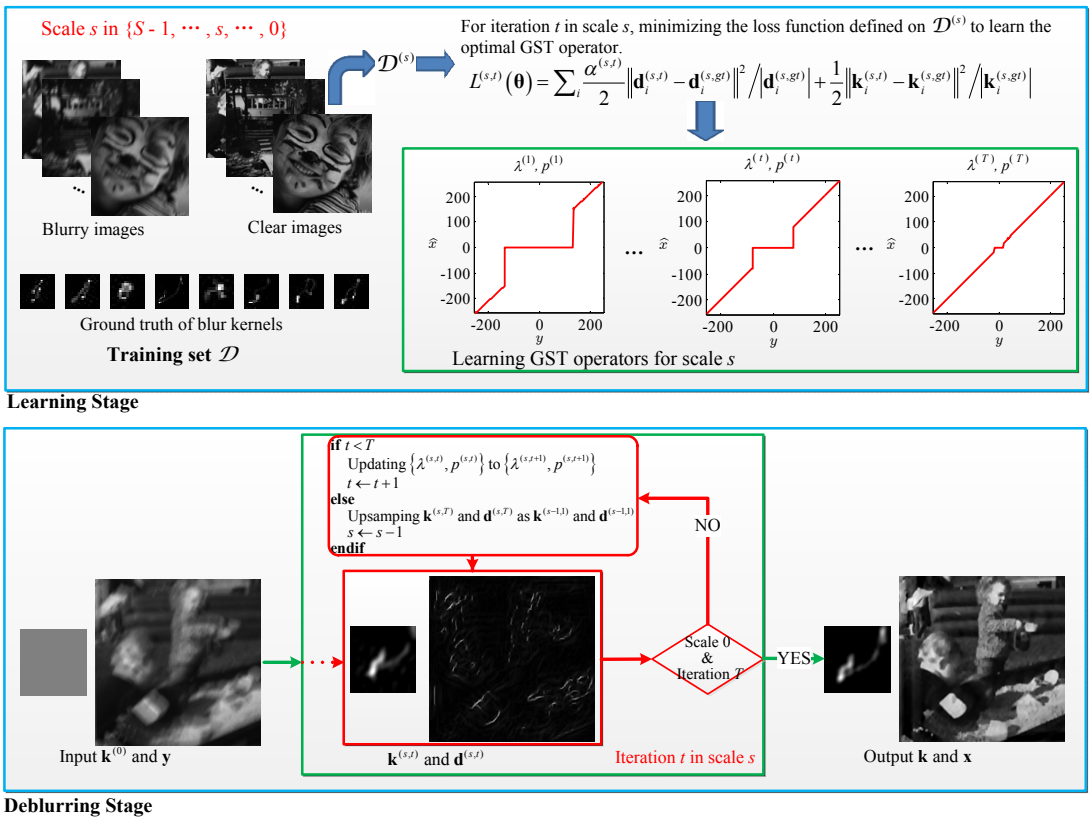


Figure 2: The proposed iteration-wise learning framework employs the coarse-to-fine strategy. In the learning stage, given the scale s, the iteration-wise GST operators can be greedily learned from the downsampled training dataset $\mathcal{D}^{(s)}$, and in the deblurring stage, the learned iteration-wise GST operators are used for blur kernel estimation.

learned parameters can be directly applied to other datasets and real blurry images.

The remainder of this paper is organized as follows. Section II reviews related work. Section III presents the proposed iteration-wise framework for blur kernel estimation together with the specially designed one-step ALM solutions. Section IV learns the optimal GST operators using the proposed principled discriminative learning framework. Section V gives the experimental results, and Section VI ends this paper with some conclusions and discussions.

II. Related Work

For better blur kernel estimation, it is critical to impose regularization to guide the MAP based algorithms to be away from the trivial delta kernel solution. In this section, we briefly review the existing regularizers on the latent image $\mathbf{x}$ and the blur kernel $\mathbf{k}$.

A. Regularization on latent image

1) ℓ_p-norm regularizers: In [23], it has been shown that the gradients of natural images can be well modeled with hyper-Laplacian distribution, where the ℓ_p-norm ($0.5 < p < 0.8$) was suggested as the regularizer in non-blind image restoration. In blind deconvolution, one natural choice is the total variation (TV) regularizer with $p = 1$ [8]. However, the small p value usually produces sparser gradients and better kernel estimation. Wipf and Zhang [6] suggested ℓ_p-norm with $p \ll 1$ and Xu et al. adopted the l_0-norm regularizer [16].

It is interesting to note that, even though Levin et al. [1] theoretically analyzed that TV based blind deconvolution (TVBD) [8] and its variants would converge to the trivial delta kernel [1, 9, 19], several TVBD algorithms [8, 9] actually succeed in estimating blur kernel and latent image. To explain the incongruence between the success of TV in [8] and the failure of TV in [1], Perrone and Favaro [9] revealed that Chan and Wong [8] 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.

Using the half-quadratic strategy, the updating of the latent image $\mathbf{x}$ in Eq. (3) can be split to the following two subproblems

$$\min_{\mathbf{x}} \frac{\lambda}{2\sigma_n^2} \|\mathbf{k} \otimes \mathbf{x} - \mathbf{y}\|^2 + \frac{\delta}{2} \|\nabla \mathbf{x} - \mathbf{w}\|^2, \tag{4}$$

$$\min_{\mathbf{w}} \frac{\delta}{2} \|\nabla \mathbf{x} - \mathbf{w}\|^2 + \|\mathbf{w}\|_p^p, \tag{5}$$

where the gradient operator $\nabla = \{\nabla_h, \nabla_v\}$ includes the horizontal and vertical directions. The $\mathbf{w}$ subproblem in (5) can be efficiently solved using the look-up table method [23] or the GST operator [18]. Specifically, when $p = 0$ the $\mathbf{w}$ subproblem can be solved by hard-thresholding [16, 18].

To improve the performance of blur kernel estimation, one usual strategy is to gradually increase the regularization

parameters λ along the iterations. However, to the best of our knowledge, no studies were given on the iteration-wise tuning of the p values. Moreover, although discriminative learning has achieved state-of-the-art performance in non-blind restoration [24–27], most existing blind deconvolution methods solely rely on handcrafted tuning of the regularization parameters.

2) Other forms of regularizers: Other gradient-based regularizers have also been developed to enhance strong edges for better kernel estimation. In [15], Krishnan et al. extended the standard ℓ_1-norm gradient prior to a normalized version $\|\nabla\mathbf{x}\|_1/\|\nabla\mathbf{x}\|_2$ to avoid the trivial delta kernel solution, and in [28], Paragios et al. proposed a discrete MRF prior to produce highly sparse gradients. Different from gradient priors that only model connections among adjacent pixels, patch-based priors that can better model large-scale structures provide a better way to extract salient edges for kernel estimation [21, 29, 30]. Moreover, by exploiting the patch sparsity of sharp image on an over-complete dictionary, sparse representation based approaches have also been developed for blind deconvolution [31, 32].

B. Regularization on blur kernel

The basic regularizer on blur kernel is the non-negative constraint $k_i \geq 0, \forall i$ and the normalization constraint $\sum_i k_i = 1$, which naturally come from the property of camera shake blur. Furthermore, the camera shake trajectory is a 1D connected path, so it is reasonable to enforce the sparsity on kernel. Although a variety of blind deconvolution methods [12, 15, 21, 33] do not explicitly impose the sparsity regularizer, in their implementations, hard-thresholding operation is performed to assign any $k_i \leq \varepsilon$ to zero, where ε is some small positive number, implicitly enhancing its sparsity. Meanwhile, other gradient sparsity regularizers, e.g., TV [8] and hyper-Laplacian [23, 34], have also been suggested to avoid the trivial delta kernel solution [10].

Moreover, based on the observation that blurring reduces the frequencies of sharp image, spectral priors of blur kernel can be extracted from the blurry observation [35–40]. Along this line, several works [38, 39] indicate that the texture-only blurry images can also be successfully handled, but still rely on the salient structures to better acquire spectral information of kernel [37, 39].

In this work, we adopt the sparsity on kernel together with the non-negative and normalization constraints, and focus on the development of iteration-wise regularizers on latent image $\mathbf{x}$.

III. Kernel estimation with iteration-wise MAP

In this section, we first formulate our iteration-wise MAP-based model in the gradient space. We then extend the GST operator for ℓ_p-norm minimization to the case with $p < 0$, which is used to sharpen salient edges while smoothing trivial textures. Finally, one-step augmented Lagrangian method (ALM) is proposed to update the latent sharp image gradient $\mathbf{d}$ and blur kernel $\mathbf{k}$ alternatively.

A. Problem Formulation

As suggested in [12, 13, 15, 33], we formulate the proposed MAP-based blind deconvolution model in the gradient space,

$$\min_{\mathbf{d},\mathbf{k}} \frac{\lambda}{2\sigma_n^2}\|\mathbf{k}\otimes\mathbf{d}-\nabla\mathbf{y}\|^2 + \phi(\mathbf{d}) + \mu\varphi(\mathbf{k}), \tag{6}$$

where $\mathbf{d} = \nabla\mathbf{x}$ with $\mathbf{d}_h = \nabla_h\mathbf{x}$ and $\mathbf{d}_v = \nabla_v\mathbf{x}$. Gradient images have been widely adopted in the existing blind deconvolution methods [1, 2, 9, 12, 15, 16]. As explained in [12], the use of gradient images can make the blur kernel estimation better conditioned. And empirical studies have validated the superiority of gradient images against intensity images [2].

As to the image priors, we impose the hyper-Laplacian prior on gradients $\mathbf{d}$,

$$\Pr(\mathbf{d}) \propto e^{-\|\mathbf{d}\|_p^p/\lambda}. \tag{7}$$

Instead of uisng fixed parameters in existing methods, in this work the parameters $\{\lambda, p\}$ are iteration-wisely set. Moreover, since the image $\mathbf{d}$ is the gradient of some image $\mathbf{x}$, the constraint $\mathbf{d} = \nabla\mathbf{x}$ should be satisfied, which has been often omitted in existing blind deconvolution models. In this work, the constraint $\mathbf{d} = \nabla\mathbf{x}$ is explicitly expressed as [41, 42],

$$\nabla_h\mathbf{d}_v = \nabla_v\mathbf{d}_h. \tag{8}$$

As to the priors on kernel $\mathbf{k}$, we simply enforce its sparsity via hyper-Laplacian prior,

$$\Pr(\mathbf{k}) \propto e^{-\mu\|\mathbf{k}\|_{0.5}^{0.5}}, \tag{9}$$

together with the non-negative constraint and the normalization constraint

$$\sum\nolimits_i k_i = 1, k_i \geq 0, \forall i. \tag{10}$$

One may choose the iteration-wise regularization parameters for $\mathbf{k}$. But our experiments show that simultaneously learning of the parameters for $\mathbf{d}$ and $\mathbf{k}$ cannot obtain better results than only learning parameters for $\mathbf{d}$. Taking both efficiency and simplicity into account, we fix $p = 0.5$ for $\mathbf{k}$ in our model.

The coarse-to-fine multi-scale strategy is adopted in most blind deconvolution methods [2, 4, 12, 13, 15, 21, 33, 43] to make the kernel estimation more robust and efficient. In this work, we also employ the multi-scale framework, where blur kernel estimation is performed over scales $(S-1, ..., s, ..., 0)$ and inner iterations $(1, ..., t, ..., T)$. The blurry input $\mathbf{y}$ is downsampled to S scales, i.e., $\mathbf{y}^{(0)}, ..., \mathbf{y}^{(s)}, ..., \mathbf{y}^{(S-1)}$. On the coarsest scale, i.e., blurry input $\mathbf{y}^{(S-1)}$, the blur kernel is first estimated, and then upsampled as the initialization of scale $S-2$. The coarse-to-fine kernel estimation procedure is recursively performed until the finest scale $\mathbf{y}^{(0)} = \mathbf{y}$. The overall multi-scale kernel estimation process is summarized in Algorithm 1.

Given scale s and iteration t, by incorporating the constraints in Eqns. (8) and (10) and the priors in Eqns. (7) and (9) into Eq. (6), the iteration-wise MAP-based blind deconvolution model is formulated as:

$$\begin{aligned}\min_{\mathbf{d},\mathbf{k}} \tfrac{\lambda^{(s,t)}}{2\sigma_n^2}\left\|\mathbf{k}\otimes\mathbf{d}-\nabla\mathbf{y}^{(s)}\right\|^2 + \|\mathbf{d}\|_{p^{(s,t)}}^{p^{(s,t)}} + \mu\|\mathbf{k}\|_{0.5}^{0.5} \\ \text{s.t.}\quad \nabla_h\mathbf{d}_v = \nabla_v\mathbf{d}_h, \textstyle\sum_i k_i = 1, k_i \geq 0, \forall i,\end{aligned} \tag{11}$$

where $\lambda^{(s,t)}$ and $p^{(s,t)}$ are iteration-wise GST parameters that can be learned from a training dataset.

Since all the scales share the same kernel estimation procedure, in the following context, given scale s and iteration t, we simplify (s,t) as (t) to more clearly present the alternative updating of the blur kernel $\mathbf{k}$ and the latent image gradient $\mathbf{d}$.

B. GST and its extension to $p<0$

The model in Eq. (11) is a non-convex ℓ_p-norm minimization problem. By far, many optimization methods [18, 44–47] have been developed for ℓ_p-norm ($0 \le p \le 1$) optimization . Among these methods, the generalized iterated shrinkage algorithm (GISA) [18] is very promising due to its computational efficiency and convergence to better solution. The key of GISA is to introduce a generalized shrinkage-thresholding (GST) operator to solve the following basic ℓ_p-norm minimization subproblem,

$$\hat{x} = \arg\min_x \frac{1}{2}(y-x)^2 + \lambda|x|^p. \tag{12}$$

The GST operator [18] is defined as

$$\hat{x} = \begin{cases} 0, & \text{if } |y| \le \tau_p^{GST}(\lambda), \\ \operatorname{sgn}(y)\, S_p^{GST}(|y|;\lambda), & \text{if } |y| > \tau_p^{GST}(\lambda), \end{cases} \tag{13}$$

where the threshold $\tau_p^{GST}(\lambda)$ is given by

$$\tau_p^{GST}(\lambda) = (2\lambda(1-p))^{\frac{1}{2-p}} + \lambda p(2\lambda(1-p))^{\frac{p-1}{2-p}}, \tag{14}$$

and the unique minimum $S_p^{GST}(|y|;\lambda)$ can be found by solving the following equation

$$S_p^{GST}(|y|;\lambda) - y + \lambda p\big(S_p^{GST}(|y|;\lambda)\big)^{p-1} = 0. \tag{15}$$

Zuo et al. [18] suggested an iterative algorithm to compute $S_p^{GST}(|y|;\lambda)$ by repeatedly performing the following shrinkage step

$$|x| = \left(|y| - \lambda p(|x|)^{p-1}\right). \tag{16}$$

When $0 \le p \le 1$, the GST operator can always obtain the optimal solution to the subproblem in Eq. (12).

From Eqns. (14) and (16), one can easily see that GST with $0 \le p < 1$ actually is a smoothing operator. However, as illustrated in [12], rough kernel estimation is benefited by both the smoothing of harmful small-scale textures and the sharpening of salient edges. To this end, Cho et al. [12] adopted the bilateral filter for image smoothing and the shock filter for edge sharpening. And one natural problem is to ask whether we can extend GST to possess the ability of thresholding and expansion. Interestingly, one can easily verify that by setting p as a negative value, the solution $S_p^{GST}(|y|;\lambda)$ by GST in Eq. (16) becomes a thresholding-expansion operator. Therefore, the extended GST operator with $p<0$ can employ the thresholding rule to suppress detailed textures and utilize the expansion rule to enhance strong edges, which in spirit is similar with the bilateral and shock filters adopted in [12].

The tradeoff between smoothness and sharpness is critical for blind deconvolution. In the early stage of kernel estimation, the sharpness plays an important role. Thus we use the GST operator with $p<0$, which can magnify the salient edges to enhance the sharpness while simultaneously introducing a larger threshold for suppressing ringing artifacts and small-scale textures. Along with iteration we increase p to appropriate positive values for optimal tradeoff between smoothness and sharpness. Moreover, instead of handcrafted tuning, we adopt a discriminative learning framework to learn the iteration-wise GST operators from a training dataset, which will be explained in detail in Section IV.

C. Alternating minimization

We adopt the alternating minimization method to solve the MAP-based blind deconvolution model in Eq. (11). (i) Fixing $\mathbf{k}$, a one-step hybrid ALM method is adopted to update $\mathbf{d}$ by considering the equality constraint in Eq. (8); (ii) Fixing $\mathbf{d}$, a one-step ALM method is employed to update $\mathbf{k}$. In the following, we describe the algorithms for updating $\mathbf{d}$ and $\mathbf{k}$ in details.

1) Updating $\mathbf{d}$ via one-step hybrid ALM: By fixing the blur kernel as the current estimation $\mathbf{k}^{(t-1)}$, the problem on $\mathbf{d}$ is formulated as

$$\min_{\mathbf{d}} \frac{\lambda^{(t)}}{2\sigma_n^2}\left\|\mathbf{k}^{(t-1)}\otimes\mathbf{d}-\nabla\mathbf{y}\right\|^2 + \|\mathbf{d}\|_{p^{(t)}}^{p^{(t)}} \ \text{ s.t. } \nabla_c^T\mathbf{d}=0, \tag{17}$$

where $\nabla_c=[\nabla_v, -\nabla_h]$. With the half-quadratic strategy, we introduce an auxiliary variable $\mathbf{w}$, decomposing the $\mathbf{d}$ problem into the following two subproblems,

$$\begin{aligned} \mathbf{w}^{(t)} &= \arg\min_{\mathbf{w}} \tfrac{\beta^{(t)}}{2}\left\|\mathbf{w}-\mathbf{d}^{(t-1)}\right\|^2 + \|\mathbf{w}\|_{p^{(t)}}^{p^{(t)}}, \\ \mathbf{d}^{(t)} &= \arg\min_{\mathbf{d}} \tfrac{\lambda^{(t)}}{2\sigma_n^2}\left\|\mathbf{k}^{(t-1)}\otimes\mathbf{d}-\nabla\mathbf{y}\right\|^2 + \tfrac{\beta^{(t)}}{2}\left\|\mathbf{w}^{(t)}-\mathbf{d}\right\|^2 \\ &\qquad \text{s.t. } \nabla_c^T\mathbf{d}=0, \end{aligned} \tag{18}$$

where β is the penalty parameter. The $\mathbf{w}$-subproblem and $\mathbf{d}$-subproblem usually need to be updated alternatively with several iterations, but this will make the relationship between $\mathbf{d}^{(t)}$ with $(\lambda^{(t)}, p^{(t)})$ hard to analyze and non-differential. Therefore, we set the number of inner-iteration to be 1, resulting in the one-step hybrid ALM method.

In Eq. (18), the solution to $\mathbf{w}$ is obtained by the extended GST operator with $p<0$. With the GST operator, the solution is,

$$w_i^{(t)} = \begin{cases} 0, & \text{if } d_i^{(t)} \le \tau_i^{(t)}, \\ \operatorname{sgn}\left(d_i^{(t)}\right)\left(\left|d_i^{(t)}\right| - \frac{1}{\beta^{(t)}}p^{(t)}\left(\left|d_i^{(t)}\right|\right)^{p^{(t)}-1}\right), & \text{else}, \end{cases} \tag{19}$$

where the threshold $\tau_i^{(t)}$ is defined as

$$\tau_i^{(t)} = \left(\frac{2}{\beta^{(t)}}\left(1-p^{(t)}\right)\right)^{\frac{1}{2-p^{(t)}}} + \frac{1}{\beta^{(t)}}p^{(t)}\left(\frac{2}{\beta^{(t)}}\left(1-p^{(t)}\right)\right)^{\frac{p^{(t)}-1}{2-p^{(t)}}}. \tag{20}$$

The $\mathbf{d}$-subproblem is a quadratic programming problem with equality constraint, which can be solved by the Lagrangian dual method [42]. The Lagrangian function is

$$\mathcal{L}(\mathbf{d},\boldsymbol{\nu}) = \frac{\lambda^{(t)}}{2\sigma_n^2}\left\|\mathbf{k}^{(t-1)}\otimes\mathbf{d}-\nabla\mathbf{y}\right\|^2 + \frac{\beta^{(t)}}{2}\left\|\mathbf{w}^{(t)}-\mathbf{d}\right\|^2 + \boldsymbol{\nu}^T\nabla_c^T\mathbf{d}, \tag{21}$$

where $\boldsymbol{\nu}$ is the Lagrangian vector. The closed-form solution can be obtained by

$$\mathbf{d}^{(t)} = \boldsymbol{\Omega}^{-1}\left(\boldsymbol{\eta} + \beta^{(t)}\mathbf{w}^{(t)}\right), \tag{22}$$

where $\boldsymbol{\Omega} = \mathbf{A}_{\mathbf{k}}^T\mathbf{A}_{\mathbf{k}} + \beta^{(t)}\mathbf{I}$, $\boldsymbol{\eta} = \frac{\lambda^{(t)}}{\sigma_n^2}\mathbf{A}_{\mathbf{k}}^T\nabla\mathbf{y} - \nabla_c\boldsymbol{\nu}^{(t)}$, and the Lagrangian vector is updated as:

$$\boldsymbol{\nu}^{(t)} = \left(\nabla_c^T\boldsymbol{\Omega}^{-1}\nabla_c\right)^{-1}\nabla_c^T\boldsymbol{\Omega}^{-1}\left(\mathbf{A}_{\mathbf{k}}^T\nabla\mathbf{y} + \beta^{(t)}\mathbf{w}^{(t+1)}\right). \tag{23}$$

The main computational burden is the matrix inversion and multiplication on the convolution matrix $\mathbf{A}_{\mathbf{k}}$ corresponding to the blur kernel $\mathbf{k}$. Assuming the periodic boundary condition of blurry image, the convolution matrix $\mathbf{A}_{\mathbf{k}}$ is a block circulant with circulant blocks (BCCB) matrix that can be diagonalized via Fourier transform, and the matrix inversion and multiplication in Eqns. (22) and (23) can be efficiently computed by using fast Fourier transform (FFT).

2) Updating $\mathbf{k}$ *via one-step ALM:* By fixing the gradient image as the updated estimation $\mathbf{d}^{(t)}$, the problem on $\mathbf{k}$ can be formulated as:

$$\min_{\mathbf{k}} \frac{\lambda^{(t)}}{2\sigma_n^2}\left\|\mathbf{k}\otimes\mathbf{d}^{(t)} - \nabla\mathbf{y}\right\|^2 + \mu\|\mathbf{k}\|_{0.5}^{0.5} \text{ s.t.} \sum_i k_i = 1, k_i \geq 0, \forall i, \tag{24}$$

By introducing two auxiliary variables $\mathbf{h} = \mathbf{k}$ and $\mathbf{g} = \mathbf{k}$, we use the standard ALM to reformulate Eq. (24) as:

$$\begin{aligned}\min_{\mathbf{k},\mathbf{h},\mathbf{g}} \tfrac{\lambda^{(t)}}{2\sigma_n^2}\left\|\mathbf{k}\otimes\mathbf{d}^{(t)} - \nabla\mathbf{y}\right\|^2 + B(\mathbf{h}) + \tfrac{\delta_1^{(t)}}{2}\left(\mathbf{1}^T\mathbf{k} - 1\right)^2 + \mu^{(t)}\|\mathbf{g}\|_{0.5}^{0.5} \\ \text{s.t. } \mathbf{k} = \mathbf{h}, \mathbf{k} = \mathbf{g},\end{aligned} \tag{25}$$

where $\mathbf{1}$ is a vector whose entries are all 1, and $B(\mathbf{h})$ is a boundary constraint defined as follows:

$$B(h_i) = \begin{cases} +\infty, & \text{if } h_i < 0, \\ 0, & \text{else.} \end{cases} \tag{26}$$

With the standard ALM, the problem on $\mathbf{k}$ in Eq. (25) can be solved by iteratively solving the following three subproblems,

$$\begin{aligned}\mathbf{h}^{(t)} &= \min_{\mathbf{h}} \tfrac{\delta_2^{(t)}}{2}\left\|\mathbf{k}^{(t-1)} - \mathbf{h}\right\|^2 + B(\mathbf{h}), \\ \mathbf{g}^{(t)} &= \min_{\mathbf{g}} \tfrac{\delta_1^{(t)}}{2}\left\|\mathbf{k}^{(t-1)} - \mathbf{g}\right\|^2 + \mu^{(t)}\|\mathbf{g}\|_{0.5}^{0.5}, \\ \mathbf{k}^{(t)} &= \min_{\mathbf{k}} \tfrac{\lambda^{(t)}}{2\sigma_n^2}\left\|\mathbf{k}\otimes\mathbf{d}^{(t)} - \nabla\mathbf{y}\right\|^2 + \tfrac{\delta_3^{(t)}}{2}\left(\mathbf{1}^T\mathbf{k} - 1\right)^2 \\ &\quad + \tfrac{\delta_2^{(t)}}{2}\left\|\mathbf{k} - \mathbf{h}^{(t)}\right\|^2 + \tfrac{\delta_1^{(t)}}{2}\left\|\mathbf{k} - \mathbf{g}^{(t)}\right\|^2.\end{aligned} \tag{27}$$

Also, the updating of $\mathbf{h}^{(t)}$, $\mathbf{g}^{(t)}$, $\mathbf{k}^{(t)}$ can be performed for several times. In this work, to make the relationship between $\mathbf{k}^{(t)}$ and $\lambda^{(t)}$ explicit, we update $\mathbf{h}^{(t)}$, $\mathbf{g}^{(t)}$, $\mathbf{k}^{(t)}$ only one step, resulting in our one-step ALM method.

As to the $\mathbf{h}$-subproblem, it can be easily solved by an entry-wise projection operator,

$$h_i^{(t)} = \begin{cases} k_i^{(t-1)}, & \text{if } k_i^{(t-1)} > 0, \\ 0, & \text{else.} \end{cases} \tag{28}$$

The $\mathbf{g}$-subproblem is a ℓ_p-norm minimization, which can be simply solved by the GST operator [18]. Finally, the $\mathbf{k}$-subproblem is a quadratic optimization problem whose closed-form solution is

$$\mathbf{k}^{(t)} = \boldsymbol{\Phi}^{-1}\boldsymbol{\zeta}, \tag{29}$$

where $\boldsymbol{\zeta} = \frac{\lambda^{(t)}}{\sigma_n^2}\mathbf{A}_{\mathbf{d}}\nabla\mathbf{y} + \delta_1^{(t)}\mathbf{g}^{(t)} + \delta_2^{(t)}\mathbf{h}^{(t)} + \delta_3^{(t)}\mathbf{1}$ and $\boldsymbol{\Phi} = \frac{\lambda^{(t)}}{\sigma_n^2}\mathbf{A}_d^T\mathbf{A}_d + \delta_1^{(t)}\mathbf{I} + \delta_2^{(t)}\mathbf{I} + \delta_3^{(t)}\mathbf{1}\mathbf{1}^T$. $\mathbf{1}\mathbf{1}^T$ is a matrix with all entries being 1, which fortunately is also a BCCB matrix, and the corresponding matrix inversion and product can be efficiently computed by FFT.

Algorithm 2 summarizes the main steps of our alternating minimization algorithm at scale s.

Algorithm 1 Multi-scale image deconvolution

Input: Blurry image $\mathbf{y}$, scale number S
Output: Blur kernel $\mathbf{k}$ and latent image $\mathbf{x}$

```
Initializing d^(S-1) and k^(S-1)
for s = S − 1 to 0 do
    Downsampling y to y^(s)
    Inputing y^(s), d^(s), k^(s) and {θ^(s,1), ..., θ^(s,t), ..., θ^(s,T)} to Algorithm 2 that returns d^(s) and k^(s)
    if s > 0 then
        Upsamping d^(s) and k^(s) to d^(s−1) and k^(s−1)
    end if
end for
k = k^(0)
Given k, recovering x by non-blind deconvolution
```

Algorithm 2 Kernel estimation on scale s

Input: Blurry image $\mathbf{y}$, $\mathbf{d}^{(0)}$, $\mathbf{k}^{(0)}$ and $\{\boldsymbol{\theta}^{(1)}, ..., \boldsymbol{\theta}^{(t)}, ..., \boldsymbol{\theta}^{(T)}\}$
Output: Blur kernel $\mathbf{k}$ and latent gradient $\mathbf{d}$

```
for t = 1 to T do
    // Lines 3-4 solve d-step Eq. (18)
    w^(t) = GST(d^(t−1), p^(t), 1/β^(t))
    d^(t) = Ω^{-1}(η + β^(t) w^(t))
    // Lines 6-8 solve k-step Eq. (27)
    h^(t) = arg min_h (δ_2^(t)/2)‖k^(t−1) − h‖^2 + B(h)
    g^(t) = GST(k^(t−1), 0.5, μ^(t)/δ_1^(t))
    k^(t) = Φ^{-1} ζ
    Updating β^(t+1), δ_1^(t+1), δ_2^(t+1), δ_3^(t+1)
end for
k = k^(t) and d = d^(t)
```

IV. Discriminative learning of iteration-wise GST operators

Given the scale number S and the inner iteration number T in each scale, the number of GST parameters to be set in the proposed iteration-wise MAP framework is $2 \times S \times T$, i.e. $\{\{(\lambda^{(S-1,t)}, p^{(S-1,t)})\}_{t=1}^T, ..., \{(\lambda^{(s,t)}, p^{(s,t)})\}_{t=1}^T, ..., \{(\lambda^{(0,t)}, p^{(0,t)})\}_{t=1}^T\}$. For such a large amount of parameters, hand-crafted tuning is impractical. Therefore, in this section we propose a discriminative learning framework [48] to learn the iteration-wise GST operators from a training dataset.

Denote the training dataset by $\mathcal{D} = \{(\mathbf{d}_i^{(gt)}, \mathbf{k}_i^{(gt)}, \nabla\mathbf{y}_i)\}_{i=1}^N$, which includes N blurry images together with the groundtruth clear images and blur kernels, i.e., the gradient $\mathbf{d}_i^{(gt)}$ of the clear image, the blur kernel $\mathbf{k}_i^{(gt)}$, and the gradient of the blurry image $\nabla\mathbf{y}_i$. Since the multi-scale framework is adopted in kernel estimation, given scale s, the training dataset $\mathcal{D}$ is sampled to the corresponding size, i.e., $\mathcal{D}^{(s)} =$

$\{(\mathbf{d}_i^{(s,gt)}, \mathbf{k}_i^{(s,gt)}, \nabla\mathbf{y}_i^{(s)})\}_{i=1}^N$ with $\mathcal{D}^{(0)} = \mathcal{D}$. Given scale s and iteration t, to estimate $\boldsymbol{\theta}^{(s,t)} = \{\lambda^{(s,t)}, p^{(s,t)}\}$, the loss function is defined as the weighted MSE on $\mathcal{D}^{(s)}$,

$$\begin{aligned} L^{(s,t)}(\boldsymbol{\theta}) &= \sum_{i=1}^{N} L_i^{(s,t)}(\boldsymbol{\theta}) = \textstyle\sum_{i=1}^{N} \alpha^{(t)} L_{\mathbf{d}_i}^{(s,t)}(\boldsymbol{\theta}) + L_{\mathbf{k}_i}^{(s,t)}(\boldsymbol{\theta}) \\ &= \sum_{i=1}^{N} \frac{\alpha^{(s,t)}}{2} \left\| \mathbf{d}_i^{(s,t)} - \mathbf{d}_i^{(s,gt)} \right\|^2 / \left| \mathbf{d}_i^{(s,gt)} \right| + \frac{1}{2} \left\| \mathbf{k}_i^{(s,t)} - \mathbf{k}_i^{(s,gt)} \right\|^2 / \left| \mathbf{k}_i^{(s,gt)} \right|, \end{aligned} \tag{30}$$

where $|\cdot|$ 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. Thus the second term in Eq. (30) should be emphasized, especially at the early stage. Therefore, we let the α value increase along with the scale numbers and iteration numbers. In the first a few iterations the GST operators are learned with small α for rapid kernel estimation, and then the GST operator is learned with larger α value for refining the estimated kernel while yielding better estimation on the latent clear image.

The overall learning procedure over scales is summarized in Algorithm 3 with more details introduced as follows. Similar to the kernel estimation in Section III, given scale s and iteration t, we simplify (s,t) as (t) to more clearly present the learning algorithm.

A. Learning algorithm

We greedily learn the GST parameters for each iteration using the gradient descent method, and the partial derivative of loss function $L_i^{(t)}(\boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}^{(t)}$ needs to be computed first. The updating rules of blur kernel $\mathbf{k}$ and the latent gradient image $\mathbf{d}$ in Section III are specially designed as one-step ALM solutions, which make the loss function differentiable. From Eqns. (19), (22), (29), and (30), we have the following observations:

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

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

$$\frac{\partial L_i^{(t)}}{\partial \boldsymbol{\theta}} = \left(\alpha^{(t)} \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial p}, \alpha^{(t)} \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial \lambda} + \frac{\partial L_{\mathbf{k}_i}^{(t)}}{\partial \lambda} \right). \tag{31}$$

The partial derivative of $L_i^{(t)}$ w.r.t. p only includes the partial derivative of $L_{\mathbf{d}_i^{(t)}}$ w.r.t. p, which based on Eq. (22) can be obtained by

$$\begin{aligned} \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial p} &= \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial \mathbf{d}_i^{(t)}} \frac{\partial \mathbf{d}_i^{(t)}}{\partial \mathbf{w}_i^{(t)}} \frac{\partial \mathbf{w}_i^{(t)}}{\partial p} \\ &= \left(\beta^{(t)} / \left| \mathbf{d}_i^{gt} \right| \right) \left(\mathbf{d}_i^{(t)} - \mathbf{d}_i^{gt} \right)^T \boldsymbol{\Omega}^{-1} \frac{\partial \mathbf{w}_i^{(t)}}{\partial p}. \end{aligned} \tag{32}$$

The partial derivative of $\mathbf{w}_i^{(t)}$ w.r.t. p can be obtained based on Eq. (19)

$$\frac{\partial w_i^{(t)}}{\partial p} = \begin{cases} 0, & \text{if } d_i^{(t)} \le \tau_i^{(t)}, \\ -\frac{\operatorname{sgn}\left(d_i^{(t)}\right)}{\beta^{(t)}} \left(\left(\left| d_i^{(t)} \right| \right)^{p-1} + p \left(\left| d_i^{(t)} \right| \right)^{p-1} \ln \left(\left| d_i^{(t)} \right| \right) \right), & \text{else}, \end{cases} \tag{33}$$

where the threshold $\tau_i^{(t)}$ in Eq. (20) is approximated by setting p as $p^{(t-1)}$.

As to the partial derivative of $L_i^{(t)}$ w.r.t. λ, it includes the partial derivatives of $L_{\mathbf{d}_i}^{(t)}$ w.r.t. λ and $L_{\mathbf{k}_i}^{(t)}$ w.r.t. λ. First, based on Eq. (22), the partial derivative of $L_{\mathbf{d}_i}^{(t)}$ w.r.t. λ can be obtained by

$$\begin{aligned} \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial \lambda} &= \frac{\partial L_{\mathbf{d}_i}^{(t)}}{\partial \mathbf{d}_i^{(t)}} \frac{\partial \mathbf{d}_i^{(t)}}{\partial \lambda} \\ &= 1 / \left(\sigma_n^2 \left| \mathbf{d}_i^{gt} \right| \right) \left(\mathbf{d}_i^{(t)} - \mathbf{d}_i^{gt} \right)^T \boldsymbol{\Omega}^{-1} \left(\mathbf{A}_{\mathbf{k}}^T \nabla\mathbf{y}_i - \mathbf{A}_{\mathbf{k}}^T \mathbf{A}_{\mathbf{k}} \right). \end{aligned} \tag{34}$$

and then based on Eq. (29), the partial derivative of $L_{\mathbf{k}_i}^{(t)}$ w.r.t. λ can be obtained by

$$\begin{aligned} \frac{\partial L_{\mathbf{k}_i}^{(t)}}{\partial \lambda} &= \frac{\partial L_{\mathbf{k}_i}^{(t)}}{\partial \mathbf{k}_i^{(t)}} \frac{\partial \mathbf{k}_i^{(t)}}{\partial \lambda} \\ &= 1 / \left(\sigma_n^2 \left| \mathbf{k}_i^{gt} \right| \right) \left(\mathbf{k}_i^{(t)} - \mathbf{k}_i^{gt} \right)^T \boldsymbol{\Phi}^{-1} \left(\mathbf{A}_{\mathbf{d}}^T \nabla\mathbf{y}_i - \mathbf{A}_{\mathbf{d}}^T \mathbf{A}_{\mathbf{d}} \right). \end{aligned} \tag{35}$$

Once the partial derivative of $L^{(t)}(\boldsymbol{\theta})$ w.r.t. $\boldsymbol{\theta}$ is obtained, the optimal $\boldsymbol{\theta}^{(t)}$ can be obtained by any gradient descent method, and we adopt the gradient-based L-BFGS method [49]. On scale s, the procedures of learning iteration-wise GST operators are summarized in Algorithm 4.

Algorithm 3 Learning GST operators over scales

Input: Training set $\mathcal{D}$, scale number S
Output: $\left\{ \{\boldsymbol{\theta}^{(S-1,t)}\}_{t=1}^T, ..., \{\boldsymbol{\theta}^{(s,t)}\}_{t=1}^T, ..., \{\boldsymbol{\theta}^{(0,t)}\}_{t=1}^T \right\}$

1: Denoting $\boldsymbol{\theta}'$ as α, μ, β, δ_1, δ_2, δ_3 and initializing $\boldsymbol{\theta}'^{(S,T)}$
2: **for** $s = S-1$ to 0 **do**
3: Downsampling training set $\mathcal{D}$ to $\mathcal{D}^{(s)}$
4: Initializing $\boldsymbol{\theta}'^{(s,1)}$ as $\boldsymbol{\theta}'^{(s+1,T)}$
5: Inputing $\mathcal{D}^{(s)}$ and $\boldsymbol{\theta}'^{(s,1)}$ to Algorithm 4 to learn the optimal parameters of scale s, i.e. $\{\boldsymbol{\theta}^{(s,t)}\}_{t=1}^T$
6: **end for**

Algorithm 4 Learning GST operators on scale s

Input: Training set $\mathcal{D}$ and $\alpha^{(1)}$, $\mu^{(1)}$, $\beta^{(1)}$, $\delta_1^{(1)}$, $\delta_2^{(1)}$, $\delta_3^{(1)}$
Output: $\left\{ \boldsymbol{\theta}^{(1)}, ..., \boldsymbol{\theta}^{(t)}, ..., \boldsymbol{\theta}^{(T)} \right\}$

1: **for** $t = 1$ to T **do**
2: grad $= 0$
3: **for** $i = 1$ to N **do**
4: Updating $\mathbf{d}_i^{(t)}$ Eq. (18) and $\mathbf{k}_i^{(t)}$ Eq. (27)
5: grad $=$ grad $+ \partial L_i^{(t)}(\boldsymbol{\theta}) / \partial \boldsymbol{\theta}$
6: **end for**
7: Using gradient based L-BFGS method to search optimal $\boldsymbol{\theta}^{(t)}$
8: Updating $\alpha^{(t+1)}$, $\mu^{(t+1)}$ and penalty parameters $\beta^{(t+1)}$, $\delta_1^{(t+1)}$, $\delta_2^{(t+1)}$, $\delta_3^{(t+1)}$
9: **end for**

B. Implementation

In our implementation, some extra constraints are taken to improve the robustness and stability of the learned iteration-wise GST operators. In [8, 15, 19, 20], the regularization parameter λ begins with some small value and gradually increases along with the iteration numbers. As to the p value, $p = 0$ is first adopted to estimate the blur kernel, and then $p = 0.5$ is adopted for the final restoration [16]. As a summary,

both λ and p values should be non-decreasing along with the iteration numbers. Thus, in our greedy learning procedure, for each scale s, the non-decreasing constraints on both λ and p are imposed, i.e., $\lambda^{(s,t+1)} \geq \lambda^{(s,t)}$ and $p^{(s,t+1)} \geq p^{(s,t)}$, and over the scales, it is reasonable to consider the constraints $\lambda^{(s-1,1)} \geq \lambda^{(s,T)}$ and $p^{(s-1,1)} \geq p^{(s,T)}$. As to the search range, λ is constrained in $[0.5, 5]$, and p is constrained in $[-1, 0.2]$.

Moreover, we set the scale number $S = 5$, the number of inner iterations in each scale $T = 20$, and the dowmsampling rate as 2. Thus, there are 200 GST parameters to be learned in the proposed iteration-wise MAP framework. Other parameters, including regularization weight μ and penalty parameters $\beta, \delta_1, \delta_2, \delta_3$, should also be non-decreasing along with the scales and iterations. Specifically, the regularization weight μ on blur kernel $\mathbf{k}$ is initialized as 1×10^{-6}, and updated by $\mu^{(t+1)} = \min\left(1.1 \times \mu^{(t)}, 1 \times 10^{-3}\right)$. The penalty parameter β is initialized as 1×10^{-4}, and updated by $\beta^{(t+1)} = \min\left(1.5 \times \beta^{(t)}, 1 \times 10^{-2}\right)$. The three δs are initialized as 1×10^{-3}, and updated by $\delta^{(t+1)} = \min\left(1.1 \times \delta^{(t)}, 1 \times 10^{-1}\right)$. Moreover, we use the noise estimation method [50] to estimate the *std*. of the additive Gaussian white noise, and it is assumed in the range $[1 \times 10^{-3}, 1 \times 10^{-2}]$.

C. Discussion

Our learning method is conceptually similar to the cascade of shrinkage field (CSF) [25], which also adopts the discriminative learning approach to learn the iteration-wise image priors on filter responses. After greedy learning of the parameters iteration-by-iteration, a joint learning over iterations is performed to globally fine-tune the iteration-wise parameters. However, the CSF is designed for non-blind deconvolution, while our model is designed for blind deconvolution where the robust blur kernel estimation can be achieved by the learned GST operators. Moreover, we adopt the multi-scale scheme to learn the iteration-wise GST operators over scales and iterations, while CSF only learns the parameters at the finest scale.

V. Experimental Results

For quantitative evaluation, two benchmark datasets of blurry images are used to test the performance of the proposed method, where Levin et al.'s dataset [1] is adopted to illustrate the learnability of iteration-wise GST operators, and Sun et al.'s dataset [21] is employed to validate the generalization of the proposed method to other blurry images. To evaluate the restoration quality, we use three quantitative metrics, i.e., PSNR, SSIM [51], and error ratio [1] which is defined as,

$$\text{Error Ratio} = \left\|\mathbf{x}^{gt} - \hat{\mathbf{x}}_{\hat{\mathbf{k}}}\right\| / \left\|\mathbf{x}^{gt} - \hat{\mathbf{x}}_{\mathbf{k}^{gt}}\right\|, \qquad (36)$$

where $\hat{\mathbf{x}}_{\hat{\mathbf{k}}}$ and $\hat{\mathbf{x}}_{\mathbf{k}^{gt}}$ are the recovered images using the estimated blur kernel $\hat{\mathbf{k}}$ and the groundtruth blur kernel $\mathbf{k}^{gt}$, respectively. To evaluate the computational efficiency, we report the CPU running time. Our method is compared with several competing methods, including three edge-based MAP approaches [12, 15, 16, 21] and one VB approach [2]. All the experiments were ran on a computer with 3.30GHz Intel(R) Xeon(R) CPU.

We further evaluate the proposed method on real world blurry images, and compare it with two state-of-the-art edge-based uniform deblurring approaches [16, 21]. The reason to choose these methods is that they are the top two competing methods based on the quantitative metrics on the two datasets.

A. Training the iteration-wise priors

Levin et al.'s dataset [1] contains 4 clear images and 8 blur kernels, and is used to determine model parameters and to demonstrate the feasibility of the proposed discriminative learning method. In the learning procedure, the trade-off parameter $\alpha^{(s,t)}$ is introduced to balance the contributions of the blur kernel $\mathbf{k}$ and the image gradient $\mathbf{d}$ terms in Eq. (30), where a small α is first adopted to avoid the trivial delta kernel solution and subsequently α is increased to guide the algorithm to converge to the groundtruth blur kernel and clear image. Thus, $\alpha^{(s,t)}$ is initialized with a sufficiently small number, i.e., $\alpha^{(S-1,1)} = 1 \times 10^{-3}$, and gradually increases along with the training iterations, i.e., $\alpha^{(s,t+1)} = \min(\rho\alpha^{(s,t)}, 1)$.

We analyze the effect of the increasing rate ρ. Fig. 3 shows the learned GST parameters obtained using three ρ values, i.e., $1.0, 1.1, 1.2$. For any ρ, both λ and p begin with small values and then gradually increase along with scales and iterations, validating the consistency of the trend for dynamic selection of GST operators for kernel estimation. The larger $\rho = 1.2$ makes α increase more rapidly, making the learned GST parameters also increase rapidly. Specially, when $\rho = 1.0$, the α is fixed to its initial value 1×10^{-3}, and the contribution of the image gradient $\mathbf{d}$ term keeps constant. Table I lists the mean PSNR, SSIM and Error Ratio values using the learned GST operators with different ρ values. In terms of all the three quantitative metrics, our learning method with $\rho = 1.1$ leads to the best results, and thus $\rho = 1.1$ is adopted in the following experiments.

Table I: The results on Levin et al.'s dataset [1] by our learning methods with different ρ values

	PSNR	SSIM	Error Ratio
$\rho = 1.2$	30.88	0.9198	1.2303
$\rho = 1.1$	30.91	0.9238	1.2210
$\rho = 1.0$	30.24	0.9076	1.2635

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

	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 [12]	28.83	0.8801	1.5402	1.3951
Levin et al. [2]	28.79	0.8922	1.5592	78.263
Xu & Jia [16]	29.45	0.9000	1.4071	1.1840
Sun et al. [21]	30.85	0.9191	1.2244	191.03
Ours(-1)	28.63	0.8899	1.6235	10.403
Ours(0.2)	29.08	0.9057	1.4181	10.830
Ours(Logistic)	30.89	0.9214	1.2228	10.549
Ours(Re-train)	30.80	0.9188	1.2257	10.981
Ours	30.91	0.9238	1.2210	10.998

To validate the effectiveness of iteration-wise priors, we also consider two variants of our method by fixing the p value as

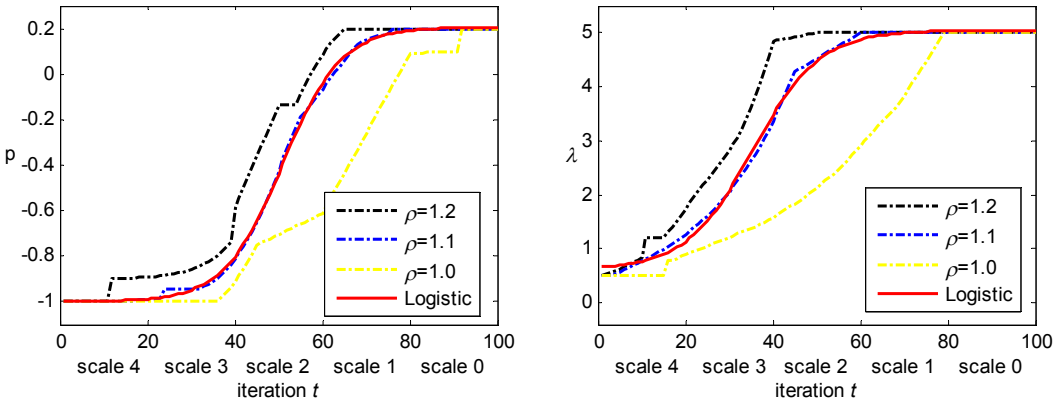


Figure 3: The learned iteration-wise p and λ values for each scale and iteration.

-1 (i.e. Ours(-1)) and 0.2 (i.e. Ours(0.2)), while the other parameters remain the same with our method (i.e., Ours). From Table II, the proposed method with iteration-wise p values outperforms Ours(0.2) and Ours(-1) in terms of all the three performance metrics. Fig. 5 shows the deblurring results using these three methods. One can see that the proposed method with iteration-wise p values can obtain more accurate estimation of blur kernel and visually more pleasing result of the deblurring image, validating the superiority of iteration-wise priors against the fixed ones.

We further evaluate the stability of the learned parameters from two aspects. In these experiments, except the iteration-wise λ and p values we keep the other parameters unchanged. First, we fit these curves with the modified logistic function to avoid the storage of the learned paramters. As shown in Fig. 3 and Table II, the fitted curves can well approximate the learned parameters, and the results based on the fitted curves are moderately inferior to those based on the learned curves. Considering that our method has 200 parameter values to store, we still report the results of our method based on the learned optimal parameters in the following experiments. Second, we use another synthetically blurred dataset to re-train the iteration-wise priors. The dataset contains 10 clear images with diverse contents and 8 blur kernels randomly selected from the dataset [52]. Then the learned priors were applied to Levin et al.'s dataset. Table II shows that the re-trained priors can also produce satisfying results. Since the images and blur kernels are quite different from Levin et al.'s dataset, it is reasonable that the results by the newly trained parameters are marginally inferior to our method based on the original parameter curves. Therefore, we conclude that the learned parameters are stable with the training dataset, and one can obtain similar performance by using the learned parameters on other training datasets.

Finally, we compare the proposed method with five competing approaches, including four MAP approaches [12, 13, 15, 21] and one VB approach [2]. The three quantitative metrics and running time of all the methods are listed in Table II. In terms of all three quantitative metrics, the proposed method performs much better than the gradient-based methods [2, 12, 13, 15] and is even a little better than Sun et al.'s patch-based method [21]. Fig. 6 shows the curves of the accumulated error ratios, and the proposed method succeeds in recovering 100% cases with the error ratio below 3, which is deemed as the threshold for visually plausible perception. Moreover, we provide the mean PSNR and SSIM values w.r.t. each individual blur kernel, as shown in Fig. 7, from which one can see that the proposed method can achieve the best deblurring quality in general. Fig. 4 shows the visual deblurring results of one blurry image. The blur kernel estimated by our method is more accurate than the other methods, leading to more visually plausible latent image. As to the computational efficiency, the proposed method is slower than the methods of Cho & Lee [12] and Xu & Jia [16], partially due to their optimized implementations in C/C++, but is more than 15 times faster than Sun et al. [21].

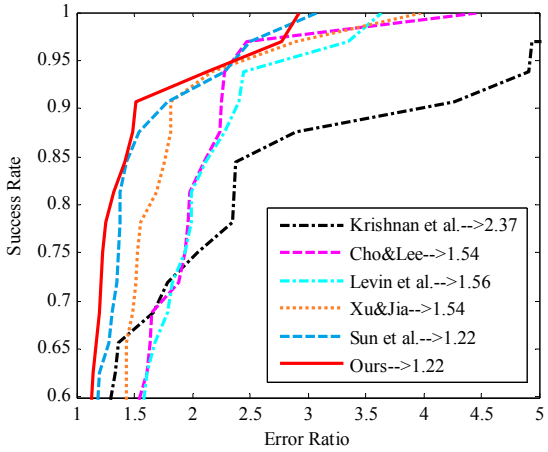


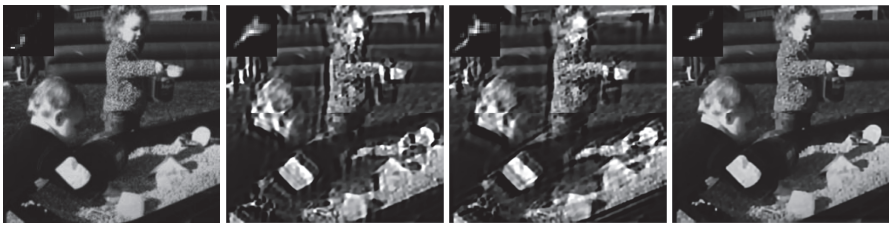
Figure 6: Comparison of error ratios of the competing methods on Levin et al.'s dataset, and the values in the legend indicate the corresponding mean error ratio.

B. Evaluation on Sun et al.'s dataset

Sun et al.'s dataset [21] includes 80 clear images and 8 blur kernels, which is used to test the generalization capability of the learned GST operators. In the blur kernel estimation stage, the proposed method directly adopts the iteration-wise GST operators learned on Levin et al.'s dataset to the blurry images from Sun et al.'s dataset [21]. In the non-blind deconvolution

Ground truth Cho & Lee [12] Krishnan et al. [15] Levin et al. [2] Xu & Jia [16] Sun et al. [21] Ours

Figure 4: Example of deblurring results on Levin et al.'s dataset.



Ground truth Ours(-1) Ours(0.2) Ours

Figure 5: Example of deblurring results of the three variants of our method on Levin et al's dataset.

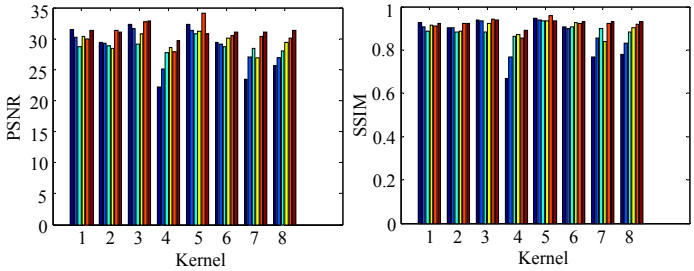


Figure 7: Mean PSNR and SSIM for each kernel on Levin et al.'s dataset. The methods from left to right are Krishnan et al. [15], Cho & Lee [12], Levin et al. [2], Xu & Jia [16], Sun et al. [21], and Ours, respectively.

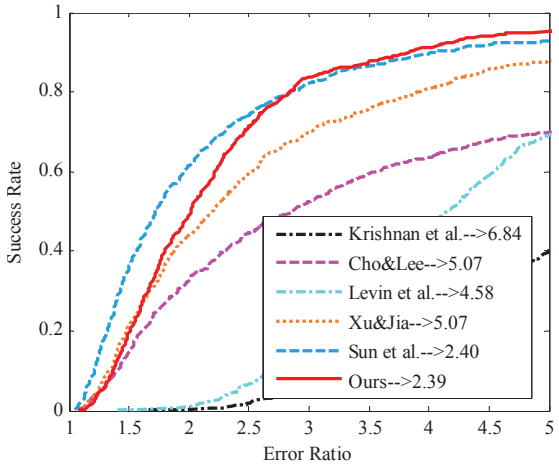


Figure 8: Comparison of error ratios of the competing methods on Sun et al.'s dataset, and the values in the legend indicate the corresponding mean error ratio.

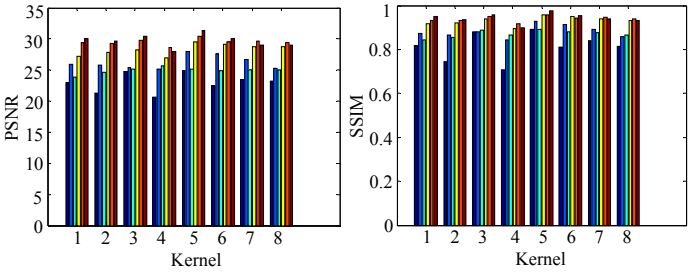


Figure 9: Mean PSNR and SSIM of each kernel on Sun et al.'s dataset. The methods from left to right are Krishnan et al. [15], Cho & Lee [12], Levin et al. [2], Xu & Jia [16], Sun et al. [21], and Ours, respectively.

Table III: Comparisons of mean PSNR, mean SSIM, mean error ratio and mean running time (seconds) on Sun et al.'s dataset [21].

	PSNR	SSIM	Error Ratio	Time
Known **k**	32.35	0.9536	1.0000	—
Krishnan et al. [15]	22.76	0.8136	6.8351	159.29
Cho & Lee [12]	26.13	0.8624	5.0731	10.518
Levin et al. [2]	24.64	0.8606	4.5798	518.59
Xu & Jia [16]	28.11	0.9016	3.2843	6.2940
Sun et al. [21]	29.32	0.9200	2.4036	3911.1
Ours(-1)	28.03	0.9032	3.2083	99.193
Ours(0.2)	28.58	0.9152	2.9802	98.231
Ours	29.35	0.9231	2.3901	98.071

stage, for fair comparison the same method [53] is used to perform the final deblurring for each of the competing methods.

Table III lists the three quantitative metrics and running time of all the methods on Sun et al.'s dataset, and the proposed method achieves better results than all the competing methods. It is interesting to note that, our method is nearly 40 times faster than Sun et al. [21], partially because the test image sizes from Sun et al.'s dataset is generally much larger than those from Levin et al.'s dataset. Moreover, the proposed method with iteration-wise p values is also superior to Ours(0.2) and

Ours(-1) on Sun et al.'s dataset, which demonstrates that the superiority of iteration-wise priors can be well generalized to other blurry images. In Fig. 11, the deblurring result with iteration-wise p values obviously suffers less artifacts at *windwill*.

As shown in Fig. 8, when the error ratio is more than 2.6, the proposed method achieves the best success rate. As illustrated in Fig. 9, in terms of both PSNR and SSIM, our method can obtain better deblurring quality for 5 out of 8 kernels. Fig. 10 shows the visual effect comparison of the competing methods, where our method performs better in recovering detailed textures. In Fig. 12, although all the methods can not accurately estimate the blur kernel, our deblurring result is more visually plausible.

C. Evaluation on real blurry images

In this subsection we evaluate the performance of the proposed method on real blurry photographs, and compare it with the the top two competing methods based on Tables II and III, i.e., Xu & Jia [16] and Sun et al. [21]. Fig. 13 shows the deblurring results on three real blurry images. For the image *roma*, our method and Sun et al.'s method can achieve satisfactory deblurring results, while the result by Xu & Jia has visible color distortions in the red close-up. The second image is taken in low light condition. The results by Xu & Jia and Sun et al. suffer severe distortions and noise in the red and green close-ups, while the result by our method is more clear and visually plausible. Moreover, the third image *vehicle* is severely blurred, and our method achieves much better result. For example, the license number can be roughly read as "N15 5826" from the deblurring result by our method, but it is difficult to be recognized from the results by both Xu & Jia [16] and Sun et al. [21]. For the green and yellow close-ups, although all the results are not good, the result by our method is visually more pleasant, while the distortions like ringing effects are much more severe for Xu & Jia and Sun et al.

VI. Conclusions

In this paper, by generalizing the GST operator to the case with $p < 0$, we proposed an iteration-wise MAP framework for blind deconvolution. Then a discriminative learning method was developed to learn iteration-wise GST operators from a blurry image set. The learned GST operators begin with $p < 0$ to avoid trivial delta kernel solution, and gradually increase with iterations for accurate blur kernel estimation. The proposed method can be directly applied to other dataset and real world blurry images. Experimental results showed that the proposed method performs better than the competing methods in terms of both quantitative metrics and visual effect, and is much faster than the state-of-the-art patch-based method [21].

The proposed iteration-wise learning method was designed on the image gradients, and thus has limitations to model patch-level structures. In our future work, we will investigate the appropriate framework to learn iteration-wise priors for image patches or filter responses. Moreover, to improve kernel estimation performance, joint learning over iterations can also be used to fine-tune the greedily learned parameters.

Acknowledgement

The authors would like to thank the associate editor and the anonymous reviewers for their constructive suggestions.

References

[1] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *CVPR*, 2009.

[2] ——, "Efficient marginal likelihood optimization in blind deconvolution," in *CVPR*, 2011.

[3] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational bayesian blind deconvolution using a total variation prior," *IEEE Transactions on Image Processing*, vol. 18, no. 1, pp. 12–26, 2009.

[4] 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)*, vol. 25, no. 3, pp. 787–794, 2006.

[5] S. D. Babacan, R. Molina, M. N. Do, and A. K. Katsaggelos, "Bayesian blind deconvolution with general sparse image priors," in *ECCV*, 2012.

[6] D. Wipf and H. Zhang, "Analysis of bayesian blind deconvolution," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2013.

[7] D. Perrone, R. Diethelm, and P. Favaro, "Blind deconvolution via lower-bounded logarithmic image priors," in *Energy Minimization Methods in Computer Vision and Pattern Recognition*, 2015.

[8] T. F. Chan and C. Wong, "Total variation blind deconvolution," *IEEE Transactions on Image Processing*, vol. 7, no. 3, pp. 370–375, 1998.

[9] D. Perrone and P. Favaro, "Total variation blind deconvolution - the devil is in the details," in *CVPR*, 2014.

[10] 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.

[11] Z. Hu, S. Cho, J. Wang, and M.-H. Yang, "Deblurring low-light images with light streaks," in *CVPR*, 2014.

[12] S. Cho and S. Lee, "Fast motion deblurring," *ACM Transactions on Graphics (TOG)*, vol. 28, no. 5, p. 145, 2009.

[13] L. Xu and J. Jia, "Two-phase kernel estimation for robust motion deblurring," in *ECCV*, 2010.

[14] N. Joshi, R. Szeliski, and D. Kriegman, "Psf estimation using sharp edge prediction," in *CVPR*, 2008.

[15] D. Krishnan, T. Tay, and R. Fergus, "Blind deconvolution using a normalized sparsity measure," in *CVPR*, 2011.

[16] L. Xu, S. Zheng, and J. Jia, "Unnatural l0 sparse representation for natural image deblurring," in *CVPR*, 2013.

[17] J. Jia and L. Xu, "Structure extraction from texture via relative total variation," *ACM Transactions on Graphics (TOG)*, vol. 31, no. 6, p. 139, 2012.

[18] W. Zuo, D. Meng, L. Zhang, X. Feng, and D. Zhang, "A generalized iterated shrinkage algorithm for non-convex sparse coding," in *ICCV*, 2013.

[19] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," *ACM Transactions on Graphics (TOG)*, vol. 27, no. 3, p. 73, 2008.

[20] A. N. Rajagopalan and R. Chellappa, *Motion deblurring-algorithms and systems*. Cambridge University Press, 2014.

[21] L. Sun, S. Cho, J. Wang, and J. Hays, "Edge-based blur kernel estimation using patch priors," in *IEEE international conference on computational photography (ICCP)*, 2013.

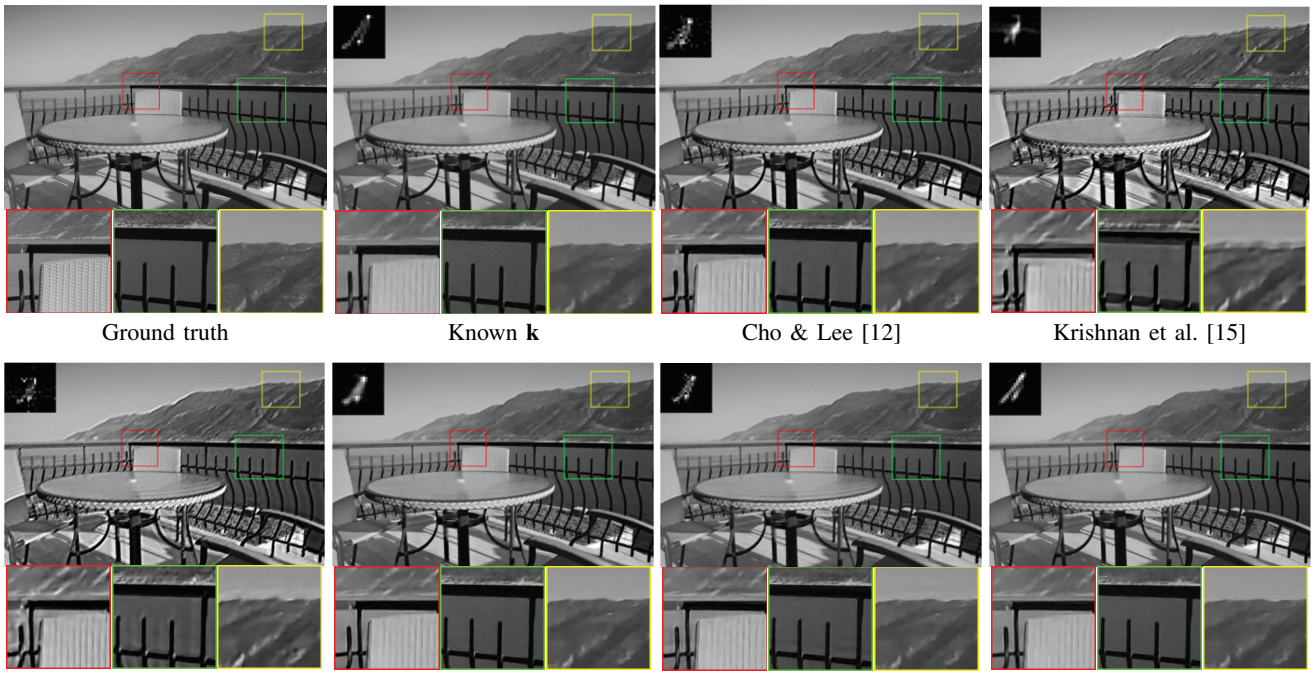
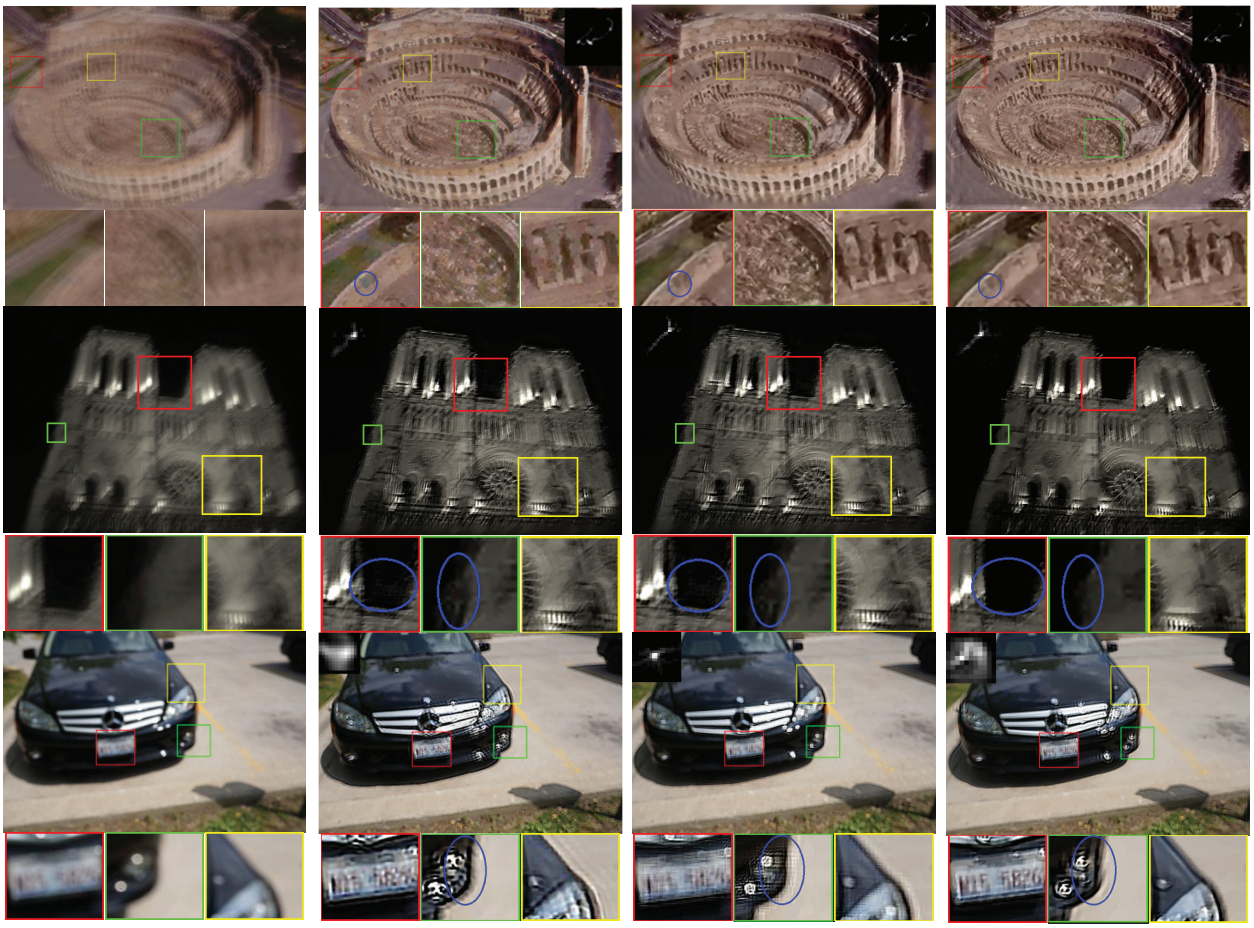


Figure 10: Example of deblurring results on Sun et al.'s dataset.

Figure 11: Example of deblurring results of our variants on Sun et al.'s dataset.

Figure 12: Example of deblurring results on Sun et al.'s dataset. Although the estimated kernels by all the methods are not accurate, our method is more visually plausible.



Blurry input | Xu & Jia [16] | Sun et al. [21] | Ours

Figure 13: Comparison of deblurring results for real camera motion blurry photographs.

[22] W. Zuo, D. Ren, S. Gu, L. Lin, and L. Zhang, "Discriminative learning of iteration-wise priors for blind deconvolution," in *CVPR*, 2015.

[23] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-laplacian priors," in *NIPS*, 2009.

[24] S. Fanello, C. Keskin, P. Kohli, S. Izadi, J. Shotton, A. Criminisi, U. Pattaccini, and T. Paek, "Filter forests for learning data-dependent convolutional kernels," in *CVPR*, 2014.

[25] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *CVPR*, 2014.

[26] C. J. Schuler, H. C. Burger, S. Harmeling, and B. Scholkopf, "A machine learning approach for non-blind image deconvolution," in *CVPR*, 2013.

[27] U. Schmidt, C. Rother, S. Nowozin, J. Jancsary, and S. Roth, "Discriminative non-blind deblurring," in *CVPR*, 2013.

[28] N. Komodakis and N. Paragios, "Mrf-based blind image deconvolution," in *Asian Conference on Computer Vision (ACCV)*, 2012.

[29] T. Michaeli and M. Irani, "Blind deblurring using internal patch recurrence," in *ECCV*, 2014.

[30] W.-S. Lai, J.-J. Ding, Y.-Y. Lin, and Y.-Y. Chuang, "Blur kernel estimation using normalized color-line priors," in *CVPR*, 2015.

[31] 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.

[32] H. Li, Y. Zhang, H. Zhang, Y. Zhu, and J. Sun, "Blind image deblurring based on sparse prior of dictionary pair," in *IEEE International Conference on Pattern Recognition (ICPR)*, 2012.

[33] J. Pan, Z. Hu, Z. Su, and M.-H. Yang, "Deblurring face images with exemplars," in *ECCV*, 2014.

[34] 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.

[35] Y. Yitzhaky, I. Mor, A. Lantzman, and N. Kopeika, "Direct method for restoration of motion-blurred images," *JOSA A*, vol. 15, no. 6, pp. 1512–1519, 1998.

[36] W. Hu, J. Xue, and N. Zheng, "Psf estimation via gradient domain correlation," *IEEE Transactions on Image Processing*, vol. 21, no. 1, pp. 386–392, 2012.

[37] G. Liu, S. Chang, and Y. Ma, "Blind image deblurring using spectral properties of convolution operators," *IEEE Transactions on Image Processing*, vol. 23, pp. 5047–5056, 2014.

[38] A. Goldstein and R. Fattal, "Blur-kernel estimation from spectral irregularities," in *ECCV*, 2012.

[39] T. Yue, S. Cho, J. Wang, and Q. Dai, "Hybrid image deblurring by fusing edge and power spectrum information," in *ECCV*, 2014.

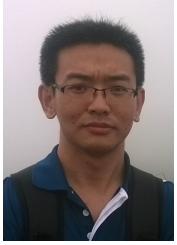
[40] D. Krishnan, J. Bruna, and R. Fergus, "Blind deconvolution with non-local sparsity reweighting," *arXiv preprint arXiv:1311.4029*, 2013.

[41] O. V. Michailovich, "An iterative shrinkage approach to total-variation image restoration," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1281–1299, 2011.

[42] D. Ren, H. Zhang, D. Zhang, and W. Zuo, "Fast total-variation based image restoration based on derivative augmented la-

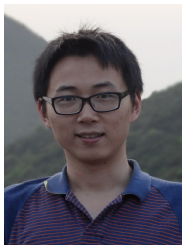
grangian method," *Neurocomputing*, 2015.
[43] L. Xu, X. Tao, and J. Jia, "Inverse kernels for fast spatial deconvolution," in *ECCV*, 2014.
[44] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," in *IEEE international conference on Acoustics, speech and signal processing (ICASSP)*, 2008.
[45] T. S. Cho, N. Joshi, C. L. Zitnick, S. B. Kang, R. Szeliski, and W. T. Freeman, "A content-aware image prior," in *CVPR*, 2010.
[46] M. Lai and J. Wang, "An unconstrained l_q minimization with $0 \leq q \leq 1$ for sparse solution of underdetermined linear systems," *SIAM Journal on Optimization*, vol. 21, no. 1, pp. 82–101, 2011.
[47] Y. She, "Thresholding-based iterative selection procedures for model selection and shrinkage," *Electronic Journal of Statistics*, vol. 3, pp. 384–415, 2009.
[48] L. Lin, X. Wang, W. Yang, and J.-H. Lai, "Discriminatively trained and-or graph models for object shape detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 5, pp. 959–972, 2015.
[49] M. Schmidt, "minfunc: unconstrained differentiable multivariate optimization in matlab," http://www.cs.ubc.ca/~schmidtm/Software/minFunc.html, 2005.
[50] D. Zoran and Y. Weiss, "Scale invariance and noise in natural images," in *ICCV*, 2009.
[51] 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*, vol. 13, no. 4, pp. 600–612, 2004.
[52] R. Köhler, M. Hirsch, B. Mohler, B. Schölkopf, and S. Harmeling, "Recording and playback of camera shake: Benchmarking blind deconvolution with a real-world database," in *ECCV*, 2012.
[53] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *ICCV*, 2011.

Wangmeng Zuo (M'09, SM'15) received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2007. From July 2004 to December 2004, from November 2005 to August 2006, and from July 2007 to February 2008, he was a Research Assistant at the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. From August 2009 to February 2010, he was a Visiting Professor in Microsoft Research Asia. He is currently a Professor in the School of Computer Science and Technology, Harbin Institute of Technology. His current research interests include discriminative learning, image modeling, low level vision, and biometrics. Dr. Zuo has published more than 50 papers in top tier academic journals and conferences including IJCV, IEEE T-IP, T-NNLS, T-IFS, CVPR, ICCV, ECCV, and NIPS. Dr. Zuo is a Senior Member of IEEE, an Associate Editor of the IET Biometrics, and the Guest Editors of Neurocomputing and Pattern Recognition.



Dongwei Ren received the B.Sc. degree in bioinfomatics and the M.Sc. degree in computer science and technology from the Harbin Institute of Technology, Harbin, China, in 2011 and 2013, respectively. He is currently working under the joint Ph.D. programmes in the School of Computer Science and Technology, Harbin Institute of Technology, and the Department of Computing, The Hong Kong Polytechnic University. From November 2012 to January 2013 and from April 2014 to August 2014, he was a Research Assistant at the Department of Computing, The Hong Kong Polytechnic University. His research interests include low level vision and discriminative learning.

David Zhang (F'08) graduated in Computer Science from Peking University. He received his M.Sc. in 1982 and his Ph.D. in 1985 in both Computer Science from the Harbin Institute of Technology (HIT), respectively. From 1986 to 1988 he was a Postdoctoral Fellow at Tsinghua University and then an Associate Professor at the Academia Sinica, Beijing. In 1994 he received his second Ph.D. in Electrical and Computer Engineering from the University of Waterloo, Ontario, Canada. Currently, he is a Chair Professor at the Hong Kong Polytechnic University where he is the Founding Director of Biometrics Research Centre (UGC/CRC) supported by the Hong Kong SAR Government in 1998. He also serves as Visiting Chair Professor in Tsinghua University and HIT, and Adjunct Professor in Shanghai Jiao Tong University, Peking University, National University of Defense Technology and the University of Waterloo. He is the Founder and Editor-in-Chief, International Journal of Image and Graphics (IJIG); Book Editor, Springer International Series on Biometrics (KISB); Organizer, the first International Conference on Biometrics Authentication (ICBA); Associate Editor of more than ten international journals including IEEE Transactions; Technical Committee Chair of IEEE SMC and the author of more than 10 books and 350 international journal papers. He has been listed as a Highly Cited Researchers in Engineering by Thomos Reters in 2014 and in 2015, respectively. Professor Zhang is a Croucher Senior Research Fellow, Distinguished Speaker of the IEEE Computer Society, and a Fellow of International Association for Pattern Recognition (IAPR).



Shuhang Gu received the BE degree from the school of astronautics, Beijing University of Aeronatics and Astronautics, China, in 2010, and the ME degree from the institute of pattern recognition and artificial intelligence, Huazhong University of Science and Technolodgy, China, in 2013. He is currently working toward the PhD degree in the Department of Computing at the Hong Kong Polytechnic University. His research interests include image restoration, sparse and low rank models.

Lei Zhang (M'04, SM'14) received the B.Sc. degree in 1995 from Shenyang Institute of Aeronautical Engineering, Shenyang, P.R. China, the M.Sc. and Ph.D degrees in Control Theory and Engineering from Northwestern Polytechnical University, Xi'an, P.R. China, respectively in 1998 and 2001. From 2001 to 2002, he was a research associate in the Dept. of Computing, The Hong Kong Polytechnic University. From Jan. 2003 to Jan. 2006 he worked as a Postdoctoral Fellow in the Dept. of Electrical and Computer Engineering, McMaster University, Canada. In 2006, he joined the Dept. of Computing, The Hong Kong Polytechnic University, as an Assistant Professor. Since July 2015, he has been a Full Professor in the same department. His research interests include Computer Vision, Pattern Recognition, Image and Video Processing, and Biometrics, etc. Prof. Zhang has published more than 200 papers in those areas. By 2016, his publications have been cited more than 17,000 times in literature. Prof. Zhang is currently an Associate Editor of IEEE Trans. on Image Processing, IEEE Trans. on CSVT and Image and Vision Computing. He was selected as the Highly Cited Researcher by Thomson Reuters. More information can be found in his homepage http://www4.comp.polyu.edu.hk/ cslzhang/.