

Universal Demosaicking for Interpolation-Friendly RGBW Color Filter Arrays

Jia Li, Chenyan Bai, and Hua Huang, *Senior Member, IEEE*

Abstract—Interpolation-friendly RGBW color filter arrays (CFAs) and the popular sequential demosaicking contain the idea of computational photography, where the CFA and the demosaicking method are co-designed. Due to the advantages, interpolation-friendly RGBW CFAs have been extensively used in commercial color cameras. However, most associated demosaicking methods rely on strict assumptions or are limited to a few specific CFAs with a given camera. In this paper, we propose a universal demosaicking method for interpolation-friendly RGBW CFAs, which enables the comparison of different CFAs. Our new method belongs to sequential demosaicking, i.e., W channel is interpolated first and then RGB channels are reconstructed with guidance from the interpolated W channel. Specifically, it first interpolates the W channel using only available W pixels followed by an aliasing reduction technique to remove aliasing artifacts. Then it employs an image decomposition model to build relations between W channel and each of RGB channels with known RGB values, which can be easily generalized to the full-size demosaicked image. We apply the linearized alternating direction method (LADM) to solve it with convergence guarantee. Our demosaicking method can be applied to all interpolation-friendly RGBW CFAs with varying color cameras and lighting conditions. Extensive experiments confirm the universal property and advantage of our proposed method with both simulated and real raw images.

Index Terms—Demosaicking, color filter array(CFA), RGBW, aliasing reduction, image decomposition model.

I. INTRODUCTION

IMAGING sensors can only sense intensity information, i.e., capturing monochromatic images. To obtain color information, most digital color cameras utilize color filter arrays (CFAs) to cover imaging sensors. A CFA allows only one color value to be recorded at each pixel, and the captured images are called raw images. Then one needs demosaicking methods to reconstruct full color images (i.e., the color images with red (R), green (G), and blue (B) channels) from these raw images. Demosaicking is an essential part of the image signal processing (ISP) pipeline. Accordingly, the CFA and the corresponding demosaicking method are both important to obtain high quality demosaicked images. Note that almost all commercial CFAs are periodic and defined on the square lattice, where the minimal repeating arrays are called CFA patterns. For non-periodic CFAs, we refer the readers to [1].

To date, the Bayer CFA [2] is the most common one used in commercialized color cameras (see Fig. 1 (a)). It is a 2×2

CFA pattern, which is composed of two G pixels diagonally, one R pixel, and one B pixel. Despite its widespread use, the RGB filters of Bayer CFA only pass through roughly one-third of the incident light. This limits the intrinsic light utilization efficiency, especially in low-light conditions [3], [4]. Moreover, the trend toward smaller pixels decreases the light-active surface per pixel, which reduces the incident photons and signal-to-noise ratio (SNR). To address such limitation, a variety of innovative CFA patterns with higher transmittance or pixel binning are explored [1], [5], which absolutely increase the amount of light reaching each pixel. In particular, it is known that the panchromatic (or white (W)) pixels do not block the visible light. Accordingly, various RGBW CFAs have been developed (see Figs. 1 (b)-(n)). More importantly, the manufacturing of RGBW CFAs is practical. Also, the increasing computational power of cameras could support relatively sophisticated demosaicking methods. As a result, RGBW CFAs have attracted renewed interest recently and used in flagship phones to provide better low-light photography performance, e.g., Oppo Reno7 [6], Vivo X80 [7], and Tecno Camon 19 [8].

The percentage and arrangement of W pixels matter. We can gain insight from the well-studied Bayer CFA. The Bayer CFA has 50% G pixels, while the remaining R and B pixels are arranged in a checkerboard pattern (see Fig. 1 (a)). Based on the two facts, most Bayer demosaicking methods belong to the *sequential demosaicking* [9], i.e., they first interpolate G channel and then estimate R and B channels by exploiting the inter-channel correlations with the interpolated G channel. The accuracy of G-channel interpolation becomes critical. The key to developing effective and efficient demosaicking methods is that G pixels are dominant and assigned as being easy to calculate derivatives, e.g., the horizontal first and second order partial derivatives. Such two properties are called *interpolation-friendly*, resulting in practical interpolation methods for the W channel [10]. So the core motivation of most Bayer demosaicking methods suggests the idea of computational photography to jointly design the CFA and the demosaicking method. Due to the widespread adoption in practice, we focus on interpolation-friendly RGBW CFAs in this work.

However, demosaicking an RGBW raw image, or RGBW demosaicking for short, is more challenging than Bayer demosaicking. This is because the RGB values of the demosaicked image are partially replaced by W values, especially when W pixels are highly dominant. Moreover, many validated experiences of Bayer demosaicking may fail for RGBW demosaicking. For instance, color differences, such as $R-G$ and

J. Li and H. Huang are with the School of Artificial Intelligence, Beijing Normal University, Beijing 100875, P.R. China (e-mail: jiali@bnu.edu.cn; huahuang@bnu.edu.cn) (Corresponding author: Hua Huang).

C. Bai is with the College of Information Engineering, Capital Normal University, Beijing 100048, P.R. China (e-mail: cybai@cnu.edu.cn).

Manuscript received April 19, 2005; revised August 26, 2015.

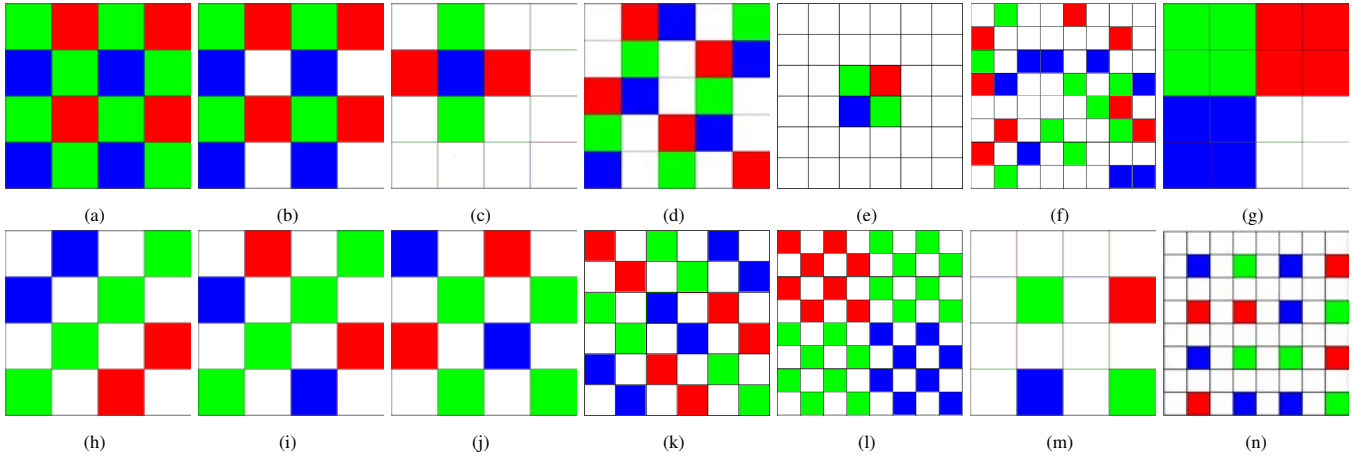


Fig. 1. The Bayer CFA pattern and several representative RGBW CFA patterns. They are the (a) Bayer CFA [2], (b) Gindele CFA [11], (c) Luo CFA [12], (d) Wang CFA [13], (e) Chakrabarti CFA [3], (f) Learned CFA [14], (g) Binning CFA [15], (h) Kodak CFA [16], (i) Sony CFA [17], (j) Yamagami CFA [18], (k) Kaizu CFA [19], (l) Hamilton CFA [20], (m) Honda CFA [21], and (n) Random CFA [22], respectively. **Images in this paper are best viewed on screen!**

$B - G$, facilitate the interpolation of R and B channels for the Bayer CFA [23]. Since W values are much larger than those of R, G, or B, color differences like $W - R$, $W - G$, and $W - B$ often produce visible artifacts. Further details on this issue can be found in subsection III-A. Residual interpolation for the Bayer CFA [10], [24], [25] needs to eliminate residual errors at available RGB pixels, which cannot yield promising results due to the low sampling rates of RGB pixels in almost all RGBW CFAs. While frequency domain methods [26], [27] perform well with the Bayer CFA, W-dominant CFAs usually have severe spectral overlap, which seriously degrades the quality of demosaicked images. Most reconstruction-based methods require an image formation model [28], [29], and they often assume a perfect linear relation between W and RGB values, which is not reliable. Recent advances in Bayer demosaicking are driven by the success of deep learning [30]–[32]. However, deep learning demands accurately paired images to achieve superior performance, which are difficult to collect. One must simulate W channels based on the assumed linear relation between W and RGB values, which causes a mismatch between the image formation model and the real observed images. Additionally, deep learning is not easy to diagnose and repair, which can adversely affect productivity.

Despite the challenges, many demosaicking methods have been developed specifically for a few interpolation-friendly RGBW CFAs and cameras (see subsection II-B). However, few of these methods can be applied to other CFAs (e.g., the Random CFA [22]), cameras, and lighting conditions. In this work, we present a universal demosaicking method for interpolation-friendly RGBW CFAs. This allows these CFAs to be compared using the same demosaicking method and provides feedback for proposing new CFAs. Our method belongs to sequential demosaicking, i.e., the W channel is first interpolated, and then the RGB channels are reconstructed based on the interpolated W channel. It does not utilize specific characteristics of given cameras and illuminants. Specifically, our W-channel interpolation recovers missing W values using only known ones. Meanwhile, our RGB-channel reconstruction uses an objective function to extract structural information

from the W channel, where known RGB values are identified by masks. Therefore, our method is applicable to various interpolation-friendly RGBW CFAs.

The contributions of this paper are:

- After extensively reviewing existing RGBW CFAs and demosaicking methods, we propose a universal demosaicking method for interpolation-friendly RGBW CFAs. It consists of two sequential steps: W-channel interpolation and RGB-channel reconstruction. Our method is both data and camera independent, and can produce demosaicked images with high visual quality under different lighting conditions.
- We develop a new interpolation method to recover the W channel, where an aliasing reduction process is proposed to further boost the recovery accuracy. Our W-channel interpolation is applicable to all interpolation-friendly RGBW CFAs, e.g., those CFAs shown in Figs. 1 (h)–(n).
- We propose an image decomposition model to reconstruct RGB channels with the guidance of W channel. It is a new spatially variant linear model between W and each of RGB channels, while keeping the known RGB values fixed. Our model is convex, and we use the linearized alternating direction method (LADM) to solve it efficiently with convergence guarantee [33]. Our RGB-channel reconstruction is workable with arbitrary RGBW CFAs.

The rest of the paper is organized as follows. In Section II, we review the existing RGBW CFAs and demosaicking methods. Then we introduce our RGBW demosaicking method in Section III. In Section IV, we describe the solving process of our image decomposition model for RGB-channel reconstruction. In Section V, we conduct experiments to test our demosaicking method. Finally, we conclude the paper in Section VI.

II. RELATED WORK

In this section, we review both the existing RGBW CFAs and demosaicking methods.

A. RGBW CFAs

We roughly categorize the existing RGBW CFAs into W-interpolation-unfriendly CFAs and W-interpolation-friendly CFAs. It should be noted that *we consider two CFA patterns are the same if one could transform to another by cropping periodically, rotating 90 degrees, and flipping in the up-down or left-right directions*. This is because we can convert a raw image by the aforementioned transformations and the identical demosaicking method does apply.

1) *W-Interpolation-Unfriendly CFAs*: As shown in Fig. 1 (b), Gindele and Gallagher [11] introduced a 2×2 CFA pattern, where one of the G pixels in the Bayer CFA pattern is replaced with a W pixel. Luo [12] described a 4×4 CFA pattern shown in Fig. 1 (c), which includes a 3×3 cross for RGB pixels and the remaining ones are W pixels. A systematic approach designs RGBW CFAs in the frequency domain [4], [13]. For example, Wang et al. [13] proposed a 5×5 CFA pattern (see Fig. 1 (d)) with 40% W pixels, which has no spectral overlap. Li et al. [4] formulated the CFA design with W pixels as a multi-objective optimization problem, which simultaneously minimizes the spectral overlap and maximizes the percentage of W pixels. Several CFA patterns are designed by placing the 2×2 Bayer CFA pattern into 4×4 [34], 6×6 [3] (see Fig. 1 (e)), and 18×18 [35] W patterns. Chakrabarti [14] jointly learned the RGBW CFA and the associated demosaicking method. The learned CFA pattern is shown in Fig. 1 (f). As shown in Fig. 1 (g), the Binning CFA pattern is similar to the Gindele CFA pattern. The difference is that its adjacent 2×2 pixels are the same color.

2) *W-Interpolation-Friendly CFAs*: Inheriting from the Bayer CFA, W-interpolation-friendly CFAs array RGB pixels in a checkerboard (see Figs. 1 (h)-(l)) or downsampled checkerboards (see Figs. 1 (m)-(n)), which generates at least 50% W pixels. The Kodak CFA [16] is the most frequently used RGBW CFA. It is a 4×4 CFA pattern and contains 2×2 groups of pixels (see Fig. 1 (h)). Each group has two W pixels and two same color pixels, which facilitates combining the color pixels within the group. The Sony CFA [17] and Yamagami CFA [18] are both 4×4 CFA patterns, which are shown in Figs. 1 (i) and (j), respectively. As shown in Fig. 1 (k), Kaizu [19] developed the 6×6 uniformly colored Kaizu CFA [19], which contains an equal percentage of R, G, and B pixels. Hamilton and Compton [20] suggested the Hamilton CFA (see Fig. 1 (l)), which is an 8×8 CFA pattern and has three imaging modes [36]. Namely, ISP can directly demosaick the RGBW raw image, or first combine each 2×2 or 4×4 pixels of the RGBW raw image and then perform demosaicking. Honda et al. [21] proposed the Honda CFA with 75% W pixels (see Fig. 1 (m)), where the RGB pixels are assigned in a downsampled checkerboard. Oh et al. introduced the 8×8 Random CFA [22] (see Fig. 1 (n)), where the R, G, and B pixels are randomly distributed.

B. Demosaicking Methods

Many demosaicking methods have been proposed for RGBW CFAs. Most of them are inspired by Bayer demosaicking and developed for a few specific CFAs. We

roughly categorize the existing demosaicking methods into five categories: heuristic methods, sequential methods, frequency domain methods, reconstruction-based methods, and deep learning-based methods.

1) *Heuristic Methods*: Heuristic methods are mostly based on reasonable assumptions on inter-channel correlations, e.g., the smoothness of color differences, and the linear relation between W and RGB values. Gindele and Gallagher [11] modified the Bayer demosaicking method developed in [37] specifically for the Gindele CFA. To address saturated W pixels, Yamagami et al. [18] estimated the color differences from RGB channels and not from W channel, which utilizes the linear relation $W = R + G + B$. Honda et al. [21] performed demosaicking based on the color separation process, which may produce a strong grid-pattern noise when the linear relation $W = R + G + B$ is not valid [38], [39]. A follow-up work was presented in [40], where an adaptive diagonal color separation process is proposed. Some methods convert RGBW raw images into Bayer ones and use Bayer demosaicking to generate demosaicked images. Getman et al. [38] estimated the missing G values at W pixels by learning a linear filter for the Gindele CFA. Choi et al. [41] reconstructed the missing G values at W pixels with a diagonal edge-adaptive filtering. Due to simplify the interpolation, the color difference domain and residual domain with the guided filter [10], [24], [25] are widely used in Bayer demosaicking. Zhang et al. [42] presented an alternative domain: optimal chrominance domain, which utilizes the linear relation $W = (R + G + B)/3$.

2) *Sequential Methods*: Sequential methods are the most popular demosaicking methods for W-dominant CFAs. They commonly consist of two sequential steps: first interpolate W channel, and then recover RGB channels with the reference of the interpolated W channel. Their main difference is how to reconstruct RGB channels. Luo [34] demosaicked the downsampled Bayer raw image and converted to the luminance-chrominance color space, e.g., CIELAB or YUV. Then they upsampled the two chrominance channels to full-size and replaced the luminance channel with W channel. Luo [12] recovered RGB channels based on a global linear model between W and R, G, or B values, where the coefficients are computed using the least-squares method with known RGB values. Some methods recover RGB channels in the color difference domain [16], [20], [43]–[45]. Recovering RGB channels with the reference of W channel can be considered as the scribble-based colorization [46]. Chakrabarti et al. [3] computed the color information at Bayer CFA patterns and propagated to the remaining pixels. Oh et al. [22] produced the demosaicked image by the scribble-based colorization, which was extended to image sequence and video by developing a three dimensional colorization method [47]. For the Kodak CFA and Honda CFA, Kwan et al. presented the pansharpening-based demosaicking method [48] and its various improved versions [49]–[51]. They performed Bayer demosaicking to obtain half-size RGB channels. Then they fused the full-size W channel and half-size RGB channels to generate the full-size demosaicked image, which can be regarded as a pansharpening problem. For the Honda CFA, Jeong et al. [52] recovered RGB channels using Laplacian

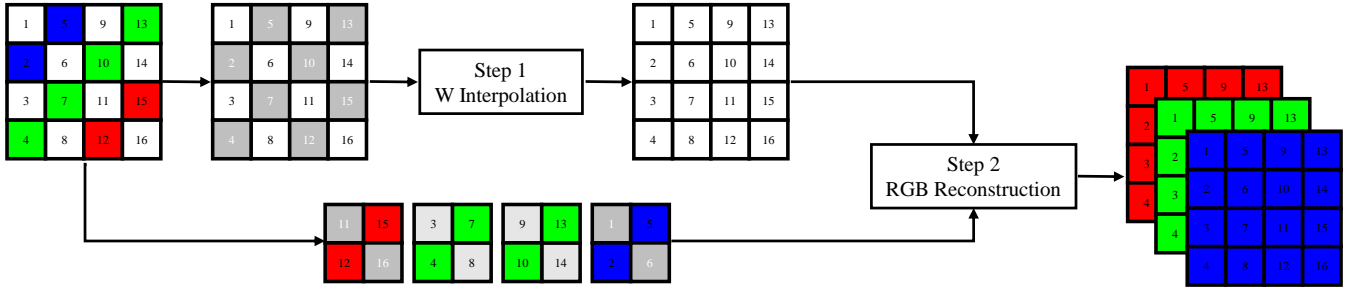


Fig. 2. Overall framework of the proposed RGBW demosaicking method. Note that gray pixels represent missing values. The number within a pixel identifies its position in the full-size raw image.

pyramid decomposition.

3) *Frequency Domain Methods*: Frequency domain methods are usually universal and applicable to arbitrary periodic CFAs. However, they may not work well for CFAs with highly dominant W pixels, e.g., the Honda CFA. Gu et al. [53] assumed that the high frequency filter-bank coefficients are sparsity and used a robust regression to reconstruct the chrominances. Singh and Singh [54] applied the Discrete Fourier Transform (DFT) to the image formation model and derived a system of linear equations. They computed the pseudo-inverse to produce the demosaicked image. The raw images by periodic CFAs have different frequency locations of luminance and chrominances [26], [27]. Based on the fact, many frequency selection-based demosaicking methods were proposed [4], [13], [55]. These methods are linear and simple, which is a good compromise between efficiency and effectiveness.

4) *Reconstruction-Based Methods*: Reconstruction-based methods solve mathematical problems for demosaicking, where the inter-channel correlations and image priors are incorporated. They require an image formation model to generate training image pairs or define a fidelity term for known values. This needs an assumption about the relation between W and RGB values. Condat [56] maximized the smoothness in the luminance-chrominance color space with the constraint of fixing known values. However, their approach needs a linear relation assumption between W and RGB values. Kim et al. [57] presented a demosaicking method based on rank minimization with a colorization constraint. Some methods formulated demosaicking as an inverse problem. Amba and Alleysson [58] proposed the linear minimum mean square error method, which uses a linear image formation model. Bai and Li [59] developed a convolutional sparse coding-based method. Kang and Jung [60] utilized the low dimensional manifold model prior. Kurniawan et al. [61] utilized the deep image prior [62] to regularize the inverse problem.

5) *Deep Learning-Based Methods*: Deep neural networks (DNNs) have attracted the attention in Bayer demosaicking [30]–[32]. DNNs learn nonlinear mappings from raw images to the corresponding RGB images. With massive training image pairs, DNNs can achieve excellent performance. However, collecting training image pairs for RGBW CFAs is more difficult than the Bayer CFA. With a two-path feed-forward neural network, Chakrabarti [14] jointly learned the RGBW CFA and the demosaicking method. As for the paired

training images, they collected many RGB images and used the linear relation $W = R + G + B$ to simulate W channels. Amba et al. [63] used a two-layer neural network to demosaic various periodic CFAs. Sharif and Jung [35] presented a deep learning-based demosaicking method for a very sparse color sensor with only 1% RGB pixels. Their method consists of two sequential steps, which borrows insight from traditional sequential methods. Kim et al. [64] proposed the DePhaseNet for RGBW demosaicking, which rearranges the Kodak CFA pattern to obtain a half-size and a quarter-size multi-channel images. The effect of quantization bit-depth of the DePhaseNet was studied in [65]. Note that RGBW CFAs can be regarded as specific multispectral filter arrays (MSFAs). However, the demosaicking methods designed for MSFAs [66], [67] cannot be properly applied to RGBW CFAs. This is because MSFAs typically have a greater number of bands and sparser spatial sampling, and they often do not have a dominant band. To design better demosaicking methods, it is necessary to fully utilize the unique characteristics of RGBW CFAs.

III. PROPOSED RGBW DEMOSAICKING METHOD

Since the Kodak CFA is the most frequently used RGBW CFA, we use it to demonstrate our demosaicking method. As shown in Fig. 2, our method is a sequential method, which has two steps. It first interpolates W channel using only known W values. Then it reconstructs RGB channels with the reference of W channel. Our W-channel interpolation method is applicable to various *W-interpolation-friendly CFAs*, where RGB pixels are arranged in checkerboard or downsampled checkerboard patterns (e.g., those CFAs shown in Figs. 1 (h)–(n)). However, our RGB-channel reconstruction method works with arbitrary CFAs. For a clear description, we summary the main notations used in this paper in Table I.

A. W-Channel Interpolation

The interpolation accuracy of W channel is critical to recover RGB channels because of the error propagation in two sequential steps. So we devote much effort to improving the performance of W-channel interpolation.

We first discuss the relation between W and RGB values in the ideal situation. Namely, the illuminant is fixed and there is a perfect linear relation between the spectral sensitivity functions of W, R, G, and B pixels. It is known that each value of a raw image is the integration of the product of the reflectance,

TABLE I
SUMMARY OF THE MAIN NOTATIONS USED IN THIS PAPER.

Notation	Definition
$*$	2D convolution
\odot	Elementwise production
\oslash	Elementwise division
$ \cdot $	Absolute value
$\mathbf{0}$	All-zero vector or matrix
$\mathbf{x}(i)$	i -th element of vector \mathbf{x}
$\text{Diag}(\mathbf{x})$	Diagonal matrix whose i -th diagonal element is $\mathbf{x}(i)$
$\langle \cdot, \cdot \rangle$	inner product between two matrices or vectors
$\mathbf{X}(i, j)$	(i, j) -th element of matrix \mathbf{X}
$\text{vec}(\mathbf{X})$	Vector formed by stacking all the columns of matrix \mathbf{X}
\mathbf{X}^T	Transpose of matrix \mathbf{X}
\mathbf{X}^{-1}	Inverse of matrix \mathbf{X}
$\ \mathbf{x}\ _2$	ℓ_2 -norm $(\sqrt{\sum_i \mathbf{x}(i)^2})$ of vector \mathbf{x}
$\ \mathbf{x}\ _\infty$	ℓ_∞ -norm $(\max_i \{ \mathbf{x}(i) \})$ of vector \mathbf{x}
$\ \mathbf{X}\ _F$	Frobenius norm $(\sqrt{\sum_i \sum_j \mathbf{X}(i, j)^2})$ of matrix \mathbf{X}

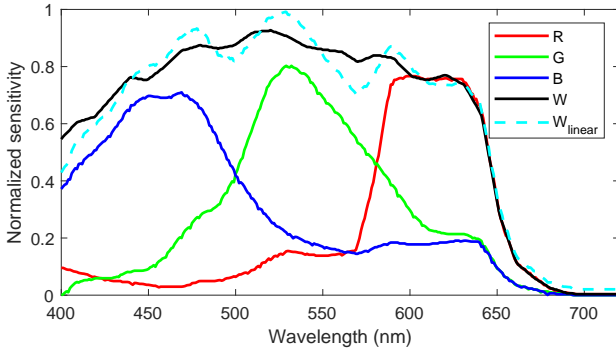


Fig. 3. Normalized spectral sensitivity functions of an RGBW sensor [39]. Note that W and W_{linear} are the actual and the linearly estimated spectral sensitivity functions of W , respectively.

pixel's spectral sensitivity function, and incident light's spectrum. Accordingly, there is a linear relation between W and RGB values, i.e., $W = \alpha_R R + \alpha_G G + \alpha_B B$, where α_R , α_G , and α_B are constant for all the images. We may estimate the three coefficients as follows. We consider the Kodak CFA shown in Fig. 2 as an example. We use an RGBW camera to capture a color checker under good lighting conditions. Every non-overlapping 4×4 patch of the raw image is downsampled to an RGBW pixel in the same manner, e.g., pixels numbered 15, 10, 5, and 6. Then we have a quarter-size RGBW image without missing values. We utilize the least-squares method to estimate the three coefficients and the computed values are $\alpha_R = 0.5722$, $\alpha_G = 0.8369$, and $\alpha_B = 0.7780$, which differ in cameras and lighting conditions. As a result, W and RGB are usually not of the same magnitude. Therefore, direct estimation of W may lead to greater accuracy. Furthermore, the sampling density and SNR of W are higher compared to RGB. Additionally, RGB pixels are typically irregular, which makes it impossible to calculate partial derivatives. These factors motivate us to recover missing W values using only known ones.

However, the ideal situation is too strict in the real world. We can see from Fig. 3 that the assumption of a perfect linear relation between the spectral sensitivity functions usually does not hold. Without considering the lighting condition, there

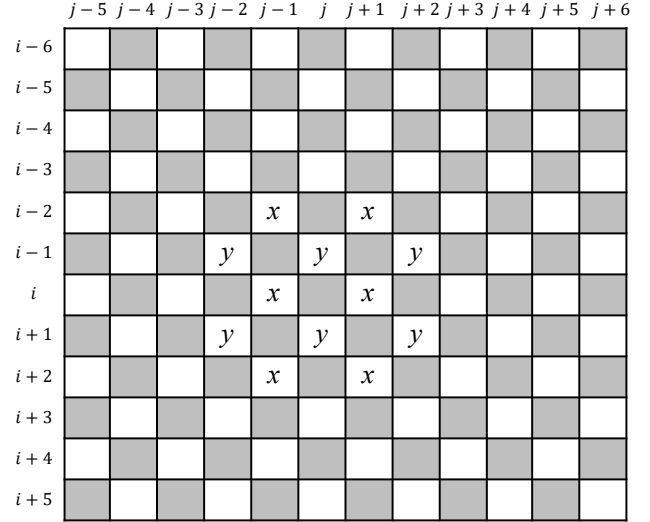


Fig. 4. Illustration of interpolating the W value at (i, j) . Note that gray pixels represent missing values. The values x and y denote W values, which are used to demonstrate aliasing reduction.

is a mismatch between the actual and the linearly estimated spectral sensitivity functions of W . Moreover, the spectral sensitivity functions of RGBW are diverse [68] and sensitive to many computer vision tasks, such as Bayer demosaicking [69] and hyperspectral image recovery [70]. Our W -channel interpolation and RGB-channel reconstruction do not depend on the linear relation between W and RGB values. As a consequence, our demosaicking method is applicable to RGBW cameras with various spectral sensitivity functions and lighting conditions.

Our W -channel interpolation is inspired by the Hamilton and Adams method [37]. For each missing W value, we first compute the horizontal and the vertical estimations and the corresponding variations. Then we average the two estimations weighted by the variations to obtain the interpolation value. We use Fig. 4 to show the interpolation of W value at (i, j) , where the missing values are assigned in a checkerboard. We may compute the horizontal first (denoted by ∂_h) and second order (denoted by ∂_h^2) partial derivatives, and the vertical first (denoted by ∂_v) and second order (denoted by ∂_v^2) partial derivatives by

$$\begin{aligned}\partial_h W(i, j) &= \frac{W(i, j+1) - W(i, j-1)}{2}, \\ \partial_h^2 W(i, j; \Delta) &= \frac{W(i, j-\Delta) + W(i, j+\Delta) - 2W(i, j)}{\Delta^2}, \\ \partial_v W(i, j) &= \frac{W(i+1, j) - W(i-1, j)}{2}, \\ \partial_v^2 W(i, j; \Delta) &= \frac{W(i-\Delta, j) + W(i+\Delta, j) - 2W(i, j)}{\Delta^2},\end{aligned}$$

where Δ is an integer. So the horizontal estimation of $W(i, j)$ is:

$$\hat{W}_h(i, j) = \frac{1}{2}(W(i, j+1) + W(i, j-1)) - \frac{1}{2}\partial_h^2 W(i, j; 1). \quad (1)$$

The horizontal estimation $\hat{W}_h(i, j)$ by the above equation is always equal to the ground truth of $W(i, j)$, which is

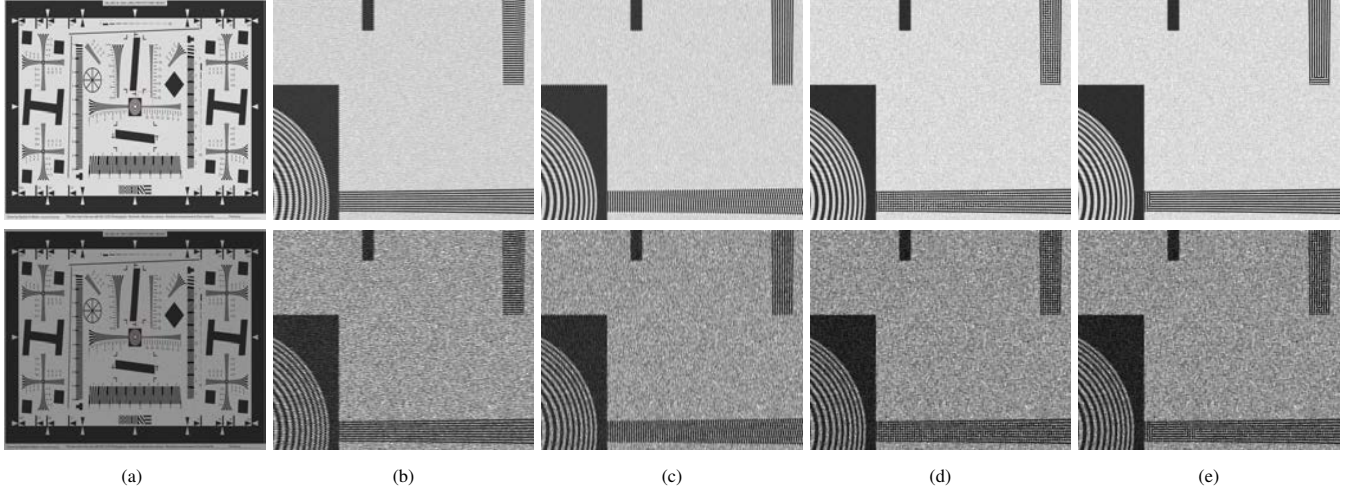


Fig. 5. The effectiveness of our aliasing reduction process. The first and second rows depict simulated resolution chart images [10] using the CIE D65 illuminant at 100 lux and 5 lux [77], respectively. (a) is the scaled image by our W-channel interpolation with the aliasing reduction process, in which the red rectangle indicates the selected patch to blow up. (b) and (c) force the interpolation along the horizontal and vertical directions, respectively. (d) and (e) perform interpolation by Eq. (2) without and with the aliasing reduction process, respectively.

our basic observation. However, we cannot directly compute $\partial_h^2 W(i, j; 1)$ as $W(i, j)$ is unknown. So we replace $\partial_h^2 W(i, j; 1)$ in Eq. (1) with $\partial_h^2 W(i, j; 2)$, which can be estimated as

$$\partial_h^2 \hat{W}(i, j; 2) = \frac{1}{2} (\partial_h^2 W(i+1, j; 2) + \partial_h^2 W(i-1, j; 2)).$$

The horizontal variation is defined as $\delta_h(i, j) = |\partial_h W(i, j)| + |2\partial_h^2 \hat{W}(i, j; 2)|$. The vertical estimation $\hat{W}_v(i, j)$ and variation $\delta_v(i, j)$ of $W(i, j)$ are calculated similarly. Finally, we estimate $W(i, j)$ as

$$\hat{W}(i, j) = \frac{\delta_v(i, j)\hat{W}_h(i, j) + \delta_h(i, j)\hat{W}_v(i, j)}{\delta_h(i, j) + \delta_v(i, j)}, \quad (2)$$

which is edge sensing.

However, the above interpolation has aliasing at regions of horizontal or vertical one-pixel-width stripes. As denoted by the x and y values in Fig. 4, both the horizontal and the vertical partial derivatives are zeros. Therefore, we cannot determine the preferred interpolation direction, which causes significant visible artifacts (see Fig. 5 (d)). We reduce such aliasing artifacts as follows. We first detect aliasing regions, i.e., horizontal or vertical one-pixel-width stripes. Our finding is that in the aliasing regions, the horizontal and the vertical variations are both very small, while the diagonal and the anti-diagonal variations are both comparatively large. We may compute the diagonal first (denoted by ∂_d) and second order (denoted by ∂_d^2) partial derivatives, and the anti-diagonal first (denoted by ∂_a) and second order (denoted by ∂_a^2) partial derivatives by

$$\begin{aligned} \partial_d W(i, j) &= \frac{W(i+1, j+1) - W(i, j)}{\sqrt{2}}, \\ \partial_d^2 W(i, j) &= \frac{W(i+1, j+1) + W(i-1, j-1) - 2W(i, j)}{2}, \\ \partial_a W(i, j) &= \frac{W(i+1, j-1) - W(i, j)}{\sqrt{2}}, \\ \partial_a^2 W(i, j) &= \frac{W(i+1, j-1) + W(i-1, j+1) - 2W(i, j)}{2}. \end{aligned}$$

We compute the diagonal variation $\delta_d(i, j)$ by

$$\delta_d(i, j) = \sum_{\Delta \in \{1, -1\}} |\partial_d W(i+\Delta, j)| + |\partial_d W(i, j+\Delta)| \\ |\sqrt{2}\partial_d^2 W(i+\Delta, j)| + |\sqrt{2}\partial_d^2 W(i, j+\Delta)|.$$

Similarly, we compute the anti-diagonal variation $\delta_a(i, j)$. We estimate the confidence of $W(i, j)$ that belongs to aliasing regions as

$$\gamma(i, j) = \frac{4\sqrt{2}(\delta_h(i, j) + \delta_v(i, j)) + \kappa}{\delta_d(i, j) + \delta_a(i, j) + \kappa}, \quad (3)$$

where $\kappa > 0$ is a small constant. Dividing δ_d and δ_a in the above equation can exclude smooth regions. Note that we consider the first and the second order partial derivatives to be of the same magnitude.

If the confidence $\gamma(i, j)$ is below a given threshold, we consider $W(i, j)$ as part of aliasing regions. When the horizontal variation is equal to or less than the vertical variation, we deem the edge direction of the pixel to be horizontal. Otherwise, we consider its edge direction as vertical. For each missing pixel in the aliasing region, we calculate all the edge directions (horizontal or vertical) of its neighbors in a window. If the majority of the edge directions in the neighborhood are horizontal, the edge direction of the pixel will also be horizontal. Similarly, if the majority of the edge directions in the neighborhood are vertical, then the edge direction of the pixel will be vertical. The effectiveness of our aliasing reduction process is shown in Fig. 5. We can observe that the demosaicked images with the aliasing reduction process have the best visual quality, even in the presence of serve noise.

B. RGB-Channel Reconstruction

We reconstruct RGB channels with the guidance of the recovered W channel by image decomposition. Our fundamental idea is constructing a relation between W and RGB values at available RGB pixels. Meanwhile, the relation should be fairly invariant among different image scales, which facilitates the

recovery of full-size RGB channels. Thus, we assume a global linear relation between W channel and R, G, or B channel, i.e., $\mathbf{C} = \mathbf{L} \odot \mathbf{I} + \mathbf{E}$, where \mathbf{C} is one of RGB channels, \mathbf{I} is W channel, and \mathbf{L} and \mathbf{E} are coefficients. Our global linear model is different from that of the guided filter [71], which uses a local linear model.

In the noiseless case, demosaicking methods should have two constraints. First, the known RGB values should be left unaltered in the demosaicked image. This is the reason that residual interpolation methods [10], [24], [25] estimate the tentative demosaicked image with the guided filter and then use bilinear interpolation to eliminate residual errors at available RGB pixels. However, the approach does not work for RGBW demosaicking as the RGB pixels are usually sampled sparsely. Simple interpolation methods will cause false color artifacts. It should be noted that keeping known pixel values can prevent oversmoothing in the presence of noise. Second, the linear coefficients \mathbf{L} and \mathbf{E} should be smooth, which makes it easier to upsample \mathbf{L} and \mathbf{E} with simple methods, e.g., bilinear interpolation. So we may use the following image decomposition model to build the relation between W channel and each of RGB channels:

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{E}} \rho_1(\mathbf{L}) + \tau \rho_2(\mathbf{E}) \\ \text{s.t. } \mathbf{M} \odot (\mathbf{C} - \mathbf{L} \odot \mathbf{I} - \mathbf{E}) = \mathbf{0}, \end{aligned} \quad (4)$$

where $\rho_1(\cdot)$ and $\rho_2(\cdot)$ are loss functions for smoothness, and the parameter $\tau > 0$ balances the two loss functions. \mathbf{M} is a mask to identify known values, in which 1 for the known value and 0 for the unknown value.

Gradient sparse priors are good choices for $\rho_1(\cdot)$ and $\rho_2(\cdot)$, e.g., l_1 -norm and l_0 -pseudo-norm (number of nonzero elements) of the gradients [72]. However, they usually lead to nonsmooth or nonconvex optimizations, which are hard to solve efficiently. So we consider l_2 -norm of gradients for fast computation. In order to get smooth linear coefficients, we apply a predefined Gaussian smoothing filter \mathbf{f}_g . Then our image decomposition model is finally formulated as:

$$\begin{aligned} \min_{\mathbf{L}, \mathbf{E}} \frac{1}{2} \sum_i \|\mathbf{L} * \mathbf{f}_i\|_F^2 + \tau \|\mathbf{E} * \mathbf{f}_i\|_F^2 \\ \text{s.t. } \mathbf{M} \odot (\mathbf{C} - (\mathbf{L} * \mathbf{f}_g) \odot \mathbf{I} - (\mathbf{E} * \mathbf{f}_g)) = \mathbf{0}, \end{aligned} \quad (5)$$

where \mathbf{f}_i denotes different derivative filters. We provide its solution process in the following section. In the experiments, we use two directional first order derivative filters, i.e., $\mathbf{f}_1 = (-1, 1)$, $\mathbf{f}_2 = (-1, 1)^T$.

In Fig. 6, we outline the process of recovering R channel with the guide of W channel. The reconstructions of G and B channels are performed in the same manner. We first downsample the full-size W channel $\tilde{\mathbf{I}}$ according to available R pixels. Then we have the downsampled R channel \mathbf{C} and its corresponding mask \mathbf{M} , as well as the corresponding downsampled W channel \mathbf{I} . Note that the downsampling operations should be applied to the same positions. We compute the linear coefficients \mathbf{L} and \mathbf{E} by solving problem (5). With the optimal \mathbf{L} and \mathbf{E} , we use bilinear interpolation to upsample $\mathbf{L} * \mathbf{f}_g$ and $\mathbf{E} * \mathbf{f}_g$ and get $\tilde{\mathbf{L}}$ and $\tilde{\mathbf{E}}$ in full-size. Finally, the full-size R channel $\hat{\mathbf{C}}$ is computed as $\hat{\mathbf{C}} = \tilde{\mathbf{L}} \odot \tilde{\mathbf{I}} + \tilde{\mathbf{E}}$. Since

we use a mask to specify known values, our RGB-channel reconstruction method works with arbitrary RGBW CFAs. Due to the limited space, we describe the processing details for the general CFAs in the supplementary file.

IV. SOLVING OUR IMAGE DECOMPOSITION MODEL

The proposed image decomposition model in (5) is convex. It can be solved by the alternating direction method (ADM) with convergence guarantee [33]. However, the subproblems of ADM for solving problem (5) require the computation of large scale matrix inversions, which results in high time complexity. Thus we use the LADM (linearized ADM) to solve problem (5) efficiently. The convergence of LADM for solving convex problems with two blocks has been well established [33].

When applying LADM to (5), we first reformulate it as:

$$\begin{aligned} \min_{\mathbf{l}, \mathbf{e}} \frac{1}{2} \sum_i \|\mathbf{F}_i \mathbf{l}\|_2^2 + \tau \|\mathbf{F}_i \mathbf{e}\|_2^2 \\ \text{s.t. } \mathbf{P}(\mathbf{c} - \mathbf{D} \mathbf{F}_g \mathbf{l} - \mathbf{F}_g \mathbf{e}) = \mathbf{0}, \end{aligned} \quad (6)$$

where \mathbf{F}_i is the convolution matrix of \mathbf{f}_i , \mathbf{F}_g is the convolution matrix of \mathbf{f}_g , $\mathbf{P} = \text{Diag}(\text{vec}(\mathbf{M}))$, $\mathbf{l} = \text{vec}(\mathbf{L})$, $\mathbf{e} = \text{vec}(\mathbf{E})$, $\mathbf{c} = \text{vec}(\mathbf{C})$, and $\mathbf{D} = \text{Diag}(\text{vec}(\mathbf{I}))$.

The augmented Lagrangian function of problem (6) is:

$$\begin{aligned} \mathcal{L}(\mathbf{l}, \mathbf{e}, \mathbf{z}) = \frac{1}{2} \sum_i \|\mathbf{F}_i \mathbf{l}\|_2^2 + \tau \|\mathbf{F}_i \mathbf{e}\|_2^2 \\ + \langle \mathbf{z}, \mathbf{P}(\mathbf{c} - \mathbf{D} \mathbf{F}_g \mathbf{l} - \mathbf{F}_g \mathbf{e}) \rangle + \frac{\beta}{2} \|\mathbf{P}(\mathbf{c} - \mathbf{D} \mathbf{F}_g \mathbf{l} - \mathbf{F}_g \mathbf{e})\|_2^2, \end{aligned} \quad (7)$$

where \mathbf{z} is the Lagrange multiplier and $\beta > 0$ is the penalty parameter. When updating \mathbf{l} , the objective function $\mathcal{L}(\mathbf{l}, \mathbf{e}^k, \mathbf{z}^k)$ is equivalent to

$$\mathcal{G}(\mathbf{l}) = \frac{1}{2} \sum_i \|\mathbf{F}_i \mathbf{l}\|_2^2 + \frac{\beta}{2} \left\| \mathbf{P}(\mathbf{c} - \mathbf{D} \mathbf{F}_g \mathbf{l} - \mathbf{F}_g \mathbf{e}^k) + \frac{\mathbf{z}^k}{\beta} \right\|_2^2.$$

We may set the derivative of \mathcal{G} w.r.t. \mathbf{l} : $\mathcal{G}'(\mathbf{l}) = \mathbf{H} \mathbf{l} - \beta(\mathbf{P} \mathbf{D} \mathbf{F}_g)^T \mathbf{x}^k$ to zero and get $\mathbf{l}^{k+1} = \beta \mathbf{H}^{-1} (\mathbf{P} \mathbf{D} \mathbf{F}_g)^T \mathbf{x}^k$, where $\mathbf{H} = \sum_i \mathbf{F}_i^T \mathbf{F}_i + \beta(\mathbf{P} \mathbf{D} \mathbf{F}_g)^T (\mathbf{P} \mathbf{D} \mathbf{F}_g)$, and $\mathbf{x}^k = \mathbf{P}(\mathbf{c} - \mathbf{F}_g \mathbf{e}^k) + \mathbf{z}^k / \beta$. The computation complexity of \mathbf{H}^{-1} is very high. By LADM, we linearize $\mathcal{G}(\mathbf{l})$ at \mathbf{l}^k and obtain:

$$\begin{aligned} \mathbf{l}^{k+1} = \arg \min_{\mathbf{l}} \mathcal{G}(\mathbf{l}^k) + \langle \mathcal{G}'(\mathbf{l}^k), \mathbf{l} - \mathbf{l}^k \rangle + \frac{\lambda_{\mathbf{H}}}{2} \|\mathbf{l} - \mathbf{l}^k\|_2^2 \\ = \mathbf{l}^k - \frac{\mathcal{G}'(\mathbf{l}^k)}{\lambda_{\mathbf{H}}}, \end{aligned} \quad (8)$$

where $\lambda_{\mathbf{H}}$ is the largest eigenvalue of \mathbf{H} , which can be calculated in advance.

Similarly, we update \mathbf{e} by

$$\begin{aligned} \mathbf{e}^{k+1} = \arg \min_{\mathbf{e}} \mathcal{S}(\mathbf{e}^k) + \langle \mathcal{S}'(\mathbf{e}^k), \mathbf{e} - \mathbf{e}^k \rangle + \frac{\lambda_{\mathbf{Q}}}{2} \|\mathbf{e} - \mathbf{e}^k\|_2^2 \\ = \mathbf{e}^k - \frac{\mathcal{S}'(\mathbf{e}^k)}{\lambda_{\mathbf{Q}}}, \end{aligned} \quad (9)$$

where

$$\mathcal{S}(\mathbf{e}) = \frac{\tau}{2} \sum_i \|\mathbf{F}_i \mathbf{e}\|_2^2 + \frac{\beta}{2} \left\| \mathbf{P}(\mathbf{c} - \mathbf{D} \mathbf{F}_g \mathbf{l}^{k+1} - \mathbf{F}_g \mathbf{e}^k) + \frac{\mathbf{z}^k}{\beta} \right\|_2^2,$$

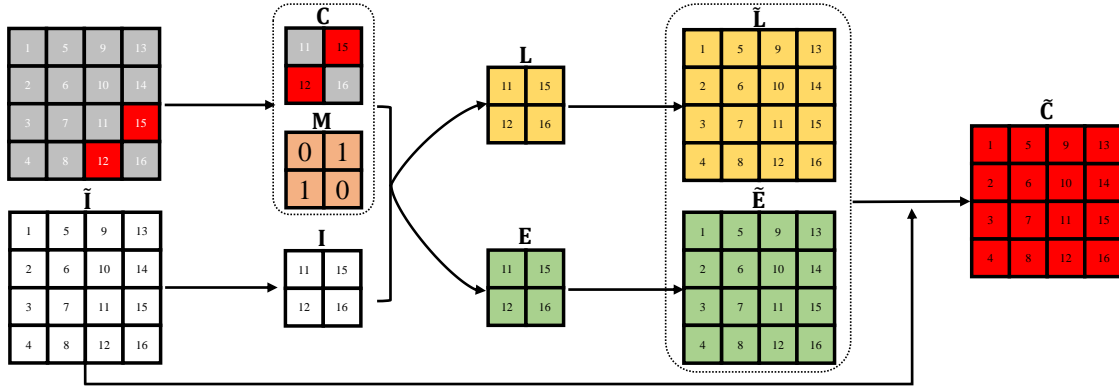


Fig. 6. The process of recovering each of RGB channels with the guide of W channel. Here takes the R channel as an example. The number within a pixel identifies its position in the full-size raw image except the 0-1 numbers in M, which represent values.

$S'(e) = Qe - \beta(PF_g)^T y^k$, $Q = \tau \sum_i F_i^T F_i + \beta(PF_g)^T (PF_g)$, $y^k = P(c - DF_g l^{k+1}) + z^k / \beta$, and λ_Q is the largest eigenvalue of Q. We update the multiplier z by

$$z^{k+1} = z^k + \beta(P(c - DF_g l^{k+1} - F_g e^{k+1})). \quad (10)$$

The stopping criteria are:

$$\max\{\|e^{k+1} - e^k\|_\infty, \|l^{k+1} - l^k\|_\infty\} < \varepsilon \quad (11)$$

$$\text{and } \|P(c - DF_g l^{k+1} - F_g e^{k+1})\|_\infty < \varepsilon. \quad (12)$$

We summarize the whole solution process in Algorithm 1. It should be noted that we implement Algorithm 1 in matrix form rather than vector form, e.g., $F_i l$ is computed as $L * f_i$. So we do not use large scale sparse matrices, e.g., F_i and P.

Algorithm 1 Algorithm for Solving Problem (6)

Input: Predefined Gaussian filter f_g , precomputed λ_H and λ_Q , positive parameters τ , $\beta = 1$, and $\varepsilon = 0.5$, $l^0 = \text{vec}(C \oslash I)$, $e^0 = 0$, $z^0 = 0$.

- 1: **while** the stopping conditions (11)-(12) are not met **do**
- 2: Update l by (8).
- 3: Update e by (9).
- 4: Update the multiplier z by (10).
- 5: **end while**

Output: l and e.

V. EXPERIMENTS

In this section, we conduct experiments to test the advantage of our demosaicking method. We first confirm the universal property of our demosaicking method. Then we compare the results quantitatively and visually with the simulated raw images that have two different noise levels. Finally, we carry out comparisons with real raw images visually, where ground-truth images are not available.

A. Experimental Settings

1) *Compared Methods:* We evaluate our W-channel interpolation and RGB-channel reconstruction separately. Interpolation-friendly CFAs for the W channel have the checkerboard (see Figs. 1 (h)-(l)) or the subsampled checkerboard patterns (see Figs. 1 (m)-(n)). The W-channel interpolation for the checkerboard pattern is similar to the G-channel

interpolation for the Bayer CFA. However, almost all of the methods interpolating the G channel for the Bayer CFA rely on the R and B values [23], [52], [75]. Consequently, they cannot be applied for W-channel interpolation. So we include the bilinear interpolation and the edge-aware interpolation in (2) without the second order partial derivative in (1) for comparison.

When comparing methods of reconstructing the RGB channels, we use our W-channel interpolation with aliasing reduction for all the methods. Many RGBW demosaicking methods require the linear relation assumption between W and RGB values, which may be not hold or varies for different cameras [38], [39], [55]. So we do not include these methods for comparison. There are only few public source codes of RGBW demosaicking methods. Since the bilinear interpolation in color difference domain [16], [43] and the pansharpening-based demosaicking method [48] work with various cameras and lighting conditions, we implement the two methods ourselves. The pansharpening-based method consists of three choices. We use our W-channel interpolation with aliasing reduction to recover W channel. We apply the state-of-the-art Bayer demosaicking method in [24] to demosaick the Bayer raw image. Due to the superiority [49], we use the GSA (Gram Schmidt Adaptive) [73], [74] pansharpening method. We also develop the guided filter-based method for comparison. For the recovery of W channel, we use our W-channel interpolation with aliasing reduction. The RGB-channel reconstruction is similar to our method shown in Fig. 6. The only difference is that it utilizes the guided filter [71] to compute L and E analogically. For the details on how to use the guided filter with a mask, we refer the readers to [24].

2) *Compared CFAs:* For the simulated raw images, we test our demosaicking methods with all the seven interpolation-friendly RGBW CFAs mentioned in the paper, i.e., all the CFAs shown in the second row of Fig. 1. Since the pansharpening-based method [48] can only work with the Kodak CFA [16] and Honda CFA [21], we carry out the comparisons with the two CFAs. For the real raw images, the Kodak CFA [16] and Honda CFA [21] are the RGBW CFAs we can get. Thus, we employ the two CFAs for comparison. The RGBW CFAs have been shown to outperform the Bayer CFA in low-light conditions. Since we evaluate RGBW demo-

TABLE II
THE UNIVERSAL PROPERTY OF OUR DEMOSAICKING METHOD. THE AVERAGE CPSNR AND S-CIELAB VALUES ARE REPORTED.

Metric	Noise	Clean			100 lux			5 lux		
	CFA pattern (Percentage of W)	D65	A	F12	D65	A	F12	D65	A	F12
CPSNR	Kodak CFA [16] (50%)	52.23	51.61	51.21	48.24	47.73	47.49	37.35	35.98	35.68
	Sony CFA [17] (50%)	52.37	51.69	51.27	48.44	47.90	47.66	37.37	36.00	35.70
	Yamagami CFA [18] (50%)	52.40	51.72	51.30	48.66	48.11	47.86	37.40	36.02	35.72
	Kaizu CFA [19] (50%)	52.17	51.69	51.35	48.72	48.23	48.00	37.37	36.00	35.70
	Hamilton CFA [20] (50%)	51.04	50.88	50.60	48.20	47.81	47.61	37.32	35.97	35.67
	Honda CFA [21] (75%)	52.18	51.93	51.71	48.80	48.30	48.10	37.40	36.02	35.72
	Random CFA [22] (75%)	52.30	52.00	51.74	48.74	48.20	48.01	37.40	36.03	35.73
S-CIELAB	Kodak CFA [16] (50%)	0.072	0.075	0.084	0.112	0.124	0.135	0.771	1.256	1.428
	Sony CFA [17] (50%)	0.072	0.076	0.084	0.111	0.123	0.134	0.771	1.256	1.429
	Yamagami CFA [18] (50%)	0.067	0.071	0.080	0.104	0.116	0.127	0.770	1.256	1.428
	Kaizu CFA [19] (50%)	0.075	0.080	0.088	0.109	0.123	0.134	0.771	1.259	1.433
	Hamilton CFA [20] (50%)	0.090	0.091	0.101	0.126	0.138	0.151	0.774	1.259	1.433
	Honda CFA [21] (75%)	0.088	0.092	0.101	0.120	0.134	0.146	0.775	1.262	1.436
	Random CFA [22] (75%)	0.087	0.092	0.102	0.125	0.142	0.154	0.774	1.259	1.433

saicking methods instead of developing new CFA patterns, we do not include the Bayer CFA for comparison.

3) *Datasets and Metrics*: We follow a similar approach in [10] to simulate RGBW raw images. We use the widely used CAVE dataset [76] as the true reflectance data of the scenes. It has 32 multispectral images with 31 bands, which are captured at every 10nm from 400nm to 700nm. These images have the same size 512×512. We include the CIE D65, CIE A, and CIE F12 illuminants for comparison. We use the spectral sensitivity functions of the RGBW camera in [39], which are shown in Fig. 3. The ground-truth RGBW images are obtained by the sum of the product of the reflectance, spectral sensitivity function, and illuminant. We normalize all the pixel values into [0, 1] and quantify to 10 bits. As in [77], we add the photon shot noise and readout noise. The simulation parameters are the same except using two illumination levels (i.e., 5 lux and 100 lux). We use two measures, CPSNR (or PSNR for a single channel) and S-CIELAB [78], for quantitative comparisons. CPSNR (or PSNR) computes the pixel-wise difference between the demosaicked image and its ground truth. A larger value may indicate better performance. S-CIELAB is a perceptual metric, where a larger value may indicate poorer visual quality. We use two RGBW cameras to capture real raw images, which use the Kodak CFA and Honda CFA, respectively. For each camera, we capture six raw images in both outdoor and indoor conditions with the automatic exposure settings. All the raw images are 10 bits per pixel with sizes of 1440×2560 and 6944×9280. We obtain these images in normal-light conditions and do not perform any processing, e.g., denoising. Since real raw images do not have ground truth, we can only compare in visual quality.

4) *Parameter Settings*: For our proposed method, we use the threshold 0.1 for aliasing confidence γ , and 15×15 window for edge direction estimation. In Eq. (5), we set the parameter $\tau = 1$, and 9×9 window for the Gaussian smoothing filter \mathbf{f}_g with standard deviation 1.25. Note that we employ grid search and determine the parameters empirically. In order to be fair, we use the same Gaussian smoothing filter for the guided filter-based method. So we exclude a 10-pixel border when computing CPSNR (or PSNR) and S-CIELAB values.

B. Evaluation of The Universal Property

We compare seven CFAs to evaluate the universal property of our demosaicking method. The average CPSNR and S-CIELAB values are reported in Table II. In terms of CPSNR, the CFAs with 50% W pixels perform better with clean images, while the CFAs with 75% W pixels achieve higher values with noisy images. However, the Random CFA [22] with 75% W pixels appears to be the exception, as it has superior values even with clean images. When comparing with S-CIELAB, the CFAs with 50% W pixels outperform those CFAs with 75% W pixels. The Yamagami CFA [18] is the best for overall performance, and the Hamilton CFA [20] is the worst for overall performance. It can be concluded that randomly and evenly assigning RGB pixels for RGBW CFAs is highly important. We want to emphasize that we are not attempting to create new CFAs.

C. Comparison with Simulated Raw Images

We first quantitatively compare our W-channel interpolation method. The average PSNR values are shown in Tabel III. The best values are in boldface. We can see that our method outperforms bilinear interpolation and edge-aware interpolation on both clean and noisy images. The PSNR values of our method without and with aliasing reduction are almost the same. However, it is evident from Fig. 5 that our method with aliasing reduction produces better visual quality. This validates the advantage of our W-channel interpolation.

Then we quantitatively compare our RGB-channel reconstruction method. The average CPSNR and S-CIELAB values are shown in Tabel IV. The best values are in boldface. In terms of CPSNR, we observe that our method outperforms the other methods, particularly in the case of clean and low-level noisy images. In terms of S-CIELAB, we can see that our method performs better than the other methods. For high-level noisy images at 5 lux, the pansharpening-based and the guided filter-based methods have competitive CPSNR and S-CIELAB values. With the same W channels, our method achieves superior performance compared to the pansharpening-based and the guided filter-based methods. This demonstrates the superiority of our RGB-channel reconstruction method.

TABLE III
EVALUATION OF OUR W-CHANNEL INTERPOLATION METHOD. THE AVERAGE PSNR VALUES ARE REPORTED. "WO AR" AND "W AR" STAND FOR WITHOUT AND WITH ALIASING REDUCTION, RESPECTIVELY.

W pattern	Noise	Clean			100 lux			5 lux		
	Method	D65	A	F12	D65	A	F12	D65	A	F12
Checkerboard	Bilinear interpolation	43.15	43.06	42.91	41.10	41.03	40.94	29.78	29.35	29.31
	Edge-aware interpolation	45.56	45.60	45.42	42.19	42.16	42.08	29.90	29.47	29.44
	Ours (wo AR)	46.80	46.86	46.64	42.65	42.63	42.55	30.00	29.56	29.52
	Ours (w AR)	46.80	46.86	46.64	42.65	42.63	42.55	30.00	29.56	29.52
Subsampled checkerboard	Bilinear interpolation	46.18	46.08	45.92	42.25	42.20	42.14	29.96	29.51	29.48
	Edge-aware interpolation	47.67	47.69	47.53	42.69	42.66	42.62	29.97	29.53	29.50
	Ours (wo AR)	50.31	50.39	50.11	42.78	42.75	42.70	30.06	29.60	29.58
	Ours (w AR)	50.31	50.40	50.11	42.78	42.76	42.70	30.06	29.61	29.58

TABLE IV
EVALUATION OF OUR RGB-CHANNEL RECONSTRUCTION METHOD. THE AVERAGE CPSNR AND S-CIELAB VALUES ARE REPORTED. NOTE THAT ALL THE COMPARED METHODS USE OUR W-CHANNEL INTERPOLATION WITH ALIASING REDUCTION.

CFA pattern	Metric	Noise	Clean			100 lux			5 lux		
		Method	D65	A	F12	D65	A	F12	D65	A	F12
Kodak	CPSNR	Bilinear interpolation	38.82	39.15	39.14	37.69	37.94	37.94	35.73	34.90	34.67
		Pansharpening	50.03	49.20	48.88	47.16	46.33	46.05	37.35	35.92	35.60
		Guided filter	51.46	51.09	50.79	48.96	48.44	48.19	37.41	36.03	35.73
		Ours	52.23	51.61	51.21	49.12	48.57	48.32	37.42	36.04	35.74
	S-CIELAB	Bilinear interpolation	0.443	0.438	0.433	0.504	0.512	0.509	0.802	1.278	1.433
		Pansharpening	0.107	0.115	0.129	0.141	0.160	0.177	0.774	1.258	1.433
		Guided filter	0.080	0.086	0.095	0.106	0.121	0.132	0.772	1.259	1.432
		Ours	0.072	0.075	0.084	0.103	0.117	0.128	0.772	1.258	1.431
Honda	CPSNR	Bilinear interpolation	38.31	38.66	38.66	36.85	37.09	37.10	35.36	34.61	34.40
		Pansharpening	49.24	48.47	48.22	45.76	45.00	44.75	37.32	35.96	35.64
		Guided filter	50.56	50.55	50.45	48.13	47.72	47.56	37.33	35.97	35.68
		Ours	52.18	51.93	51.71	48.45	48.00	47.83	37.35	35.98	35.69
	S-CIELAB	Bilinear interpolation	0.469	0.462	0.456	0.545	0.554	0.552	0.809	1.279	1.432
		Pansharpening	0.141	0.151	0.167	0.187	0.207	0.226	0.779	1.270	1.446
		Guided filter	0.108	0.120	0.133	0.129	0.151	0.167	0.779	1.269	1.445
		Ours	0.088	0.092	0.101	0.123	0.142	0.157	0.777	1.267	1.442

Due to the limited space, we present part of the visual comparison in Fig. 7, which uses the Honda CFA and CIE D65 illuminant at 100 lux. We can see that the visual quality of our method is better than that of other methods, especially in color fidelity. The bilinear interpolation and pansharpening-based method can result in false color along highly contrasted edges. The guided filter-based method suffers from halo artifacts near strong edges and may have color desaturation. Our method fixes known values and hence suppresses these limitations.

D. Comparison with Real Raw Images

The comparison with real raw images is typically the most important evaluation for image demosaicking. For real raw images, we do not have the ground truth. So we can only conduct visual comparison. Although all the raw images are taken in normal-light conditions, they are inevitably degraded by noise. For all the compared methods, we directly perform demosaicking without denoising.

Fig. 8 shows the visual comparison with the Kodak CFA. We can see that our method has better visual quality than the other methods (Please read the captions for the descriptions on visual difference.). It should be noted that the pansharpening-based method produces severe false color at color checker regions. The reason is as follows. The GSA pansharpening method [73], [74] requires a synthetic intensity channel to approximate the W channel. This involves solving a least-squares problem where the parameters can either be estimated

globally or locally. The global approach calculates the parameters using all pixels, resulting in uniform parameters across all pixels. However, the global approach may produce false colors for large images, which is the case here. The local approach, on the other hand, computes the parameters using only pixels in a local region (e.g., 31×31 pixels), and averages overlapping estimates. However, the local approach may not have full row rank for large smooth regions, leading to low-quality images. Additionally, the local approach yields lower CPSNR values compared to the global approach, and may cause saturated regions to be too bright. It can be difficult to choose between the two approaches. Therefore, we use the GSA's default setting, which uses the global approach.

Fig. 9 shows the visual comparison with the Honda CFA. We can see that the visual quality of our method is better than that of the other methods (Please read the captions for the descriptions on visual difference.). It should be noted that the guided filter-based method does not fix the known RGB values. The guided filter [71] uses the least-squares method to compute the coefficients. When the changing colors occupy small proportions, they will be ignored and assigned to be the dominant colors. So the guided filter-based method usually produces false color or color desaturation at small color changing regions. Since our method keeps the known RGB values unchanged, it preserves the small color changing regions well. The two more convincing comparisons further validate the advantage of our method.

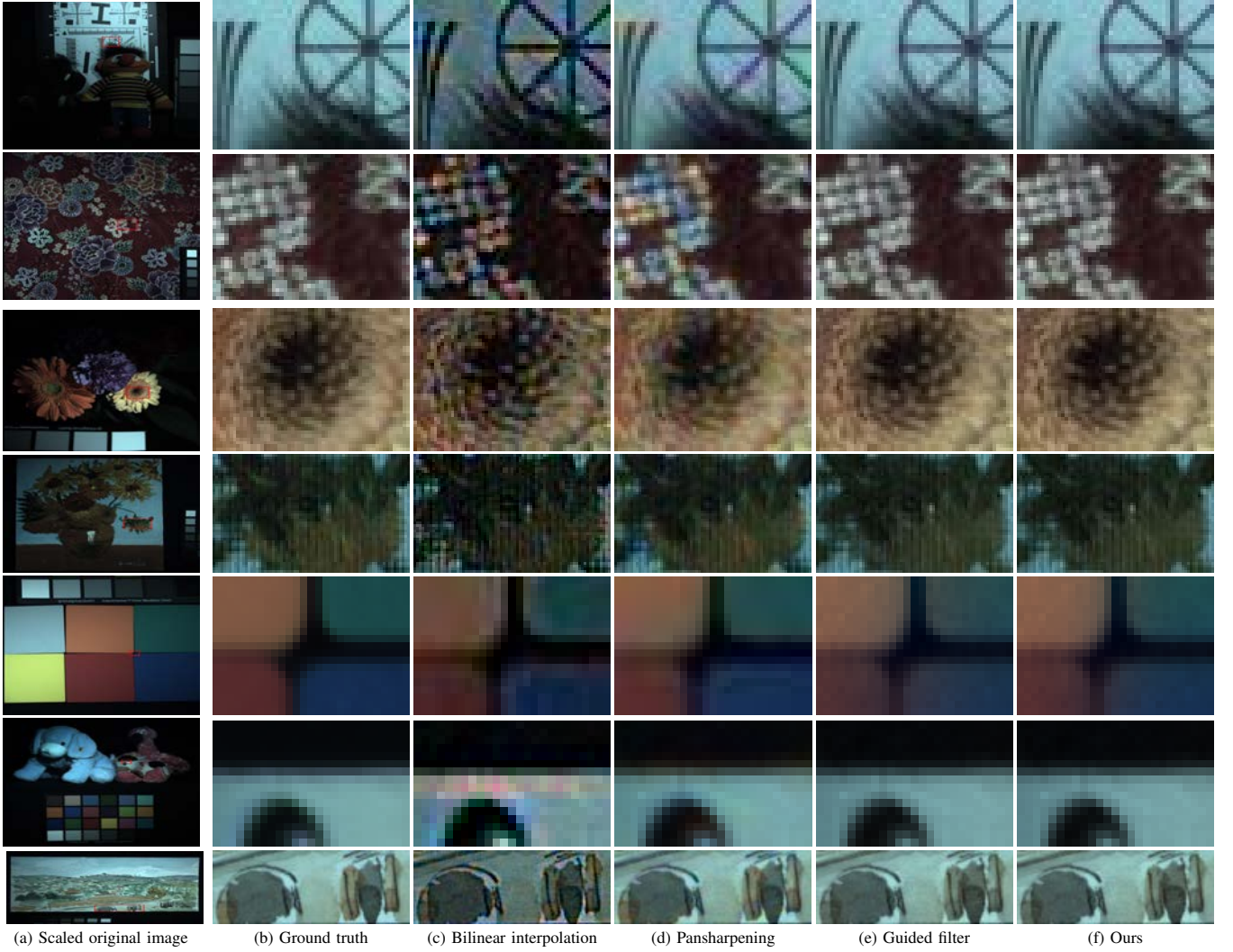


Fig. 7. Blowups of some demosaicked images by the Honda CFA with simulated raw images under the CIE D65 illuminant at 100 lux. In each group, (a) is the scaled original noisy image, in which the red rectangle indicates the selected patch to blow up; (b) is the clean ground truth; (c)-(e) and (f) are the images demosaicked by the bilinear interpolation, pansharpening-based method, guided filter-based method, and our method, respectively. From all the seven groups of images, we can clearly see that the images demosaicked by our method have better subjective quality, while those by the other three methods have severe false color artifacts or color desaturation.

VI. CONCLUSIONS

In this paper, we have proposed a universal demosaicking method for interpolation-friendly RGBW CFAs, which has two sequential steps. To accomplish this, we did not utilize specific characteristics of given cameras and illuminants. More specifically, we interpolated missing W values using only known ones. We also developed an aliasing reduction process to remove aliasing artifacts. With the recovered W channel, we introduced an image decomposition model to estimate the relation between W channel and each of RGB channels with available RGB pixels, which can be easily generalized to the full-size demosaicked image. Extensive experiments on both simulated and real raw images have demonstrated the superiority of our demosaicking method. In the future, we will adapt our demosaicking method to joint denoising and demosaicking. We will also consider handling overexposure W pixels in our demosaicking method.

ACKNOWLEDGMENT

J. Li is supported by NSF of China (grant nos. 61802269 and 61972132) and the Fundamental Research Funds for the Central Universities.

REFERENCES

- [1] C. Bai, J. Li, Z. Lin, J. Yu, and Y.-W. Chen, “Penrose demosaicking,” *IEEE Transactions on Image Processing*, vol. 24, no. 5, pp. 1672–1684, 2015.
- [2] B. E. Bayer, “Color imaging array,” Jul. 20 1976, U.S. Patent 3971065.
- [3] A. Chakrabarti, W. T. Freeman, and T. Zickler, “Rethinking color cameras,” in *Proceedings of the IEEE International Conference on Computational Photography*. IEEE, 2014, pp. 1–8.
- [4] J. Li, C. Bai, Z. Lin, and J. Yu, “Automatic design of high-sensitivity color filter arrays with panchromatic pixels,” *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 870–883, 2017.
- [5] J. Zhang, J. Jia, A. Sheng, and K. Hirakawa, “Pixel binning for high dynamic range color image sensor using square sampling lattice,” *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2229–2241, 2018.
- [6] “OPPO Reno7,” <https://www.oppo.com/en/smartphones/series-reno/reno7/>, August 25, 2022.
- [7] “VIVO X80,” <https://www.vivo.com/en/products/x80/>, August 25, 2022.

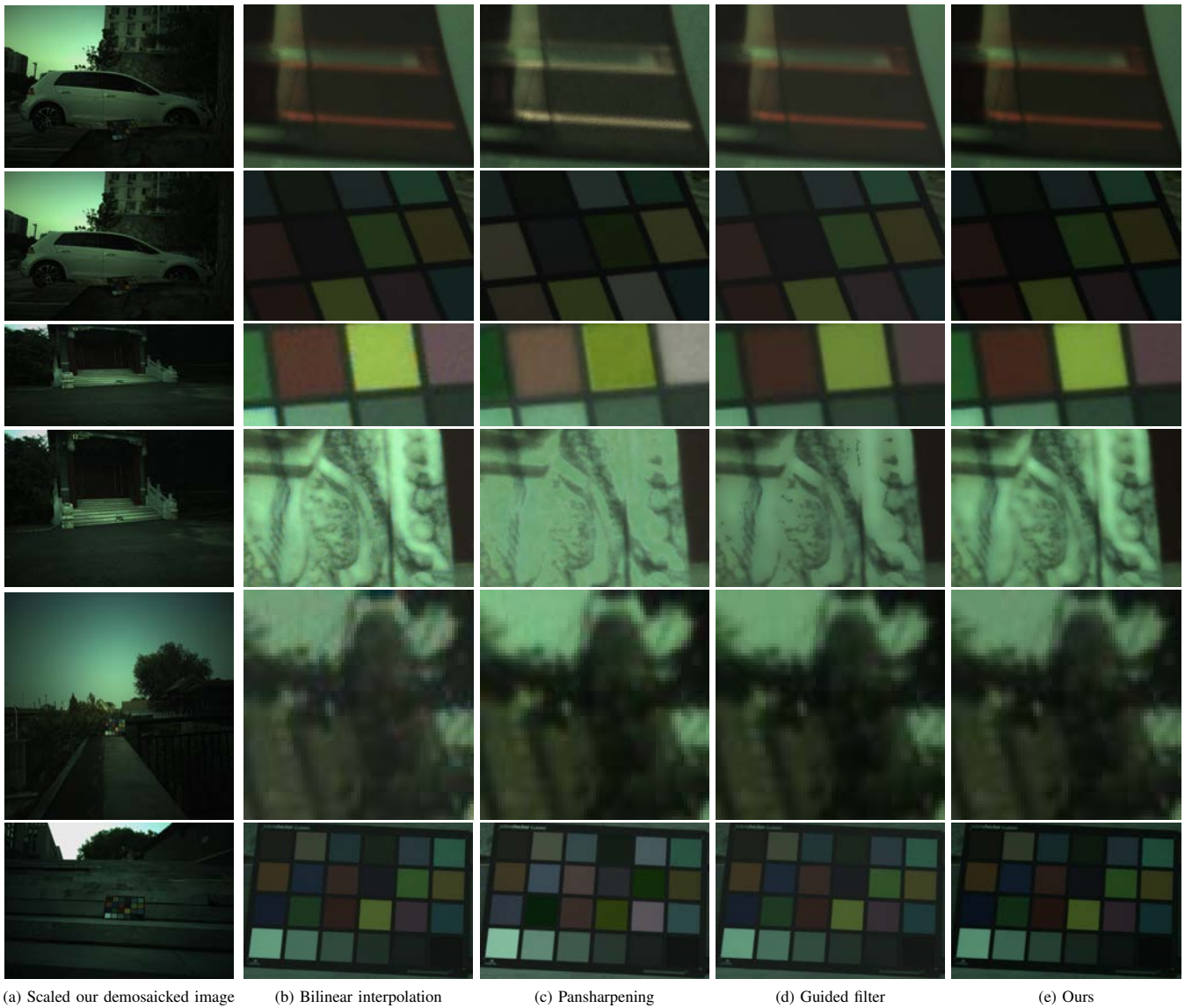


Fig. 8. Blowups of some demosaicked images by the Kodak CFA with real raw images. In each group, (a) is the scaled our demosaicked image, in which the red rectangle indicates the selected patch to blow up; (b)-(d) and (e) are the images demosaicked by the bilinear interpolation, pansharpening-based method, guided filter-based method, and our method, respectively. We can see that bilinear interpolation produces severe false color artifacts at texture regions. The pansharpening-based method has false color and color desaturation. The guided filter-based method smooths tiny textures. Our method generates high quality demosaicked images.

- [8] "Tecno Camon 19," <https://www.tecno-mobile.com/phones/product-detail/product/camon-19/>, August 25, 2022.
- [9] X. Li, B. Gunturk, and L. Zhang, "Image demosaicing: A systematic survey," in *Proceedings of the Visual Communications and Image Processing*, vol. 6822, SPIE, 2008, pp. 489–503.
- [10] Y. Monno, S. Kikuchi, M. Tanaka, and M. Okutomi, "A practical one-shot multispectral imaging system using a single image sensor," *IEEE Transactions on Image Processing*, vol. 24, no. 10, pp. 3048–3059, 2015.
- [11] E. B. Gindele and A. C. Gallagher, "Sparsely sampled image sensing device with color and luminance photosites," Nov. 5 2002, US Patent 6,476,865.
- [12] G. Luo, "Color filter array with sparse color sampling crosses for mobile phone image sensors," in *Proceedings of the International Image Sensor Workshop*, 2007, pp. 162–165.
- [13] J. Wang, C. Zhang, and P. Hao, "New color filter arrays of high light sensitivity and high demosaicking performance," in *Proceedings of the International Conference on Image Processing*. IEEE, 2011, pp. 3153–3156.
- [14] A. Chakrabarti, "Learning sensor multiplexing design through back-propagation," *Advances in Neural Information Processing Systems*, vol. 29, pp. 3081–3089, 2016.
- [15] M. Fujita, Y. Lee, E. Shim, K. Lee, B. Kim, and T. Kim, "Image sensor," Jun. 10 2021, US Patent App. 16/996,047.
- [16] J. T. Compton and J. F. Hamilton Jr, "Image sensor with improved light sensitivity," Feb. 1 2007, U.S. Patent 20070024931A1.
- [17] M. Tachi, "Image processing device, image processing method, and program pertaining to image correction," Nov. 20 2012, US Patent 8,314,863.
- [18] T. Yamagami, T. Sasaki, and A. Suga, "Image signal processing apparatus having a color filter with offset luminance filter elements," Jun. 21 1994, US Patent 5,323,233.
- [19] S. Kaizu, "Image processing apparatus, imaging device, image processing method, and program for reducing noise or false colors in an image," Jul. 4 2017, US Patent 9,699,429.
- [20] J. F. Hamilton Jr and J. T. Compton, "Processing color and panchromatic pixels," Sep. 25 2012, US Patent 8,274,715.
- [21] H. Honda, Y. Iida, Y. Egawa, and H. Seki, "A color CMOS imager with 4X4 white-RGB color filter array for increased low-illumination signal-to-noise ratio," *IEEE Transactions on Electron Devices*, vol. 56, no. 11, pp. 2398–2402, 2009.
- [22] P. Oh, S. Lee, and M. G. Kang, "Colorization-based RGB-white color

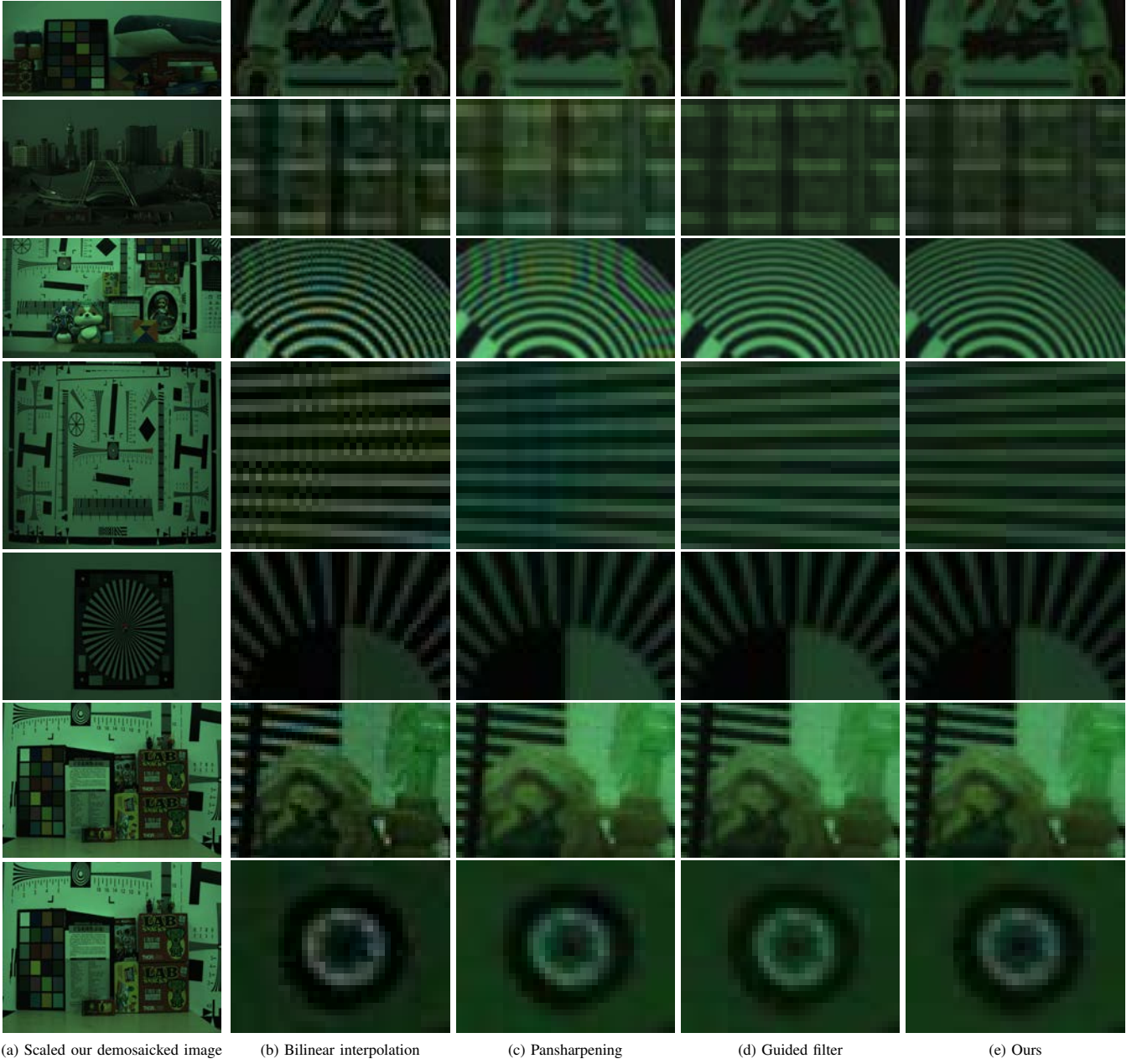
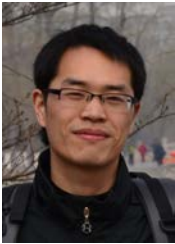


Fig. 9. Blowups of some demosaicked images by the Honda CFA with real raw images. The notations are the same as those in Fig. 8. We can see that bilinear interpolation and the pansharpening-based method have severe false color at highly textured regions or along edges. The guided filter-based method produces false color or color desaturation in small color changing regions. Our method could obtain demosaicked images with better visual quality.

- interpolation using color filter array with randomly sampled pattern,” *Sensors*, vol. 17, no. 7, p. 1523, 2017.
- [23] Y. Niu, J. Ouyang, W. Zuo, and F. Wang, “Low cost edge sensing for high quality demosaicking,” *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2415–2427, 2018.
- [24] D. Kiku, Y. Monno, M. Tanaka, and M. Okutomi, “Beyond color difference: Residual interpolation for color image demosaicking,” *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1288–1300, 2016.
- [25] W. Ye and K.-K. Ma, “Color image demosaicing using iterative residual interpolation,” *IEEE Transactions on Image Processing*, vol. 24, no. 12, pp. 5879–5891, 2015.
- [26] P. Hao, Y. Li, Z. Lin, and E. Dubois, “A geometric method for optimal design of color filter arrays,” *IEEE Transactions on Image Processing*, vol. 20, no. 3, pp. 709–722, 2011.
- [27] C. Bai, J. Li, Z. Lin, and J. Yu, “Automatic design of color filter arrays in the frequency domain,” *IEEE Transactions on Image Processing*, vol. 25, no. 4, pp. 1793–1807, 2016.
- [28] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, and R. M. Mersereau, “Demosaicking: color filter array interpolation,” *IEEE Signal Processing Magazine*, vol. 22, no. 1, pp. 44–54, 2005.
- [29] D. Menon and G. Calvagno, “Color image demosaicking: An overview,” *Signal Processing: Image Communication*, vol. 26, no. 8-9, pp. 518–533, 2011.
- [30] M. Gharbi, G. Chaurasia, S. Paris, and F. Durand, “Deep joint demosaicking and denoising,” *ACM Transactions on Graphics*, vol. 35, no. 6, pp. 1–12, 2016.
- [31] D. S. Tan, W.-Y. Chen, and K.-L. Hua, “Deepdemosaicking: Adaptive image demosaicking via multiple deep fully convolutional networks,” *IEEE Transactions on Image Processing*, vol. 27, no. 5, pp. 2408–2419, 2018.
- [32] F. Kokkinos and S. Lefkimmiatis, “Iterative joint image demosaicking and denoising using a residual denoising network,” *IEEE Transactions on Image Processing*, vol. 28, no. 8, pp. 4177–4188, 2019.
- [33] Z. Lin, R. Liu, and Z. Su, “Linearized alternating direction method

- with adaptive penalty for low-rank representation,” *Advances in Neural Information Processing Systems*, vol. 24, 2011.
- [34] G. Luo, “A novel color filter array with 75% transparent elements,” in *Digital Photography III*, vol. 6502. SPIE, 2007, pp. 273–280.
- [35] S. Sharif and Y. J. Jung, “Deep color reconstruction for a sparse color sensor,” *Optics Express*, vol. 27, no. 17, pp. 23 661–23 681, 2019.
- [36] A. M. Magnani, “Luminance-adaptive processing of hexa-deca RGBW color filter arrays in CMOS image sensors,” May 31 2022, US Patent 11,350,048.
- [37] J. E. Adams Jr and J. F. Hamilton Jr, “Adaptive color plane interpolation in single sensor color electronic camera,” Jul. 29 1997, US Patent 5,652,621.
- [38] A. Getman, J. Kim, and T.-C. Kim, “Imaging system having White-RGB color filter array,” in *Proceedings of the IEEE International Conference on Image Processing*. IEEE, 2010, pp. 569–572.
- [39] P.-H. Su, P.-C. Chen, and H. H. Chen, “Compensation of spectral mismatch to enhance WRGB demosaicking,” in *Proceedings of the IEEE International Conference on Image Processing*. IEEE, 2015, pp. 68–72.
- [40] C.-S. Wang and J.-W. Chong, “An improved white-RGB color filter array based CMOS imaging system for cell phones in low-light environments,” *IEICE Transactions on Information and Systems*, vol. 97, no. 5, pp. 1386–1389, 2014.
- [41] J. H. Choi, D. Y. Choi, and B. C. Song, “Demosaicking algorithm for white-RGB CFA images,” *IET Image Processing*, vol. 13, no. 5, pp. 811–816, 2019.
- [42] C. Zhang, Y. Li, J. Wang, and P. Hao, “Universal demosaicking of color filter arrays,” *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5173–5186, 2016.
- [43] M. Kumar, E. O. Morales, J. E. Adams, and W. Hao, “New digital camera sensor architecture for low light imaging,” in *Proceedings of the International Conference on Image Processing*. IEEE, 2009, pp. 2681–2684.
- [44] J. Kim and M. G. Kang, “Color interpolation algorithm for the Sony-RGBW color filter array,” *Electronic Imaging*, vol. 2018, no. 13, pp. 439–1–439–4, 2018.
- [45] K.-L. Chung, T.-H. Chan, and S.-N. Chen, “Effective three-stage demosaicking method for RGBW CFA images using the iterative error-compensation based approach,” *Sensors*, vol. 20, no. 14, p. 3908, 2020.
- [46] A. Levin, D. Lischinski, and Y. Weiss, “Colorization using optimization,” *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 689–694, 2004.
- [47] S. ho Lee, P. Oh, and M. G. Kang, “Three dimensional colorization based image/video reconstruction from white-dominant RGBW pattern images,” *Digital Signal Processing*, vol. 93, pp. 87–101, 2019.
- [48] C. Kwan, B. Chou, L.-Y. M. Kwan, and B. Budavari, “Debayering RGBW color filter arrays: A pansharpening approach,” in *Proceedings of the Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON)*. IEEE, 2017, pp. 94–100.
- [49] C. Kwan, J. Larkin, and B. Ayhan, “Demosaicing of CFA 3.0 with applications to low lighting images,” *Sensors*, vol. 20, no. 12, p. 3423, 2020.
- [50] C. Kwan and B. Chou, “Further improvement of debayering performance of RGBW color filter arrays using deep learning and pansharpening techniques,” *Journal of Imaging*, vol. 5, no. 8, p. 68, 2019.
- [51] C. Kwan and J. Larkin, “Further improvements of CFA 3.0 by combining inpainting and pansharpening techniques,” *Signal & Image Processing: An International Journal (SIPIJ) Vol.*, vol. 11, 2020.
- [52] K. Jeong, J. Kim, and M. G. Kang, “Color demosaicing of RGBW color filter array based on Laplacian pyramid,” *Sensors*, vol. 22, no. 8, p. 2981, 2022.
- [53] J. Gu, P. J. Wolfe, and K. Hirakawa, “Filterbank-based universal demosaicking,” in *Proceedings of the International Conference on Image Processing*. IEEE, 2010, pp. 1981–1984.
- [54] M. Singh and T. Singh, “Linear universal demosaicking of regular pattern color filter arrays,” in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2012, pp. 1277–1280.
- [55] M. Rafinazari and E. Dubois, “Demosaicking algorithms for RGBW color filter arrays,” *Electronic Imaging*, vol. 2016, no. 20, pp. 1–6, 2016.
- [56] L. Condat, “A generic variational approach for demosaicking from an arbitrary color filter array,” in *Proceedings of the IEEE International Conference on Image Processing*. IEEE, 2009, pp. 1625–1628.
- [57] H. Kim, S. Lee, and M. G. Kang, “Demosaicing of RGBW color filter array based on rank minimization with colorization constraint,” *Sensors*, vol. 20, no. 16, p. 4458, 2020.
- [58] P. Amba and D. Alleysson, “LMMSE demosaicing for multicolor CFAs,” in *Proceedings of the Color and Imaging Conference*, vol. 2018, no. 1. Society for Imaging Science and Technology, 2018, pp. 151–156.
- [59] C. Bai and J. Li, “Convolutional sparse coding for demosaicking with panchromatic pixels,” *Signal Processing: Image Communication*, vol. 77, pp. 20–27, 2019.
- [60] M. Kang and M. Jung, “Low-dimensional manifold model for demosaicking from a RGBW color filter array,” *Signal, Image and Video Processing*, vol. 14, no. 1, pp. 143–150, 2020.
- [61] E. Kurniawan, Y. Park, and S. Lee, “Noise-resistant demosaicking with deep image prior network and random RGBW color filter array,” *Sensors*, vol. 22, no. 5, p. 1767, 2022.
- [62] D. Ulyanov, A. Vedaldi, and V. Lempitsky, “Deep image prior,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9446–9454.
- [63] P. Amba, D. Alleysson, and M. Mermillod, “Demosaicing using dual layer feedforward neural network,” in *Proceedings of the Color and Imaging Conference*, vol. 2018, no. 1. Society for Imaging Science and Technology, 2018, pp. 211–218.
- [64] I. Kim, Y. Seo, D. Lim, J. Lee, W. Choi, and S. Song, “DePhaseNet: A deep convolutional network using phase differentiated layers and frequency based custom loss for RGBW image sensor demosaicing,” *Electronic Imaging*, vol. 34, pp. 1–5, 2022.
- [65] Y. Seo, I. Kim, J. Lee, W. Choi, and S. Song, “On quantization of convolutional neural networks for image restoration,” *Electronic Imaging*, vol. 34, pp. 1–5, 2022.
- [66] K. Feng, Y. Zhao, J. C.-W. Chan, S. G. Kong, X. Zhang, and B. Wang, “Mosaic convolution-attention network for demosaicing multispectral filter array images,” *IEEE Transactions on Computational Imaging*, vol. 7, pp. 864–878, 2021.
- [67] S. Liu, Y. Zhang, J. Chen, K. P. Lim, and S. Rahardja, “A deep joint network for multispectral demosaicking based on pseudo-panchromatic images,” *IEEE Journal of Selected Topics in Signal Processing*, vol. 16, no. 4, pp. 622–635, 2022.
- [68] J. Jiang, D. Liu, J. Gu, and S. Süsstrunk, “What is the space of spectral sensitivity functions for digital color cameras?” in *Proceedings of the IEEE Workshop on Applications of Computer Vision*. IEEE, 2013, pp. 168–179.
- [69] M. Parmar and S. J. Reeves, “Selection of optimal spectral sensitivity functions for color filter arrays,” *IEEE Transactions on Image Processing*, vol. 19, no. 12, pp. 3190–3203, 2010.
- [70] Y. Fu, T. Zhang, Y. Zheng, D. Zhang, and H. Huang, “Joint camera spectral response selection and hyperspectral image recovery,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 1, pp. 256–272, 2020.
- [71] K. He, J. Sun, and X. Tang, “Guided image filtering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 6, pp. 1397–1409, 2012.
- [72] L. Xu, S. Zheng, and J. Jia, “Unnatural L0 sparse representation for natural image deblurring,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 1107–1114.
- [73] B. Aiazzi, S. Baronti, and M. Selva, “Improving component substitution pansharpening through multivariate regression of MS + Pan data,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 45, no. 10, pp. 3230–3239, 2007.
- [74] G. Vivone, L. Alparone, J. Chanussot, M. Dalla Mura, A. Garzelli, G. A. Licciardi, R. Restaino, and L. Wald, “A critical comparison among pansharpening algorithms,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 5, pp. 2565–2586, 2014.
- [75] I. Pekkuksen and Y. Altunbasak, “Gradient based threshold free color filter array interpolation,” in *Proceedings of the IEEE International Conference on Image Processing*. IEEE, 2010, pp. 137–140.
- [76] F. Yasuma, T. Mitsunaga, D. Iso, and S. K. Nayar, “Generalized assorted pixel camera: postcapture control of resolution, dynamic range, and spectrum,” *IEEE Transactions on Image Processing*, vol. 19, no. 9, pp. 2241–2253, 2010.
- [77] A. Clouet, J. Vaillant, and D. Alleysson, “The geometry of noise in color and spectral image sensors,” *Sensors*, vol. 20, no. 16, p. 4487, 2020.
- [78] X. Zhang and B. A. Wandell, “A spatial extension of CIELAB for digital color-image reproduction,” *Journal of the Society for Information Display*, vol. 5, no. 1, pp. 61–63, 1997.



Jia Li received his Ph.D. degree in computer science from Beijing Jiaotong University in 2017. He was a Postdoctoral Researcher at Peking University from 2018 to 2020. He is currently a Lecturer in the School of Artificial Intelligence, Beijing Normal University. His research interests include machine learning, computer vision, and image processing.



Chenyan Bai received the B.E. and M.S. degrees from Hebei University, Baoding, China, in 2008 and 2011, respectively, and the Ph.D. degree from Beijing Jiaotong University, Beijing, China, in 2016, all in computer science. Currently, she is a Lecturer at the College of Information Engineering, Capital Normal University, Beijing, China. Her research interests include image processing and machine learning.



Hua Huang (Senior Member, IEEE) received his B.S., M.S., and Ph.D. degrees from Xi'an Jiaotong University, Xian, China, in 1996, 2001, and 2006, respectively. He is currently a Professor in School of Artificial Intelligence, Beijing Normal University. His current research interests include image and video processing, computer graphics, and pattern recognition.

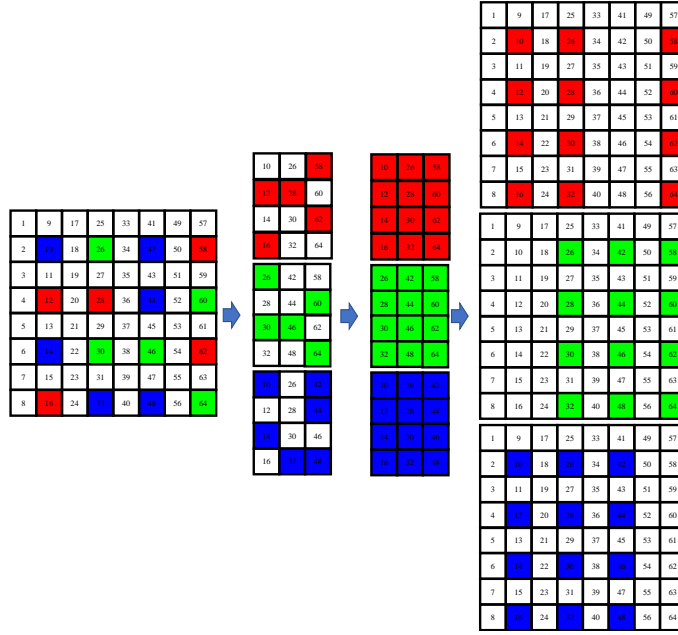


Fig. 10. The process of recovering RGB channels for the Random CFA [22].

APPENDIX A THE DETAILS FOR RGB-CHANNEL RECONSTRUCTION

To explain how our RGB-channel reconstruction can be adapted to any CFA pattern, we use Fig. 10 as an example. It is the Random CFA [22], which is more general. Our RGB-channel reconstruction consists of three steps. First, with the requirement of preserving the relative position of pixels, we select the smallest repeating pattern for each of RGB channels. Then we have \mathbf{C} and the corresponding mask \mathbf{M} . Second, we solve the image decomposition model in (5) to recover all the values in \mathbf{L} and \mathbf{E} . Third, we map all the pixels in \mathbf{L} and \mathbf{E} to the original scales $\tilde{\mathbf{L}}$ and $\tilde{\mathbf{E}}$, where the missing values are zeros. Then we perform the weighted interpolation as:

$$\hat{\mathbf{L}} = f_g(\tilde{\mathbf{L}}) \odot f_g(\tilde{\mathbf{N}}),$$

where f_g is a Gaussian smoothing filter, and $\tilde{\mathbf{N}}$ is the mask to identify available pixels in $\tilde{\mathbf{L}}$. Each pixel is estimated as the weighted average of its neighboring pixels, with the weight being inversely proportional to the distance from the point of interest. Dividing by $f_g(\tilde{\mathbf{N}})$ ensures that the sum of interpolation coefficients equals 1. The window size k of f_g is determined by the maximum distance between unknown values in $\tilde{\mathbf{L}}$. We empirically set the standard deviation of f_g to be $k/6$. We may leave the known values in $\tilde{\mathbf{L}}$ unchanged. $\tilde{\mathbf{E}}$ can be computed similarly. Typically, interpolation-friendly RGBW CFAs are W-dominant. Therefore, we consistently carry out our RGB-channel reconstruction with downsampled RGB channels. Consequently, we apply bilinear interpolation for regular sampling grids and weighted interpolation for irregular sampling grids.

We use the smallest repeating pattern to form \mathbf{C} for the following reasons. First, the maximum distances between known RGB values are relatively small, making it easy to select an appropriate size for the Gaussian smoothing filter f_g in (5). Second, using bilinear or weighted interpolation requires less computational power compared to solving the image decomposition model for larger images in (5). Therefore, employing the smallest repeating pattern can truly accelerate the calculation. Thus, we may conclude that the smallest repeating pattern is the only choice. Our empirical experiments show that other processing details have only minor effects on the results. We test the Kodak CFA [16] with bilinear and weighted interpolation and find that the influence can be disregarded.

APPENDIX B OUR DEMOSAICKED IMAGES OF REAL RAW IMAGES

We provide the scaled our demosaicked images of real raw images, so that the readers can easily evaluate our demosaicking method. The resolution of raw images by the Kodak CFA [16] is 6944×9280 pixels, while the resolution of raw images by the Honda CFA [21] is 1440×2560 pixels. All the scenes are shot both indoors and outdoors using automatic exposure settings. There is no further processing, such as denoising. Note that the raw images taken with the Kodak CFA are captured using a mobile phone camera, while those taken with the Honda CFA are captured using a surveillance camera. As a result, the demosaicked images from the Kodak CFA exhibit higher levels of image noise, while the demosaicked images from the Honda CFA show lower levels of image noise.



Fig. 11. Results of the Kodak CFA [16].



Fig. 12. More results of the Kodak CFA [16].



Fig. 13. Two more results of the Kodak CFA [16].

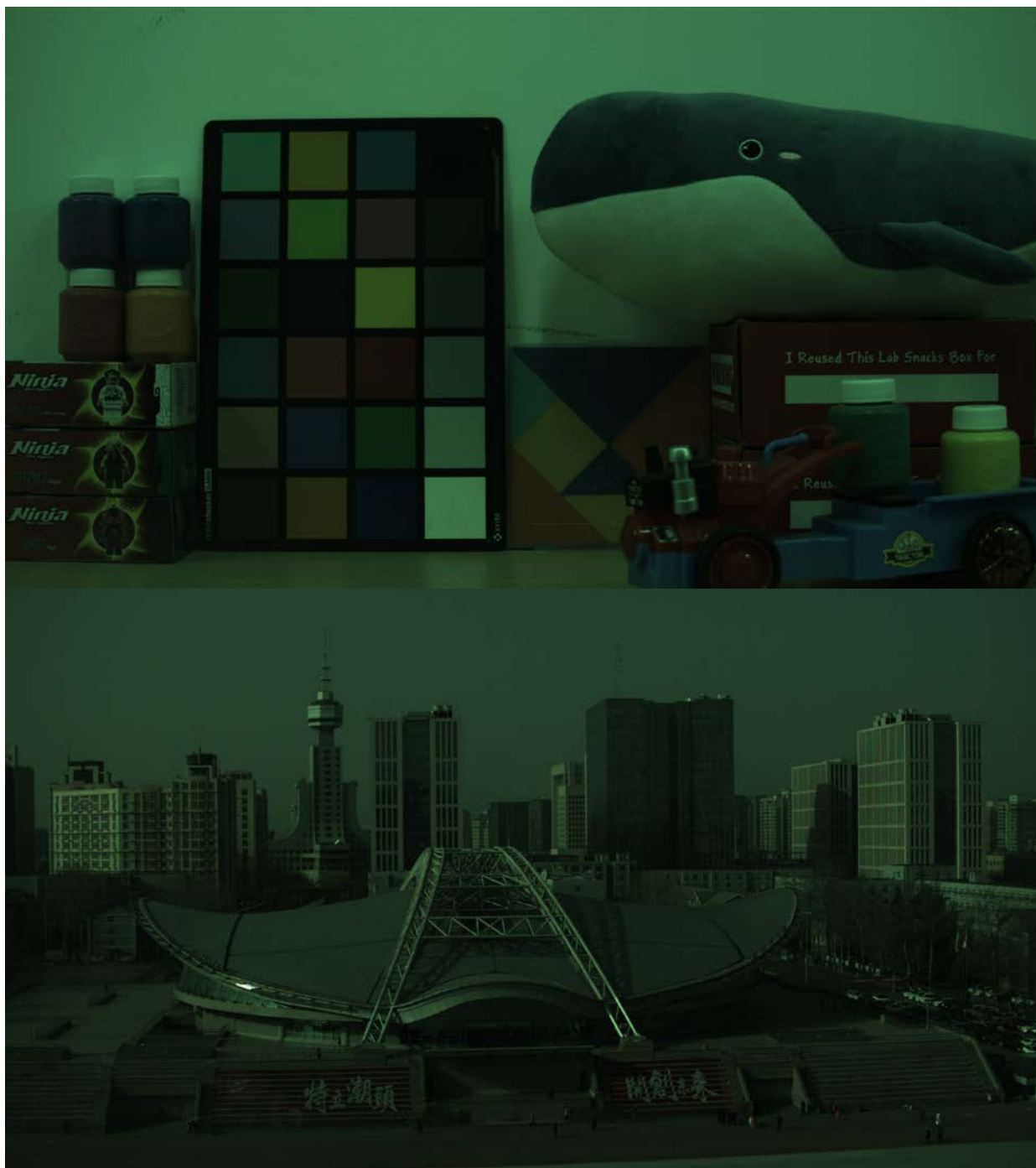


Fig. 14. Results of the Honda CFA [21].

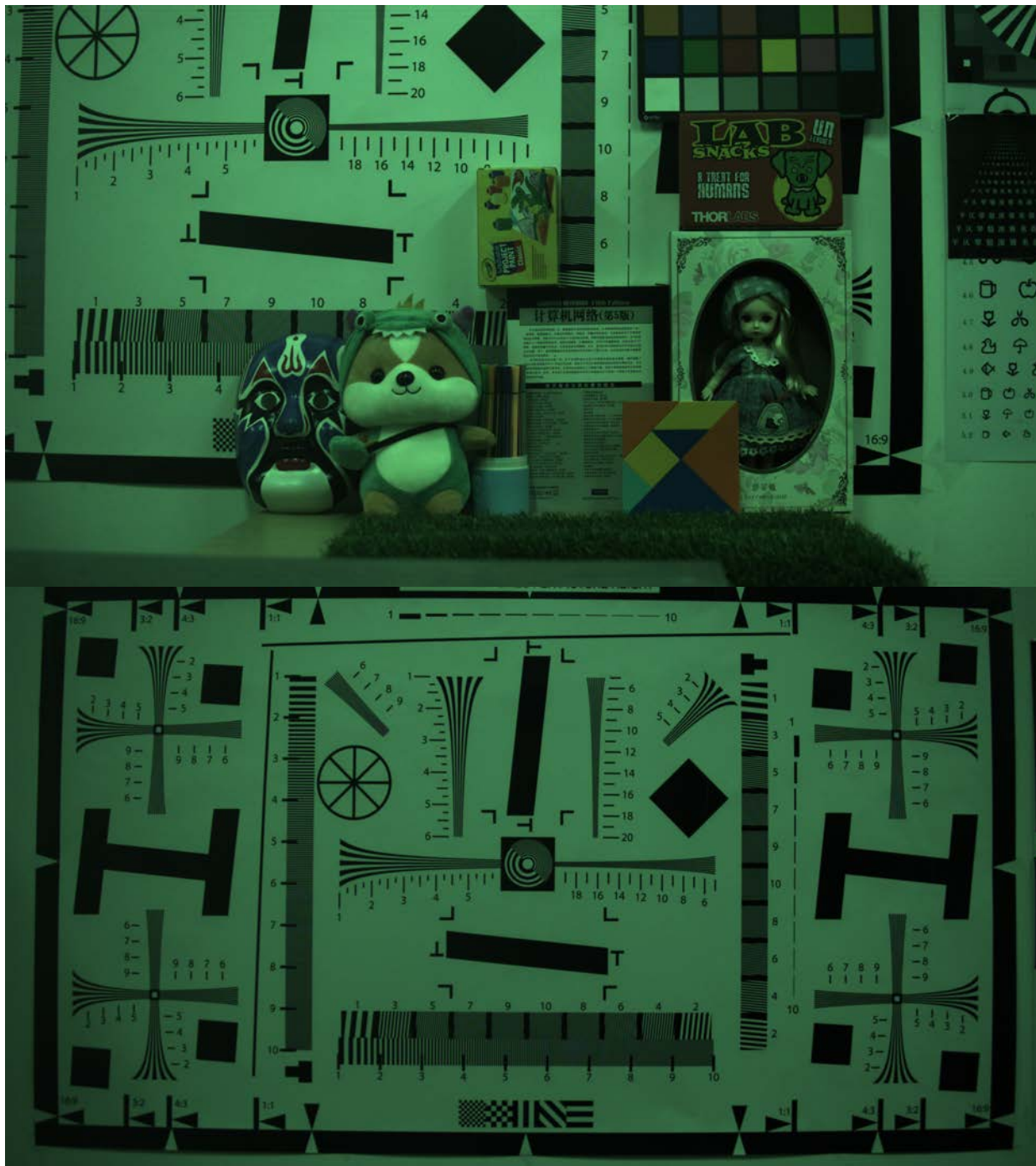


Fig. 15. More results of the Honda CFA [21].

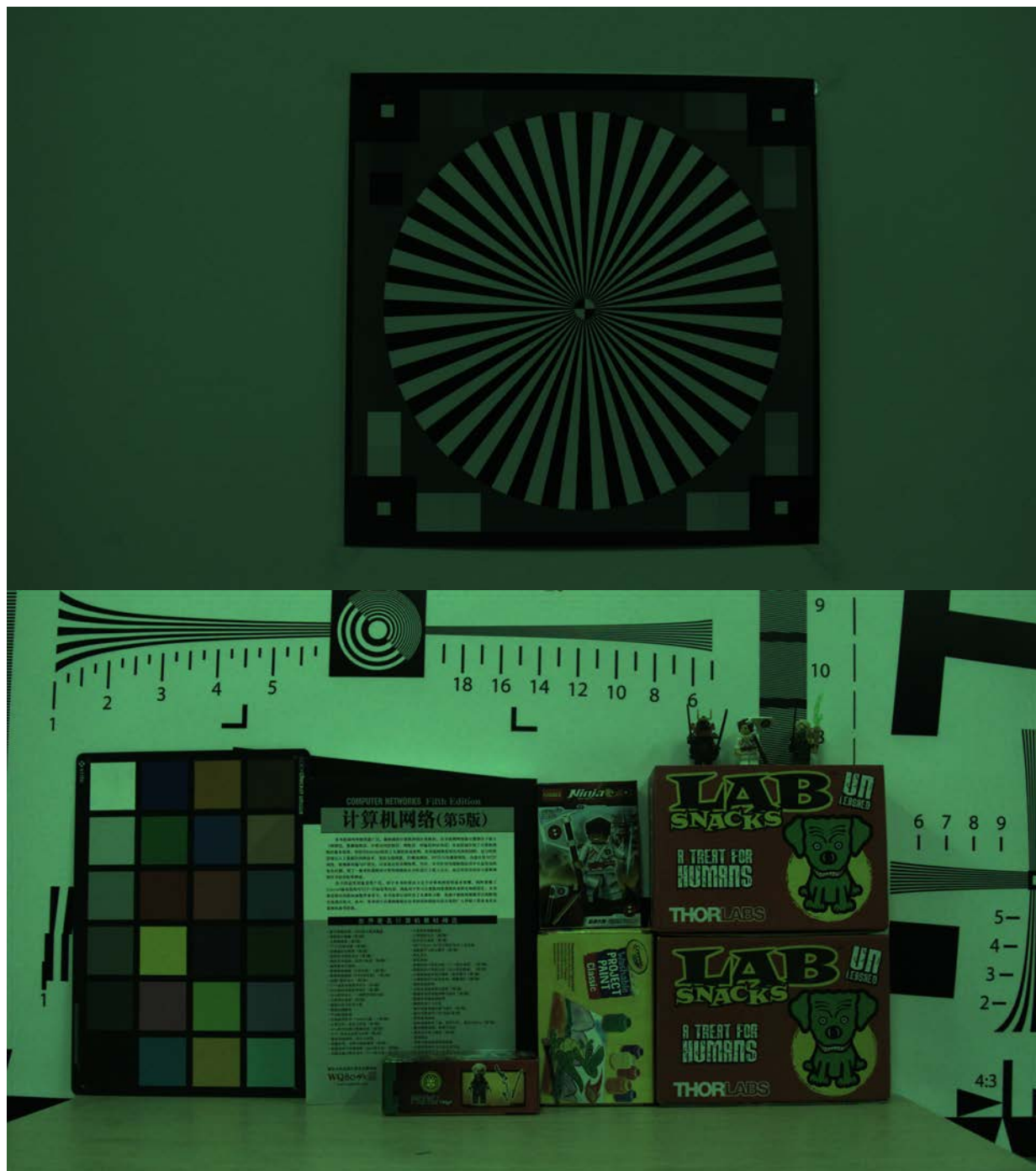


Fig. 16. Two more results of the Honda CFA [21].