COLOR IMAGE DEMOSAICKING VIA DEEP RESIDUAL LEARNING

Runjie Tan[1], Kai Zhang[2,1], Wangmeng Zuo[2], Lei Zhang[1,]*

[1]Department of Computing, The Hong Kong Polytechnic University, Hong Kong, CHINA

[2]School of Computer Science and Technology, Harbin Institute of Technology, Harbin, CHINA

{csrjtan, cslzhang}@comp.polyu.edu.hk, zhkmath@163.com, cswmzuo@gmail.com

ABSTRACT

Color demosaicking plays a key role in digital imaging with a color filter array. Most existing demosaicking methods are based on hand-crafted priors, which may exhibit unpleasant visual artifacts in hard cases (e.g., regions with high color saturation and sharp color transition). This paper presents a customized convolutional neural network (CNN), which is trained in an end-to-end manner from natural color images to address the color demosaicking problem. Specifically, by utilizing the residual learning strategy, our network learns the demosaicking prior with a two-stage architecture: the first stage aims to recover an intermediate result of the G channel as guidance prior, while the second stage uses the intermediate G channel information to guide the reconstruction of final color demosaicking result. Our experimental results on the widely-used Kodak and McMaster datasets and a new dataset demonstrate that the proposed CNN model not only yields superior results to state-of-the-art demosaicking algorithms both quantitatively and qualitatively, but also enjoys a fast demosaicking speed by GPU computation.

Index Terms— Color Demosaicking, Convolutional Neural Network, Demosaicking Dataset, Residual Learning

1. INTRODUCTION

Most digital cameras adopt a single CCD/CMOS sensor with a color filter array (CFA) to capture the natural scenes. In order to reconstruct a full color image from the mosaicked CFA image, the missing color components need to be recovered by a process called color demosaicking (CDM). Since color demosaicking locates at the early stage of the digital imaging pipeline, it will largely affect quality of acquired color images. Among the many CFA patterns, the solution suggested by Bayer [1] in 1976, is still among the most popular and efficient one.

The CDM algorithms have been well-studied in the past decades [2]. At the early stage, basic interpolation methods such as bilinear interpolation [3] have been used for

CDM. Later on, researchers have been focusing on introducing various image priors (e.g., inter-channel correlation [4–7], sparsity [8] and non-local similarity [8, 9]) into the CDM algorithms. For a better study of inter-channel correlation, Zhang *et al.* [5] developed an adaptive filtering method by directional linear minimum mean-square error estimation (DLMMSE). Mairal *et al.* [8] proposed a learned simultaneous sparse coding (LSSC) method to learn the sparse and self-similarity priors of images. In order to exploit the non-local image redundancy, Zhang *et al.* [9] developed a demosaicking method by local directional interpolation and non-local adaptive thresholding (LDI-NAT). Recently, the residual interpolation (RI) based methods [7, 10, 11] have been proposed to interpolate the differences between observed and tentatively estimated pixel values, which explore the image sparse residual prior.

However, the above-mentioned hand-crafted prior learning methods can still exhibit obvious artifacts in the demosaicking results. As a data-driven model, the convolutional neural networks (CNNs) [13] have shown their effectiveness in both high-level and low-level vision problems. The deep residual learning network [15] has been successfully applied to image recognition and restoration applications with a very deep network architecture. The CDM problem can be formulated as a deep residual learning procedure; however, the study of using deep residual learning for CDM is still lacking. In Bayer pattern based CDM, since the number of captured G pixels is twice of that of the R and B pixels, most state-of-the-art CDM methods [4–7, 9–11] restore the G channel first, followed by the reconstruction of R/B channels. Such demosaicking domain knowledge can be integrated into the design of CNN architectures for a more effective CDM algorithms.

In this paper, we address the CDM problem by learning a deep residual convolutional neural network that can be trained in an end-to-end manner. The powerful CNN model allows us to learn the adaptively demosaicking priors directly from a large amount of training images rather than learning or predefining some hand-crafted priors. Particularly, in order to integrate the demosaicking domain knowledge into the CNN, we design a two-phase network architecture to explore the G-guidance prior for R and B channel reconstruction. To ac-

celerate the training, the first stage only reconstructs the intermediate estimation of G channel, while the second stage reconstructs the R and B channels with the guidance of reconstructed G channel. The G channel will also be further refined in the second stage.

The contribution of this work is summarized as follows. (1) We propose an end-to-end deep residual demosaicking model by taking advantage of the recent development of CNN technologies (2) We design a customized CNN model for CDM, which adopts a two-stage architecture to incorporate the demosaicking domain knowledge. Specifically, the network first constrains the G channel and then restores the full color images with the guidance of tentative G image. (3) We present a new dataset for more comprehensively evaluating the CDM algorithms. Experiments show that our method significantly outperforms state-of-the-arts on the Kodak, Mc-Master, and the new dataset both quantitatively and qualitatively.

2. METHOD

Recent years have witnessed the great success of CNN in a wide variety of challenging applications. Such a success can be attributed to the advances of deep CNN models and learning strategies, as well as the availability of large scale visual data. The representative achievements include the proposals of Rectified Linear Unit (ReLU) [13], batch normalization (BN) [14] and residual learning [15]. Other factors such as the efficient optimization algorithms, implementation on modern GPUs, and the easy access to large scale datasets are also very important. In the following, we first discuss the modeling of CDM problem and then illustrate our deep demosaicking network from architecture design and network training.

2.1. Demosaicking Modeling

The CDM problem can be modeled as maximum a posteriori (MAP) estimation problem. In particular, by using the Markov random fields inference as the analysis-based prior, in [17] the CDM is formulated as the following optimization problem:

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{y} - \mathbf{M} \odot \mathbf{x}\|^2 + \lambda \sum_{k=1}^{K} \sum_{p=1}^{N} \rho_k((\mathbf{f}_k * \mathbf{x})_p) \qquad (1)$$

where $\mathbf{y}$ is the CFA image, $\mathbf{x}$ is the desired full-color image, $\mathbf{M}$ is a 3-dimensional matrix with binary elements indicating the missing color values in Bayer pattern, '$\odot$' denotes element-wise multiplication, N denotes the image size, λ is the regularization parameter, $\mathbf{f}_k * \mathbf{x}$ stands for the convolution of the image $\mathbf{x}$ with the k-th filter kernel $\mathbf{f}_k$, and $\rho_k(\cdot)$ represents the k-th penalty function.

The gradient descent algorithm can be employed to solve the optimization problem in Eqn. (1). For CDM, it is natural to assume that $\mathbf{M} \odot \mathbf{x}_t = \mathbf{M} \odot \mathbf{x}_{t-1} = \mathbf{M} \odot \mathbf{y}$. By setting the step parameter of gradient descent to be 1, the t-th inference step can be given by

$$\mathbf{x}_t = \mathbf{x}_{t-1} - \lambda \sum_{k=1}^{K} (\bar{\mathbf{f}}_k * \phi_k(\mathbf{f}_k * \mathbf{x}_{t-1})) \qquad (2)$$

where $\bar{\mathbf{f}}_k$ is the adjoint filter of $\mathbf{f}_k$ (i.e., $\bar{\mathbf{f}}_k$ is obtained by rotating 180 degrees the filter $\mathbf{f}_k$ and $\rho'_k(\cdot) = \phi_k(\cdot)$.

From Eqn. (2), we can see that $\mathbf{x}_t$ is updated by using a two-layer network to learn the residual $\mathbf{x}_t - \mathbf{x}_{t-1}$. This motivates us to combine the proposed network with the residual learning formulation [15] for CDM. By letting $\bar{\mathbf{f}}_k$ be an independent filter to $\mathbf{f}_k$; and both of them can be learned from data, we can easily increase the network depth and adopt ReLU for nonlinearity.

The above analysis indicates intuitively the relation between MAP inference of CDM and residual learning, which is helpful to the design of our proposed demosaicking network architecture. Our model is not strictly based on but is similar in spirit to the MAP-based inference. Moreover, the recent achievements of CNN can be utilized to train a powerful CDM network. These properties make our model very promising in terms of both quantitative metrics and computational efficiency.

2.2. Network Architecture

In color demosaicking with Bayer pattern CFA, 50% of the pixels are sampled from the G channel, while 25% of the pixels are sampled from the R and B channels, respectively. Since the G channel has more samples than R/B channels, in general it will have much better reconstruction accuracy than R/B channels. On the other hand, the high correlations among R/G/B channels are crucial factors for achieving accurate CDM results. Therefore, in many traditional CDM algorithms [4–7, 9–11], the G channel is recovered first, then it is used to guide the reconstruction of R/B channels. In order to incorporate this domain knowledge into the CNN, we propose a two-phase scheme for the CDM. Moreover, in order to adopt the residual learning network structure, a simple CDM algorithm can be used to initialize the full color image so that the residual image can be computed. Instead of directly outputting the demosaicked image, the proposed network is designed to predict the residual image, i.e., the difference between the initialized full color image and the desired ground-truth full color image.

Fig. 1 shows our proposed CNN architecture for CDM. The proposed network contains two basic modules of K-layer CNNs, stacked by convolutional layers, batch normalization and ReLU nonlinearity layers. For each module, the first layer uses 64 filters of size 3×3 to generate 64 feature maps, while the last convolutional layer adopts the filter of size 3×3×64 to generate the corresponding output. In the hidden layers
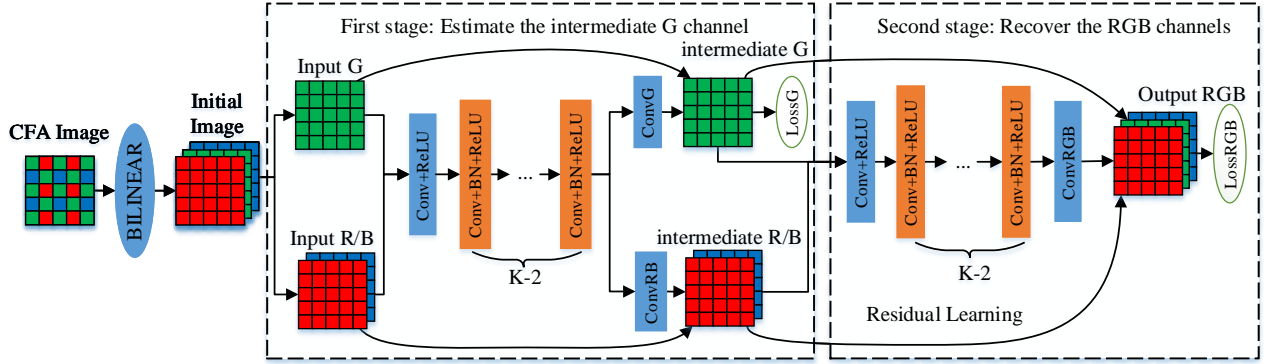
Fig. 1: *The architecture of the proposed deep residual CNN for color demosaicking. The input of the network is initialized by applying bilinear interpolation to the color filter array image. The network consists of two main stages. The first stage aims to recover an intermediate result of the G channel as guidance prior, while the second stage uses the intermediate G channel information to guide the reconstruction of the final RGB channels.*

of each module, 64 filters of size $3\times3\times64$ are adopted. The number of layers K for each module is set to 5. As a result, the proposed network has a depth of 10 and a receptive field of size 21×21. Note that we directly pad zeros before each 3×3 convolution to ensure that each feature map of the middle layers has the same size of the input image. To ease the training as well as boost the performance, the residual learning strategy is adopted to learn the difference between the initialized image and ground-truth image.

As discussed before, the G channel usually has more detailed information than R/B channels. The first stage of our network produces an intermediate reconstruction of the G channel. As any reconstruction error introduced in the G channel will be inevitably propagated to the subsequent processing steps, a G channel loss function is used in the first stage. In the second stage, the reconstructed intermediate G channel together with the initialized R/B channels are used as input to produce a more accurate demosaicking result. A loss function defined on the R/G/B channels is used for the second stage training.

To sum up, the proposed CNN based CDM model has two distinct characteristics. First, instead of using the CFA image as input, we take the initial images by simple bilinear interpolation as input, and then the residual learning strategy is used to reconstruct the demosaicked images. Second, the proposed model adopts a two-stage scheme to make use of the detailed G channel information to guide the reconstruction of R/B channels.

2.3. Network Training

Training Data. It is widely known that deep learning generally benefits from the availability of large scale training data. In order to train the proposed network effectively, we need

to collect enough high quality full color images as training data. Recently, Ma *et al.* [18] established the Waterloo Exploration Database (WED) with 4,744 high-quality natural images for the study of image quality assessment methods. We randomly choose 4,644 images from WED as our training dataset, which are sufficient for the training of a deep network. (The remaining 100 images are used for testing.) To generate the corresponding input image of a ground-truth image, we first downsample a CFA image from it based on the Bayer pattern, and then adopt the simple bilinear interpolation demosaicking method to generate an initial demosaicked image. Patches are extracted from the demosaicked/ground-truth image pairs for training. As the patch size should be larger than the receptive field size (i.e., 21×21) such that enough spatial information can be involved for better reconstruction. In practice, we set the training patch size as 50×50. Finally, we generate $3,000\times128$ patch pairs for training, and rotation or flip based data augmentation is used during training.

Loss Function. Given the training set $(\mathbf{y}_i, \mathbf{x}_i)_{i=1}^N$, where $\mathbf{y}_i$ is the i-th input of initially demosaicked image, and $\mathbf{x}_i$ is the corresponding ground-truth image. The following average mean square error loss function is used to learn the network parameter Θ,

$$\ell(\Theta) = \frac{1}{2N} \sum_{i=1}^{N} \left(\|\mathcal{F}(\mathbf{y}_{i,G}; \Theta_1) - \mathbf{x}_{i,G}\|^2 \right.$$
$$\left. + \|\mathcal{F}(\mathbf{y}_i; \Theta_1, \Theta_2) - \mathbf{x}_i\|^2 \right) \tag{3}$$

where Θ_1 and Θ_2 are the parameters of the first stage and second stage, respectively, N denotes the number of patch pairs. And $\mathbf{y}_{i,G}$ is the G channel of i-th initially demosaicked image, $\mathbf{x}_{i,G}$ is the G channel of the corresponding ground truth image.

Parameter Setting. In our experiments, the network weights are initialized based on the method in [19]. The Adam [16] solver is adopted to optimize the network parameters Θ. With batch normalization, the learning rate is started from $2e-4$ and then fixed to $1e-4$. Specifically, the training is terminated if the training error is fixed within three sequential epochs. For the other hyper-parameters of Adam, we use its default setting. The min-batch size is always set to 128. Rotation or flip based data augmentation is used during min-batch learning. All the experiments are carried out in the Matlab (R2015b) programming environment on a PC with Intel(R) Core(TM) i7-5930K CPU 3.50GHz and an Nvidia Titan X GPU. Benefiting the merits of batch normalization, residual learning, good starting point and G-channel guided reconstruction, our model converges very fast and takes about only one day to train the model.

3. EXPERIMENTS

Fig. 2: *The WED-CDM Dataset. One hundred representative full color images from WED are selected to form this testing set of CDM algorithms.*

3.1. Testing Datasets and Compared Methods

There are two widely-used datasets (i.e., Kodak and McMaster) for evaluating CDM performance in literatures. The Kodak dataset contains 24 digital color images (768×512) derived from various of film source materials. However, it has been revealed that the Kodak dataset has relatively low resolution and high spectral correlation, making it unsuitable for objective evaluation of CDM algorithms for modern digital cameras. To remedy this, Zhang *et al.* [9] proposed the McMaster dataset which contains 18 sub-images (500×500) cropped from eight high resolution natural images. In the past decades, most CDM algorithms have been developed, optimized and tested based on these two datasets. However, the McMaster is still limited in color gradations and scene variety. To make a more objective comparison of existing CDM algorithms, we use the remaining 100 high quality full color images from the WED dataset (the other 4,644 images are used as training data for our CNN model) to form a new testing set, named WED for Color Demosaicking (WED-CDM).

The WED-CDM contains five categories of images: animal, building, food, human and plant with 20 images each category. Some example images are shown in Fig. 2.

Each full color image from the datasets is firstly downsampled according to the Bayer pattern [1], and the downsampled CFA images are input to the competing CDM algorithms for full color image reconstruction. Our proposed method is compared with seven state-of-the-art demosaicking algorithms, including AHD [4], DLMMSE [5], GBTF [6], LDI-NAT [9], RI [7], ARI [11] and MLRI [10].

3.2. Experimental Results

Quantitative Evaluation. To quantitatively evaluate the objective performance of CDM algorithms, the PSNR and the composite PSNR (CPSNR) are adopted to measure the quality of each color channel and all the three channels, respectively. The PSNR/CPSNR results of different methods on Kodak, McMaster and WED-CDM datasets are listed in Table 1. We can see that, the proposed method achieves a much better average PSNR/CPNSR results than the compared methods on all the three datasets. Particularly, our method outperforms the second best method by 2.2 dB, 1.4 dB and 1.8 dB on Kodak, McMaster and WED-CDM, respectively.

Qualitative Evaluation. Besides the PSNR improvement, our proposed method also shows clear advantages in terms of subjective quality assessment. Fig. 3 and Fig. 4 illustrate the cropped demosaicked images of different methods. It can be seen that the visual quality perception is consistent with the subjective quantitative evaluation. In particular, false color artifacts and zippers along edges accompanied by the compared methods can be easily observed, while no obvious artifacts are observed from the demosaicked image of the proposed method.

Running Time. The proposed demosaicking method is implemented in MatConvNet [20]. We compare its testing time on CPU/GPU with the other seven state-of-the-art demosaicking methods. All algorithms are tested on the same machine and running environment described in section 2.3. The total running time on Kodak and McMaster (42 images) of different methods is presented in Table 2. It can be seen that the proposed method not only enjoys a fast demosaicking time on GPU and but also has a competitive speed on CPU.
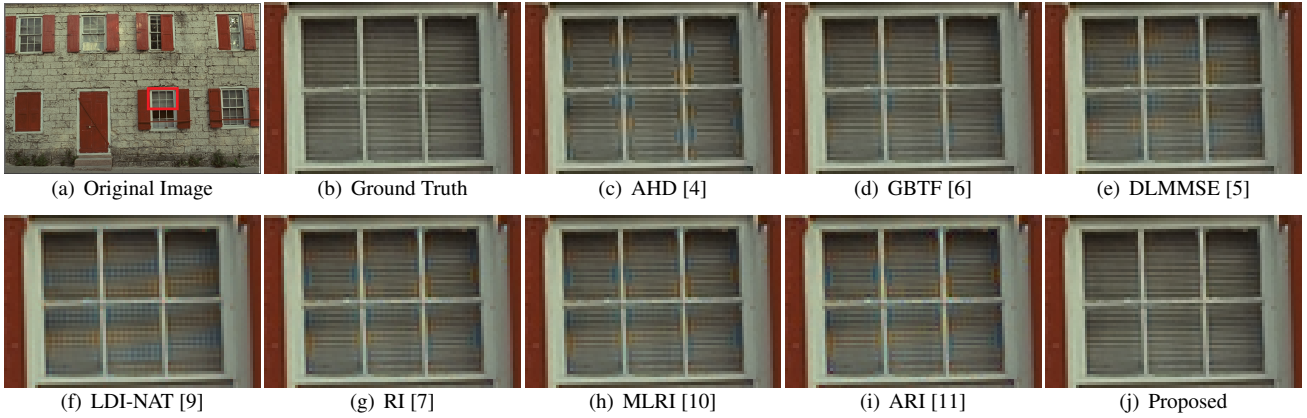
4. CONCLUSION

This paper presented a powerful color demosaicking (CDM) scheme by embracing the popular deep CNNs. Motivated by the demosaicking modeling with analysis prior, we adopted a residual learning formulation for CDM problem. In particular, considering that the missing pixels of the G channel are only half of those of the R/B channel, we designed a two-stage architecture, which first estimates the G channel and then uti-

Table 1: Average PSNR performance result (in dB) on three datasets. The best results are highlight in bold.
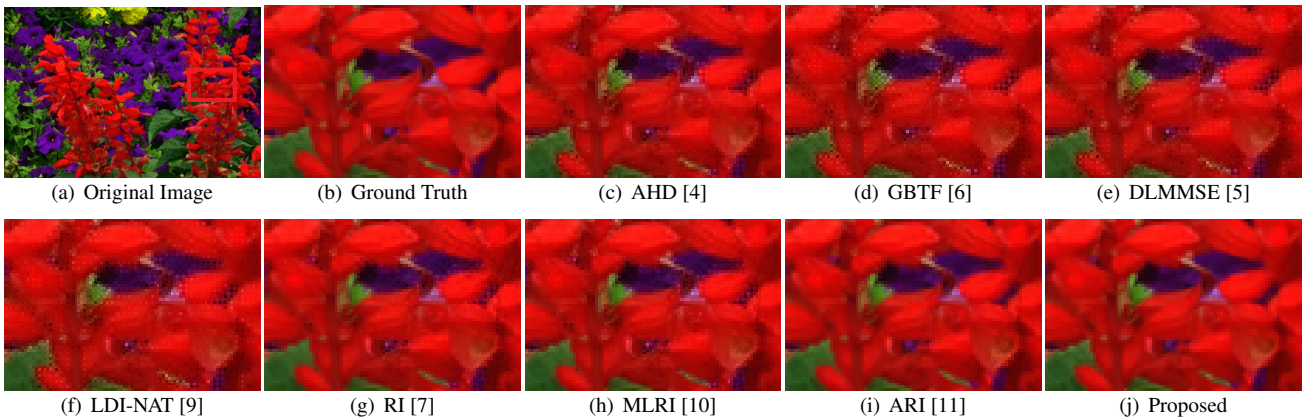
Dataset	Kodak Dataset (24)				McMaster Dataset (18)				WED-CDM Dataset (100)			
Methods	R	G	B	RGB	R	G	B	RGB	R	G	B	RGB
AHD	37.23	39.76	37.43	37.96	34.19	37.80	33.26	34.62	35.22	38.74	34.98	35.93
GBTF	39.68	43.34	40.01	40.62	33.98	37.34	33.07	34.38	35.67	39.50	35.36	36.39
DLMMSE	39.18	42.63	39.58	40.11	34.03	37.99	33.04	34.47	35.55	39.79	35.25	36.32
LDI-NAT	36.99	39.44	37.12	37.69	36.28	39.76	34.39	36.2	36.2	40.08	35.69	36.82
RI	37.82	41.00	37.80	38.56	36.07	39.99	35.35	36.48	36.52	40.58	36.17	37.21
MLRI	38.34	40.86	38.26	38.95	36.59	40.02	35.41	36.77	36.89	40.47	36.45	37.48
ARI	39.10	42.31	38.90	39.79	37.41	40.72	36.05	37.52	37.18	40.92	36.84	37.86
Ours	**41.38**	**44.85**	**41.04**	**42.04**	**39.14**	**42.10**	**37.31**	**38.98**	**39.01**	**43.04**	**38.54**	**39.67**

Table 2: Total demosaicking time on Kodak and McMaster (42 images).

Methods	AHD	GBTF	DLMMSE	LDI-NAT	RI	MLRI	ARI	Ours(CPU)	Ours(GPU)
Time(s)	32	301	531	39766	39	52	64	136	29

(a) Original Image   (b) Ground Truth   (c) AHD [4]   (d) GBTF [6]   (e) DLMMSE [5]

(f) LDI-NAT [9]   (g) RI [7]   (h) MLRI [10]   (i) ARI [11]   (j) Proposed

Fig. 3: The comparison of demosaicking results on Kodak 01.

(a) Original Image   (b) Ground Truth   (c) AHD [4]   (d) GBTF [6]   (e) DLMMSE [5]

(f) LDI-NAT [9]   (g) RI [7]   (h) MLRI [10]   (i) ARI [11]   (j) Proposed

Fig. 4: The comparison of demosaicking results on McMaster 17.

lizes the tentative G channel to guide the final demosaicking result. While most existing CDM methods are based on hand-crafted priors which are limited in capturing the complex and long range dependency among pixels, the proposed method directly learns the demosaicking priors from training data and embed them into the CNN-based demosaicking framework in an end-to-end manner. Experiments on the Kodak, McMaster, and our proposed WED-CDM datasets have demonstrated the superiority of our method compared with the state-of-the-art demosaicking algorithms.

5. REFERENCES

[1] Bryce E Bayer, "Color imaging array," July 20 1976, US Patent 3,971,065.

[2] Xin Li, Bahadir Gunturk, and Lei Zhang, "Image de-mosaicing: A systematic survey," in *Electronic Imaging 2008*. International Society for Optics and Photonics, 2008, pp. 68221J–68221J.

[3] Philippe Longere, Xuemei Zhang, Peter B Delahunt, and David H Brainard, "Perceptual assessment of de-mosaicing algorithm performance," *Proceedings of the IEEE*, vol. 90, no. 1, pp. 123–132, 2002.

[4] Keigo Hirakawa and Thomas W Parks, "Adaptive homogeneity-directed demosaicing algorithm," *IEEE Transactions on Image Processing*, vol. 14, no. 3, pp. 360–369, 2005.

[5] Lei Zhang and Xiaolin Wu, "Color demosaicking via directional linear minimum mean square-error estimation," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2167–2178, 2005.

[6] Ibrahim Pekkucuksen and Yucel Altunbasak, "Gradient based threshold free color filter array interpolation," in *2010 IEEE International Conference on Image Processing*. IEEE, 2010, pp. 137–140.

[7] Daisuke Kiku, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi, "Residual interpolation for color image demosaicking," in *2013 IEEE International Conference on Image Processing*. IEEE, 2013, pp. 2304–2308.

[8] Julien Mairal, Francis Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman, "Non-local sparse models for image restoration," in *2009 IEEE 12th International Conference on Computer Vision*. IEEE, 2009, pp. 2272–2279.

[9] Lei Zhang, Xiaolin Wu, Antoni Buades, and Xin Li, "Color demosaicking by local directional interpolation and nonlocal adaptive thresholding," *Journal of Electronic Imaging*, vol. 20, no. 2, pp. 023016–023016–16, 2011.

[10] Daisuke Kiku, Yusuke Monno, Masayuki Tanaka, and Masatoshi Okutomi, "Minimized-laplacian residual interpolation for color image demosaicking," in *Electronic Imaging*. International Society for Optics and Photonics, 2014, pp. 90230L–90230L.

[11] Yusuke Monno, Daisuke Kiku, Masayuki Tanaka, and Masatoshi Okutomi, "Adaptive residual interpolation for color image demosaicking," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 3861–3865.

[12] Teresa Klatzer, Kerstin Hammernik, Patrick Knobelreiter, and Thomas Pock, "Learning joint demosaicing and denoising based on sequential energy minimization," in *Computational Photography (ICCP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 1–11.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[14] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," *arXiv preprint arXiv:1512.03385*, 2015.

[16] Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[17] Jian Sun and Marshall F Tappen, "Separable markov random field model and its applications in low level vision," *IEEE Transactions on Image Processing*, vol. 22, no. 1, pp. 402–407, 2013.

[18] 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*, 2016.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1026–1034.

[20] Andrea Vedaldi and Karel Lenc, "Matconvnet: Convolutional neural networks for matlab," in *Proceedings of the 23rd ACM international conference on Multimedia*. ACM, 2015, pp. 689–692.