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

Lei Xiao[2,1] Jue Wang[3] Wolfgang Heidrich[1,2] Michael Hirsch[4]

[1]KAUST [2]University of British Columbia [3]Adobe Research
[4]MPI for Intelligent Systems

The attached supplementary files include the result images of the PSNR and OCR comparisons in Fig.6 of the main paper, the full resolution result images of Fig.4-5 in the main paper, and more results on real-world images.

In this supplementary pdf, Sec. 1 gives the derivation details of important equations in the main paper, and Sec. 2 shows more details and results of our algorithm.

1 Derivations

This section gives the details of Eq.6-10 in the main paper. Note, that we use the gray box ▒ to indicate the equations and figures from the main paper.

Derivation of Eq.6-7. The objective of latent image estimation at t-th iteration is defined as

$$\mathbf{x}^t = \underset{\mathbf{x}^t}{\operatorname{argmin}} ||\mathbf{y} - \mathbf{k}^{t-1} \otimes \mathbf{x}^t||_2^2 + \sum_{i=1}^{N} \rho_i^t(\mathbf{F}_i^t \mathbf{x}^t) \tag{1}$$

Applying the half-quadratic splitting [1] on each filter response $\mathbf{F}_i^t \mathbf{x}^t$, we have

$$\mathbf{x}^t = \underset{\mathbf{x}^t, \mathbf{u}_i^t}{\operatorname{argmin}} ||\mathbf{y} - \mathbf{k}^{t-1} \otimes \mathbf{x}^t||_2^2 + \sum_{i=1}^{N} \left(\rho_i^t(\mathbf{u}_i^t) + \frac{\beta^t}{2} ||\mathbf{F}_i^t \mathbf{x}^t - \mathbf{u}_i^t||_2^2 \right) \tag{2}$$

The half-quadratic optimization technique [1] solves Eq. 2 by alternatively updating $\mathbf{x}^t$ and $\mathbf{u}_i^t$. More specifically, for $\mathbf{x}^t$ update,

$$\mathbf{x}^t = \underset{\mathbf{x}^t}{\operatorname{argmin}} ||\mathbf{y} - \mathbf{k}^{t-1} \otimes \mathbf{x}^t||_2^2 + \sum_{i=1}^{N} \frac{\beta^t}{2} ||\mathbf{F}_i^t \mathbf{x}^t - \mathbf{u}_i^t||_2^2 \tag{3}$$

This is solved by setting the gradient of the right-hand linear least squares to be zero,

$$\left(\mathbf{K}_{t-1}^\top \mathbf{K}_{t-1} + \frac{\beta^t}{2} \sum_{i=1}^{N} {\mathbf{F}_i^t}^\top \mathbf{F}_i^t \right) \mathbf{x}^t = \mathbf{K}_{t-1}^\top \mathbf{y} + \frac{\beta^t}{2} \sum_{i=1}^{N} {\mathbf{F}_i^t}^\top \mathbf{u}_i^t \tag{4}$$

where matrix $\mathbf{K}_{t-1}$ represents corresponding convolution with blur kernel $\mathbf{k}^{t-1}$. Because $\mathbf{K}_{t-1}$ and $\mathbf{F}_i^t$ represent convolution process, Eq. 4 can be efficiently solved in Fourier domain. Let $\lambda^t = \beta^t/2$, we have

$$\mathbf{x}^t = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\mathbf{K}_{t-1}^\top \mathbf{y} + \lambda^t \sum_{i=1}^N \mathbf{F}_i^{t\top}\mathbf{u}_i^t)}{\mathcal{F}(\mathbf{K}_{t-1}^\top)\cdot\mathcal{F}(\mathbf{K}_{t-1}) + \lambda^t \sum_{i=1}^N \mathcal{F}(\mathbf{F}_i^{t\top})\cdot\mathcal{F}(\mathbf{F}_i^t)}\right] \tag{5}$$

For $\mathbf{u}_i^t$ update,

$$\mathbf{u}_i^t = \underset{\mathbf{u}_i^t}{\operatorname{argmin}}\, \rho_i^t(\mathbf{u}_i^t) + \frac{\beta^t}{2}||\mathbf{F}_i^t\mathbf{x}^{t-1} - \mathbf{u}_i^t||_2^2 \tag{6}$$

This minimization problem is pixel-wise separable. The key idea of [5] is to replace this minimization problem with a shrinkage function ψ_i^t, i.e.,

$$\mathbf{u}_i^t = \psi_i^t(\mathbf{F}_i^t\mathbf{x}^{t-1}) \tag{7}$$

By Substituting Eq. 7 to Eq. 5, we have

$$\mathbf{x}^t = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\mathbf{K}_{t-1}^\top \mathbf{y} + \lambda^t \sum_{i=1}^N \mathbf{F}_i^{t\top}\psi_i^t(\mathbf{F}_i^t\mathbf{x}^{t-1}))}{\mathcal{F}(\mathbf{K}_{t-1}^\top)\cdot\mathcal{F}(\mathbf{K}_{t-1}) + \lambda^t \sum_{i=1}^N \mathcal{F}(\mathbf{F}_i^{t\top})\cdot\mathcal{F}(\mathbf{F}_i^t)}\right] \tag{8}$$

Note that we replace $\mathbf{x}^{t-1}$ with $\mathbf{z}^{t-1}$ in our blind deblurring framework. Therefore, Eq. 8 becomes

$$\mathbf{x}^t = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\mathbf{K}_{t-1}^\top \mathbf{y} + \lambda^t \sum_{i=1}^N \mathbf{F}_i^{t\top}\psi_i^t(\mathbf{F}_i^t\mathbf{z}^{t-1}))}{\mathcal{F}(\mathbf{K}_{t-1}^\top)\cdot\mathcal{F}(\mathbf{K}_{t-1}) + \lambda^t \sum_{i=1}^N \mathcal{F}(\mathbf{F}_i^{t\top})\cdot\mathcal{F}(\mathbf{F}_i^t)}\right] \tag{9}$$

which is Eq.6 in the main paper. Eq.7 in the main paper can be derived in similar way thus it's omitted here.

Derivation of Eq.8. The objective of kernel estimation is defined as

$$\mathbf{k}^t = \underset{\mathbf{k}^t}{\operatorname{argmin}}\, ||\mathbf{y} - \mathbf{k}^t \otimes \mathbf{z}^t||_2^2 + \tau^t||\mathbf{k}^t||_2^2 \tag{10}$$

This is solved by setting the gradient of the right-hand linear least squares to be zero,

$$(\mathbf{Z}_t^\top\mathbf{Z}_t + \tau^t)\mathbf{k}^t = \mathbf{Z}_t^\top\mathbf{y} \tag{11}$$

where matrix $\mathbf{Z}_t$ represents corresponding convolution with image $\mathbf{z}^t$. Eq. 11 can be efficiently solved in Fourier domain:

$$\mathbf{k}^t = \mathcal{F}^{-1}\left[\frac{\mathcal{F}(\mathbf{z}^t)^*\cdot\mathcal{F}(\mathbf{y})}{\mathcal{F}(\mathbf{z}^t)^*\cdot\mathcal{F}(\mathbf{z}^t) + \tau^t}\right] \tag{12}$$

which is Eq.8 in the main paper. $*$ represents conjugate transpose.

Calculation of Eq.9. For $\mathbf{x}^t$ update, the training loss $\ell = ||\mathbf{x}^t - \bar{\mathbf{x}}||_2^2$. Its gradient w.r.t. the model parameters $\Theta^t = (\mathbf{f}_i^t, \psi_i^t, \lambda^t)$ is computed as

$$\begin{aligned}\frac{\partial \ell}{\Theta^t} &= \frac{\partial \mathbf{x}^t}{\partial \Theta^t}\frac{\partial \ell}{\mathbf{x}^t} \\ &= 2\frac{\partial \mathbf{x}^t}{\partial \Theta^t}(\mathbf{x}^t - \bar{\mathbf{x}})\end{aligned} \tag{13}$$

In order to compute $\partial \mathbf{x}^t / \partial \Theta^t$, $\mathbf{x}^t$ in Eq. 9 can be rewritten as

$$\mathbf{x}^t = \left(\mathbf{K}_{t-1}^\mathsf{T}\mathbf{K}_{t-1} + \lambda^t \sum_{i=1}^{N} \mathbf{F}_i^{t\mathsf{T}}\mathbf{F}_i^t\right)^{-1}\left(\mathbf{K}_{t-1}^\mathsf{T}\mathbf{y} + \lambda^t \sum_{i=1}^{N} \mathbf{F}_i^{t\mathsf{T}}\psi_i^t(\mathbf{F}_i^t\mathbf{z}^{t-1})\right) \tag{14}$$

Then $\partial \mathbf{x}^t / \partial \Theta^t$ can be derived following the matrix calculus rules, and we refer the derivation details of $\partial \mathbf{x}^t / \partial \Theta^t$ to the supplemental material of [5].

Calculation of Eq.10. For $\mathbf{z}^t$ and $\mathbf{k}^t$ update, the training loss $\ell = ||\mathbf{k}^t - \bar{\mathbf{k}}||_2^2 + \alpha||\mathbf{z}^t - \bar{\mathbf{x}}||_2^2$. Its gradient w.r.t. the model parameters $\Omega^t = (\mathbf{g}_i^t, \phi_i^t, \eta^t, \tau^t)$ is computed as

$$\frac{\partial \ell}{\partial \Omega^t} = \frac{\partial \mathbf{z}^t}{\partial \Omega^t}\frac{\partial \mathbf{k}^t}{\partial \mathbf{z}^t}\frac{\partial \ell}{\partial \mathbf{k}^t} + \frac{\partial \mathbf{k}^t}{\partial \Omega^t}\frac{\partial \ell}{\partial \mathbf{k}^t} + \frac{\partial \mathbf{z}^t}{\partial \Omega^t}\frac{\partial \ell}{\partial \mathbf{z}^t} \tag{15}$$

where

$$\frac{\partial \ell}{\partial \mathbf{k}^t} = 2(\mathbf{k}^t - \bar{\mathbf{k}}) \tag{16}$$

$$\frac{\partial \ell}{\partial \mathbf{z}^t} = 2\alpha(\mathbf{z}^t - \bar{\mathbf{x}}) \tag{17}$$

In order to compute $\partial \mathbf{k}^t / \partial \mathbf{z}^t$ and $\partial \mathbf{k}^t / \partial \Omega^t$, $\mathbf{k}^t$ in Eq. 12 can be rewritten as

$$\mathbf{k}^t = (\mathbf{Z}_t^\mathsf{T}\mathbf{Z}_t + \tau^t)^{-1}(\mathbf{Z}_t^\mathsf{T}\mathbf{y}) \tag{18}$$

For brevity, we denote $(\mathbf{Z}_t^\mathsf{T}\mathbf{Z}_t + \tau^t)$ as $\mathbf{\Pi}$, and $(\mathbf{Z}_t^\mathsf{T}\mathbf{y})$ as $\mathbf{\Lambda}$, then $\mathbf{k}^t = \mathbf{\Pi}^{-1}\mathbf{\Lambda}$. We use the square bracket around any image $\mathbf{a}$, i.e. $[\mathbf{a}]$, to indicate the matrix representing convolution with $\mathbf{a}$. Moreover, we define matrix $\mathcal{R}$ representing rotating a 2D image by 180 degrees, i.e., $\mathcal{R}\mathbf{a}$ means rotating the 2D image $\mathbf{a}$ by 180 degrees. Then, we have

$$\begin{aligned}\frac{\partial \mathbf{k}^t}{\partial \mathbf{z}^t} &= \frac{\partial(\mathbf{\Pi}^{-1}\mathbf{\Lambda})}{\partial \mathbf{z}^t} \\ &= \frac{\partial \mathbf{\Lambda}}{\partial \mathbf{z}^t}\mathbf{\Pi}^{-1} - \mathbf{\Lambda}^\mathsf{T}\mathbf{\Pi}^{-1}\frac{\partial \mathbf{\Pi}}{\partial \mathbf{z}^t}\mathbf{\Pi}^{-1} \\ &= \frac{\partial(\mathbf{Z}_t^\mathsf{T}\mathbf{y})}{\partial \mathbf{z}^t}\mathbf{\Pi}^{-1} - \frac{\partial(\mathbf{\Pi}\mathbf{k}^t)}{\partial \mathbf{z}^t}\mathbf{\Pi}^{-1} \\ &= \frac{\partial([\mathbf{y}]\mathcal{R}\mathbf{z}^t)}{\partial \mathbf{z}^t}\mathbf{\Pi}^{-1} - \frac{\partial((\mathbf{Z}_t^\mathsf{T}\mathbf{Z}_t + \tau^t)\mathbf{k}^t)}{\partial \mathbf{z}^t}\mathbf{\Pi}^{-1} \\ &= \mathcal{R}^\mathsf{T}[\mathbf{y}]^\mathsf{T}\mathbf{\Pi}^{-1} - \frac{\partial(\mathbf{Z}_t^\mathsf{T}\mathbf{Z}_t\mathbf{k}^t)}{\partial \mathbf{z}^t}\mathbf{\Pi}^{-1} \\ &= \mathcal{R}^\mathsf{T}[\mathbf{y}]^\mathsf{T}\mathbf{\Pi}^{-1} - (\mathcal{R}^\mathsf{T}[\mathbf{Z}_t\mathbf{k}^t]^\mathsf{T} + \mathbf{K}_t^\mathsf{T}\mathbf{Z}_t)\mathbf{\Pi}^{-1} \\ &= (\mathcal{R}^\mathsf{T}[\mathbf{y}]^\mathsf{T} - \mathcal{R}^\mathsf{T}[\mathbf{Z}_t\mathbf{k}^t]^\mathsf{T} - \mathbf{K}_t^\mathsf{T}\mathbf{Z}_t)\mathbf{\Pi}^{-1}\end{aligned} \tag{19}$$

And,

$$\begin{aligned}\frac{\partial \mathbf{k}^t}{\partial \Omega^t} &= \frac{\partial \mathbf{\Pi}^{-1}\mathbf{\Lambda}}{\partial \tau^t} \\ &= \frac{\partial \mathbf{\Lambda}}{\partial \tau^t}\mathbf{\Pi}^{-1} - \mathbf{\Lambda}^\top \mathbf{\Pi}^{-1}\frac{\partial \mathbf{\Pi}}{\partial \tau^t}\mathbf{\Pi}^{-1} \\ &= -\mathbf{\Lambda}^\top \mathbf{\Pi}^{-1}\frac{\partial(\mathbf{Z}_t^\top \mathbf{Z}_t + \tau^t)}{\partial \tau^t}\mathbf{\Pi}^{-1} \\ &= -\mathbf{K}_t^\top \mathbf{\Pi}^{-1}\end{aligned} \tag{20}$$

By substituting Eq. 16, 17, 19, 20 into Eq. 15, we have

$$\begin{aligned}&\frac{\partial \ell}{\partial \Omega^t} = -2\mathbf{K}_t^\top \mathbf{\Pi}^{-1}(\mathbf{k}^t - \bar{\mathbf{k}}) \\ &+ 2\frac{\partial \mathbf{z}^t}{\partial \Omega^t}\left((\mathcal{R}^\top[\mathbf{y}]^\top - \mathcal{R}^\top[\mathbf{Z}_t\mathbf{k}^t]^\top - \mathbf{K}_t^\top \mathbf{Z}_t)\mathbf{\Pi}^{-1}(\mathbf{k}^t - \bar{\mathbf{k}}) + \alpha(\mathbf{z}^t - \bar{\mathbf{x}})\right)\end{aligned} \tag{21}$$

The derivation of $\partial \mathbf{z}^t / \partial \Omega^t$ is similar as $\partial \mathbf{x}^t / \partial \Theta^t$, and we refer the details to the supplemental material of [5]. We will make our training code publicly available with the paper.

2 More algorithm details

Example training data. Fig. 1 visualizes example blur kernels and images used at our training.

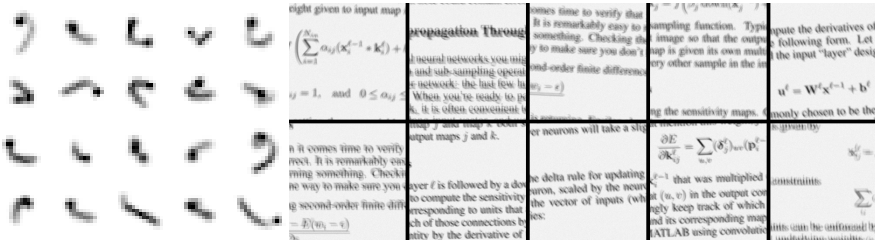


Fig. 1. Example kernels and images used at training of our algorithm.

Intermediate results. Our method uses a multi-scale approach to prevent bad local optima. Fig. 2 and 3 shows intermediate results of our estimated image $\mathbf{x}$ and kernel $\mathbf{k}$ at each scale. Note that our algorithm simultaneously estimates the latent image and blur kernel, and does not need extra non-blind deconvolution steps.

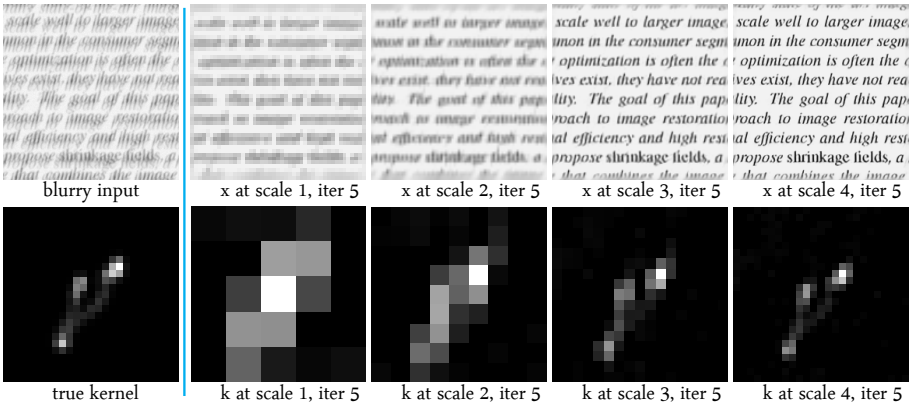


Fig. 2. Example intermediate results of our algorithm on a synthetic test image. Please zoom in for better view.

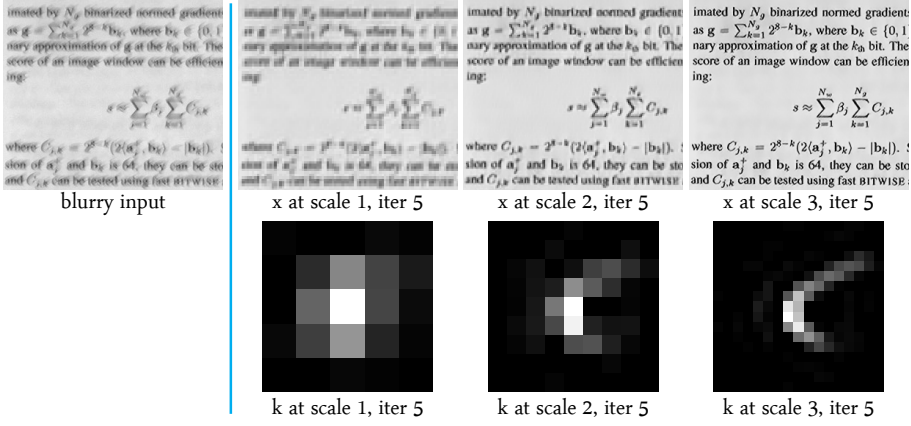
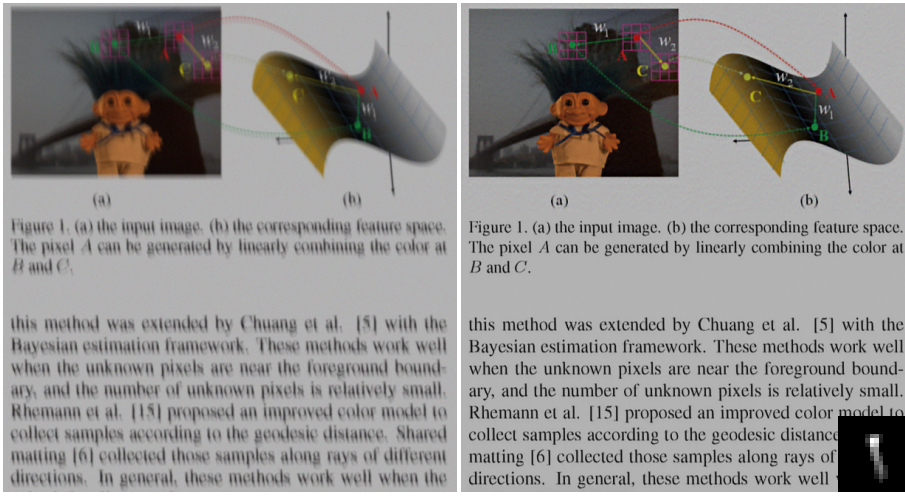


Fig. 3. Example intermediate results (cropped) of our algorithm on a real-world test image (shown in Fig.5 in the main paper). Please zoom in for better view.

Documents with color figures. In the main paper we focus on the text documents, however our algorithm can be easily extended to handle the documents with color figures, as shown in Fig. 4. First, we run our blind deblurring algorithm on the text regions to recover the latent text images and blur kernels. Second, we take the kernels estimated from the text regions around the color figures (optionally interpolate the kernels using the EFF model [2] for spatially-varying blur), and deblur the color figure regions as non-blind deconvolution. Finally, we align the text and figure regions to generate the final image.

This is a benefit of our algorithm as we jointly estimate the text image and the blur kernel, and the latter can be further used for deblurring non-text regions non-blindly. Hradiš et al. [3] does not recover the blur kernel thus cannot handle the figure regions in the document.

In our current implementation, we manually mark the figure regions for above processes. As a future work, we are interested in training a classifier to automatically predict the figure regions directly from the blurry input.



(a) Blurry input (b) Our deblurred image

Fig. 4. Result on example document containing color figures. The right-bottom corner in (b) shows the blur kernel estimated from the text regions. In this example we simply use [4] for the non-blind deblurring step, although [5] can be used instead for improved results.

References

1. Geman, D., Yang, C.: Nonlinear image recovery with half-quadratic regularization. Image Processing, IEEE Transactions on 4(7), 932–946 (1995)
2. Hirsch, M., Sra, S., Scholkopf, B., Harmeling, S.: Efficient filter flow for space-variant multiframe blind deconvolution. In: CVPR 2010
3. Hradiš, M., Kotera, J., Zemcík, P., Šroubek, F.: Convolutional neural networks for direct text deblurring. In: BMVC 2015
4. Krishnan, D., Fergus, R.: Fast image deconvolution using hyper-laplacian priors. In: NIPS 2009
5. Schmidt, U., Roth, S.: Shrinkage fields for effective image restoration. In: CVPR 2014