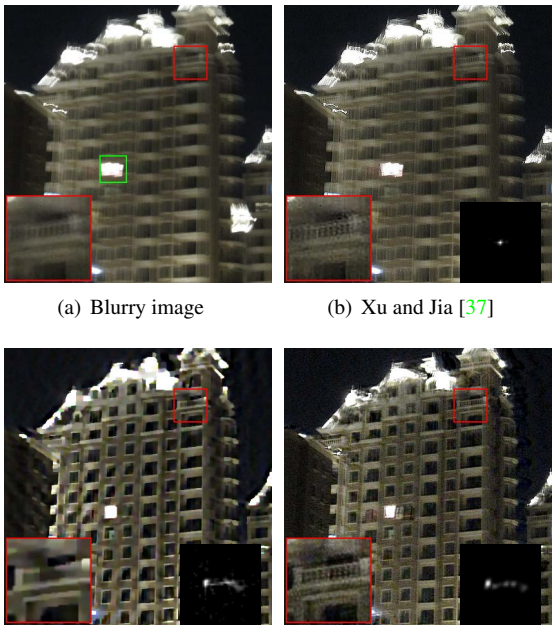
Self-paced Kernel Estimation for Robust Blind Image Deblurring

Dong Gong[†‡], Mingkui Tan[§], Yanning Zhang[†], Anton van den Hengel[‡], Qinfeng Shi[‡*]
[†]School of Computer Science and Engineering, Northwestern Polytechnical University, China
[‡]School of Computer Science, The University of Adelaide, Australia
[§]School of Software Engineering, South China University of Technology, China
edgong01@gmail.com, mingkuitan@scut.edu.cn, ynzhang@nwpu.edu.cn,
{anton.vandenhengel,javen.shi}@adelaide.edu.au

Abstract

The challenge in blind image deblurring is to remove the effects of blur with limited prior information about the nature of the blur process. Existing methods often assume that the blur image is produced by linear convolution with additive Gaussian noise. However, including even a small number of outliers to this model in the kernel estimation process can significantly reduce the resulting image quality. Previous methods mainly rely on some simple but unreliable heuristics to identify outliers for kernel estimation. Rather than attempt to identify outliers to the model a priori, we instead propose to sequentially identify inliers, and gradually incorporate them into the estimation process. The self-paced kernel estimation scheme we propose represents a generalization of existing self-paced learning approaches, in which we gradually detect and include reliable inlier pixel sets in a blurred image for kernel estimation. Moreover, we automatically activate a subset of significant gradients w.r.t. the reliable inlier pixels, and then update the intermediate sharp image and the kernel accordingly. Experiments on both synthetic data and real-world images with various kinds of outliers demonstrate the effectiveness and robustness of the proposed method compared to the state-of-the-art methods.



(a) Blurry image (b) Xu and Jia [37]

(c) Pan *et al.* [26] (d) Ours

Figure 1. Deblurring the image with outliers. (a) A real blurry image with outliers, containing saturated regions (*e.g.* pixels in the green box) and unknown noise. (b)-(d) Deblurring results of Xu and Jia [37], Pan *et al.* [26] and the proposed method, respectively.

1. Introduction

Image blur is a ubiquitous problem in image capture due to variety factors such as camera movement [8, 6] and out-of-focus effects [20]. The increasing usage of light-weight hand-held photography devices (*e.g.* cell-phones) has significantly increased the demand for practical image deblurring, especially when facing imperfect imaging situations, such as selfies.

To deal with the image blur, a classical way is to adopt a spatially invariant linear blur kernel $\mathbf{k}$ [8, 35], and then model the blurred image $\mathbf{y}$ by a convolution process on the latent sharp image $\mathbf{x}$ with the blur kernel $\mathbf{k}$:

$$\mathbf{y} = \mathbf{x} * \mathbf{k} + \mathbf{e}\,, \tag{1}$$

where $*$ denotes the convolution operator, and $\mathbf{e}$ refers to additive noise. Here $\mathbf{y} \in \mathbb{R}^n$, $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{k} \in \mathbb{R}^m$ and $\mathbf{e} \in \mathbb{R}^n$. Note that the blur kernel is often unknown. Thus, *blind image deblurring* aims to estimate both $\mathbf{x}$ and $\mathbf{k}$ from

*This work was supported by NSFC (61231016, 61572405, 61602185), China 863 Program (2015AA016402), ARC (DP160100703). M. Tan is supported by the Recruitment Program for Young Professionals, China

2380-7504/17 $31.00 © 2017 IEEE
DOI 10.1109/ICCV.2017.184

the observation $\mathbf{y}$, which is a highly ill-posed problem. In practice, the challenge is often addressed by solving a non-convex optimization problem [35, 9]. Most existing blind image deblurring methods [8, 6, 25, 35, 9] assume that the noise $\mathbf{e}$ in (1) follows an i.i.d. zero-mean Gaussian distribution. However, these approaches often generate visual artifacts in the deblurred image, as the model (1) and the i.i.d. Gaussian assumption are too restrictive to model real blurred images.

In general, outliers to the above model arise primarily [7] from non-Gaussian noise (*e.g.* impulse noise) [2], saturation [7, 33], and nonlinear camera response functions (CRF) [32]. Non-blind deconvolution methods, which assume the kernel $\mathbf{k}$ is known, have been developed recently [7, 33] to deal with outliers, but blind image deconvolution, specifically kernel estimation, for such data has received far less attention. For blind image deblurring, although the influence of the nonlinear CRF on kernel estimation can be avoided through preprocessing [32], saturation and non-Gaussian noise cannot be so easily addressed [26] (see Figure 1). Existing kernel estimation methods have attempted to address the problem of outliers to the model via simple preprocessing techniques [33], and/or loss functions that are claimed to be robust to specific noise distributions [7, 26, 37] (*e.g.* ℓ_1-loss corresponding to Laplacian distribution). Others have defined heuristics for identifying and removing problematic areas of images [13, 9], but these methods typically only address a specific cause of such noise, and fail to generalize.

As with general parameter estimation problems, fitting the model to data with outliers typically results in a grossly distorted estimate. However, identifying outliers a priori is difficult, which requires sorting out and analyzing the inliers. We thus propose here an extension of the self-paced learning (SPL) for deblurring images with outliers, which attempts to identify the confident inliers first.

The proposed self-paced kernel estimation method gradually includes pixels into the model estimation, starting with the obvious inliers, and moving on to the more challenging decisions. While doing so, it also encourages a graph-based correlation among the selected pixels. According to the selected pixels, a gradient activation method is used to obtain the intermediate image $\mathbf{x}$ with only the most significant components. The kernel $\mathbf{k}$ is estimated using only the selected inliers, and thus is unlikely to be affected by outliers. In this paper, we make the following contributions:

- In contrast to existing methods that rely on heuristic outlier removal processes or loss functions based on specific outlier characterisations, we instead introduce a robust kernel estimation scheme that incrementally includes inlier pixels that are faithful to the model, by the model itself.
- We propose a novel self-paced kernel estimation scheme by binding the self-paced learning regime with a gradient activation scheme. This approach automatically activates the significant gradients that are beneficial for kernel estimation on the selected inlier pixels.
- We propose a generalization to self-paced learning that encourages the incorporation among the selected samples. For deblurring, we propose a graph-based regularizer encouraging the selected inliers falling within a neighborhood, since the connected pixels are more beneficial than the isolated ones for kernel estimation.

2. Related work

Notation. Let $\mathbf{A} = [A_{i,j}] \in \mathbb{R}^{m\times n}$ and $\mathbf{v} = [v_1, ..., v_n]^\mathsf{T} \in \mathbb{R}^n$ denote a matrix and a vector, respectively, where $^\mathsf{T}$ denotes the transpose of a vector/matrix. Let $\mathbf{0}$ and $\mathbf{1}$ be vectors with all zeros and all ones, respectively. Let $\mathbf{I}$ be the identity matrix. Let $\odot$ and $\otimes$ denote the element-wise product and Kronecker product, respectively. Given a positive integer n, let $[n] = \{1, ..., n\}$. Given any index set $\mathcal{T} \subseteq [n]$, let $\mathcal{T}^c$ be the complementary set of $\mathcal{T}$, i.e. $\mathcal{T}^c = [n] \setminus \mathcal{T}$.For a vector $\mathbf{v} \in \mathbb{R}^n$, let v_i denote the i-th element of $\mathbf{v}$, and $\mathbf{v}_\mathcal{T}$ denote the subvector indexed by $\mathcal{T}$. Let $\mathrm{diag}(\mathbf{v})$ be a diagonal matrix with diagonal elements equal to the vector $\mathbf{v}$. Let $\|\mathbf{v}\|_p$ be the ℓ_p-norm. Let $\|\mathbf{v}\|_\mathbf{M}$ be the weighted ℓ_2 norm defined as $\|\mathbf{v}\|^2_\mathbf{M} = \mathbf{v}^\mathsf{T}\mathbf{M}\mathbf{v}$ with $\mathbf{M} \in \mathbb{R}^{n\times n}$.

Conventional blind image deblurring. To address the fact that the general blind image deblurring is ill-posed, many works focus on investigating image priors and regularizers. A series of widely-used priors and regularizers are based on image gradient sparsity, such as the total variational regularizer [5, 29], the Gaussian scale mixture prior [8], the ℓ_1/ℓ_2-norm based prior [17], and the ℓ_0-norm regularizer [39, 25]. Non-gradient-based priors have also been proposed, such as the edge-based patch prior [30], the color line based prior [19], and the dark channel prior [27]. Various estimation strategies have also been proposed for effective kernel estimation, such as edge-extraction based maximum-a-posterior (MAP) [6], gradient activation based MAP [9], and variational Bayesian methods [22, 1, 40]. Although these methods generally work well, their performance degrades when input images are contaminated by outliers.

Image deblurring considering outliers. The influence of outliers on image deblurring is attracting increasing attention [7, 13, 25, 33, 38], but most methods focus on non-blind deconvolution [7, 37, 33]. Harmeling *et al.* [12] proposed a multi-frame deblurring method which detects saturated pixels using a thresholding operation, for example. Xu and Jia [37] improve robustness to non-Gaussian noise using an ℓ_1-loss function. Cho *et al.* [7] proposed an expectation-maximization algorithm to iteratively detect outliers and recover images specifically for non-blind deconvolution. Whyte *et al.* [33] proposed a Richardson-Lucy

[23] based deconvolution method to alleviate the ringing caused by saturation. Xu *et al.* [38] applied a deep neural network to restore blurred images with saturated pixels.

For blind image deblurring, some methods exclude the outliers via thresholding the blurred image [33, 9]. This heuristic approach has a range of problems, including sensitivity to the threshold value. Hu *et al.* [13] address low-light images with saturation specifically by detecting the light streaks explicitly. Pan *et al.* [25] proposed an ℓ_0-norm based text image prior, which can also be used for saturated images, but is impractical for handling non-Gaussian noise. The fact that the nonlinearity in the CRF violates the model in (1) was addressed by Tai *et al.* [32] which proposed to estimate the CRF from the blurred image itself.

The closest approach to that proposed here is an iterative energy minimization method proposed by Pan *et al.* [26], which removes the outliers from selected edges at each iteration through an empirically designed outlier identification strategy.

Self-paced learning. Inspired by the learning process of humans [3, 18], self-paced learning (SPL) was originally proposed to learn a latent variable model by focussing first the *easy* training data, where *a set of samples is easy if it admits a good fit in the model space* [18]. SPL has since been applied to a range of problems including visual tracking [31], event detection [14] and matrix factorization [42]. It is of particular interest here that Jiang *et al.* [14] proposed selecting samples which are diverse, as well as easy.

3. Robust kernel estimation model

3.1. Blur model with outliers

Given a sharp image $\mathbf{x}$ and a blur kernel $\mathbf{k}$, previous works [26, 7] have modeled the blurred image as $\mathbf{y} = f(\mathbf{x} * \mathbf{k}) + \mathbf{e}_{\text{nG}}$, where $f(\cdot)$ is a nonlinear function which produces the outliers and $\mathbf{e}_{\text{nG}}$ denotes noise with a non-Gaussian distribution [37]. For example, for the case of saturation, $f(\mathbf{x} * \mathbf{k})$ clips the values to lie within the range that the image representation can record (typically $0 - 255$) [7]. However, the formulation of $f(\cdot)$ is usually unknown, and analytically estimating it is difficult.

We propose here, in contrast, an additive model:

$$\mathbf{y} = \mathbf{k} * \mathbf{x} + \mathbf{e} + \mathbf{o}, \tag{2}$$

where $\mathbf{e}$ represents the Gaussian noise, and $\mathbf{o}$ represents a term producing the "outliers", or the noise not modelled by $\mathbf{e}$. For example, if saturation is the only resource of outliers, $\mathbf{o}$ implicitly absorbs a term $f(\mathbf{x} * \mathbf{k}) - \mathbf{x} * \mathbf{k}$ corresponding to saturation, which is unknown. This additive model makes separately treating the Gaussian noise $\mathbf{e}$ and the outlier term $\mathbf{o}$ possible. Following [35, 9], the above model can be equivalently represented as

$$\mathbf{y} = \mathbf{H}(\mathbf{k})\mathbf{x} + \mathbf{e} + \mathbf{o} = \mathbf{A}(\mathbf{x})\mathbf{k} + \mathbf{e} + \mathbf{o}, \tag{3}$$

where $\mathbf{H}(\mathbf{k}) \in \mathbb{R}^{n\times n}$ and $\mathbf{A}(\mathbf{x}) \in \mathbb{R}^{n\times m}$ are convolution matrices associated with $\mathbf{k}$ and $\mathbf{x}$, respectively.

3.2. Self-paced kernel estimation model

In kernel estimation, the Gaussian noise term $\mathbf{e}$ in (3) can be easily handled via ℓ_2-loss (*i.e.* least square loss). For $\mathbf{o}$, we often do not have prior knowledge regarding the distribution over it. However, it is reasonable to assume $\mathbf{o}$ is sparse, since, usually only a small portion of pixels are significantly contaminated. Motivated by this, we introduce a binary vector $\mathbf{v} \in \{0,1\}^n$ to index the pixels, where 0 is to indicate a possible outlier and excludes the outlier from kernel estimation, and 1 is to indicate a selected inlier. Following this, we can formulate the kernel estimation problem as a *self-paced learning* problem. Let the squared ℓ_2-norm be the regularizer of the blur kernel $\mathbf{k}$, and $\mathbf{k} \in \mathcal{K} = \{\mathbf{k} \mid \|\mathbf{k}\|_1 = 1, k_i \geq 0, \forall i\}$. We can model kernel estimation as an SPL problem:

$$\min_{\substack{\mathbf{k}\in\mathcal{K},\\ \mathbf{x},\mathbf{v}\in\Upsilon}} \frac{1}{2}\sum_{i=1}^{n} v_i(y_i - [\mathbf{x}*\mathbf{k}]_i)^2 + \lambda\Omega(\mathbf{x}) + \frac{\gamma}{2}\|\mathbf{k}\|_2^2 + \rho\Upsilon(\mathbf{v}), \tag{4}$$

where λ, γ and ρ are non-negative regularization weights, $\Omega(\cdot)$ denotes the regularizer for $\mathbf{x}$ and $\Upsilon(\cdot)$ is the self-paced regularizer controlling the sample selection pace [18, 14].

Defining $\Upsilon(\mathbf{v})$. By solving problem (4) given an appropriate ρ, the SPL scheme will gradually incorporate the "easy" pixel samples into the estimation process. Considering the ill-posed nature of the problem, and the correlations between pixels corresponding to the same underlying content (*e.g.* edges, flat areas, *etc* .), we also restrict pixel selection by defining the domain of $\mathbf{v}$ based on an underlying *weighted graph* $G = (V, E)$ [11]. We define G on a lattice structure on the elements of $\mathbf{v}$, *i.e.* letting $V = [n]$ and letting E represent the 2D neighborhood relationship among v_i's, and denote the edge weights with $c : E \to \mathbb{R}_0^+$. In our implementation, we let $c(e) = 1, \forall e \in E$ to indicate the distances between the neighbor vertices on the graph. With this definition, we can restrict $\mathbf{v}$ in a graph structured domain [11] by identifying its support with a *forest* $F = (V_F, E_F)$ (a union of individual trees) as

$$\Psi = \{\mathbf{v} \in \{0,1\}^n | \text{supp}(\mathbf{v}) = V_F, F \subseteq G, \phi(F) = b\}, \tag{5}$$

where V_F denotes the vertices of the forest F, $\phi(F)$ denotes that the number of connected components of F and b is a positive integer value. Because $F \subseteq G$, supp$(\mathbf{v})$ should be the vertices of a *subgraph* of G. Let $g(\mathbf{v})$ denote the total weight of edges in the forest F corresponding to supp$(\mathbf{v})$. Specifically, $g(\mathbf{v}) = \sum_{e\in E_F} c(e)$, when supp$(\mathbf{v}) = V_F$. We define a novel self-paced regularizer as

$$\Upsilon(\mathbf{v}) = -\varrho\|\mathbf{v}\|_1 + \alpha g(\mathbf{v}), \tag{6}$$

where $-\|\mathbf{v}\|_1$ is the easiness term from previous SPL studies [14, 18], $g(\mathbf{v})$ is referred to as the connective term (as it encourages connectedness among supp($\mathbf{v}$)), and α and ϱ are positive regularizer parameters ($\varrho = 2$ and $\alpha = 0.5$ are used in this paper). The negative ℓ_1-norm controls the pace at which pixels are included in kernel estimation, and favors selecting easy pixels at first. In detail, if $\alpha = 0$, given fixed $\mathbf{x}$ and $\mathbf{k}$, the conventional SPL scheme tends to select pixels with $\frac{1}{2}(y_i - [\mathbf{x} * \mathbf{k}]_i)^2 \leq \lambda\varrho$ [18]. Together with the graph guided feasible domain Ψ, minimizing $g(\mathbf{v})$ helps to trim the isolated components in $\mathbf{v}$ and make the selected inliers tend to flock together.

Regularization on $\mathbf{x}$. For kernel estimation, focusing on a subset of gradients of $\mathbf{x}$ alleviates problems caused by fine-grained textures [37, 25, 9] and increases the accuracy on the estimated $\mathbf{k}$. As the gradient activation scheme has shown its effectiveness at kernel estimation in [9], we use this scheme to update the intermediate image $\mathbf{x}$. Following the technique in [10], we directly perform the gradient activation on image $\mathbf{x}$ instead of the image gradients [9], which is helpful for identifying the outliers.

Let $\mathbf{Dx} \in \mathbb{R}^{2n}$ be the gradients of $\mathbf{x}$[1]. Instead of encouraging the sparsity on gradients directly by letting $\Omega(\mathbf{x}) = \|\mathbf{Dx}\|_1$, we activate the nonzero gradients in $\mathbf{Dx}$ while solving $\mathbf{x}$ by introducing an auxiliary variable $\mathbf{z} \in \mathbb{R}^{2n}$ and a binary indicator $\mathbf{w} \in \{0,1\}^{2n}$, and letting $\mathbf{Dx} = \mathbf{z} \odot \mathbf{w}$. Moreover, to encourage the sparsity of the activated gradients, we define a feasible domain of $\mathbf{w}$ as $\Lambda = \{\mathbf{w} \mid \|\mathbf{w}\|_0 \leq \kappa, \mathbf{w} \in \{0,1\}^{2n}\}$, where the integer κ controls the number of the zero elements in $\mathbf{z}$.

Self-paced kernel estimation. Based on the regularizations on $\mathbf{x}$ and $\mathbf{v}$ introduced above, we reformulate problem (4) as the following self-paced kernel estimation problem:

$$\min_{\substack{\mathbf{k},\mathbf{v}\in\Psi \\ \mathbf{x},\mathbf{w}\in\Lambda,\mathbf{z}}} \frac{1}{2}\sum_{i=1}^{n} v_i(y_i - [\mathbf{x} * \mathbf{k}]_i)^2 + \lambda\|\mathbf{z}\|_1 + \frac{\gamma}{2}\|\mathbf{k}\|_2^2 + \rho\Upsilon(\mathbf{v}), \quad \text{s.t. } \mathbf{Dx} = \mathbf{z} \odot \mathbf{w},\ \mathbf{k} \in \mathcal{K}, \tag{7}$$

where $\lambda\|\mathbf{z}\|_1$ is used to reduce the ill-posedness.

4. Self-paced robust kernel estimation

As the SPL [18, 14], we address problem (7) using an alternating search strategy (ASS), that is, by alternatively optimizing $\{\mathbf{x}, \mathbf{w}\}$, $\mathbf{k}$ and $\mathbf{v}$, while fixing the other variables. The kernel estimation problem (7) can thus be solved by addressing the subproblems introduced below.

4.1. Estimating the intermediate image $\mathbf{x}$

Since $\mathbf{x}$ and $\mathbf{w}$ are coupled, we estimate the intermediate image $\mathbf{x}$ (for fixed $\mathbf{k}$ and $\mathbf{v}$) by optimizing $\mathbf{x}$ and $\mathbf{w}$ jointly by solving the subproblem:

$$\min_{\mathbf{w}\in\Lambda}\min_{\mathbf{x},\boldsymbol{\xi},\mathbf{z}} \frac{1}{2}\|\boldsymbol{\xi}\|_2^2 + \lambda\|\mathbf{z}\|_1, \quad \text{s.t. } \boldsymbol{\xi} = \text{diag}(\mathbf{v})(\mathbf{y} - \mathbf{Hx}),\ \mathbf{Dx} = \mathbf{z} \odot \mathbf{w}, \tag{8}$$

where $\boldsymbol{\xi} = \text{diag}(\mathbf{v})(\mathbf{y} - \mathbf{Hx})$ denotes the fitting error given the indicator vector $\mathbf{v}$. In (8), κ reflects a rough knowledge of the sparsity of $\mathbf{Dx}$, and there are $|\Lambda| = \sum_{i=0}^{\kappa}\binom{2n}{i}$ feasible $\mathbf{w}$'s in Λ. Problem (8) tends to find an optimal $\mathbf{w}$ from Λ, which minimizes the regression loss $\|\boldsymbol{\xi}\|_2^2$ by constraining $\mathbf{x}$. The problem in (8) is hard to solve since it is a mixed integer programming problem. We address it by relaxing it to a convex quadratically constrained linear programming (QCLP) [28] problem:

$$\min_{\boldsymbol{\alpha}\in\mathcal{A},\theta\in\mathbb{R}} \theta, \ \text{ s.t. } \phi(\boldsymbol{\alpha}, \mathbf{w}) \leq \theta, \forall \mathbf{w} \in \Lambda, \tag{9}$$

where θ is an auxiliary variable, $\phi(\boldsymbol{\alpha}, \mathbf{w}) = -\frac{1}{2}\|\boldsymbol{\alpha}\|_2^2 + \boldsymbol{\alpha}^\mathsf{T}\mathbf{y}, \boldsymbol{\alpha} \in \mathcal{A}_\mathbf{w}$ and $\mathcal{A} = \cap_{\mathbf{w}\in\Lambda}\mathcal{A}_\mathbf{w}$ with $\mathcal{A}_\mathbf{w} = \{\boldsymbol{\alpha} | \mathbf{H}^\mathsf{T}\text{diag}(\mathbf{v})\boldsymbol{\alpha} = \mathbf{D}^\mathsf{T}\boldsymbol{\beta}, \|\text{diag}(\mathbf{w})\boldsymbol{\beta}\|_\infty \leq \lambda, \boldsymbol{\alpha} \in [-h, h]^n\}$. $\boldsymbol{\alpha} \in \mathbb{R}^n$ and $\boldsymbol{\beta} \in \mathbb{R}^{2n}$ refer to the Lagrangian dual variables w.r.t. the two constraints in (8), respectively. There is $\boldsymbol{\alpha} = \boldsymbol{\xi}^* = \text{diag}(\mathbf{v})(\mathbf{y} - \mathbf{Hx}^*)$ at the optimality of the inner problem w.r.t. $\mathbf{x}$, $\boldsymbol{\xi}$ and $\mathbf{z}$.

There are $T = \sum_{i=1}^{\kappa}\binom{2n}{i}$ constraints in problem (9), making the problem difficult to address directly. However, most constraints are inactive at the optimum as only a subset of nonzero elements in $\mathbf{Dx}$ are significant for kernel estimation. We thus seek to address problem (9) using a cutting-plane method [24, 10], which iteratively detects the most violated constraint and solves a subproblem w.r.t. the active constraints. The overall cutting-plane method is shown in Algorithm 1.

The most-violated constraints selection reflects the activation of the appropriate gradients in $\mathbf{Dx}$. According to the definition of $\phi(\boldsymbol{\alpha}, \mathbf{w})$, similar to [10], we first obtain $\boldsymbol{\beta}$ by solving

$$\min_{\boldsymbol{\beta}} \frac{1}{2}\|\mathbf{D}^\mathsf{T}\boldsymbol{\beta} - \mathbf{H}^\mathsf{T}\text{diag}(\mathbf{v})\boldsymbol{\alpha}\|_2^2 + \frac{r}{2}\|\boldsymbol{\beta}\|_2^2, \tag{10}$$

where the regularizer $\frac{r}{2}\|\boldsymbol{\beta}\|_2^2$ with $r \geq 0$ is for reducing ill-posedness. In practice, we set $r = 2$ and solve problem (10) via a conjugate gradient (CG) algorithm. The most-violated constraint can then be detected by choosing the κ largest $|\beta_i|$ and recording their indices into a set $\mathcal{C}^t$. Following that, we update the active gradient set $\mathcal{S}^t = \mathcal{S}^{t-1} \cup \mathcal{C}^t$, and update $\mathbf{x}$ by solving a convex equality-constrained problem which is the primal representation of (9) with the active $\mathbf{w}$'s:

$$\min_{\mathbf{x},\mathbf{z}_\mathcal{S}} \frac{1}{2}\|\mathbf{y} - \mathbf{Hx}\|^2_{\text{diag}(\mathbf{v})} + \lambda\|\mathbf{z}_\mathcal{S}\|_1 \quad \text{s.t. } (\mathbf{Dx})_\mathcal{S} = \mathbf{z}_\mathcal{S}, (\mathbf{Dx})_{\mathcal{S}^c} = \mathbf{0}. \tag{11}$$

[1] $\mathbf{D} = [\mathbf{D}_v^\mathsf{T}, \mathbf{D}_h^\mathsf{T}]^\mathsf{T} \in \{-1, 0, 1\}^{2n\times n}$, where $\mathbf{D}_v$ and $\mathbf{D}_h$ denote the convolution matrices for vertical and horizontal directions, respectively.

As shown in (11), we focus on the activated gradients (*i.e.* $(\mathbf{D}\mathbf{x})_{\mathcal{S}}$) and restrict the inactive gradients to be 0 while solving for $\mathbf{x}$. We address problem (11) using an alternating direction method of multipliers (ADMM) [4] approach. Note that, following [10], we set κ as the number of elements in $\boldsymbol{\beta}^0$ whose absolute values are larger than $0.5 \times \|\boldsymbol{\beta}^0\|_\infty$.

Algorithm 1: Cutting planes for updating $\mathbf{x}$

Input: Blurry image $\mathbf{y}$, $\mathbf{H}$, $\mathbf{v}$ and λ.

Initialize $\mathbf{x}^0 = \mathbf{1} \otimes (\mathbf{1}^\top \mathbf{y}/n)$, $\mathcal{S}^0 = \emptyset$, $\boldsymbol{\xi}^0 = \text{diag}(\mathbf{v})(\mathbf{y} - \mathbf{H}\mathbf{x}^0)$, and iteration index $t = 0$;

while *Stopping conditions are not achieved* **do**

  Find the most violated constraint: Compute $\mathbf{H}^\top \text{diag}(\mathbf{v})\boldsymbol{\alpha}^t$, get $\boldsymbol{\beta}^t$ via solving problem (10), and record the indices of κ largest $|\beta_j^t|$ into $\mathcal{C}^{t+1}$;

  Let $\mathcal{S}^{t+1} = \mathcal{S}^t \cup \mathcal{C}^{t+1}$;

  Subproblem solving: Solve subproblem (11) using ADMM algorithm, obtaining $\mathbf{x}^{t+1}$;

  Let $\boldsymbol{\alpha}^{t+1} = \text{diag}(\mathbf{v})(\mathbf{y} - \mathbf{H}\mathbf{x}^{t+1})$, and $t = t + 1$;

4.2. Estimating the blur kernel $\mathbf{k}$

Given a fixed $\mathbf{v}$ and a fixed $\mathbf{x}$, the kernel estimation subproblem is $\min_{\mathbf{k}\in\mathcal{K}} \frac{1}{2}\|\mathbf{y} - \mathbf{x} * \mathbf{k}\|^2_{\text{diag}(\mathbf{v})} + \frac{\gamma}{2}\|\mathbf{k}\|_2^2$, which is a weighted least square problem. Since estimating such kernels from image gradients has been shown to be more accurate [6, 39, 27], we let $\nabla\mathbf{y}$ and $\nabla\mathbf{x}$ denote the gradients of blurry image $\mathbf{y}$ and the given $\mathbf{x}$, respectively, and address the kernel estimation by solving the following quadratic problem:

$$\min_{\mathbf{k}} \frac{1}{2}\|\nabla\mathbf{y} - \mathbf{A}(\nabla\mathbf{x})\mathbf{k}\|^2_{\text{diag}(\mathbf{v})} + \frac{\gamma}{2}\|\mathbf{k}\|_2^2, \tag{12}$$

where $\mathbf{A}(\nabla\mathbf{x})$ denotes the convolution matrix of the gradient image $\nabla\mathbf{x}$. Problem (12) is quadratic with respect to $\mathbf{k}$, which can be effectively addressed using the CG algorithm. After obtaining $\mathbf{k}$, we then set the negative elements k_i as 0, and let $\mathbf{k} = \mathbf{k}/\|\mathbf{k}\|_1$, which ensures $\mathbf{k} \in \mathcal{K}$.

4.3. Updating the pixel selection vector $\mathbf{v}$

By fixing $\mathbf{x}$ and $\mathbf{k}$ and substituting $\Upsilon(\mathbf{v})$ into (7), we update the binary selection vector $\mathbf{v}$ by solving the subproblem:

$$\min_{\mathbf{v}\in\Psi} \frac{1}{2}\sum_{i=1}^{n} v_i(y_i - [\mathbf{x} * \mathbf{k}]_i)^2 - \rho\varrho\|\mathbf{v}\|_1 + \rho\alpha g(\mathbf{v}). \tag{13}$$

Let $\mathbf{l} = \frac{1}{2}(\mathbf{y} - \mathbf{H}\mathbf{x}) \odot (\mathbf{y} - \mathbf{H}\mathbf{x})$ denote the fitting error. After reformulating (13), we address for $\mathbf{v}$ equivalently by solving the following problem:

$$\min_{\mathbf{v}\in\Psi} \sum_{i=1}^{n} (\rho\varrho - l_i)(1 - v_i) + \rho\alpha g(\mathbf{v}). \tag{14}$$

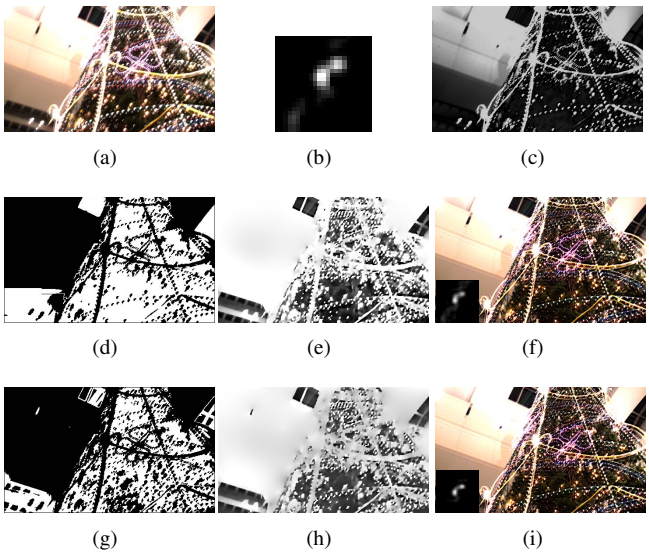


Figure 2. Effectiveness of the proposed self-paced regularizer $\Upsilon(\mathbf{v})$. (a) Blurred image $\mathbf{y}$. (b) Initialization of the blur kernel $\mathbf{k}$. (c) The loss map $\mathbf{l}$ generated using the input $\mathbf{y}$, the initialization of $\mathbf{k}$ and an $\mathbf{x}$ initialized as $\mathbf{1} \otimes (\mathbf{1}^\top \mathbf{y}/n)$. (d) Estimated $\mathbf{v}$ by solving the (14) with $\Upsilon(\mathbf{v})$ and the $\mathbf{l}$ in (c). (e) The estimated intermediate $\mathbf{x}$ given the $\mathbf{v}$ in (d). (f) Estimated $\mathbf{k}$ and the final recovered image with $\Upsilon(\mathbf{v})$. (g)-(i) show the results similar to (d)-(f) but with only $-\|\mathbf{v}\|_1$ as the self-paced regularizer.

Detailed derivations are left in the supplementary material.

Solving problem (14) can be achieved by finding a subforest $F = (V_F, E_F)$ in G such that the total weight $g(\mathbf{v}) = \sum_{e\in E_F} c(e)$ is minimized and $(\rho\varrho - l_i)v_i$ is maximized. After obtaining F, the solution $\mathbf{v}$ can be calculated by letting $v_i = 1$ if $i \in V_F$ and $v_i = 0$ for others.

Problem (13) can be equivalently treated as a *prize-collecting Steiner forest (PCSF)* problem [11], since $\rho\varrho - l_i, \forall i \in V$ can be seen as the *node prize* and the edge weight $c(e), \forall e \in E$ can be considered as the *edge cost*, w.r.t. the graph G. The PCSF problem seeks to find a forest containing multiple minimum-cost spanning trees where the prizes on the nodes are maximized. Although the PCSF problem is NP-hard, we can address it efficiently using a solver in [11], relying on an iterative clustering scheme named Goemans-Williamson scheme [15]. As shown in Figure 2, the proposed graph guided self-paced regularizer $\Upsilon(\mathbf{v})$ and the corresponding PCSF based solver tend to select connected pixels instead of independently selecting the elements with $l_i \leq \rho$ as the conventional SPL method [18], which leads to better intermediate image and more accurate results.

4.4. Self-paced algorithm for kernel estimation

Based on the algorithms for the subproblems of updating $\mathbf{x}$, $\mathbf{k}$ and $\mathbf{v}$, we summarize the alternative self-paced kernel estimation method in Algorithm 2. To improve the robustness, we implement the kernel estimation using a coarse-to-

Algorithm 2: Self-paced kernel estimation

Input: Blurry image $\mathbf{y}$, λ, γ, $\eta > 1$, $\rho_{\max}$.

1 Initialize $\mathbf{k}^0$ and $\mathbf{x}^0$ and ρ^0;

2 **for** $k = 1 : K$ **do**

3 Calculate $\mathbf{l}^k = \frac{1}{2}(\mathbf{y} - \mathbf{H}\mathbf{x}^{k-1}) \odot (\mathbf{y} - \mathbf{H}\mathbf{x}^{k-1})$, and update $\mathbf{v}^k$ by solving the PCSF problem (14);

4 Given $\mathbf{v}^k$ and $\mathbf{k}^{k-1}$, update $\mathbf{x}^k$ by solving problem (8) via Algorithm 1;

5 Estimate $\mathbf{k}^k$ according to problem (12);

6 Update pace parameter $\rho^k = \min\{\eta\rho^{k-1}, \rho^{\min}\}$.

fine scheme following the image pyramid [8, 39].

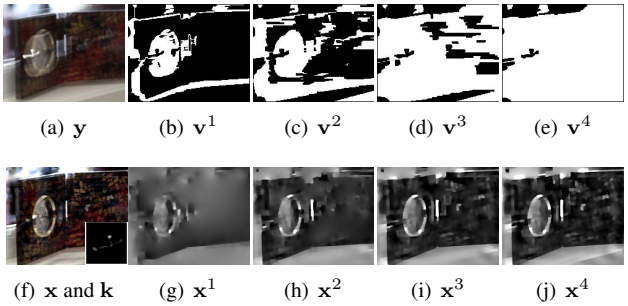
Relying on the coarse-to-fine scheme, in Step 1 of Algorithm 2, we initialize the blur kernel $\mathbf{k}^0$ with the result from the previous coarser pyramid level. The $\mathbf{x}^0$ is initialized as an image where all pixels' values are set as the mean values of y_i's, *i.e.* $\mathbf{x}^0 = \mathbf{1} \otimes (\mathbf{1}^\top\mathbf{y}/n)$. In Step 4, given fixed $\mathbf{k}^{k-1}$ and $\mathbf{v}^k$, we solve the gradient activation problem (8) and obtain an intermediate image $\mathbf{x}^k$ which only contains the most significant gradients to fit the regions indicated by $v_i = 1$. The blur kernel $\mathbf{k}$ is then estimated based on indicator $\mathbf{v}^k$ and the $\mathbf{x}^k$ containing only significant components.

Similar to the conventional SPL algorithms [18, 14], in Step 6, we update the pace parameter ρ via a factor $\eta \geq 1$, where the minimum value of ρ is set as $\rho^{\min}$. Figure 3 records an example of the updating process of $\mathbf{v}$ and the corresponding $\mathbf{x}$ in the iterations of Algorithm 2. In the early stage, easy pixels (*e.g.* strong edges in Figure 3 (b) and (g)) that mostly appear like inliers are involved first. With the decreasing of the fitting error (*i.e.* $l_i, \forall i \in [n]$) and ρ, more complex components are gradually introduced (*e.g.* regions containing complex details in Figure 3 (c)-(e) and (h)-(j)). We stop SPL when the stopping condition $(\sqrt{\mathbf{1}^\top\mathbf{l}^k} - \sqrt{\mathbf{1}^\top\mathbf{l}^{k-1}})/\sqrt{\mathbf{1}^\top\mathbf{l}^{k-1}} \leq \varepsilon$ is achieved, where ε is a small tolerance parameter. The maximum iteration number K is set as 5 in our implementation. As shown in Figure 3 (e) and (j) the regions contaminated can be excluded from the kernel estimation.

After obtaining the blur kernel, we recover the final sharp image via non-blind deconvolution methods, such as the sparse prior based method [20], the EM based method [7] for images with outliers, and the method for image with saturated regions [34].

5. Experiments

We evaluate the performance of the proposed method on both synthetic and real images, and compare with the state-of-the-art image deblurring methods. The proposed method is implemented in MATLAB, which takes 3 minutes for estimating a 19×19 blur kernel from a 255×255 image with an Intel Core i5 2.5 GHz CUP and 8 GB RAM.



(a) $\mathbf{y}$ (b) $\mathbf{v}^1$ (c) $\mathbf{v}^2$ (d) $\mathbf{v}^3$ (e) $\mathbf{v}^4$

(f) $\mathbf{x}$ and $\mathbf{k}$ (g) $\mathbf{x}^1$ (h) $\mathbf{x}^2$ (i) $\mathbf{x}^3$ (j) $\mathbf{x}^4$

Figure 3. The proposed self-paced scheme incrementally involves pixels for kernel estimation. (a) Blurred image $\mathbf{y}$. (b)-(e) The $\mathbf{v}$ in iteration #1-#4 of Algorithm 2. (g)-(j) The intermediate image $\mathbf{x}$ corresponding the $\mathbf{v}$ in (b)-(e). (f) Estimated $\mathbf{k}$ and recovered $\mathbf{x}$.

5.1. Evaluation on synthetic data

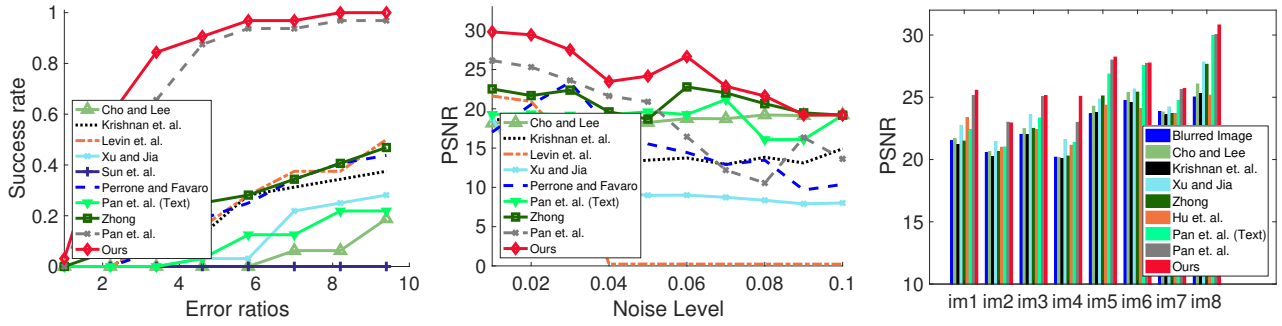
We evaluate the effectiveness of the proposed method on the synthetic data with non-Gaussian noise (salt-and-pepper noise) and saturated regions.

Images with non-Gaussian noise. We first investigate a benchmark dataset from Levin *et al.* [21], which consists of 32 blurred images by blurring 4 sharp images using 8 motion blur kernels. We add i.i.d. salt-and-pepper noise with noise level 0.01 to the blurred images, and then perform kernel estimation. The estimated kernels are used to recovering the noise-free blurred images using the same non-blind deconvolution method in [20]. In this way, the evaluation of the kernel accuracies is not influenced by the performance of the non-blind deconvolution method on noisy images [43]. We measure the sum-of-squared-difference (SSD) error between the recovered results and the ground truth.

We compare the proposed method with the state-of-the-art methods from Cho and Lee [6], Krishnan *et al.* [17], Levin *et al.* [22], Xu and Jia [37], Sun *et al.* [30], Perrone and Favaro [29], Pan *et al.* (Text) [25], Zhong *et al.* [43] and Pan *et al.* [26], and record the cumulative curves of SSD error ratio [21] in Figure 4 (a). As shown in the figure, the performance of Pan *et al.* [26] and ours are much better than that of others, and the proposed method performs the best among the competitors.

We evaluate different methods with increasing noise level from 0.01 to 0.1, and record the peak signal-to-noise ratio (PSNR) values of different methods in Figure 4 (b)[2]. It is observed that the performances of previous conventional methods decrease seriously when noise level increases. The performance of Pan *et al.* [26] degrades maybe because the method is sensitive to the parameter setting. The proposed method outperforms the other methods for all noise levels.

[2]Comparison under different noise levels is performed on the first image with the largest kernel in the dataset of Levin *et al.* [21].



(a) (b) (c)

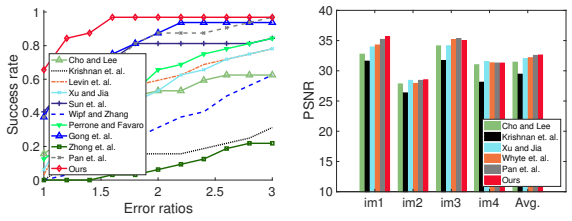
Figure 4. Comparison on the synthetic data with outliers. (a) Cumulative curves of SSD error ratio for data with salt-and-pepper noise. (b) SSD error for date contaminated by salt-and-pepper noise with increasing noise level. (c) Average PSNR values for data with saturation.

Images with saturation. To validate the proposed method's robustness on saturation, we first create a dataset with 64 images with saturated ranges by blurring 8 ground truth images using 8 motion blur kernels from [21]. For dataset generation, we first stretch the intensity the ground truth from $[0, 1]$ to $[0, 2]$ and apply the 8 different blur kernel to generate blurry images. We then clip the image intensities into $[0, 1]$. Finally, the Gaussian white noise with standard derivation 0.005 is added to each blurred images. We estimate blur kernels using each method and recover the final sharp image using the same non-blind deconvolution method in [13] for comparison. We record the average PSNR values w.r.t. the 8 images for different methods in Figure 4 (c). The performances of Pan *et al.* [26] and ours are superior to other conventional methods because the outliers are considered in these two methods. Furthermore, most PSNR values of the proposed method are higher due to the self-paced kernel estimation scheme.

Images without outliers. We also compare the proposed method with the state-of-the-art methods on the noise-free dataset [21]. The cumulative curves of SSD error is shown in Figure 5 (a). On images without outliers, the kernel estimation methods with outlier-handling process achieve better performances than the conventional results, which is identical to the results in [26] Specifically, the proposed method achieves the best performance in this experiment.

5.2. Evaluation on real-world images

Images with saturation and unknown noise. We first conduct experiments on low-light blurry images, which are often contaminated by outliers such as saturation and non-Gaussian noise. Apart from the example in Figure 1, we perform an experiment on the other real-world images with several saturated regions and unknown noise (See Figure 6 (a)). Although the blur kernels estimated by the methods from [6, 25, 26] have clear trajectories, the corresponding recovered images contain many ringing artifacts and blurry



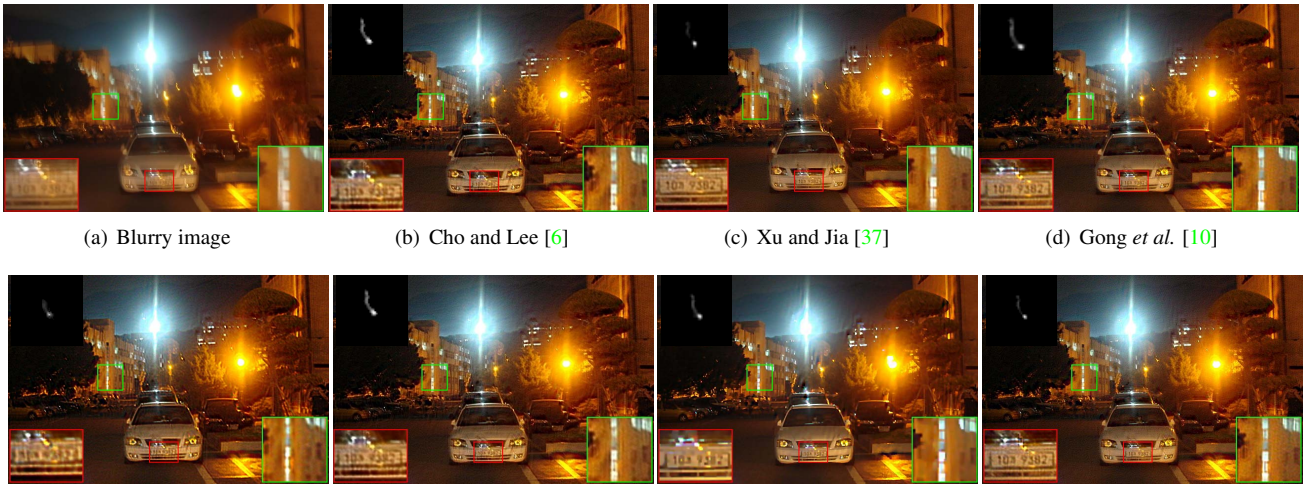
(a) (b)

Figure 5. Comparison on the images without outliers. (a) Cumulative curves of SSD error ratio for Levin *et al.* 's synthetic dataset [21]. (b) PSNR values for Köhler *et al.* 's real-world dataset [16].

details. The methods from [9, 13] and our method recover more details (such as the plate number) due to the more accurate and noiseless blur kernels.

Images with saturated regions. Since the saturation is a common and important origin [33] of the outliers, we specifically investigate the performance of the proposed method on an example with multiples saturated regions. We compare our method with several state-of-the-art methods seeking to handle saturation for deblurring [25, 13, 26]. As shown in Figure 7, the results of the ℓ_0-norm based method [25] and the robust kernel estimation method [26] suffers blurry details and/or ringing artifacts because of the error in the estimated kernel. However, our recovered image is sharper and cleaner due to the accurate kernel. The method of Hu *et al.* [13] is less ineffective than others since its light streak detection process fails.

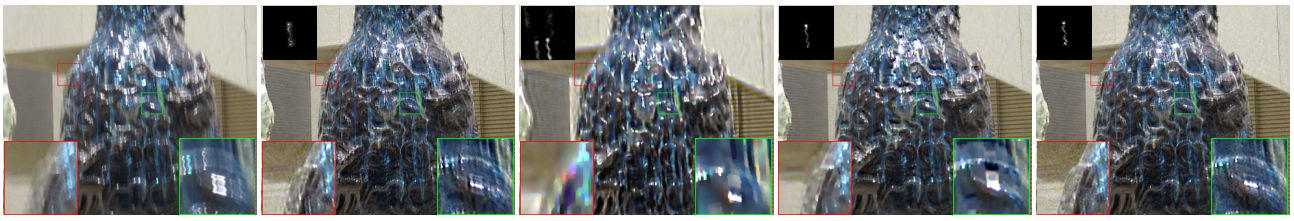
Images without outliers. We also study the benefits of the proposed method on deblurring images without outliers. Figure 8 (a) shows a real blurry image without outliers from [39]. We put the $\mathbf{v}$ from the last iteration of the proposed method at the top right conner of Figure 8 (e), which excludes some small scale edges that are hard to estimate accurately and may confuse kernel estimation [41]. Figure 8 shows that the proposed method reveals more natural details with less ringing artifacts.



(a) Blurry image (b) Cho and Lee [6] (c) Xu and Jia [37] (d) Gong *et al.* [10]

(e) Pan *et al.* [25] (f) Hu *et al.* [13] (g) Pan *et al.* [26] (h) Ours

Figure 6. Deblurring results on a real-world image with saturation and unknown noise.



(a) Blurry image (b) Pan *et al.* [25] (c) Hu *et al.* [13] (d) Pan *et al.* [26] (e) Ours

Figure 7. Deblurring results on a real-world image with saturated regions.

(a) Blurry image (b) Xu and Jia [37] (c) Pan *et al.* [25] (d) Pan *et al.* [26] (e) Ours

Figure 8. Deblurring results on a real-world image without outliers.

Dataset from Köhler *et al.* [16]. A benchmark dataset without outliers is also used to test the proposed method, which contains 48 (800×800) real blurry images from 4 sharp images and 12 blur types. We mainly focus on the 32 blurry images with mostly-invariant kernels [36] and record the comparison on PSNR values in Figure 5 (b), which suggests that the performance of the proposed method is better than most of the state-of-the-art methods and is on par with the best result of previous methods.

More experimental results can be found in the supplemental material.

6. Conclusion

In this paper, we proposed a robust self-paced kernel estimation method for blind image deblurring. To handle the outliers for the blur model, the method gradually involves inlier pixels based on model itself for kernel updating. Specifically, a novel graph-based self-paced regularizer was proposed to encourage the connectedness among the selected pixels, and a gradient activation method is used to update intermediate image with respected to the selected pixels. Experimental results on both synthetic and real-world data show the excellence of the proposed method.

References

[1] S. D. Babacan, R. Molina, M. N. Do, and A. K. Katsaggelos. Bayesian blind deconvolution with general sparse image priors. In *European Conference on Computer Vision (ECCV)*, pages 341–355. 2012. 2

[2] L. Bar, N. Kiryati, and N. Sochen. Image deblurring in the presence of impulsive noise. *International Journal of Computer Vision (IJCV)*, 70(3):279–298, 2006. 2

[3] Y. Bengio, J. Louradour, R. Collobert, and J. Weston. Curriculum learning. In *International Conference on Machine Learning (ICML)*, pages 41–48, 2009. 3

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 2011. 5

[5] T. F. Chan and C.-K. Wong. Total variation blind deconvolution. *IEEE Transactions on Image Processing*, pages 370–375, 1998. 2

[6] S. Cho and S. Lee. Fast motion deblurring. *SIGGRAPH ASIA*, 2009. 1, 2, 5, 6, 7, 8

[7] S. Cho, J. Wang, and S. Lee. Handling outliers in non-blind image deconvolution. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 495–502, 2011. 2, 3, 6

[8] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman. Removing camera shake from a single photograph. In *ACM Transactions on Graphics*, 2006. 1, 2, 6

[9] D. Gong, M. Tan, Y. Zhang, A. van den Hengel, and Q. Shi. Blind image deconvolution by automatic gradient activation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3, 4, 7

[10] D. Gong, M. Tan, Y. Zhang, A. van den Hengel, and Q. Shi. Mpgl: An efficient matching pursuit method for generalized lasso. In *The AAAI Conference on Artificial Intelligence*, 2017. 4, 5, 8

[11] C. Hegde, P. Indyk, and L. Schmidt. A nearly-linear time framework for graph-structured sparsity. In *International Conference on Machine Learning (ICML)*, pages 928–937, 2015. 3, 5

[12] M. Hirsch, S. Harmeling, S. Sra, and B. Schölkopf. Online multi-frame blind deconvolution with super-resolution and saturation correction. *Astronomy & Astrophysics*, 531:A9, 2011. 2

[13] Z. Hu, S. Cho, J. Wang, and M.-H. Yang. Deblurring low-light images with light streaks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3382–3389, 2014. 2, 3, 7, 8

[14] L. Jiang, D. Meng, S.-I. Yu, Z. Lan, S. Shan, and A. Hauptmann. Self-paced learning with diversity. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2078–2086, 2014. 3, 4, 6

[15] M. Jünger and W. Pulleyblank. New primal and dual matching heuristics. *Algorithmica*, 13(4):357–380, 1995. 5

[16] 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 *European Conference on Computer Vision (ECCV)*. 2012. 7, 8

[17] D. Krishnan, T. Tay, and R. Fergus. Blind deconvolution using a normalized sparsity measure. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 233–240, 2011. 2, 6

[18] M. P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1189–1197, 2010. 3, 4, 5, 6

[19] W.-S. Lai, J.-J. Ding, Y.-Y. Lin, and Y.-Y. Chuang. Blur kernel estimation using normalized color-line priors. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 64–72, 2015. 2

[20] A. Levin, R. Fergus, F. Durand, and W. T. Freeman. Image and depth from a conventional camera with a coded aperture. In *ACM Transactions on Graphics*, page 70, 2007. 1, 6

[21] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Understanding and evaluating blind deconvolution algorithms. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1964–1971, 2009. 6, 7

[22] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman. Efficient marginal likelihood optimization in blind deconvolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2657–2664, 2011. 2, 6

[23] L. B. Lucy. An iterative technique for the rectification of observed distributions. *The astronomical journal*, 79:745, 1974. 3

[24] A. Mutapcic and S. Boyd. Cutting-set methods for robust convex optimization with pessimizing oracles. *Optimization Methods & Software*, 2009. 4

[25] J. Pan, Z. Hu, Z. Su, and M.-H. Yang. Deblurring text images via l_0-regularized intensity and gradient prior. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2901–2908, 2014. 2, 3, 4, 6, 7, 8

[26] J. Pan, Z. Lin, Z. Su, and M.-H. Yang. Robust kernel estimation with outliers handling for image deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2800–2808, 2016. 1, 2, 3, 6, 7, 8

[27] J. Pan, D. Sun, H. Pfister, and M.-H. Yang. Blind image deblurring using dark channel prior. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 5

[28] E. Pee and J. O. Royset. On solving large-scale finite minimax problems using exponential smoothing. *Journal of Optimization Theory and Applications*, 2011. 4

[29] D. Perrone and P. Favaro. Total variation blind deconvolution: The devil is in the details. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 2, 6

[30] 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)*, pages 1–8, 2013. 2, 6

[31] J. S. Supancic and D. Ramanan. Self-paced learning for long-term tracking. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2379–2386, 2013. 3

[32] Y.-W. Tai, X. Chen, S. Kim, S. J. Kim, F. Li, J. Yang, J. Yu, Y. Matsushita, and M. S. Brown. Nonlinear camera response functions and image deblurring: Theoretical analysis and practice. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35(10):2498–2512, 2013. 2, 3

[33] O. Whyte, J. Sivic, and A. Zisserman. Deblurring shaken and partially saturated images. *International Journal of Computer Vision (IJCV)*, 2014. 2, 3, 7

[34] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *International journal of computer vision*, pages 168–186, 2012. 6

[35] D. Wipf and H. Zhang. Revisiting bayesian blind deconvolution. *The Journal of Machine Learning Research*, pages 3595–3634, 2014. 1, 2, 3

[36] L. Xiao, J. Gregson, F. Heide, and W. Heidrich. Stochastic blind motion deblurring. *IEEE Transactions on Image Processing*, 2015. 8

[37] L. Xu and J. Jia. Two-phase kernel estimation for robust motion deblurring. In *European Conference on Computer Vision (ECCV)*, pages 157–170, 2010. 1, 2, 3, 4, 6, 8

[38] L. Xu, J. S. Ren, C. Liu, and J. Jia. Deep convolutional neural network for image deconvolution. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1790–1798, 2014. 2, 3

[39] L. Xu, S. Zheng, and J. Jia. Unnatural l0 sparse representation for natural image deblurring. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1107–1114, 2013. 2, 5, 6, 7

[40] H. Zhang, D. Wipf, and Y. Zhang. Multi-image blind deblurring using a coupled adaptive sparse prior. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 2

[41] H. Zhang and J. Yang. Scale adaptive blind deblurring. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3005–3013, 2014. 7

[42] Q. Zhao, D. Meng, L. Jiang, Q. Xie, Z. Xu, and A. G. Hauptmann. Self-paced learning for matrix factorization. In *The AAAI Conference on Artificial Intelligence*, pages 3196–3202, 2015. 3

[43] L. Zhong, S. Cho, D. Metaxas, S. Paris, and J. Wang. Handling noise in single image deblurring using directional filters. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013. 6