LOSSLESS/NEAR-LOSSLESS COLOR IMAGE CODING BY INVERSE DEMOSAICING

Ryo Kuroiwa[1], *Ryo Matsuoka*[1], *Seisuke Kyochi*[1*], *Keiichiro Shirai*[2] *and Masahiro Okuda*[1]

1: Dept. of Information and Media Engineering, The University of Kitakyushu
1-1 Hibikino, Wakamatsu, Kitakyushu, Fukuoka 808-0135, Japan
2: Dept. of Computer Science and Engineering, Shinshu University
4-17-1 Wakasato, Nagano 380-8553, Japan
Email: s1mca012@eng.kitakyu-u.ac.jp, {s-kyochi, okuda-m}@kitakyu-u.ac.jp, keiichi@shinshu-u.ac.jp

ABSTRACT

In this paper, we introduce a novel framework for lossless/near-lossless (LS/NLS) color image coding assisted by an inverse demosaicing. Conventional frameworks are typically based on prediction (and quantization for NLS coding) followed by entropy coding, such as the JPEG-LS for bit rate saving. The approach of this work is totally different from the conventional ones. Basically, color images are created by demosaicing Bayer-pattern color filter array (CFA) whose operator can be expressed as square matrices. By using the (pseudo) inverse matrix of a joint demosaicing and color-to-gray conversion, the proposed decoder can recover the color image from its corresponding gray image data which is losslessly transmitted by the proposed encoder. Thus, LS/NLS color image reconstruction can be achieved while saving a bit rate significantly. In addition, using the same framework of color image coding, LS/NLS CFA coding can be realized by a comparable bit rate with JPEG-LS.

Index Terms— lossless/near-lossless color image coding, demosaicing, inverse problem

1. INTRODUCTION

Lossless (LS) color image coding has been important for high quality image transmission, medical image compression, and so on. As well as LS image coding, near-lossless (NLS) one which allows low reconstruction error (e.g., ± 1 or ± 2) at each pixel is also important. A typical LS/NLS image coder is JPEG-LS [1], where spatial prediction (and residual quantization for NLS coding) followed by entropy coding is performed. Until now, some related methods of JPEG-LS based LS/NLS color image coding have been proposed [2–4].

This paper proposes a novel framework for LS/NLS color image coding that utilizes an image demosaicing algorithm for Bayer-pattern color filter array (CFA) [5, 6]. Needless to say, most of recent digital cameras firstly capture natural images via a CCD/CMOS sensor with a color filter, and then store it in the CFA format. Then, full color image is produced by demosaicing operation. Mathematically speaking, this procedure can be characterized as a corresponding matrix multiplication to CFA. This paper clarifies that, if the demosaicing algorithm is known, the original color image can be recovered from its corresponding gray image by its inverse demosaicing procedure. Specifically, we formulate demosaicing and color-to-gray operations as matrix multiplications and utilize its inverse (or pseudo-inverse) matrix to recover the CFA raw data from the transmitted lossless gray image. Then performing demosaicing followed by an offset compensation, we can recover the color image at decoder side. This experimental results show the proposed image coding can save a bit rate significantly while hiding color imformation. In addition, the original CFA raw data can be transmitted by using the same framework of color image transmission with comparable bit rate with the JPEG-LS.

The rest of this paper is organized as follows. Demosaicing are briefly reviewed in Sec. 2. Then, Sec. 3 shows the proposed LS/NLS image coding algorithm. Applications of this proposed method and numerical evaluation with the LS/NLS mode of JPEG-LS are shown in Sec. 4. Finally, this paper is concluded in Sec. 5.

Notations: Matrices are indicated in upper-case bold face letters, while vectors are indicated in lower-case ones. $\mathbb{R}^N$ and $\mathbb{R}^{M\times N}$ denotes the sets of M-dimensional real-valued vectors and $M \times N$ real-valued matrices, respectively. Each entry of $\boldsymbol{X} \in \mathbb{R}^{M\times N}$ is indicated by $\boldsymbol{X}(m, n)$. Especially, for images, $\boldsymbol{X} \in \mathbb{R}^{M\times N}$ denotes the M [rows] $\times$ N [column] image.

2. PRELIMINARIES

2.1. Demosaicing

As shown in Fig. 1, according to its pixel positions of the CFA $\boldsymbol{I}(m, n)$ ($\boldsymbol{I} \in \mathbb{R}^{M\times N}$), we denote four samples in the (2×2)-block as $R^{(1)}_{m,n}$, $R^{(2)}_{m,n}$, $R^{(3)}_{m,n}$ and $R^{(4)}_{m,n}$ for the red channel, and the same notation is used for the G and B channels. The four pixels of G channel, for example, $G^{(1)}_{m,n}$, $G^{(2)}_{m,n}$, $G^{(3)}_{m,n}$ and $G^{(4)}_{m,n}$ are defined as

$$\begin{aligned} G^{(1)}_{m,n} =& \boldsymbol{I}(m, n),\ G^{(2)}_{m,n} = \boldsymbol{I}(m, n + 1) \\ G^{(3)}_{m,n} =& \boldsymbol{I}(m + 1, n),\ G^{(4)}_{m,n} = \boldsymbol{I}(m + 1, n + 1), \end{aligned} \tag{1}$$

Then, missing samples $G^{(2)}_{m,n}$ and $G^{(3)}_{m,n}$ are interpolated as

$$G^{(2)}_{\cdot,\cdot} = T_{G^{(2)}}(G^{(1)}_{\cdot,\cdot}, G^{(4)}_{\cdot,\cdot}), \quad G^{(3)}_{\cdot,\cdot} = T_{G^{(3)}}(G^{(1)}_{\cdot,\cdot}, G^{(4)}_{\cdot,\cdot}), \tag{2}$$

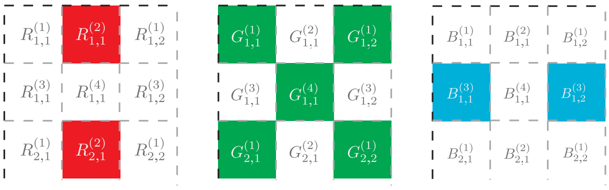
where $T_{G^{(2)}}$ and $T_{G^{(3)}}$ are some demosaicing operators. For instance, considering $T_{G^{(2)}}$, $T_{G^{(3)}}$ as bilinear interpolation operators, an instance of $T_{G^{(2)}}$ would be

$$\begin{aligned} G^{(2)}_{m,n} =& T_{G^{(2)}}(G^{(1)}_{m,n}, G^{(4)}_{m,n}) \\ =& \beta_{41} G^{(1)}_{m,n} + \beta_{42} G^{(1)}_{m,n+1} + \beta_{43} G^{(4)}_{m,n} + \beta_{44} G^{(4)}_{m+1,n}, \end{aligned} \tag{3}$$

where $\beta_{4k} \in \mathbb{R}$ ($k = 1 \ldots 4$) represents weighting factors, and in addition, the estimated values are rounded to integers. $T_{G^{(3)}}$ and those for other color channels can be similarly defined as a weighted

*This work was supported by JSPS KAKENHI Grant Number 24860055.

978-1-4799-2893-4/14/$31.00



(a) R channel (b) G channel (c) B channel

Fig. 1. R, G and B channels obtained from the original CFA $\boldsymbol{I}$. $R^{(1)}_{m,n}$, $R^{(3)}_{m,n}$, $R^{(4)}_{m,n}$, $G^{(2)}_{m,n}$, $G^{(3)}_{m,n}$, $B^{(1)}_{m,n}$, $B^{(2)}_{m,n}$ $B^{(4)}_{m,n}$ are to be interpolated.

combination. Adaptive interpolation algorithms (e.g. [6]) can be used as the demosaicing operators as well.

Here, the demosaicing operators are represented as some matrix form. Let $\boldsymbol{x} \in \mathbb{R}^{MN}$ be a vectorized version of the original CFA $\boldsymbol{I} \in \mathbb{R}^{M\times N}$, i.e. $\boldsymbol{x}(Mn+m) = \boldsymbol{I}(m,n)$. Letting $\boldsymbol{g} \in \mathbb{R}^{MN}$ be the interpolated G channel in (3) can be expressed in a matrix form as $\boldsymbol{g} = \boldsymbol{A}_G\boldsymbol{x}$, where the demosaicing matrix $\boldsymbol{A}_G \in \mathbb{R}^{MN\times MN}$ consists of the weighting factors. In a similar fashion, $\boldsymbol{r}$, $\boldsymbol{b} \in \mathbb{R}^{MN\times MN}$ can be expressed by using the demosaicing matrices $\boldsymbol{A}_R$, $\boldsymbol{A}_B \in \mathbb{R}^{MN\times MN}$ as $\boldsymbol{r} = \boldsymbol{A}_R\boldsymbol{x}$ and $\boldsymbol{b} = \boldsymbol{A}_B\boldsymbol{x}$.

3. PROPOSED ALGORITHM

An overview of the proposed NLS and LS coding algorithm is illustrated in Fig. 2(a)–(d). As explained in the previous section, it is assumed that demosaicing algorithm is known at both the encoder and the decoder.

3.1. Overview

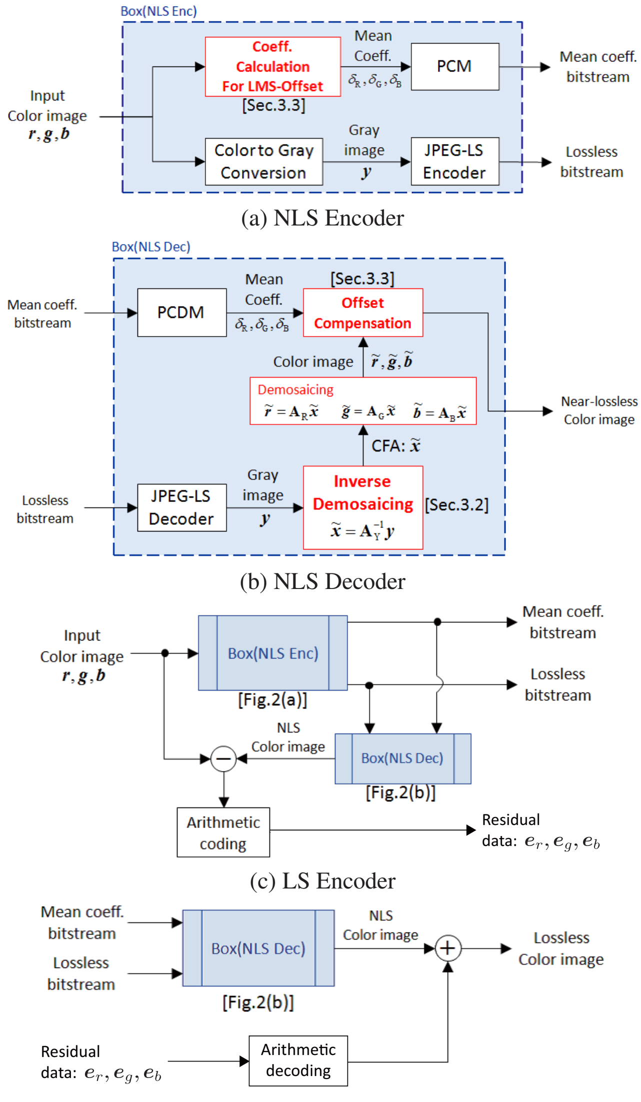
Now, we first present the encoder description. In the case of NLS transmission (shown in Fig. 2(a)), first, a given color image $\boldsymbol{r}$, $\boldsymbol{g}$, $\boldsymbol{b} \in \mathbb{R}^{MN}$ is converted to the gray image $\boldsymbol{y} \in \mathbb{R}^{MN}$. After the conversion, the gray image is encoded by JPEG-LS and transmitted. In parallel, three coefficients for least mean square offsets δ_R, δ_G, $\delta_B \in \mathbb{R}$ explained in Sec. 3.3 are computed from the original color image and transmitted to the decoder. In the LS transmission (see in Fig. 2(c)), additional residual signals $\boldsymbol{e}_r$, $\boldsymbol{e}_g$, $\boldsymbol{e}_b \in \mathbb{R}^{MN}$ are transmitted, which is simply the difference of original and near-losslessly reconstructed color image.

The block diagram of the decoder description is shown in Fig. 2(b), in which the gray image is decoded, and the inverse demosaicing algorithm presented in Sec.3.2 is applied. After the demosaicing and the offset compensation explained in Sec. 3.3, the near-lossless color image is recovered. In addition, the decoding process in the LS transmission (in Fig. 2(d)) is completed by adding the residual data to near-lossless color image.

For transmission of the mean coefficients (δ_R, δ_G, δ_B), the pulse code modulation (PCM) and the pulse code demodulation (PCDM) are used, and arithmetic encoding/decoding is applied to the residual data ($\boldsymbol{e}_r$, $\boldsymbol{e}_g$, $\boldsymbol{e}_b$).

3.2. Inverse demosaicing

Let $\boldsymbol{x} \in \mathbb{R}^{MN}$ be an original CFA (unknown), $\boldsymbol{A}_R$, $\boldsymbol{A}_G$, $\boldsymbol{A}_B \in \mathbb{R}^{MN\times MN}$ be some demosaicing matrices and $\boldsymbol{r}$, $\boldsymbol{g}$ and $\boldsymbol{b}$ ($\in \mathbb{R}^{MN}$) be the R, G, and B channel interpolated by demosaicing. A gray image $\boldsymbol{y} \in \mathbb{R}^{MN\times MN}$ to be transmitted is typically yielded by the



(a) NLS Encoder

(b) NLS Decoder

(c) LS Encoder

(d) LS Decoder

Fig. 2. The diagram of the proposed LS/NLS algorithm (Encoder/Decoder).

weighted sum of $\boldsymbol{r}$, $\boldsymbol{g}$ and $\boldsymbol{b}$ as

$$\boldsymbol{y} = p_R\boldsymbol{r} + p_G\boldsymbol{g} + p_B\boldsymbol{b} = \boldsymbol{A}_Y\boldsymbol{x}, \tag{4}$$

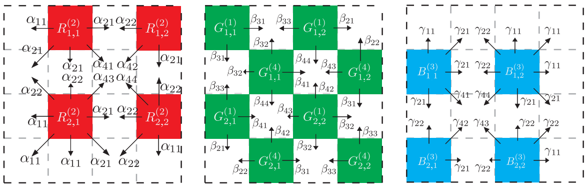
where $\boldsymbol{A}_Y = p_R\boldsymbol{A}_R + p_G\boldsymbol{A}_G + p_B\boldsymbol{A}_B$, p_R, p_G, $p_B \in \mathbb{R}$ are coefficients of the color-to-gray conversion. From the losslessly coded gray image $\boldsymbol{y}$ at the decoder side, the unknown CFA $\tilde{\boldsymbol{x}}$ can be recovered as

$$\tilde{\boldsymbol{x}} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{A}_Y\boldsymbol{x}\|_{L^2}, \tag{5}$$

where $\|\cdot\|_{L^2}$ denotes L^2 norm. The minimizer can be obtained as $\tilde{\boldsymbol{x}} = \boldsymbol{A}_Y^{\dagger}\boldsymbol{y}$, where $\boldsymbol{A}_Y^{\dagger} = (\boldsymbol{A}_Y^T\boldsymbol{A}_Y)^{-1}\boldsymbol{A}_Y^T$ denotes the pseudo inverse of $\boldsymbol{A}_Y$. Finally, the R, G and B channels are reconstructed as $\tilde{\boldsymbol{r}} = \boldsymbol{A}_R\tilde{\boldsymbol{x}}$, $\tilde{\boldsymbol{g}} = \boldsymbol{A}_G\tilde{\boldsymbol{x}}$ and $\tilde{\boldsymbol{b}} = \boldsymbol{A}_B\tilde{\boldsymbol{x}}$.

3.3. Least mean square offset compensation

Ideally, if the matrix $\boldsymbol{A}_Y$ is non-singular, the color channels $\tilde{\boldsymbol{r}}$, $\tilde{\boldsymbol{g}}$ and $\tilde{\boldsymbol{b}}$ can be recovered from the transmitted gray image by inverse



(a) R channel (b) G channel (c) B channel

Fig. 3. Demosaicing operator used in the simulation. For simplicity the CFA is depicted as the 4×4 array.

demosaicing. However, if the matrix $\boldsymbol{A}_Y$ is singular, a reconstructed image suffers from error. In fact, the example of $\boldsymbol{A}_Y$ used in the simulation (Section 4) is not full-rank. In this paper, the least mean square offset compensation is carried out for each channel:

$$\tilde{\delta}_R = \arg\min_{\delta_R} \|\boldsymbol{r} - (\tilde{\boldsymbol{r}} + \delta_R \boldsymbol{1})\|^2_{L^2}, \tag{6}$$

where $\boldsymbol{1}$ is a vector of all ones. Taking the derivative and setting it to 0, the solutions of (6) is obtained by $\tilde{\delta}_R = \mu_{\boldsymbol{r}} - \mu_{\tilde{\boldsymbol{r}}}$, where, μ_f is the mean of f (i.e. $f \in \mathbb{R}^D$, $\mu_f = 1/D \sum_{k=0}^{D-1} f(k)$). Similarly, $\tilde{\delta}_G = \mu_{\boldsymbol{g}} - \mu_{\tilde{\boldsymbol{g}}}$, $\tilde{\delta}_B = \mu_{\boldsymbol{b}} - \mu_{\tilde{\boldsymbol{b}}}$. In the proposed algorithm, the mean coefficients of the R, G and B channel of the original color image are calculated in the encoder and transmit those values as side information. As mentioned previously the PCM and the PCDM are used for the transmission of the mean coefficients (see Fig.2 (a) and (b)).

4. EXPERIMENTAL RESULTS

In this paper, the proposed algorithm is demonstrated for the applications described in the following subsections. The missing color samples are interpolated as shown in Fig. 3. The parameters α, β and γ denote the weighting factors for R, G and B channel, respectively. Moreover, α_{mn} denotes the n-th interpolation weighting factor for R channel samples to be estimated from m-neighboring samples. β_{mn} and γ_{mn} are defined similarly. The full color images (8 [bit/sample]) given in Fig. 4 are used in this simulation.

4.1. Experiment : Color image compression

In this section, we show low-bit rate color image transmission as an application of the proposed method. the weights of the demosaicing matrix are restricted to integer values described as follows.

$$\begin{cases} \alpha_{11} = \gamma_{11} = 4,\ \alpha_{21} = \alpha_{22} = \gamma_{21} = \gamma_{22} = 2 \\ \alpha_{41} = \alpha_{42} = \alpha_{43} = \alpha_{44} = \gamma_{41} = \gamma_{42} = \gamma_{43} = \gamma_{44} = 1 \end{cases}$$
$$\begin{cases} \beta_{21} = \beta_{22} = 2, \beta_{31} = \beta_{33} = 1, \beta_{32} = 2 \\ \beta_{41} = \beta_{42} = \beta_{43} = \beta_{44} = 1 \end{cases} \tag{7}$$

Moreover, the weighting coefficients of the color-to-gray conversion in (4) are defined as $p_R = p_G = p_B = 1$. Therefore, the resulting transformation matrix $\boldsymbol{A}_Y$ is also an integer matrix. It can be verified that if $\boldsymbol{A}_Y \in \mathbb{R}^{MN \times MN}$, its rank equals to $MN - 1$ (not full-rank). Based on the matrix configuration, we demonstrate the proposed method as the following steps:

1. Setting test CFA: Create a Bayer-pattern CFA $I_b(m, n)$ with 12 [bit/sample] from the test images by downsampling and scaling each of R, G and B channels.

(a) (b)

(c) (d) (e)

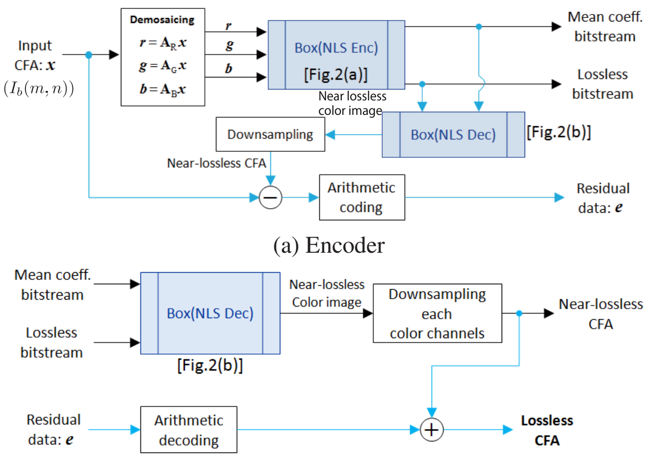
Fig. 4. Test images used in the simulation: (a) *Barbara* (256×256), (b) *Boy* (256×256), (c) *House* (256×256), (d) *Lena* (512×512), (e) *Pepper* (512×512)

Table 1. Result of Color Image Coding

The proposed method in NLS coding					
Test image	*Barbara*	*Boy*	*Lena*	*House*	*Pepper*
File size [KByte]	95	78	350	86	351
PSNR [dB]	50.93	51.08	51.49	51.23	50.84
Near-lossless color image coding JPEG-LS					
Test image	*Barbara*	*Boy*	*Lena*	*House*	*Pepper*
File size [KByte]	213	158	778	188	770
PSNR [dB]	49.93	49.94	49.89	49.89	49.98
The proposed method in LS coding					
Test image	*Barbara*	*Boy*	*Lena*	*House*	*Pepper*
Total size [KByte]	118	102	446	111	446
Lossless color image coding JPEG-LS					
Test image	*Barbara*	*Boy*	*Lena*	*House*	*Pepper*
File size [KByte]	251	195	928	225	918

2. Setting test color image: The CFA is interpolated by the demosaicing matrix defined as (7). The resulting color image (14 [bit/sample]) is set as the original color image $I_c(m, n)$. The data is stored in the suitable format, such as 16bit PPM format.
3. Procedure in the proposed encoder: Assume that the original test CFA is unknown and the test color image is given. The gray image (16 [bit/sample]) is obtained by summing R, G and B channels. Then, it is compressed by the LS mode of JPEG-LS and transmitted with the mean coefficients for R, G and B rounded by 16 bits and coded by the PCM. When the LS transmission, we send a residual data $E(m, n)$ between $I_c(m, n)$ and locally decoded color image $\tilde{\boldsymbol{I}}_c(m, n)$ (which is denoted in the next step).
4. Procedure in the proposed decoder: At the decoder, the CFA $\tilde{\boldsymbol{I}}_b(m, n)$ is recovered by the inverse demosaicing algorithm, and after the offset compensation by the transmitted mean coefficients decoded by the PCDM, the color image $\tilde{\boldsymbol{I}}_c(m, n)$ is reconstructed. In addition, the original color image $I_c(m, n)$ is losslessly obtained by summing the color image $\tilde{\boldsymbol{I}}_c(m, n)$ to the residual data $E(m, n)$ which was reconstructed by arithmetic decoding.

As a comparison, we perform the LS/NLS mode of JPEG-LS to $I_c(m, n)$ directly. The results are summarized in Table 1, where



(a) Encoder

(b) Decoder

Fig. 5. The diagram of the CFA LS/NLS coding algorithm (Encoder/Decoder). Blue arrows only work when LS coding.

"Total size" includes the result of the residual data coded by arithmetic coding to the result of proposed method in NLS encoding. For all the test images, PSNR of the proposed method shows a higher value than conventional ones. Since it is not necessary to send all the color information but only gray image, the proposed method can achieve much lower bit rate in both LS and NLS transmissions while hiding color information. Note that, the original full color image cannot be recovered without using the proposed decoding.

4.2. Experiment : Bayer CFA raw data compression

In a similar fashion, the proposed method can also be applied to CFA LS/NLS transmission, which is illustrated in Fig. 5. Specifically, the following steps are performed.

1. The additional process that creates color image from the input CFA by demosaicing is appended in the first step in the encoder side.
2. When LS transmission, residual data is obtained by the difference of input CFA and near-losslessly reconstructed CFA.
3. At the decoder side, near-lossless CFA is created from near-lossless color image by downsampling R, G, B channels and locating them in the Bayer-pattern CFA.
4. The decoding process in the LS transmission, the lossless CFA is obtained by adding the residual data to the near-lossless CFA.

We evaluated this CFA LS/NLS transmission. As a comparison, we perform the LS/NLS mode of JPEG-LS to $I_b(m,n)$ directly. The results are summarized in Table 2, In the NLS transmission, PSNR of the proposed method is higher than the conventional ones for all the test samples. In the cases of both the LS/NLS transmissions, the proposed method can save the bit rate at the same level as the conventional ones. As one more functionality, since gray image is transmitted, the original CFA cannot be recovered without using the proposed method.

5. CONCLUSION

In this paper, we introduced a novel framework for LS/NLS color image (and CFA) coding by inverse demosaicing. The proposed approach is different from conventional methods in the following two aspects; 1) representing demosaicing (and color-to-gray conversion) procedure as a matrix form and 2) multiplying its inverse demosaicing matrix to a transmitted LS gray image and performing the offset compensation. In the simulation, we demonstrated two applications for evaluation. In the color image transmission, the proposed algorithm can achieve LS/NLS image coding with quite limited bit rates compared with JPEG-LS which sends all color information. Moreover, the original color information cannot be obtained without using the proposed decoder because a gray image is only transmitted. In NLS transmission, PSNR of the proposed method is higher than the conventional ones. The proposed algorithm can also apply to CFA LS/NLS transmission while achieving a comparable compression ratio to conventional method. As well as color image transmission, the original CFA information cannot be recovered without using the proposed decoder.

Table 2. Result of CFA data Coding

The proposed method in NLS coding					
Test image	*Barbara*	*Boy*	*Lena*	*House*	*Pepper*
File size [KByte]	95	78	350	86	351
PSNR [dB]	50.64	51.34	51.47	51.33	51.16
Near-lossless CFA coding JPEG-LS					
Test image	*Barbara*	*Boy*	*Lena*	*House*	*Pepper*
File size [KByte]	76	73	343	77	349
PSNR [dB]	49.93	49.92	49.89	49.91	50.01
The proposed method in LS coding					
Test image	*Barbara*	*Boy*	*Lena*	*House*	*Pepper*
Total size [KByte]	103	86	383	94	384
Lossless CFA coding JPEG-LS					
Test image	*Barbara*	*Boy*	*Lena*	*House*	*Pepper*
File size [KByte]	88	86	392	90	400

6. ACKNOWLEDGEMENT

The authors would like to thank Dr. M. Yamagishi for helpful discussion and comments to the presented work.

7. REFERENCES

[1] M. J. Weinberger, G. Seroussi and G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," *IEEE Trans. Image Process.*, vol. 9, pp. 1309–1324, Aug. 2000.

[2] A. Okumura and J. Suzuki, "Nearly lossless compression of color still images and its performance evaluation," *IEICE Trans. Commun. B-I*, vol. J78-B-I, no. 7, pp. 279–290, July 1995.

[3] T. Nakachi, T. Fujii and J. Suzuki, "Lossless and near-lossless compression of still color images," in *Proc. IEEE Int. Conf. Image Process. (ICIP),* vol. 1, pp. 453–457, Oct. 1999.

[4] M. Domanski and K. Rakowski, "A simple technique for near lossless coding of color images", in *Proc. IEEE Int. Symp. Circuits and Syst. (ISCAS)*, vol. 111, pp. 299–302, 2000.

[5] B. E. Bayer, "Color Imaging Array," U.S. Patent 3,971,065, 1976.

[6] D. Menon, S. Andriani and G. Calvagno, "Demosaicing with directional filtering and *a posteriori* decision," *IEEE Transactions on Image Process.*, vol. 16, no. 1, pp. 132–141, Jan. 2007.