

Automatic Removal of Chromatic Aberration from a Single Image

Sing Bing Kang

Microsoft Research, One Microsoft Way, Redmond, WA 98052

SingBing.Kang@microsoft.com

Abstract

Many high resolution images exhibit chromatic aberration (CA), where the color channels appear shifted. Unfortunately, merely compensating for these shifts is sometimes inadequate, because the intensities are modified by other effects such as spatially-varying defocus and (surprisingly) in-camera sharpening. In this paper, we start from the basic principles of image formation to characterize CA, and show how its effects can be substantially reduced. We also show results of CA correction on a number of high-resolution images taken with different cameras.

1. Introduction

The popularity of digital photography is spurred by lowering costs and higher sensor resolution. Unfortunately, the increase in sensor resolution (larger sensor size) is often not accompanied by increase in the quality of the optics. As a result, aberrations such as chromatic aberration (CA) are more pronounced in higher resolution cameras with relatively inexpensive lenses. Sample close-up cropped views are shown in Figure 1. Note the slight relative shifts in the color channels as seen on the top row. Observe also the under- and over-shooting at the edges (ringing), especially for the first example. Merely compensating for the color shifts would not be adequate in these cases. In the third example, the purple artifacts are also known as *purple fringing*.

Such artifacts are unacceptable in professional photography; digital photos are typically touched up manually using software such as Adobe[®] PhotoShop. Other applications such as PTLens[®] use a lookup table of precomputed parameters to reduce the artifacts. (It is not clear how these applications correct for CA, but it is possible that they use constant radial magnification and/or linear color transformation, since the number of degrees of freedom available to the user is typically limited to just two or three.) From the computer vision perspective, understanding how these artifacts occur and how they can be automatically reduced is important for maximizing the amount of useful visual data

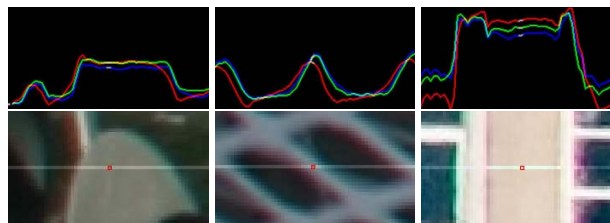


Figure 1. Effect of chromatic aberration. In all cases, the image center is to the right of the cropped images. The graphs on the top row are the color intensity profiles along the middle scanline.

for analysis (e.g., segmentation, edge detection).

Techniques have been proposed to reduce the effect of CA. Classical techniques rely on optimal lens design [9] or special hardware to recover the aberration coefficients through wavefront measurements (e.g., using a diffraction grating and scanning electron microscope or some form of interferometer [2]). Others rely on primarily software solutions; one such technique involves precalibrating the color channels for optimal focus, magnification, and shift [11]. However, producing a CA-free image subsequently requires taking three separate images, each using the optimal settings for a different color channel.

Boult and Wolberg [1], on the other hand, use only a single image; they warp the red and green color channels to the blue channel. This is accomplished by fitting edge displacements with cubic splines, with blue edges as the reference. However, they do not account for the attendant relative defocus and intensity distortion (under- and over-shooting) effects. In addition, because the computed displacement maps do not explicitly model the image formation process, it is not clear if the extracted maps can be used to directly correct a different image taken by the same camera. This is because the extracted maps are highly dependent on the relative density and distribution of the edges.

In this paper, we focus on how we can undo most of the effects of chromatic aberration from a single image. Before we present our approach, we first review a few important concepts in optics. More detailed descriptions of optical concepts can be found in references such as [9, 10, 12].

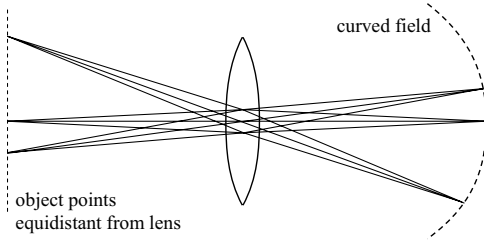


Figure 2. Illustration of field curvature. Rays coming from points from the same distance from the lens but at differing distances from the optic axis are focussed at different distances from the lens.

2. Optical concepts

Optical aberrations are categorized as monochromatic or chromatic. The mathematical treatment of geometrical imagery, now known as aberration theory, was first described by Seidel in 1856. There are five primary monochromatic (Seidel) aberrations: distortion (the familiar barrel or pincushion distortion), spherical aberration, coma, astigmatism, and field curvature. Chromatic aberrations are caused by the refractive index of the lens being wavelength-dependent, and can be separated into longitudinal and lateral components. By chromatic aberration (CA), we refer to the combination of longitudinal and lateral aberrations. In this paper, we focus on CA. The effect of CA is tightly coupled with the notions of field curvature and astigmatism.

2.1. Field curvature and astigmatism

The ideal imaging surface is flat (i.e., having a flat field) and coincident with the sensor surface. In reality, for a typical camera, the off-axis parts of the image are closer to the lens than the on-axis parts (Figure 2). In the absence of all other aberrations, the curved image surface is known as the *Petzval* surface.

In an optical system, a plane containing the optical axis is called the tangential or meridional plane. The sagittal plane is orthogonal to the tangential plane and intersects the tangential plane along the chief (or primary) ray of the object. Astigmatism refers to the phenomenon where rays coming from the object focus in two lines on different surfaces, as shown in Figure 3. One focussed line corresponds to the tangential rays while the other corresponds to the sagittal rays. The “best” image of the object is considered to be at a location halfway between these two surfaces; this location yields what is called the circle of least confusion.

The combination of field curvature and astigmatism produces a more complex “best” imaging surface, which we refer to as the *focal surface*. In our paper, we attempt to recover the *relative* focal surface using the green channel as reference. We chose green as the reference due to its implicit higher reliability, since the number of green sam-

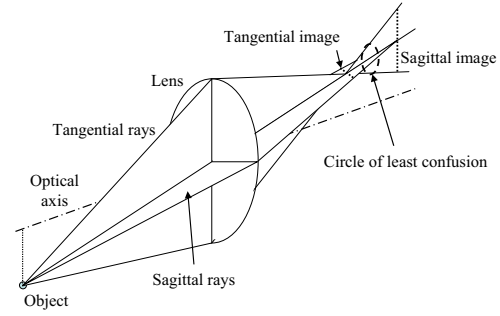


Figure 3. Illustration of astigmatism.

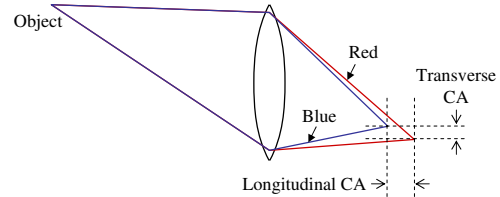


Figure 4. Illustration of longitudinal and lateral chromatic aberration (only the red and blue rays are shown here).

ples in a raw Bayer mosaic image (for single CCD sensors) are twice those for red and blue. We have also observed from images taken with five different high-resolution cameras that the green channel tends to be the sharpest.

2.2. Chromatic and spherical aberrations

The lens properties vary with wavelength because the index of refraction is wavelength-dependent. In addition to the monochromatic (Seidel) aberrations, chromatic aberration contains wavelength dependent shifts in focus. These shifts can be decomposed into *longitudinal chromatic aberration* (shifts along the optical axis) and *lateral chromatic aberration* (shifts perpendicular to the optical axis). Lateral chromatic aberration, otherwise known as transverse chromatic aberration, combines with spherical aberration (which is a function of aperture radius) to determine the image height of a colored ray. This combined effect is called *spherochromatic aberration* (SCA) or *Gauss error* [8]. In this paper, for simplicity, by CA, we are actually referring to the combination of conventional CA and spherical aberration. Figure 4 illustrates the occurrence of CA.

3. Our approach

The imaging pipeline we use in our analysis is shown in Figure 5. The pipeline consists of optical, sampling, and post-processing stages, in that order. Note that the other optical effects such as radial distortion and vignetting, which can be estimated independently, are outside the scope of this work. R_s, G_s, B_s are incident continuous scene colors and R_I, G_I, B_I are the observed discrete image colors. There

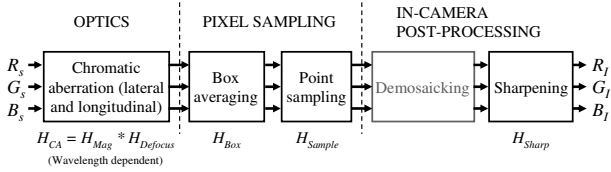


Figure 5. Simplified imaging pipeline.

are three parts to the pipeline: optics, pixel sampling, and in-camera post-processing. We made a number of assumptions in our work to simplify our analysis. First, we assume that the effects of demosaicking (i.e., the process of raw-to-color conversion) is relatively small. We ignored the in-camera color transformation used for white balancing, and assume that the image aspect ratio is one.

3.1. Optics

Chromatic aberration (CA) results in each color channel having its own focal surface being deviated from the ideal flat field. We model the focal surface as 3D surface that is radially symmetric, so it is sufficient to recast the lateral and longitudinal shift functions along the radial direction ($d_{Lat}(r)$ and $d_{Long}(r)$, respectively). We also assume that the focal surface is smooth. We model these deviations induced by $d_{Lat}(r)$ and $d_{Long}(r)$ as radially dependent magnification (convolution kernel H_{Mag}) and defocusing (convolution kernel $H_{Defocus}$), respectively, i.e., $H_{CA} = H_{Mag} * H_{Defocus}$. If the incoming light distribution is M_s , the light distribution that impinges on the sensor is thus

$$M = M_s * H_{CA}. \quad (1)$$

H_{CA} is implemented in two stages:

1. **Magnification:** We warp the image using the radial function $(x', y')^T = (x, y)^T(a_0 + a_1r(x, y) + a_2r^2(x, y) + a_3r^3(x, y))$, where $r(x, y) = \sqrt{x^2 + y^2}$ is the radius relative to some center (c_x, c_y) . We use four parameters a_0 , a_1 , a_2 , and a_3 to characterize radially-varying magnification.
2. **Defocus:** $H_{Defocus}$ is of the form of a pillbox shown in Figure 6, and it is characterized using a single parameter ρ that specifies the size of its support. Since the kernel is discrete, we compute weights based on whether the position is wholly inside the pillbox, partially inside, or wholly outside. The weights are normalized to sum to one. We characterize the radial distribution of ρ by $\rho(r) = \rho_0 + \rho_1r + \rho_2r^2$, with r being the distance of the pixel to center (c_x, c_y) (same as magnification center).

Please note that because the effect of CA is spatially-varying, it is easier to analyze and implement in the spatial domain rather than frequency domain.

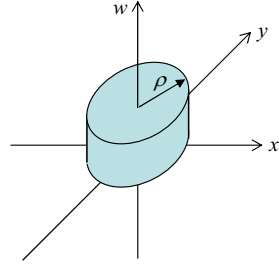


Figure 6. Defocus kernel (pillbox). w is the weight distribution over (x, y) , such that the sum over the spatial support (radius ρ) is one.

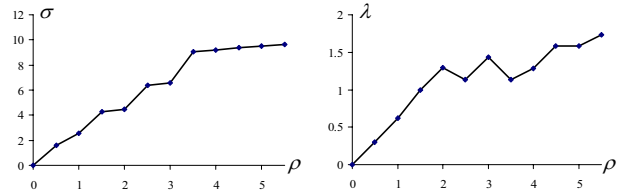


Figure 7. Link between defocus parameter ρ and SHS parameters σ and λ .

In our analysis, we need to invert these processes. Fortunately, the magnification stage can be easily inverted. However, without good priors, undoing the defocus effect is impossible if high-frequency components are present. We “undo” defocus using a sharpening filter given by (5), but with a major modification—we suppress ringing (characteristic of a typical sharpening operation) by restricting values to within the local minimum and maximum *original* intensities. In other words, if the sharpened value is below the local minimum, we replace it with the local minimum; similarly, we take the local maximum if the sharpened value exceeds it. We call this operation *sharpening with halo suppression* (SHS).

In order to maintain the same number of degrees of freedom in the forward and inverse process of CA, we empirically link defocus parameter ρ with SHS parameters σ and λ . We do this by running simulations of defocusing a checkerboard pattern and minimizing the reconstruction error using the SHS filter. The graphs illustrating this link are shown in Figure 7.

3.2. Pixel sampling

The pixel sampling analysis follows that in [5]. Assume the sensor to be a rectangular array of rectangular sensing elements (pixels), and that each pixel has uniform quantum efficiency over its area. In spatial domain, the averaging function is a box:

$$H_{Box}(x, y, w_x, w_y) = \text{Rect}_2\left(\frac{x}{w_x}, \frac{y}{w_y}\right), \quad (2)$$

where w_x and w_y are the pixel length and width, respectively, and $\text{Rect}_2()$ is the 2D rectangular function. Suppose the pixel spacings are d_x and d_y along the x- and y-directions, respectively, with (ϕ_x, ϕ_y) being the phase shift of the grid relative to the optic center. The sampling function is then

$$H_{\text{Sample}}(x, y, d_x, d_y, \phi_x, \phi_y) = \frac{1}{d_x d_y} \text{Shah}_2\left(\frac{1}{d_x}(x - \phi_x), \frac{1}{d_y}(y - \phi_y)\right), \quad (3)$$

where $\text{Shah}_2()$ is the 2D Shah function (also known as the sampling or replicating function).

If M_s is the continuous image formed on the image plane, its discrete version M_d is

$$M_d = (M_s * H_{\text{Box}}) \cdot H_{\text{Sample}} = \Theta(M_s). \quad (4)$$

As can be seen later (Section 3.4), we need to invert this process, i.e., approximate a continuous signal $\Phi(X)$ given its discrete counterpart X . However, the pixel sampling process results in a bandlimited signal, which is the best we can theoretically reconstruct. In our implementation, we approximate $\Phi(X)$ through super-sampling by a factor of three the input resolution of X and using bicubic interpolation to compute in-between values. This choice of approximation is motivated by speed and memory footprint considerations. To “sample” back to the original input resolution, we apply $\Theta(\cdot)$ to the super-sampled image with $w_x = w_y = 3$, $d_x = d_y = 3$, and $\phi_x = \phi_y = 0$.

3.3. In-camera post-processing

Single CCD-based cameras use optical and electrical low-pass filters to reduce aliasing caused by the mosaic array in front of the sensor. For a given CCD size and pixel pitch, the anti-aliasing filter design has a significant impact on the reduced sharpness of the image [7]. Many digital cameras, by default, apply some amount of internal sharpening in an attempt to counteract the reduced sharpness. However, this sometimes has the effect of significantly enhancing jaggies, noise, and other artifacts such as ringing. Figure 8 shows the effect of in-camera sharpening for the Canon PowerShot SD500, which allows the user to select a less sharpening effect. (Most lower-end consumer cameras do not have the “less-sharp” option.)

We model the in-camera sharpening using a separable unsharp mask¹ H_{Sharp} :

$$H_{\text{Sharp}} = H_1 + \lambda(H_1 - H_\sigma), \quad (5)$$

¹The seemingly unintuitive name “unsharp mask” used for sharpening has its origins in the printing industry; see www.normankoren.com/PWP_contrast_masking.html for a more detailed explanation.

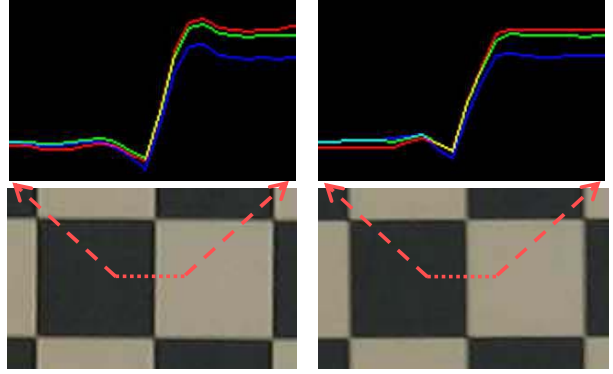


Figure 8. Examples of in-camera sharpening (Canon PowerShot SD500). The full image resolution is 3072×2304 ; only subimages of 160×100 and color intensity profiles at a particular edge location are shown here. Notice the significant ringing for the default setting (left); even the less-sharp setting has some ringing (right).

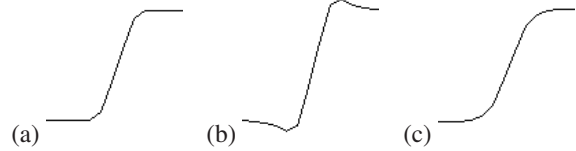


Figure 9. Result of undoing the effect of sharpening: (a) original defocused, (b) sharpened using (5), and (c) unsharpened using (7). In this simulation, an ideal checkerboard pattern was used with defocusing parameter $\rho = 0.5$ and sharpening (and inverse sharpening) parameters $\sigma = 2.0$ and $\lambda = 0.7$.

where H_1 is a kernel with one at the center and zero everywhere else, H_σ is a Gaussian blur kernel with parameter σ , and λ is the amount of high-frequency contribution. Note that this particular parameterization of the unsharp mask has two degrees of freedom. If the observed discrete image is M_I , we have

$$M_I = M_d * H_{\text{Sharp}}. \quad (6)$$

We use the second-order approximation to invert the effect of sharpening, i.e.,

$$H_{\text{Sharp}}^{-1} \approx H_1 - \lambda(H_1 - H_\sigma) + \lambda^2(H_1 - H_\sigma)^2. \quad (7)$$

This is a form of blurring. Figure 9 shows profiles of a 2D edge (which is part of a 2D checkerboard pattern). Notice that the original (leftmost) and “unsharpened” (rightmost) profiles are very similar. This approximation works well for $\lambda < 1$; $\lambda \geq 1$ results in some residual over- and under-shooting.

There is also the effect of demosaicking, which has two effects: zippering (when colors are misestimated, typically at color boundaries, resulting in speckle noise), and blurring (due to interpolation). The algorithms used for demosaicking can vary significantly, with the type of interpolation

ranging from bilinear to median [3] to gradient-sensitive (e.g., [4]). In this work, we assume that the effect of demosaicking is minor compared to all the other effects, and that there is *net sharpening* as a result of in-camera post-processing.

In our implementation, the width of H_{Sharp} and H_{Sharp}^{-1} is $(2\lceil 3\sigma \rceil + 1)$ pixels, with $\lceil x \rceil$ being the ceiling of x .

3.4. Putting everything together

From (1), (4), and (6), the discrete output color channels are related to their respective continuous inputs by

$$\begin{aligned} R_I &= \Theta(R_s * H_{CA,R}) * H_{Sharp} \\ G_I &= \Theta(G_s * H_{CA,G}) * H_{Sharp} \\ B_I &= \Theta(B_s * H_{CA,B}) * H_{Sharp}. \end{aligned} \quad (8)$$

Suppose we are able to reasonably undo the in-camera sharpening and approximate the continuous (bandlimited) signal $\Phi(X)$ from its discrete counterpart X . Then, we have

$$\begin{aligned} \Phi(R_I * H_{Sharp}^{-1}) &= R_s * H_{CA,R} \\ \Phi(G_I * H_{Sharp}^{-1}) &= G_s * H_{CA,G} \\ \Phi(B_I * H_{Sharp}^{-1}) &= B_s * H_{CA,B}. \end{aligned} \quad (9)$$

Unfortunately, unless we know the exact scene color distribution, there is ambiguity of R_s, G_s, B_s up to some radial functions of magnification and defocusing. There is also an ambiguity associated with H_{Sharp}^{-1} . Since our goal is to minimize the effect of CA, we can set one of the color channels as the reference and assume it to have a flat field. As mentioned in Section 2.1, we chose green as the reference.

With the use of the green channel as the flat field reference, we have

$$\begin{aligned} \Phi(R_I * H_{Sharp}^{-1}) * H_{CA, RG}^{-1} &= R'_s \\ \Phi(G_I * H_{Sharp}^{-1}) &= G'_s \\ \Phi(B_I * H_{Sharp}^{-1}) * H_{CA, BG}^{-1} &= B'_s, \end{aligned} \quad (10)$$

with $G'_s = G_s * H_{CA,G}$, $R'_s = R_s * H_{CA,G}$, $B'_s = B_s * H_{CA,G}$, $H_{CA, RG} = H_{CA, G}^{-1} * H_{CA, R}$, and $H_{CA, BG} = H_{CA, G}^{-1} * H_{CA, B}$.

To remove the ambiguity associated with H_{Sharp}^{-1} in (9), we estimate it from the image independently of CA using the technique described in Section 3.6. Given H_{Sharp}^{-1} and the constraint that $R'_s = G'_s = B'_s$ in regions where similarly strong edges exist in all three color bands (Section 3.5), we can approximate $H_{CA, RG}^{-1}$ and $H_{CA, BG}^{-1}$ by minimizing

$$\|\Phi(R_I * H_{Sharp}^{-1}) * H_{CA, RG}^{-1} - \Phi(G_I * H_{Sharp}^{-1})\|_E^2 \quad (11)$$

and

$$\|\Phi(B_I * H_{Sharp}^{-1}) * H_{CA, BG}^{-1} - \Phi(G_I * H_{Sharp}^{-1})\|_E^2, \quad (12)$$

respectively². The subscript E in $\|\cdot\|_E$ is used to indicate computation over the edge regions. Since H_{Sharp}^{-1} is known, we can rewrite (11) and (12) as

$$\|\Phi(R'_I) * H_{CA, RG}^{-1} - \Phi(G'_I)\|_E^2 \quad (13)$$

and

$$\|\Phi(B'_I) * H_{CA, BG}^{-1} - \Phi(G'_I)\|_E^2, \quad (14)$$

respectively, with $R'_I = R_I * H_{Sharp}^{-1}$, $G'_I = G_I * H_{Sharp}^{-1}$, and $B'_I = B_I * H_{Sharp}^{-1}$.

Recall that R_I, G_I, B_I are the input discrete image color distributions and that we wish to find an optimal transformation for R_I and B_I to minimize the effects of CA. Once $H_{CA, RG}^{-1}$ and $H_{CA, BG}^{-1}$ are found, the desired transformed R_I and B_I over the entire image are

$$R''_I = \Theta(\Phi(R'_I) * H_{CA, RG}^{-1}) * H_{Sharp} \quad (15)$$

and

$$B''_I = \Theta(\Phi(B'_I) * H_{CA, BG}^{-1}) * H_{Sharp}. \quad (16)$$

If there is no in-camera sharpening, (15) and (16) reduce to $R''_I = \Theta(\Phi(R_I) * H_{CA, RG}^{-1})$ and $B''_I = \Theta(\Phi(B_I) * H_{CA, BG}^{-1})$, respectively. Note that for each color channel (R, B), the number of unknowns to be computed is nine (four to characterize radially-varying magnification, two for the radial center, and three to characterize radially-varying defocus; see Section 3.1).

We first used Levenberg-Marquardt (L-M) to optimize (13) and (14), and surprisingly it did not work well. It turns out that the increments used to estimate local gradients were too small to make a measurable change in the objective functions, and L-M stalled in many occasions. We instead used the Nelder-Mead simplex method [6] and specified a reasonable set of starting points. This simplex method is based on evaluating a function at the vertices of a simplex, then iteratively shrinking the simplex as better points are found until some desired bound is obtained.

Even though the green channel is the reference channel, it may contain ringing effects. We removed most of such occurrences by replacing G_I with $(G_I * H_{Sharp}^{-1}) * H_{Sharp}$. Note that here $H_{Sharp}^{-1} * H_{Sharp}$ is not identity because H_{Sharp} is implemented as sharpening with halo suppression (SHS, see Section 3.1).

3.5. Finding strong color edges

Our algorithm relies on being able to locate strong color edges that are in very close proximity to each other. We apply the difference-of-Gaussians (DOG) filter over the three

²There is one implementation detail: before we run the optimization for B_I , we compensate for the global intensity scale and bias of G_I relative to B_I over the edge regions. (We do *not* want to modify B_I , as that would modify its CA characteristics.) The process is repeated for R_I .



Figure 10. Result of extracting collocated strong color edges. Left: original color image (3008×2000), right: detected edge pixels.

color channels independently. The strong edges are found by using a high threshold, and isolated edge pixels (those without any 4-connected neighbors) are removed. The local gradients allow us to estimate the normals at the edge pixels. For each edge pixel in a color band, there must be nearby (within 2 pixels) edge pixels in the other color bands; otherwise it is rejected. An example of extracted strong collocated color edges is shown to the right of Figure 10.

3.6. Estimating H_{sharp}^{-1}

We use the results of Section 3.5 to estimate the two parameters λ and σ for H_{sharp}^{-1} . More specifically, we use areas within 11 pixels of these edges (using the edge locations in the green channel as reference locations). Then, we define the cost of ringing at the vicinity of these edges along their normal directions. To reduce the amount of re-sampling, we analyze along either the horizontal or vertical direction, depending on which is closer (in angles) to the normal direction.

Given the intensity profile as shown in Figure 11 and setting $\epsilon = 2$ pixels and $\xi = 9$ pixels in all our experiments, we compute $(\mu_{left}, \sigma_{left})$ and $(\mu_{right}, \sigma_{right})$. We define the penalty as $E_{ring} = (\sigma_{left} + \sigma_{right}) / |\mu_{left} - \mu_{right}|$. Here we wish to maximize the difference between the left and right parts of the signal while minimizing their variances. The former ensures we do not oversmooth, while the latter acts to suppress signal variation due to ringing. We estimate the optimal values of λ and σ in (7) by minimizing the sum of the penalties over all the edge points and over all three color channels. The Nelder-Mead simplex algorithm is used for parameter estimation.

A typical result is shown in Figure 12. We simulate the sharpening by applying $H_{sharpen}$ (no halo suppression) with $\sigma = 2.75$ and $\lambda = 0.75$. Starting with the original image in (a), we arrive at the simulated in-camera sharpened input (b). Notice the ringing in the intensity profile. Using (b) as input to our estimator, we arrive at $H_{sharpen}^{-1}$ with $\sigma = 3.5$ and $\lambda = 0.35$. While these numbers may seem quite different from the ground truth, the resulting reconstructed image and profile shown in (c) are very similar to the original (a). The original has been reasonably well recovered.

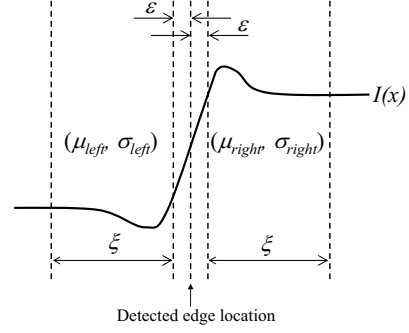


Figure 11. Defining the cost of under- and over-shooting. Given the 1D profile of the detected edge $I(x)$, we subdivide the curve into three parts: the left $[-\xi - \epsilon, -\epsilon]$, the middle $[-\epsilon, \epsilon]$, and the right $[\epsilon, \xi + \epsilon]$. The location of the edge is set to $x = 0$ here. The mean and standard deviation of $I(x)$ in the left part are μ_{left} and σ_{left} , respectively. μ_{right} and σ_{right} are similarly defined for the right part of $I(x)$.

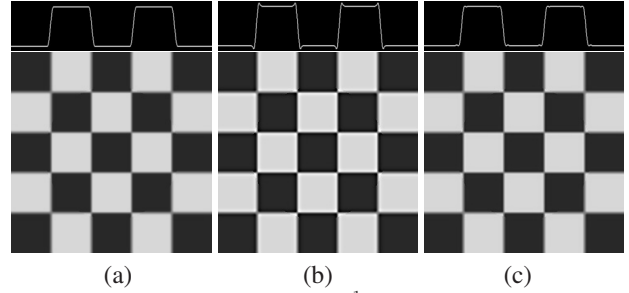


Figure 12. Result of estimating H_{sharp}^{-1} . The resolution of each image is 250×250 . The intensity profiles along a horizontal scanline are shown on top. (a) Original image, (b) simulating in-camera sharpening using H_{sharp} with $\sigma = 2.75$ and $\lambda = 0.75$ (notice the ringing effect in the profile), (c) after applying estimated H_{sharp}^{-1} with $\sigma = 3.5$ and $\lambda = 0.35$. H_{sharp}^{-1} was estimated from (b). Notice the considerable similarity between (a) and (c).

4. Results

We now show results of using our technique on a number of different images (Figures 10 and 13). Figure 14 compares our result with that for a version that computes only a radial shift function (with the same parameters $a_0, \dots, a_3$). This example shows that while compensating just for the color shift improves the quality of the image significantly, more can be done to further reduce the CA-related artifacts. Our technique was able to significantly reduce the ringing in the signal in addition to handling the color shift.

For the 3072×2048 image in Figure 14, it took about 6 secs to extract the edge image, 2 minutes to recover the two parameters for $H_{sharpen}^{-1}$, and 13.5 minutes to estimate the nine other CA-related parameters (associated with magnification, defocussing, and radial center). If only the radial magnification function (specified by $a_0, \dots, a_3, c_x$, and

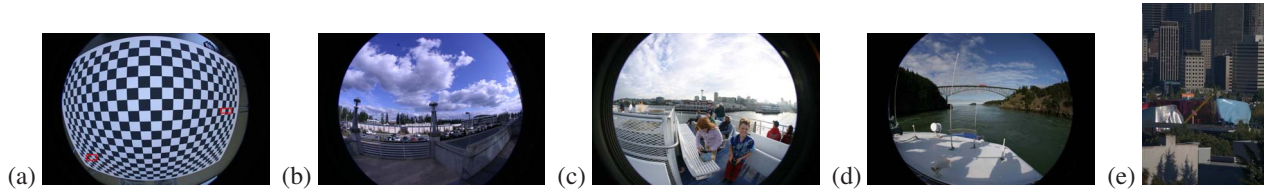


Figure 13. Images used in our experiments (in addition to the 3008×2000 image in Figure 10). The resolutions are: (a-d) 3072×2048 , and (e) 3328×4992 .

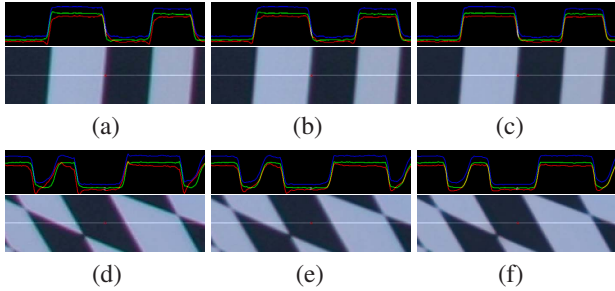


Figure 14. Results for checkerboard pattern. The original image is Figure 13(a), with close-up areas highlighted within red boxes. (a-c) Close-up views of right box, and (d-f) close-up views of left box. (a,d) Original image, (b,e) radial shift correction only, (c,f) our technique. Note that shifting alone cannot compensate for the in-camera sharpening, while our technique explicitly accounts for it. The color intensity profiles show that most of the ringing effects have been removed.

c_y ; see Section 3.1) is estimated, it took about 30 secs overall. We ran our experiments on a Pentium 4 PC with 3.2 GHz CPU and 1 GB RAM. These timings were comparable with those for the other images reported in this paper (after scaling for the number of pixels). Surprisingly, the center of the radial magnification function is very close to the image center (to within 20 pixels) in all our experiments, which suggests it is adequate to assume the radial center is coincident with the image center. Thus, in practice, we probably need only estimate seven instead of nine parameters per color channel.

The advantage of parameterizing CA is that we can apply the recovered values to correct other images that are known to be taken with the same camera under the same settings. Figure 15 shows the result of applying the recovered parameters on another image taken with the same camera. The amount of time taken was 2.5 minutes.

Three more sets of results are shown in Figure 16. Note that in these cases, the ringing effect has been significantly suppressed. Merely compensating for the shifts would have been unable to account for the ringing.

The image in Figures 13(e) and 17 is huge: its resolution is 3328×4992 . The lens used is a high-quality (and expensive) one with only about 5° vertical field of view. Despite this, because of the very high resolution, CA is noticeable,

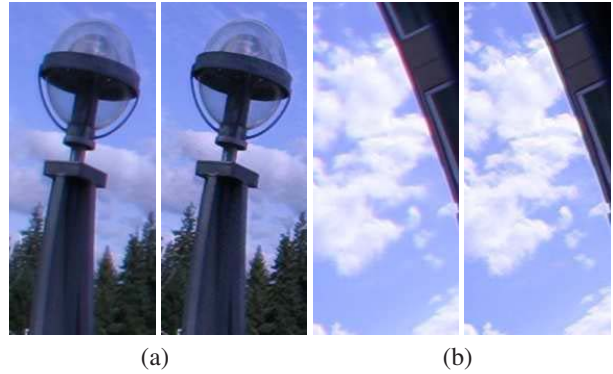


Figure 15. Result of correcting the image in Figure 13(b) using parameters recovered from the image in Figure 13(a). (a-b) Close-ups of before and after pairs. The edges in the corrected image appear substantially less reddish. In (b), the residual artifact at the edge of the building is caused by saturation (which our technique cannot handle properly at present).



Figure 17. Result of correcting a very high resolution image (Figure 13(e)). From left to right: close-ups of before, after compensating for shift only, and after applying our technique.

though only at relatively moderate levels. In this particular case, compensating for the shifts only seems adequate.

5. Discussion

As Figure 15(b) and parts of Figure 16 show, our technique does not handle saturated pixels perfectly. More generally, it cannot undo operations that resulted in loss of information due to a variety of processes such as discretization (especially at low intensities), clipping (saturation), and local averaging (cannot totally undo effects of box averag-

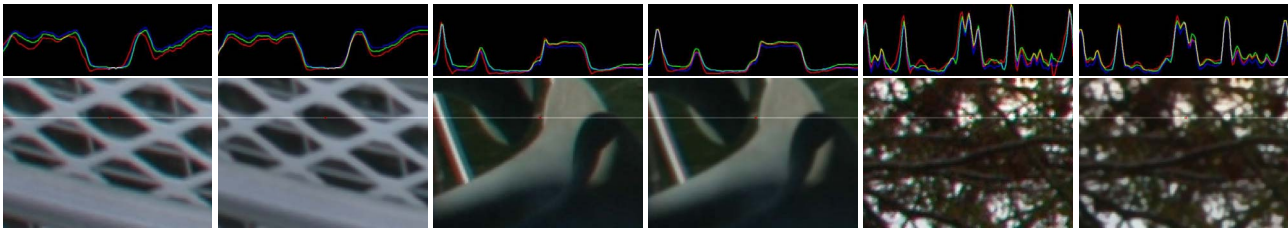


Figure 16. Close-ups of original and after applying our technique for three additional high-resolution images (each larger than 3000×2000). The top row shows the color intensity profiles associated with horizontal scanlines highlighted in the images.

ing or defocusing). However, our technique implicitly handles noise through parameter fitting over a very large number of pixels.

It is possible that the ringing effects may actually be part of the scene color distribution. In such cases, our technique will over-estimate the amount of sharpening required and hence over-sharpen the image. This may cause visible banding or cartooning effects. More specifically, if the image has finely textured areas, our technique has the tendency to blur them while making the more dominant edges stronger. There is also the artistic perspective—photos that are a bit sharpened tend to look better. However, while our technique suppresses significant ringing (which does look undesirable), if necessary, it can be forced to under-estimate the sharpening parameters (not shown in this paper).

If naive techniques for demosaicking are used (e.g., bilinear interpolation), it is possible that its effect is no longer negligible. Unfortunately, accounting for demosaicking adds considerable complexity to the analysis, as it requires that the color bands be locally correlated everywhere in some non-trivial fashion. Currently our technique only registers the color channels in the vicinity of strong edges.

Our technique does not handle purple fringing properly, which tends to occur right next to saturated regions and is only partly caused by CA. Purple fringing is significantly more complex, being produced by microlenses (lenslets directly on top of pixels, creating local CA effects). Its effect is accentuated by blooming³ and clipping.

6. Conclusions

We showed how we can characterize chromatic aberration (CA) using the basic principles of optics and imaging, and how we can use this knowledge to correct digital images with CA. We show that CA is more complicated than just relative shifting of color channels. Interestingly, in-camera sharpening contributes to the problem by causing ringing in the signal, and we explicitly model it as well.

Results show that our new model for representing CA is a reasonable one. Future topics for investigation include differentiating between fine detail and ringing artifacts to

avoid over-estimating the amount of sharpening, capitalizing on the GPU to reduce computation time, and handling pixel saturation.

Acknowledgments

I would like to thank Steve White for introducing me to the problem of chromatic aberration, Rick Szeliski for discussions on image processing, and Norman Koren for providing me pointers to articles related to image artifacts.

References

- [1] T. E. Boult and G. Wolberg. Correcting chromatic aberrations using image warping. In *IEEE Conf. on Computer Vision and Pattern Recognition*, pages 684–687, June 1992.
- [2] N. R. Farrar, A. H. Smith, D. R. Busath, and D. Taitano. In-situ measurement of lens aberrations. In *Optical Microlithography XIII (SPIE)*, volume 4000, pages 18–29, July 2000.
- [3] W. Freeman. Median filter for reconstructing missing color samples. In *U.S. Patent No. 4,724,395*, 1988.
- [4] C. Laroche and M. A. Prescott. Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients. In *U.S. Patent No. 5,629,734*, 1994.
- [5] S. K. Nayar, M. Watanabe, and M. Noguchi. Real-time focus range sensor. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(12):1186–1198, December 1996.
- [6] J. A. Nelder and R. Mead. A simplex method for function minimization. *Comput. J.*, 7:308–313, 1965.
- [7] B. H. Pillman. Impact of CCD size, pixel pitch, and anti-aliasing filter design on sharpness of digital camera print. In *Image Processing, Image Quality, Image Capture, Systems Conference (PICS)*, pages 216–220, March 2000.
- [8] S. F. Ray. *Applied Photographic Optics (3rd ed.)*. Focal Press, Woburn, MA, 2002.
- [9] G. H. Smith. *Practical Computer-Aided Lens Design*. Willmann-Bell, Inc., Richmond, VA, 1998.
- [10] W. J. Welford. *Aberrations of Optical Systems*. Adam Hilger Ltd., Bristol, UK, 1986.
- [11] R. Willson and S. A. Shafer. Active lens control for high precision computer imaging. In *Int'l Conf. on Robotics and Automation*, pages 2063–2070, April 1991.
- [12] J. C. Wyant and K. Creath. *Basic Wavefront Aberration Theory for Optical Metrology*, volume XI of *Applied Optics and Optical Engineering*, chapter 1. Academic Press, 1992.

³www.dpreview.com/learn/?/Glossary/Optical/chromatic_aberration_01.htm