# Color Plane Interpolation Using Alternating Projections 

Bahadir K. Gunturk, Student Member, IEEE, Yucel Altunbasak, Senior Member, IEEE, and Russell M. Mersereau, Fellow, IEEE


#### Abstract

Most commercial digital cameras use color filter arrays to sample red, green, and blue colors according to a specific pattern. At the location of each pixel only one color sample is taken, and the values of the other colors must be interpolated using neighboring samples. This color plane interpolation is known as demosaicing; it is one of the important tasks in a digital camera pipeline. If demosaicing is not performed appropriately, images suffer from highly visible color artifacts. In this paper we present a new demosaicing technique that uses inter-channel correlation effectively in an alternating-projections scheme. We have compared this technique with six state-of-the-art demosaicing techniques, and it outperforms all of them, both visually and in terms of mean square error.


Index Terms-Bayer pattern, color filter array, demosaicing, POCS.

## I. Introduction

SINGLE-CHIP digital cameras use color filter arrays to sample different spectral components, such as red, green, and blue. At the location of each pixel only one color sample is taken, and the other colors must be interpolated from neighboring samples. This color plane interpolation is known as demosaicing, and it is one of the important tasks in a digital camera pipeline. If demosaicing is not performed appropriately, images suffer from highly-visible color artifacts. The most commonly used color pattern is the "Bayer" pattern [1]. As seen in Fig. 1, in a Bayer pattern, green samples are obtained on a quincunx lattice (checkerboard pattern), and red and blue samples are obtained on rectangular lattices. The density of the red and blue samples is one-half that of the green ones.

Demosaicing methods can be grouped into two distinct classes. The first class applies well-known interpolation techniques to each color channel separately. These techniques include nearest-neighbor replication, bilinear interpolation, and cubic spline interpolation. Although these single-channel algorithms can provide satisfactory results in smooth regions of an image, they usually fail in high-frequency regions,

[^0]

Fig. 1. Bayer pattern.
especially along edges. For natural images better performance is possible than is achieved by these techniques because of the high cross-correlation between color channels. The second class of algorithms exploits this inter-channel correlation, and has significantly better performance than the first class. One approach in this class is smooth hue transition [2]-[4]. Smooth hue transition algorithms are based on the assumption that hue does not change abruptly between neighboring pixel locations. As a first step, these algorithms interpolate the luminance (green) ${ }^{1}$ channel, which is usually done using bilinear interpolation. The chrominance channels (red and blue) are estimated from the bilinearly interpolated "red hue" (red-to-green ratio) and "blue hue" (blue-to-green ratio). To be more explicit, the interpolated "red hue" and "blue hue" values are multiplied by the green value to determine the missing red and blue values at a particular pixel location. Instead of interpolating the hue, it is also possible to interpolate the logarithm of the hue [3], [4].

Another approach that exploits inter-channel correlation is edge-directed interpolation [5]-[9]. The main difference between this approach and the previous one is that the bilinear interpolation of the green channel is replaced by adaptive interpolation to prevent interpolating across edges. In [5], first-order horizontal and vertical gradients are computed at each missing green location on the Bayer pattern. If the horizontal gradient is greater and the vertical gradient is less than a predetermined threshold, suggesting a possible edge in the horizontal direction, interpolation is performed along the vertical direction. If the vertical gradient is larger and the horizontal gradient is less than the threshold, interpolation is performed only in the horizontal direction. When the horizontal and vertical gradients are nearly equal, (that is, both gradients

[^1]

Fig. 2. Images used in the experiments. (These images are referred as Image 1 to Image 20 in the paper, enumerated from left-to-right, and top-to-bottom.).

TABLE I
Inter-Channel Correlation in Different Subbands

|  | Red/Green Corr. Coef. |  |  |  | BIue/Green Corr. Coef. |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | LL | LH | HL | HH | LL | LH | HL | HH |
| 1 | 0.8376 | 0.9959 | 0.9903 | 0.9866 | 0.9911 | 0.9970 | 0.9940 | 0.9838 |
| 2 | 0.4792 | 0.9275 | 0.9346 | 0.9414 | 0.9723 | 0.9891 | 0.9889 | 0.9431 |
| 3 | 0.8854 | 0.9275 | 0.9899 | 0.9822 | 0.8917 | 0.9754 | 0.9856 | 0.9682 |
| 4 | 0.9765 | 0.9963 | 0.9926 | 0.9794 | 0.9924 | 0.9917 | 0.9810 | 0.9605 |
| 5 | 0.8234 | 0.9789 | 0.9800 | 0.9483 | 0.9045 | 0.9717 | 0.9761 | 0.9391 |
| 6 | 0.9629 | 0.9947 | 0.9937 | 0.9822 | 0.9725 | 0.9932 | 0.9931 | 0.9770 |
| 7 | 0.9462 | 0.9906 | 0.9861 | 0.9404 | 0.8464 | 0.9834 | 0.9776 | 0.9252 |
| 8 | 0.9540 | 0.9867 | 0.9841 | 0.9458 | 0.9698 | 0.9807 | 0.9706 | 0.9224 |
| 9 | 0.8083 | 0.9863 | 0.9870 | 0.9768 | 0.9730 | 0.9942 | 0.9933 | 0.9765 |
| 10 | 0.9112 | 0.9899 | 0.9759 | 0.9316 | 0.9670 | 0.9878 | 0.9784 | 0.9120 |
| 11 | 0.9793 | 0.9968 | 0.9953 | 0.9932 | 0.9603 | 0.9895 | 0.9854 | 0.9805 |
| 12 | 0.8595 | 0.9548 | 0.9445 | 0.9421 | 0.9882 | 0.9809 | 0.9841 | 0.9506 |
| 13 | 0.9838 | 0.9962 | 0.9895 | 0.9599 | 0.9492 | 0.9943 | 0.9853 | 0.9471 |
| 14 | 0.9852 | 0.9864 | 0.9831 | 0.9519 | 0.9778 | 0.9811 | 0.9784 | 0.9418 |
| 15 | 0.9120 | 0.9832 | 0.9812 | 0.9771 | 0.8113 | 0.9746 | 0.9630 | 0.9700 |
| 16 | 0.9652 | 0.9961 | 0.9911 | 0.9562 | 0.9078 | 0.9923 | 0.9919 | 0.9578 |
| 17 | 0.9956 | 0.9873 | 0.9913 | 0.9588 | 0.9767 | 0.9594 | 0.9773 | 0.9177 |
| 18 | 0.8850 | 0.9926 | 0.9901 | 0.9771 | 0.9094 | 0.9871 | 0.9838 | 0.9614 |
| 19 | 0.8660 | 0.9542 | 0.9448 | 0.9352 | 0.8698 | 0.9693 | 0.9737 | 0.9598 |
| 20 | 0.9765 | 0.9809 | 0.9741 | 0.9460 | 0.9654 | 0.9696 | 0.9629 | 0.9259 |

are less than or greater than the threshold), the green value is obtained by averaging its four neighbors. Interpolation of the red and blue channels can be done by either interpolating color ratios (as in smooth hue transition) or by interpolating the color differences instead of the color ratios.

A different version of this approach was proposed by Laroche and Prescott [6]. There the chrominance channels are used instead of the luminance channel to determine the gradients. In order to determine the horizontal and vertical gradients at a blue (red) sample, second-order derivatives of blue (red) values are computed in the corresponding direction. The red and blue chan-
nels are interpolated as for the smooth hue transition approach, but this time the color differences are interpolated instead of the color ratios.

Instead of interpolating color differences or color ratios, it is also possible to use the inter-channel correlation as a correction term in the interpolation [7]-[9]. In [9], Hamilton and Adams used second-order derivatives of the chrominance samples as correction terms in the green channel interpolation. To determine the gradient at a blue (red) sample location, the second-order derivative of blue (red) pixels values are added to the first-order derivative of the green values. The second-order
derivative of the blue (red) pixels is also added to the average of the green values in the minimum gradient direction. The red and blue channels are interpolated similarly with second-order green derivatives used as the correction terms.
Kimmel later combined the smooth hue transition and edgedirected interpolation approaches in an iterative scheme [10]. In his algorithm, first-order derivatives of the green channel information are used to compute edge indicators in eight possible directions. Hue values are interpolated using these edge indicators, and missing color intensities are determined according to the interpolated hues. The color channels are then updated iteratively to obey the color-ratio rule. He also proposed an inverse diffusion process to enhance the images further.

There are also more complicated demosaicing approaches. In [11], Chang et al. applied interpolation using a threshold-based variable number of gradients. In that approach, a set of gradients is computed in the $5 \times 5$ neighborhood of the pixel under consideration. A threshold is determined for those gradients, and the missing value is computed using the pixels corresponding to the gradients that pass the threshold. A similar algorithm was proposed in [12], where the green channel is used to determine the pattern at a particular pixel, and then a missing red (blue) pixel value is estimated as a weighted average of the neighboring pixels according to the pattern. In addition, there are pattern recognition [13], restoration-based [14]-[16], and sampling theory point of view [17], [18] approaches.

In this paper, we present a very effective means of using inter-channel correlation in demosaicing. The algorithm defines constraint sets based on the observed color samples and prior knowledge about the correlation between the channels. It reconstructs the color channels by projecting the initial estimates onto these constraint sets. We have compared our algorithm with the various other techniques that we have outlined above, and it outperforms them both visually and in terms of its mean square error. Section II presents the motivation and details of this algorithm. Its experimental performance and comparisons with other techniques are given in Section III. A complexity analysis is provided in Section IV.

## II. Demosaicing Using Alternating Projections

There are two observations that are important for the demosaicing problem. The first is that for natural images there is a high correlation between the red, green, and blue channels. All three channels are very likely to have the same texture and edge locations. The second observation is that digital cameras use a color filter array (CFA) in which the luminance (green) channel is sampled at a higher rate than the chrominance (red and blue) channels. Therefore, the green channel is less likely to be aliased, and details are preserved better in the green channel than in the red and blue channels. In demosaicing, it is the interpolation of the red and blue channels that is the limiting factor in performance. Color artifacts, which become severe in high-frequency regions such as edges, are caused primarily by aliasing in the red and blue channels. Although this fact is acknowledged by the authors of most demosaicing


Fig. 3. CFA sampling of the images. (a) Frequency support of an image. (b) Spectrum of the sampled green channel. (c) Spectrum of the sampled red and blue channels.
algorithms, inter-channel correlation has not been used effectively to retrieve the aliased high-frequency information in the red and blue channels. This paper proposes a new demosaicing algorithm that does remove aliasing in these channels using an alternating-projections scheme. It defines constraint sets using

TABLE II
Correlation Between Original Images and Bilinearly Interpolated Observations in Different Subbands

| $\begin{gathered} \hline \text { Image } \\ \text { n॰ } \\ \hline \end{gathered}$ | Red channel |  |  |  | Green channel |  |  |  | Blue channel |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  | LL | LH | HL | HH | LL | LH | HL | HH | LL | LH | HL | HH |
| 1 | 0.9721 | 0.5936 | 0.6416 | 0.3029 | 0.9979 | 0.8968 | 0.8078 | 0.3012 | 0.9789 | 0.5956 | 0.5266 | 0.3052 |
| 2 | 0.9923 | 0.7130 | 0.5974 | 0.2597 | 0.9982 | 0.8763 | 0.8496 | 0.3532 | 0.9871 | 0.6752 | 0.6377 | 0.2639 |
| 3 | 0.9912 | 0.6094 | 0.6326 | 0.3393 | 0.9983 | 0.8680 | 0.8497 | 0.3581 | 0.9893 | 0.6460 | 0.5537 | 0.3355 |
| 4 | 0.9954 | 0.5888 | 0.5994 | 0.3023 | 0.9993 | 0.9296 | 0.6762 | 0.3124 | 0.9918 | 0.6196 | 0.5511 | 0.3023 |
| 5 | 0.9973 | 0.6928 | 0.6999 | 0.2726 | 0.9993 | 0.9185 | 0.8751 | 0.3072 | 0.9969 | 0.6962 | 0.6776 | 0.2622 |
| 6 | 0.9850 | 0.5750 | 0.5722 | 0.2827 | 0.9986 | 0.7953 | 0.8936 | 0.2528 | 0.9807 | 0.6009 | 0.5391 | 0.2844 |
| 7 | 0.9960 | 0.6649 | 0.6641 | 0.2510 | 0.9993 | 0.9008 | 0.8692 | 0.2509 | 0.9940 | 0.6814 | 0.6285 | 0.2502 |
| 8 | 0.9965 | 0.6683 | 0.6808 | 0.2938 | 0.9994 | 0.9006 | 0.8397 | 0.3205 | 0.9945 | 0.6807 | 0.6228 | 0.2838 |
| 9 | 0.9915 | 0.6185 | 0.5991 | 0.2958 | 0.9990 | 0.8831 | 0.8059 | 0.3378 | 0.9932 | 0.6206 | 0.5745 | 0.2923 |
| 10 | 0.9963 | 0.6407 | 0.7349 | 0.2561 | 0.9997 | 0.9170 | 0.8235 | 0.2757 | 0.9970 | 0.6629 | 0.6223 | 0.2358 |
| 11 | 0.9842 | 0.5547 | 0.5367 | 0.3013 | 0.9968 | 0.8826 | 0.7177 | 0.4017 | 0.9860 | 0.5727 | 0.4999 | 0.3017 |
| 12 | 0.9986 | 0.6747 | 0.6726 | 0.2847 | 0.9999 | 0.7869 | 0.9066 | 0.3259 | 0.9987 | 0.6691 | 0.5136 | 0.2715 |
| 13 | 0.9955 | 0.6243 | 0.7102 | 0.2694 | 0.9995 | 0.9431 | 0.6533 | 0.3163 | 0.9944 | 0.6029 | 0.6623 | 0.2560 |
| 14 | 0.9975 | 0.6695 | 0.6711 | 0.2810 | 0.9995 | 0.8598 | 0.8677 | 0.3545 | 0.9972 | 0.6633 | 0.5752 | 0.2806 |
| 15 | 0.9932 | 0.5781 | 0.6146 | 0.2920 | 0.9972 | 0.8427 | 0.8111 | 0.3533 | 0.9812 | 0.5872 | 0.5363 | 0.2839 |
| 16 | 0.9924 | 0.6597 | 0.5801 | 0.2880 | 0.9989 | 0.8950 | 0.8288 | 0.2445 | 0.9906 | 0.6410 | 0.5602 | 0.2788 |
| 17 | 0.9989 | 0.6211 | 0.7646 | 0.2976 | 0.9998 | 0.8261 | 0.8450 | 0.3070 | 0.9992 | 0.6944 | 0.7254 | 0.2756 |
| 18 | 0.9910 | 0.6174 | 0.6133 | 0.2996 | 0.9987 | 0.9156 | 0.7558 | 0.3255 | 0.9933 | 0.6242 | 0.6232 | 0.2967 |
| 19 | 0.9956 | 0.7095 | 0.6023 | 0.2841 | 0.9990 | 0.8524 | 0.8714 | 0.3234 | 0.9953 | 0.6641 | 0.5696 | 0.2872 |
| 20 | 0.9935 | 0.6468 | 0.6387 | 0.2993 | 0.9986 | 0.8750 | 0.8047 | 0.3658 | 0.9915 | 0.6305 | 0.6041 | 0.3254 |

both the inter-channel correlation and the observed data, and reconstructs the red and blue channels by projecting initial estimates onto these constraint sets.

Section II-A quantifies the degree of cross-correlation between the color channels. Section II-B illustrates the aliasing that results from CFA sampling and motivates a detail-retrieving interpolation scheme. Section II-C derives the constraint sets used by the proposed demosaicing scheme. Section II-D presents the details of the implementation and Section II-E describes some extensions.

## A. Inter-Channel Correlation

In natural images the color channels are highly mutually correlated. Since all three channels are very likely to have the same edge content, we expect this inter-channel correlation to be even higher when it is measured between the high-frequency components. (The reason for investigating correlation in the high-frequency components will become evident in Section II-B.) In order to illustrate this we decomposed the three color channels of 20 natural images (Fig. 2) into subbands. We used two-dimensional separable filters constructed from a low-pass filter $\left(h_{0}=\left[\begin{array}{lll}1 & 2 & 1\end{array}\right] / 4\right)$ and a high-pass filter ( $h_{1}=\left[\begin{array}{lll}1 & -2 & 1\end{array}\right] / 4$ ) to decompose each image into its four subbands: (LL) both rows and columns are low-pass filtered, (LH) rows are low-pass filtered, columns are high-pass filtered, (HL) rows are high-pass filtered, columns are low-pass filtered, (HH) both rows and columns are high-pass filtered. The inter-channel correlation coefficients for each of these four subbands was computed using the formula

$$
\begin{equation*}
C_{x, y}=\frac{\sum_{\left(n_{1}, n_{2}\right)}\left(x\left(n_{1}, n_{2}\right)-\mu_{x}\right)\left(y\left(n_{1}, n_{2}\right)-\mu_{y}\right)}{\sqrt{\sum_{\left(n_{1}, n_{2}\right)}\left(x\left(n_{1}, n_{2}\right)-\mu_{x}\right)^{2}} \sqrt{\sum_{\left(n_{1}, n_{2}\right)}\left(y\left(n_{1}, n_{2}\right)-\mu_{y}\right)^{2}}} \tag{1}
\end{equation*}
$$

where $\left(n_{1}, n_{2}\right)$ are integers denoting the spatial coordinates, $x\left(n_{1}, n_{2}\right)$ and $y\left(n_{1}, n_{2}\right)$ are the samples of two different color channels within a subband, and $\mu_{x}$ and $\mu_{y}$ are the means of $x\left(n_{1}, n_{1}\right)$ and $y\left(n_{1}, n_{2}\right)$, respectively. The summation is done over all possible $\left(n_{1}, n_{2}\right)$ in a subband. The correlation coefficients between the red and green, and blue and green channels are tabulated in Table I. As seen in that table, the correlation coefficients for the high-frequency subbands $(L H, H L, H H)$ are larger than 0.9 for all images, and the highest correlation coefficient for a particular image is among these subbands. The low-frequency subbands $L L$ are also highly correlated (their correlation coefficients are greater than 0.8 for most of the images.), but they are not as highly correlated as the high frequency subbands.

Section II-B examines the effects of CFA sampling on these subbands. In particular, we show that the high-frequency subbands of the red and blue channels are the most affected.

## B. Color Plane Sampling

As seen in Fig. 1, in a Bayer pattern the green channel, sampled with a quincunx lattice, is less likely to be aliased than the red and blue channels, which are sampled with less dense rectangular lattices. This can easily be illustrated in the frequency domain. Fig. 3(a) depicts the Fourier spectrum of an image with $f_{m}$ being the maximum observable frequency. When this image is captured with a digital camera, the color planes are sampled according to a CFA, which is generally the Bayer pattern. As illustrated in Fig. 3(b) and (c), while there is no aliasing in the green channel, the red and blue channels are aliased.

This can also be confirmed for the images in Fig. 2. In Table II, the correlation coefficients between the original channels and the bilinearly-interpolated (from the CFA samples) channels are displayed for all subbands. Two important things can be observed in that table. First, the high-frequency ( $L H, H L, H H$ ) subbands are degraded the most. Second this
degradation is more severe in the red and blue channels than in the green channel, especially in the $L H$ and $H L$ subbands.

In Section II-A, we showed that the color channels are highly mutually correlated, especially in the high-frequency subbands. In this section, we illustrated the fact that the high-frequency subbands of the red and blue channels are affected the most in CFA sampling. These two observations imply that the high-frequency information of the green channel can be used to help estimate the high-frequency components of the red and blue channels. One way to achieve this is with a set-theoretic reconstruction.

## C. Constraint Sets

Set-theoretic reconstruction techniques produce solutions that are consistent with the information arising from observed data or prior knowledge about the solution. Each piece of information is associated with a constraint set in the solution space, and the intersection of these sets represents the space of acceptable solutions [19]. For the demosaicing problem, we define two types of constraint sets, one coming from the observed data, and the other based on the prior knowledge of the inter-channel correlation.

The first constraint set comes from the observed color samples. The interpolated color channels must be consistent with the color samples captured by the digital camera. We denote $O\left(n_{1}, n_{2}\right)$ as this observed data, which has red, green, and blue samples placed according to the CFA used. $\left(n_{1}, n_{2}\right)$ are ordered pairs of integers denoting the pixel locations. By defining $\Lambda_{R}$, $\Lambda_{G}$, and $\Lambda_{B}$ as the set of pixel locations, $\left(n_{1}, n_{2}\right)$, that have the samples of red, green, and blue channels, respectively, we can write the "observation" constraint set as follows:

$$
\begin{align*}
& C_{o}=\left\{S\left(n_{1}, n_{2}\right): S\left(n_{1}, n_{2}\right)=O\left(n_{1}, n_{2}\right)\right. \\
& \left.\forall\left(n_{1}, n_{2}\right) \in \Lambda_{S}, S=R, G, B\right\} \tag{2}
\end{align*}
$$

where $S$ is a generic symbol for the interpolated color channels, which can be $R$ for the red channel, $G$ for the green channel, and $B$ for the blue channel.
The second constraint set is a result of the Sections II-A and B. In Section II-A, it was shown that color channels have very similar detail (high-frequency) subbands. This information would not be enough to define constraint sets if all channels lost the same amount of information in sampling. However, Section II-B pointed out that the red and blue channels lose more information (details) than the green channel when captured with a color filter array. Therefore, we can define constraint sets on the red and blue channels that force their high-frequency components to be similar to the high-frequency components of the green channel. This proves to be a very effective constraint set, since the main source of color artifacts in a demosaiced image is the inconsistency of the channels, especially, along the edges.

Before formulating this constraint set, we need to provide some information about the filter bank structure that is used to decompose the channels. Referring to Fig. 4, the filter bank


Fig. 4. Analysis and synthesis filterbanks for one-level decomposition.


Fig. 5. Convergence for one-level decomposition.


Fig. 6. Analysis and synthesis filterbanks for two-level decomposition.


Fig. 7. Convergence for two-level decomposition.
performs an undecimated wavelet transform, with $H_{0}(z)$ and $H_{1}(z)$ denoting low-pass and high-pass filters, respectively. These analysis filters $\left(H_{0}(z)\right.$ and $\left.H_{1}(z)\right)$ constitute a perfect


Fig. 8. Fine tuning of green channel is done from observed red and blue samples.
reconstruction filter bank with the synthesis filters $G_{0}(z)$ and $G_{1}(z)$. The perfect reconstruction condition can be written as

$$
\begin{equation*}
H_{0}(z) G_{0}(z)+H_{1}(z) G_{1}(z)=1 \tag{3}
\end{equation*}
$$

By denoting $h_{0}(\cdot)$ and $h_{1}(\cdot)$ as the impulse responses of $H_{0}(z)$ and $H_{1}(z)$, respectively, we can write the four subbands of a two-dimensional signal $S\left(n_{1}, n_{2}\right)$ as follows:

$$
\begin{align*}
& \left(\mathcal{W}_{1} S\right)\left(n_{1}, n_{2}\right)=h_{0}\left(n_{1}\right) *\left[h_{0}\left(n_{2}\right) * S\left(n_{1}, n_{2}\right)\right]  \tag{4}\\
& \left(\mathcal{W}_{2} S\right)\left(n_{1}, n_{2}\right)=h_{1}\left(n_{1}\right) *\left[h_{0}\left(n_{2}\right) * S\left(n_{1}, n_{2}\right)\right]  \tag{5}\\
& \left(\mathcal{W}_{3} S\right)\left(n_{1}, n_{2}\right)=h_{0}\left(n_{1}\right) *\left[h_{1}\left(n_{2}\right) * S\left(n_{1}, n_{2}\right)\right]  \tag{6}\\
& \left(\mathcal{W}_{4} S\right)\left(n_{1}, n_{2}\right)=h_{1}\left(n_{1}\right) *\left[h_{1}\left(n_{2}\right) * S\left(n_{1}, n_{2}\right)\right] \tag{7}
\end{align*}
$$

where $\left(\mathcal{W}_{1} S\right)$ is the approximation subband, and $\left(\mathcal{W}_{2} S\right)$, $\left(\mathcal{W}_{3} S\right),\left(\mathcal{W}_{4} S\right)$ are the horizontal, vertical, and diagonal detail subbands, respectively.

Now we can define the "detail" constraint set $C_{d}$ that forces the details (high-frequency components) of the red and blue channels to be similar to the details of the green channel as shown in (8) at the bottom of the page, where $T\left(n_{1}, n_{2}\right)$ is a positive threshold that quantifies the "closeness" of the detail subbands to each other. If the color channels are highly correlated, then the threshold should be small; if the correlation is not high, then the threshold should be larger. Although $T\left(n_{1}, n_{2}\right)$ is a function of image coordinates in general, it is also possible to use a predetermined fixed value for it. One choice is to set $T\left(n_{1}, n_{2}\right)$ to zero for all $\left(n_{1}, n_{2}\right)$, which is result of the high-correlation assumption. Later in the paper, we also discuss how to choose a nonuniform threshold.

## D. Alternating Projections Algorithm

This section presents an alternating-projections algorithm to reconstruct the red and blue channels. We first derive the projec-
tion operators corresponding to the "observation" and "detail" constraint sets given in the Section II-C. Convergence issues and enhancement of the green channel are then addressed. Finally, the complete algorithm is presented.

1) Projection Operators: The first constraint set that is used in the reconstruction is the "observation" constraint set given in (2). Referring to that equation, we can write the projection $P_{o}[\cdot]$ onto the "observation" constraint set as follows:

$$
P_{o}\left[S\left(n_{1}, n_{2}\right)\right]=\left\{\begin{array}{c}
O\left(n_{1}, n_{2}\right),\left(n_{1}, n_{2}\right) \in \Lambda_{S}  \tag{9}\\
S\left(n_{1}, n_{2}\right), \text { otherwise }
\end{array}\right\}
$$

where $S$ is the color channel, which can be the red $(R)$, green $(G)$, or blue $(B)$ channel.

The other constraint set is the "detail" constraint set given in (8). In order to write the projection onto this constraint set, we first need to define the filtering operations in the synthesis stage of the filter bank. Letting $g_{0}(\cdot)$ and $g_{1}(\cdot)$ denote the impulse responses corresponding to $G_{0}(z)$ and $G_{1}(z)$, we can write the four filtering operations on a two-dimensional signal $X\left(n_{1}, n_{2}\right)$ as follows:

$$
\begin{align*}
& \left(\mathcal{U}_{1} X\right)\left(n_{1}, n_{2}\right)=g_{0}\left(n_{1}\right) *\left[g_{0}\left(n_{2}\right) * X\left(n_{1}, n_{2}\right)\right]  \tag{10}\\
& \left(\mathcal{U}_{2} X\right)\left(n_{1}, n_{2}\right)=g_{1}\left(n_{1}\right) *\left[g_{0}\left(n_{2}\right) * X\left(n_{1}, n_{2}\right)\right]  \tag{11}\\
& \left(\mathcal{U}_{3} X\right)\left(n_{1}, n_{2}\right)=g_{0}\left(n_{1}\right) *\left[g_{1}\left(n_{2}\right) * X\left(n_{1}, n_{2}\right)\right]  \tag{12}\\
& \left(\mathcal{U}_{4} X\right)\left(n_{1}, n_{2}\right)=g_{1}\left(n_{1}\right) *\left[g_{1}\left(n_{2}\right) * X\left(n_{1}, n_{2}\right)\right] \tag{13}
\end{align*}
$$

where $\mathcal{U}_{1}, \mathcal{U}_{2}, \mathcal{U}_{3}, \mathcal{U}_{4}$ are the synthesis filtering operators. As stated earlier, these form a perfect reconstruction filter bank with the analysis filtering operators $\mathcal{W}_{1}, \mathcal{W}_{2}, \mathcal{W}_{3}$, and $\mathcal{W}_{4}$

$$
\begin{align*}
S\left(n_{1}, n_{2}\right)= & \mathcal{U}_{1}\left(\mathcal{W}_{1} S\right)\left(n_{1}, n_{2}\right)+\mathcal{U}_{2}\left(\mathcal{W}_{2} S\right)\left(n_{1}, n_{2}\right) \\
& +\mathcal{U}_{3}\left(\mathcal{W}_{3} S\right)\left(n_{1}, n_{2}\right)+\mathcal{U}_{4}\left(\mathcal{W}_{4} S\right)\left(n_{1}, n_{2}\right) \tag{14}
\end{align*}
$$

$$
C_{d}=\left\{\begin{array}{c}
S\left(n_{1}, n_{2}\right):\left|\left(\mathcal{W}_{k} S\right)\left(n_{1}, n_{2}\right)-\left(\mathcal{W}_{k} G\right)\left(n_{1}, n_{2}\right)\right| \leq T\left(n_{1}, n_{2}\right)  \tag{8}\\
\forall\left(n_{1}, n_{2}\right), \text { for } k=2,3,4 \text { and } S=R, B
\end{array}\right\}
$$

TABLE III
Mean Square Error Comparison of Different Methods

| $\begin{gathered} \text { Image } \\ \text { no } \\ \hline \end{gathered}$ | Channel | Mean square error for different methods |  |  |  |  |  |  |  |  |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
|  |  | $\begin{aligned} & \hline \text { Bilinear } \\ & \text { interp. } \\ & \hline \end{aligned}$ | $\begin{gathered} \text { Method in } \\ {[2]} \\ \hline \end{gathered}$ | $\begin{gathered} \text { Method in } \\ {[6]} \end{gathered}$ | $\begin{gathered} \text { Method in } \\ {[5]} \end{gathered}$ | Method in [9] | $\begin{gathered} \text { Method in } \\ \text { [11] } \end{gathered}$ | $\begin{gathered} \text { Method in } \\ {[10]} \end{gathered}$ | $\begin{gathered} \text { Proposed } \\ (1-L) \end{gathered}$ | $\begin{gathered} \hline \text { Propased } \\ (2-L) \\ \hline \end{gathered}$ |
| 1 | Red | 187.9089 | 92.8195 | 53.5751 | 84.3346 | 51.9910 | 71.9113 | 28.6331 | 11.1537 | 11.9266 |
|  | Green | 72.1162 | 72.1162 | 46.3505 | 67.2416 | 21.4822 | 15.7100 | 20.0978 | 7.1227 | 7.1227 |
|  | Blue | 205.0097 | 94.5357 | 52.9721 | 83.4813 | 53.6638 | 82.8315 | 32.1395 | 11.6294 | 10.0408 |
| 2 | Red | 41.7608 | 51.5908 | 17.7585 | 76.5047 | 16.6541 | 17.9117 | 59.2494 | 8.0177 | 13.0637 |
|  | Green | 15.2170 | 15.2170 | 11.8072 | 15.7971 | 5.2777 | 4.7003 | 9.1604 | 8.7961 | 8.7961 |
|  | Blue | 38.6815 | 20.9121 | 13.4500 | 23.5842 | 12.1303 | 13.2157 | 12.9478 | 7.7456 | 7.7202 |
| 3 | Red | 169.8735 | 96.8888 | 48.4138 | 56.1798 | 42.2292 | 37.5406 | 57.9294 | 10.4808 | 14.8410 |
|  | Green | 75.3730 | 75.3730 | 50.7147 | 67.0559 | 16.2820 | 17.8096 | 73.2381 | 8.6595 | 8.6595 |
|  | Blue | 169.5238 | 79.6867 | 51.0465 | 95.3246 | 42.7336 | 41.1926 | 32.5129 | 17.6556 | 23.7133 |
| 4 | Red | 142.3374 | 66.2179 | 36.4665 | 60.6662 | 37.4170 | 50.8664 | 21.4296 | 8.9276 | 8.2246 |
|  | Green | 51.6217 | 51.6217 | 32.5744 | 51.8792 | 15.5885 | 12.4186 | 16.2496 | 5.6919 | 5.6919 |
|  | Blue | 135.0159 | 69.5935 | 39.9058 | 63.5090 | 38.0842 | 48.8259 | 29.3244 | 11.2759 | 11.1623 |
| 5 | Red | 36.1627 | 17.8795 | 9.7105 | 16.5978 | 9.8012 | 9.0955 | 12.5014 | 3.6303 | 6.2425 |
|  | Green | 14.7244 | 14.7244 | 9.3786 | 12.3191 | 3.8815 | 3.9935 | 29.4243 | 3.6542 | 3.6542 |
|  | Blue | 36.9836 | 18.7047 | 10.8722 | 22.8823 | 10.5688 | 9.1739 | 11.5855 | 6.5624 | 9.3744 |
| 6 | Red | 361.1668 | 157.4050 | 71.2011 | 116.3806 | 84.7959 | 164.4306 | 40.5317 | 18.4496 | 19.9498 |
|  | Green | 117.5322 | 117.5322 | 58.9497 | 87.1483 | 28.4313 | 25.5357 | 26.8525 | 11.3828 | 11.3828 |
|  | Blue | 365.6230 | 162.0675 | 73.1896 | 112.6440 | 86.2732 | 169.9609 | 44.7260 | 22.8969 | 23.1047 |
| 7 | Red | 46.2915 | 21.4934 | 11.4020 | 16.7672 | 11.6370 | 14.3500 | 7.38577 | 3.5461 | 4.3098 |
|  | Green | 17.5777 | 17.5777 | 10.2293 | 14.3326 | 4.4693 | 4.2732 | 7.41351 | 3.3418 | 3.3418 |
|  | Blue | 45.6932 | 21.7906 | 12.0721 | 18.6969 | 11.8510 | 15.4414 | 7.6779 | 6.7670 | 9.4430 |
| 8 | Red | 46.0860 | 23.4119 | 12.9790 | 19.7443 | 11.7921 | 12.3641 | 8.36021 | 3.8132 | 4.8498 |
|  | Green | 19.4041 | 19.4041 | 11.9138 | 16.3728 | 4.6615 | 4.0277 | 4.6207 | 3.6606 | 3.6606 |
|  | Blue | 49.2973 | 23.0224 | 14.6964 | 20.0613 | 13.4102 | 13.7024 | 7.66359 | 5.6762 | 6.6428 |
| 9 | Red | 97.2040 | 50.5133 | 29.8320 | 53.8813 | 28.6785 | 30.6635 | 23.2256 | 8.1776 | 10.2030 |
|  | Green | 38.2468 | 38.2468 | 27.0450 | 37.7190 | 12.0979 | 9.9917 | 22.5401 | 5.8512 | 5.8512 |
|  | Blue | 93.8299 | 47.5779 | 29.2492 | 51.1127 | 26.7295 | 28.6749 | 20.8198 | 8.9623 | 8.9076 |
| 10 | Red | 40.9047 | 22.1820 | 10.4737 | 16.9456 | 10.8859 | 14.5784 | 7.15264 | 3.3880 | 4.7873 |
|  | Green | 15.0617 | 15.0617 | 8.1748 | 12.6545 | 3.6963 | 3.4156 | 3.86487 | 3.1580 | 3.1580 |
|  | Blue | 44.3114 | 20.7030 | 10.4823 | 15.5438 | 11.1874 | 15.5967 | 7.31925 | 4.3254 | 4.8950 |
| 11 | Red | 317.2656 | 144.7688 | 124.4631 | 170.9637 | 107.9393 | 80.2820 | 55.6570 | 23.3247 | 17.5040 |
|  | Green | 144.0955 | 144.0955 | 121.6625 | 163.5400 | 54.1035 | 39.2386 | 99.1851 | 15.0249 | 15.0249 |
|  | Blue | 324.3623 | 167.3950 | 135.8525 | 201.6013 | 115.3881 | 88.3770 | 91.6544 | 31.0839 | 28.2410 |
| 12 | Red | 54.0245 | 55.5710 | 22.9565 | 67.4766 | 21.3817 | 22.0560 | 43.5004 | 8.4860 | 13.5237 |
|  | Green | 21.8710 | 21.8710 | 15.1751 | 18.9863 | 7.2481 | 5.5038 | 7.61045 | 8.1075 | 8.1075 |
|  | Blue | 62.3868 | 26.2786 | 17.5949 | 24.3682 | 18.1835 | 22.4072 | 9.02726 | 7.8151 | 8.0626 |
| 13 | Red | 60.1798 | 29.4225 | 16.1500 | 28.2998 | 16.5090 | 23.0927 | 9.51217 | 3.9657 | 3.5305 |
|  | Green | 21.7635 | 21.7635 | 13.5537 | 22.6938 | 6.8096 | 5.6587 | 8.15223 | 3.0419 | 3.0419 |
|  | Blue | 60.6834 | 30.7311 | 16.7104 | 28.0346 | 16.6368 | 22.5951 | 13.2754 | 5.2673 | 5.3687 |
| 14 | Red | 46.7877 | 23.1555 | 16.1704 | 34.7360 | 14.0212 | 11.5421 | 10.2657 | 4.6324 | 4.4979 |
|  | Green | 22.7296 | 22.7296 | 17.4425 | 22.1423 | 7.2229 | 6.3324 | 17.2543 | 3.7388 | 3.7388 |
|  | Blue | 54.5740 | 26.9042 | 19.2051 | 29.9782 | 17.3880 | 14.0801 | 14.4194 | 7.0185 | 7.4754 |
| 15 | Red | 122.7638 | 60.7294 | 46.9059 | 75.6976 | 40.2998 | 30.7597 | 30.8517 | 11.8054 | 12.9477 |
|  | Green | 60.4622 | 60.4622 | 47.8300 | 64.2247 | 21.0464 | 16.4279 | 51.0574 | 10.1858 | 10.1858 |
|  | Blue | 130.1658 | 69.5558 | 54.5523 | 92.7093 | 47.4073 | 38.2464 | 38.8936 | 17.5744 | 19.8079 |
| 16 | Red | 128.8738 | 56.5123 | 22.7103 | 37.2010 | 28.2224 | 63.8097 | 13.4602 | 6.4493 | 6.6832 |
|  | Green | 44.3272 | 44.3272 | 20.6299 | 29.5803 | 9.2259 | 8.8671 | 14.9775 | 5.2919 | 5.2919 |
|  | Blue | 124.8694 | 62.6212 | 23.8586 | 45.8549 | 27.3332 | 64.5105 | 22.1372 | 8.6431 | 9.7768 |
| 17 | Red | 61.4497 | 33.4027 | 15.6353 | 21.9867 | 16.0100 | 12.9693 | 8.86245 | 4.9608 | 5.2015 |
|  | Green | 30.3986 | 30.3986 | 15.3200 | 19.8501 | 6.8153 | 5.3587 | 15.1647 | 7.1316 | 7.1316 |
|  | Blue | 56.9281 | 30.2105 | 19.4766 | 30.4042 | 18.6600 | 16.5535 | 13.3502 | 9.2426 | 11.0392 |
| 18 | Red | 111.0566 | 51.8098 | 34.6219 | 55.6550 | 32.4667 | 33.7863 | 15.7315 | 8.0718 | 7.5613 |
|  | Green | 45.4554 | 45.4554 | 32.4018 | 46.3240 | 14.6395 | 10.8145 | 16.4300 | 6.2735 | 6.2735 |
|  | Blue | 113.3915 | 57.0697 | 38.2285 | 57.9721 | 34.6850 | 38.6562 | 24.5898 | 11.3049 | 11.7922 |
| 19 | Red | 65.9768 | 35.4940 | 24.8584 | 38.9006 | 22.2870 | 23.2553 | 16.1701 | 13.4359 | 15.8164 |
|  | Green | 30.1957 | 30.1957 | 21.9572 | 29.6137 | 10.3009 | 9.3900 | 11.3859 | 8.3866 | 8.3866 |
|  | Blue | 76.1882 | 38.1415 | 26.2763 | 37.0819 | 25.0622 | 32.0137 | 20.0780 | 14.2484 | 16.9749 |
| 20 | Red | 149.2869 | 70.5418 | 61.5271 | 83.2984 | 49.3905 | 40.4256 | 29.0343 | 25.7426 | 25.8744 |
|  | Green | 76.3340 | 76.3340 | 64.1117 | 83.3698 | 28.5436 | 22.3596 | 26.1093 | 17.7896 | 17.7896 |
|  | Blue | 192.4042 | 99.6933 | 85.4603 | 119.5614 | 78.4803 | 64.6248 | 50.7653 | 32.8394 | 32.1965 |

Now, we can write the projection $P_{d}\left[S\left(n_{1}, n_{2}\right)\right]$ of a color channel $S\left(n_{1}, n_{2}\right)$ onto the "detail" constraint set $C_{d}$ as follows. Referring to (8), we define $r_{k}\left(n_{1}, n_{2}\right)$ as the residual

$$
\begin{equation*}
r_{k}\left(n_{1}, n_{2}\right)=\left(\mathcal{W}_{k} S\right)\left(n_{1}, n_{2}\right)-\left(\mathcal{W}_{k} G\right)\left(n_{1}, n_{2}\right) \tag{15}
\end{equation*}
$$

When this residual is less than the threshold $T\left(n_{1}, n_{2}\right)$ in magnitude, the subband value $\left(\mathcal{W}_{k} S\right)\left(n_{1}, n_{2}\right)$ is not changed.

Otherwise, it has to be changed so that the residual $r_{k}\left(n_{1}, n_{2}\right)$ is less $T\left(n_{1}, n_{2}\right)$ in magnitude. This projection operator can be written as

$$
\begin{equation*}
P_{d}\left[S\left(n_{1}, n_{2}\right)\right]=\mathcal{U}_{1}\left(\mathcal{W}_{1} S\right)\left(n_{1}, n_{2}\right)+\sum_{k=2}^{4} \mathcal{U}_{k}\left(\mathcal{W}_{k}^{\prime} S\right)\left(n_{1}, n_{2}\right) \tag{16}
\end{equation*}
$$

where (see (17) at the bottom of the page). The "observation" projection ensures that the interpolated channels are consistent with the observed data; the "detail" projection reconstructs the high-frequency information of the red and blue channels, and imposes edge consistency between the channels. By alternately applying these two projections onto the initial red and blue channel estimates, we are able to enhance these channels.
2) Convergence: The Constraint sets given in (2) and (8) are convex. (The proofs are provided in the Appendix.) Therefore, an initial estimate converges to a solution in the feasibility set by projecting it onto these constraint sets iteratively. We have also verified it experimentally. Using the proposed algorithm, we updated the chrominance (red and blue) channels iteratively from the initial estimates. In each iteration, the chrominance channels are updated by the "detail" projection, followed by the "observation" projection. A typical convergence plot is given in Fig. 5. As seen in that figure, the mean square error of the red and blue channels converges in about five iterations. (That plot is for Image 16 in Fig. 2. The initial estimates for the red and blue channels were obtained by bilinear interpolation. The green channel was interpolated using a method that will be explained in Section II-D3.)

Instead of performing a one-level subband decomposition, it is also possible to decompose the signals further. As done with undecimated wavelet transforms, the low-pass $(L L)$ subbands can be decomposed by using filters $H_{0}\left(z^{2}\right)$, and $H_{1}\left(z^{2}\right)$. This filterbank structure is shown in Fig. 6 for a two-level decomposition. Convergence for the two-level decomposition, which is illustrated in Fig. 7 is faster than for the one-level decomposition.
3) Updating the Green Channel: The algorithm we have discussed so far reconstructs the high-frequency information of the red and blue channels. The performance of this reconstruction directly depends on the accuracy of the green channel interpolation. The edge-directed interpolation methods discussed in Section I provide satisfactory performance in general, but it is still possible to obtain better results using a method similar to the red-blue interpolation we have presented. Referring to Fig. 8, we can update the green channel as follows.

1) Interpolate the green channel to get an initial estimate. Either bilinear or edge-directed interpolation methods can be used for this step.
2) Use the observed samples of the blue channel to form a downsampled version of the blue channel. Note that all pixels of this downsampled image are observed data.
3) Use the interpolated green samples at the corresponding (blue) locations to form a downsampled version of the green channel. Note that the pixels of this downsampled image are all interpolated values.
4) Decompose these blue and green downsampled channels into their subbands, as was done in Section II-C.
5) Replace the high-frequency ( $L H, H L, H H$ ) subbands of the green channel with those of the blue channel. (Note that this corresponds to setting the threshold $T\left(n_{1}, n_{2}\right)$ to zero.)
6) Reconstruct the downsampled green channel, and insert the pixels in their corresponding locations in the initial green channel estimate.
7) Repeat the same procedure for the pixels at the red samples.
With this scheme, significant improvement over bilinear interpolation and other adaptive algorithms can be achieved in the green channel. We used the edge-directed interpolation procedure proposed in [9] to obtain the initial green channel estimates. The results are discussed in Section III.
8) Complete Algorithm: The pseudo-code of the complete algorithm is as follows.
9) Initial interpolation: Interpolate the red, green, and blue channels to obtain initial estimates. Bilinear or edgedirected interpolation algorithms can be used for this initial interpolation.
10) Update the green channel: Update the green channel using the scheme explained in Section II-D3.
11) "Detail" projection: Decompose all three channels with a filter bank. At each level of decomposition, there will be four subbands. Update the detail (high-frequency) subbands of the red and blue channels using (17) and reconstruct these channels using (16).
12) "Observation" projection: Compare the samples of the reconstructed red and blue channels with the original (observed) samples. Insert the observed samples into the reconstructed channels at their corresponding pixel locations as given in (9).
13) Iteration: Go to Step 3, and repeat the procedure until a stopping criterion is achieved.

$$
\left(\mathcal{W}_{k}^{\prime} S\right)\left(n_{1}, n_{2}\right)=\left\{\begin{array}{c}
\left(\mathcal{W}_{k} G+T\right)\left(n_{1}, n_{2}\right) ; r_{k}\left(n_{1}, n_{2}\right)>T\left(n_{1}, n_{2}\right)  \tag{17}\\
\left(\mathcal{W}_{k} S\right)\left(n_{1}, n_{2}\right) ;\left|r_{k}\left(n_{1}, n_{2}\right)\right| \leq T\left(n_{1}, n_{2}\right) \\
\left(\mathcal{W}_{k} G-T\right)\left(n_{1}, n_{2}\right) ; r_{k}\left(n_{1}, n_{2}\right)<-T\left(n_{1}, n_{2}\right)
\end{array}\right\}
$$

$$
\begin{equation*}
K_{S}\left(n_{1}, n_{2}\right)=\frac{\sum_{(i, j) \in N_{\left(n_{1}, n_{2}\right)}}\left(S(i, j)-\mu_{S}\right)\left(G(i, j)-\mu_{G}\right)}{\sqrt{(i, j) \in N_{\left(n_{1}, n_{2}\right)}\left(S(i, j)-\mu_{S}\right)^{2}} \sqrt{\sum_{(i, j) \in N_{\left(n_{1}, n_{2}\right)}}\left(G(i, j)-\mu_{G}\right)^{2}}} \tag{18}
\end{equation*}
$$



Fig. 9. Comparison of the methods for Image 4. (a) Crop from the original Image 4. (b) Method in [2]. (c) Method in [5]. (d) Method in [6].

## E. Extensions to the Algorithm

It is also possible to extend the proposed algorithm in several different ways.

1) Correlation surface: The threshold $T\left(n_{1}, n_{2}\right)$ in the "detail" projection provides a way of controlling the
amount of the correlation between the channels that is used by the algorithm. If the channels are totally uncorrelated the threshold should be large enough to turn the "detail" projection into an identity projection. If the channels are highly correlated the threshold should be close to zero. One problem, however, is that the correlation between


Fig. 9. (Continued.) Comparison of the methods for Image 4. (e) Method in [9]. (f) Method in [10]. (g) Method in [11]. (h) Proposed (1-L, 8 iterations).
the channels is not necessarily uniform; there may be both high-correlation and low-correlation regions within the same image. This can be overcome by estimating the correlation locally and adjusting the threshold $T\left(n_{1}, n_{2}\right)$ accordingly. One way to compute a local correlation surface is to move a small window over the color
planes, compute the correlation between them, and assign a correlation coefficient to the pixel at the center of the window. By mapping the values on the correlation surface to the threshold $T\left(n_{1}, n_{2}\right)$, the algorithm can be made more effective for images that have nonuniform correlation surfaces.

(a) Crop from the original Image 6.

(c) Method in [5].

(b) Method in [2].

(d) Method in [6].

Fig. 10. Comparison of the methods for Image 6. (a) Crop from the original Image 6. (b) Method in [2]. (c) Method in [5]. (d) Method in [6].

Denoting $K_{S}\left(n_{1}, n_{2}\right)$ as the correlation surface between channel $S$-red or blue-and the green channel, the proposed method computes the correlation surface as shown in (18) at the bottom of page 1004, where $N_{\left(n_{1}, n_{2}\right)}$ is a neighborhood about location $\left(n_{1}, n_{2}\right)$, and $\mu_{S}$ and $\mu_{G}$ are the means of channels $S$ and $G$ in that neighborhood. One choice for $N_{\left(n_{1}, n_{2}\right)}$ might be a $5 \times 5$
window. This formula will give a correlation surface with values ranging between zero and one. This correlation surface is then passed to a function that will return a large value when $K_{S}\left(n_{1}, n_{2}\right)$ is small and a small value when $K_{S}\left(n_{1}, n_{2}\right)$ is large. The choice of such a function requires further research and experimentation, and we leave it as an open problem.


Fig. 10. (Continued.) Comparison of the methods for Image 6. (e) Method in [9]. (f) Method in [10]. (g) Method in [11]. (h) Proposed (1-L, 8 iterations).
2) Smoothness projection: Other constraint sets can be included in the algorithm easily. One such constraint is a smoothness constraint. Smooth hue (color ratio) and smooth color difference transitions are the basis of some demosaicing algorithms that we have already cited [2]-[4].

An easy way to include a smoothness projection is to interpolate the color ratio or difference to get an estimated color value at a certain location ( $n_{1}, n_{2}$ ), and constrain the results to lie in a certain neighborhood of that estimate. This is also an open area that should be investigated.

(a) Crop from the original Image 16.

(c) Method in [5].

(b) Method in [2].

(d) Method in [6].

Fig. 11. Comparison of the methods for Image 16. (a) Crop from the original Image 16. (b) Method in [2]. (c) Method in [5]. (d) Method in [6].

## III. EXPERIMENTAL RESULTS

In our experiments, we used the images shown in Fig. 2. These images are film captures and digitized with photo scanner. Full color channels are available, and the CFA is simulated by sampling the channels. The sampled channels are used to test the demosaicing algorithms.

We used bilinear interpolation for the red and blue channels, and the edge-directed interpolation method given in [9] for the green channel to get the initial estimates. The method proposed in Section II-D3 was used to refine the initial estimate of the green channel. The following filters were used in the experiments: $h_{0}=\left[\begin{array}{lll}1 & 2 & 1\end{array}\right] / 4$;


Fig. 11. (Continued.) Comparison of the methods for Image 16. (e) Method in [9]. (f) Method in [10]. (g) Method in [11]. (h) Proposed (1-L, 8 iterations).
$h_{1}=\left[\begin{array}{ccc}1 & -2 & 1\end{array}\right] / 4 ; g_{0}=\left[\begin{array}{ccccc}-1 & 2 & 6 & 2 & -1\end{array}\right] / 8$; and $g_{1}=\left[\begin{array}{ccccc}1 & 2 & -6 & 2 & 1\end{array}\right] / 8$. The threshold $T\left(n_{1}, n_{2}\right)$ was set to zero for all $\left(n_{1}, n_{2}\right)$. We did the experiments for both one-level decomposition and two-level decomposition. The number of iterations for one-level (1-L) and two-level (2-L) decompositions was eight and four, respectively. The perfor-
mance in terms of mean squared error can be seen in Table III for both our and various other demosaicing algorithms [2], [5], [6], [9]-[11]. As seen in that table, the proposed algorithm has the lowest mean squared error in almost all cases. Among these algorithms, [9] and [11] have comparable performance in the green channel for some images. (The ones whose performance
was better than the proposed method are highlighted.) However, their red and blue channel performance was worse in all cases, which make them worse visually. Another successful method was Kimmel's method [10]. In that paper the red, green, and blue channels were corrected iteratively to satisfy the color ratio rule, and the number of iterations was set to three. However, we found that algorithm to be prone to color artifacts, and iterating three times made the results worse in such cases. Therefore, in our implementation we did color correction only once.
We also provide some examples from the images used in the experiments for visual comparison. Fig. 9-11 show cropped segments from original images (Images 4, 6, and 16 in Fig. 2), and the corresponding reconstructed images from the demosaicing algorithms that were used in comparison. Close examination of those figures verifies the effectiveness of the proposed algorithm.

## IV. Complexity Analysis

Let $l_{h_{0}}, l_{h_{1}}, l_{g_{0}}$, and $l_{g_{1}}$ denote the lengths of the filters $h_{0}$, $h_{1}, g_{0}$, and $g_{1}$, respectively, and let $M$ and $N$ denote the width and height of an image. Each channel is decomposed into four subbands by convolving its rows and columns with filters $h_{0}$ and $h_{1}$. This requires approximately $\left(2 l_{h_{0}}+4 l_{h_{1}}\right) M N$ multiplications and additions for each channel. Including the reconstruction stage the total number of additions and multiplications is $\left[2\left(l_{h_{0}}+l_{g_{0}}\right)+4\left(l_{h_{1}}+l_{g_{1}}\right)\right] M N$ for each channel. Typically, three iterations is enough for updating the red and blue channels, which will require a total of $\left[12\left(l_{h_{0}}+l_{g_{0}}\right)+24\left(l_{h_{1}}+l_{g_{1}}\right)\right] M N$ operations for the red and blue channels. As a result, $\left[14\left(l_{h_{0}}+l_{g_{0}}\right)+28\left(l_{h_{1}}+l_{g_{1}}\right)\right] M N$ operations are required for the iteration stages. We also update the initial estimate of the green channel as proposed in Section II-D with a one-level decomposition, and one iteration. This adds approximately $\left[2\left(l_{h_{0}}+l_{g_{0}}\right)+4\left(l_{h_{1}}+l_{g_{1}}\right)\right] M N$ operations to the total count, which brings the total operation count to $\left[16\left(l_{h_{0}}+l_{g_{0}}\right)+32\left(l_{h_{1}}+l_{g_{1}}\right)\right] M N$. For the filters used in the experiment, this number is $384 M N$. If four iterations are done, the total complexity is $480 M N$. If a two-level decomposition is performed, a single iteration should be sufficient. Under this assumption using the filters in this paper, the total complexity for a two-level decomposition is also $384 M N$ additions and multiplications.

## V. Conclusion

In this paper we presented a demosaicing algorithm that exploits inter-channel correlation in an alternating projections scheme. Two constraint sets are defined based on the observed data and the prior knowledge about the correlation of the channels, and initial estimates are projected onto these constraint sets to reconstruct the channels. The proposed algorithm was compared with well-known demosaicing algorithms, and it showed an outstanding performance both visually and in terms of mean square error at a reasonable computational complexity.
The question of uncorrelated color channels has also been addressed, and a threshold selection procedure has been pro-
posed. However, in the experiments this was not needed, and setting the threshold to zero worked very well. Threshold selection and inclusion of other constraint sets are left as future work. It should also be noted the test images used are film captures that were digitized with photo scanner. Therefore, they have different noise power spectrums compared to actual digital camera captures, and more thorough performance analysis of the demosaicing algorithm should be done for different capture and digitization paths.

## APPENDIX

## Convexity of the Constraint Sets

We outline the convexity proofs of the observation and detail constraint sets that are given in (2) and (8), respectively.

## A. Observation Constraint Set $C_{o}$

Let $S_{1}\left(n_{1}, n_{2}\right)$ and $S_{2}\left(n_{1}, n_{2}\right)$ be any two points in the set $C_{0}$. That is,

$$
\begin{equation*}
S_{1}\left(n_{1}, n_{2}\right)=O\left(n_{1}, n_{2}\right) \quad \forall\left(n_{1}, n_{2}\right) \in \Lambda_{S} \tag{19}
\end{equation*}
$$

and

$$
\begin{equation*}
S_{2}\left(n_{1}, n_{2}\right)=O\left(n_{1}, n_{2}\right) \quad \forall\left(n_{1}, n_{2}\right) \in \Lambda_{S} \tag{20}
\end{equation*}
$$

For convexity, we need to show that all points of the line segment connecting $S_{1}\left(n_{1}, n_{2}\right)$ and $S_{2}\left(n_{1}, n_{2}\right)$ remain in the set $C_{o}$. Let $S_{3}\left(n_{1}, n_{2}\right) \equiv \alpha S_{1}\left(n_{1}, n_{2}\right)+(1-\alpha) S_{2}\left(n_{1}, n_{2}\right)$ be this line segment $(0 \leq \alpha \leq 1)$. Using (19) and (20), we get:

$$
\begin{align*}
S_{3}\left(n_{1}, n_{2}\right)= & \alpha S_{1}\left(n_{1}, n_{2}\right)+(1-\alpha) S_{2}\left(n_{1}, n_{2}\right) \\
= & \alpha O\left(n_{1}, n_{2}\right)+(1-\alpha) O\left(n_{1}, n_{2}\right) \\
& \forall\left(n_{1}, n_{2}\right) \in \Lambda_{S} \\
= & O\left(n_{1}, n_{2}\right), \quad \forall\left(n_{1}, n_{2}\right) \in \Lambda_{S} \tag{21}
\end{align*}
$$

That is, $S_{3}\left(n_{1}, n_{2}\right) \in C_{o}$.

## B. Detail Constraint Set $C_{d}$

Let $S_{1}\left(n_{1}, n_{2}\right)$ and $S_{2}\left(n_{1}, n_{2}\right)$ be any two points in the set $C_{d}$. Referring to (5)-(7) in the manuscript, we can write

$$
\begin{align*}
\mid h_{x}\left(n_{1}\right) *\left[h_{y}\left(n_{2}\right) * S_{1}\left(n_{1}, n_{2}\right)\right]-\left(\mathcal{W}_{k} G\right) & \left(n_{1}, n_{2}\right) \mid \\
& \leq T\left(n_{1}, n_{2}\right) \tag{22}
\end{align*}
$$

and

$$
\begin{align*}
\mid h_{x}\left(n_{1}\right) *\left[h_{y}\left(n_{2}\right) * S_{2}\left(n_{1}, n_{2}\right)\right]-\left(\mathcal{W}_{k} G\right) & \left(n_{1}, n_{2}\right) \mid \\
& \leq T\left(n_{1}, n_{2}\right) \tag{23}
\end{align*}
$$

where the subscripts $x$ and $y$ are chosen according to the value of $k$ as in (5)-(7).

Again, we need to show that all points of the line segment connecting $S_{1}\left(n_{1}, n_{2}\right)$ and $S_{2}\left(n_{1}, n_{2}\right)$ remain in the set $C_{d}$, for convexity.

Let $S_{3}\left(n_{1}, n_{2}\right) \equiv \alpha S_{1}\left(n_{1}, n_{2}\right)+(1-\alpha) S_{2}\left(n_{1}, n_{2}\right)$ for $0 \leq \alpha \leq 1$. We will now show that $S_{3}\left(n_{1}, n_{2}\right)$ is in $C_{d}$. We will omit the indices $n_{1}$ and $n_{2}$ in the notation to simplify the
equations. See (24) at the bottom of the page. Add and subtract $\alpha\left(\mathcal{W}_{k} G\right)$ inside the (24) and regroup the terms to get (see (25) at the bottom of the page). Use the triangular inequality and the inequalities given in (22) and (23) to get

$$
\begin{align*}
\left|h_{x} *\left[h_{y} * S_{3}\right]-\left(W_{k} G\right)\right| \leq & \alpha\left|\left\{h_{x} *\left[h_{y} * S_{1}\right]-\left(W_{k} G\right)\right\}\right| \\
& +(1-\alpha) \\
& \times\left|\left\{h_{x} *\left[h_{y} * S_{2}\right]-\left(W_{k} G\right)\right\}\right| \\
\leq & \alpha T+(1-\alpha) T \\
= & T . \tag{26}
\end{align*}
$$

Therefore, $S_{3}\left(n_{1}, n_{2}\right) \in C_{d}$.

## Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments, and J. Glotzbach and Dr. R. W. Schafer for providing the test images.

## References

[1] B. E. Bayer, "Color imaging array," U.S. Patent 3971 065, July 1976.
[2] D. R. Cok, "Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal," U.S. Patent 4642 678, Feb. 1987.
[3] J. A. Weldy, "Optimized design for a single-sensor color electronic camera system," Proc. SPIE, vol. 1071, pp. 300-307, 1988.
[4] J. E. Adams Jr., "Interactions between color plane interpolation and other image processing functions in electronic photography," Proc. SPIE, vol. 2416, pp. 144-151, Feb. 1995.
[5] R. H. Hibbard, "Apparatus and method for adaptively interpolating a full color image utilizing luminance gradients," U.S. Patent 5382 976, Jan. 1995.
[6] C. A. Laroche and M. A. Prescott, "Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients," U.S. Patent 5373 322, Dec. 1994.
[7] J. E. Adams and J. F. Hamilton Jr., "Adaptive color plane interpolation in single color electronic camera," U.S. Patent 5 506 619, Apr. 1996.
[8] J. E. Adams Jr., "Design of practical color filter array interpolation algorithms for digital cameras," Proc. SPIE, vol. 3028, pp. 117-125, Feb. 1997.
[9] J. F. Hamilton Jr. and J. E. Adams, "Adaptive color plane interpolation in single sensor color electronic camera," U.S. Patent 5 629734, May 1997.
[10] R. Kimmel, "Demosaicing: Image reconstruction from CCD samples," IEEE Trans. Image Processing, vol. 8, pp. 1221-1228, 1999.
[11] E. Chang, S. Cheung, and D. Y. Pan, "Color filter array recovery using a threshold-based variable number of gradients," Proc. SPIE, vol. 3650, pp. 36-43, 1999.
[12] X. Wu, W. K. Choi, and P. Bao, "Color restoration from digital camera data by pattern matching," Proc. SPIE, vol. 3018, pp. 12-17, 1997.
[13] D. R. Cok, "Signal processing method and apparatus for sampled image signals," U.S. Patent 4630307 , 1984.
[14] D. Taubman, "Generalized wiener reconstruction of images from color sensor data using a scale invariant prior," in IEEE Proc. Int. Conf. Image Processing, vol. 3, 2000, pp. 801-804.
[15] H. J. Trussell and R. E. Hartwig, "Mathematics for demosaicking," IEEE Trans. Image Processing, vol. 11, pp. 485-492, Apr. 2002.
[16] J. Mukherjee, R. Parthasarathi, and S. Goyal, "Markov random field processing for color demosaicing," Pattern Recognit. Lett., vol. 22, pp. 339-351, 2001.
[17] J. E. Adams Jr., "Design of color filter array interpolation algorithms for digital cameras, Part 2," in IEEE Proc. Int. Conf. Image Processing, vol. 1, 1998, pp. 488-492.
[18] J. W. Glotzbach, R. W. Schafer, and K. Illgner, "A method of color filter array interpolation with alias cancellation properties," in IEEE Proc. Int. Conf. Image Processing, vol. 1, 2001, pp. 141-144.
[19] P. L. Combettes, "The foundations of set theoretic estimation," Proc. IEEE, vol. 81, pp. 182-208, Feb. 1993.


Bahadir K. Gunturk (S'01) received the B.S. degree in electrical engineering from Bilkent University, Ankara, Turkey, in 1999, and the M.S. degree in electrical engineering from Georgia Institute of Technology (Georgia Tech), Atlanta, in 2001. He is currently pursuing his Ph .D. degree at Georgia Tech.

His research interests include image/video processing, multimedia communications, and computer vision.
Mr. Gunturk received the Outstanding Research Award from the Center for Signal and Image Processing, Georgia Tech, in 2001.

$$
\begin{align*}
\left|h_{x} *\left[h_{y} * S_{3}\right]-\left(\mathcal{W}_{k} G\right)\right| & =\left|h_{x} *\left[h_{y} *\left(\alpha S_{1}+(1-\alpha) S_{2}\right)\right]-\left(W_{k} G\right)\right| \\
& =\left|\alpha h_{x} *\left[h_{y} * S_{1}\right]+(1-\alpha) h_{x} *\left[h_{y} * S_{2}\right]-\left(\mathcal{W}_{k} G\right)\right| \tag{24}
\end{align*}
$$

$$
\begin{align*}
\left|h_{x} *\left[h_{y} * S_{3}\right]-\left(W_{k} G\right)\right| & =\left|\alpha h_{x} *\left[h_{y} * S_{1}\right]+(1-\alpha) h_{x} *\left[h_{y} * S_{2}\right]-\left(W_{k} G\right)+\alpha\left(W_{k} G\right)-\alpha\left(W_{k} G\right)\right| \\
& =\left|\alpha\left\{h_{x} *\left[h_{y} * S_{1}\right]-\left(W_{k} G\right)\right\}+(1-\alpha)\left\{h_{x} *\left[h_{y} * S_{2}\right]-\left(W_{k} G\right)\right\}\right| \tag{25}
\end{align*}
$$



Yucel Altunbasak (S'94-M'97-SM'01) received the B.S. degree from Bilkent University, Ankara, Turkey, in 1992 with highest honors. He received the M.S. and Ph.D. degrees from the University of Rochester, Rochester, NY, in 1993 and 1996, respectively. His Ph.D. research involved mesh-based video representation and coding.

He joined Hewlett-Packard Research Laboratories (HPL), Palo Alto, CA, in 1996. His position at HPL provided him with the opportunity to work on a diverse set of research topics, such as video processing, coding, and communications, multimedia streaming and networking, and inverse problems in signal processing. He also taught digital video and signal processing courses at Stanford University, Stanford, CA, and San Jose State University, San Jose, CA, as a consulting Assistant Professor. He joined the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, in 1999 as an Assistant Professor. His research agenda involves both telecommunications and DSP, particularly bridging these two fields. He is currently working on industrial- and government-sponsored projects related to media communications, networked video, and interactive video. His research interests include video and multimedia signal processing, inverse problems in imaging, and network distribution of compressed multimedia content. His research efforts have resulted in 24 journal papers, seven patents, five patent applications, and 45 conference publications. He is an area/associate editor for Signal Processing: Image Communications and for the Journal of Circuits, Systems, and Signal Processing. He will serve as a co-chair for the Advanced Signal Processing for Communications Symposia at ICC'03. He also serves as a session chair in technical conferences, as a panel reviewer for government funding agencies, and as a technical reviewer for various journals and conferences in the field of signal processing and communications.

Dr. Altunbasak is an area/associate editor for IEEE Transactions on Image Processing. He is a member of the IEEE Signal Processing Society's IMDSP Technical Committee. He received the National Science Foundation (NSF) CAREER Award in 2002.


Russell M. Mersereau (S'69-M'73-SM'78-F'83) received the S.B. and S.M. degrees in 1969 and the Sc.D. degree in 1973 from the Massachusetts Institute of Technology, Cambridge.

He joined the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, in 1975. His current research interests are in the development of algorithms for the enhancement, modeling, and coding of computerized images, synthesis aperture radar, and computer vision. In the past, this research has been directed to problems of distorted signals from partial information of those signals, computer image processing and coding, the effect of image coders on human perception of images, and applications of digital signal processing methods in speech processing, digital communications, and pattern recognition. He is the coauthor of the text Multidimensional Digital Signal Processing.
Dr. Mersereau has served on the editorial board of the Proceedings of the IEEE and as Associate Editor for signal processing of the IEEE Transactions on Acoustics, Speech, and Signal Processing and Signal Processing Letters. He is the corecipient of the 1976 Bowder J. Thompson Memorial Prize from the IEEE for the best technical paper by an author under the age of 30, a recipient of the 1977 Research Unit Award of the Southeastern Section of the ASEE, and three teaching awards. He was awarded the 1990 Society Award of the Signal Processing Society. He is currently the Vice President for Awards and Membership of the Signal Processing Society.


[^0]:    Manuscript received August 23, 2001; revised April 17, 2002. This work was supported in part by the Office of Naval Research (ONR) under Award N00014-01-1-0619 and by the National Science Foundation under Award CCR-0113681. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Mark S. Drew.
    B. K. Gunturk and R. M. Mersereau are with the Center for Signal and Image Processing, Georgia Institute of Technology, Atlanta, GA 30332-0250 USA (e-mail: bahadir@ece.gatech.edu; rmm@ece.gatech.edu).
    Y. Altunbasak is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0250 USA (e-mail: yucel@ece.gatech.edu).

    Publisher Item Identifier 10.1109/TIP.2002.801121.

[^1]:    ${ }^{1}$ Strictly speaking, the luminance response of the human visual system is different from the response to green light, but it is common to refer to the green channel as the luminance channel since its frequency is close to the peak of the human luminance frequency response.

