Learning High-Order Filters for Efficient Blind Deconvolution of Document Photographs

Lei Xiao[2,1](✉), Jue Wang[3], Wolfgang Heidrich[1,2], and Michael Hirsch[4]

[1] KAUST, Thuwal, Saudi Arabia
[2] University of British Columbia, Vancouver, Canada
leixiao@cs.ubc.ca
[3] Adobe Research, Seattle, USA
[4] MPI for Intelligent Systems, Tübingen, Germany

Abstract. Photographs of text documents taken by hand-held cameras can be easily degraded by camera motion during exposure. In this paper, we propose a new method for blind deconvolution of document images. Observing that document images are usually dominated by small-scale high-order structures, we propose to learn a multi-scale, interleaved cascade of shrinkage fields model, which contains a series of high-order filters to facilitate joint recovery of blur kernel and latent image. With extensive experiments, we show that our method produces high quality results and is highly efficient at the same time, making it a practical choice for deblurring high resolution text images captured by modern mobile devices.

Keywords: Text document · Camera motion · Blind deblurring · High-order filters

1 Introduction

Taking photographs of text documents (printed articles, receipts, newspapers, books, etc.) instead of scanning them has become increasingly common due to the popularity of mobile cameras. However, photos taken by hand-held cameras are likely to suffer from blur caused by camera shake during exposure. This is critical for document images, as slight blur can prevent existing optical-character-recognition (OCR) techniques from extracting correct text from them. Removing blur and recovering sharp, eligible document images is thus highly desirable. As in many previous work, we assume a simple image formation model for each local text region as

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

where $\mathbf{y}$ represents the degraded image, $\mathbf{x}$ the sharp latent image, matrix $\mathbf{K}$ the corresponding 2D convolution with blur kernel $\mathbf{k}$, and $\mathbf{n}$ white Gaussian noise.

Electronic supplementary material The online version of this chapter (doi:10.1007/978-3-319-46487-9_45) contains supplementary material, which is available to authorized users.

B. Leibe et al. (Eds.): ECCV 2016, Part III, LNCS 9907, pp. 734–749, 2016.
DOI: 10.1007/978-3-319-46487-9_45

The goal of the post-processing is to recover $\mathbf{x}$ and $\mathbf{k}$ from single input $\mathbf{y}$, which is known as blind deconvolution or blind deblurring. This problem is highly ill-posed and non-convex. As shown in many previous work, good prior knowledge of both $\mathbf{x}$ and $\mathbf{k}$ is crucial for constraining the solution space and robust optimization. Specifically, most previous methods focus on designing effective priors for $\mathbf{x}$, while $\mathbf{k}$ is usually restricted to be smooth.

Recent text image deblurring methods use sparse gradient priors (e.g., total variation [3], ℓ_0 gradient [5,14]) and text-specific priors (e.g., text classifier [5], ℓ_0 intensity [14]) for sharp latent image estimation. These methods can produce high-quality results in many cases, however their practical adaptation is hampered by several drawbacks. Firstly, their use of sparse gradient priors usually forces the recovered image to be piece-wise constant. Although these priors are effective for images with large-font text (i.e., high pixel-per-inch (PPI)), they do not work well for photographs of common text documents such as printed articles and newspapers where the font sizes are typically small [10]. Furthermore, these methods employ iterative sparse optimization techniques that are usually time-consuming for high resolution images taken by modern cameras (e.g., up to a few megapixels).

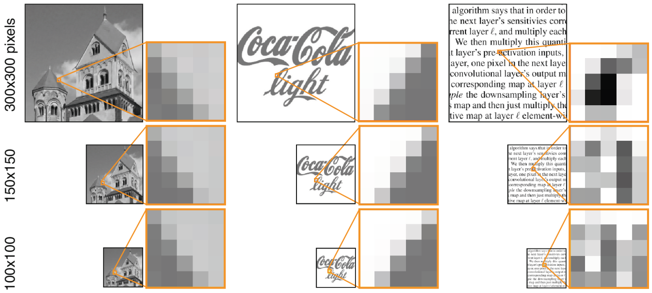


Fig. 1. Visual comparison between a natural image (left), a large-font text image (middle) and a common text document image at 150 PPI (right) at various scales.

In this paper, we propose a new algorithm for practical document deblurring that achieves both high quality and high efficiency. In contrast to previous works relying on low-order filter statistics, our algorithm aims to capture the domain-specific property of document images by learning a series of scale- and iteration-wise high-order filters. A motivational example is shown in Fig. 1, where we compare small patches extracted from a natural image, a large-font text image and a common text document image. Since most deblurring methods adopt a multi-scale framework in order to avoid bad local optima, we compare patches extracted from multiple scales. Evidently, the natural image and large-font text image both contain long, clear edges at all scales, making the use of sparse gradient priors effective. In contrast, patches from the document image with a small font size are mostly composed of small-scale high-order structures, especially at coarse scales, which makes sparse gradient priors to be inaccurate.

This observation motivates us to use high-order filter statistics as effective regularization for deblurring document images. We use a discriminative approach and learn such regularization terms by training a multi-scale, interleaved cascade of shrinkage field models [18], which was recently proposed as an effective tool for image restoration.

Our main contributions include:

- We demonstrate the importance of using high-order filters in text document image restoration.
- We propose a new algorithm for fast and high-quality deblurring of document photographs, suitable for processing high resolution images captured by modern mobile devices.
- Unlike the recent convolutional-neural-network (CNN) based document deblurring method [10], our approach is robust to page orientation, font style and text language, even though such variants are not included at our training.

2 Related Work

Blind Deblurring of Natural Images. Most deblurring methods solve the non-convex problem by alternately estimating latent image $\mathbf{x}$ and blur kernel $\mathbf{k}$, with an emphasis on designing effective priors on $\mathbf{x}$. Krishnan et al. [11] introduced a scale-invariant ℓ_1/ℓ_2 prior, which compensates for the attenuation of high frequencies in the blurry image. Xu et al. [24] used the ℓ_0 regularizer on the image gradient. Xiao et al. [22] used a color-channel edge-concurrence prior to facilitate chromatic kernel recovery. Goldstein and Fattal [8] estimated the kernel from the power spectrum of the blurred image. Yue et al. [25] improved [8] by fusing it with sparse gradient prior. Sun et al. [21] imposed patch priors to recover good partial latent images for kernel estimation. Michaeli and Irani [13] exploited the recurrence of small image patches across different scales of single natural images. Anwar et al. [2] learned a class-specific prior of image frequency spectrum for the restoration of frequencies that cannot be recovered with generic priors. Zuo et al. [26] learned iteration-wise parameters of the ℓ_p regularizer on image gradients. Schelten et al. [16] trained cascaded interleaved regression tree field (RTF) [19] to *post*-improve the result of other blind deblurring methods for natural images.

Another type of methods use explicit nonlinear filters to extract large-scale image edges from which kernels can be estimated rapidly. Cho and Lee [6] adopted a combination of shock and bilateral filters to predict sharp edges. Xu and Jia [23] improved [6] by neglecting edges with small spatial support as they impede kernel estimation. Schuler et al. [20] learned such nonlinear filters with a multi-layer convolutional neural network.

Blind Deblurring of Document Images. Most recent methods of text deblurring use the same sparse gradient assumption developed for natural images, and augment it with additional text-specific regularization. Chen et al. [3] and Cho et al. [5] applied explicit text pixel segmentation and enforced the text pixels to be dark or have similar colors. Pan et al. [14] used ℓ_0-regularized intensity and gradient priors for text deblurring. As discussed in Sect. 1 and as we will show in our experiments in Sect. 4, the use of sparse gradient priors makes such methods work well for large-font text images, but fail on common document images that have smaller fonts.

Hradiš et al. [10] trained a convolutional neural network to directly predict the sharp patch from a small blurry one, without considering the image formation model and explicit blur kernel estimation. With a large enough model and training dataset, this method produces good results on English documents with severe noise, large defocus blurs or simple motion blur. However, this method fails on more complicated motion trajectories, and is sensitive to page orientation, font style and text languages. Furthermore, this method often produces "hallucinated" characters or words which appears to be sharp and natural in the output image, but are completely wrong semantically. This undesirable side-effect severely limits its application range as most users do not expect the text to be changed in the deblurring process.

Discriminative Learning Methods for Image Restoration. Recently several methods were proposed to use trainable random field models for image restoration (denoising and *non-blind* deconvolution where the blur kernel is known a priori). These methods have achieved high-quality results with attractive run-times [4,18,19]. One representative technique is the shrinkage fields method [18], which reduces the optimization problem of random field models into cascaded quadratic minimization problems that can be efficiently solved in Fourier domain. In this paper, we extend this idea to the more challenging *blind* deconvolution problem, and employ the cascaded shrinkage fields model to capture high-order statistics of text document images.

3 Our Algorithm

The shrinkage fields (SF) model has been recently proposed as an effective and efficient tool for image restoration [18]. It has been successfully applied to both image denoising and non-blind image deconvolution, producing state-of-the-art results while maintaining high computational efficiency. Motivated by this success, we adopt the shrinkage field model for the challenging problem of *blind* deblurring of document images. In particular, we propose a multi-scale, interleaved cascade of shrinkage fields (CSF) which estimates the unknown blur kernel while progressively refining the estimation of the latent image. This is also partly inspired by [16], which proposes an interleaved cascade of regression tree fields (RTF) to *post*-improve the results of state-of-the-art natural image deblurring methods. However, in contrast to [16], our method *does not* depend on an initial kernel estimation from an auxiliary method. Instead, we estimate both the unknown blur kernel and latent sharp image from a single blurry input image.

3.1 Cascade of Shrinkage Fields (CSF)

The shrinkage field model can be derived from the field of experts (FoE) model [15]:

$$\underset{\mathbf{x}}{\operatorname{argmin}}\ \mathcal{D}(\mathbf{x}, \mathbf{y}) + \sum\nolimits_{i=1}^{N} \rho_i(\mathbf{F}_i \mathbf{x}), \tag{2}$$

where $\mathcal{D}$ represents the data fidelity given measurement $\mathbf{y}$, matrix $\mathbf{F}_i$ represents the corresponding 2D convolution with filter $\mathbf{f}_i$, and ρ_i is the penalty on the filter response. Half-quadratic optimization [7], a popular approach for the optimization of common random field models, introduces auxiliary variables $\mathbf{u}_i$ for all filter responses $\mathbf{F}_i\mathbf{x}$ and replaces the energy optimization problem Eq. 2 with a quadratic relaxation:

$$\underset{\mathbf{x},\mathbf{u}}{\operatorname{argmin}}\ \mathcal{D}(\mathbf{x}, \mathbf{y}) + \sum\nolimits_{i=1}^{N} \left(\beta ||\mathbf{F}_i \mathbf{x} - \mathbf{u}_i||_2^2 + \rho_i(\mathbf{u}_i)\right), \tag{3}$$

which for $\beta \to \infty$ converges to the original problem in Eq. 2. The key insight of [18] is that the minimizer of the second term w.r.t. $\mathbf{u}_i$ can be replaced by a flexible 1D shrinkage function ψ_i of filter response $\mathbf{F}_i\mathbf{x}$. Different from standard random fields which are parameterized through potential functions, SF models the shrinkage functions associated with the potential directly. Given data formation model as in Eq. 1, this reduces the original optimization problem Eq. 2 to a single quadratic minimization problem in each iteration, which can be solved efficiently as

$$\mathbf{x}^t = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\mathbf{K}_{t-1}^{\mathsf{T}}\mathbf{y} + \lambda^t \sum_{i=1}^{N} \mathbf{F}_i^{t^{\mathsf{T}}} \psi_i^t(\mathbf{F}_i^t \mathbf{x}^{t-1}))}{\mathcal{F}(\mathbf{K}_{t-1}^{\mathsf{T}}) \cdot \mathcal{F}(\mathbf{K}_{t-1}) + \lambda^t \sum_{i=1}^{N} \mathcal{F}(\mathbf{F}_i^{t^{\mathsf{T}}}) \cdot \mathcal{F}(\mathbf{F}_i^t)}\right], \tag{4}$$

where t is iteration index, $\mathbf{K}$ is the blur kernel matrix, $\mathcal{F}$ and $\mathcal{F}^{-1}$ indicate Fourier transform and its inverse, and ψ_i the shrinkage function. The model parameters $\Theta^t = (\mathbf{f}_i^t, \psi_i^t, \lambda^t)$ are trained by loss-minimization, e.g. by minimizing the ℓ_2 error between estimated images $\mathbf{x}^t$ and the ground truth. Performing multiple predictions of Eq. 4 is known as a cascade of shrinkage fields. For more details on the shrinkage fields model we refer readers to the supplemental material and [18].

3.2 Multi-scale Interleaved CSF for Blind Deconvolution

We do not follow the commonly used two-step deblurring procedure where kernel estimation and final latent image recovery are separated. Instead, we learn an interleaved CSF that directly produces both the estimated blur kernel and the predicted latent image. Our interleaved CSF is obtained by stacking multiple SFs into a cascade that is intermitted by kernel refinement steps. This cascade generates a sequence of iteratively refined blur kernel and latent image estimates, i.e. $\{\mathbf{k}^t\}_{t=1,..,T}$ and $\{\mathbf{x}^t\}_{t=1,..,T}$ respectively. At each stage of the cascade, we employ

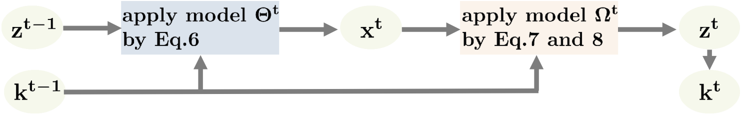

$\mathbf{z^{t-1}}$
apply model $\mathbf{\Theta^t}$ by Eq.6
$\mathbf{x^t}$
apply model $\mathbf{\Omega^t}$ by Eq.7 and 8
$\mathbf{z^t}$
$\mathbf{k^{t-1}}$
$\mathbf{k^t}$

Fig. 2. Algorithm architecture.

a separately trained SF model for sharp image restoration. In addition, we learn an auxiliary SF model which generates a latent image $\mathbf{z}^t$ that is used to facilitate blur kernel estimation. The reason of including this extra SF model at each stage is to allow for selecting features that might benefit kernel estimation and eliminating other features and artifacts. Note that the idea of introducing such a latent feature image for improving kernel estimation is not new, and is a rather common practice in recent state-of-the-art blind deconvolution methods [6,23]. Figure 2 depicts a schematic illustration of a single stage of our interleaved CSF approach.

More specifically, given the input image $\mathbf{y}$, our method recovers $\mathbf{k}$ and $\mathbf{x}$ simultaneously by solving the following optimization problem:

$$\begin{aligned}(\mathbf{x},\mathbf{k}) = &\operatorname*{argmin}_{\mathbf{x},\mathbf{k}} ||\mathbf{y}-\mathbf{k}\otimes\mathbf{x}||_2^2 + \sum\nolimits_{i=1}^{N} \rho_i(\mathbf{F}_i\mathbf{x}) + \tau||\mathbf{k}||_2^2, \\ &s.t. \quad \mathbf{k}\geq 0, ||\mathbf{k}||_1 = 1\end{aligned} \tag{5}$$

To this end, our proposed interleaved CSF alternates between the following blur kernel and latent image estimation steps:

Update $\mathbf{x}^t$. For sharp image update we train a SF model with parameters $\Theta^t = (\mathbf{f}_i^t, \psi_i^t, \lambda^t)$. Analogously to Eq. 4 we obtain the following update for $\mathbf{x}^t$ at iteration t:

$$\mathbf{x}^t = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\mathbf{K}_{t-1}^{\mathsf{T}}\mathbf{y} + \lambda^t \sum_{i=1}^{N} \mathbf{F}_i^{t\,\mathsf{T}} \psi_i^t(\mathbf{F}_i^t\mathbf{z}^{t-1}))}{\mathcal{F}(\mathbf{K}_{t-1}^{\mathsf{T}})\cdot\mathcal{F}(\mathbf{K}_{t-1}) + \lambda^t\sum_{i=1}^{N}\mathcal{F}(\mathbf{F}_i^{t\,\mathsf{T}})\cdot\mathcal{F}(\mathbf{F}_i^t)}\right] \tag{6}$$

Update $\mathbf{z}^t$ and $\mathbf{k}^t$. For kernel estimation we first update the latent image $\mathbf{z}^t$ from $\mathbf{x}^t$ by learning a separate SF model. Denoting convolution with filter $\mathbf{g}_i^t$ by matrix $\mathbf{G}_i^t$, we have:

$$\mathbf{z}^t = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\mathbf{K}_{t-1}^{\mathsf{T}}\mathbf{y} + \eta^t \sum_{i=1}^{N} \mathbf{G}_i^{t\,\mathsf{T}} \phi_i^t(\mathbf{G}_i^t\mathbf{x}^t))}{\mathcal{F}(\mathbf{K}_{t-1}^{\mathsf{T}})\cdot\mathcal{F}(\mathbf{K}_{t-1}) + \eta^t\sum_{i=1}^{N}\mathcal{F}(\mathbf{G}_i^{t\,\mathsf{T}})\cdot\mathcal{F}(\mathbf{G}_i^t)}\right] \tag{7}$$

For kernel estimation we employ a simple Thikonov prior. Given the estimated latent image $\mathbf{z}^t$ and the blurry input image $\mathbf{y}$, the update for $\mathbf{k}^t$ reads:

$$\mathbf{k}^t = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\mathbf{z}^t)^*\cdot\mathcal{F}(\mathbf{y})}{\mathcal{F}(\mathbf{z}^t)^*\cdot\mathcal{F}(\mathbf{z}^t)+\tau^t}\right], \tag{8}$$

where $*$ indicates complex conjugate. The model parameters learned at this step are denoted as $\Omega^t = (\mathbf{g}_i^t, \phi_i^t, \eta^t, \tau^t)$. Note that Ω^t are trained to facilitate the update of both kernel $\mathbf{k}^t$ and image $\mathbf{z}^t$.

The $\mathbf{x}^t$ update step in Eq. 6 takes $\mathbf{z}^{t-1}$ rather than $\mathbf{x}^{t-1}$ as input, as $\mathbf{z}^{t-1}$ improves from $\mathbf{x}^{t-1}$ w.r.t. removing blur by Eq. 7 at iteration $t-1$. $\mathbf{x}^t$ and $\mathbf{z}^t$ is observed to converge as the latent image and kernel are recovered.

Algorithm 1. Blind deblurring at one scale

```
Input: blurry image $\mathbf{y}$
Output: estimated image $\mathbf{x}$ and kernel $\mathbf{k}$.
for $t = 1$ to 5 do
    Update $\mathbf{x}^t$ by Eq. 6.
    Update $\mathbf{z}^t$ by Eq. 7.
    Update $\mathbf{k}^t$ by Eq. 8.
    $\mathbf{k}^t = \max(0, \mathbf{k}^t), \mathbf{k}^t = \mathbf{k}^t / ||\mathbf{k}^t||_1$.
end for
```

Algorithm 1 summarizes the proposed approach for blind deblurring of document images. Note that there is translation and scaling ambiguity between the sharp image and blur kernel at blind deconvolution. The estimated kernel is normalized such that all its pixel values sum up to one. In Algorithm 2 for training, $\mathbf{x}^t$ is shifted to better align with the ground truth image $\bar{\mathbf{x}}$, before updating $\mathbf{k}$. We find that our algorithm usually converges in 5 iterations per scale.

3.3 Learning

Our interleaved CSF has two sets of model parameters at every stage $t = 1, .., 5$, one for sharp image restoration, $\Theta^t = (\mathbf{f}_i^t, \psi_i^t, \lambda^t)$, and the other for blur kernel estimation, $\Omega^t = (\mathbf{g}_i^t, \phi_i^t, \eta^t, \tau^t)$. All model parameters are learned through loss-minimization.

Algorithm 2. Learning at one scale

```
Input: blurry image $\mathbf{y}$; true image $\bar{\mathbf{x}}$; true kernel $\bar{\mathbf{k}}$.
Output: model parameters $(\mathbf{f}_i^t, \psi_i^t, \lambda^t, \mathbf{g}_i^t, \phi_i^t, \eta^t, \tau^t)$
for $t = 1$ to 5 do
    Train model parameters: $(\mathbf{f}_i^t, \psi_i^t, \lambda^t)$ to minimize $||\mathbf{x}^t - \bar{\mathbf{x}}||_2^2$ with gradient given
    in Eq. 9.
    Update $\mathbf{x}^t$ by Eq. 6.
    Shift $\mathbf{x}^t$ to better align with $\bar{\mathbf{x}}$.
    Train model parameters: $(\mathbf{g}_i^t, \phi_i^t, \eta^t, \tau^t)$ to minimize $||\mathbf{k}^t - \bar{\mathbf{k}}||_2^2 + \alpha||\mathbf{z}^t - \bar{\mathbf{x}}||_2^2$ with
    gradient given in Eq. 10.
    Update $\mathbf{z}^t$ by Eq. 7.
    Update $\mathbf{k}^t$ by Eq. 8.
    $\mathbf{k}^t = \max(0, \mathbf{k}^t), \mathbf{k}^t = \mathbf{k}^t / ||\mathbf{k}^t||_1$.
end for
```

Note that in addition to the blurry input image, each model receives also the previous image and blur kernel predictions as input, which are progressively refined at each iteration. This is in contrast to the non-blind deconvolution setting of [18], where the blur kernel is known and is kept fixed throughout all stages. Our interleaved CSF model is trained in a greedy fashion, i.e. stage by stage such that the learned SF models at one stage are able to adapt to the kernel and latent image estimated at the previous stage.

More specifically, at each stage we update our model parameters by iterating between the following two steps:

Update $\mathbf{x}^t$. To learn the model parameters Θ^t, we minimize the ℓ_2 error between the current image estimate and the ground truth image $\bar{\mathbf{x}}$, i.e. $\ell = ||\mathbf{x}^t - \bar{\mathbf{x}}||_2^2$. Its gradient w.r.t. the model parameters $\Theta^t = (\mathbf{f}_i^t, \psi_i^t, \lambda^t)$ can be readily computed as

$$\frac{\partial \ell}{\Theta^t} = \frac{\partial \mathbf{x}^t}{\partial \Theta^t} \frac{\partial \ell}{\mathbf{x}^t} \tag{9}$$

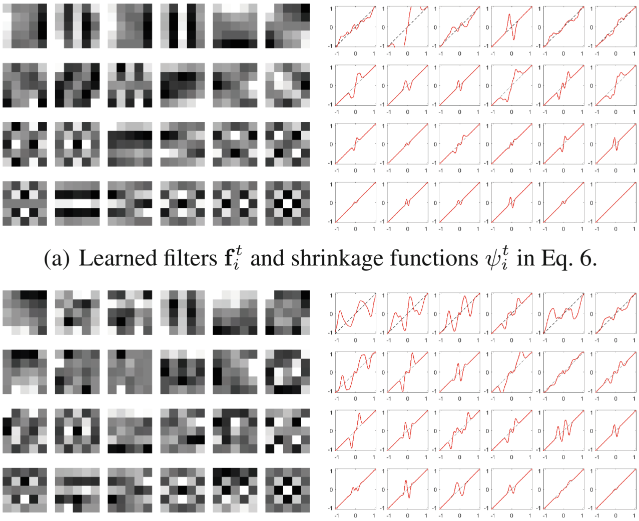
The derivatives for specific model parameters are omitted here for brevity, but can be found in the supplemental material.

Update $\mathbf{z}^t$ and $\mathbf{k}^t$. The model parameters Ω^t of the SF models for kernel estimation at stage t are learned by minimizing the loss function $\ell = ||\mathbf{k}^t - \bar{\mathbf{k}}||_2^2 + \alpha||\mathbf{z}^t - \bar{\mathbf{x}}||_2^2$, where $\bar{\mathbf{k}}$ denotes the ground truth blur kernel and α is a coupling constant. This loss accounts for errors in the kernel but also prevents the latent image used in Eq. (8) to diverge. Its gradient w.r.t. the model parameters $\Omega^t = (\mathbf{g}_i^t, \phi_i^t, \eta^t, \tau^t)$ reads

$$\frac{\partial \ell}{\partial \Omega^t} = \frac{\partial \mathbf{z}^t}{\partial \Omega^t} \frac{\partial \mathbf{k}^t}{\partial \mathbf{z}^t} \frac{\partial \ell}{\partial \mathbf{k}^t} + \frac{\partial \mathbf{k}^t}{\partial \Omega^t} \frac{\partial \ell}{\partial \mathbf{k}^t} + \frac{\partial \mathbf{z}^t}{\partial \Omega^t} \frac{\partial \ell}{\partial \mathbf{z}^t} \tag{10}$$

Again, details for the computation of the derivatives w.r.t. to specific model parameters are included in the supplemental material. We want to point out that the kernel estimation error $||\mathbf{k}^t - \bar{\mathbf{k}}||_2^2$ is back-propagated to the model parameters $(\mathbf{g}_i^t, \phi_i^t, \eta^t)$ in the SF for $\mathbf{z}^t$. Hence, the latent image $\mathbf{z}^t$ is tailored for accurate kernel estimation and predicted such that the refinement in $\mathbf{k}^t$ in each iteration is optimal. This differs from related work in [16,26].

Multi-scale Approach. Our algorithm uses a multi-scale approach to prevent bad local optima. The kernel widths that are used at different scales are 5, 9, 17, 25 pixels. At each scale s, the blurry image $\mathbf{y}^s$, the true latent image $\bar{\mathbf{x}}^s$ and $\bar{\mathbf{k}}^s$ are downsampled (and normalized for $\bar{\mathbf{k}}^s$) from their original resolution. The scale index s is omitted for convenience. At the beginning of each scale $s > 1$, the estimated image $\mathbf{x}$ is initialized by bicubic upsampling its estimation at the previous scale, and the blur kernel $\mathbf{k}$ is initialized by nearest-neighbor upsampling, followed by re-normalization. At the coarsest scale $s = 1$, $\mathbf{x}$ is initialized as $\mathbf{y}$ and $\mathbf{k}$ is initialized as a delta peak. The coupling constant α in kernel estimation loss is defined as $\alpha = r \cdot \eta$, where r is the ratio between pixel numbers in kernel $\mathbf{k}^t$ and image $\mathbf{z}^t$ at current scale, η is initialized with 1 at the coarsest scale and at each subsequent scale it is multiplied by a factor of 0.25. Algorithm 2 summarizes our learning procedure for a single scale of our CSF model.



(a) Learned filters $\mathbf{f}_i^t$ and shrinkage functions ψ_i^t in Eq. 6.

(b) Learned filters $\mathbf{g}_i^t$ and shrinkage functions ϕ_i^t in Eq. 7.

Fig. 3. Learned filters and shrinkage functions (at 3rd scale, 1st iteration) for updating $\mathbf{x}^t$ (Eq. 6) and $\mathbf{z}^t$, $\mathbf{k}^t$ (Eq. 7), respectively. Other parameters learned at this iteration: λ^t=0.5757, η^t=0.0218, τ^t=0.0018.

Model Complexity. In both the model Θ^t for $\mathbf{x}^t$ and model Ω^t for ($\mathbf{z}^t$, $\mathbf{k}^t$), we choose to use 24 filters $\mathbf{f}_i^t$ of size 5×5 for trade-off between result quality, model complexity and time efficiency. As in [18], we initialize the filters with a DCT filter bank. Each shrinkage function ψ_i^t and ϕ_i^t are composed of 51 equidistant-positioned radial basis functions (RBFs) and are initialized as identity function. We further enforce central symmetry to the shrinkage functions, so that the number of trainable RBFs reduces by half to 25. Figure 3 visualizes some learned models.

Training Datasets. We have found that that our method works well with a relatively small training dataset without over-fitting. We collected 20 motion blur kernels from [18], and randomly rotated them to generate 60 different kernels. We collected 60 sharp patches of 250×250 pixels cropped from documents rendered around 175 PPI, and rotated each with a random angle between -4 and 4 degrees. We then generated 60 blurry images by convolving each pair of sharp image and kernel, followed by adding white Gaussian noise and quantizing to 8 bits. We used the L-BFGS solver [17] in Matlab for training, which took about 12 h on a desktop with an Intel Xeon CPU.

4 Results

In this section we evaluate the proposed algorithm on both synthetic and real-world images. We compare with Pan et al. [14] and Hradiš et al. [10],

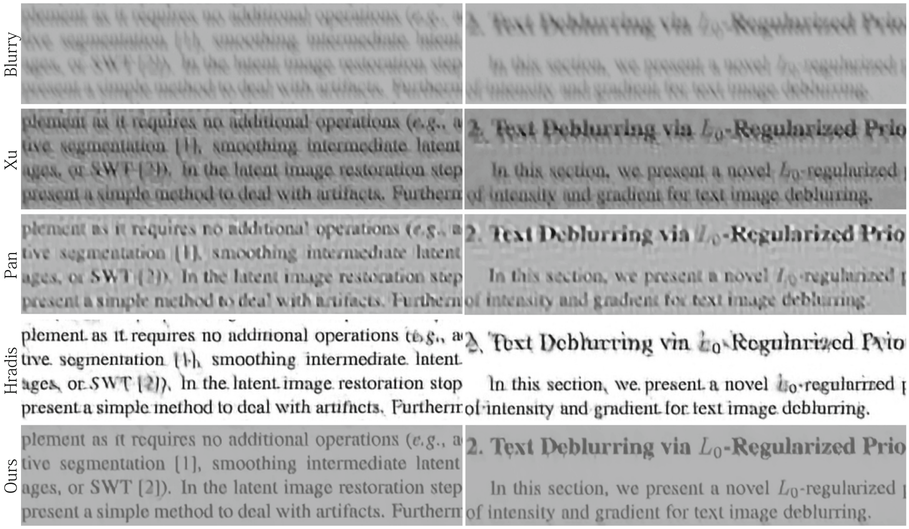


Fig. 4. Comparison on a real image taken from [10]. Row 1–5 from top to bottom show the blurry image, result of Xu [1], Pan [14], Hradiš et al. [10] and our method. Two cropped regions are shown here, the full resolution results along with more examples can be found in the supplemental.

the state-of-the-art methods for text image blind deblurring, and the natural image deblurring software produced by Xu [1], which are based on recently proposed state-of-the-art techniques [23, 24]. We used the code and binaries provided by the authors and tuned the parameters to generate the best possible results.

Real-World Images. In Figs. 4 and 5 we show comparisons on real images. The result images of Xu [1] and Pan [14] contain obvious artifacts due to ineffective image priors that lead to inaccurate kernel estimation. Hradiš et al. [10] fails to recover many characters and distorted the font type and illumination. Our method produces the best results in these cases, and our results are both visually pleasing and highly legible. The full resolution images and more results are included in the supplemental material.

Quantitative Comparisons. For quantitative evaluation, we test all methods on a synthetic dataset and compare results in terms of the peak-signal-to-noise-ratio (PSNR). We collect 8 sharp document images with 250×250 pixels cropped from documents rendered at 150 PPI (similar PPI as used for training in [10]). Each image is blurred with 8 kernels at 25×25 collected from [12], followed by adding 1 % Gaussian noise and 8-bit quantization. In Fig. 6, we show the average PSNR values of all 8 test images synthesized with the same blur kernel. Our method outperforms other methods in all cases by 0.5–6.0 dB. Hradiš et al. [10] has close performance to ours on kernel #3, which is close to defocus blur. It also performs reasonably well on kernel #6 which features a simple motion path,

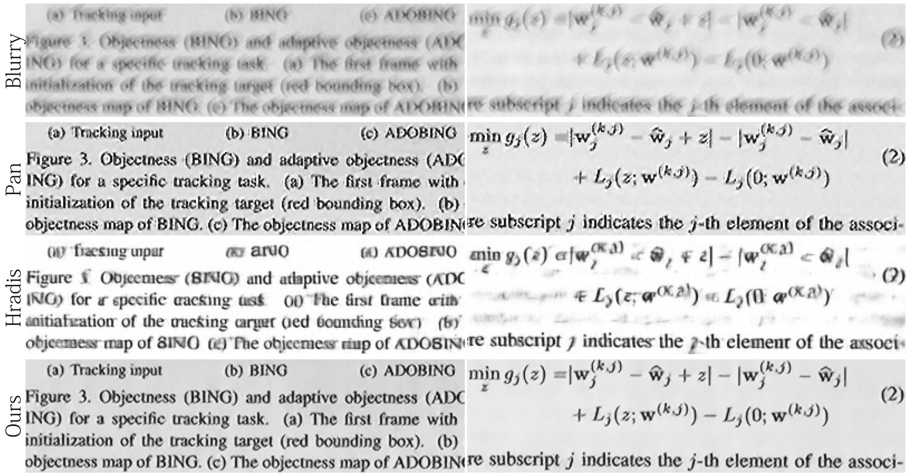


Fig. 5. Comparison on a real image taken from [10]. Row 1–4 from top to bottom show the blurry image, result of Pan [14], Hradiš et al. [10] and our method. Two cropped regions are shown, the full resolution results along with more results can be found in the supplemental.

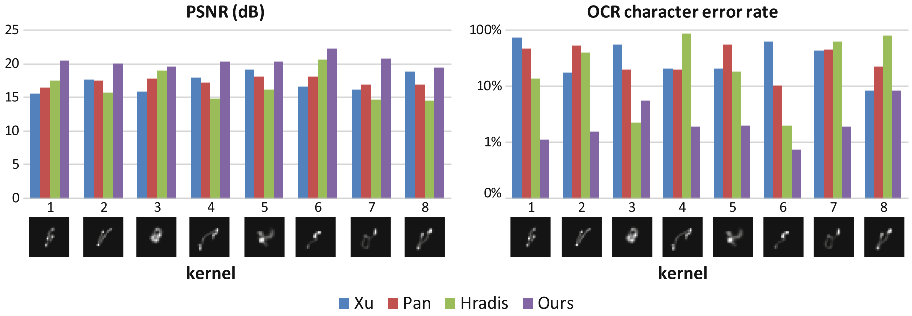
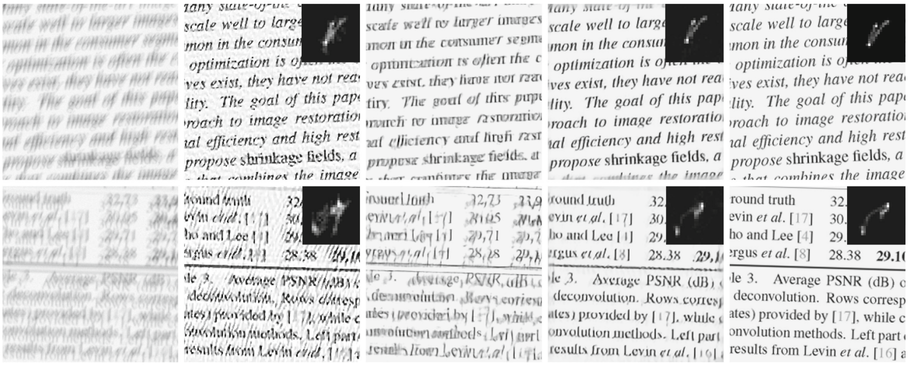


Fig. 6. PSNR and OCR comparison on a synthetic test dataset with 8 blur kernels.

but fails on other more challenging kernels. Some results along with the estimated kernels are shown in Fig. 7 for visual comparison.

An interesting question one may ask is whether improved deblur can directly lead to better optical-character-recognition (OCR) accuracy. To answer this question we evaluate OCR accuracy using the software ABBYY FineReader 12. We collected 8 sharp document images from the OCR test dataset in [10]. Each document image contains a continuous paragraph. We synthesized 64 blurry images with the 8 kernels and 1 % Gaussian noise similarly as in the PSNR comparison. We run the OCR software and used the script provided by [10] to compute the average character error rate for all 8 test images synthesized with



(a) Blurry (b) Pan [14] (c) Hradiš [10] (d) Ours (e) Ground truth

Fig. 7. Comparison on synthetic images from the PSNR experiments in Fig. 6. Note that the original results of [10] break the illumination of the images. We clamp the intensity of their results to match the ground truth image before computing the PSNR values.

Table 1. Run-time comparison (in seconds).

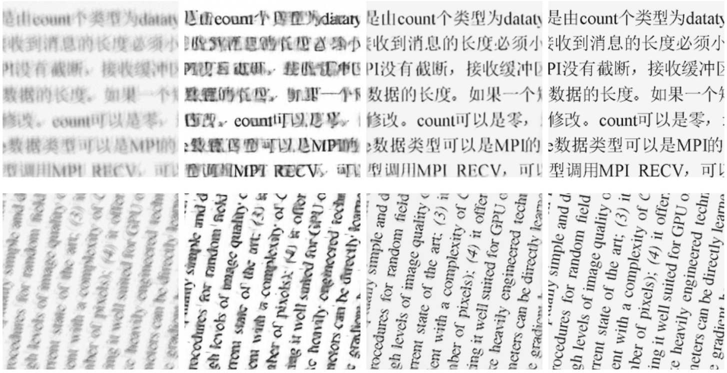
Image size	256^2	512^2	1024^2
Xu [1] (C++)	14.8	33.4	-
Pan [14] (Matlab)	19.6	84.3	271.9
Hradiš et al. [10] (C++)	48.5	193.7	594.9
Hradiš et al. [10] (GPU)	0.3	1.0	3.1
Ours (Matlab)	2.0	3.9	11.4
Pre-computation (Matlab)	1.8	4.6	15.3

the same kernel[1]. The results are shown in Fig. 6. They are consistent with the PSNR results also in Fig. 6. Hradiš et al. [10] performs well on kernel #3 and #6 but fails on other challenging kernels, while our method is consistently better than others. All the test images and results for PSNR and OCR comparisons are included in the supplemental material.

Run-Time Comparison. Table 1 provides a comparison on computational efficiency, using images blurred by a 17×17 kernel at three different resolutions. The experiments were done on an Intel i7 CPU with 16 GB RAM and a GeForce GTX TITAN GPU. Assuming the image sensor resolution is a known priori[2], we pre-compute the FFTs of the trained filters $\mathbf{f}_i$ and $\mathbf{g}_i$ for maximal efficiency. We report the timing of our Matlab implementation on CPU. A GPU implementation should significantly reduce the time as our method only

[1] We used the script ‘eval.py’ downloaded from the author webpage [10] to compute the error rate (after a bug was fixed).

[2] This is a common assumption especially for batch processing of document images.



(a) Blurry (b) Hradiš [10] (c) Ours (d) Ground truth

Fig. 8. Comparison on non-English text and severely rotated images. Note that such non-English text and large rotation were not included in our training dataset.

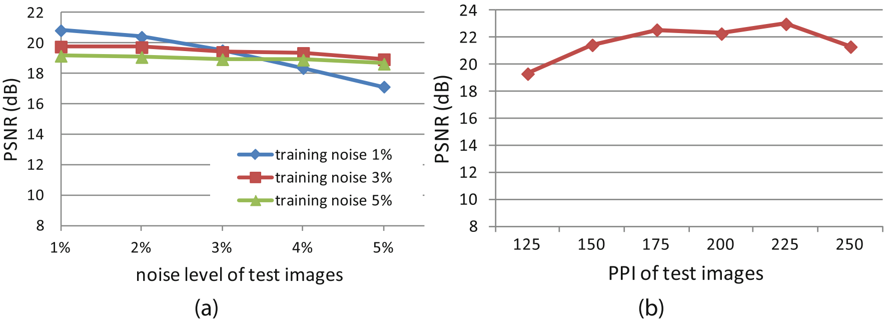
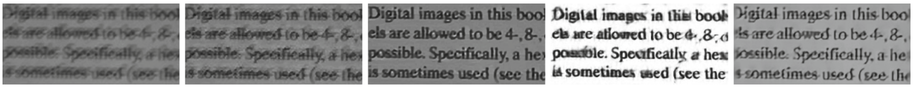


Fig. 9. Robustness test on noise level and image PPI (pixel-per-inch).

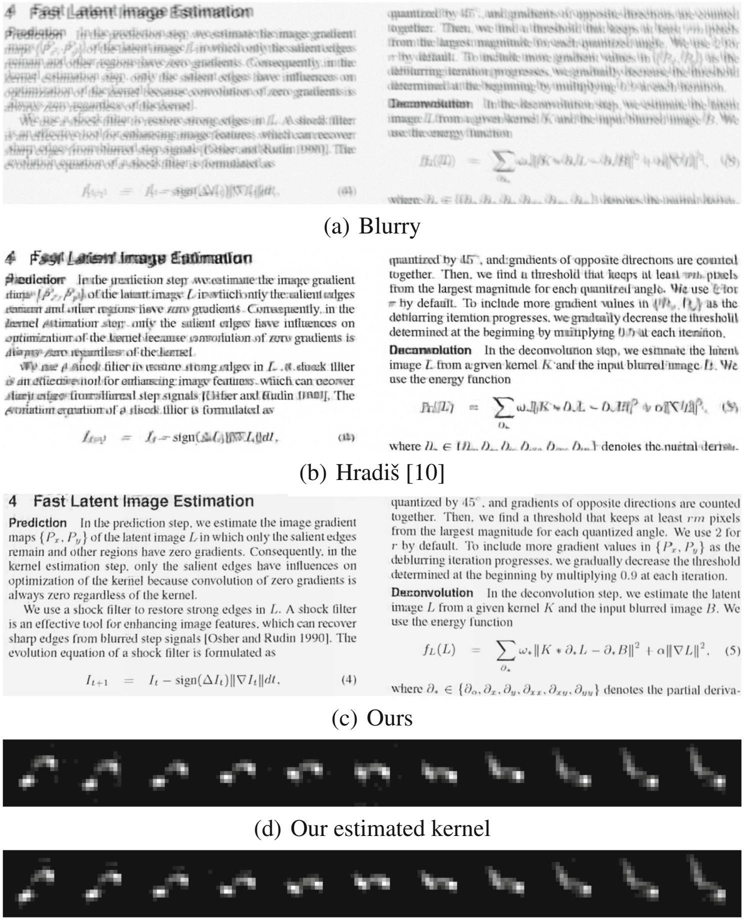


(a) Blurry (b) Xu [1] (c) Pan [14] (d) Hradiš [10] (e) Ours

Fig. 10. Comparison on a real image with large-font text. The reference results are from [10]. Following [10], the input of (d) Hradiš' and (e) our method was downsampled by factor of 3.

requires FFT, 2D convolution and 1D look-up-table (LUT) operations, which is our future work.

Robustness. In Fig. 8, we show results on non-English text and severely rotated image. Although both Hradiš et al. [10] and our method are only trained on English text data, our method can be applied to non-English text as well. This is a great benefit of our method as we do not need to train on every different language, or increase the model complexity to handle them as [10] would need to do.



(a) Blurry

(b) Hradiš [10]

(c) Ours

4 Fast Latent Image Estimation

Prediction In the prediction step, we estimate the image gradient maps $\{P_x, P_y\}$ of the latent image L in which only the salient edges remain and other regions have zero gradients. Consequently, in the kernel estimation step, only the salient edges have influences on optimization of the kernel because convolution of zero gradients is always zero regardless of the kernel.

We use a shock filter to restore strong edges in L. A shock filter is an effective tool for enhancing image features, which can recover sharp edges from blurred step signals [Osher and Rudin 1990]. The evolution equation of a shock filter is formulated as

$$I_{t+1} = I_t - \mathrm{sign}(\Delta I_t)\|\nabla I_t\|dt, \quad (4)$$

quantized by 45°, and gradients of opposite directions are counted together. Then, we find a threshold that keeps at least $r m$ pixels from the largest magnitude for each quantized angle. We use 2 for r by default. To include more gradient values in $\{P_x, P_y\}$ as the deblurring iteration progresses, we gradually decrease the threshold determined at the beginning by multiplying 0.9 at each iteration.

Deconvolution In the deconvolution step, we estimate the latent image L from a given kernel K and the input blurred image B. We use the energy function

$$f_L(L) = \sum_{\partial_*} \omega_* \|K * \partial_* L - \partial_* B\|^2 + \alpha\|\nabla L\|^2, \quad (5)$$

where $\partial_* \in \{\partial_o, \partial_x, \partial_y, \partial_{xx}, \partial_{xy}, \partial_{yy}\}$ denotes the partial deriva-

(d) Our estimated kernel

(e) Ground truth kernel

Fig. 11. Results on spatially-varying blur kernel. The blurry input is synthesized with the EFF model [9] to approximate practical pixel-wise variant blur.

Our method is also robust against a significant change of page orientation, which cannot be handled well by [10].

In Fig. 9, we show the results of our method when the noise level and PPI of the test data differs from the training data. Figure 9(a) shows that the performance of our method is fairly steady when the noise level in the test images is not too much higher than that of the training data, meaning that the models trained at sparse noise levels are sufficient for practical use. Figure 9(b) shows that our method works well in a fairly broad range of image PPIs given the training data are around 175 PPI.

In Fig. 10, we show a comparison on a real image with large-font text. Following [10], the input of Hradiš' and our method was downsampled by factor of 3 in order to apply the trained models without re-training. Although such downsampling breaks the image formation model in Eq. 1, our method can still generate reasonable result.

Non-uniform Blur. Our method can be easily extended to handle non-uniform blur by dividing the image into overlapped tiles, deblurring each tile with our proposed algorithm, and then realigning the resulting tiles to generate the final estimated image. An example is shown in Fig. 11.

5 Conclusion and Discussion

In this paper we present a new algorithm for fast and high-quality blind deconvolution of document photographs. Our key idea is to to use high-order filters for document image regularization, and propose to learn such filters and influences from training data using multi-scale, interleaved cascade of shrinkage field models. Extensive experiments demonstrate that our approach not only produces higher quality results than the state-of-the-art methods, but is also computational efficient, and robust against noise level, language and page orientation changes that are not included in the training data.

Our method also has some limitations. It cannot fully recover the details of an image if it is degraded by large out-of-focus blur. In such case, Hradiš et al. [10] may outperform our method given its excellent synthesis ability. As future work it would be interesting to combine both approaches. Although we only show learning our model on document photographs, we believe such a framework can also be applied to other domain-specific images, which we plan to explore in the future. The code, dataset and other supplemental material will be available on the author's webpage.

Acknowledgement. This work was supported in part by Adobe and Baseline Funding of KAUST. Part of this work was done when the first author was an intern at Adobe Research. The authors thank the anonymous reviewers for helpful suggestions.

References

1. Robust deblurring software. www.cse.cuhk.edu.hk/~leojia/deblurring.htm
2. Anwar, S., Phuoc Huynh, C., Porikli, F.: Class-specific image deblurring. In: ICCV (2015)
3. Chen, X., He, X., Yang, J., Wu, Q.: An effective document image deblurring algorithm. In: CVPR (2011)
4. Chen, Y., Yu, W., Pock, T.: On learning optimized reaction diffusion processes for effective image restoration. In: CVPR (2015)
5. Cho, H., Wang, J., Lee, S.: Text image deblurring using text-specific properties. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 524–537. Springer, Heidelberg (2012)
6. Cho, S., Lee, S.: Fast motion deblurring. ACM Trans. Graph. **28**(5) (2009)
7. Geman, D., Yang, C.: Nonlinear image recovery with half-quadratic regularization. IEEE Trans. Image Process. **4**(7), 932–946 (1995)

8. Goldstein, A., Fattal, R.: Blur-Kernel estimation from spectral irregularities. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part V. LNCS, vol. 7576, pp. 622–635. Springer, Heidelberg (2012)
9. Hirsch, M., Sra, S., Scholkopf, B., Harmeling, S.: Efficient filter flow for space-variant multiframe blind deconvolution. In: CVPR (2010)
10. Hradiš, M., Kotera, J., Zemcík, P., Šroubek, F.: Convolutional neural networks for direct text deblurring. In: BMVC (2015)
11. Krishnan, D., Tay, T., Fergus, R.: Blind deconvolution using a normalized sparsity measure. In: CVPR (2011)
12. Levin, A., Weiss, Y., Durand, F., Freeman, W.T.: Understanding and evaluating blind deconvolution algorithms. In: CVPR (2009)
13. Michaeli, T., Irani, M.: Blind deblurring using internal patch recurrence. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part III. LNCS, vol. 8691, pp. 783–798. Springer, Heidelberg (2014)
14. Pan, J., Hu, Z., Su, Z., Yang, M.H.: Deblurring text images via. l0-regularized intensity and gradient prior. In: CVPR (2014)
15. Roth, S., Black, M.J.: Fields of experts: a framework for learning image priors. In: CVPR (2005)
16. Schelten, K., Nowozin, S., Jancsary, J., Rother, C., Roth, S.: Interleaved regression tree field cascades for blind image deconvolution. In: WACV (2015)
17. Schmidt, M.: minfunc: unconstrained differentiable multivariate optimization in matlab. http://www.cs.ubc.ca/~schmidtm/Software/minFunc.html
18. Schmidt, U., Roth, S.: Shrinkage fields for effective image restoration. In: CVPR (2014)
19. Schmidt, U., Rother, C., Nowozin, S., Jancsary, J., Roth, S.: Discriminative non-blind deblurring. In: CVPR (2013)
20. Schuler, C.J., Hirsch, M., Harmeling, S., Schölkopf, B.: Learning to deblur (2014). arXiv preprint arXiv:1406.7444
21. Sun, L., Cho, S., Wang, J., Hays, J.: Edge-based blur kernel estimation using patch priors. In: ICCP (2013)
22. Xiao, L., Gregson, J., Heide, F., Heidrich, W.: Stochastic blind motion deblurring. IEEE Trans. Image Process. **24**(10), 3071–3085 (2015)
23. Xu, L., Jia, J.: Two-Phase Kernel estimation for robust motion deblurring. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 157–170. Springer, Heidelberg (2010)
24. Xu, L., Zheng, S., Jia, J.: Unnatural l0 sparse representation for natural image deblurring. In: CVPR (2013)
25. Yue, T., Cho, S., Wang, J., Dai, Q.: Hybrid image deblurring by fusing edge and power spectrum information. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014, Part VII. LNCS, vol. 8695, pp. 79–93. Springer, Heidelberg (2014)
26. Zuo, W., Ren, D., Gu, S., Lin, L., Zhang, L.: Discriminative learning of iteration-wise priors for blind deconvolution. In: CVPR (2015)