LEARNING BLIND DENOISING NETWORK FOR NOISY IMAGE DEBLURRING

Meiya Chen *Yi Chang* *Shuning Cao* *Luxin Yan*⋆

School of Artificial Intelligence and Automation, Huazhong University of Science and Technology

ABSTRACT

Noisy image deblurring is to recover the blurry image in the presence of the random noise. The key to this problem is to know the noise level in each iteration. The existing methods *manually* adjust the regularization parameter for varying noise levels, which are quite inaccuracy and tedious for practical application. In this work, we discover that the noise level and the denoiser are tightly coupled. Consequently, we propose a blind denoising convolutional neural network (BDCNN) consisting of two stages: a down-sampling regression network for estimating noise level and a fully convolutional network for denoising, such that our model could *adaptively* handle the unknown noise level during iterations. Further, the BDCNN functions as a discriminative prior and is plugged into the iterative deblurring framework for noisy image deblurring. Experimental results demonstrate that the proposed method outperforms state-of-the-art methods in terms of practicability and performance.

Index Terms— Noisy image deblurring, blind denoising network, plug-and-play, iterative deblurring framework

1. INTRODUCTION

Image deblurring has been extensively studied and achieved great success in the past decades. The blurred degradation procedure can be formulated as $\boldsymbol{y} = \boldsymbol{K}\boldsymbol{x} + \boldsymbol{n}$, where $\boldsymbol{y}$, $\boldsymbol{K}$, $\boldsymbol{x}$ and $\boldsymbol{n}$ are blurry image, blurred operator, latent image and additive noise, respectively. Many natural image prior-based approaches [1, 2, 3, 4, 5, 6, 7] have been proposed to solve the deblurring problem, while they only focus on deblurring the scenario with slight noise. Blurry image with considerable noise is also common in real photography.

To address the noisy image deblurring problem, Aram *et al.* [8] proposed the iterative optimization framework, in which the image deblurring problem was decomposed into two iterative sub-problems including deconvolution and denoising. Since the deconvolution sub-problem is easy to obtain the closed-form solution, they [8, 9, 10, 11, 12] focused on designing a plug-and-play denoiser prior. The denoising sub-problem was formulated as a pioneering BM3D [13] frame in [8]. In addition, the abundant hand-crafted denoiser priors were designed to constrain the optimization-based method, such as total variation [14], sparsity [15, 16] and low-rank [17]. Recently, the discriminative learning approaches [18, 19] were popular to denoising. Taking advantage of learning denoiser prior, the authors [10, 11, 12] plugged the convolutional neural networks (CNN) into the iterative framework for noisy image deblurring.

The key of existing methods lies in properly designed denoiser prior. The denoiser prior consists of not only denoiser but also the regularization parameter (relate to noise level), which directly decides the degree of denoising. Most existing methods manually obtain the parameter by tedious adjustment. However, on the one hand, the empirical parameter is not accurate enough to accommodate the complicated variations of noise level in deblurring iterations, leaving the under-denoising or over-smoothing results. On the other hand, the learning-based methods are custom-trained for specific noise levels, in which a set of models are redundantly.

To overcome the two problems, we propose a novel blind denoising convolutional neural network (BDCNN) as a plug-and-play prior and integrate it into the iterative deblurring framework for noisy image deblurring. We formulate the correlation between noise level and denoiser via a two-stage guidance network, including noise level estimation and image denoising components. The first stage gradually learns the global features for estimating noise level, which provides the discriminative attention to guide the image denoising in the second stage. Benefiting from the two-stage guidance network, a single trained model achieves comparable performance as a set of specific trained model. Rather than manual adjustment, the proposed method adaptively estimates the varied regularization parameter from input image in iterations, resulting in promising performance and robustness of algorithm. The contributions can be summarized as follows:

- We incorporate the BDCNN into iterative optimization framework for noisy image deblurring. BDCNN adaptively learns the noise level and denoiser to frees the user from tedious parameter tuning during iterations, and avoids the under-denoising or over-smoothing results.
- We propose an efficient two-stage blind denoising network, cascading noise level estimation sub-network and image denoising sub-network, which could adaptively suppress noise via a single trained model.
- Qualitative and quantitative experimental results demonstrate that the proposed method outperforms state-of-the-art methods in terms of practicability and performance.

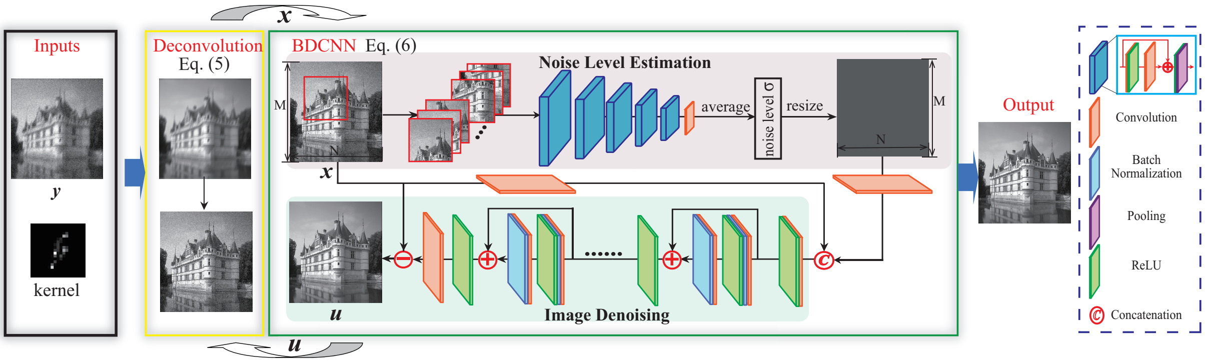


Fig. 1. The framework of proposed noisy image deblurring method. Given the blurry image and kernel, the proposed method iteratively carries out deconvolution and adaptive noise removal. The noise removal is implemented by BDCNN, which consists of noise level estimation and image denoising.

2. PROPOSED METHOD

2.1. Iterative Deblurring Framework

Maximum a *posteriori* (MAP) is a common framework to formulate the image deblurring problem. For the additive zero-mean Gaussian white noise, the minimization criterion is given as

$$\hat{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{K}\boldsymbol{x}\right\|_2^2 + \lambda\varphi\left(\boldsymbol{x}\right), \tag{1}$$

where the $\frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{K}\boldsymbol{x}\right\|_2^2$ and $\varphi\left(\boldsymbol{x}\right)$ are fidelity and prior terms respectively, and λ is a trade-off parameter.

Generally, half-quadratic splitting technique [10] can be adopted to solve Eq. (1). We introduce auxiliary variable $\boldsymbol{u}$ corresponding to $\boldsymbol{x}$, and rewrite the Eq. (1) as

$$\hat{\boldsymbol{x}}, \hat{\boldsymbol{u}} = \arg\min_{\boldsymbol{x},\boldsymbol{u}} \left\|\boldsymbol{y}-\boldsymbol{K}\boldsymbol{x}\right\|_2^2 + \lambda\varphi\left(\boldsymbol{u}\right) + \frac{\eta}{2}\left\|\boldsymbol{u}-\boldsymbol{x}\right\|_2^2, \tag{2}$$

where η is a penalty parameter. The fidelity term and prior term are decoupled into two individual subproblems. We obtain the solution for Eq. (2) by alternatively solving

$$\boldsymbol{x}_{k+1} = \arg\min_{\boldsymbol{x}} \frac{1}{2}\left\|\boldsymbol{y}-\boldsymbol{K}\boldsymbol{x}\right\|_2^2 + \eta\left\|\boldsymbol{u}_k-\boldsymbol{x}\right\|_2^2, \tag{3}$$

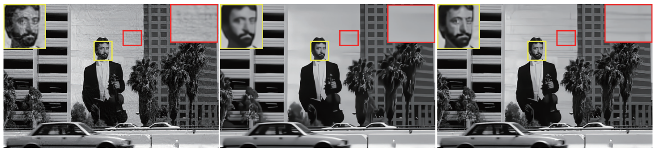
$$\boldsymbol{u}_{k+1} = \arg\min_{\boldsymbol{u}} \frac{1}{2}\left\|\boldsymbol{x}_{k+1}-\boldsymbol{u}\right\|_2^2 + \gamma\varphi\left(\boldsymbol{u}\right), \tag{4}$$

where γ is related to noise level σ. $\gamma = (\sqrt{\lambda/\eta})^2 = \sigma^2$ is the regularization parameter of Eq. (4), and k is the iteration number.

Specifically, the fidelity term is involved in Eq. (3) corresponding to the deconvolution problem, which is a quadratic regularized least-squares minimization problem. Therefore, we can obtain the fast closed-form solution through Fast Fourier transform (FFT)

$$\boldsymbol{x}_{k+1} = \mathcal{F}^{-1}\left(\frac{\overline{\mathcal{F}\left(\boldsymbol{K}\right)}\mathcal{F}\left(\boldsymbol{y}\right) + 2\eta\mathcal{F}\left(\boldsymbol{u}_k\right)}{\overline{\mathcal{F}\left(\boldsymbol{K}\right)}\mathcal{F}\left(\boldsymbol{K}\right) + 2\eta}\right), \tag{5}$$

where $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ are Fourier transform and inverse Fourier transform, respectively.



(a) under denoise (29.11, 0.8625) (b) over denoise (29.76, 0.8755) (c) proper denoise (31.21, 0.9199)

Fig. 2. Evaluation of the importance of regularization paramete. (a) is the result with smaller regularization parameter, which is under-denoising and contains many artefacts. (b) is the result with larger regularization parameter, which is over-smoothing and damages the details. (c) is the result with the proper regularization parameter used in the proposed method.

Equation (4) involving the prior term can be interpreted as a denoising problem. Instead of hand-crafted prior and manually designed regularization parameter, we develop a two-stage blind denoising network to handle the denoising problem, which can be formulated as

$$\boldsymbol{u}_{k+1} = BDCNN(\boldsymbol{x}_{k+1}). \tag{6}$$

BDCNN adaptively learns noise level from input and then denoising. We iteratively solve Eq. (5) and (6) to obtain the clean image, and the overall framework is shown in Fig. 1.

2.2. Motivation

Why blind denoising network? Most of existing methods obtain the solution of Eq. (4) via solving

$$\boldsymbol{u}_{k+1} = Denoiser(\boldsymbol{x}_{k+1}, \gamma), \tag{7}$$

where the regularization parameter γ is obtained via tedious parameter tuning. Since γ is dynamically varying with iterations, it is hard to manually adjust to obtain a promising solution of γ in each iteration. Thus, deblurring results always are affected by the inaccurate empirical parameter. As shown in Fig. 2, the artefacts and over-smoothing occur in results when the regularization parameter is lower or higher than the promising solution. To avoid these negative effects, we ex-

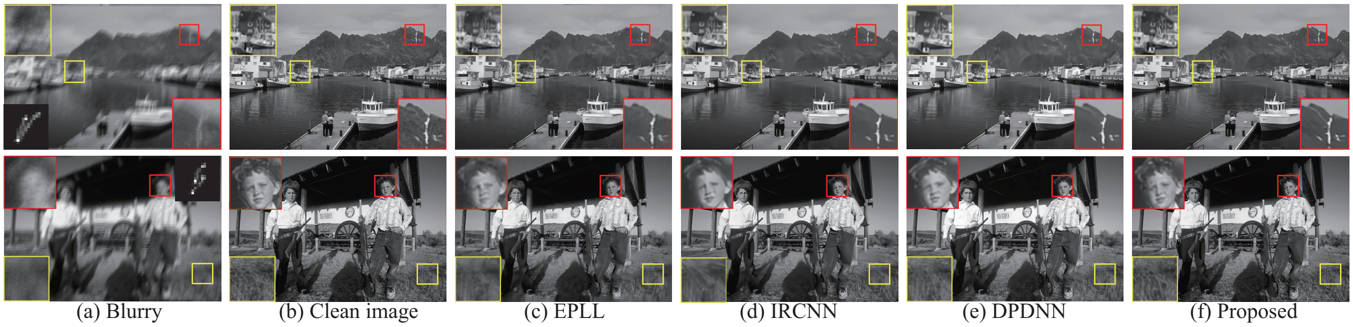

(a) Blurry (b) Clean image (c) EPLL (d) IRCNN (e) DPDNN (f) Proposed

Fig. 3. Visual comparison of deblurring results. The two rows are blurry images with additive $\sigma = 2$, and 2.55, respectively.

plicitly model the physical meaning of γ, and develop a blind denoising network to adaptively learn an accurate noise level during iterations.

Why Two-stage? We firstly look closer to the denoising problem in Eq. (4). When the prior is defined in intensity space, the optimization-based solution can be formulated as

$$\boldsymbol{u}_k = shrink\left(\boldsymbol{x}_k, \gamma\right), \tag{8}$$

where $shrink$ is the soft thresholding operator. Given the intermediate image $\boldsymbol{x}_k$ and regularization parameter γ in each iteration, soft thresholding shrinks components of the image above the threshold. Thus, $shrink$ can be interpreted as a denoiser and γ is the threshold.

Motivated by Eq. (8), we explicitly model noise level and estimate the parameter γ. Then, rather than the shallow denoising operator $shrink$, we develop a deep effective denoising network to perform denoising. Thus, we design a two-stage network inspired by optimization as follow

$$\boldsymbol{u}_k = Net_2\left(\boldsymbol{x}_k, Net_1(\boldsymbol{x}_k)\right), \tag{9}$$

where Net_1 and Net_2 are noise level estimation sub-network and image denoising sub-network, respectively. It is worth noting that Eq. (9) shares the same formulation with Eq. (8), which implies the two-stage blind denoising network. The first stage estimates the parameter γ in Eq. (8), which is fed to the second stage combining with the noisy image.

2.3. Architecture of Blind Denoising Network

Noise Level Estimation Sub-network. The optimization-based methods, such as noise level estimation algorithm [20], have achieved state-of-the-art performance. However, they separate the noise level estimation and image denoising. To jointly model the two sub-problems, we develop a gradually down-sampling regression network to estimate the noise level. Given an input image $\boldsymbol{x}_k$, we extract the sliding patches $\boldsymbol{p}_k^i$ with the size of 64×64, where the stride of patch is 64. Since each sliding patch will obtain a noise level, the final level is the average of all noise levels. The formulation of network can be written as

$$\sigma_k = Net_1(\boldsymbol{x}_k) = \frac{1}{N}\sum\nolimits_{i=1}^{N} \mathcal{F}_1(\boldsymbol{p}_k^i), \tag{10}$$

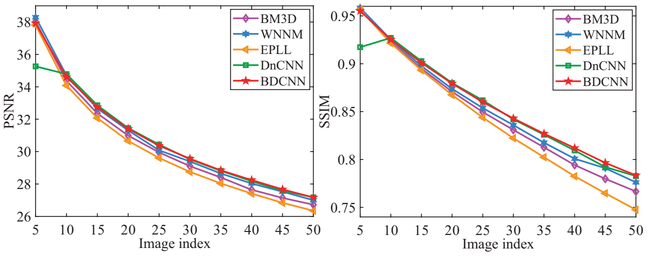


Fig. 4. Denoising results with different noise level.

where N is the number of sliding patches. The regression network can obtain a successive value of noise level, and consists of five residual blocks [21] without batch normalization operations. To ensure that the output is a single value, the plain convolution is adopted in the last layer with 2×2 filter size.

Image Denoising Sub-network. Image denoising sub-network takes both $\boldsymbol{x}_k$ and σ_k as input to obtain the final denoising result, while the two inputs have different physical meaning. To eliminate the ambiguity, we firstly resize the $\hat{\sigma}_k$ to the map with the same size of input image, and then adopt a convolutional layer $\mathcal{C}$ to extract features of image and noise level, respectively. The features are directly fed into the denoising network as follow

$$\boldsymbol{u}_{k+1} = Net_2(\boldsymbol{x}_k, \sigma_k) = \mathcal{F}_2(\mathcal{C}(\boldsymbol{x}_k), \mathcal{C}(\mathcal{S}(\sigma_k))), \tag{11}$$

where the $\mathcal{S}$ is the resize operator. Net_2 consists of five residual blocks without pooling and a plain convolution. Instead of directly predicting a clean image, we adopt the residual learning strategy as present in [19]. Except for special instructions, the number of feature channel is 64 and the filter size is 3×3.

2.4. Training Details

We train the BDCNN in an end-to-end manner. The first stage is supervised by $\mathcal{L}_{level}$, meaning the $\mathcal{L}_2$ distance between truth level and prediction level. The second stage is also supervised by $\mathcal{L}_{level}$, which means the $\mathcal{L}_2$ between clean image and restoration image. The total loss can be minimized via

$$\mathcal{L}_{total} = w_1\mathcal{L}_{level} + w_2\mathcal{L}_{image}, \tag{12}$$

Table 1. Comparative deblurring results (PSNR/SSIM) of different blur with three additive noise level.

Motion blur (19 × 19)					
σ	Blurry	EPLL [22]	IRCNN [10]	DPDNN [12]	BDCNN
2	22.92/0.5713	29.42/0.8691	31.54/0.8739	30.58/0.8698	**32.03/0.8998**
2.55	22.88/0.5600	28.73/0.8432	30.84/0.8770	30.60/0.8769	**31.08/0.8802**
7.56	22.13/0.4151	25.26/0.6826	27.32/0.7609	-	**27.33/0.7626**
Motion blur (17 × 17)					
σ	Blurry	EPLL [22]	IRCNN [10]	DPDNN[12]	BDCNN
2	22.39/0.5453	29.12/0.8615	31.37/0.8712	30.52/0.8697	**31.67/0.8926**
2.55	22.35/0.5342	28.40/0.8339	30.66/0.8575	30.59/**0.8770**	**30.73**/0.8714
7.56	21.69/0.3924	24.96/0.6743	**27.05**/0.7516	-	**27.05/0.7535**

Table 2. The accuracy evaluation of noise level estimation.

	Liu *et al.* [20]			BDCNN		
Truth σ	Avg.	RMSE	Std. dev.	Avg.	RMSE	Std. dev.
10	**10.016**	**0.128**	**0.018**	10.097	0.334	0.112
20	19.834	**0.187**	**0.038**	**20.072**	0.306	0.095
30	29.626	**0.271**	0.082	**30.071**	0.277	**0.078**
40	39.446	0.308	0.106	**40.077**	**0.202**	**0.042**
50	49.294	0.371	0.155	**49.672**	**0.213**	**0.047**

where $\mathcal{L}_{total}$ is total loss, and w_1 and w_2 denote the trade-off parameters, which are set to be 16 and 1, respectively. We separately train the two stages, and then use the total loss function to fine-tune the network.

3. EXPERIMENTAL RESULTS

Experimental Setting. We collect 432 images from Berkeley Segmentation Dataset (BSD) and 400 images from Waterloo Exploration Database [23] to generate the training dataset. Testing data contains the 68 images from BSD (BSD68) and 12 widely used testing images (Set12). During testing, the total iteration number is 20. In the training stage, we set the range of Gaussian noise levels as $\sigma \in [1, 50]$, and the patch size as 64×64 where the slide of patch is 64. The training dataset is reused five times to train a single model.

Image Denoising. Five methods are chosen for comparison, including EPLL [22], BM3D [13], WNNM [24], DnCNN [19]. Figure 4 shows the evaluation results on Set12 dataset with additive noise. The comparative denoising methods are evaluated with the ground truth. Note that, even a single trained model of proposed method also achieves comparable performance of a set of DnCNN trained models, and outperforms the other traditional methods.

Image Deblurring. Four noisy image deblurring methods, including EPLL [22], IRCNN [10] and DPDNN [12], are used. We test DPDNN on the trained model (σ=2.55) provided by the authors. We evaluate on BSD68 simulated with three noise levels and two blurs. As shown in Table 1, although the single trained model is used in the proposed method, it also outperforms the comparative methods. Figure 3 illustrates visual comparisons of the deblurring methods. The EPLL remains some blur and IRCNN tends to over-smooth details, and our method performs better in terms of removing blur and preserving details. The above results indicate that the proposed method is more robust for different noise level and outperforms the comparative methods.

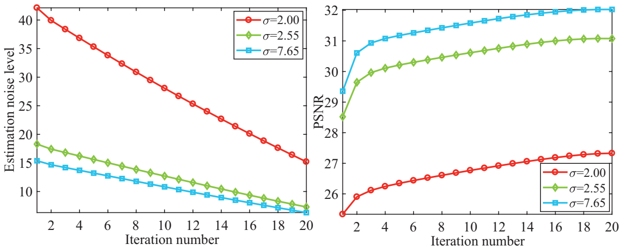


Fig. 5. The varying curves of noise level and PSNR value in iterations. The noise level is estimated after deconvolution, and PSNR is evaluated after denoising in each iteration. Note that, deconvolution will amplify the noise, resulting in much higher the noise level than the original additive noise.

Accuracy of Noise Level Estimation. We compare the noise level estimation results with Liu *et al.* [20], as shown in Table 2. The average (Avg.), root mean square error (RMSE) and standard deviation (Std. dev.) is evaluated on Set12 dataset with synthetic noise. BDCNN achieves comparable but a little worse results than Liu *et al.* for low noise level. For the high noise level, the proposed method obtains the more accurate results. In addition, the proposed method is more stable than [20] for a large range of noise level, which is essential for varying noise level during iterations.

Analysis of Deblurring Iteration. Figure 5 shows the estimation noise level and PSNR in the process of iterative deblurring. We can observe that noise level decreases approximately linearly with the iterations, which proves the rationality of parametric adjustment strategy in optimization-based methods. The rate of decline is diverse for different additive noise level, which means the single fixed strategy in existing methods is not robust for different noise level. It also partially explains why we design BDCNN to adaptively estimate the noise level in each iteration. The PSNR increases with iteration going on, and slowly changes in the last several iterations, which validates the convergence of the proposed method.

4. CONCLUSION

In this paper, we have introduced a blind denoising network under the iterative deblurring framework to address the noisy image deblurring. We argue that noise level and prior are tightly coupled and develop a two-stage network to jointly learn the noise level and denoiser prior, which improves the robustness of noise and frees the user from tedious parameter tuning. Experimental results demonstrate that the proposed method outperforms the state-of-the-art methods.

5. REFERENCES

[1] Q. Shan, J. Jia, and A. Agarwala, "High-quality motion deblurring from a single image," *ACM Trans. Graph.*, pp. 73:1–73:10, 2008.

[2] L. Xu, S. Zheng, and J. Jia, "Unnatural L_0 sparse representation for natural image deblurring," in *Proc. CVPR*, 2013, pp. 1107–1114.

[3] S. Cho and S. Lee, "Fast motion deblurring," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 145, 2009.

[4] D. Krishnan and R. Fergus, "Fast image deconvolution using hyper-Laplacian priors," in *Proc. NIPS*, 2009, pp. 1033–1041.

[5] J. Pan, Z. Hu, Z. Su, and M. H. Yang, "L_0-regularized intensity and gradient prior for deblurring text images and beyond," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 2, pp. 342–355, 2017.

[6] Jinshan Pan and Zhixun Su, "Fast l^0-regularized kernel estimation for robust motion deblurring," *IEEE Signal Process. Lett.*, vol. 20, no. 9, pp. 841–844, 2013.

[7] Jiaming Liu, Yu Sun, Xiaojian Xu, and Ulugbek S Kamilov, "Image restoration using total variation regularized deep image prior," in *Proc. ICASSP*, 2019, pp. 7715–7719.

[8] A. Danielyan, V. Katkovnik, and K. Egiazarian, "BM3D frames and variational image deblurring," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1715–1728, 2012.

[9] Dong Hye Ye, Somesh Srivastava, Jean-Baptiste Thibault, Ken Sauer, and Charles Bouman, "Deep residual learning for model-based iterative ct reconstruction using plug-and-play framework," in *Proc. ICASSP*, 2018, pp. 6668–6672.

[10] K. Zhang, W. Zuo, S. Gu, and L. Zhang, "Learning deep CNN denoiser prior for image restoration," in *Proc. CVPR*, 2017, pp. 2808–2817.

[11] J. Zhang, J. Pan, W. Lai, R. Lau, and M. Yang, "Learning fully convolutional networks for iterative non-blind deconvolution," in *Proc. CVPR*, 2017, pp. 3817–3825.

[12] Weisheng Dong, Peiyao Wang, Wotao Yin, and Guangming Shi, "Denoising prior driven deep neural network for image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2018.

[13] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, 2007.

[14] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D*, vol. 60, no. 1-4, pp. 259–268, 1992.

[15] Yi Zhang and Keigo Hirakawa, "Blind deblurring and denoising of images corrupted by unidirectional object motion blur and sensor noise," *IEEE Trans. Image Process.*, vol. 25, no. 9, pp. 4129–4144, 2016.

[16] Weihong Ren, Jiandong Tian, and Yandong Tang, "Blind deconvolution with nonlocal similarity and l_0 sparsity for noisy image," *IEEE Signal Process. Lett.*, vol. 23, no. 4, pp. 439–443, 2016.

[17] Y. Chang, L. Yan, and S. Zhong, "Hyper-laplacian regularized unidirectional low-rank tensor recovery for multispectral image denoising," in *Proc. CVPR*, 2017, pp. 4260–4268.

[18] U. Schmidt and S. Roth, "Shrinkage fields for effective image restoration," in *Proc. CVPR*, 2014, pp. 2774–2781.

[19] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Trans. Image Process.*, vol. 26, no. 7, pp. 3142–3155, 2017.

[20] X. Liu, M. Tanaka, and M. Okutomi, "Single-image noise level estimation for blind denoising," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 5226–5237, 2013.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, 2016, pp. 770–778.

[22] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. ICCV*, 2011, pp. 479–486.

[23] K. Ma, Z. Duanmu, Q. Wu, Z. Wang, H. Yong, H. Li, and L. Zhang, "Waterloo exploration database: New challenges for image quality assessment models," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 1004–1016, 2017.

[24] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *Proc. CVPR*, 2014, pp. 2862–2869.