

**A COLOR FILTER ARRAY INTERPOLATION
METHOD FOR DIGITAL CAMERAS USING
ALIAS CANCELLATION**

A Thesis
Presented to
The Academic Faculty

by

Vikram V. Appia

In Partial Fulfillment
of the Requirements for the Degree
Master of Science

School of Electrical and Computer Engineering
Georgia Institute of Technology
May 2008

A COLOR FILTER ARRAY INTERPOLATION METHOD FOR DIGITAL CAMERAS USING ALIAS CANCELLATION

Approved by:

Russell M. Mersereau, Advisor
School of Electrical and Computer
Engineering

Georgia Institute of Technology

Anthony J. Yezzi

School of Electrical and Computer
Engineering

Georgia Institute of Technology

Yucel Altunbasak

School of Electrical and Computer
Engineering

Georgia Institute of Technology

Date Approved: March 26, 2008.

Contents

List of Tables	vi
List of Figures	vii
SUMMARY	xi
I INTRODUCTION	1
1.1 Basic Demosaicking Methods	2
1.1.1 Bilinear Interpolation	2
1.1.2 Adams and Hamilton (AH) Method	4
1.2 Chirp Image	5
1.3 Thesis Outline	5
II SAMPLING EQUATIONS FOR A BAYER ARRAY	8
2.1 Introduction	8
2.2 Sampling equations for Rectangular Sampling Grid	9
2.3 Sampling Equation for the Quincunx Sampling Grid	13
2.4 Bayer Array Sampling Equations	14
2.5 Frequency Domain Interpretation	14
III ALIAS CANCELLATION	18
3.1 Introduction	18

3.2	Background	19
3.3	Alias Cancellation in Green Image	21
3.4	Alias Cancellation in Red/Blue Images	24
3.5	Filter Design	28
3.5.1	Filters for Green Image	28
3.5.2	Filters for Red/Blue Images	29
3.6	Results	30
IV	GREEN IMAGE INTERPOLATION	37
4.1	Introduction	37
4.2	Adaptive Homogeneity-Directed Interpolation	38
4.2.1	Generating f_H and f_V	39
4.2.2	Neighborhood Metric Model and Homogeneity Maps	42
4.3	Projection Onto Convex Sets	47
4.4	Results	49
V	RED/BLUE IMAGE INTERPOLATION	53
5.1	Introduction	53
5.2	Results	53
VI	EXPERIMENTAL RESULTS	58
6.1	Comparison	58

6.2 Color Image Results	59
VII CONCLUSION	67
REFERENCES	68

List of Tables

Table 1	MSE Results: (a) Bilinear (b) AH (c) Alias Cancellation.	32
Table 2	MSE for green image in the 24 color images from Kodak color image database.(a) AH-POCS (b) Adaptive-Homogeneity (c) Proposed Method.	51
Table 3	MSE for red and blue image in the 24 color images from Kodak color image database.(a) AH-POCS (b) Adaptive-Homogeneity (c) Proposed Method.	56
Table 4	Mean, Std. Deviation and Inter-Quartile Range (IQR) in MSE for the 24 color images in Kodak color image database. (a)AH-POCS (b) Adaptive-Homogeneity (c) Proposed Method.	59

List of Figures

Figure 1	Bayer CFA.	2
Figure 2	Bilinear Interpolation.	3
Figure 3	Adams and Hamilton method.	4
Figure 4	Chirp image.	6
Figure 5	Lighthouse image.	7
Figure 6	Bayer Sampled grids: (a) Red (b) Green (c) Blue.	8
Figure 7	Four possible offset positions for rectangular grids.	9
Figure 8	Block diagram for rectangular sampling grid with no offset.	9
Figure 9	Block diagram for rectangular sampling grid with horizontal phase shift.	11
Figure 10	Block diagram for rectangular sampling grid with vertical phase shift.	12
Figure 11	Block diagram for rectangular sampling grid with diagonal offset.	12
Figure 12	Two possible locations of quincunx sampling grid.	13
Figure 13	Four possible phase positions for rectangular grids. Shaded regions correspond to spectral copies that are multiplied by -1.	16
Figure 14	The two possible phase positions for quincunx grids. Shaded regions correspond to spectral copies that are multiplied by -1.	17
Figure 15	Example of aliasing artifacts.	19
Figure 16	Filters used for the red/blue images.	21

Figure 17	Filters used for green image.	22
Figure 18	Implementation of modulation in spatial domain. The hashed circles are negative, with a phase shift π of and the dark circles are positive. The dark circles in the output are multiplied by 2 and the empty ones by zero.	23
Figure 19	Modulation along the horizontal axis in spatial domain.	26
Figure 20	Modulation along the vertical axis in spatial domain.	27
Figure 21	Modulation along both horizontal and vertical Axes.	27
Figure 22	Filters used for Green image.	28
Figure 23	Filters used for red/blue images.	30
Figure 24	Chirp image: (a) Original (b) Bilinear (c) AH (d) Alias Cancellation.	31
Figure 25	Fence image: (a) Original (b) Bilinear (c) AH (d) Alias Cancellation.	32
Figure 26	Nyquist regions.	34
Figure 27	Alias cancellation in green channel.	35
Figure 28	Alias cancellation in red channel.	36
Figure 29	Diamond-shaped quincunx region for green image.	38
Figure 30	Bayer Array.	39
Figure 31	Red-Green and Green-Blue rows indicating missing green values.	39
Figure 32	1-D lowpass filter response.	40
Figure 33	Horizontal and Vertical filtered Chirp image.	40

Figure 34	Horizontal and Vertical filtered Lighthouse image.	41
Figure 35	Homogeneity maps of f_H and f_V , Chosen direction for interpolation.	45
Figure 36	Green image after appropriate directions have been selected.	46
Figure 37	Result of green interpolation on chirp image.	46
Figure 38	POCS block diagram.	47
Figure 39	Simplified block diagram.	48
Figure 40	The LL and H_{eff} filters used in POCS.	48
Figure 41	Green image generated by POCS(Using Adaptive-Homogeneity directed edge-based interpolation).	49
Figure 42	Result of green interpolation on chirp image.	52
Figure 43	Result of red interpolation on Fence image.	54
Figure 44	Red Image: Green image generated in Chapter (4) was used to cancel aliasing.	54
Figure 45	Result of red interpolation on chirp image.	57
Figure 46	Aliasing present in the Fence image and error image.	59
Figure 47	Average MSE over the 24 color images.	60
Figure 48	Result on chirp image.	61
Figure 49	Result on chirp image.	62
Figure 50	Lighthouse image.	63
Figure 51	Lighthouse image (contd.)	64

Figure 52 Houses image.	65
Figure 53 Houses image (contd.)	66

SUMMARY

To reduce cost, many digital cameras use a single sensor array instead of using three arrays for the red, green and blue. Thus at each pixel location only the red, green or blue intensity value is available. And to generate a complete color image, the camera must estimate the missing two values at each pixel location. Color filter arrays are used to capture only one portion of the spectrum (Red, Green or Blue) at each location. Various arrangements of the Color Filter Array (CFA) are possible, but the Bayer array is the most commonly used arrangement and we will deal exclusively with the Bayer array in this thesis.

Since each of the three colors channels are effectively downsampled, it leads to aliasing artifacts. This thesis will analyze the effects of aliasing in the frequency-domain and present a method to reduce the deterioration in image quality due to aliasing artifacts.

Two reference algorithms, AH-POCS (Adams and Hamilton - Projection Onto Convex Sets) and Adaptive Homogeneity-Directed interpolation, are discussed in detail. Both algorithms use the assumption that there is high correlation in the high-frequency regions to reduce aliasing. AH-POCS uses alias cancellation technique to reduce aliasing in the red and blue images, while the Adaptive Homogeneity-Directed interpolation algorithm is an edge-directed algorithm. We present here an algorithm that combines these two techniques and provides a better result on average when compared to the reference algorithms.

Chapter I

INTRODUCTION

A digital color image in a digital camera requires the red, green and blue (R,G,B) intensity value at each pixel location. Hence a camera would require three sensors to capture the complete information required for a digital image. But to reduce the cost, digital cameras generally use only one sensor array with a color filter array in the path between the lens and the sensor. Instead of measuring the red, green, and blue values at every pixel, the color filters built onto each pixel will capture only one portion of the visible spectrum (red, green or blue).

Many arrangements of Color Filter Array (CFA) exist, but the Bayer CFA [1] is the most commonly used and this thesis will deal exclusively with Bayer CFA. Figure 1 shows the arrangement of R, G and B for a Bayer array. The green is measured at half the total number of pixels. Thus, the green is sampled at twice the rate of the red and blue samples. This is because the human visual is more sensitive to changes in green than red and blue.

To generate a complete image these cameras must estimate (interpolate) the missing values at each pixel location; this process is referred to as demosaicking or CFA interpolation. Each of the three colors are downsampled in the Bayer array which results in aliasing. This aliasing leads to color artifacts in the image containing sharp edges when interpolated. The thesis presents a method to reduce these aliasing artifacts.

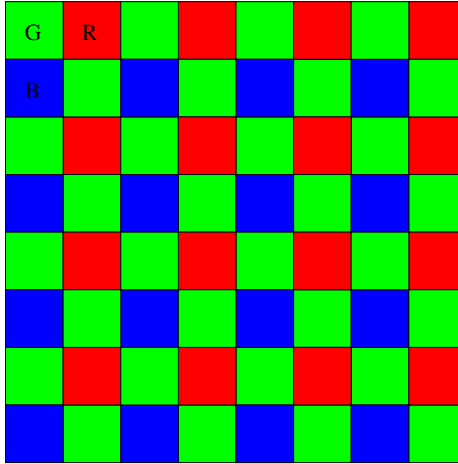


Figure 1: Bayer CFA.

1.1 *Basic Demosaicking Methods*

1.1.1 Bilinear Interpolation

This is the simplest and most intuitive method of demosaicking. It treats each of the three color images (R,G,B) independently and uses typical bilinear interpolation on each of these images to generate the full color image. Thus, it takes the average of each of the neighboring pixel to estimate the missing value, which is illustrated in Figure 2. Each of the missing green values has four known neighbors, but for red and blue we will come across three scenarios as shown in Figure 2(b),(c) and (d). Thus the missing green values are calculated as below:

$$\hat{g} = \frac{g_1 + g_2 + g_3 + g_4}{4}.$$

This can be incorporated into a filter kernel given by,

$$H_g = \begin{bmatrix} 0 & 1/4 & 0 \\ 1/4 & 1 & 1/4 \\ 0 & 1/4 & 0 \end{bmatrix}.$$

The interpolation can be simplified to a simple convolution of the green image with this filter kernel.

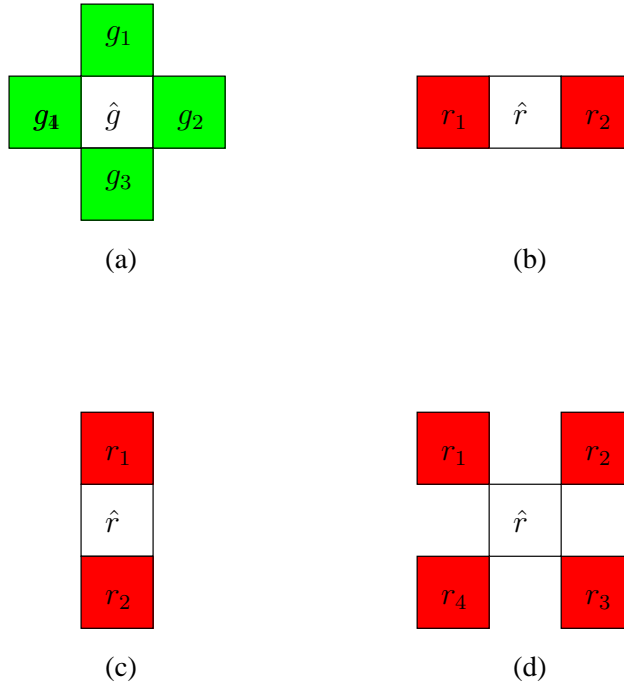


Figure 2: Bilinear Interpolation.

If a missing red location has two known neighbors as in Figure 2(b),(c) we take the average,

$$\hat{r} = \frac{r_1 + r_2}{2}$$

while with four known neighbors it becomes,

$$\hat{r} = \frac{r_1 + r_2 + r_3 + r_4}{4}.$$

These can be combined to form a simple filtering kernel given by,

$$H_r = \begin{bmatrix} 1/4 & 1/2 & 1/4 \\ 1/2 & 1 & 1/2 \\ 1/4 & 1/2 & 1/4 \end{bmatrix}.$$

Figure 5(a) shows the result of bilinear interpolation on the lighthouse test image. We see color artifacts along the wall of the house and the picket fence. The edges present generate aliasing and create these disturbing artifacts. In addition, the edges in the final output image look blurred.

1.1.2 Adams and Hamilton (AH) Method

This method, explained in [2], uses local edge information from the other channels to improve the interpolation. Here, at each pixel the orientation of an edge is detected and interpolation is performed along the edge. To classify each pixel, a second-order gradient along the horizontal and vertical directions are computed,

$$C_h = 2|g_2 - g_4| + |2r_3 - r_1 - r_5|$$

$$C_v = 2|g_7 - g_8| + |2r_3 - r_6 - r_9|$$

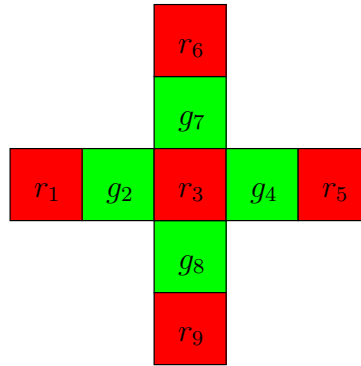


Figure 3: Adams and Hamilton method.

And if $C_h > C_v$ then the pixel is interpolated in the horizontal direction and if $C_h \leq C_v$ vertical interpolation is used. To the interpolated green value a correction term from the red image is added,

$$g_3 = \frac{1}{2}(g_2 + g_4) + \frac{1}{2}(2r_3 + r_1 + r_5) \text{ or}$$

$$g_3 = \frac{1}{2}(g_7 + g_8) + \frac{1}{2}(2r_3 + r_6 + r_9).$$

The result of using this method is shown in Figure 5(b). We see a large improvement in regions along the fence and the wall of the house just by choosing the direction of interpolation intelligently.

1.2 Chirp Image

A chirp signal sweeps across the entire frequency space. The synthetic chirp image shown in Figure 4(a) is a two-dimensional chirp signal that sweeps the frequency space given by,

$$[\omega_1, \omega_2] : -\pi \leq \omega_1 < \pi, -\pi \leq \omega_2 < \pi.$$

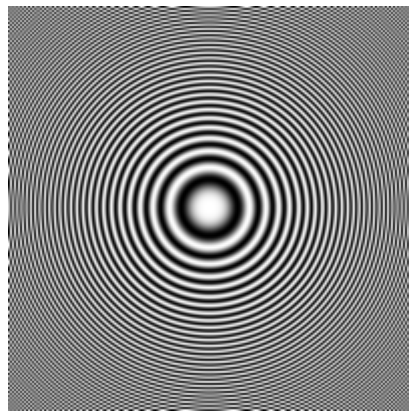
In the discrete spatial domain the intensity at each pixel location, $I(u, v)$, will be

$$I(u, v) = \cos(u^2 + v^2).$$

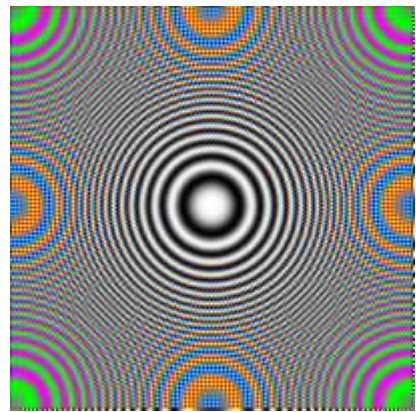
Thus, the instantaneous frequency at a given location of the image corresponds to the spatial frequencies at that co-ordinate in the frequency space. This makes the chirp image very useful in analyzing the problem in the frequency domain. Figures 4(b) and (c) show the result of bilinear and AH interpolation on the chirp image. The aliasing present in the image is caused by subsampling the image and improperly reconstructing it. The high-frequency components are reconstructed as low-frequency components causing distortions in the reconstructed chirp image that correspond to its high-frequency regions.

1.3 Thesis Outline

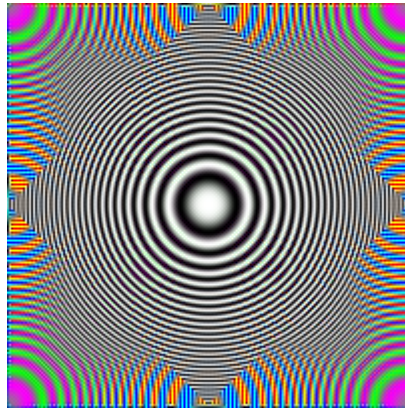
We analyze the demosaicking problem using sampling theory in Chapter 2. The reference algorithm AH-POCS [3], which is discussed, uses the fact that the green image is sampled at a higher frequency than the red and blue images. Hence the green image retains much more high frequency information. Also, the algorithm assumes the green image is band-limited to its Nyquist sampling frequency. The information from the green channel is used to cancel aliasing that occurs in the red and blue channels based on the assumption that all three channels are highly correlated [4] in the high frequency regions. In most cases the assumption that the green signal



(a) Chirp Image



(b) Bilinear Interpolation



(c) AH

Figure 4: Chirp image.

is bandlimited is not true. Since none of the green, blue or red images are sampled adequately, interpolation leads to residual aliasing artifacts in all three channels.

We then derive equations to cancel aliasing assuming that the given green image is not band-limited in Chapter 3. Then in Chapter 4 we will try to improve the green interpolation by reducing the aliasing directly. Finally, we use the interpolated green image to cancel aliasing in the red/blue images in Chapter 5. And Chapter 6 compares the various algorithms quantitatively on a database of 24 standard color images. Chapter 7 concludes the thesis and discusses future research possibilities.



(a) Bilinear Interpolation



(b) AH

Figure 5: Lighthouse image.

Chapter II

SAMPLING EQUATIONS FOR A BAYER ARRAY

2.1 Introduction

In this chapter we derive the sampling equations to describe the system and understand the frequency-domain interpretation of these equations. As introduced in the previous chapter, a Bayer color filter array is used to capture the information in most digital cameras. The sampled grid generated has interleaved samples of the red, green and blue color planes in a single array. To analyze and understand the problem better we will use three separate grids for red, green and blue as shown in Figure 6. The grids are shown with respect to the full sampling grid of the image; the open circles in each grid indicate missing grid values that must be interpolated to obtain a full color image. It is worth noticing that the red and blue have rectangular sampling grids, whereas the green has a quincunx sampling grid.

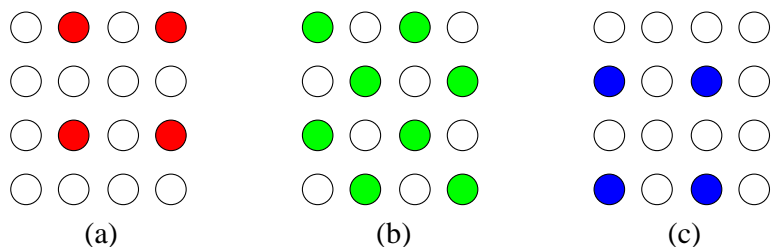


Figure 6: Bayer Sampled grids: (a) Red (b) Green (c) Blue.

We begin by deriving the sampling equations for a rectangular grid. Once this equation is obtained, it is only a matter of applying the correct phase shift to obtain the sampling equations for red and blue. The green sampling grid can also be obtained by combining two rectangular grids, as will be discussed in detail in this chapter. We discuss the frequency-domain representation for all possible phase shifts of the

rectangular grid and the two possible phase shifts of the quincunx grids at the end of this chapter.

2.2 Sampling equations for Rectangular Sampling Grid

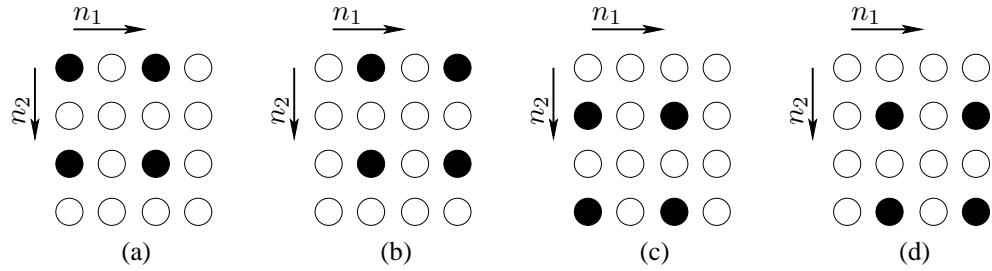


Figure 7: Four possible offset positions for rectangular grids.

Figure 7 shows the four possible offset positions for the rectangular sampling grid. We index the grid points from 0 along the n_1 and n_2 axes. Notice that the Figure 7(b) resembles the red sampling grid in the Bayer array and Figure 7(c) resembles the blue sampling grid. Adding Figure 7(a) and Figure 7(d) will give the green sampling grid.

We will derive the equation for the rectangular sampling system given in Figure 7(a) first and then give corresponding phase shifts to derive the other equations. Figure 8 shows a block diagram for a rectangular sampled grid with no phase shift. This system can be described by:

$$x_a[n_1, n_2] = \begin{cases} x[n_1, n_2], & \text{if } n_1, n_2 \text{ even} \\ 0 & \text{otherwise} \end{cases}$$



Figure 8: Block diagram for rectangular sampling grid with no offset.

We look at the Fourier transform interpretation of this equation, as we are interested in its frequency-domain representation. Using the derivation in [5], we define the 2-D Fourier transform of $X_s[n_1, n_2] = x[Kn_1, Ln_2]$, the downsampling operation that downsamples by a factor of K in the horizontal direction and a factor of L in the vertical direction as,

$$X(e^{j\omega_1}, e^{j\omega_2}) = \frac{1}{KL} \sum_{i_1=0}^{K-1} \sum_{i_2=0}^{L-1} X(e^{j\omega_1/K - 2\pi i_1/K}, e^{j\omega_2/K - 2\pi i_2/L}). \quad (1)$$

The Fourier transform of

$$x_a[n_1, n_2] = \begin{cases} x[n_1/K, n_2/L], & \text{if } n_1 = iK, n_2 = jL \text{ where } i, j \text{ are integers} \\ 0 & \text{otherwise} \end{cases}$$

which represent the corresponding upsampling operation, is

$$X_a(e^{j\omega_1}, e^{j\omega_2}) = X(e^{jK\omega_1}, e^{jL\omega_2}). \quad (2)$$

For the rectangular grids shown in Figure 7, we have K=L=2. Substituting this value into (1), we get

$$\begin{aligned} X_s(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} X_s(e^{j\omega_1/2}, e^{j\omega_2/2}) + \frac{1}{4} X_s(e^{j\omega_1/2 - \pi}, e^{j\omega_2/2}) \\ &\quad + \frac{1}{4} X_s(e^{j\omega_1/2}, e^{j\omega_2/2 - \pi}) + \frac{1}{4} X_s(e^{j\omega_1/2 - \pi}, e^{j\omega_2/2 - \pi}). \end{aligned} \quad (3)$$

Using K=L=2 in (2) we get $X_a(e^{j\omega_1}, e^{j\omega_2}) = X_s(e^{j2\omega_1}, e^{j2\omega_2})$,

and the overall sampling equation becomes

$$\begin{aligned} X_a(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} X_s(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{4} X_s(e^{j(\omega_1 - \pi)}, e^{j\omega_2}) \\ &\quad + \frac{1}{4} X_s(e^{j\omega_1}, e^{j(\omega_2 - \pi)}) + \frac{1}{4} X_s(e^{j(\omega_1 - \pi)}, e^{j(\omega_2 - \pi)}). \end{aligned} \quad (4)$$

Now consider Figure 7(b). Here the grid is shifted one position in the horizontal direction. The block diagram for this system is shown in Figure 9. Here the signal is

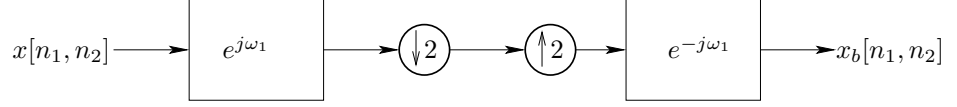


Figure 9: Block diagram for rectangular sampling grid with horizontal phase shift.

advanced one position in the horizontal direction before downsampling and then it is delayed once in the horizontal direction after it is upsampled.

The Fourier transform of the input to the downsampling block will be $e^{j\omega_1} X(e^{j\omega_1}, e^{j\omega_2})$.

Downsampling gives

$$\begin{aligned} X(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} e^{j(\omega_1/2)} X(e^{j(\omega_1/2)}, e^{j(\omega_2/2)}) + \frac{1}{4} e^{j(\omega_1/2-\pi)} X(e^{j(\omega_1/2-\pi)}, e^{j(\omega_2/2)}) \\ &\quad + \frac{1}{4} e^{j(\omega_1/2)} X(e^{j(\omega_1/2)}, e^{j(\omega_2/2-\pi)}) + \frac{1}{4} e^{j(\omega_1/2-\pi)} X(e^{j(\omega_1/2-\pi)}, e^{j(\omega_2/2-\pi)}). \end{aligned} \quad (5)$$

Taking, $e^{j\omega_1/2-\pi} = -e^{j\omega_1/2}$ in the second and fourth terms of (5), we get

$$\begin{aligned} X(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} e^{j(\omega_1/2)} X(e^{j(\omega_1/2)}, e^{j(\omega_2/2)}) - \frac{1}{4} e^{j(\omega_1/2)} X(e^{j(\omega_1/2-\pi)}, e^{j(\omega_2/2)}) \\ &\quad + \frac{1}{4} e^{j(\omega_1/2)} X(e^{j(\omega_1/2)}, e^{j(\omega_2/2-\pi)}) - \frac{1}{4} e^{j(\omega_1/2)} X(e^{j(\omega_1/2-\pi)}, e^{j(\omega_2/2-\pi)}). \end{aligned} \quad (6)$$

Upsampling this signal we get

$$\begin{aligned} X(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} e^{j\omega_1} X(e^{j\omega_1}, e^{j\omega_2}) - \frac{1}{4} e^{j\omega_1} X(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad + \frac{1}{4} e^{j\omega_1} X(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - \frac{1}{4} e^{j\omega_1} X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \end{aligned} \quad (7)$$

Delaying the signal one sample horizontally is equivalent to multiplying the Fourier transform of the signal by $e^{-j\omega_1}$, which gives

$$\begin{aligned} X_b(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} X(e^{j\omega_1}, e^{j\omega_2}) - \frac{1}{4} X(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad + \frac{1}{4} X(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - \frac{1}{4} X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \end{aligned} \quad (8)$$

Eqn.(8) gives the sampling equation for grid in Figure 7(b).

In Figure 7(c) the signal is advanced one position in the vertical direction before downsampling and then is delayed one sample in the vertical direction before it is upsampled. The block diagram for this system is shown in Figure 10. Following a similar analysis we can derive the final sampling equation,

$$\begin{aligned} X_c(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4}X(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{4}X(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad - \frac{1}{4}X(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - \frac{1}{4}X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \end{aligned} \quad (9)$$

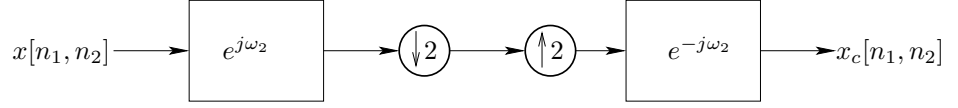


Figure 10: Block diagram for rectangular sampling grid with vertical phase shift.

In Figure 7(d) the signal is advanced one position in both the vertical and horizontal directions before downsampling and then is delayed once in both directions before it is upsampled. The block diagram for this system is given in Figure 11. Here the samples are multiplied by a factor of $e^{-j\omega_1}e^{-j\omega_2}$ after upsampling to give the final sampling equation,

$$\begin{aligned} X_d(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4}X(e^{j\omega_1}, e^{j\omega_2}) - \frac{1}{4}X(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad - \frac{1}{4}X(e^{j\omega_1}, e^{j(\omega_2-\pi)}) + \frac{1}{4}X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \end{aligned} \quad (10)$$

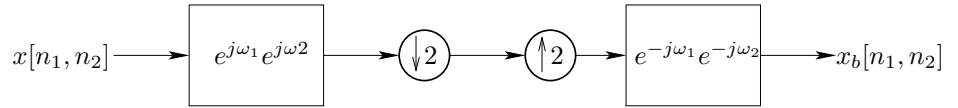


Figure 11: Block diagram for rectangular sampling grid with diagonal offset.

2.3 Sampling Equation for the Quincunx Sampling Grid

As mentioned earlier adding Figure 7(a) and Figure 7(d) will give a quincunx sampling grid resembling the Green image in the Bayer array as in Figure 12(a). Thus we have

$$x_q[n_1, n_2] = x_a[n_1, n_2] + x_d[n_1, n_2].$$

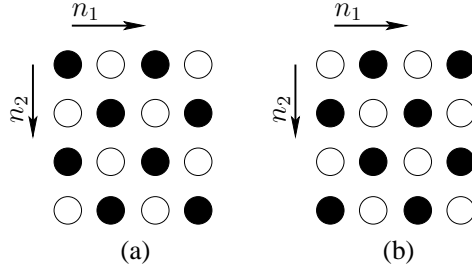


Figure 12: Two possible locations of quincunx sampling grid.

Adding (5) and (10), we get

$$X_q(e^{j\omega_1}, e^{j\omega_2}) = X_a(e^{j\omega_1}, e^{j\omega_2}) + X_d(e^{j\omega_1}, e^{j\omega_2})$$

$$\begin{aligned} X_q(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4}X(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{4}X(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad + \frac{1}{4}X(e^{j\omega_1}, e^{j(\omega_2-\pi)}) + \frac{1}{4}X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}) \\ &\quad + \frac{1}{4}X(e^{j\omega_1}, e^{j\omega_2}) - \frac{1}{4}X(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad - \frac{1}{4}X(e^{j\omega_1}, e^{j(\omega_2-\pi)}) + \frac{1}{4}X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}) \end{aligned}$$

which simplifies to,

$$X_q(e^{j\omega_1}, e^{j\omega_2}) = \frac{1}{2}X(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{2}X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \quad (11)$$

Similarly, the shifted quincunx grid shown in Figure 12(b) can be thought of as the addition of Figure 7(b) and Figure 7(c) giving us the sampling equation,

$$\hat{X}_q(e^{j\omega_1}, e^{j\omega_2}) = \frac{1}{2}X(e^{j\omega_1}, e^{j\omega_2}) - \frac{1}{2}X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \quad (12)$$

2.4 Bayer Array Sampling Equations

We will translate the sampling equations derived in the previous sections to the red, green and blue images in a Bayer Array. The red resembles the rectangular grid with one horizontal shift and using (8) we have

$$\begin{aligned} R_s(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4}R(e^{j\omega_1}, e^{j\omega_2}) - \frac{1}{4}R(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad + \frac{1}{4}R(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - \frac{1}{4}R(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \end{aligned} \quad (13)$$

Similarly, the blue resembles the rectangular grid with one vertical shift given in (9)

$$\begin{aligned} B_s(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4}B(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{4}B(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad - \frac{1}{4}B(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - \frac{1}{4}B(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}), \end{aligned} \quad (14)$$

and the green is the quincunx grid with no shift given in (11)

$$G_s(e^{j\omega_1}, e^{j\omega_2}) = \frac{1}{2}G(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{2}G(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \quad (15)$$

2.5 Frequency Domain Interpretation

In a Bayer array the red, blue and green images are sampled at a rate lower than the full sampled array. The red and blue, sampled on rectangular grids, have $1/4^{th}$ of the full sampling rate and the green, sampled on a quincunx grid has $1/2$ the sampling rate. Because of this, we will see spectral copies (aliases) at periodic locations in the frequency domain. The Nyquist region for these grids are shown in Figure 13(a) and Figure 14(a).

Since the red/blue are sampled $1/4^{th}$ sampling rate, there are four possible location for the sampling grid as explained in Figure 7. Corresponding to each of these locations the spectral copies at different locations have different phase inversions as shown in Figure 13(b)-(e). Figure 13(b) corresponds to the sampling grid Figure 7(a) with no offset. Here all the spectral copies have same phase as the original spectrum.

Figure 13(c) corresponds to the sampling grid in Figure 7(b) with an offset along the horizontal direction. Here the spectral copies with a horizontal shift have a phase inversion and it is indicated by the shaded region. Figure 13(d) corresponds to the sampling grid in Figure 7(c) with an offset along the vertical direction. Here the spectral copies with a vertical shift have a phase inversion. Figure 13(e) corresponds to the sampling grid in Figure 7(d) with a horizontal and vertical offset. Here the spectral copies with a horizontal or vertical shift have a phase inversion and the copies with both horizontal and vertical shifts have the same phase as the original spectrum. It is worth noting that these images are graphical representations of (4),(8),(9) and (10).

The green is sampled at half the rate on a quincunx grid, so it has two possible locations for the sampling grid. Figure 14(b) corresponds to sampling grid with no phase shift shown in Figure 12(a). Figure 14(c) corresponds to sampling grid with a phase shift as shown in Figure 12(b). Again we note that these figures represent equations (11) and (12).

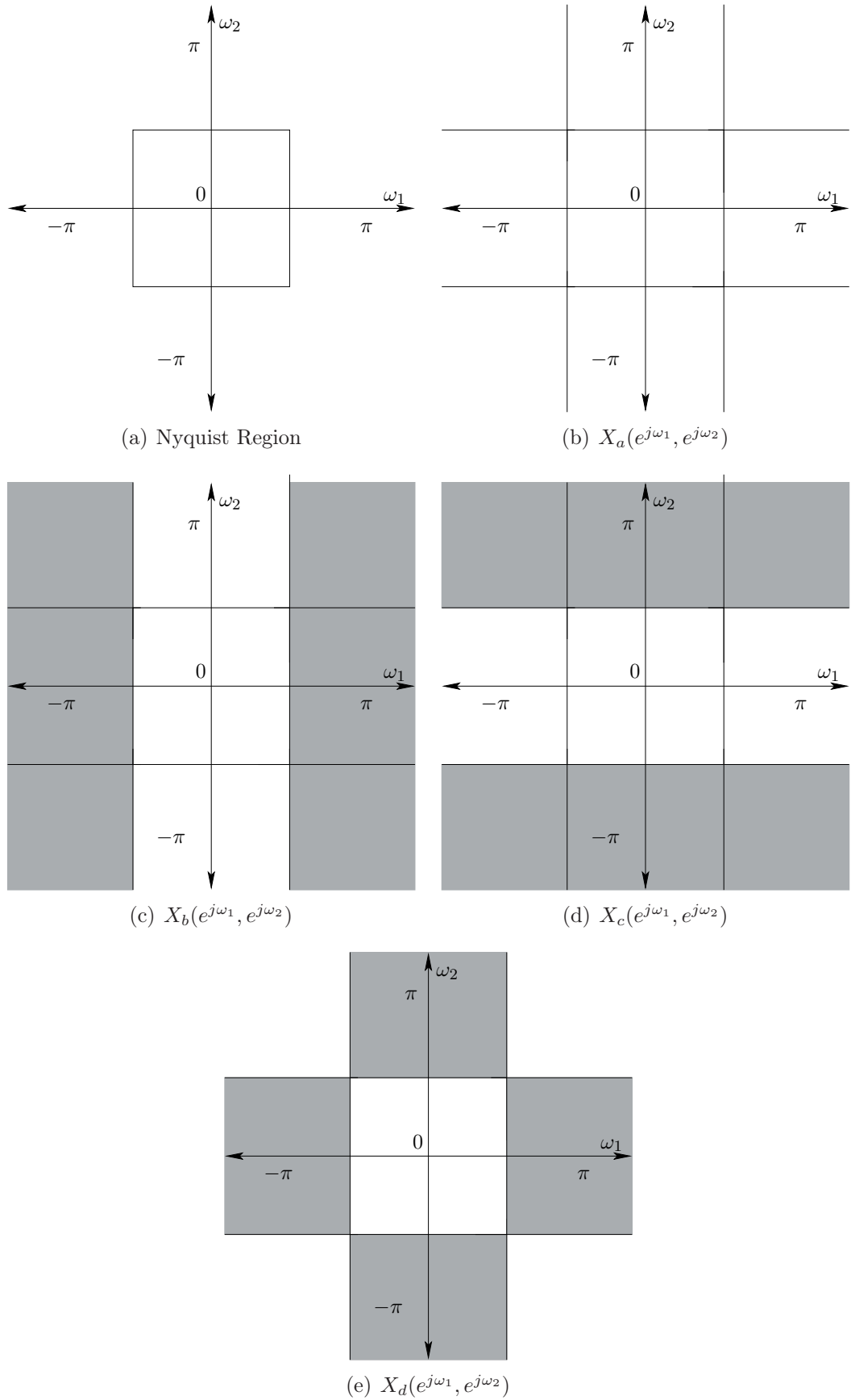


Figure 13: Four possible phase positions for rectangular grids. Shaded regions correspond to spectral copies that are multiplied by -1.

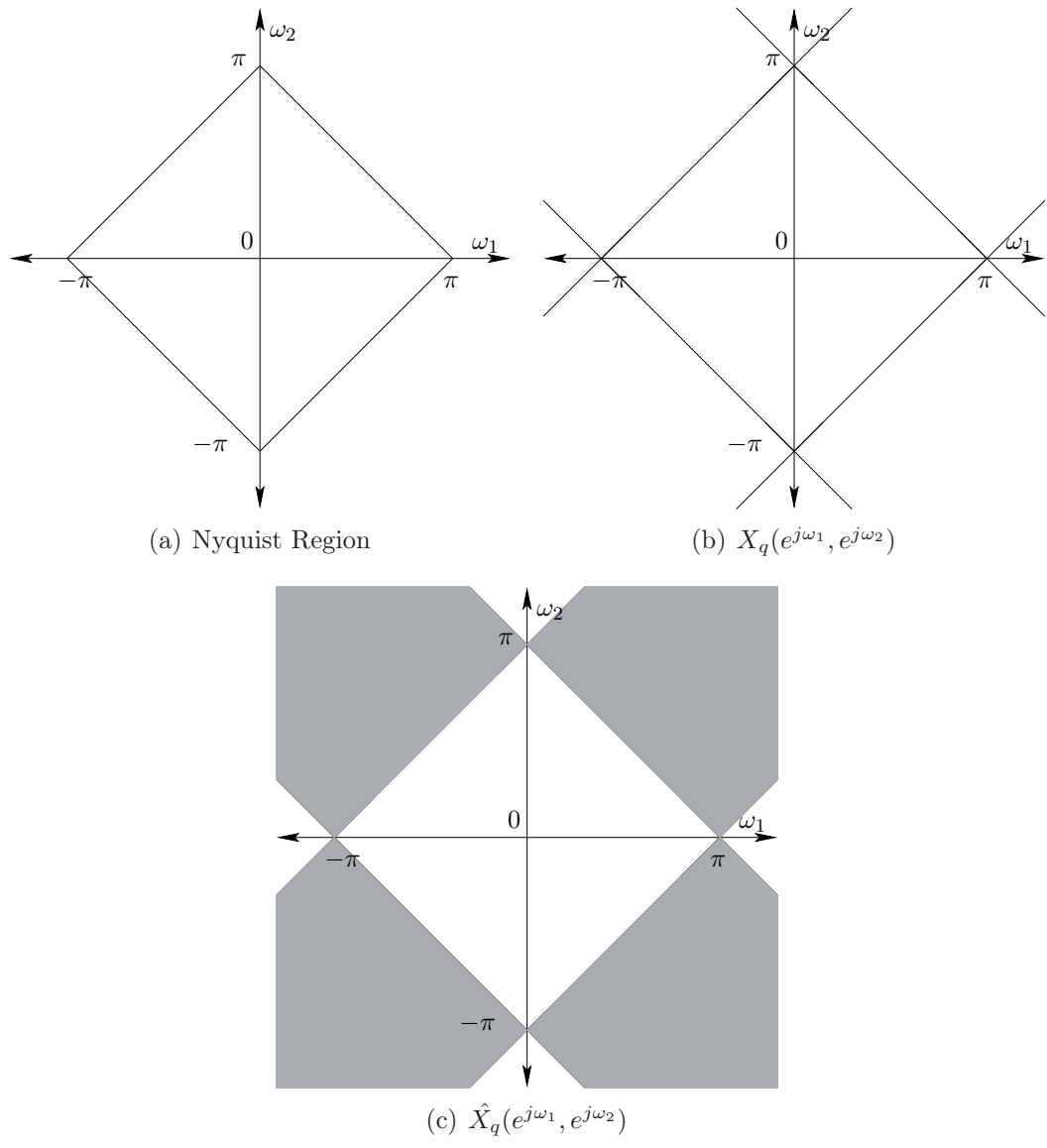


Figure 14: The two possible phase positions for quincunx grids. Shaded regions correspond to spectral copies that are multiplied by -1.

Chapter III

ALIAS CANCELLATION

3.1 Introduction

In the previous chapter we derived the sampling equations for the red, green and blue channels for a Bayer array. Equations (13), (14) and (15) indicate the presence of aliasing in the red, green and blue images. In this chapter we will show how we can cancel aliasing that is present in each of these images provided we have a full sampled array, X . Since the red and blue images are sampled on rectangular sampling grids at half the rate of the green image we will derive two sets of equations, one to cancel aliasing in the red/blue images and the other to cancel aliasing in the green image.

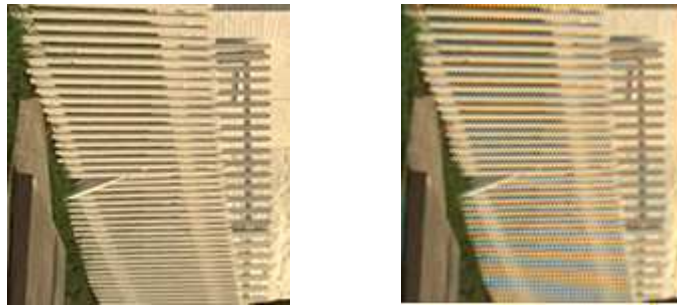
This analysis is based on the assumption that the high frequency information in red, green and blue channels are highly correlated [4]. We consider X , the fully sampled array, to have high frequency content similar to the three channels. We will isolate the high frequency components from X and modulate them appropriately to estimate the aliasing that is present in the red, green and blue images. Using this estimate, we cancel the aliasing present and also add the high frequency content to improve the sharpness of these images.

In reality we do not have the fully sampled array, X , but the test images used in our experiments do have fully sampled red, green and blue images. The Bayer sampled array is generated from these images to test the CFA interpolation algorithms. We will use one of the fully sampled images as our X and test the alias cancellation algorithm derived in this chapter. Using this simple experiment, we test how effective the alias cancellation equations are and in the process we will also validate the assumption of high correlation in the high-frequency regions.

3.2 Background

Figure 26(a), (b) and (c) shows the Nyquist sampling bandwidth limits for the fully sampled image, X , the red/blue images, and the green image respectively. The Nyquist sampling region for the green image consists half the spectral area of the fully sampled image and that for the red/blue images consists $(\frac{1}{4})^{th}$ of the spectral area. This can be observed in Figure 26(d) and Figure 26(e) where the missing high frequency regions are highlighted.

The ideal interpolation filters have the same shape as the Nyquist regions shown. Thus, the high frequency content is not reconstructed in the final image which leads to its blurred edges. Aliasing is caused because the high frequencies are not blocked before sampling. The ideal lowpass filters used to reconstruct will not be able to distinguish between the low-frequency components and the aliased components, which leads to aliasing artifacts as shown in Figure 15. These aliasing artifacts become more conspicuous because of the inherent phase shift between the red and blue sampled images. Bilinear interpolation is used to generate the image to illustrate aliasing.



(a) Original

(b) Bilinear Interpolation

Figure 15: Example of aliasing artifacts.

We consider X to be a fully sampled array which does not have any aliasing in it. Using appropriate highpass filters we will isolate the high-frequency components.

Then we can modulate these high-frequency components into the low-frequency regions and cancel the aliasing present in the red, green and blue channels.

Figure 27 illustrates the process of canceling aliasing in the green channel. Consider an image that contains an edge oriented along a diagonal, this will generate a high-frequency component as shown in Figure 27(a). Interpolating with the quincunx sampling grid of green would replicate the spectrum once with a horizontal shift and a vertical shift as shown in Figure 27(b). Since the aliased components are replicated in the low-frequency they are present in the interpolated output image. Figure 27(c) shows the highpass filter applied to the full sampled array to isolate the high frequency components. Then modulating the filtered output by $(-1)^{n_1+n_2}$ shifts the high-frequency components into the low-frequency region as shown in Figure 27(d). And since we are assuming the high frequency content to be similar in all the images, we can use this modulated output to cancel aliasing in the green image. Also we will add the filtered high frequency components to the green image as shown in Figure 27(e).

Since the red and blue channels are identical except for the phase shift, we will discuss canceling aliasing in the red channel only. Figure 28 illustrates the process of canceling aliasing in the red channel. We will consider a feature in the image that generates a high-frequency component along the horizontal axis as shown in Figure 28(a). Interpolating the red image with a rectangular sampling grid will replicate the feature three times by shifting it horizontally, vertically and both horizontally and vertically. This is illustrated in Figure 28(b). Since the aliased components are replicated thrice, we need three highpass filters to cancel the aliasing. Figure 28 shows the process of canceling aliasing using the horizontal highpass filter and a similar procedure can be used with the remaining two filters to cancel the aliasing completely from the red image. Figure 28(c) shows the horizontal highpass filter applied to the full sampled array to isolate the high frequency components. Then

modulating the filtered output by $(-1)^{n_1}$ shifts the high-frequency components into the low-frequency region as shown in Figure 28(d). Then we add these high-frequency components to increase the sharpness of the red image as shown in Figure 28(e).

Figure 16 shows the three highpass filters that are applied to cancel aliasing in the red image. We will modulate the filter marked by region 'c' and the filter marked by region 'd' with $(-1)^{n_2}$ and $(-1)^{n_1+n_2}$ respectively to shift their spectra into the low-frequency region.

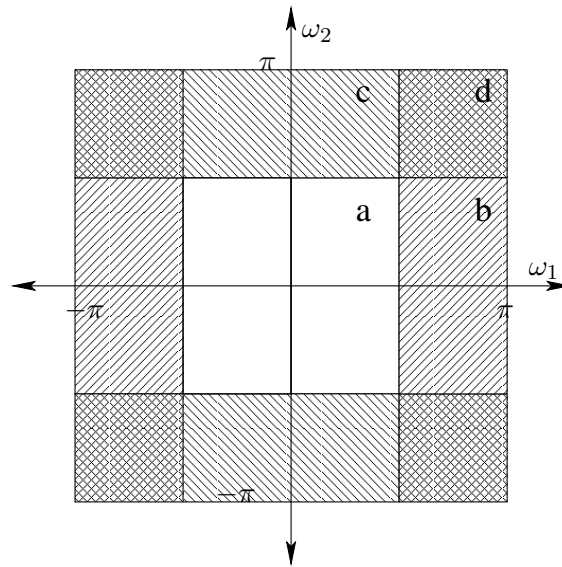


Figure 16: Filters used for the red/blue images.

3.3 Alias Cancellation in Green Image

Consider the fully sampled array $X(e^{j\omega_1}, e^{j\omega_2})$. The equation for the sampled green image is as given below:

$$G_s(e^{j\omega_1}, e^{j\omega_2}) = \frac{1}{2}G(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{2}G(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}).$$

Since the Nyquist region for green is a diamond shaped region, we will consider using the two filters H_g and H_{hp} shown in Figure 17. The filter H_g has a gain of 2 in the

passband while the filter H_{hp} has a stopband in the diamond shaped region and a gain of 1 in its passband.

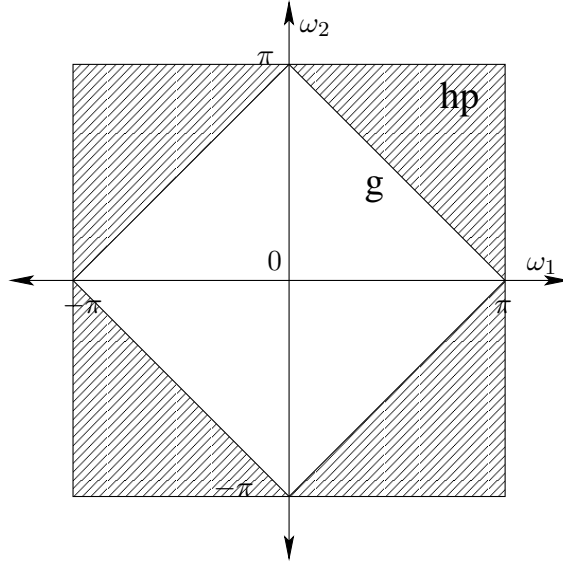


Figure 17: Filters used for green image.

Applying the highpass filter H_{hp} to X to isolate the high-frequency components gives

$$X_{hp}(e^{j\omega_1}, e^{j\omega_2}) = X(e^{j\omega_1}, e^{j\omega_2})H_{hp}(e^{j\omega_1}, e^{j\omega_2}). \quad (16)$$

We will modulate this into the low-frequency region by shifting it horizontally and vertically to get

$$X_{hp}^m(e^{j\omega_1}, e^{j\omega_2}) = X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})H_{hp}(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \quad (17)$$

Interpolating the green image with the lowpass filter, H_g , will give

$$G_i(e^{j\omega_1}, e^{j\omega_2}) = \frac{1}{2}[G(e^{j\omega_1}, e^{j\omega_2}) + G(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})]H_g(e^{j\omega_1}, e^{j\omega_2}). \quad (18)$$

The aliased components in the interpolated green image are contained in the term $G(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})H_g(e^{j\omega_1}, e^{j\omega_2})$ and in the modulated X image it is contained in the term $X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})H_{hp}$. Since we would like to cancel aliasing, we will subtract the modulated high-frequency region from the green image. Also, we would

like to add the correct high-frequency components which by assumption we get from $X(e^{j\omega_1}, e^{j\omega_2})H_{hp}(e^{j\omega_1}, e^{j\omega_2})$. Thus taking (18)-(17)+(16) will give the final green image

$$\hat{G}(e^{j\omega_1}, e^{j\omega_2}) = G_i(e^{j\omega_1}, e^{j\omega_2}) - X_{hp}^m(e^{j\omega_1}, e^{j\omega_2}) + X_{hp}(e^{j\omega_1}, e^{j\omega_2}). \quad (19)$$

Implementation

In the spatial domain, modulating the high-frequency components once both in the horizontal and vertical directions involves multiplying the terms with $(-1)^{n_1+n_2}$. Thus in spatial domain (18) becomes

$$\hat{g}[n_1, n_2] = g_i[n_1, n_2] - (-1)^{n_1+n_2}x_{hp}[n_1, n_2] + x_{hp}[n_1, n_2]. \quad (20)$$

Considering the components added from image X we have the term

$$[1 - (-1)^{n_1+n_2}]x_{hp}[n_1, n_2].$$

When $n_1 + n_2$ is even this term becomes zero; When $n_1 + n_2$ is odd it becomes $2 * x_{hp}[n_1, n_2]$. This is illustrated in Figure 18

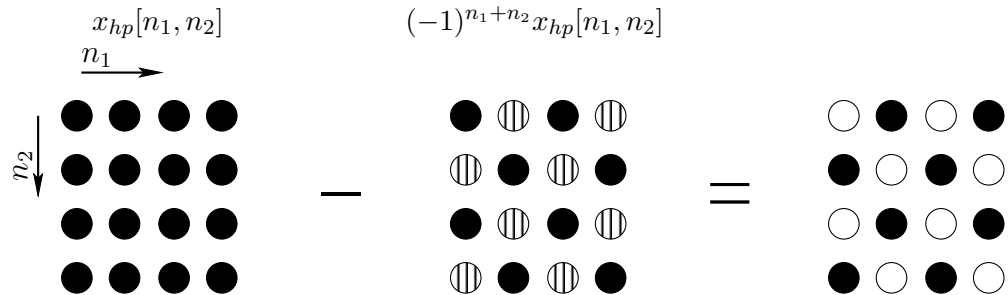


Figure 18: Implementation of modulation in spatial domain. The hashed circles are negative, with a phase shift π of and the dark circles are positive. The dark circles in the output are multiplied by 2 and the empty ones by zero.

Effectively it adds zero at locations where original green from the Bayer array is present and adds the alias correction term at the interpolated samples of green.

3.4 Alias Cancellation in Red/Blue Images

The sampling equation for the red image is

$$R_s(e^{j\omega_1}, e^{j\omega_2}) = \frac{1}{4}R(e^{j\omega_1}, e^{j\omega_2}) - \frac{1}{4}R(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ + \frac{1}{4}R(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - \frac{1}{4}R(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}).$$

Red is sampled on a rectangular sampling grid and its Nyquist region is as shown in Figure 7(a). As explained in the previous section, the high-frequency components are aliased three times. We will use the filters H_a, H_b, H_c and H_d shown in Figure 16. The filter H_a has a gain of '4' in the passband and the other three filters have a gain of '1' in their passbands.

Applying the horizontal highpass filter H_b to X we get

$$X_b(e^{j\omega_1}, e^{j\omega_2}) = X(e^{j\omega_1}, e^{j\omega_2})H_b(e^{j\omega_1}, e^{j\omega_2}). \quad (21)$$

We will then modulate these high-frequency components into the low-frequency region by shifting them horizontally,

$$X_b^m(e^{j\omega_1}, e^{j\omega_2}) = X(e^{j(\omega_1-\pi)}, e^{j\omega_2})H_b(e^{j(\omega_1-\pi)}, e^{j\omega_2}). \quad (22)$$

Similarly, applying H_c and H_d filters and modulating them into the low-frequency region we get the following equations:

$$X_c(e^{j\omega_1}, e^{j\omega_2}) = X(e^{j\omega_1}, e^{j\omega_2})H_c(e^{j\omega_1}, e^{j\omega_2}) \quad (23)$$

$$X_c^m(e^{j\omega_1}, e^{j\omega_2}) = X(e^{j\omega_1}, e^{j(\omega_2-\pi)})H_c(e^{j\omega_1}, e^{j(\omega_2-\pi)}) \quad (24)$$

and,

$$X_d(e^{j\omega_1}, e^{j\omega_2}) = X(e^{j\omega_1}, e^{j\omega_2})H_d(e^{j\omega_1}, e^{j\omega_2}) \quad (25)$$

$$X_d^m(e^{j\omega_1}, e^{j\omega_2}) = X(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})H_d(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \quad (26)$$

We get the interpolated red image by applying the lowpass filter H_a ,

$$\begin{aligned} R_i(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} [R(e^{j\omega_1}, e^{j\omega_2}) - R(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad + R(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - R(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})] H_a(e^{j\omega_1}, e^{j\omega_2}). \end{aligned} \quad (27)$$

Each of the three aliased terms present in this interpolated red image can be canceled by the modulated terms generated by filters H_b , H_c and H_d . Looking at (27) suggests we need to add (22) and (26) to cancel the aliasing generated by the terms $R(e^{j(\omega_1-\pi)}, e^{j\omega_2})$ and $R(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})$, and subtract (24) to negate $R(e^{j\omega_1}, e^{j(\omega_2-\pi)})$. Also we will add the high-frequency components from X to the red image. Thus taking (27)+(22)-(24)+(26)+(21)+(23)+(25) we get,

$$\begin{aligned} \hat{R}(e^{j\omega_1}, e^{j\omega_2}) &= R_i(e^{j\omega_1}, e^{j\omega_2}) + X_b^m(e^{j\omega_1}, e^{j\omega_2}) - X_c^m(e^{j\omega_1}, e^{j\omega_2}) + X_d^m(e^{j\omega_1}, e^{j\omega_2}) \\ &\quad + X_b(e^{j\omega_1}, e^{j\omega_2}) + X_c(e^{j\omega_1}, e^{j\omega_2}) + X_d(e^{j\omega_1}, e^{j\omega_2}). \end{aligned} \quad (28)$$

Now, consider the sampling equation for blue image,

$$\begin{aligned} B_s(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} B(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{4} B(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad - \frac{1}{4} B(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - \frac{1}{4} B(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}). \end{aligned}$$

Interpolating the blue image with the lowpass filter H_a gives,

$$\begin{aligned} B_i(e^{j\omega_1}, e^{j\omega_2}) &= \frac{1}{4} [B(e^{j\omega_1}, e^{j\omega_2}) + B(e^{j(\omega_1-\pi)}, e^{j\omega_2}) \\ &\quad - B(e^{j\omega_1}, e^{j(\omega_2-\pi)}) - B(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})] H_a(e^{j\omega_1}, e^{j\omega_2}). \end{aligned} \quad (29)$$

Following a similar analysis as described for the red image we get the final equation for the blue image,

$$\begin{aligned} \hat{B}(e^{j\omega_1}, e^{j\omega_2}) &= B_i(e^{j\omega_1}, e^{j\omega_2}) - X_b^m(e^{j\omega_1}, e^{j\omega_2}) + X_c^m(e^{j\omega_1}, e^{j\omega_2}) + X_d^m(e^{j\omega_1}, e^{j\omega_2}) \\ &\quad + X_b(e^{j\omega_1}, e^{j\omega_2}) + X_c(e^{j\omega_1}, e^{j\omega_2}) + X_d(e^{j\omega_1}, e^{j\omega_2}). \end{aligned} \quad (30)$$

Implementation

We have to modulate the high-frequency components from the filters H_b , H_c and H_d into the low-frequency region to cancel aliasing. Here we explain the implementation for the red image; that for blue is very similar.

In the spatial domain the horizontal high-frequency components X_b separated by H_b and its modulated components are to be added as per (28). Multiplying $x_b[n_1, n_2]$ by $(-1)^{n_1}$ modulates it horizontally with a phase shift π giving,

$$[1 + (-1)^{n_1}]x_b[n_1, n_2].$$

Thus for columns where n_1 is odd this term will be zero and for even columns it will be twice the value of $x_b[n_1, n_2]$ as shown in Figure 19.

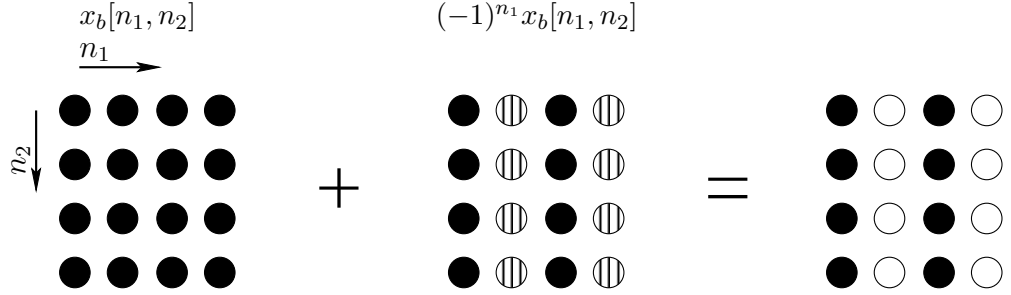


Figure 19: Modulation along the horizontal axis in spatial domain.

Similarly, for modulating the vertical high-frequency components into the low-frequency region with a phase shift π , we multiply the terms by $(-1)^{n_2}$,

$$[1 - (-1)^{n_2}]x_c[n_1, n_2].$$

This means that for even rows this term will be zero and for odd rows it will be $2x_c[n_1, n_2]$ as shown in Figure 20.

Modulating from filter H_d involves multiplying the terms with $(-1)^{n_1+n_2}$ as explained in (20). In the spatial domain, the components X_d and X_d^m are to be added as per (28) giving,

$$[1 + (-1)^{n_1+n_2}]x_d[n_1, n_2].$$

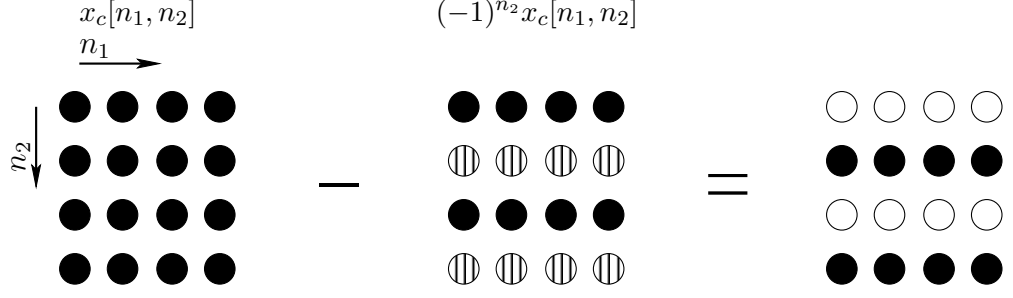


Figure 20: Modulation along the vertical axis in spatial domain.

When $n_1 + n_2$ is odd this term becomes zero; when $n_1 + n_2$ is even it becomes $2x_d[n_1, n_2]$. This is illustrated in Figure 21.

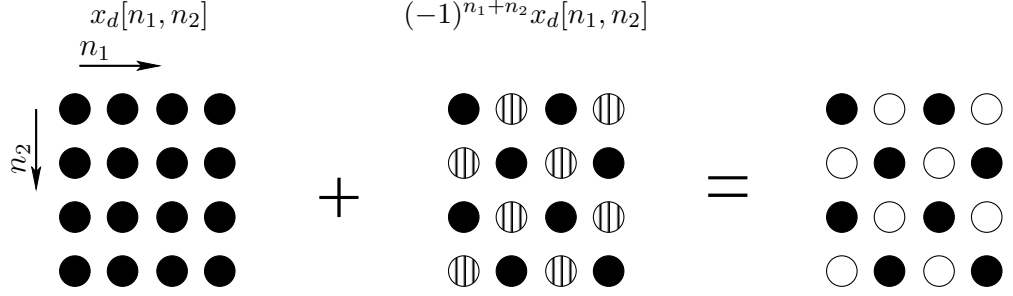


Figure 21: Modulation along both horizontal and vertical Axes.

Thus, the final equation for interpolation of red becomes,

$$\begin{aligned} \hat{r}[n_1, n_2] = & r_i[n_1, n_2] + x_b[n_1, n_2] + (-1)^{n_1} x_b[n_1, n_2] + x_c[n_1, n_2] \\ & - (-1)^{n_2} x_c[n_1, n_2] + x_d[n_1, n_2] + (-1)^{n_1+n_2} x_d[n_1, n_2]. \end{aligned} \quad (31)$$

Similarly, the final equation for the interpolation of blue is,

$$\begin{aligned} \hat{b}[n_1, n_2] = & b_i[n_1, n_2] + x_b[n_1, n_2] - (-1)^{n_1} x_b[n_1, n_2] + x_c[n_1, n_2] \\ & + (-1)^{n_2} x_c[n_1, n_2] + x_d[n_1, n_2] + (-1)^{n_1+n_2} x_d[n_1, n_2]. \end{aligned} \quad (32)$$

3.5 Filter Design

3.5.1 Filters for Green Image

We would like to design two filters, H_g and H_{hpg} as shown in Figure 17. As explained in Chapter 1, the effective filtering kernel used for bilinear interpolation in the green image is,

$$H_g = \begin{bmatrix} 0 & 1/4 & 0 \\ 1/4 & 1 & 1/4 \\ 0 & 1/4 & 0 \end{bmatrix}.$$

And it has a gain of 2 in the passband as shown in Figure 22(a). To design a filter with stop band in the diamond shaped region and a gain 1 in passband, we take, $H_{hpg}[n_1, n_2] = \delta[n_1, n_2] - \frac{1}{2}H_g[n_1, n_2]$ giving,

$$H_{hpg} = \begin{bmatrix} 0 & -1/8 & 0 \\ -1/8 & 1/2 & -1/8 \\ 0 & -1/8 & 0 \end{bmatrix}.$$

The filter response for H_{hpg} is given in Figure 22(b).

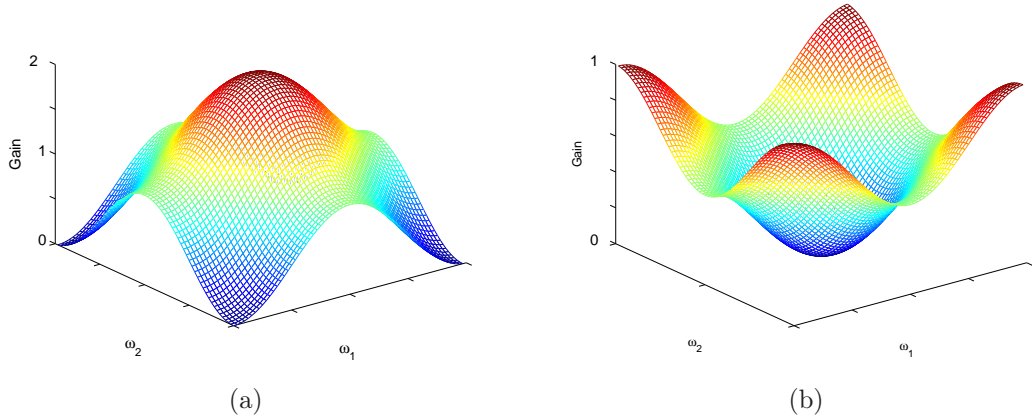


Figure 22: Filters used for Green image.

The filters can easily be extended beyond a 3x3 filter; once H_g is defined we can obtain H_{hpg} using $H_{hpg}[n_1, n_2] = \delta[n_1, n_2] - \frac{1}{2}H_g[n_1, n_2]$.

3.5.2 Filters for Red/Blue Images

As shown in Figure 16, we need four filters, H_a , H_b , H_c and H_d . The filter H_a has a passband gain of 4 and the other filters have a gain of 1. Again, let us consider the filter kernel used in bilinear interpolation in Chapter 1,

$$H_a = \begin{bmatrix} 1/4 & 1/2 & 1/4 \\ 1/2 & 1 & 1/2 \\ 1/4 & 1/2 & 1/4 \end{bmatrix}.$$

It has a gain of 4 in the passband as shown in Figure 23(a). Designing filters H_b and H_c involves multiplying H_a with $1/4(-1)^{n_1}$ and $1/4(-1)^{n_2}$ respectively. Multiplying with a factor of $1/4$ reduces the gain to 1 and multiplying by $(-1)^{n_1}$ and $(-1)^{n_2}$ shift the filter responses horizontally and vertically.

$$H_b = \frac{1}{4} \begin{bmatrix} 1/4 & -1/2 & 1/4 \\ 1/2 & -1 & 1/2 \\ 1/4 & -1/2 & 1/4 \end{bmatrix} = \begin{bmatrix} 1/16 & -1/8 & 1/16 \\ 1/8 & -1/4 & 1/8 \\ 1/16 & -1/8 & 1/16 \end{bmatrix}$$

and,

$$H_c = \begin{bmatrix} 1/16 & 1/8 & 1/16 \\ -1/8 & -1/4 & -1/8 \\ 1/16 & 1/8 & 1/16 \end{bmatrix}.$$

The frequency responses of filters corresponding to kernels H_b and H_c are shown in Figure 23(b) and Figure 23(c), respectively.

We get H_d by multiplying H_a with $(-1)^{n_1+n_2}$,

$$H_d = \begin{bmatrix} 1/16 & -1/8 & 1/16 \\ -1/8 & 1/4 & -1/8 \\ 1/16 & -1/8 & 1/16 \end{bmatrix}.$$

The frequency response of H_d is shown in Figure 23(d).

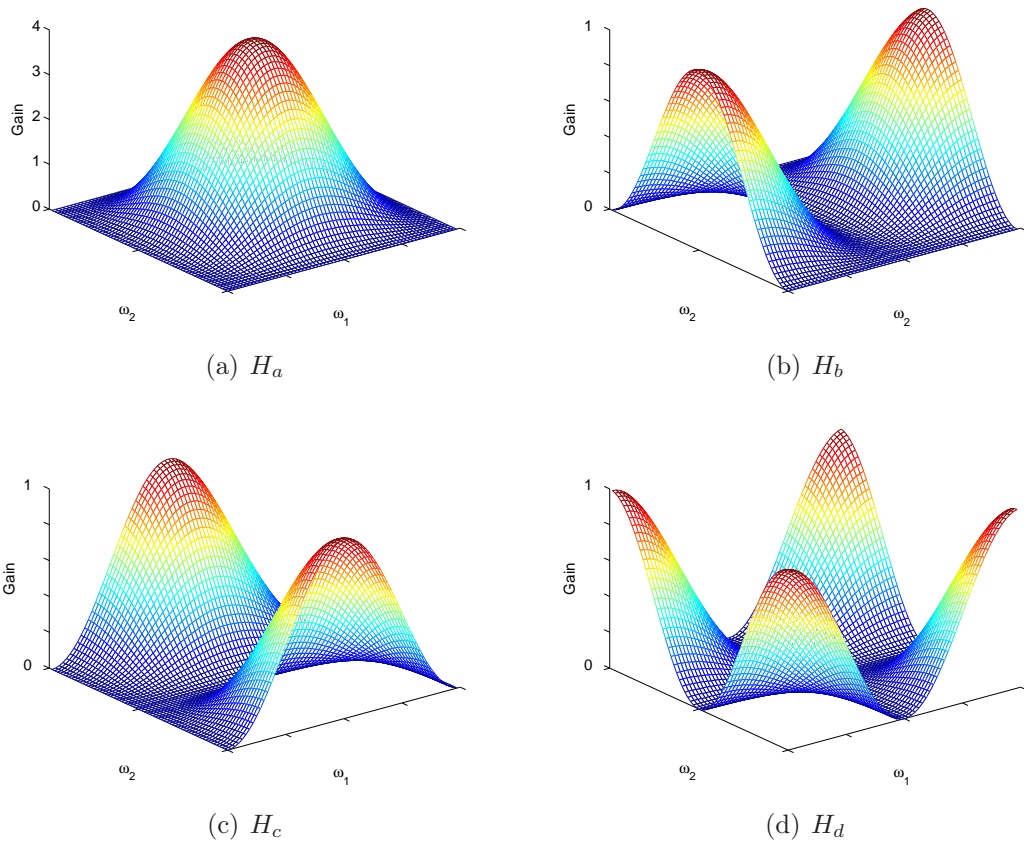


Figure 23: Filters used for red/blue images.

Following the same rules we can design larger filters once the lowpass filter H_a is given. If H_a is a separable half-band filter, then each of the other filters will also be a half-band filter. Using half-band filters significantly reduces the complexity in designing and implementing these filters. While increasing the filter length may give a better approximation of the ideal filter response, it adds to the complexity of the algorithm and may also produce ringing artifacts around the edges.

3.6 Results

When using a Bayer array, none of the three channels is fully sampled. So we do not have a fully sampled array. For the purpose of our experiments, we use images with fully sampled red, green and blue channels from which we generate the Bayer array.

So, in this section we will consider the fully sampled green array as our X and cancel the aliasing present in the red and blue channels. We will compare the results with two basic approaches for CFA interpolation: the bilinear interpolation and the AH method [2]. Figure 24(a) shows the result of using the green image as X , and the filters designed were 9x9 filters. Other sizes were also tested but 9x9 filters gave the best results. Figure 24(b) and Figure 24(c) show the results of the bilinear and the AH methods.

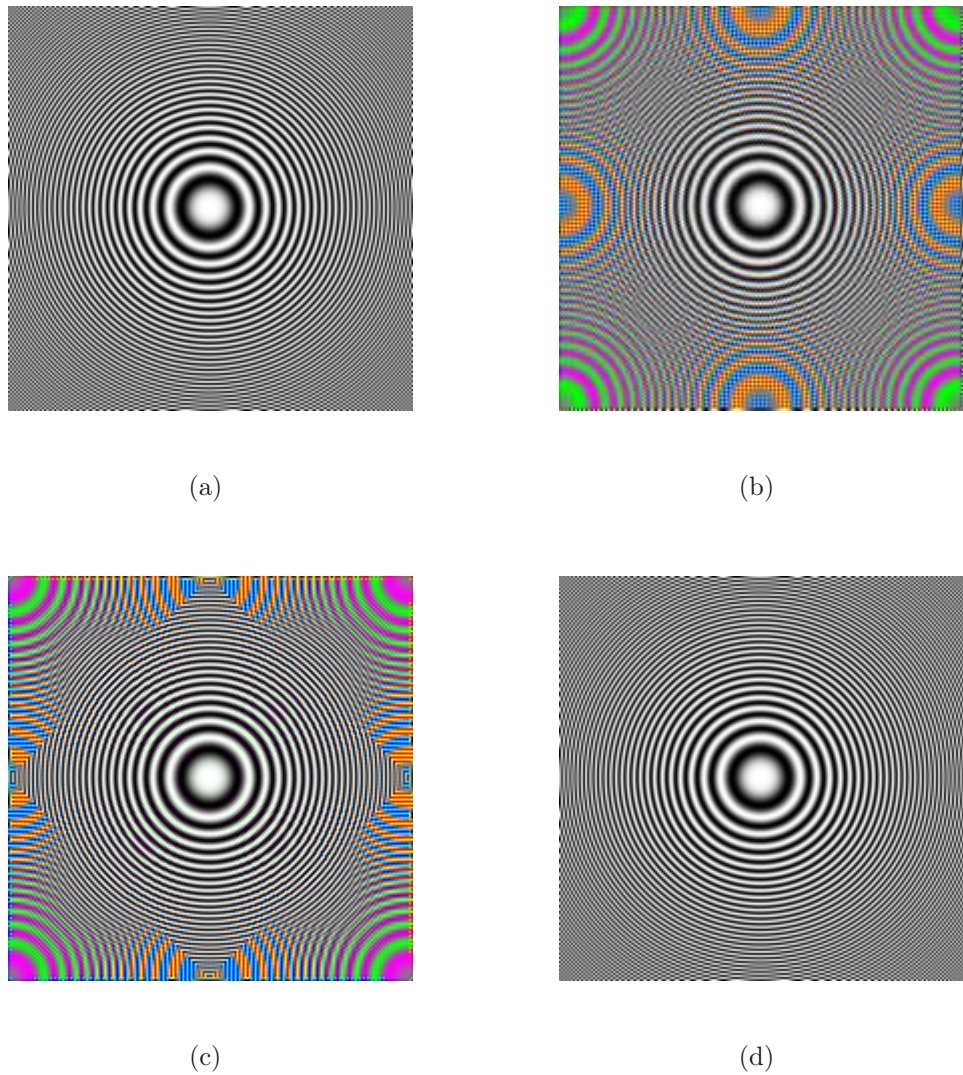


Figure 24: Chirp image: (a) Original (b) Bilinear (c) AH (d) Alias Cancellation.

We notice that we are able to recover the chirp image without any aliasing if we

use the green image to cancel the aliasing. This is mainly because the green image in the chirp image has identical high frequency regions as the red and blue images. That will not be the case for natural images, although we make the assumption of high correlation in the three channels [4, 6]. The Mean Squared Error (MSE) in Table 1 also suggests complete cancellation of the aliasing in the chirp image.

Figure 25 makes the same comparison using a test image. We notice here that the recovered image has no aliasing artifacts although there is a small MSE. This shows that the high-frequency regions of the three channels are not identical but are very similar as suggested in [4].

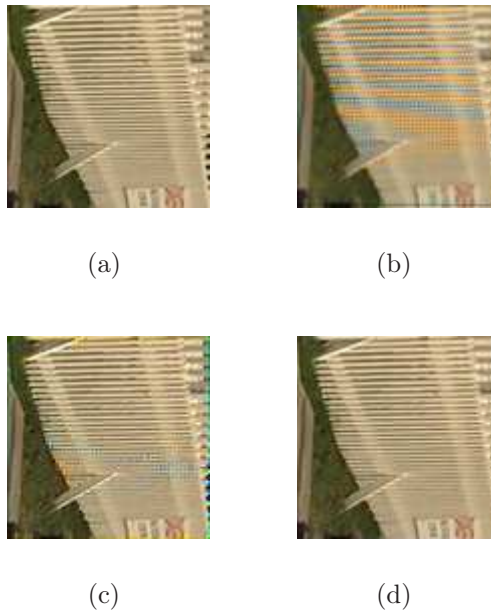


Figure 25: Fence image: (a) Original (b) Bilinear (c) AH (d) Alias Cancellation.

Table 1: MSE Results: (a) Bilinear (b) AH (c) Alias Cancellation.

	(a)			(b)			(c)		
	R	G	B	R	G	B	R	G	B
Fence	93.33	94.93	93.55	14.72	11.05	14.58	1.83	0.00	2.06
Chirp	117.78	117.51	117.78	51.4	39.38	51.4	0.00	0.00	0.00

Thus, using the original green image to cancel aliasing, we remove the aliasing almost completely. This also suggests that the MSE attained here is an upper bound

on what we can obtain using the alias cancellation method. Unfortunately, we will not have a fully sampled green image when we use a Bayer array. Since green is sampled at a higher sampling rate than the red and blue images, an interpolated green image is less likely to suffer aliasing artifacts. In the next chapter we will discuss methods to interpolate the green image with minimal aliasing and then use the interpolated green image as our X to cancel aliasing in the red and blue channels.

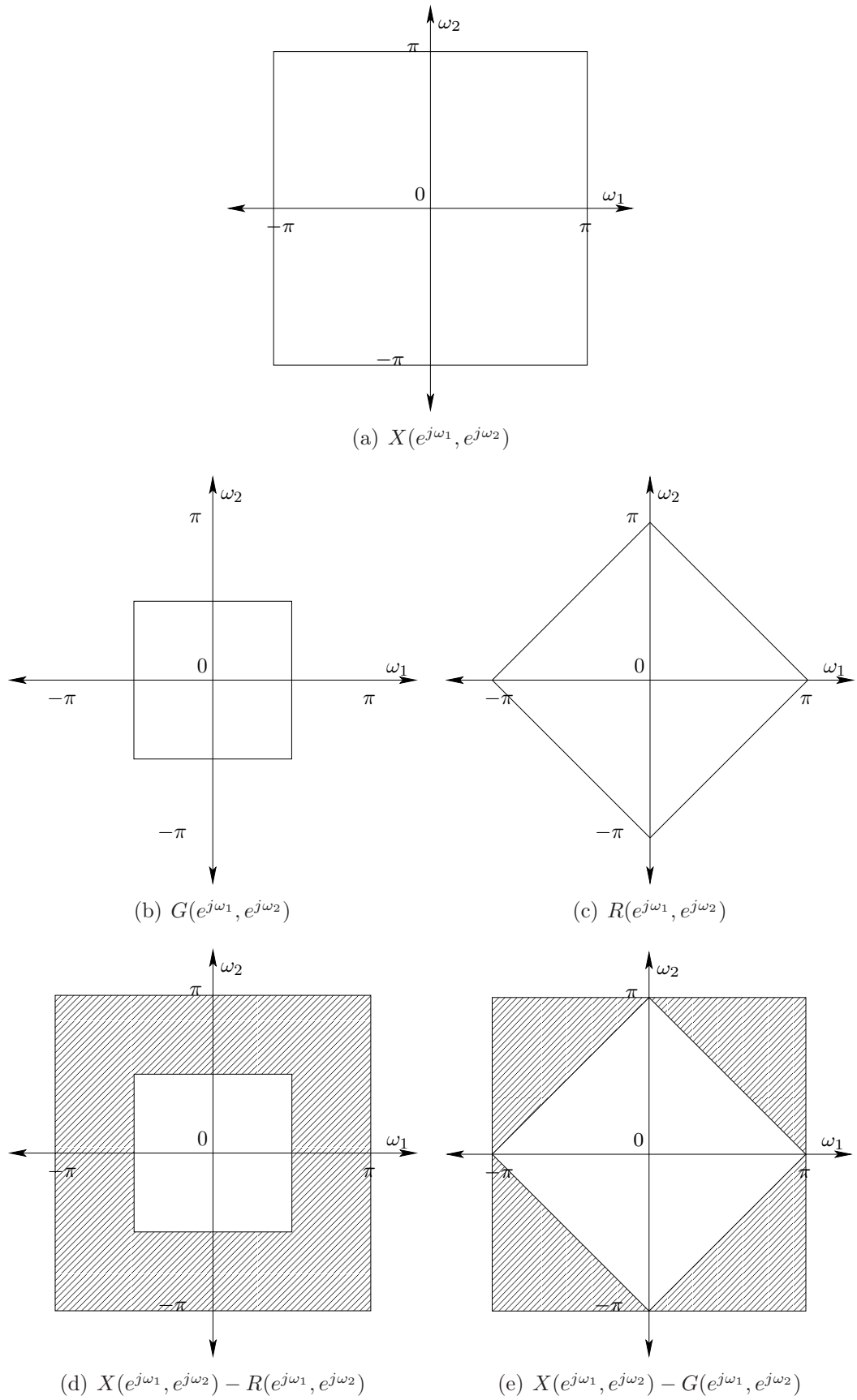


Figure 26: Nyquist regions.

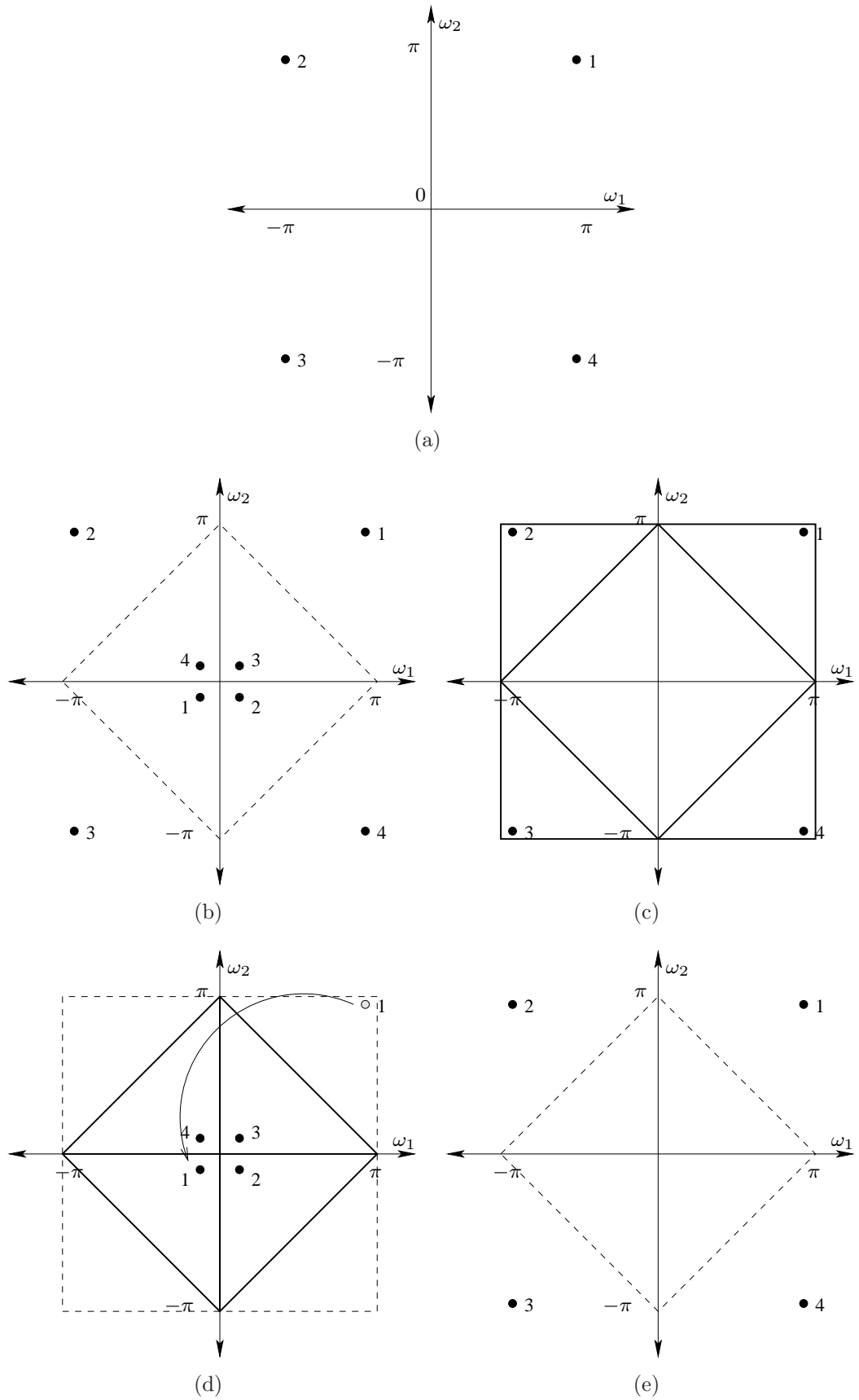


Figure 27: Alias cancellation in green channel.

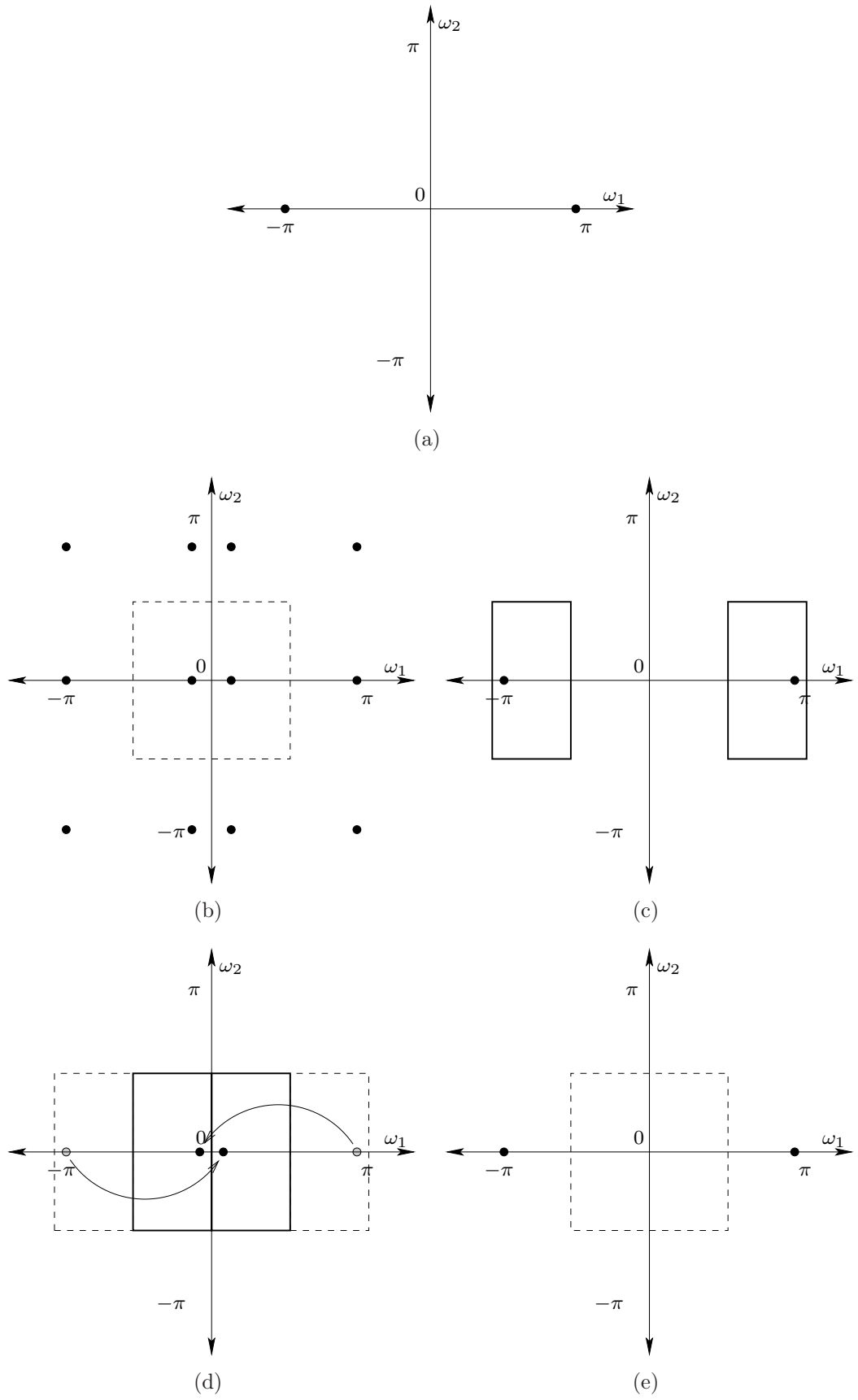


Figure 28: Alias cancellation in red channel.

Chapter IV

GREEN IMAGE INTERPOLATION

4.1 *Introduction*

As discussed earlier, to cancel aliasing in the red and blue channels using the green image we need an alias-free green image. In this chapter we try to minimize aliasing artifacts in the interpolated green image. The green in a Bayer array is sampled in a quincunx grid and the Nyquist region in the frequency-domain is shown in Figure 29. As discussed in Chapter 1, problems in demosaicking algorithms are mainly caused by aliasing artifacts. Various algorithms [6, 7, 2, 8] incorporate edge directionality to alleviate these artifacts. These algorithms rely on the assumption that there is a very high correlation between the red, green and blue channels in the high frequency region as suggested in [4]. Also [9, 10, 8, 11] make use of the fact that although the green image changes rapidly in the high frequency regions, the $R - G$ and $B - G$ images vary relatively slowly. And [12, 3, 13, 4] use the high correlation assumption to generate a correction term to cancel aliasing in the green channel using information from the red/blue channel and vice versa.

The sampling equation derived for green in Chapter 2 has the term $G(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)})$ which suggests the presence of aliasing.

$$G_s(e^{j\omega_1}, e^{j\omega_2}) = \frac{1}{2}G(e^{j\omega_1}, e^{j\omega_2}) + \frac{1}{2}G(e^{j(\omega_1-\pi)}, e^{j(\omega_2-\pi)}).$$

Glotzbach et al. [3] use the green image to cancel aliasing in the red/blue, but assume that the green image is band-limited to the diamond-shaped Nyquist region shown in Figure 29, thus, neglecting the aliasing present in green image.

We will combine the adaptive homogeneity-directed method as the edge-directed

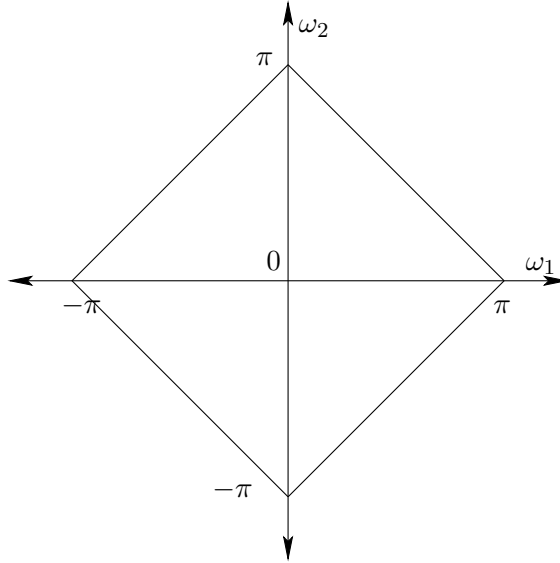


Figure 29: Diamond-shaped quincunx region for green image.

interpolation algorithm [8, 11] and POCS [4, 6] to generate alias correction for the green image. This produces a significantly improved green image in comparison to some of the other reference algorithms discussed. We will also analyze the effects of the interpolation in the frequency-domain of the green image. We also describe how it improves the overall output image to assume that the green image is alias free in the entire frequency-domain rather than assuming the green image to be band-limited to the diamond-shaped Nyquist region.

4.2 *Adaptive Homogeneity-Directed Interpolation*

In this method the green image is first interpolated along the horizontal and then the vertical direction; the outputs are called f_H and f_V , respectively. Then a homogeneity map, developed using a metric neighborhood model is used to select the direction with the fewest artifacts to get the final green image.

4.2.1 Generating f_H and f_V

In this section we discuss the horizontal interpolation of the green image (f_H). f_V can be obtained following the same steps. We need to estimate the missing green values using the pixels where the green values are known. Looking at the Bayer pattern in Figure 30 the first row is a green-red row where the missing green value g_r needs to be estimated from the neighboring green values. This can be done by taking $g_r = \frac{g_1 + g_2}{2}$ as shown in Figure 31. Similarly in a blue-green row it would be $g_b = \frac{g_1 + g_2}{2}$.

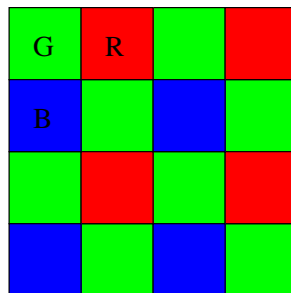


Figure 30: Bayer Array.



Figure 31: Red-Green and Green-Blue rows indicating missing green values.

This can be thought of as low-pass filtering in the 1-D domain. Thus, we have

$$G[n_1, n_2] = G[n_1, n_2] * h(n),$$

where $h(n)$ is a 1-D filter with the frequency response shown in Figure 32.

The chirp image is also used to illustrate the frequency response of the horizontal and vertical filtering in Figure 33. Notice in the f_H and f_V images corresponding to the chirp that the regions along the horizontal and vertical high frequency, respectively, are interpolated without aliasing. The vertical high frequency region in f_H shows aliasing. Similarly in the f_H of the lighthouse image shown in Figure 34 we can

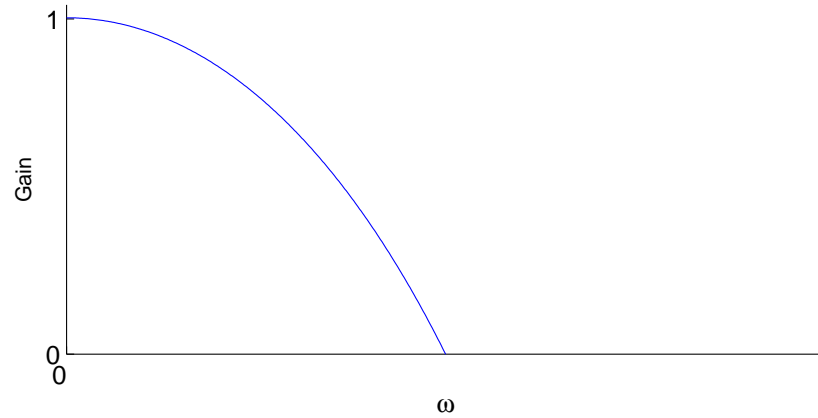


Figure 32: 1-D lowpass filter response.

see aliasing artifacts along the picket fence. This is caused by interpolation across the edge boundary. Similarly f_V shows aliasing on the horizontal stripes along the wall of the house.

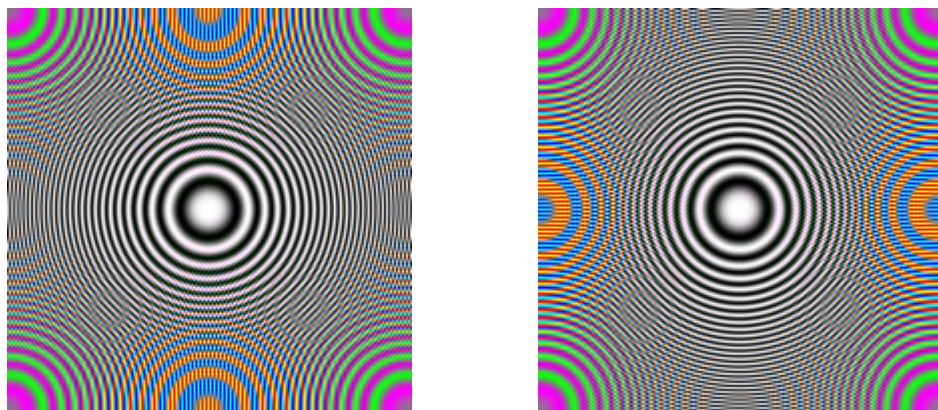


Figure 33: Horizontal and Vertical filtered Chirp image.



Figure 34: Horizontal and Vertical filtered Lighthouse image.

4.2.2 Neighborhood Metric Model and Homogeneity Maps

In this section we discuss how the correct interpolation direction is selected once the f_H and f_V have been generated. We use the local homogeneity as a measure to choose between the horizontal and vertical directions for interpolation. The local homogeneity map is generated using a color homogeneity metric [8]. This metric is generated with the constraint that the chrominance and luminance are similar within a small neighborhood.

The Metric Neighborhood Model

The metric neighborhood model is used to identify groups of pixels with similar chrominance and luminance values. We will define the X as the 2-D pixel position and Y as the set of RGB values in the image. The color image can be defined as a function $f : X \rightarrow Y$. Now, we define a distance function for X as $d_X : X \times X \rightarrow R$.

$$d_X([n_1, n_2], [n_3, n_4]) = \sqrt{(n_3 - n_1)^2 + (n_4 - n_2)^2},$$

By neighborhood we refer only to the immediate neighbors of the center pixel *i.e.* pixels which are within a distance of 1 from the center pixel. In general the neighborhood can be defined as

$$B(n_1, n_2, \delta) = \{[n_p, n_q] \in X \mid d_X([n_1, n_2], [n_p, n_q]) \leq \delta\}.$$

Since the metric is based on the luminance and chrominance we will first convert the RGB values to CIELAB space. It also helps that the end user of the output is human and the CIELAB space is modeled to suit the human visual system. The color space conversion is denoted by $f : Y \rightarrow \hat{Y}$ where \hat{Y} describes the color space CIELAB. The distance functions in the luminance (L) and color (a-b) planes are given by

$$\begin{aligned} d_L([n_1, n_2], [n_3, n_4]) &= |[n_1, n_2]_L - [n_3, n_4]_L| \\ d_C([n_1, n_2], [n_3, n_4]) &= \sqrt{([n_1, n_2]_a - [n_3, n_4]_a)^2 + ([n_1, n_2]_b - [n_3, n_4]_b)^2}. \end{aligned} \quad (33)$$

where $\hat{y}(n_1, n_2) = [[n_1, n_2]_L, [n_1, n_2]_a, [n_1, n_2]_b]$.

Now we choose ϵ_L and ϵ_C as thresholds to choose between the variation along the horizontal and vertical directions in the luminance and color planes. We define the luminance level neighborhood (L_f) and the color neighborhood (C_f) as:

$$\begin{aligned} L_f(n_1, n_2, \epsilon_L) &= \{[n_p, n_q] \in X \mid d_L([n_1, n_2], [n_p, n_q]) \leq \epsilon_L\} \\ C_f(n_1, n_2, \epsilon_C) &= \{[n_p, n_q] \in X \mid d_C([n_1, n_2], [n_p, n_q]) \leq \epsilon_C\}. \end{aligned} \quad (34)$$

Once we have defined B, L_f, C_f , we can now define the homogeneity as

$$H_f(n_1, n_2) = |L_f(n_1, n_2, \epsilon_L) \cap C_f(n_1, n_2, \epsilon_C) \cap B(n_1, n_2, \delta)| / |B(n_1, n_2, \delta)|. \quad (35)$$

This equation indicates that $H_f(n_1, n_2)$ is the measure of similarity around a pixel. Pixels with artifacts will have a smaller value of homogeneity.

In [8] it is shown that instead of taking $\epsilon_L = \epsilon_C = 1$, extracting this parameter adaptively from the image improves the result. We would like to assign ϵ_L, ϵ_C with values that reflect the variation along the object in which the current pixel location, $[n_1, n_2] \in X$ lies.

Consider that our pixel of interest $[n_1, n_2]$, lies on an object boundary oriented in the horizontal direction. So, we would want $f(n_1, n_2) = f_H(n_1, n_2)$ and pixels on either side of $[n_1, n_2]$ in the horizontal direction must lie on the same object implying higher homogeneity values. So, we choose ϵ_L and ϵ_C to be the larger of the distance between $f_H([n_1, n_2])$ and each of the adjacent pixel to the left and right. Thus we have

$$\begin{aligned} \epsilon_{LH}(n_1, n_2) &= \max\{d_L(f_H(n_1, n_2), f_H(n_1 - 1, n_2)), d_L(f_H(n_1, n_2), f_H(n_1 + 1, n_2))\} \\ \epsilon_{CH}(n_1, n_2) &= \max\{d_C(f_H(n_1, n_2), f_H(n_1 - 1, n_2)), d_C(f_H(n_1, n_2), f_H(n_1 + 1, n_2))\}. \end{aligned} \quad (36)$$

Similarly, for pixels along a vertical edge we would want $f(n_1, n_2) = f_V(n_1, n_2)$

and ϵ_L , ϵ_C can be written as:

$$\begin{aligned}\epsilon_{LV}(n_1, n_2) &= \max\{d_L(f_V(n_1, n_2), f_V(n_1, n_2 - 1)), d_L(f_V(n_1, n_2), f_V(n_1, n_2 + 1))\} \\ \epsilon_{CV}(n_1, n_2) &= \max\{d_C(f_V(n_1, n_2), f_V(n_1, n_2 - 1)), d_C(f_V(n_1, n_2), f_V(n_1, n_2 + 1))\}.\end{aligned}\tag{37}$$

Since we want ϵ_L and ϵ_C to reflect variation along the same object we choose,

$$\begin{aligned}\epsilon_L(n_1, n_2) &= \min\{\epsilon_{LH}(n_1, n_2), \epsilon_{LV}(n_1, n_2)\} \\ \epsilon_C(n_1, n_2) &= \min\{\epsilon_{CH}(n_1, n_2), \epsilon_{CV}(n_1, n_2)\}.\end{aligned}\tag{38}$$

Using these values we can calculate the homogeneity maps H_{f_H} , H_{f_V} corresponding to f_H and f_V respectively.

Homogeneity Maps

As seen in Figure 34 in f_H and f_V we see artifacts along the fence when the wrong direction of interpolation is chosen. This can also be viewed with respect to homogeneity maps as shown in Figure 35. Here we see that the value of homogeneity is higher along the fence in H_{f_V} than H_{f_H} , indicating that higher homogeneity values correspond to the correct direction of interpolation. Based on this observation we would like to combine f_H and f_V and form a single image f by choosing the pixel that has higher homogeneity. Thus at a given pixel location $[n_1, n_2] \in X$, we have

$$f(n_1, n_2) = \begin{cases} f_V(n_1, n_2), & \text{if } H_{f_V}(n_1, n_2) > H_{f_H}(n_1, n_2), \\ f_H(n_1, n_2), & \text{if } H_{f_V}(n_1, n_2) \leq H_{f_H}(n_1, n_2). \end{cases}\tag{39}$$

We can also generalize this technique to include images interpolated along the diagonal directions as well. Also, frequent switching of interpolation direction may lead to discontinuities in the output image. Thus a simple averaging filter $A = [1, 1, 1; 1, 1, 1; 1, 1, 1]/9$ is used on the homogeneity map to reduce these discontinuities. Incorporating this averaging, (39) is re-written as

$$f(n_1, n_2) = \begin{cases} f_V(n_1, n_2), & \text{if } A * H_{f_V}(n_1, n_2) > H_{f_H}(n_1, n_2), \\ f_H(n_1, n_2), & \text{if } A * H_{f_V}(n_1, n_2) \leq H_{f_H}(n_1, n_2). \end{cases}\tag{40}$$

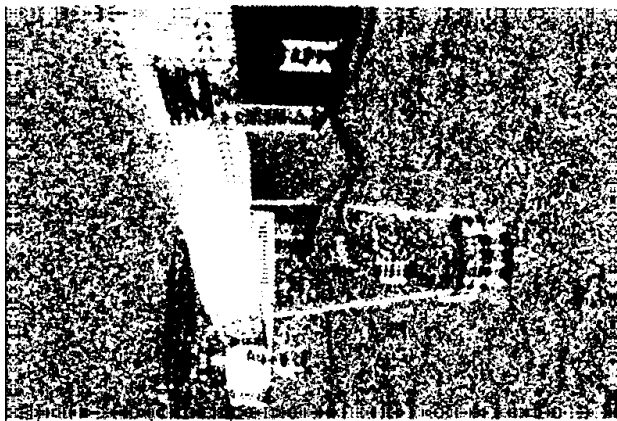
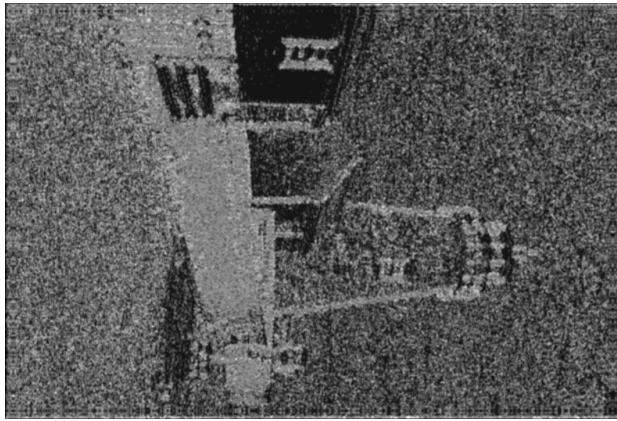
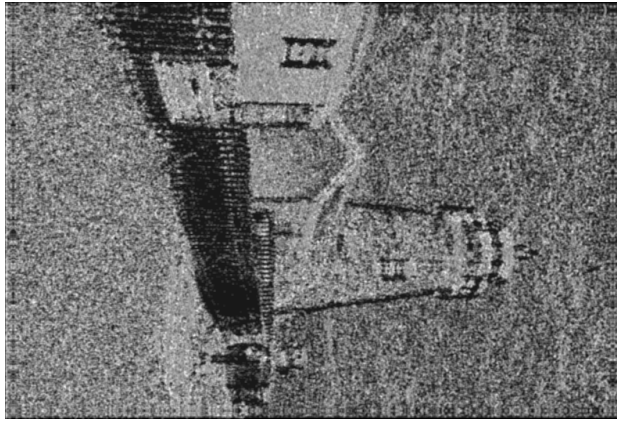
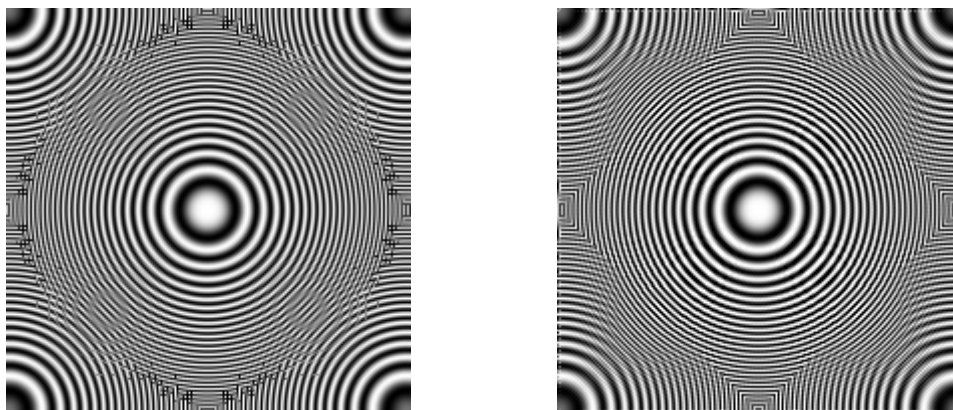


Figure 35: Homogeneity maps of f_H and f_V , Chosen direction for interpolation.

Figure 35 shows the homogeneity maps of f_H and f_V and also the map of direction chosen for interpolation, where *white* indicates that the vertical interpolation direction is chosen and *black* indicates horizontal interpolation. And Figure 36 shows the final green image of lighthouse after the correct direction for interpolation is chosen at each pixel.



Figure 36: Green image after appropriate directions have been selected.



(a) Green image after appropriate directions have been selected.

(b) AH

Figure 37: Result of green interpolation on chirp image.

Figure 37(a) shows the final green image of the chirp image after the correct

direction of interpolation has been chosen at each pixel. We compare it with AH [2] in Figure 37(b) and notice that we get a better reconstruction of some of the components along the horizontal and vertical high-frequency regions.

4.3 Projection Onto Convex Sets

The Projection Onto Convex Sets (POCS) algorithm for CFA interpolation uses filterbank analysis and synthesis pair. POCS as described by Gunturk et al. in [6] uses the Adams and Hamilton (edge directed interpolation) method to interpolate the green image and corrects this image using information from the red and blue images. Here we propose to use Adaptive-Homogeneity directed interpolation on the green image and use POCS as a post-processing step on the green image.

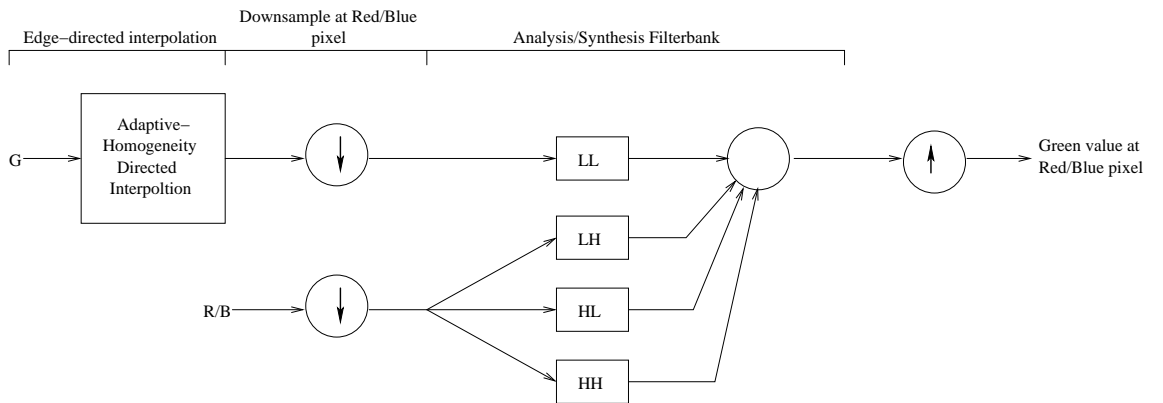


Figure 38: POCS block diagram.

Figure 38 shows the complete block diagram for POCS used for the green image. Each image is decomposed into four sub-bands. Two dimensional separable filters are generated using a lowpass 1-D filter ($h_0 = [1, 2, 1]/4$) and a highpass 1-D filter ($h_0 = [1, -2, 1]/4$) to decompose the images into four sub-bands LL, LH, HL, HH . The green image is downsampled to contain only the interpolated values of the green image at red/blue pixel locations. This is then passed through the filterbank along with the original red/blue pixel. The low-frequency sub-band (LL) of the green is combined

with the high-frequency(LH, HL, HH) information from the red/blue images. The high-frequency content of red/blue act as correction terms to the interpolated green value.

The simplified block diagram shown in Figure 39 combines the three higher sub-bands to form H_{eff} . Figure 40 shows the filter response for the LL filter used for the interpolated green samples and the H_{eff} filter used on the red and blue samples.

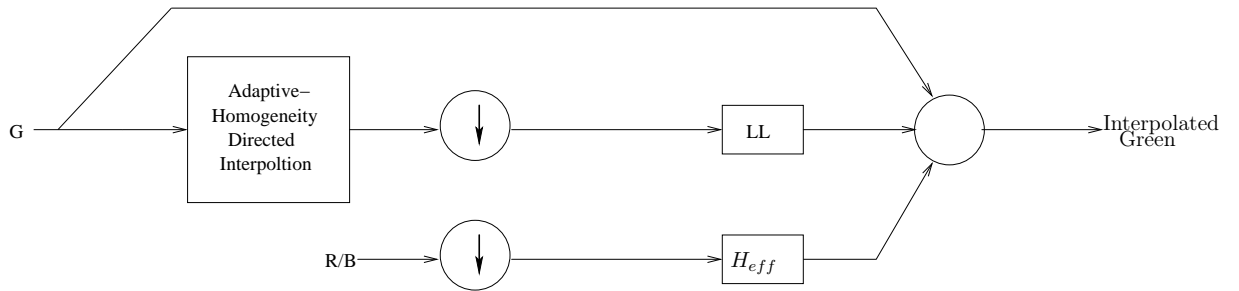


Figure 39: Simplified block diagram.

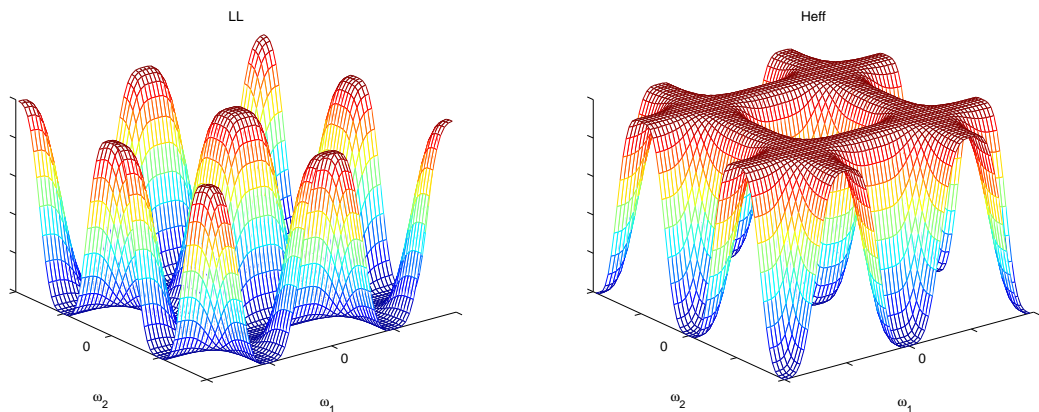


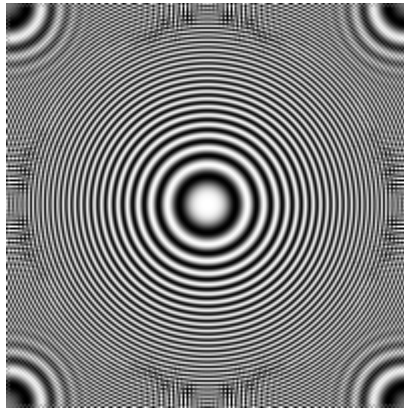
Figure 40: The LL and H_{eff} filters used in POCS.

Since edge-directed interpolation is used to generate the green pixels passed through the filterbank, we can consider the output generated by LL to contain horizontal or vertical high frequencies only. Although the mid frequencies are partially suppressed by the filter, the H_{eff} compensates for this by adding information from the red/blue

samples. Figure 41 shows the final green image of the lighthouse and chirp images using POCS.



(a)



(b)

Figure 41: Green image generated by POCS(Using Adaptive-Homogeneity directed edge-based interpolation).

4.4 *Results*

Using the chirp image we can compare the frequency response of the various algorithms. Figure 42 shows the green image when the chirp image is interpolated with various algorithms. We see that, except for the bilinear interpolated image, all of the

other algorithms reconstruct without any aliasing inside the diamond-shaped Nyquist region.

The aim of this chapter was to interpolate green with the least possible amount of aliasing. Thus, we would like to try and improve the interpolation of the high-frequency regions outside the Nyquist Region. The AH-POCS uses POCS as a post-processing step, with correction filters to add mid-frequency values from the red/blue images to green image interpolated using AH. This is shown in Figure 42(d). As explained earlier the Homogeneity directed interpolation reconstructs the horizontal and vertical high-frequency regions better than the AH method. Thus, adding mid-frequencies from red/blue to this image as in Figure 42(f), we are able to recover a better reconstructed image than Figure 42(d). Figure 42(e) shows the result from [8], which uses median filtering (non-linear filtering) to reduce aliasing.

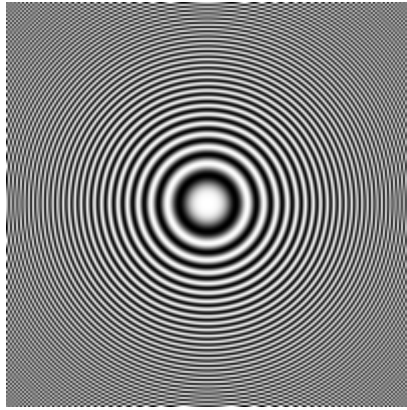
Table 2 shows the MSE in the green channel of the 24 images in the Kodak color image database using the reference algorithms AH-POCS[3], Adaptive-Homogeneity Directed method [8] and the POCS as explained in this section.

We notice that the proposed method shows an improvement in MSE compared to the other two methods. The amount of energy in the high-frequency is a very small portion of the total energy; hence the small difference in MSE may seem insignificant. However, a majority of the red/blue interpolation techniques depend on the green interpolated image. Hence, any improvement in the green interpolation will reflect in the red/blue images as well.

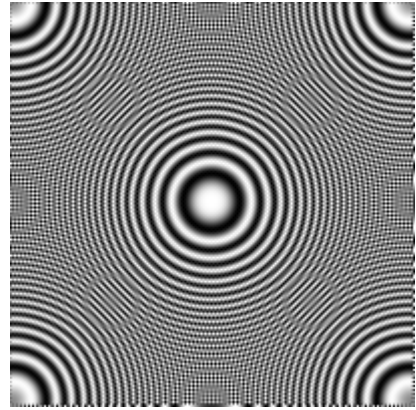
As shown in the results for the chirp image, the proposed method reconstructs the high-frequency components outside the Nyquist shaped diamond region. Hence, using this green image to cancel aliasing in the red and blue images not only within the diamond-shaped Nyquist region as discussed in [3], but in the entire frequency domain will interpolate higher frequency components better.

Table 2: MSE for green image in the 24 color images from Kodak color image database.(a) AH-POCS (b) Adaptive-Homogeneity (c) Proposed Method.

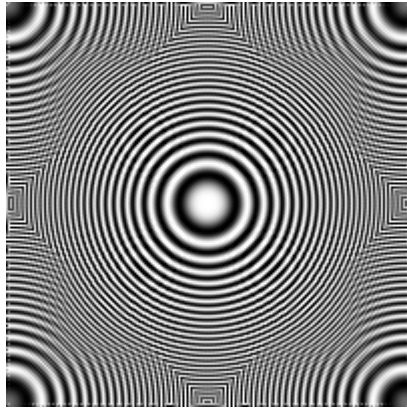
Image	AH-POCS	Adaptive-Homogeneity	Proposed Method
1	4.39	7.61	3.51
2	1.71	2.42	1.86
3	1.29	1.46	1.15
4	1.88	2.60	1.85
5	3.60	5.61	3.54
6	2.79	3.77	1.88
7	1.36	1.83	1.37
8	4.49	7.08	4.38
9	1.57	1.77	1.42
10	1.30	1.73	1.23
11	2.28	3.65	2.12
12	1.22	1.34	1.00
13	5.57	13.19	6.20
14	3.23	4.85	3.44
15	1.75	2.72	1.75
16	1.35	1.89	0.91
17	1.26	2.25	1.35
18	3.61	6.10	3.61
19	2.20	3.64	1.94
20	1.50	2.47	1.41
21	2.54	4.85	2.40
22	3.01	4.46	2.98
23	1.34	1.41	1.28
24	3.57	5.15	3.26



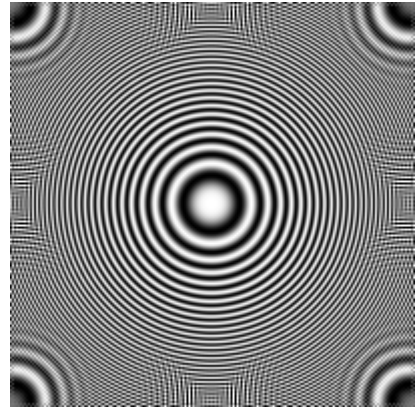
(a) Original



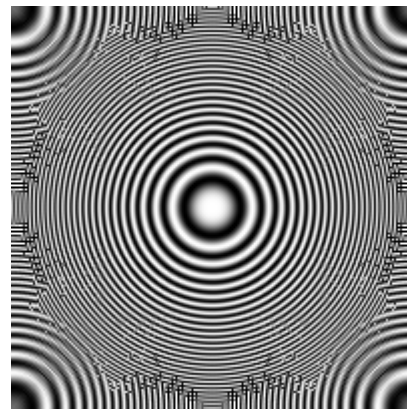
(b) Bilinear MSE : 118.05



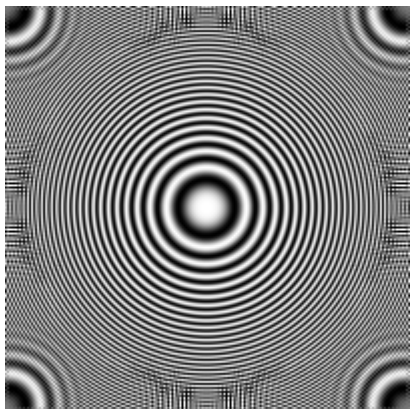
(c) AH MSE : 41.68



(d) AH-POCS[3] MSE : 15.89



(e) Adaptive Homogeneity[8] MSE : 34.98



(f) Proposed Method MSE : 13.99

Figure 42: Result of green interpolation on chirp image.

Chapter V

RED/BLUE IMAGE INTERPOLATION

5.1 *Introduction*

We will use the alias cancellation method discussed in Chapter 3 to interpolate the red/blue image. The aliasing in the red and blue images are cancelled using the high-frequency information from the interpolated green image. We used X which was a fully sampled array to cancel aliasing in Chapter 3. Here we will assume the interpolated green image as alias-free and use it to cancel aliasing in red. Once we have X , following the procedure discussed in Section 3.4 we can generate the red and blue images.

5.2 *Results*

Figure 45 shows a comparison of the red interpolation in various algorithms using the chirp image. Figure 45(b) with bilinear interpolation shows that the regions outside the Nyquist region for red suffer from aliasing. In Figures 45(c) and (e) we see that the images are identical to those generated by the respective algorithms for green interpolation. However, in Figure 45(d) we notice that although the mid-frequency regions (regions at an angle of 45° from the origin) are reconstructed without aliasing in the green image, the red image shows aliasing in those regions. This is because AH-POCS assumes the green image to be band-limited to the diamond shaped Nyquist region, and cancels aliasing present only within that region. Thus, although green cancels aliasing present in regions outside its Nyquist region, the same is not translated into the red image. In Figure 45(f), since aliasing in the entire frequency domain is cancelled, we see that the mid-frequency regions are also reconstructed identical

to that in green image. The interpolation of blue image will also display similar characteristics as it is sampled in a rectangular grid like red.

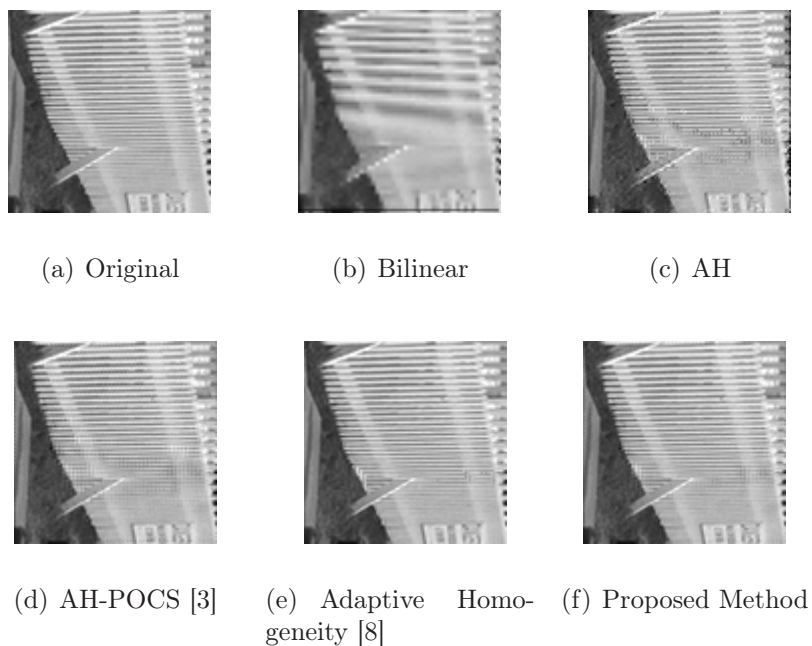


Figure 43: Result of red interpolation on Fence image.



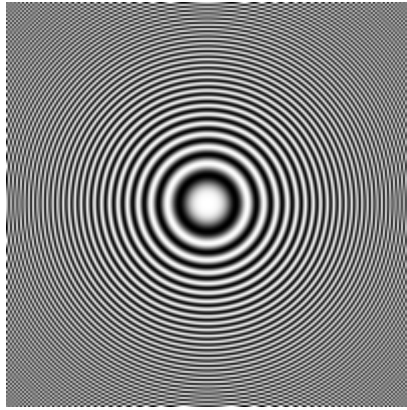
Figure 44: Red Image: Green image generated in Chapter (4) was used to cancel aliasing.

Figure 43 shows a comparison of the red interpolated fence image using various algorithms. Figure 44 shows the interpolated red image for the lighthouse image.

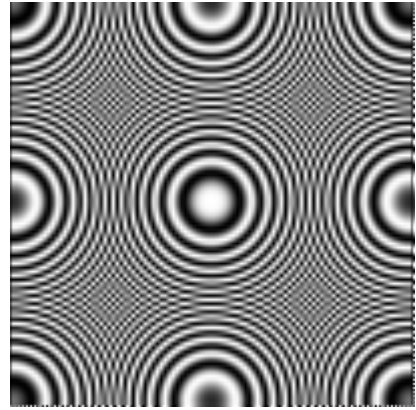
And Table 3 shows the MSE in the red and blue channels of the 24 images in the Kodak color image database using the reference algorithms AH-POCS [3], Adaptive-Homogeneity Directed method [8] and alias cancellation. We see that there is a significant improvement in MSE for all the images using alias cancellation as described in this chapter to using AH-POCS. This is mainly because of two improvements over AH-POCS, firstly because a better edge-directed interpolation is used to interpolate green image and then we also cancel aliasing outside the diamond shaped Nyquist region. Thus, the quality of red and blue image is highly dependant on the green interpolated image quality.

Table 3: MSE for red and blue image in the 24 color images from Kodak color image database.(a) AH-POCS (b) Adaptive-Homogeneity (c) Proposed Method.

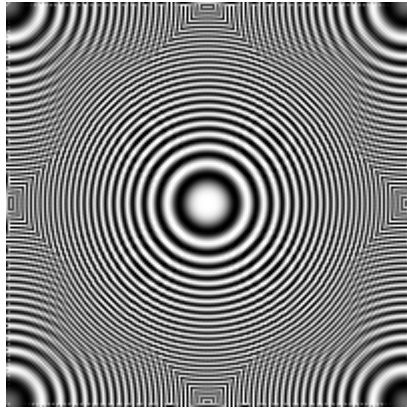
Image	AH-POCS		Adaptive-Homogeneity		Proposed Method	
	R	B	R	B	R	B
1	10.36	9.82	9.82	10.39	7.16	7.04
2	6.73	4.57	7.29	3.13	6.54	3.14
3	2.50	3.29	2.31	3.36	1.97	2.85
4	5.03	3.67	7.26	3.14	5.46	2.58
5	9.47	10.36	8.74	11.01	6.47	8.36
6	6.10	7.95	4.55	6.77	3.28	5.43
7	3.07	3.88	2.77	4.60	2.39	3.40
8	12.72	13.26	11.71	12.38	9.15	10.21
9	3.30	3.60	2.81	3.47	2.38	2.90
10	2.95	3.39	2.71	3.54	2.23	2.87
11	5.98	6.36	5.52	5.84	4.36	4.62
12	2.93	2.80	2.76	2.62	2.25	2.09
13	16.54	18.92	14.98	22.45	10.09	16.17
14	8.53	9.39	8.12	8.78	7.11	7.83
15	7.10	4.15	8.62	4.48	6.95	3.29
16	3.25	4.16	2.28	3.48	1.55	2.58
17	3.25	3.63	2.80	4.05	2.16	3.00
18	7.92	9.37	8.12	11.73	5.71	8.68
19	5.26	6.18	4.43	6.10	3.31	4.70
20	3.03	5.33	2.73	5.84	1.98	4.63
21	6.20	8.03	5.53	8.66	3.85	6.49
22	5.68	7.03	5.93	8.99	4.51	6.61
23	3.01	2.66	2.95	2.65	2.71	2.29
24	6.61	9.90	5.73	13.01	4.61	9.81



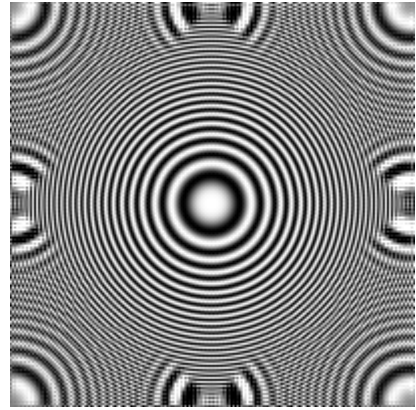
(a) Original



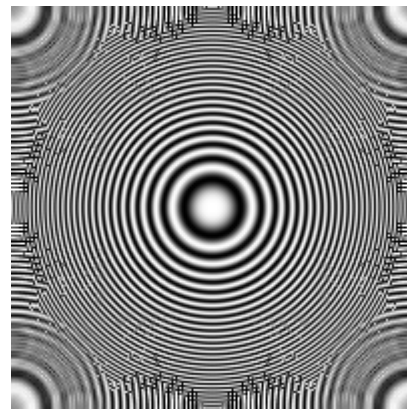
(b) Bilinear MSE : 117.98



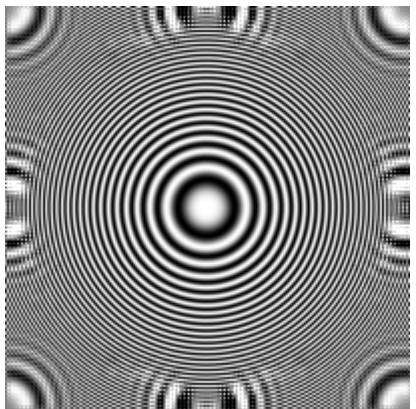
(c) AH MSE : 55.32



(d) AH-POCS[3] MSE : 45.32



(e) Adaptive Homogeneity[8] MSE : 38.65



(f) Proposed Method MSE : 21.91

Figure 45: Result of red interpolation on chirp image.

Chapter VI

EXPERIMENTAL RESULTS

6.1 Comparison

Up to this point we have compared the results on the red, blue and green images separately; in this chapter we will compare color images. Figure 48 shows the full color interpolated chirp image and the diamond-shaped Nyquist regions are marked to aid in analyzing these images. We see that the proposed method shows a complete alias cancellation in mid-frequency regions (regions at an angle of 45° from the origin) and regions just outside the marked Nyquist region along the horizontal and vertical high-frequency directions. Figure 49 show the error images (difference between the original and output image). We notice that in Figure 49(e) there is a complete alias cancellation within in the diamond shaped Nyquist region as in Figure 49(c), also some regions outside the Nyquist region show alias cancellation. Figure 49(d) show bright regions indicating the presence of aliasing in all three channels in those locations.

Figure 46 shows a comparison of the fence region from the lighthouse image, which suffer significantly due to aliasing artifacts. Since the average error levels in these algorithms are very low we also compare the error images. We see that the aliasing artifacts are less apparent in the proposed method as compared to the other methods and less energy in the error image suggests a better interpolation.

The results in Figure 47 confirm that the proposed method has the lowest average MSE over the 24 color test images. Table 4 summarizes the MSE dataset obtained for the three channels. The standard deviation and Inter-Quartile ranges are calculated so that the outliers in the dataset do not bias the results. We see that the proposed algorithm marginally outperforms the other two algorithms used.

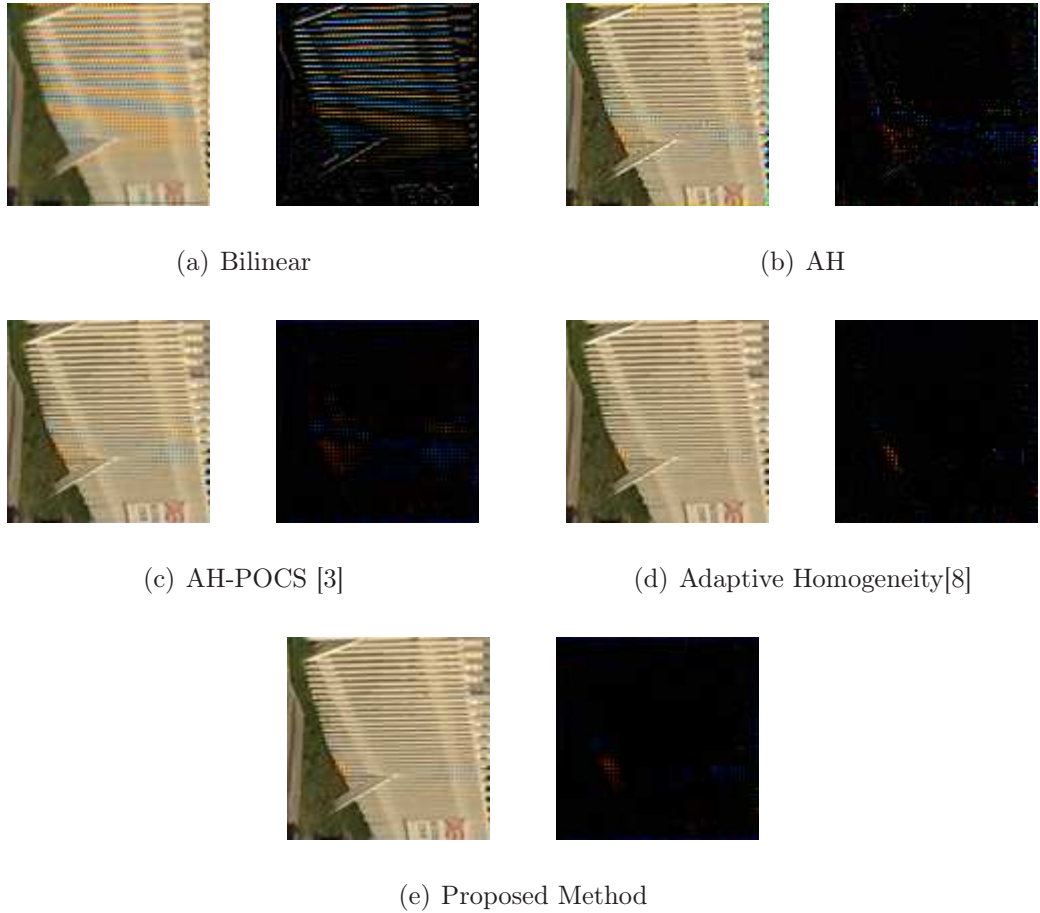


Figure 46: Aliasing present in the Fence image and error image.

Table 4: Mean, Std. Deviation and Inter-Quartile Range (IQR) in MSE for the 24 color images in Kodak color image database. (a)AH-POCS (b) Adaptive-Homogeneity (c) Proposed Method.

	(a)			(b)			(c)		
	R	G	B	R	G	B	R	G	B
Mean	6.15	2.45	6.74	5.86	3.91	7.11	4.51	2.32	5.48
Std. Deviation	3.50	1.23	3.92	3.35	2.71	4.69	2.43	1.30	3.40
IQR	4.03	1.88	5.74	5.35	3.02	5.52	4.22	1.91	4.17

6.2 Color Image Results

We will conclude this chapter by presenting full color interpolated output images. For each example image shown, we compare the original, bilinear interpolation method, AH-POCS [3] and the proposed method.

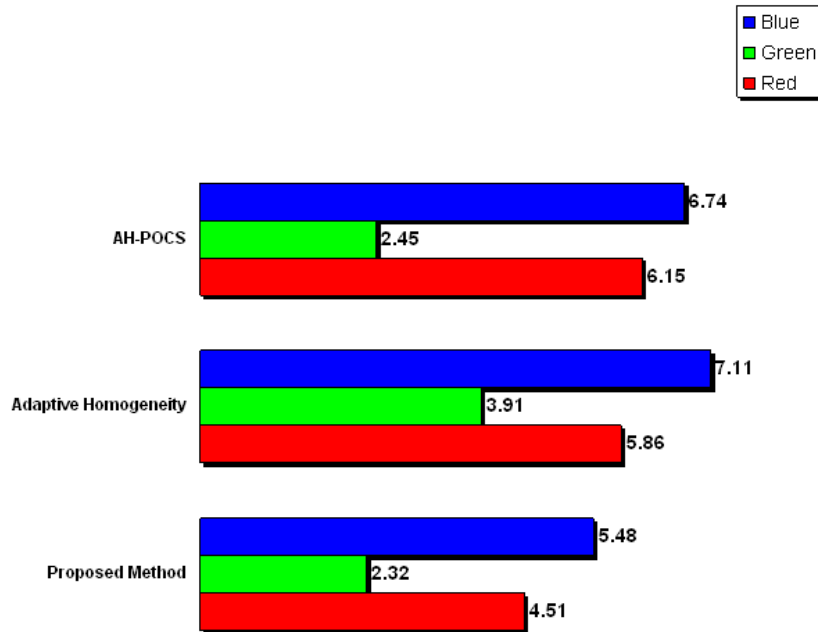
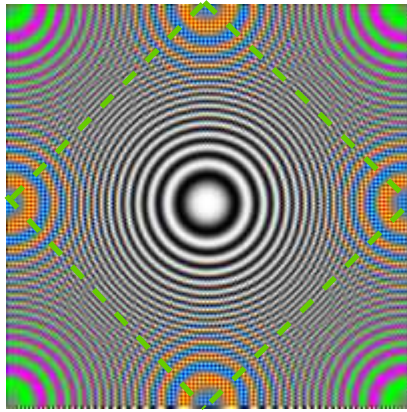


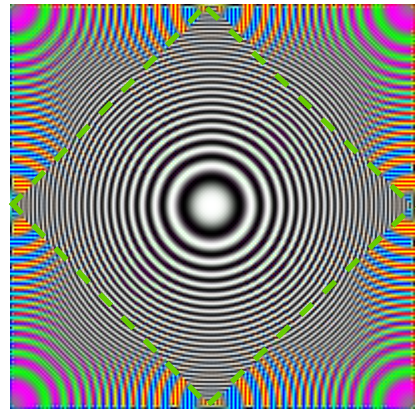
Figure 47: Average MSE over the 24 color images.

The first image is the Lighthouse image, which has been used throughout the thesis to compare results. The aliasing artifacts become apparent in the regions containing higher intensity, the disturbing artifacts along the fence and the wall of the house in the bilinear interpolation illustrate this fact. This is reduced to a large extent in the AH-POCS method, and the proposed method shows a slight improvement along the picket fence when compared to AH-POCS.

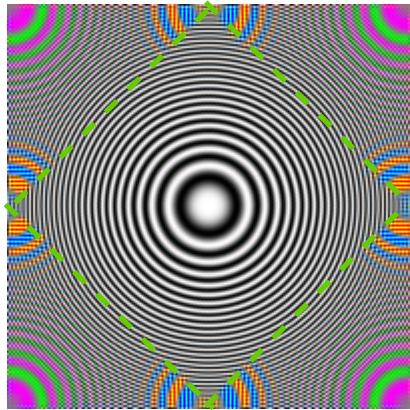
The second set of images (Houses) contains a row of houses. In the bilinear interpolation we see aliasing artifacts along the window panes of these houses. This is because we see sharp edges in the original image at these locations. Also the edges in the image look blurred in bilinear interpolation. The AH-POCS and the proposed method outputs are very similar and both methods remove the aliasing to a large extent and also add sharpness to the image.



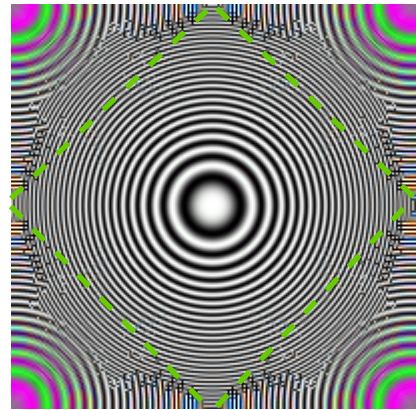
(a) Bilinear



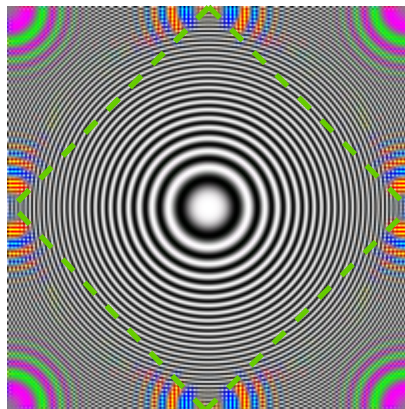
(b) AH



(c) AH-POCS [3]

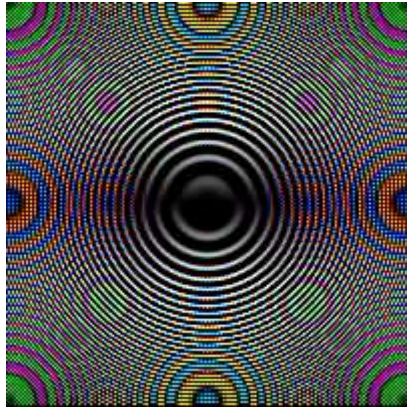


(d) Adaptive Homogeneity [8]



(e) Proposed Method

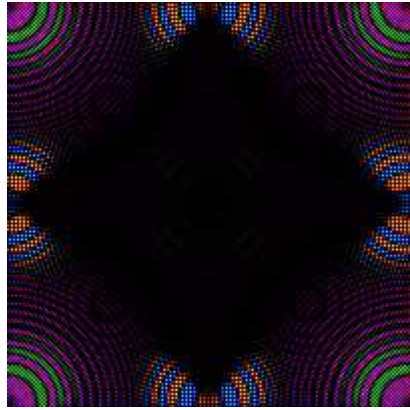
Figure 48: Result on chirp image.



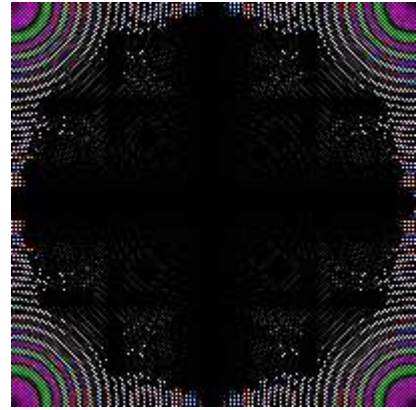
(a) Bilinear



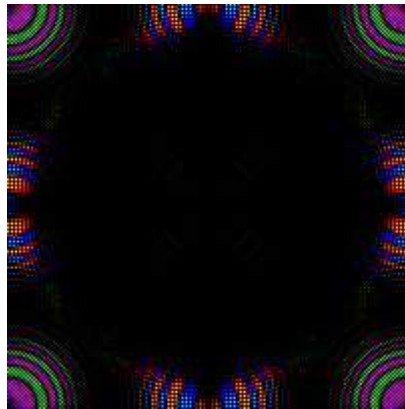
(b) AH



(c) AH-POCS [3]



(d) Adaptive Homogeneity [8]



(e) Proposed Method

Figure 49: Result on chirp image.



(a) Original



(b) Bilinear Interpolation

Figure 50: Lighthouse image.



(a) AH-POCS



(b) Proposed Method

Figure 51: Lighthouse image (contd.)



(a) Original



(b) Bilinear Interpolation

Figure 52: Houses image.



(a) AH-POCS



(b) Proposed Method

Figure 53: Houses image (contd.)

Chapter VII

CONCLUSION

The thesis provides an overview of CFA interpolation used in digital cameras. It starts by describing simple methods for CFA interpolation, and later reviews more sophisticated algorithms which exploit the high correlation in the R,G,B channels. The red and blue are sampled at half the rate of green; hence green has more information than the other two channels. Using this fact, we use the green channel to cancel aliasing in red and blue channels.

Since the red and blue interpolation are completely dependant on how well the green channel is interpolated, how we interpolate green channel becomes very critical. The thesis dedicates a chapter on green interpolation, comparing different techniques to interpolate green. We conclude that combining the adaptive-homogeneity directed method and POCS generates a green image with the least MSE. Later assuming this green to be alias-free, we cancel aliasing in the red/blue channels to generate the complete color image. Results suggest that this algorithm performs better than some of the existing algorithms for CFA interpolation.

Future Work

This algorithm can work very well in cameras which provide RAW data as output, allowing the user to do post-processing offline. The research here has been aimed only at developing an algorithm to improve the results of CFA interpolation, hence the complexity of implementing on a real-time system has not been analyzed.

Also as mentioned earlier the final result is highly dependant on how we interpolate green image. Research on improving the green interpolation will generate better overall results.

REFERENCES

- [1] ADAMS, J., “Design of practical color filter array interpolation algorithms for digital cameras,” in *Proc. SPIE Vol. 3028*, p. 117 125, 1997.
- [2] ADAMS, J., “Design of practical color filter array interpolation algorithms for digital cameras, part 2,” in *Proc. SPIE Vol. 3028*, p. 488 492, 1998.
- [3] B. K. GUNTURK, J. GLOTZBACH, Y. A. R. W. S. and MERSEREAU, R. M., “Demosaicking: Color filter array interpolation in single chip digital cameras,”
- [4] BAYER, B. E., “Color imaging array.,” 1976.
- [5] DUDGEON, D. E. and MERSEREAU, R. M., *Multidimensional Signal Processing*. Prentice Hall, Inc., NJ: Englewood, 1984.
- [6] GLOTZBACH, J. W., *A Color Filter Array Interpolation Method Based on Sampling Theory*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, August 2004.
- [7] GLOTZBACH, J., SCHAFER, R., and ILLGNER, K., “A method of color filter array interpolation with alias cancellation properties,” p. I: 141 144, 2001.
- [8] GUNTURK, B. K., *Multi-Frame Information Fusion For Image And Video Enhancement*. PhD thesis, Georgia Institute of Technology, Atlanta, GA, August 2003.
- [9] GUNTURK, B. K., ALTUNBASAK, Y., and MERSEREAU, R. M., “Color plane interpolation using alternating projections,” *IEEE Transactions on Image Processing*, vol. 11, p. 997 1013, Sept.2002.

- [10] GUNTURK, B. K., GLOTZBACH, J. W., ALTUNBASAK, Y., and MERSEREAU, R. M., “Demosaicking:color filter array interpolation,” *IEEE Signal Processing Magazine*, p. 44 54, Jan.2005.
- [11] HAMILTON, JR., J. F. and ADAMS, JR., J. E., “Adaptive color plan interpolation in single sensor color electronic camera.,” 1997.
- [12] HIRAKAWA, K. and PARKS, T. W., “Adaptive homogeneity-directed demosaicking algorithm,” *IEEE Transactions on Image Processing*, vol. 14, no. 3, p. 360 369, March.2005.
- [13] HIRAKAWA, K. and PARKS, T. W., “Adaptive homogeneity-directed demosaicking algorithm,” in *ICIP (3)*, p. 669 672, 2003.