A Learning-based Approach to Image Demosaicking with Spatial Autocorrelation Analysis

Min-Kook Choi, Chan Joo, Hyun-Gyu Lee, Sang-Chul Lee
Department of Computer and Information Engineering, Inha University, Korea
mkchoi@inha.edu, challyju@gmail.com, hg_lee@inha.edu, sclee@inha.ac.kr

Abstract

We introduce a two stage image demosaicking method for Bayer color filter array (CFA) images. Pixel interpolation using a Bayesian and/or SVM classifier is followed by renegotiation of the interpolated image with an auto-correlation function (ACF), which is applied to the distribution of edge strengths at each pixel of the interpolated image. This second stage can also be used to post-process images produced by other demosaicking methods. Experimental results obtained with the Kodak PhotoCD benchmark show that our method shows enhanced edge and texture details and when compared with three other methods

1. Introduction

Almost all digital cameras now use a color filter array (CFA) so that each element of the image sensor acquires the intensity of a single primary color. The red, green and blue panes of the CFA are commonly arranged in a Bayer pattern. Demosaicking is then required to construct an image with full-color pixels from the Bayer data. In spite of long-standing efforts to obtain the better result for the demosaicking, this problem is still challenging due to large missing information of pixels on three color panes.

A large number of methods have been proposed to obtain the color components at each pixel from the Bayer data: they have been surveyed by Menon and Calvagno [1] who categorized algorithms as heuristic methods, direction interpolation, frequency domain approaches, wavelet-based methods, reconstruction approaches, and joint demosaicking. They also compared results from 12 algorithms in terms of PSNR, using image from the Kodak PhotoCD dataset [2].

Pekkucuksen and Altunbasak [3] put forward a gradient-based CFA interpolation algorithm, using adaptive filtering to interpolate the missing color data with subsequent post processing. Wang et al. [4] developed an adaptive demosaicking algorithm that finds both non-local similarity and local correlation (NLS-LC) in a CFA image. A patch-based correlation measurement is then used to choose the most similar color distribution from elsewhere in the image to replace the missing color data. Hu et al. [5] proposed a graph-based regularization framework for demosaicking in which an image is considered to smooth signal, and a weighted graph is constructed which replacements the similarity between adjacent pairs of pixels. Learning based approaches have also been used to interpolate missing pixel data. For example, He et al. [6] proposed a self-learning method using that uses support vector regression with an image pyramid. This method avoids the need for prior knowledge, unlike typical learning techniques.

More recently, Duran and Buades [7] tackled state-of-art demosaicking algorithms, which could raise interpolation artifacts, considering inter-channel correlation locally selecting the best interpolation direction. To reduce the interpolation artifacts, they introduced an algorithm using nonlocal image self-similarity when local neighbor has ambiguous local geometry. Monno et al. [8] used

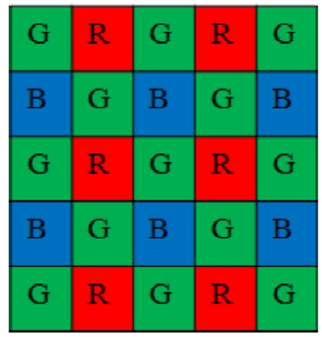

Figure 1. An example of the Bayer pattern

an iterative framework exploiting residual interpolation (RI) which performs the interpolation in a residual domain. They adaptively applied the combined RI algorithms at each pixel based on adjusting iteration number. Learning based post-processing for demosaicked images is developed by Wu et al. [9] using adjusted anchored neighborhood regression. This post-processing method embedded prior information from the training images and the outputs of the demosaicking method of choice into learned linear regressors. They effectively reduced the color artifacts from demosaicked images at test using IMAX and Kodak datasets.

Several researchers who have implemented image demosaicking algorithms have released the source code [10, 11, 12, 13, 14, 15] to techniques to be compared. These algorithms have been based on many different techniques including the use of edge and color difference information [10], non-local sparse models [11], alternating projection [12], contour stencils [13], adaptive thresholding [14], and de-multiplexing using least-squares fitting [15].

In this paper, we propose a two-stage algorithm for image demosaicking which combines machine learning technique with the use of an auto-correlation function (ACF) from Bayer pattern (shown in Fig. 1). In the first stage, features are extracted from the image which determines spatial directions for interpolating missing color data, and then two classifiers are calculated: one using Bayesian theorem and the other a support vector machine (SVM), which is trained using on images of real scenes. Interpolation is performed using a method similar to that of Lu [16]. The second stage involves renegotiation the color at each pixel by analyzing the edge strength in each color channel, by means of [17].

The remainder of this paper is arranged as follows. In Section 2, we present our Bayesian and SVM classifier and their training, and in Section 3 we show how ACFs are used to renegotiate pixel value in the interpolated image. Experiments obtained with the Kodak PhotoCD benchmark dataset are presented in Section 4. Conclusions and suggestions for future work are given in Section 5.

2. Learning based Image Demosaicking

To interpolate the missing color data at each pixel of an input image determined using a Bayer pattern $I_B \in \mathbb{N}^{m \times n}$, we develop learning-based spatial interpolation methods with Bayesian or SVM frameworks. Figure 1 shows an example of the Bayer pattern used in color filter array. Given a sample Bayer pattern I_B, our classifier determines pixel colors by horizontal, vertical and diagonal interpolation, using a feature vector $\mathbf{x} = [x_1, x_2, x_3, x_4]^T$, so that:

$$
\begin{aligned}
\mathbf{x}_{g,b} &= \left[g_{j+1,i} - g_{j-1,i}, 2b_{j,i} - b_{j+2,i} - b_{j-2,i}, g_{j,i+1} - g_{j,i-1}, 2b_{j,i} - b_{j,i+2} - b_{j,i-2}\right]^T, \\
\mathbf{x}_{g,r} &= \left[g_{j+1,i} - g_{j-1,i}, 2r_{j,i} - r_{j+2,i} - r_{j-2,i}, g_{j,i+1} - g_{j,i-1}, 2r_{j,i} - r_{j,i+2} - r_{j,i-2}\right]^T, \\
\mathbf{x}_{r,b} &= \left[r_{j+1,i+1} - r_{j-1,i-1}, 2b_{j,i} - b_{j+2,i+2} - b_{j-2,i-2}, r_{j+1,i-1} - r_{j-1,i+1}, 2b_{j,i} - b_{j-2,i+2} - b_{j+2,i-2}\right]^T, \\
\mathbf{x}_{b,r} &= \left[b_{j+1,i+1} - b_{j-1,i-1}, 2r_{j,i} - r_{j+2,i+2} - r_{j-2,i-2}, b_{j+1,i-1} - b_{j-1,i+1}, 2b_{j,i} - r_{j-2,i+2} - r_{j+2,i-2}\right]^T, \\
\mathbf{x}_{r,g} &= \left[r_{j+1,i} - r_{j-1,i}, 2g_{j,i} - g_{j+2,i} - g_{j-2,i}, r_{j,i+1} - r_{j,i-1}, 2g_{j,i} - g_{j,i+2} - g_{j,i-2}\right]^T, \\
\mathbf{x}_{b,g} &= \left[b_{j+1,i} - b_{j-1,i}, 2g_{j,i} - g_{j+2,i} - g_{j-2,i}, b_{j,i+1} - b_{j,i-1}, 2g_{j,i} - g_{j,i+2} - g_{j,i-2}\right]^T,
\end{aligned} \tag{1}
$$

where the first subscript is the intensity of the interpolating color pixel, and the second subscript is the intensity of the input color pixel. r, g and b are the red, green and blue intensities of a pixel at (j, i). Six types of feature vectors, based on the Bayer pattern, are extracted from a set of training images $\mathcal{I} \ni \{I_{t,1}, I_{t,2}, \ldots, I_{t,n}\}$ to train both the Bayesian and SVM classifiers. We use a training of n images, which are not taken from the Kodak PhotoCD dataset. We rejected feature vectors with any zero elements to exclude situations that features are extracted from flat area.

We use an interpolation formula similar to that used by Lu [16], except that any does not require weights to be applied to the input color values:

$$
\begin{aligned}
g(b)^H_{j,i} &= \frac{g_{j+1,i} + g_{j-1,i}}{2} + \frac{2b_{j,i} - b_{j+2,i} - b_{j-2,i}}{4}, \\
g(b)^V_{j,i} &= \frac{g_{j,i+1} + g_{j,i-1}}{2} + \frac{2b_{j,i} - b_{j,i+2} - b_{j,i-2}}{4}, \\
r(b)^{45^\circ}_{j,i} &= \frac{r_{j+1,i+1} + r_{j-1,i-1}}{2} + \frac{2b_{j,i} - b_{j+2,i+2} - b_{j-2,i-2}}{4}, \\
r(b)^{135^\circ}_{j,i} &= \frac{r_{j+1,i-1} + r_{j-1,i+1}}{2} + \frac{2b_{j,i} - b_{j+2,i-2} - b_{j-2,i+2}}{4},
\end{aligned} \tag{2}
$$

where $g(b)^H_{j,i}$ interpolates the green channel intensity in the horizontal direction at the pixel at (j, i), which is covered by a blue panes of the Bayer filters. The other interpolates $g(r)^H_{j,i}$, $g(r)^V_{j,i}$, $b(r)^{45^\circ}_{j,i}$, $b(r)^{135^\circ}_{j,i}$, $r(g)^H_{j,i}$, $r(g)^V_{j,i}$, $b(g)^H_{j,i}$, and $b(g)^V_{j,i}$ deals with other situations that occur on the Bayer pattern in a similar way. Figure 2 shows the four different cases for the missing pixel interpolation and grey pixels in 5 by 5 local region are to be interpolated two color pixel values.

If we consider the feature vector $\mathbf{x}$ to be a Gaussian random vector $\mathbf{x} \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, then Bayesian quadratic decision functions are expressed by the following equations:

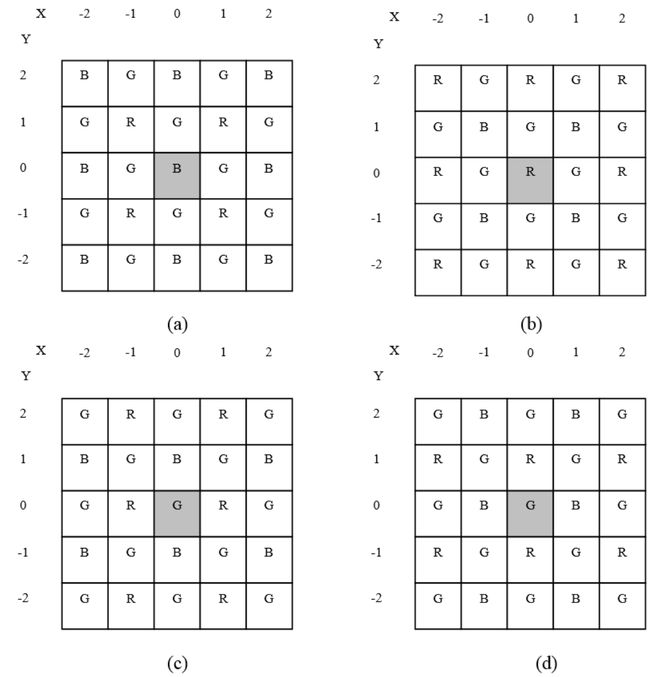

Figure 2. Four different cases of for missing pixel interpolation. Grey pixels indicate the target pixel that would be interpolated in 5 by 5 local region.

$$
\begin{aligned}
&\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_H)^T \boldsymbol{\Sigma}_H^{-1} (\mathbf{x} - \boldsymbol{\mu}_H) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_V)^T \boldsymbol{\Sigma}_V^{-1} (\mathbf{x} - \boldsymbol{\mu}_V) \lessgtr \eta, \\
&\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{45^\circ})^T \Sigma_{45^\circ}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{45^\circ}) - \frac{1}{2}(\mathbf{x} - \boldsymbol{\mu}_{135^\circ})^T \boldsymbol{\Sigma}_{135^\circ}^{-1} (\mathbf{x} - \boldsymbol{\mu}_{135^\circ}) \lessgtr \eta,
\end{aligned} \tag{3}
$$

where $\boldsymbol{\mu} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{x}_s$ is a mean of all N random vectors, N is the total number of feature vectors, $\boldsymbol{\Sigma} = \frac{1}{n-1}\sum_{k=1}^{N} (\mathbf{x}_s - \boldsymbol{\mu})(\mathbf{x}_s - \boldsymbol{\mu})^T$ is their covariance matrix, $\eta = \frac{1}{2}\ln\frac{|\boldsymbol{\Sigma}|P_1^2}{|\boldsymbol{\Sigma}|P_2^2}$ is the decision constant, and P_1 and P_2 are the prior probabilities, that are set to 0.5.

We built a second classifier using the SVM framework from the same feature vector $\mathbf{x}$. This is binary class classifier that determines the direction of interpolation, and uses maximum margin criterion [15]:

$$
\begin{aligned}
&\max_{\alpha} L(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2\lambda}\sum_{i=1}^{N} (\alpha_i y_i)(\alpha_j y_j) K(\mathbf{x}_i, \mathbf{x}_j), \\
&\text{s.t. } \mathbf{y}\boldsymbol{\alpha} = 0, \qquad 0 \le \alpha_i \le \frac{1}{N},
\end{aligned} \tag{4}
$$

where α is the Lagrange multiplier, λ is regularization parameter, $y \ni \{-1, 1\}$ is the class variable for the binary classification.

We can now obtain an interpolated color image $I \in \mathbb{N}^{m \times n \times 3}$ by applying these two classifiers to the interpolation formula at every pixel.

3. Auto-correlation function (ACF) analysis

The second stage of our algorithm uses an auto-correlation function to extract the directional strength of any edge that may be present at a pixel in each color channel. We use edge strength determining renegotiation weights to update the intensity of every pixel in every color channel in the interpolated color image I. We extract a structure tensor A at each pixel (i, j), using a technique similar to an existing method of [17].

$$A_r = \sum_{j=1}^{m} \sum_{i=1}^{n} w(j,i) \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \approx \begin{bmatrix} \lambda_1 & c_1 \\ c_2 & \lambda_2 \end{bmatrix} \quad (5)$$

where r is a color channel, w is a Gaussian weight function, I_x and I_y are one-dimensional derivatives of pixel intensity in a certain color channel at (j, i), and λ_1 and λ_2 constants which express the directional strength of any edge at (j, i). Similar equations can be formulated for the blue and green channels.

We also define formulas for the renegotiation of pixel intensity, which are a development of Lu's interpolation method [16], in which the constants λ_1 and λ_2 are normalized weights. The following formula $f_{j,i,r}$ applies to the red channel at pixel (j, i).

$$\begin{aligned} f_{j,i,r} = \frac{\lambda_1}{\lambda_1+\lambda_2}\Big[&\frac{r_{j+1,i}+r_{j-1,i}}{2} + \frac{\kappa_1}{8}\left(2g_{j,i}-g_{j+2,i}-g_{j-2,i}\right) \\ &+ \frac{\kappa_2}{8}\left(2b_{j,i}-b_{j+2,i}-b_{j-2,i}\right)\Big] \\ &+ \frac{\lambda_2}{\lambda_1+\lambda_2}\Big[\frac{r_{j,i+1}+r_{j,i-1}}{2} \\ &+ \frac{\kappa_1}{8}\left(2g_{j,i}-g_{j,i+2}-g_{j,i-2}\right) \\ &+ \frac{\kappa_2}{8}\left(2b_{j,i}-b_{j,i+2}-b_{j,i-2}\right)\Big] \quad (6) \end{aligned}$$

where κ_1 and κ_2 are regularization parameters which respectively adjust the effect of green and blue pixel intensities. The formulas for $f_{j,i,g}$ and $f_{j,i,b}$ are similar. This ACF analysis can be used in conjunction with other demosaicking methods.

4. Experimental results

We performed experiments with the Kodak PhotoCD benchmark dataset, which consists of 24 images of real scenes, such as these shown in Figure 3. We trained the Bayesian and SVM classifiers on 15 images of real scenes, such as these in Figure 4, which were not taken from the Kodak dataset. After training the SVM classifier using the training dataset and the six types of feature vectors, we improve SVM parameters by grid optimization [18]. The regularization parameter λ and kernel parameter a gamma g for RBF kernel function are respectively set to 100 and 0.1. To test second, renegotiation step, we optimized the parameters which control the relative contribution of different color channels in the interpolation such as both κ_1 and κ_2 to 0.4.

We compared our algorithm with three state-of-the algorithms identified [1]. These methods are iterative demosaicking using weighted-edge and color-difference (IDWC) [10], alternating projection (AP) [12], and nonlocal adaptive thresholding (NAT) [14]. Figure 5 and 6, showing the original image, two images processed by our algorithm, and competing algorithms, allows qualitative assessment of the results: Highly textured regions, e.g., wooden surface, waves on water surface, repetitive window patterns, grass, and feathers, show noticeable enhancement of texture details where the gray-level values have more variability without distorting the intensity patterns seriously.

5. Conclusion

We proposed a two-stage method for demosaicking: the first stage uses a classifier trained on images, and the second stage involves post-processing by renegotiate pixel in intensities all channel of the interpolated image. We have shown that our learning-based demosaicking followed by channel intensity renegotiation using ACF achieved encouraging results in terms of edge and texture enhancement.

In the future, we plan to examine improved interpolation formulas which reflect the intensity distribution of each color channel as a way that corresponds more precisely to the layout of the Bayer filters. Further, since our method is more focused in enhancing textures and edges in images, our method can be combined with a segmentation method in order to locally enhance highly textured regions.

Figure 3. Examples of images from the Kodak dataset.

Figure 4. Examples of training images.

Acknowledgement

This work was supported by (1) Next–Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT

(a)

(b)

(c)

(d)

(e)

(f)

Figure 5. Portion of five different Kodak PhotoCD benchmark images, no. 2, 3, 5, 7, and 24: (a) original image, (b) iterative demosaicking using weighted-edge and color-difference (IDWC), (c) alternating projection (AP), (d) nonlocal adaptive thresholding (NAT), (e) proposed method with Bayesian classifier, and (f) proposed method with SVM classifier. Proposed methods provide better detail texture or pattern of rigid objects and structures in demosaicked images.

& Future Planning (2012M3C4A7032781), (2) the ICT R&D program of MSIP/IITP. [2014(10047078), 3D reconstruction technology development for scene of car accident using multi view black box image].

References

[1] D. Menon, and G. Calvagno, "Color Image Demosaicking: An Overview", Signal Processing: Image Communication, 26, pp.518-533, 2011.

[2] URL: http://www.cipr.rpi.edu/resource/stills/kodak.html.

[3] I. Pekkucuksen, and Y. Altunbasak, "Gradient based Threshold Free Color Filter Array Interpolation", In Proc. of International Conference of Image Processing, 2010.

[4] G. Wang, X. Zhu, and Z. Gan, "Image Demosaicing by Non-local Similarity and Local Correlation", In Proc. of International Conference on Signal Processing, 2012.

[5] C. Hu, L. Cheng, and Y. M. Lu, "Graph-based Regularization for Color Image Demosaicking", In Proc. of International Conference of Image Processing, 2012.

(a)

(b)

(c)

(d)

(e)

(f)

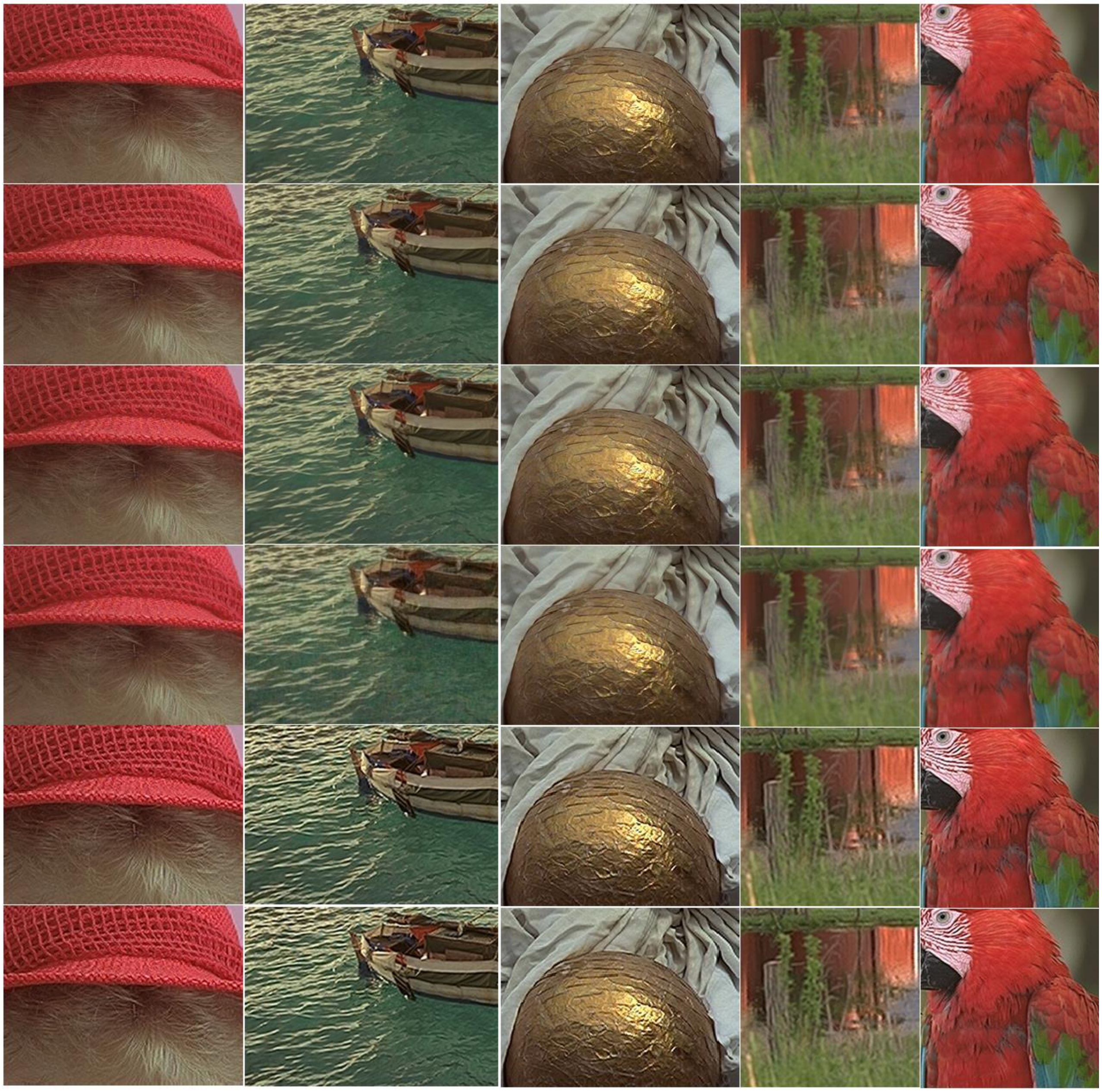

Figure 6. Portion of five different Kodak PhotoCD benchmark images, no. 4, 6, 17, 22, and 23: (a) original image, (b) iterative demosaicking using weighted-edge and color-difference (IDWC), (c) alternating projection (AP), (d) nonlocal adaptive thresholding (NAT), (e) proposed method with Bayesian classifier, and (f) proposed method with SVM classifier. Proposed methods also provide better details of high frequency information such as waves, complex texture, grass leaf, hair, and feather in demosaicked images.

[6] F. –L. He, Y. –C. F. Wang, and K. –L. Hua, "A Self-Learning Approach to Color Demosaicking via Support Vector Regression", In Proc. of International Conference of Image Processing, 2012.

[7] J. Duran and A. Buades, "Self-Similarity and Spectral Correlation Adaptive Algorithm for Color Demosaicking", IEEE Transactions on Image Processing, 23(9), pp. 4031-4040, 2014.

[8] Y. Monno, D. Diku, M. Tanaka, and M. Okutomi, "Adaptive Residual Interpolation for Color Image Demosaicking", IEEE International Conference on Image Processing, 2015.

[9] J. Wu, R. Timofte, and L. V. Gool, "Efficient Regression Priors for Post-processing Demosaiced Images", IEEE International Conference on Image Processing, 2015.

[10] C. –Y. Su, "Highly Effective Iterative Demosaicing Using Weighted-Edge and Color-Difference Interpolations", IEEE Trans-actions on Consumer Electronics, 52(2), pp. 639-645, 2006.

[11] J. Mairal, F. Bach, J. Ponce, G. Sapiro, and A. Zisserman, "Non-local Sparse Models for Image Restoration", In Proc. of International Conference of Computer Vision, 2009.

[12] Y. M. Lu, M. Karzand, and M. Vetterli, "Demosaicking by Alternating Projections: Theory and Fast One-Step Implementation", IEEE Transactions on Image Processing, 19(8), pp. 2085-2098, 2010.

[13] P. Getreuer, "Color Demosaicing with Contour Stencils", In Proc. of International Conference on Digital Signal Processing, 2011.

[14] L. Zhang, X. Wu, A. Buades, and X. Li, "Color Demosaicking by Local Directional Interpolation and Nonlocal Adaptive Thresholding", Journal of Electronic Imaging, 20(2), 2011.

[15] G. Jeon, and E. Dubois, "Demosaicking of Noisy Bayer-Sampled Color Images with Least-Squares Luma-Chroma De-multiplexing and

[16] Noise Level Estimation", IEEE Transactions on Image Processing, 22(1), pp.146-156, 2013.

[17] N. Lu, and Z. Lu "Method and System for Interpolating Missing Picture Elements in a Single Color Component Array Obtained from a Single Color Sensor", U.S. Patent, No. 5805217, 1998.

[18] C. Harris, and M. Stephens, "A Combined Corner and Edge Detector", In Proc. of Alvey Vision Conference, 1988.

[19] C-., C. Chang, and C-., J. Lin, "LIBSVM: A library for support vector machine', ACM Transaction on IST, 2(27) pp.1-27, 2011, S/W available at http://www.scie.ntu.edu.tw/cjlin/ligsvm

Author Biography

Min-Kook Choi is a Ph.D. candidate in the Department of Computer and Information Engineering at Inha University, Incheon, Korea. Choi's research interest is in machine learning for computer vision and video processing.

Chan Joo received him Bachelor's degree in Computer Information Engineering from the University of Inha. Since then he has worked in Image processing laboratory at Inha University, Incheon, Korea. His work has focused on the low level vision and medical image.

Hyun-Gyu Lee is a Ph.D. candidate in the Department of Computer and Information Engineering at Inha University, Incheon, Korea. Lee's research interest is in medical image analysis, computer vision and image processing.

Sang-Chul Lee is an associative professor in the Department of Computer and Information Engineering at Inha University, Incheon, Korea. Prof. Lee's scientific interest is in computer vision, image processing and multimedia information retrieval.