JOINT DEMOSAICING AND DENOISING

Keigo Hirakawa[*]

New England Conservatory of Music
Boston, MA 02115
kh237@cornell.edu

Thomas W. Parks

Cornell University
Electrical and Computer Engineering
Ithaca, NY 14853

ABSTRACT

The output image of a digital camera is subject to a severe degradation due to noise in the image sensor. This paper proposes a novel technique to combine demosaicing and denoising procedures systematically into a single operation by exploiting their obvious similarities. We first design a filter *as if* we are optimally estimating a pixel value from a noisy *single-color* image. With additional constraints, we show that the same filter coefficients are appropriate for CFA interpolation (demosaicing) given noisy sensor data. The proposed technique can combine many existing denoising algorithms with the demosaicing operation. In this paper, a total least squares denoising method is used to demonstrate the concept. The algorithm is tested on color images with pseudo-random noise and on raw sensor data from a real CMOS digital camera that we calibrated. The experimental results confirm that the proposed method suppresses noise (CMOS image sensor noise model) while effectively interpolating the missing pixel components, demonstrating a significant improvement in image quality when compared to treating demosaicing and denoising problems independently.

1. INTRODUCTION

A typical digital camera is subject to influences from noise in the image sensor. This sensor noise, often characterized as signal-dependent noise, is amplified by a series of image processing steps needed to produce a full-color representation of an image displayable on a monitor or a printer. A cost-effective digital camera uses a single-chip image sensor, with alternating patterns of red, green, and blue color filters applied to each pixel location. A method for reconstructing a full three-color representation of color image by estimating the missing pixel components from the color filter array (CFA) sampling pattern is called demosaicing. In demosaicing, we would like to preserve the sharpness of the edges while interpolating the missing pixel components. In the presence of noise, noise patterns form false edge structures, sharpening amplifies high frequency noise, and interpolation adds a structure to the noise too complicated to analyze. Removing noise after demosaicing, therefore, is impractical. Removing noise before the image processing pipeline is equally problematic because determining an image structure, necessary for effective noise reduction, from a sparse sampling lattice is difficult.

In recent years, many demosaicing algorithms have been published [6] [5] [9] [14] [10]. None of them address the image sensor noise problem explicitly (to the best of the knowledge of the authors). Considerable work has also been done on image denoising for signal-independent noise. While they are useful general methods, most neither take into account signal-dependent noise models nor accommodate CFA sampling patterns.

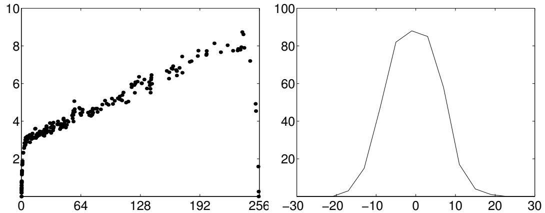

Fig. 1. (left) Standard deviation of noise vs. image sensor value. (right) Histogram of noise.

Noting that image interpolation and image denoising are both estimation problems, this paper proposes a unified approach to performing demosaicing and image denoising simultaneously. The novelty of our work is the development of a *constraint*, under which an optimal filter for estimating a pixel value from a noisy *single-color* image is also an optimal filter for demosaicing given noisy sensor data. Furthermore, many existing image denoising algorithms can be combined with the demosaicing operation using this proposed technique because this constraint is not very restrictive. For example, one may choose bilateral filtering because of computational efficiency, while another may choose a more sophisticated image denoising method for higher image quality.

2. CMOS SENSOR NOISE

In order for the denoising method to be effective, it is important to understand the noise characteristics in an image sensor. The CMOS photodiode active pixel sensor (APS) typically uses a photodiode and three transistors, all major sources of noise. While investigating the source of noise is beyond the scope of this paper, the readout noise takes the following general form [13]:

$$Y(i,j) = X(i,j) + (k_0 + k_1 X(i,j))\delta(i,j), \qquad (1)$$

where $X(i,j)$ and $Y(i,j)$ are the ideal and measured sensor values at pixel location (i,j), respectively, $\delta(i,j) \sim \mathcal{N}(0,1)$ is noise, and $k_0, k_1 \in \mathbb{R}$ are parameters.

Define $\mathcal{E}\{\cdot\}$ as the expectation operator. We independently verified the relationship in (1) by calibrating Agilent Technologies camera evaluation board HDCP-2000, equipped with a 300K pixel CMOS sensor [2]. Inside a room with controlled lighting, the Macbeth color chart is placed in the view of the camera in a fixed position. Assuming that $\mathcal{E}\{Y\} = X$ and that the colors inside the squares on the color chart are uniform, the average and the variance of 400 points measured from one square are taken to be the true X value and the noise variance for that X value, respectively. We repeat this experiment with varying levels of lighting. The programmable gain amplifier is held constant throughout the calibration experiments. All images are captured in unprocessed raw sensor data format. We assume that the variation among the pixel sensors is small compared to the level of noise.

[*] Hirakawa is on leave from Cornell University. We would like to thank Texas Instruments, Agilent Technologies, and Dr. B. Gunturk for their help.

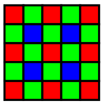

Fig. 2. Example $n \times n$ window, y_0, cropped from noisy sensor.

In fig. 1, the standard deviation of noise is plotted against the signal strength. It is clear that the standard deviation of the noise and the pixel values are roughly related by an affine equation, as in (1). Moreover, the histogram of noise reveals that it is not unreasonable to call the shape of the noise distribution Gaussian.

3. FILTER DESIGN

In this section, motivation for combining denoising and demosaicing methods is considered (the discussion is *independent* of the choice of the denoising algorithm). In this paper, the task of estimating pixel values from sparsely-sampled noisy sensor data is treated as a filter-design problem.

Let R, G, and B be the noise-free red, green, and blue images, respectively. Define R_s, G_s, and B_s as the red, green, and blue pixel values sampled by the image sensor according to the CFA pattern. In this paper, we work with Bayer pattern CFA, although the results extend to more general cases [1]. For ease of notation, let $X = R_s \cup G_s \cup B_s$ be the ideal image sensor output, Y is the measured value or the noisy image sensor output, and we assume they are related by (1). Our objective is to estimate R, G, and B given Y. In this section, a technique to estimate G from Y is presented (estimation of R and B is done in the same manner).

Consider a $n \times n$ window cropped from noisy sensor values, Y, as in fig. 2. Let us call this image patch y_0 (the pixel at the center of this patch is $y_0(0,0)$), and the corresponding ideal (i.e. noise free) sensor values x_0. Suppose the we are interested in estimating the *ideal* green pixel value at the center ($\hat{G}(0,0)$) by taking a linear combination of the measured values in this window:

$$\hat{G}(0,0) = \sum_{i,j} \alpha(i,j) y_0(i,j) \tag{2}$$

Note that even if the center of y_0 is green, we must still estimate the noise-free green pixel value. Therefore unlike the demosaicing problem, the above formula applies regardless of the color of the noisy center pixel $y_0(0,0)$ (i.e. we do not draw a distinction between the estimation of a missing pixel component from noisy data and the estimation of ideal pixel value when the noisy pixel value is already given). One obvious approach to choosing α is to treat each color plane independently, i.e. use only green pixels to estimate $G(0,0)$. However, many have argued that this is ineffective because it does not take advantage of the spatial redundancies between the different colors [6] [5] [9] [14].

We instead begin by assuming that the difference images $R - G, B - G, R - B$ are bandlimited signals [6]. This is equivalent to stating that the high-frequency components of R, G, and B are similar, while the low-frequency components may be dissimilar. Therefore, we impose a constraint that the coefficients corresponding to noisy red and blue values ($\{\alpha(i,j)\}_{i,j\in\{-2,0,2\}}$ and $\{\alpha(i,j)\}_{i,j\in\{-1,1\}}$ in fig. 2, respectively) add up to 0 when estimating $G(0,0)$, respectively. These coefficients are high-pass filters, effectively, and this guarantees that the low-frequency components of R_s and B_s do not contribute to the estimation of G.

But what should the filter coefficients be? Since only the high frequency components of R_s and B_s gets passed by α,

$$\hat{G}(0,0) = \sum_{i,j} \alpha(i,j)[G(i,j) + (k_0 + k_1 x_0(i,j))\delta(i,j)].$$

This substitution suggests that we design filter coefficients $\alpha(i,j)$ *as if* we are optimally estimating a pixel value from a noisy single-color image. Because the single-color image is unavailable from the noisy sensor data Y, we adapt another generalization, motivated by multi-resolution analysis [7]. If α is chosen such that

$$G(0,0) \simeq \sum_{i,j} \alpha(i,j)[G(2i,2j) + (k_0 + k_1 x_0(i,j))\delta(i,j)],$$

then

$$G(0,0) \simeq \sum_{i,j} \alpha(i,j)[G(i,j) + (k_0 + k_1 x_0(i,j))\delta(i,j)].$$

That is, the filter α designed to estimate $G(0,0)$ from the downsampled green image, $G(2i,2j)$, would also yield a satisfactory estimate if applied to full-resolution green image, $G(i,j)$. Working with $G(2i,2j)$ is convenient since downsampling Y by two in horizontal and vertical directions yields two smaller green images.

To summarize, the strategy for choosing the filter coefficients consists of three major steps:

1. Design filter α as if we are estimating $G(0,0)$ by taking a linear combination of $\{G(2i,2j)\}$.
2. Add a restriction to the filter such that the coefficients corresponding to the noisy red and blue values add up to zero, respectively.
3. Apply the filter to noisy image sensor output Y using (2).

We remind the readers that the same technique is used for estimating $R(0,0)$ and $B(0,0)$ from Y.

4. DENOISING METHOD

We are left with the task of designing a denoising algorithm that will fulfill the constraints outlined in section 3. There are many existing image denoising algorithms that are compatible with these constraints, offering flexibilities and choices in the design of an image processing pipeline. In this section, we describe a demosaicing algorithm based on a total least squares (TLS) image denoising method [7] as a proof-of-concept case study. Again, we focus exclusively on the linear estimation of $G(0,0)$, although the same techniques are used to estimate $R(0,0)$ and $B(0,0)$.

4.1. TLS Denoising Problem

In this section, we are interested in designing a filter α such that $\hat{G}(0,0) = \sum \alpha(i,j)[G(2i,2j) + (k_0 + k_1 x_0(i,j))\delta(i,j)]$ is an optimal estimate of $G(0,0)$ in the TLS sense.

Let Y_1, Y_2, Y_3, Y_4 be the four noisy single-color images obtained from downsampling Y by 2 in both horizontal and vertical directions. Define $\{y_1, \ldots, y_m\}$ as a set of vectorized $n \times n$ image patches cropped from $Y_1, \ldots, Y_4$, let $\{x_1, \ldots, x_m\}$ be the *ideal* red, green, and blue image patches (i.e. noise free) corresponding to $\{y_1, \ldots, y_m\}$, respectively, and $z_k = x_k + k_0\delta_k + k_1 \texttt{diag}(x_0)\delta_k$, where δ_k is a noise vector, and x_0 is as before. Suppose $\alpha \in \mathbb{R}^{n^2}$ is chosen such that for all k, ${z_k}^T\alpha$ is an optimal estimate for the center value in x_k. If this family of image patches is *similar* to y_0 then it is reasonable to assume that α will be a good filter for (2), also. A measure of similarity is introduced later.

Define $X^d = [x_1, \ldots, x_m]^T$, $Y^d = [y_1, \ldots, y_m]^T$, $Z^d = [z_1, \ldots, z_m]^T$, and let x^d be the column in X^d that corresponds to the center pixels of $\{x_1, \ldots, x_m\}$. In order that $Z^d\alpha$ be an optimal estimate for x^d in the TLS sense, α solves the following:

$$\min_{\alpha} \|A[E, e_0]M^T B\|_F^2, \tag{3}$$

subject to $(Z^d + E)\alpha = x^d + e_0$, where $\alpha = M\beta$. Note that α is in the column space of M, constraining α such that coefficients corresponding to red and blue pixels add up to zero, respectively. Our strategy is to solve for optimal β, and set $\alpha = M\beta$.

A variation of the TLS problem (3) using an affine approximation model was solved by de Groen [3]. He showed that the cost function, $\|A[E, e]MB\|_F^2$, is reduced greatly when the column-means of $A[Z^d, x^d]MB$ are subtracted from their respective columns first, suggesting a better model fit. In this paper, we modify the approach outlined in section 3 to take advantage of the affine approximation technique. More specifically, we solve for α that minimizes (3) subject to $(\tilde{Z}^d + E)\alpha = \tilde{x}^d + e_0$. Here, $\tilde{Z}^d = Z^d - [1, \ldots, 1]^T \bar{z}$ and $\tilde{x}^d = x^d - [1, \ldots, 1]^T \bar{x}^d$, where the entries in $\bar{z}$ are the average values of columns in Z, respectively, and $\bar{x}^d$ is the average value of x^d. $\tilde{X}^d$ and $\tilde{Y}^d$ are defined similarly. Note that the average of the column in Y^d corresponding to the center pixel is a good approximation for $\bar{x}^d$. Once α is solved, our optimal estimate for x^d is $\hat{x}^d = \tilde{Z}^d\alpha + \bar{x}^d$. More importantly, let $\bar{y}_0 \in \mathbb{R}^{n^2}$ be the vector average of $n \times n$ image patches cropped from noisy sensor output Y that are in the spatial vicinity of y_0 and whose locations of red and blue pixels match that of y_0. Our best estimate for $G(0,0)$ is

$$\hat{G}(0,0) = \tilde{y}_0\alpha + \bar{x}^d,$$

where $\tilde{y}_0 = y_0 - \bar{y}_0$.

4.2. Solution to TLS

Solving the TLS system is straightforward [7]. Let $N = n^2 - 1$, $A = \texttt{diag}(a_1, \ldots, a_m)$, and $B = \texttt{diag}(b_1, \ldots, b_{n^2-1})$. Using singular value decomposition $A[\tilde{Z}^d, \tilde{x}^d]B = U\Sigma V^T$, where $\Sigma = \texttt{diag}(\sigma_1, \ldots, \sigma_N)$ and $\sigma_k^2 > \sigma_{k+1}^2$, optimal β is [4]:

$$\beta = -\texttt{diag}(b_1, \ldots, b_{N-1})[v_{1,N}, \ldots, v_{N-1,N}]^T v_{N,N}^{-1} b_N, \quad (4)$$

where $[v_{1,N}, \ldots, v_{1,N}]^T$ is the right singular vector corresponding to σ_N. However, $\tilde{x}^d$ is not available in the denoising problem, thus making it difficult to compute V from singular value decomposition. Instead, define the matrix P:

$$\begin{aligned} P &= (A[\tilde{Z}^d, \tilde{x}^d]MB)^T (A[\tilde{Z}^d, \tilde{x}^d]MB) \\ &= (U\Sigma V^T)^T (U\Sigma V^T) = V\Sigma^2 V^T. \end{aligned}$$

Our strategy is to estimate P and obtain V through its eigen decomposition. Note that $\mathcal{E}\{\delta_k\} = 0$ and $\mathcal{E}\{\delta_k {\delta_l}^T\} = I$ if $k = l$ and 0 if otherwise. When $m \gg N$, $P = \mathcal{E}\{P\}$, and

$$P = B^T M^T \begin{bmatrix} P_{ZZ} & \tilde{X}^{gT} A^2 \tilde{x}^d \\ \tilde{x}^{gT} A^2 \tilde{X}^d & \tilde{x}^{gT} A^2 \tilde{x}^d \end{bmatrix} MB, \quad (5)$$

where $P_{ZZ} = \mathcal{E}\{\tilde{Z}^{gT} A^2 \tilde{Z}^d\}$. With some manipulations,

$$P_{ZZ} = \tilde{X}^{gT} A^2 \tilde{X}^d + \texttt{diag}(k_0 + k_1 x_0)^2 \left(\sum_{i=1}^{m} a_i^2 \right). \quad (6)$$

Given Y, P can be estimated. Let $P_{YY} = \tilde{Y}^{gT} A^2 \tilde{Y}^d$, and $\tilde{x}_i$ and $\tilde{y}_i$ are the ith row of $\tilde{X}^d$ and $\tilde{Y}^d$, respectively (hence $x_i = \bar{x} + \tilde{x}_i$). For $m \gg N$, $P_{YY} = \mathcal{E}\{P_{YY}\}$, and

$$\begin{aligned} P_{YY} = \tilde{X}^{gT} A^2 \tilde{X}^d + \sum_{i=1}^{m} a_i^2 \Big(&\texttt{diag}(k_0 + k_1\bar{x})^2 + k_1^2 \texttt{diag}(\tilde{x}_i)^2 \\ &+ 2k_1 \texttt{diag}(k_0 + k_1\bar{x}) \texttt{diag}(\tilde{x}_i) \Big). \end{aligned}$$

Using $\mathcal{E}\{\sum_i a_i^2 \tilde{y}_i\} = \sum_i a_i^2 \tilde{x}_i$ and the fact that diagonal entries of $\tilde{X}^{gT} A^2 \tilde{X}^d$ and $\sum_i a_i^2 \texttt{diag}(\tilde{x}_i)^2$ are identical, $\tilde{X}^{gT} A^2 \tilde{X}^d$ is estimated using the following procedure:

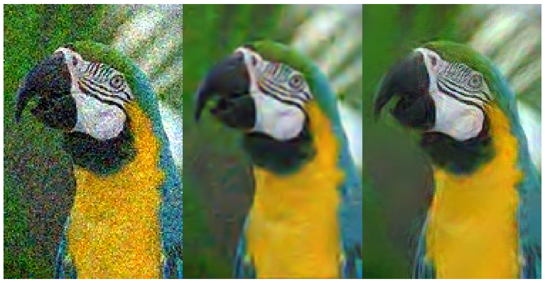

Fig. 3. An example using *parrot* picture: (left) method in [5], (middle) method in [5] and [11], (right) proposed method.

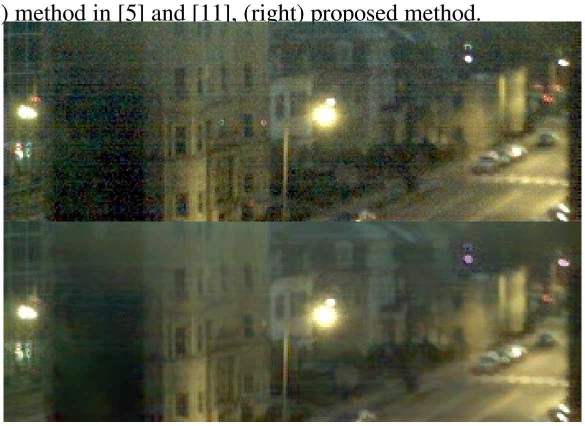

Fig. 4. An image captured with Agilent CMOS camera: (top) method in [5], (bottom) proposed method.

1. Compute $P_{YY} = \tilde{Y}^{gT} A^2 \tilde{Y}^d$.
2. Compute $P_{YY} - \sum a_i^2 [\texttt{diag}(k_0 + k_1\bar{x})^2 + 2k_1 \texttt{diag}(k_0 + k_1\bar{x}) \texttt{diag}(\tilde{y}_i)]$.
3. Multiply the diagonal entries of step 2 by $(1 + k_1^2)^{-1}$.

Call this estimate P_{XX}. Estimates of $\tilde{X}^{gT} A^2 \tilde{x}^d$, $\tilde{x}^{gT} A^2 \tilde{X}^d$, and $\tilde{x}^{gT} A^2 \tilde{x}^d$ are obtained by taking appropriate rows and columns of P_{XX}. P_{ZZ} is computed from P_{XX} using (6) and exchanging $\bar{y}_0$ in lieu of x_0 (substitution is justified in [7]). Thus, P is computable.

Optimal β is computed from (4), where V is given by the eigen decomposition of P in (5). Filter coefficients $\alpha = M\beta$ solves (3) subject to $(\tilde{Z}^d + E)\alpha = \tilde{x}^d + e_0$. Our best estimate for $G(0,0)$ is

$$\hat{G}(0,0) = \tilde{y}_0^T \alpha + \bar{x}^d = \tilde{y}_0^T M\beta + \bar{x}^d.$$

Same technique is used to estimate $R(0,0)$ and $B(0,0)$.

4.3. Denoising Improvements

Above, A and B are weighting matrices. The $n \times n$ image patches $\{y_1, \ldots, y_m\}$ are taken from $Y_1, \ldots, Y_4$ in the spatial vicinity of $G(0,0)$ [7]. However, y_k is not meaningful unless it is reasonably similar to x_0. To prioritize $\{y_1, \ldots, y_m\}$ in the order of similarity, larger weight is given if $H^T y_k$ is similar to $H^T y_0$: $a_k = exp(-(y_0 - y_k)^T H H^T (y_0 - y_k)/k_A)$, where $k_A \in \mathbb{R}$ is constant. In our simulation, B is fixed ($b_1, ..., b_{N-1} = 1$ and $b_N = 0.5$) and H is a highpass filter.

4.4. Pre-Processing

The effectiveness of the TLS denoising algorithm depends on the ability to estimate P matrix accurately. Given $\delta(i,j) \sim N(0,1)$, pixels occasionally stand out because $\delta(i,j)$ at that pixel position is far greater than its standard deviation. This is problematic because it degrades our estimate for P greatly. To work around this problem, we propose to *prune* the outliers. For each pixel location in Y,

Table 1. Performance of demosaicing and denoising algorithms on the "parrots" image, evaluated using average SCIELAB error [15]. Noise levels considered were $(k_0, k_1) = (0, 0)$, $(25, 0)$, and $(10, 0.1)$. n/a means not available or not necessary.

	demosaicing method in [5]			demosaicing method in [6]			proposed method		
	$(0,0)$	$(25,0)$	$(10,0.1)$	$(0,0)$	$(25,0)$	$(10,0.1)$	$(0,0)$	$(25,0)$	$(10,0.1)$
no denoising	0.8108	6.5052	4.6319	0.7768	6.4220	4.6731	0.9922	3.7504	2.9535
denoising method in [12]	n/a	4.1660	n/a	n/a	4.2926	n/a	n/a	n/a	n/a
denoising method in [11]	n/a	4.1166	n/a	n/a	4.2271	n/a	n/a	n/a	n/a
denoising method in [7]	n/a	3.8954	3.0283	n/a	4.0702	3.1912	n/a	n/a	n/a

1. Let w be a set of pixels in Y that fall within the $L \times L$ neighborhood of the pixel of interest, and whose color is the same as the pixel of interest.
2. Find the kth largest and kth smallest pixel values in w.
3. If the pixel of interest is larger (smaller) than the kth largest (smallest) value in w, replace it with the kth largest (smallest) pixel value in w.

This pre-processing procedure is a particularly good match for working with image sensors because defective pixels (*hot-pixels* or *dead-pixels*) due to manufacturing variabilities will be removed, also.

5. IMPLEMENTATION AND RESULTS

TLS algorithm is implemented by taking 5×5 image patches from a 25×25 neighborhood. Pre-processing had a window size of 11×11, and we picked the 4th largest (smallest) pixel values. Parameters k_0 and k_1 were available *a priori*. Experiments were performed on color images corrupted according to (1) using pseudo-random noise and sampled according to CFA. We compare our results to the state-of-the-art demosaicing algorithms [5] [6] followed by denoising algorithms [11] [12] [7]. Denoising algorithms were performed on each color plane independently ([12] and [11] do not work with signal-dependent noise). Table 1 clearly shows the benefits to considering demosaicing and denoising as a single operation. Note also that in the absence of noise, the other demosaicing algorithms may perform better than the proposed algorithm. Fig. 3 shows example output images. The amplification of noise is seen due to demosaicing, and while applying denoising algorithms helps the overall image quality, the proposed algorithm is both sharper and significantly less noisy.

Fig. 4 shows the images from an experiment using images taken from an Agilent CMOS camera in low light. Images were captured in a raw-data format with the same setup as above. The parameters used were $(k_0, k_1) = (3, 0.02)$. After demosaicing, the images were processed with color space conversion and gamma correction ($\gamma = 1.8$). The illuminant was known *a priori*. The demosaicing methods in [5] maintain high contrast, but *grainy* noise is highly visible in the dark regions. The proposed algorithm eliminates graininess.

6. CONCLUSION

This paper presented a unified method to combine demosaicing and image denoising procedures. The filtering coefficients were restricted such that only the high frequency components of the image signals contribute to the estimation of pixel values of different colors. With substitutions, the multi-colored demosaicing/denoising problem was simplified to a single-color denoising problem. A Total Least Squares algorithm was developed as a proof-of-concept, and it was tested on color images with pseudo-random noise and on raw sensor data from a real CMOS digital camera. The experimental results verify that performing demosaicing and denoising simultaneously is far more effective than treating the demosaicing and denoising problems independently.

7. REFERENCES

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

[2] M. Borg, R. Mentzer, K. Singh, "Digital Imaging using CMOS sensors," *Proc. International IC-China Conference and Exhibition, Eletronic Engineering Times*, pp. 37-47, 2001.

[3] P. de Groen, "An Introduction to Total Least Squares", Nieuw Archief voor Wiskunde, Vierde serie, deel 14, 1996.

[4] G. H. Golub, C. F. Van Loan, "Matrix Computations", The Johns Hopkins University Press, 3rd ed., 1996.

[5] B. K. Gunturk, J. Glotzbach, Y. Altunbasak, R. W. Schafer, R. M. Mersereau, "Demosaicking: Color filter array interpolation in single chip digital cameras," *IEEE Signal Processing Magazine* (Special Issue on Color Image Processing), vol. 22, January 2005.

[6] K. Hirakawa, T. W. Parks, "Adaptive Homogeneity-Directed Demosaicing Algorithm," *IEEE Trans Image Processing*, vol. 14, no. 3, March 2005.

[7] K. Hirakawa, T. W. Parks, "Image Denoising for Signal-Dependent Noise," *ICASSP*, 2005.

[8] I. M. Johnstone, B. W. Silverman, "Wavelet threshold estimators for data with correlated noise," *Journal of Royal Statist. Soc.*, vol. B 59, 1997.

[9] W. Lu, Y.-P, Tan, "Color Filter Array Demosaiciking: New Method and Performance Measures," *IEEE Trans. Image Processing*, vol. 12, October 2003.

[10] R. Lukac, K. N. Plataniotis, D. Hatzinakos, M. Aleksic, "A novel cost effective demosaicing approach," *IEEE Trans. Consumer Electronics*, vol. 50, February 2004.

[11] A. Pizurica, W. Philips, I. Lemahieu, M. Acheroy, "A joint inter- and intrascale statistical model for Bayesian wavelet based image denoising," *IEEE Trans. Image Processing*, vol. 11, 2002.

[12] J. Portilla, V. Strela, M. J. Wainwright, E. P. Simoncelli, "Image Denoising Using Scale Mixture of Gaussians in the Wavelet Domain,"*IEEE Trans. Image Processing*, vol. 12, 2003.

[13] H. Tian, B. Fowler, A. E. Gamal, "Analysis of Temporal Noise in CMOS Photodiode Active Pixel Sensor," *IEEE Jnl. Solid-State Circuits*, vol. 36, 2001.

[14] X. Wu, N. Zhang, "Primary-Consistent Soft-Decision Color Demosaic for Digital Cameras," *IEEE Proc. ICIP*, vol. 1, September, 2003.

[15] X. Zhang, D. A. Silverstein, J. E. Farrell, B. A. Wandell, "Color Image Quality Metric S-CIELAB and its application on Halftone Texture Visibility," *COMPCON97 Digest of Papers*, IEEE, pp. 44-48, 1997.