NBD-GAP: NON-BLIND IMAGE DEBLURRING WITHOUT CLEAN TARGET IMAGES

Nithin Gopalakrishnan Nair, Rajeev Yasarla and Vishal M. Patel

Dept. of Electrical and Computer Engineering, Johns Hopkins University, MD, USA
{ngopala2, ryasarl1, vpatel36}@jhu.edu

2022 IEEE International Conference on Image Processing (ICIP) | 978-1-6654-9620-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICIP46576.2022.9897543

ABSTRACT

In recent years, deep neural network-based restoration methods have achieved state-of-the-art results in various image deblurring tasks. However, one major drawback of deep learning-based deblurring networks is that large amounts of blurry-clean image pairs are required for training to achieve good performance. Moreover, deep networks often fail to perform well when the blurry images and the blur kernels during testing are very different from the ones used during training. This happens mainly because of the overfitting of the network parameters on the training data. In this work, we present a method that addresses these issues. We view the non-blind image deblurring problem as a denoising problem. To do so, we perform Wiener filtering on a pair of blurry images with the corresponding blur kernels. This results in a pair of images with colored noise. Hence, the deblurring problem is translated into a denoising problem. We then solve the denoising problem without using explicit clean target images. Extensive experiments are conducted to show that our method achieves results that are on par to the state-of-the-art non-blind deblurring works.

***Index Terms*—** Non blind deblurring, Wiener Deconvolution, No-reference.

1. INTRODUCTION

Motion blur is a common and prominent problem that occurs in hand-held photography. It destroys the aesthetics of the image and adversely affects the performance of many computer vision applications [1, 2]. In this work, we focus on the case of uniform blur where the blurred image is represented by,

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

where y is the blurry image, x is the latent clean image, k is the blur kernel, n is the additive white Gaussian noise, and $*$ denotes the convolution operation. Existing deblurring methods that restore the clean image x can be grouped into blind and non-blind deblurring methods. Non-blind deblurring methods take both the blurry image y and the blur kernel k as input to restore x. On the other hand, blind deblurring is a more difficult problem that requires only the blurry image y as input to restore x. In this paper, we mainly focus on the non-blind deblurring problem.

Early non-blind deblurring algorithms used statistical techniques to derive minimum mean square error (MMSE) solutions such as the Wiener filter [3] and Richardson-Lucy algorithm [4]. Although these algorithms work well for very low noise levels, they add a significant amount of colored noise when the noise level increases. Also, these methods suffer from serious ringing artifacts and cannot deal with degradations caused by large motions. Another line of deconvolution algorithms focus on developing effective image priors [5, 6, 7, 8] using natural image statistics. But, these priors are very reliant on the distribution of the natural images and often lead to highly non-convex optimization problems, hence requiring expensive computational power.

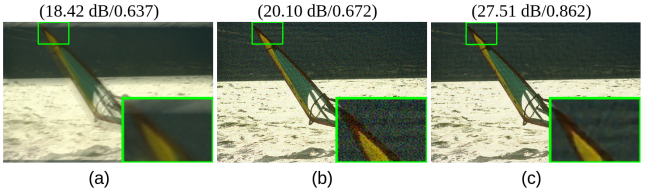


Fig. 1: Sample output from our restoration network. (a) Blurry input image. (b) Wiener filtered image. Note the artifacts and noise after Wiener filtering. (c) Restored Image.

Recently neural networks have been used for various image deblurring tasks [9, 10, 11] and have achieved state-of-the-art results in different deblurring scenarios [12, 13, 10]. There are two major techniques by which deep learning has been adopted for non-blind deblurring. One line of works develop a model which uses both the blurred image and the kernel as input to address the deblurring problem [14, 15]. Another line of works decomposes the problem into a denoising problem using a deconvolution algorithm or deep neural network. Then address the deconvolution and denoising problems separately [16, 9, 17]. These techniques require large amounts of paired data, i.e., blurred images and their corresponding clean target pairs, to train the deep networks. These algorithms also fail to generalize well when the blur kernels during testing differ from the training kernels. Non-blind deblurring methods are often used as a black-box module for solving the blind deblurring algorithms [18, 19, 20, 21] where the kernels and the latent images are found iteratively until the latent clean image is restored. Hence, there exists a need to develop non-blind deblurring algorithms that generalize over unseen images and kernels and learn with less data. To this end, we propose an approach that can be trained without clean target images and can generalize well to unseen data and kernels.

In this work, we follow the second line of techniques as mentioned above and propose a Non-blind image deblurring technique. Our method doesn't require the actual clean target data corresponding to the blurry input images. Specifically, we first perform Wiener filtering to deblur two different blurry images of the same scene and get a pair of images with colored noise. Hence, we decompose the deblurring problem into a denoising problem. The key idea behind our denoising network is motivated by the work of Lehtinan et al. [22] where the authors have performed denoising of images corrupted by both Gaussian as well as colored noise without explicit training using clean target images. The main idea is that when a network is trained with multiple pairs of noisy images of the same latent scene, the expected loss between the network output and the noisy target is approximately equal to the expected loss between the network output and the corresponding clean target. The network hence

This work was supported by the NSF CAREER award 2045489.

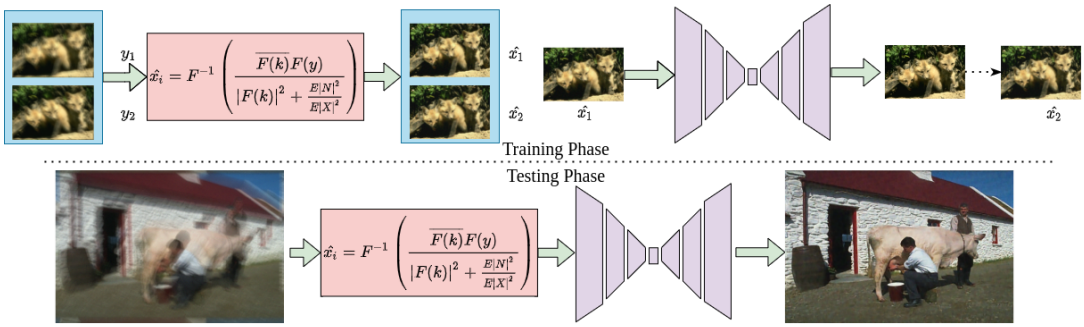


Fig. 2: An overview of the proposed network. During training, two input blurred images are processed through Wiener filtering to produce the corresponding two noisy images. Then these two noisy images are used as input and targetto train a network. During testing, for restoration, a single image is Wiener filtered and passed through the denoising network to obtain the restored image.

learns the distribution of noise and can perform denoising. Inspired by this, we utilize Wiener deconvolution to produce a pair of images with zero centered colored noise. We then treat one of the outputs of Wiener deconvolution as source and the other as the target to train our network.

2. PROPOSED METHOD

2.1. Deconvolution as a denoising problem

Given y and k in (1), our objective is to restore x. A simple solution to this problem is finding an inverse filter g where,

$$\hat{x} = g * y, \tag{2}$$

such that we can make an estimate of the clean latent image represented by $\hat{x}$. The ideal solution for $\hat{x}_{ideal}$ minimizes the expected mean square error between the clean latent image x and the estimate

$$\hat{x}_{ideal} = \min \; E|x - \hat{x}|^2. \tag{3}$$

This inverse filtering problem can be easily solved in the Fourier domain and the solution for the inverse filter g is the Wiener filter defined by

$$G = \frac{K^H}{|K|^2 + \frac{E|X|^2}{E|N|^2}}, \tag{4}$$

where G, K, X, N denote the variables in the Fourier domain and H denotes the Hermitian operator (detailed derivation is given in the supplementary document). The corresponding solution turns out to be $\hat{x}_{ideal} = x + n_c$, where

$$n_c = IFFT\left(\frac{K^H N}{|K|^2 + \frac{E|N|^2}{E|X|^2}}\right). \tag{5}$$

Here, n_c represents the inverse Fourier transform of the colored noise present along with the image after Wiener filtering. Now, since $\frac{E|N|^2}{E|X|^2}$ is an even function, the Fourier transform of n_c is odd, and the inverse Fourier transform is also odd. Hence, its mean is zero. Thus the output of Wiener filtering contains the latent image corrupted with zero mean colored random noise. Hence by performing Wiener filtering, the deblurring problem converts into a denoising problem. One crucial aspect of Wiener filtering is estimating the noise to signal ratio (NSR) term defined by $\frac{E|N|^2}{E|X|^2}$. In our work, similar to [17], we first find a median filtered estimate of the blurred image and compute its variance with the original blurred image to estimate NSR. Multiple works have previously taken this approach to simplify the deconvolution problem to a denoising problem by utilizing various transformations such as shearlet [23], and wavelet [24].

Given blurry images y with the corresponding blur kernels k, we first convert the deconvolution task into a denoising by applying Wiener filtering as follows,

$$\hat{x} = F^{-1}\left(\frac{\overline{F(k)}F(y)}{|F(k)|^2 + \frac{E|N|^2}{E|X|^2}}\right), \tag{6}$$

where F denotes the Fourier transform. In practical scenarios, the output of Wiener filtering is often corrupted with ringing artifacts along with the coloured noise. The colored noise could be removed to an extent by training the network using another set of noisy images as targets as the output like in the case of Lehtinan *et al*[22].

2.2. Network architecture

The overall pipeline of our network is shown in Figure 2. During the training process, we create two different blurry images using different blur kernels and random noise signals. Then, each of these blurred images are Wiener filtered to generate two images of the same latent scene with different random coloured noise. We use a simple U-net based architecture as our base network. We then use one of these images as the input and the other image as target output of the network.

2.3. Loss function

Let I_n and $\hat{I}_2$, denote two noisy images of dimensions $C \times H \times W$. The loss function used to train our network is

$$\mathcal{L}_2 = \frac{1}{CHW}\sum_{i,j,k}\|I_1 - \hat{I}_2\|^2. \tag{7}$$

3. EXPERIMENTS

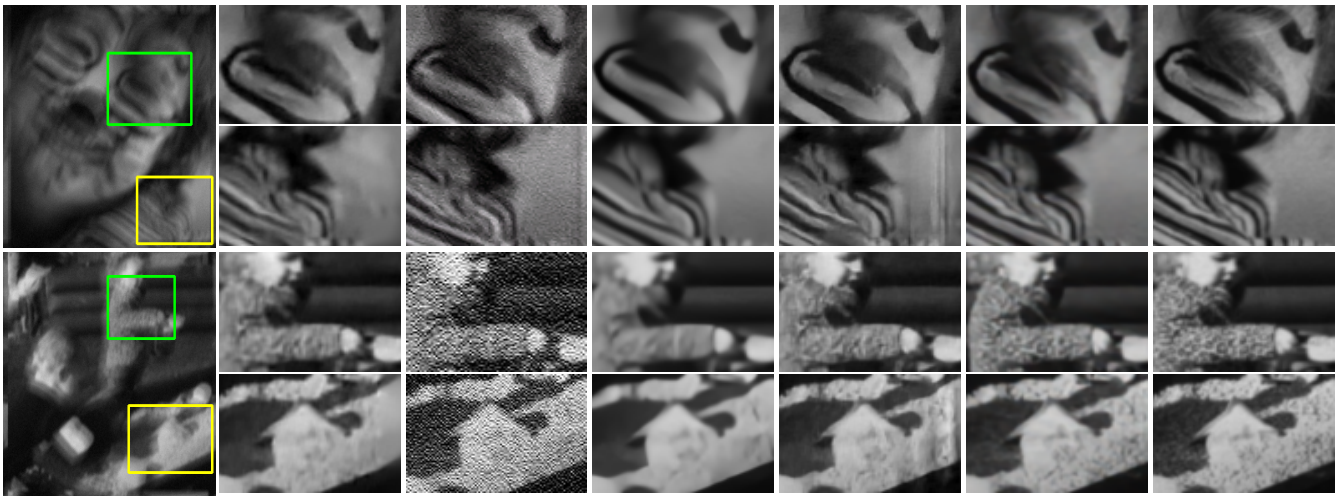
We use a batch size of 8 and use the AdamW optimizer with parameters $\beta_1 = 0.9$ and$\beta_2 = 0.999$. We set the initial learning rate to 10^{-4} and reduce it by half every 10 epochs. We train our network for a total of 60 epochs. ALl the experiments are run in

3.1. Training Dataset

For creating our training dataset, we follow the approach that has been followed in multiple non-blind deblurring works [11, 9, 28, 17]. We create a dataset of 1,000 images consisting of 600 images from the Waterloo exploration dataset [30] and 400 images from the Berkeley Segmentation dataset [31]. We synthesize kernels of random sizes varying from 11×11 to 35×35 according to the method proposed by [32] and convolve them with a patch of size 256×256 cropped from a clean image. We finally apply additive Gaussian

Table 1: Quantitative evaluation on Levin et al[25] dataset consisting of 32 images in terms of PSNR, SSIM. Please note that all the deep learning based methods used for comparison ad achieves better result than us in the $\sigma = 12.75$ dataset is a supervised technique. MD-Motion deblurring, NBD- Non-Blind Deblurring.

Noise	Metrics	Supervised-MD	Supervised-NBD					No clean target training-NBD	
		Kupyn[26]	Gong [27]	Son [17]	IRCNN[28]	Zhang[9]	Kruse[15]	EPLL [29]	OURS
$\sigma = 2.55$	PSNR	24.52	32.32	31.56	32.94	32.87	33.68	33.72	**34.97**
	SSIM	0.712	0.913	0.896	0.912	0.919	0.922	0.929	**0.950**
$\sigma = 7.65$	PSNR	23.97	29.22	28.95	30.51	29.55	29.80	29.05	**30.21**
	SSIM	0.641	0.852	0.837	0.875	0.860	0.857	0.845	**0.883**
$\sigma = 12.75$	PSNR	22.21	27.13	27.35	**27.92**	27.77	27.95	26.53	26.91
	SSIM	0.542	0.791	0.802	0.821	0.814	0.812	0.778	**0.822**



Noisy Image | EPLL [29] | Son [17] | IRCNN [28] | Kruse [15] | NBD-GAP(ours) | Ground-Truth

Fig. 3: Qualitative evaluations on the Levin et al dataset [25] for $\sigma = 2.55$ noise level.

noise of standard deviation $\{2.55, 7.65, 12.75\}$ to form blurry images.

3.2. Testing Dataset

For testing our network, we use three different testing datasets. Similar to the approaches [11, 9, 17], we use the popular benchmark datasets from Levin et al. [25]. Levin et al. [33] consists of 4 grayscale images respectively. We utilize the 8 standard kernels released by [25] and blur these images and add noise of different variances in the range of $\{2.55, 7.65, 12.75\}$ to create blurry images. We also create a synthetic dataset using the 100 test images of the BSD dataset [31] and blur it with a random blur kernel generated using [32]. Note that the kernels used for the testing are different from the ones used for training.

3.3. Results

We evaluate the performance of our network by comparing qualitatively and quantitatively with different supervised works in non-blind deblurring. For comparison, we use a combination of classical non-blind deblurring algorithms [29], supervised deep networks designed for non-blind deblurring [9, 28, 17], methods which utilize a combination of supervised deep learning and optimization algorithms [27] and one supervised motion deblurring network [26]. Peak signal to noise ratio (PSNR) and structural similarity index measure (SSIM) are used to measure the performance of different methods quantitatively.

3.3.1. Results on the Levin et al.[25] dataset

Table.1 shows the quantitative results for different noise levels in the dataset prepared using images and kernels released by Levin et al. [25]. We can from Figure 3 that conventional patch-based method EPLL [29] but creates a small amount of unwanted artifacts. Son et al. [17] converts the deblurring problem to a denoising problem, similar to us. But we can see that some noise is still retained in the output. Note that the deep learning-based method IRCNN [28] achieves better results than us in the $\sigma = 7.65$ noise case. However, IRCNN is a supervised method and is trained with a lot of paired training data.

3.3.2. Results on the BSD500 dataset [31]

Since both Levin et al.[25] and Sun et al. [34] consist of gray scale images, to evaluate the performance on coloured images, we generate a test set using 100 test images from [31] using kernels from [32] for three different noise levels. As we can see in Table 2 and Fig. 4, our method is able to perform well and produce results with fewer

Table 2: Quantitative evaluation on 500 blurry images generated from the BSD500 dataset [31] in terms of PSNR, SSIM. Please note that all the deep learning-based methods used for comparison achieve better results than us in the $\sigma = 7.65$ noise case are supervised techniques. MD denotes Motion deblurring and NBD denotes Non-Blind Deblurring.

Noise	Metrics	Supervised-MD	Supervised-NBD					Self-supervised-NBD	
		Kupyn[26]	Gong [27]	Son [17]	IRCNN[28]	Zhang[9]	Kruse[15]	EPLL [29]	OURS
$\sigma = 2.55$	PSNR	23.25	27.61	27.40	27.85	27.55	27.22	27.49	**30.27**
	SSIM	0.712	0.841	0.831	0.854	0.840	0.825	0.839	**0.862**
$\sigma = 7.65$	PSNR	23.97	25.73	25.51	25.63	26.70	25.61	25.62	**26.81**
	SSIM	0.641	0.751	0.729	0.758	**0.802**	0.738	0.746	0.761
$\sigma = 12.75$	PSNR	22.21	24.35	24.42	24.66	24.60	24.53	24.71	**24.82**
	SSIM	0.542	0.649	0.641	0.660	0.652	0.677	**0.688**	0.685

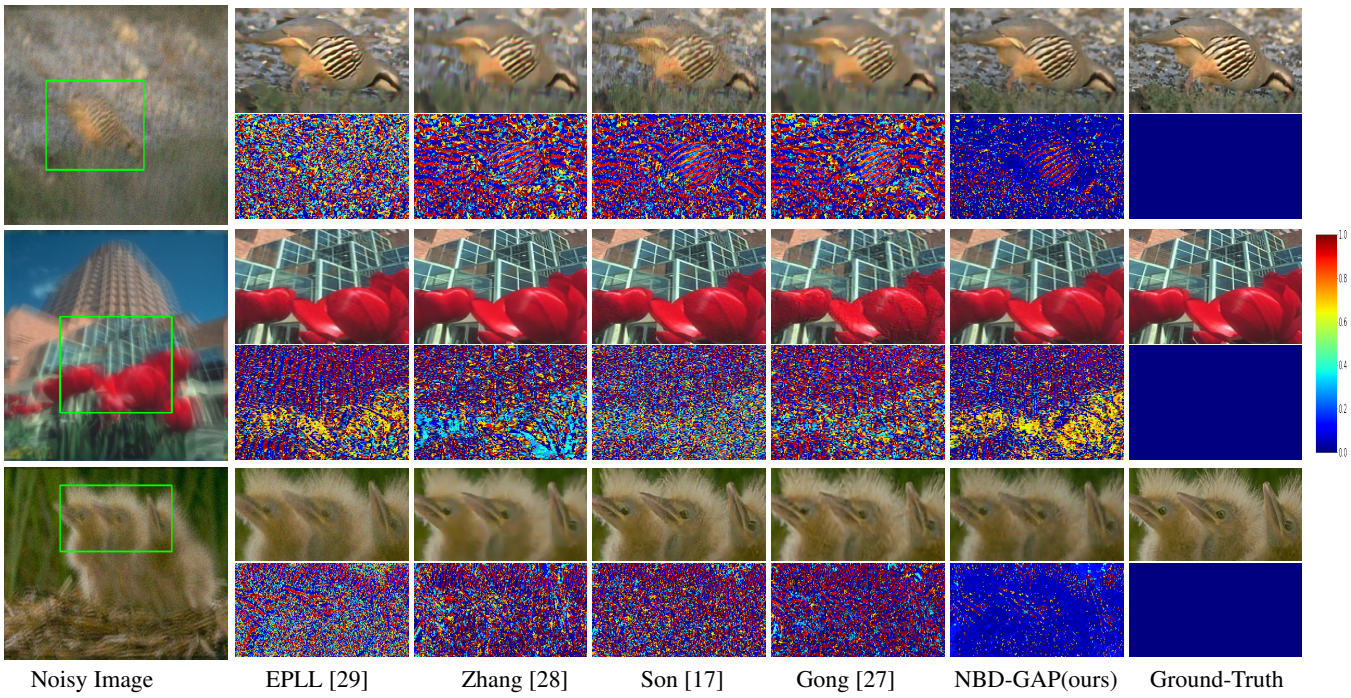


Fig. 4: Qualitative evaluations on the BSD500 dataset [25] for $\sigma = 7.65$ noise level and $\sigma = 12.75$ noise level. The First two figures from top represent images of $\sigma = 7.65$ noise level and the third figure has $\sigma = 12.75$ noise level. The corresponding absolute error map from the ground truth is shown below each cropped patch.

artifacts and noise compared to [17]. Our method is even able to produce good results at higher levels of noise. The residual maps show that the absolute error is much lesser in our method when compared to the other techniques.

4. CONCLUSION

In this paper, we proposed the first work in non-blind deblurring, where we train a deep network without clean targets. We achieve this by converting the deblurring problem to a denoising problem using Wiener filtering. We perform extensive experiments on multiple datasets to show the relevance of our network and show that our network can generalize well over multiple datasets. Furthermore, it is shown that our network is able to produce good results even at higher levels of noise.

5. REFERENCES

[1] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas, "Deblurgan: Blind motion deblurring using conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8183–8192.

[2] Igor Vasiljevic, Ayan Chakrabarti, and Gregory Shakhnarovich, "Examining the impact of blur on recognition by convolutional networks," *arXiv preprint arXiv:1611.05760*, 2016.

[3] Norbert Wiener, "Extrapolation, interpolation and smoothing of stationary," *Time Series, with Engineering Appli cations*, 1949.

[4] William Hadley Richardson, "Bayesian-based iterative method of image restoration," *JoSA*, vol. 62, no. 1, pp. 55–59, 1972.

[5] Dilip Krishnan and Rob Fergus, "Fast image deconvolution using hyper-laplacian priors," *Advances in neural information processing systems*, vol. 22, pp. 1033–1041, 2009.

[6] Antoni Buades, Bartomeu Coll, and J-M Morel, "A non-local algorithm for image denoising," in *2005 IEEE Computer Society Confer-*

ence on Computer Vision and Pattern Recognition (CVPR'05). IEEE, 2005, vol. 2, pp. 60–65.

[7] Stefan Roth and Michael J Black, "Fields of experts: A framework for learning image priors," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*. IEEE, 2005, vol. 2, pp. 860–867.

[8] Libin Sun, Sunghyun Cho, Jue Wang, and James Hays, "Good image priors for non-blind deconvolution," in *European conference on computer vision*. Springer, 2014, pp. 231–246.

[9] Jiawei Zhang, Jinshan Pan, Wei-Sheng Lai, Rynson WH Lau, and Ming-Hsuan Yang, "Learning fully convolutional networks for iterative non-blind deconvolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3817–3825.

[10] MR Mahesh Mohan, GK Nithin, and AN Rajagopalan, "Deep dynamic scene deblurring for unconstrained dual-lens cameras," *IEEE Transactions on Image Processing*, vol. 30, pp. 4479–4491, 2021.

[11] Jiangxin Dong, Stefan Roth, and Bernt Schiele, "Deep wiener deconvolution: Wiener meets deep learning for image deblurring," *arXiv preprint arXiv:2103.09962*, 2021.

[12] Dawit Mureja Argaw, Junsik Kim, Francois Rameau, and In So Kweon, "Motion-blurred video interpolation and extrapolation," in *AAAI Conference on Artificial Intelligence*, 2021.

[13] Jiangxin Dong, Stefan Roth, and Bernt Schiele, "Learning spatially-variant map models for non-blind image deblurring," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 4886–4895.

[14] Li Xu, Jimmy S Ren, Ce Liu, and Jiaya Jia, "Deep convolutional neural network for image deconvolution," *Advances in neural information processing systems*, vol. 27, pp. 1790–1798, 2014.

[15] Jakob Kruse, Carsten Rother, and Uwe Schmidt, "Learning to push the limits of efficient fft-based image deconvolution," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 4586–4594.

[16] Christian J Schuler, Harold Christopher Burger, Stefan Harmeling, and Bernhard Scholkopf, "A machine learning approach for non-blind image deconvolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1067–1074.

[17] Hyeongseok Son and Seungyong Lee, "Fast non-blind deconvolution via regularized residual networks with long/short skip-connections," in *2017 IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2017, pp. 1–10.

[18] Wei-Sheng Lai, Jia-Bin Huang, Zhe Hu, Narendra Ahuja, and Ming-Hsuan Yang, "A comparative study for single image blind deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1701–1709.

[19] Sunghyun Cho and Seungyong Lee, "Fast motion deblurring," in *ACM SIGGRAPH Asia 2009 papers*, pp. 1–8. ACM, 2009.

[20] Subeesh Vasu, Venkatesh Reddy Maligireddy, and AN Rajagopalan, "Non-blind deblurring: Handling kernel uncertainty with cnns," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3272–3281.

[21] Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T Roweis, and William T Freeman, "Removing camera shake from a single photograph," in *ACM SIGGRAPH 2006 Papers*, pp. 787–794. ACM, 2006.

[22] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila, "Noise2noise: Learning image restoration without clean data," *arXiv preprint arXiv:1803.04189*, 2018.

[23] Vishal M Patel, Glenn R Easley, and Dennis M Healy, "Shearlet-based deconvolution," *IEEE Transactions on Image Processing*, vol. 18, no. 12, pp. 2673–2685, 2009.

[24] Ramesh Neelamani, Hyeokho Choi, and Richard Baraniuk, "Forward: Fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Transactions on signal processing*, vol. 52, no. 2, pp. 418–433, 2004.

[25] Anat Levin, Yair Weiss, Fredo Durand, and William T Freeman, "Understanding and evaluating blind deconvolution algorithms," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 1964–1971.

[26] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang, "Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8878–8887.

[27] Dong Gong, Zhen Zhang, Qinfeng Shi, Anton van den Hengel, Chunhua Shen, and Yanning Zhang, "Learning deep gradient descent optimization for image deconvolution," *arXiv preprint arXiv:1804.03368*, 2018.

[28] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei Zhang, "Learning deep cnn denoiser prior for image restoration," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3929–3938.

[29] Daniel Zoran and Yair Weiss, "From learning models of natural image patches to whole image restoration," in *2011 International Conference on Computer Vision*. IEEE, 2011, pp. 479–486.

[30] Kede Ma, Zhengfang Duanmu, Qingbo Wu, Zhou Wang, Hongwei Yong, Hongliang Li, and Lei Zhang, "Waterloo exploration database: New challenges for image quality assessment models," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 1004–1016, 2016.

[31] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*. IEEE, 2001, vol. 2, pp. 416–423.

[32] Uwe Schmidt, Jeremy Jancsary, Sebastian Nowozin, Stefan Roth, and Carsten Rother, "Cascades of regression tree fields for image restoration," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 4, pp. 677–689, 2015.

[33] Anat Levin, Rob Fergus, Frédo Durand, and William T Freeman, "Image and depth from a conventional camera with a coded aperture," *ACM transactions on graphics (TOG)*, vol. 26, no. 3, pp. 70–es, 2007.

[34] Libin Sun, Sunghyun Cho, Jue Wang, and James Hays, "Edge-based blur kernel estimation using patch priors," in *IEEE International Conference on Computational Photography (ICCP)*. IEEE, 2013, pp. 1–8.