Demosaicing by Smoothing along 1D Features

Boris Ajdin, Matthias B. Hullin, Christian Fuchs, Hans-Peter Seidel, Hendrik P. A. Lensch
MPI Informatik
Saarbrücken Germany
{bajdin,hullin,cfuchs,hpseidel,lensch}@mpi-inf.mpg.de

Abstract

Most digital cameras capture color pictures in the form of an image mosaic, recording only one color channel at each pixel position. Therefore an interpolation algorithm needs to be applied to reconstruct the missing color information. In this paper we present a novel Bayer pattern demosaicing approach, employing stochastic global optimization performed on a pixel neighborhood. We are minimizing a newly developed cost function that increases smoothness along one-dimensional image features. While previous algorithms have been developed focusing on LDR images only, our optimization scheme and the underlying cost function are designed to handle both LDR and HDR images, creating less demosaicing artifacts, compared to previous approaches.

1. Introduction

Most digital cameras use a single-chip CCD or CMOS photo-sensitive sensor array to capture images. At each sensor position the intensity of light is measured by accumulating photoelectrons. In order to capture a color image, a color filter array (CFA) is placed in front of the sensor, permitting only light of a certain wavelength range (usually red, green or blue) to pass through. Hence, compared to the full RGB color image, two thirds of the color data is missing. The most frequently used CFA pattern in today's color cameras is the Bayer pattern [2]. The Bayer mosaic consists of red, green and blue pixels arranged in a regular grid, with twice the number of green pixels compared to the red and blue ones, as depicted in Figure 1(a).

Given a color mosaic recorded in this fashion, the missing data needs to be determined in order to obtain a full-color full-resolution image. This is done by a process called demosaicing, for which a number of algorithms exist. Most of them try to reconstruct the missing colors by interpolating within a small pixel neighborhood. Many of them, however, introduce artifacts into the image that are very unpleasant for human eyes, namely false colors and "zippering" (alternating patterns) along edges in the image, as shown in Figure 1. Other methods introduce blurring of the image which avoids creating these artifacts but nevertheless reduces the overall quality of the resulting image.

(a) (b) (c)

Figure 1. Artifacts arising from improper demosaicing of input images with a Bayer pattern CFA (a): false colors (b) and zippering (c).

Another restriction of most digital cameras is their limited capability of capturing the full dynamic range of natural scenes. The scene dynamic range is defined as the ratio of the largest and smallest luminance value present in the scene. This value often exceeds the dynamic range of the camera sensor. By taking multiple shots of the same scene with different exposure settings and combining them into one high dynamic range (HDR) image, we can compensate for the low dynamic range (LDR) of the camera sensor.

In this paper we will present a novel demosaicing algorithm which iteratively improves the image quality. Our main observation, concerning the problem of demosaicing, is that the search for a plausible reconstruction must not be performed considering just the pixel's local neighborhood in the CFA image. Instead, the reconstructed colors have to be consistent with the neighboring pixels colors in the final full-color image.

We have therefore developed a global optimization scheme which operates simultaneously on all pixels in a neighborhood, searching for the optimal pixel configuration in that neighborhood. In this manner we are able to produce images with drastically reduced artifacts, outperforming existing demosaicing algorithms for LDR images. The optimization is performed by minimizing our newly developed cost-function, which exploits the local pixel coherence in natural-looking images. Furthermore, the design of the al-

gorithm is independent of the dynamic range of the image, making it usable for HDR images as well. We will also show that it is advised, for multiple exposure HDR imaging, to perform HDR reconstruction of multiple exposure images prior to demosaicing, since it results in superior image quality and reduced computational costs.

2. Related Work

2.1. Demosaicing

Since the introduction of the Bayer CFA pattern in 1976, the most popular CFA arrangement, there have been many proposed solutions for the demosaicing problem. For a detailed survey on existing demosaicing methods we refer the readers to [10, 24].

Basic demosaicing methods rely on bilinear or bicubic filtering to estimate the two missing color values per pixel. They are computationally undemanding and perform well in smooth regions, but fail to produce good results near object or texture edges. In such regions simple interpolation of each color channel separately leads to the "zipper" effect, often in conjunction with false colors, as shown in Figure 1. Therefore it seems reasonable to perform any interpolation along edges in the image and not across them, since the pixel values along the edges remain roughly constant.

The most common edge detection approach is to compute gradients in the image, which is a basis of many demosaicing methods [12, 18, 28]. Kakarala and Baharov [14] use Jacobians of red, green and blue directions to detect edges. Weighted average interpolation [16] is an extension of the edge-detection methods that computes the likelihood of an edge in a certain direction, after which the pixel is computed as the weighted average of its neighbors. Hamilton and Adams [11] use discrete Laplacians in the blue and red channels as correction terms for green channel interpolation, thus reducing aliasing created by simple averaging. Chang et al. [4] use multiple gradients and then interpolate orthogonal to the estimated gradient direction of the image, as implemented in the *dcraw* package [5]. Takahashi et al. [27] use combined green and red/blue gradients to determine the interpolation direction of the missing green pixels, while the red and blue color channels are interpolated using the already interpolated green channel. The main problem of gradient-based algorithms is that the interpolation direction is estimated directly from the CFA image, and might deviate from the direction in the correct full RGB image. In our algorithm we use Takahashi's AAI algorithm, exploiting its properties for reconstructing red and blue channel, given the reconstructed green channel. We, however, estimate both the interpolation direction and the intensity in the green channel by performing a global neighborhood optimization, as explained in Section 3.1.

A second commonly used observation is that the color channels in naturally occurring images tend to be strongly correlated. Cok [6] first exploited this fact by separating the color space into luminance and hue (color), with the assumption that the latter is roughly constant within each object in the image. In order to prevent abrupt color changes, Cok first interpolated the green channel (representative for luminance), and then red/blue channels by using the constant hue principle. The constant hue assumption is used by various algorithms [11, 16, 18, 27]. Bennett et al. [3] propose a probabilistic model stating that at most two representative colors exist within each pixel neighborhood. Our algorithm performs minimization over an objective function which favors color smoothness, but does not enforce the constant hue assumption explicitly, since it actually may fail at boundaries between patches of different hue.

In recent years, there have been many new approaches for solving the demanding problem of demosaicing. Alleysson et al. [1] managed to remove some image artifacts by preprocessing the image in Fourier space, reducing aliasing in luminance and chrominance channels. Narasimhan and Nayar [22] apply a learning approach for demosaicing, deriving a set of polynomial demosaicing filter kernels from a set of high-quality three-channel images. Adaptive demosaicing [23] uses bilateral filtering together with an analysis of the local properties of the image to increase color similarity in pixel neighborhoods. The algorithm is able to smooth out pixel noise and to sharpen object edges.

A group of demosaicing algorithms is based on the idea that iterative minimization of some function expression may lead to better final results. These functions implicitly contain some color smoothness and color correlation terms, as illustrated in Keren and Osadchy [15]. Gunturk et al. [9] use an iterative scheme that enforces similarity between high frequencies in color channels, keeping the measured data. We also perform function minimization, using a cost function which enforces smoothness only along 1D features, without directly enforcing color constraints that might not be applicable in some image regions.

Farsiu et al. [8] formulate the demosaicing problem in a different manner. A general framework for combined treatment of demosaicing and super-resolution problems is presented. Instead of using a single image captured by the image sensor, the authors make use of multiple frames, which are slightly offset, noisy, blurred and sampled using the CFA. By optimizing over various regularization terms the images are combined together into one demosaiced high resolution image. Nevertheless, since the majority of imaging systems consist of only one imaging sensor, the proposed framework cannot be used for most practical applications.

The adaptive homogeneity-directed demosaicing (AHD) algorithm [13] uses a local measure of homogeneity to impose similarity in luminance and chrominance within the

image. The image is interpolated in both horizontal and vertical directions, after which every resulting pixel is transferred from RGB to CIELab color space. One of the two possible directions is chosen based on the local homogeneity measure, and the resulting image is further processed using iterative gradient-aware median filtering. Homogeneity-directed demosaicing is recognized as one of the best performing demosaicing algorithms [10]. However, a certain number of artifacts, mainly false colors, still remain in the resulting images.

2.2. HDR

Another aspect we are investigating in this paper is demosaicing of high dynamic range images, which might either be captured using a high dynamic range CFA camera or by combining a series of CFA images captured with different exposure times. After calibrating the camera response curve from known [7, 25] or unknown exposure settings [21] the input images are linearized and then combined using a weighted sum which accounts for the reliability of the reported digital values [25].

Most of the previous demosaicing approaches implicitly assume a limited dynamic range in the CFA image. Our objective function for explicitly expressing smoothness is designed to accommodate arbitrary ranges.

A technique for tone mapping and demosaicing HDR images inspired by the human visual system has been presented by Alleysson et al. [20]. The HDR reconstruction is based on psycho-physical evidence, and linear interpolation is used for demosaicing. Based on the HVS model the authors argue that demosaicing should follow HDR reconstruction. We support this statement by another argument based on the non-linearities in LDR images (Section 4).

3. Demosaicing Algorithm

Before describing our demosaicing algorithm we will present some insights that guided its development.

3.1. Demosaicing properties

Our demosaicing approach is, to a small part, based on the asymmetric average interpolation (AAI) algorithm by Takahashi et al. [27]. The AAI algorithm first interpolates the green channel only, after which red and blue channels are interpolated using the full green channel image. To determine the interpolation direction the algorithm uses combined pixel gradients, and the interpolation alone is performed with the assumption of slowly varying red-green and blue-green color differences.

Although the interpolation method itself does not perform well in many cases, we found that interpolating red and blue from the CFA image, given the correct green channel, does create almost perfect results as demonstrated in

(a) (b)

Figure 2. a) Applying the AAI algorithm to Bayer CFA data vs. b) applying the AAI algorithm for red/blue color channels to the image with perfect green channel. Notice how the AAI red/blue channel interpolation creates an almost perfect image.

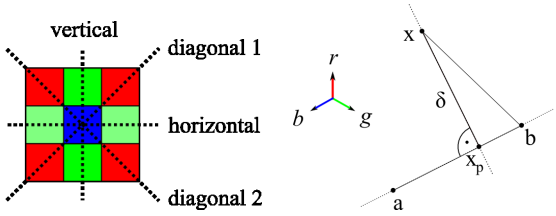
Figure 2. We take advantage of this property in our algorithm by optimizing over the green channel only, using the AAI to fill in red and blue color channels, reducing the number of free variables drastically. We will refer to the AAI algorithm, performing on the image with the full green channel information, as AAIG.

Figure 3. Failure cases of local gradient estimation. From left to right: CFA image, AAI interpolated, ground truth.

Another important demosaicing property is that per-pixel gradient-based methods cannot make proper decisions in all situations. To illustrate this we have zoomed in on a single pixel frequency stripe pattern, where the gradient estimated locally from the CFA image indicates an edge in the wrong direction (Figure 3). Occasionally, determining the right direction for pixel interpolation locally is impossible — the pixel in question is actually isolated and no edge is apparent. In such cases, any local interpolation scheme is destined to fail. Therefore, we have developed an algorithm which globally optimizes over all pixels in a pixel neighborhood simultaneously, avoiding wrong local gradients.

3.2. Overview of the Algorithm

We formulate demosaicing as a global optimization problem over the entire image. From the previous subsection we know that it is sufficient to optimize over the missing green pixels only. The objective function we minimize is defined over the resulting full color RGB image, *after* red



(a) (b)

Figure 4. (a) The 4 discrete directions along which the 1D features are traced, for which our smoothness term is evaluated. (b) distance to the line in RGB space ($\delta = S_p$)

and blue have been reconstructed using AAI. It is defined as a sum of local smoothness terms evaluated for every pixel. The per-pixel smoothness is defined using our projection metric, evaluated along four different directions through the pixel (Figure 4 (a)), as the best of the four values.

The resulting objective function contains a large number of local minima, requiring an appropriate optimization algorithm for which we chose the simultaneous perturbation stochastic approximation (SPSA) algorithm [19, 26]. Instead of optimizing over all green pixels at the same time, we optimize over a small pixel neighborhood, since the directional ambiguity is rather localized.

Before we start describing our demosaicing method we will briefly introduce the notation that will be followed throughout this paper. Pixels within the image are indexed with small latin letters ($x, y, \ldots$ or $a, b \ldots$). Distance in Euclidean space is given with δ. Letter d is a directional indicator, and formulas $x + d$ and $x - d$ indicate pixels that are offset by one pixel in direction d, but on opposite sides.

3.3. Projected smoothness measure

The core of our optimization is a simple, yet surprisingly general objective function, designed to increase smoothness along 1D features at every pixel x. We assume that no pixel is completely uncorrelated to its neighboring pixels, in which case it would be impossible to reconstruct its correct color. Every pixel bears some correlation with at least one of its neighbors (in natural images even more often with two neighboring pixels, along one direction). Since there is no robust way to determine the orientation of this 1D feature from the CFA image directly, we try out all four possible directions in the interpolated RGB image explicitly (Figure 4 (a)). Let $I(x)$ be the vector in RGB space, representing the reconstructed color of pixel x. We define a projected smoothness measure at pixel x along direction d as:

$$S_p(x,d) = \frac{(I(b) - I(a))^{\perp} \cdot (I(x) - I(a))}{\|(I(b) - I(a))^{\perp}\|} \qquad (1)$$

$$\text{with } a = x - d \text{ and } b = x + d.$$

In the above Equation $(I(b) - I(a))^{\perp}$ indicates a vector in RGB space orthogonal to the vector $I(b) - I(a)$ in the $I(a), I(x), I(b)$ plane. This formula allows the color of x

(a) (b)

Figure 5. Regularizing diagonals: (a) without, a checkerboard pattern is created. (b) It is removed by $\Delta(x,d)$.

to move closer to $I(a)$ or $I(b)$, as long as it stays close to the line spanned by $I(a)$ and $I(b)$, as illustrated in Figure 4 (b). $I(a)$ and $I(b)$ define a color transition, and with the projection metric we try to obtain the smoothest transition that interpolates the given CFA sample at x. For every pixel we select the direction d that has the minimum color projection smoothness.

When evaluating this projection metric along diagonals we need a *diagonal regularization* term to prevent the algorithm from optimizing two completely independent layers, since diagonals consist only of green or red/blue pixels. Figure 5 (a) demonstrates the case when every other pixel chooses a different diagonal, resulting in clearly visible artifacts. In order to couple these two layers we add a small fraction of the vertical and horizontal measure to the diagonal measure:

$$\Delta(x,d) = \begin{cases} 0 & , d \in \{d_h, d_v\} \\ \beta(S_p(x,d_h) + S_p(x,d_v)) & , d = d_{diag} \end{cases} \qquad (2)$$

We experimentally found $\beta = 0.06$ to work very well for all images, without penalizing diagonals too much. Additional regularization is required in order to prevent the objective function from assigning less energy to directions with high gradients. To accomplish this we include the *color deviation* term. For each color channel, along direction d, we compute the standard deviation and add the resulting values:

$$sd(x,d) = \sum_{c \in \{r,g,b\}} sd_c(I(x-d), I(x), I(x+d)), \qquad (3)$$

where sd_c is the standard deviation of color channel c for the three given RGB pixel values.

With the projected distance measure for a single pixel, and a single direction at hand, we now state the final energy function of our optimization. Instead of optimizing the entire image at once we iteratively optimize over every pixel neighborhood $Nbh(x)$ of 3×3 pixels, minimizing the following energy:

$$E(x) = \sum_{y \in Nbh(x)} \left(\min_d \frac{S_p(y,d) + \Delta(y,d)}{B(x)} + \lambda \, sd(y,d) \right) \qquad (4)$$

with $\lambda = 0.1$. In order to be independent of the dynamic range of the image, the error is scaled according to the pixel brightness term $B(x)$. We compute $B(x)$ by applying a Gaussian filter with a $3x3$ kernel on the RGB image.

3.4. Optimization Strategy

It is important to note that updating a single green pixel influences the interpolated colors in a 3×3 neighborhood around the pixel. In order to obtain the optimal value for a single green pixel, we therefore have to optimize over all unknown green pixels in a 3×3 neighborhood (5 in total). Since we define the smoothness along one out of four directions per pixel, the objective function for this small neighborhood typically contains a large number of local minima. To find a global optimum we have employed the simultaneous perturbation stochastic approximation (SPSA) algorithm. The SPSA algorithm samples the gradient around the current estimate stochastically, using only two samples, and is therefore able to occasionally jump out of local minima. The cheap estimation procedure for the gradient allows for efficient optimization of high dimensional problems. Since the SPSA guarantees convergence to the global optimum only stochastically, we run the SPSA five times with about 100 iterations for every 3×3 neighborhood and iterate five times over the entire image to propagate changes. The full iteration pipeline is outlined in Figure 6.

```
start with initial interpolated green
// image loop
for 5 iterations over the image
  // interpolate missing green
  for each red/blue CFA pixel
    compute initial energy
    // pixel loop
    for 5 iterations over the pixel
      select SPSA parameters
      perform SPSA with 100 iterations
      select best
    end for;
    if best>initial set pixel value
  end for;
end for;
```

Figure 6. Global optimization procedure.

Further runtime optimizations include: **Pixel selection.** Instead of updating every pixel in every iteration of the outer (image) loop, it is easy to detect which pixels might need further improvement. Smooth image regions, where the correct signal is perfectly sampled, do not need any update at all. We determine those regions by simply evaluating the energy function for each pixel and testing whether the energy is lower than an adaptive threshold value. Also, after every image loop we exclude those pixels that were reconstructed properly by the SPSA algorithm. We set the threshold at 5% of the maximum pixel energy value in the image.

Reducing iterations: The initially interpolated guess (e.g. the AAI interpolated image) may produce many pixels which need optimizing, but are in the proximity of the global minimum. It is therefore reasonable to perform the very first image iteration with a rather narrow search radius of the SPSA, i.e. using it as a stochastic steepest decent rather than globally exploring the error landscape. Since we are searching for the local minima, we perform only one pixel iteration. For the next image iteration steps we gradually adapt the SPSA parameters such that the algorithm is able to leave local minima and search for the globally optimal solution. This is achieved by increasing the perturbation values of the SPSA, which allows the algorithm to perform larger steps in its search. We stop the inner SPSA loop if the induced pixel change is less then 0.2% of the pixel's original value.

4. HDR Demosaicing Model

Reconstructing a high dynamic range image from multiple exposure requires first linearizing the measured digital counts and then blending between the different exposures. In this work we assume the response curve of the camera to be known (approaches for response curve reconstruction are given in [7, 25]). We follow the approach by Robertson et al. [25] to estimate the radiance L at a pixel x from the measured linearized intensities I_i of an exposure series with N exposure times t_i. A simple average could be computed as $L_{hdr}(x) = \sum_{i=0}^{N} \frac{I_i}{t_i}$. Robertson et al. proposed computing a weighted average using Gaussian weights, in order to dampen the noise introduced by badly exposed pixels.

4.1. Order of Demosaicing and HDR

If a CFA camera is used, demosaicing needs to be combined with HDR reconstruction to obtain a full-color HDR image. The order of the operations is critical. LDR images contain non-linear effects (response curve, over and under exposure) which are correctly counteracted by the weighting function if HDR reconstruction is performed prior to demosaicing. On the other hand, demosaicing the LDR images first would spread the non-linearity of the LDR information to neighboring pixels in a non-recoverable way. The input to the HDR reconstruction is already corrupted and can no longer be corrected through blending. In Table 1 we present the peak signal-to-noise ratio (PSNR) results of applying demosaicing first, followed by the HDR reconstruction, and compare them to the PSNR values if the order of demosaicing and HDR is reversed. The results show that the latter performs much better. Figure 7 demonstrates our demosaicing scheme applied to an HDR image.

5. Results

In this section we will present results obtained by our method and compare them with the AAI algorithm [27], as well as a state-of-the-art algorithm, the AHD interpolation method [13].

Figure 7. Examples of HDR reconstruction followed by demosaicing.

HDR Pipeline PSNR - dataset 1		
	Dem. + HDR PSNR	HDR + Dem. PSNR
Ours	32.2326	47.1861
AAIG	30.577	48.0565
HDR Pipeline PSNR - dataset 2		
	Dem. + HDR PSNR	HDR + Dem. PSNR
Ours	23.4334	45.3813
AAIG	23.3481	47.1674

Table 1. PSNR values comparison for two presented datasets. The reconstruction quality is much improved if the HDR reconstruction is performed before demosaicing.

5.1. Measuring demosaicing error

Measuring the quality of a demosaiced image is not a simple task since it is not easy to obtain a full RGB image for comparison. There exist several standard sets of LDR RGB images which offer high-quality images at low resolutions, e.g. the Kodak CD set [17]. The standard testing method is to reduce these images to the Bayer CFA pattern, demosaic them and then compare to the original ones. However, even if both raw camera data and the full RGB image are available, it is not at all clear how to compare two images.

In this paper we will present several comparison tests, mainly image PSNR and the mean squared error (MSE). Note, however, that such comparisons might be misleading. There exist artifacts, such as false colors, that immediately appear implausible to the observer, whereas uncorrelated noise, resulting in a comparably large "distance" to the reference image, might be perceived as completely unsuspicious and natural. Having in mind that the majority of demosaiced images will be subjectively evaluated by human observers, the perceived quality of demosaiced images should be evaluated with psycho-visual experiments. We instead present some demosaicing examples where the improvements made by our approach are clearly visible.

5.2. Demosaicing results

Since most of the existing algorithms perform reasonably well on a large number of pictures, we put our focus on such pictures that have shown to be particularly difficult to demosaic. Figure 10 shows the lighthouse test scene and some zoomed-in regions, where our algorithm outperforms previous approaches. The image is not perfectly reconstructed though, since there is a slight presence of faint false colors in the high-frequency fence structure. Figure 11 shows the house scene, with the window region zoomed in. Artifacts which are present in images reconstructed with AAI and AHD are removed with our algorithm. In Figure 12 we show that our algorithm occasionally suffers from slight zippering artifacts, which are rare and do not spread across large image areas.

PSNR values			
	lighthouse	house	window
Ours	39.7907	38.8117	39.7268
AHD	38.1076	35.0847	40.3453
AAI	36.7592	33.0647	40.6047
AAIG	46.3841	46.6925	46.8902
MSE values			
	lighthouse	house	window
Ours	0.02676	0.03353	0.02715
AHD	0.03943	0.07908	0.02355
AAI	0.05378	0.12591	0.02219
AAIG	0.00586	0.00546	0.00522

Table 2. PSNR and MSE value comparisons for three datasets. Our algorithm manages to outperform both AHD and AAI algorithms.

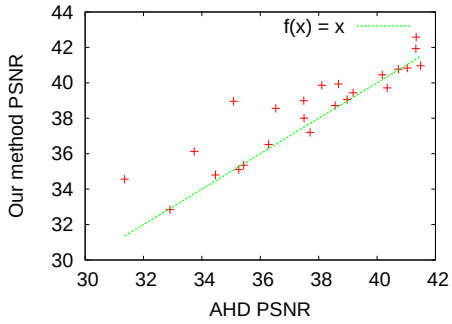
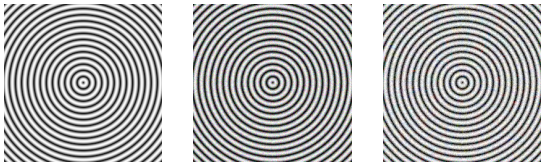


Figure 8. Plot of PSNR values ratios for AHD and our method (measured on the full Kodak set). Note that the corresponding values tend to be above the $f(x) = x$ line, ploted in green.

In Table 2 we present the PSNR measurements and the relative errors for three test scenes, which clearly confirm the superiority of our algorithm compared to both AAI and AHD interpolation methods. The PSNR value of the almost perfect AAIG reconstruction serves as a reference for comparison. Furthermore, we have tested both AHD and our method on noisy synthetic data (see Figure 9), where both methods perform roughly the same (note that both AHD and our method perform demosaicing without denoising).

In Figure 8 we have plotted the PSNR values of our method against the PSNR values of the AHD method. The plot clearly shows an improvement of the PSNR in most of the tested images. In cases where AHD demosaicing out-



(a) (b) (c)

Figure 9. Noise images: a) Ground Truth, b) AHD (PSNR $= 20.4852$), c) Ours (PSNR $= 20.4983$).

performs our method, the difference is not decisive.

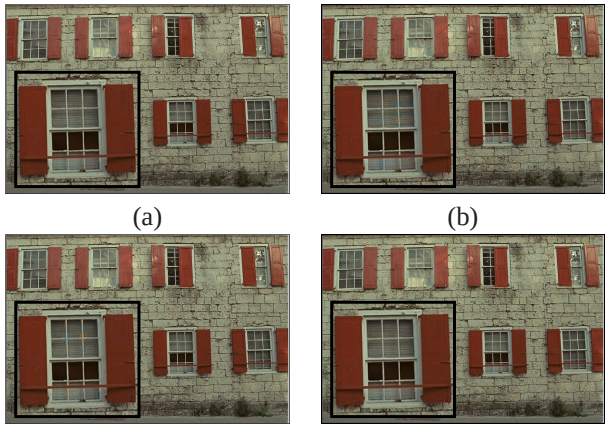
(a) (b) (c) (d)

Figure 10. Lighthouse: a) RGB, b) AAI, c) AHD, d) Ours).

We have also compared execution times of our demosaicing algorithm with those of the AHD algorithm. For the "House" image in Figure 11, AHD computation time is 63 seconds, while our optimization scheme needs 101 seconds. The reason for slower computation time lies in our global optimization, which cannot be performed with a small number of iterations.

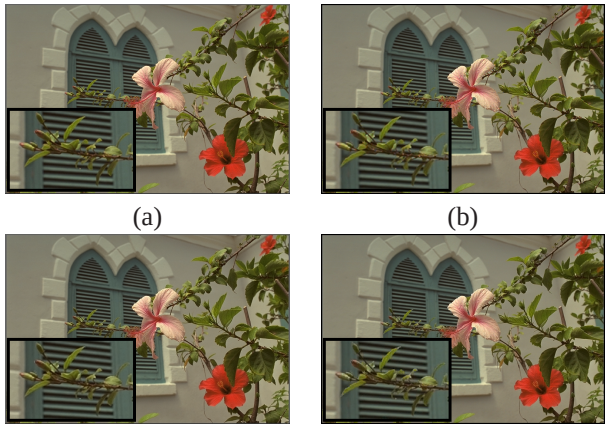
6. Conclusion and Future Work

In this paper we have presented a novel demosaicing algorithm that globally minimizes a simple, yet effective objective function, producing significantly less artifacts in difficult image regions compared to state-of-the art demosaicing techniques. Our objective function tries to increase the smoothness along one out of the four directions in every pixel of the image.



(a) (b) (c) (d)

Figure 11. House: a) RGB, b) AAI, c) AHD, d) Ours.



(a) (b) (c) (d)

Figure 12. Window: a) RGB, b) AAI, c) AHD, d) Ours.

It is interesting to note that, while the objective function does not explicitly assume any coupling between the color channels, our approach rarely produces false colors. This indicates that our objective function corresponds to a property found in most natural images. In the future one might exploit the power of this function as a regularization term in other image reconstruction tasks, e.g. super resolution or deconvolution. Furthermore, we would like to concentrate on further improving our demosaicing approach by addressing explicitly the problem of noise in the CFA data, as well as increasing the efficiency of the algorithm.

Besides the novel demosaicing algorithm, we have confirmed that in the context of HDR imaging, the order of demosaicing and HDR reconstruction should be inverted. First the HDR reconstruction on the CFA images should be performed, followed by demosaicing the HDR CFA image

for superior image quality, as well as decreased computational costs.

7. Acknowledgements

We would like to thank Joachim Giesen for the fruitful discussion concerning global optimization strategies. Also, we would like to thank our reviewers for their suggestions and comments. This work has been partially funded by the DFG Emmy Noether fellowship (Le 1341/1-1) and the Max Planck Center for Visual Computing and Communication (BMBF-FKZ01IMC01).

References

[1] D. Alleysson, S. Süsstrunk, and J. Hérault. Linear demosaicing inspired by the human visual system. *IEEE Transactions on Image Processing*, 14(4):439–449, 2005. 2

[2] B. Bayer. Color imaging array. U.S. Patent No. 3,971,065, July 1976. 1

[3] E. P. Bennett, M. Uyttendaele, C. L. Zitnick, R. Szeliski, and S. B. Kang. Video and image bayesian demosaicing with a two color image prior. In *9th European Conference on Computer Vision (ECCV)*, pages 508–521, 2006. 2

[4] E. Chang, S. Cheung, and D. Y. Pan. Color filter array recovery using a threshold-based variable number of gradients. In *Proc. SPIE Vol. 3650, p. 36-43, Sensors, Cameras, and Applications for Digital Photography*, pages 36–43, Mar. 1999. 2

[5] D. Coffin. Dcraw. http://www.cybercom.net/ dcoffin/dcraw/. 2

[6] D. Cok. Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal. U.S. Patent No. 4,642,678, 1987. 2

[7] P. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH 1997*, pages 369–378, 1997. 3, 5

[8] S. Farsiu, E. M., and M. P. Multi-frame demosaicing and super-resolution of color images. *IEEE Trans. on Image Processing*, 15:141–159, 2006. 2

[9] B. Gunturk, Y. Altunbasak, and R. Mersereau. Color plane interpolation using alternating projections. *IEEE Transactions on Image Processing*, 11(9):997–1013, 2002. 2

[10] B. Gunturk, J. Glotzbach, Y. Altunbasak, R. Schafer, and R. Mersereau. Demosaicking: Color filter array interpolation in single-chip digital cameras. *IEEE Signal Processing Magazine*, 22(1):44–54, 2005. 2, 3

[11] J. Hamilton and J. Adams. Adaptive color plane interpolation in single sensor color electronic camera. U.S. Patent No. 5,629,734, 1997. 2

[12] R. Hibbard. Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients. U.S. Patent No. 5,382,976, 1995. 2

[13] K. Hirakawa and T. Parks. Adaptive homogeneity-directed demosaicing algorithm. *IEEE Transactions on Image Processing*, 14(3):360–369, 2005. 2, 5

[14] R. Kakarala and Z. Baharov. Adaptive demosaicing with the principal vector method. *IEEE Transactions on Consumer Electronics*, 48(4):932–937, Nov 2002. 2

[15] D. Keren and M. Osadchy. Restoring subsampled color images. *Machine Vision and Applications*, 11(4):197–202, Dec 1999. 2

[16] R. Kimmel. Demosaicing: Image reconstruction from color ccd samples. *IEEE Transactions on Image Processing*, 8:1221–1228, 1999. 2

[17] Kodak Eastman Company. PhotoCD PCD0992. http://r0k.us/graphics/kodak/. 6

[18] C. Laroche and M. Prescott. Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients. U.S. Patent No. 5,373,322, 1994. 2

[19] J. L. Maryak and D. C. Chin. Global random optimization by simultaneous perturbation stochastic approximation. In *WSC '01: Proceedings of the 33nd conference on Winter simulation*, pages 307–312, Washington, DC, USA, 2001. IEEE Computer Society. 4

[20] L. Meylan, D. Alleysson, and S. Süsstrunk. A Model of Retinal Local Adaptation for the Tone Mapping of Color Filter Array Images. *Journal of the Optical Society of America A (JOSA A)*, 24(9):2807–2816, 2007. 3

[21] T. Mitsunaga and S. Nayar. Radiometric Self Calibration. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 374–380, Jun 1999. 3

[22] S. Narasimhan and S. Nayar. Enhancing Resolution along Multiple Imaging Dimensions using Assorted Pixels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(4):518–530, Apr 2005. 2

[23] R. Ramanath and W. Snyder. Adaptive demosaicking. *Journal of Electronic Imaging*, 12(4):633–642, 2003. 2

[24] R. Ramanath, W. Snyder, G. Bilbro, and W. Sander. Demosaicking methods for bayer color arrays. *Journal of Electronic Imaging*, 11(3):306–315, 2002. 2

[25] M. Robertson, S. Borman, and R. Stevenson. Estimation-theoretic approach to dynamic range improvement using multiple exposures. *Journal of Electronic Imaging*, 12(2):219–228, 2003. 3, 5

[26] J. C. Spall. Overview of the simultaneous perturbation method for efficient optimization. *Hopkins APL Technical Digest*, 19:482–492, 1998. 4

[27] Y. Takahashi, H. Kikuchi, S. Muramatsu, Y. Abe, and N. Mizutani. A new color demosaicing method using asymmetric average interpolation and its iteration. *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, E88-A(8):2108–2116, 2005. 2, 3, 5

[28] C.-Y. Tsai and K.-T. Song. A new edge-adaptive demosaicing algorithm for color filter arrays. *Image Vision Comput.*, 25(9):1495–1508, 2007. 2