

Demosaicing Methods for Pseudo-Random Bayer Color Filter Array

Georgi S. Zapryanov and Iva N. Nikolova

Abstract—In order to record color images in single-sensor digital cameras the incident illumination to the two-dimensional image sensing array (CCD or CMOS) is filtered so the different sensing elements receive different colored illumination. The filters are arranged in patterns across the face of the image sensing array. One of these arrays is Pseudo-random Bayer CFA. In contrast to an original Bayer, the filter here is received by overlaying of Pseudo-random pattern. This allows the images that are concurrent to the original Bayer’s filter regular structure to be better perceivable. The objective of this paper is to introduce and compare a few commonly used interpolation algorithms that are adapted for Pseudo-random Bayer CFA. Their efficiency is assessed by means of experimental studies with different type test images. An error metrics like mean squared error are calculated in RGB color space.

Index Terms—Demosaicing, Pseudo-random Bayer pattern, color filter array (CFA), Interpolation, Image Processing

I. INTRODUCTION

IN the recent years digital image capturing has found wide application in different areas. Unlike color film, all image sensing sensors [5], except Foveon are sensitive to the luminance component of color. In order to extract the color information the sensing elements are overlain by a matrix with alternate color filters, most frequently red, green, blue (RGB). The Bayer pattern [1] is one of the most generally used color filters. There are many different realizations of such patterns. One of them is Pseudo-Random Bayer (PRB) [2]. Here the pattern from Fig. 1 is received by overlaying of pseudo-random pattern. The green pixels location is saved as it stands at the original Bayer filter, as the blue pixels and the red ones determine the pseudo-randomness. Since the human eye is most sensitive to brightness and its peak of luminance response curve is around 550 nm (a green light), the green pixels are twice as many.

Obviously, the location of the separate filters in the PRB pattern in each sensing element (i.e. pixel) shows that we have information for just one color component (R, G or B). By analogy with color film, in order to receive full color image, we should have a three color planes (RGB) per each pixel. To

obtain the missing color information, we use the currently available color information using interpolation, also known as *demosaicing*. A comparative study of Bayer CFA demosaicing algorithms is presented in [3] and [4]. The purpose of this paper is to investigate the applicability of PRB filter for initiating visual images through modifying some of the mostly used demosaicing algorithms for PRB filter, and to make comparison between them, as well as between them and the “normal” Bayer filter. For the purpose of comparison a specially created vector and real photographic images are used. Two methods are used in order to estimate the algorithms for interpolation: *objective* – normalized MSE in RGB color space, and *subjective* – a visual comparison of the quality of the obtained results.

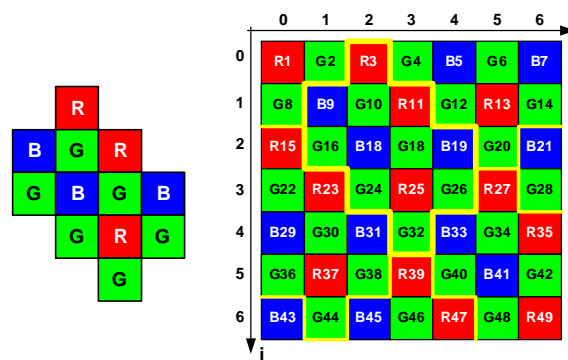


Fig. 1. Pseudo-Random Bayer CFA. Left: pattern; right: pattern, overlaid on matrix 7x7. A pattern is outlined with yellow color.

The paper is organized as follows. Section II, briefly presents some of the demosaicing methods and their essential features. The used test images and the motivation for selecting them are given in Section III. Section IV discusses results, obtained via our experimental studies. Finally, concluding remarks are provided in the last section.

II. AN INVESTIGATED ALGORITHMS AND FEATURES OF THEIR REALIZATION

In contrast to the original Bayer filter with PRB the processing is complicated by the fact that depending on its neighbors, pixels are divided into more groups. Depending on these groups some of the well known algorithms for interpolation could be fully applied as well as for PRB filter, with others the implementation is possible only for some types of pixels, while third algorithms cannot be executed with this

Manuscript received September 21, 2005.

G. S. Zapryanov is with the Computer Systems Department, Technical University of Sofia, Bulgaria (email: gszap@tu-sofia.bg).

I. N. Nikolova is with the Computer Systems Department, Technical University of Sofia, Bulgaria (email: inni@tu-sofia.bg).

type of pattern. Keeping in mind Fig. 1, we have the following position of the pixels, shown in Fig. 2, for which there is a measured value (red, green or blue center):

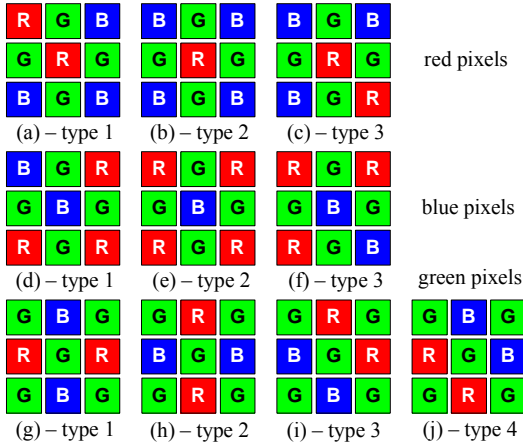


Fig. 2. Various type pixels in Pseudo-random Bayer filter

- Equations (1)-(3) give the conditions for finding the types of red pixels (red center), as showed in Fig. 2(a)-(c):

$$(((i+j) \bmod 6) = 4) \text{ and } ((|i-j| \bmod 4) = 2) \quad (1)$$

$$(((i+j) \bmod 6) = 0) \text{ and } ((|i-j| \bmod 4) = 0) \quad (2)$$

$$(((i+j) \bmod 6) = 2) \text{ and } ((|i-j| \bmod 4) = 2) \quad (3)$$

- Equations (4)-(6) give the conditions for finding the types of blue pixels (blue center), as showed in Fig. 2(d)-(f):

$$(((i+j) \bmod 6) = 4) \text{ and } ((|i-j| \bmod 4) = 0) \quad (4)$$

$$(((i+j) \bmod 6) = 0) \text{ and } ((|i-j| \bmod 4) = 2) \quad (5)$$

$$(((i+j) \bmod 6) = 2) \text{ and } ((|i-j| \bmod 4) = 0) \quad (6)$$

- Equations (7)-(10) give the conditions for finding the types of green pixels (green center), as showed in Fig. 2(g)-(j) (if $i>0$ or $j>0$):

$$(((i-1+j) \bmod 6) = 4) \text{ and } ((|i-1-j| \bmod 4) = 2) \quad (7)$$

or

$$(((i-1+j) \bmod 6) = 0) \text{ and } ((|i-1-j| \bmod 4) = 0)$$

$$(((i-1+j) \bmod 6) = 4) \text{ and } ((|i-1-j| \bmod 4) = 0) \quad (8)$$

or

$$(((i-1+j) \bmod 6) = 0) \text{ and } ((|i-1-j| \bmod 4) = 2)$$

$$(((i-1+j) \bmod 6) = 2) \text{ and } ((|i-1-j| \bmod 4) = 0) \quad (9)$$

$$(((i-1+j) \bmod 6) = 2) \text{ and } ((|i-1-j| \bmod 4) = 2) \quad (10)$$

A. Bilinear Interpolation

Let's consider the pixels array (Fig.1). The green color value to blue and red pixels to all three types is calculated similarly: the average of the four green neighboring pixels. The condition for blue and red pixel of unspecified type

is $(i+j) \bmod 2 = 0$. Evaluating the blue/red pixels on the green positions from Types 1 and 2 we take the average of pixel values from top-bottom or left-right and for Types 3 and 4 – left-bottom or top-right of the corresponding color. The evaluation of blue/red pixels from Type 1 and 3 upon the red/blue position is performed as average of the tree neighboring pixels with the evaluated color, and at Types 2 and 4 as average of the four neighbor pixels with evaluated color. For pixel location 25 (Fig.1) for instance, it is known only the value of R25. In this pixel $B25 = (B19+B31+B33)/3$, and $G25 = (G18+G24+G26+G32)/4$.

B. Smooth Hue Transition Interpolation – SHTI (Constant Hue-Based Interpolation, Interpolation of color ratios)

By definition *hue* is the most perceivable qualitative color characteristics due to which we can give a name to the color, for example red, green, yellow, etc., and it changes with the change of the length of the wave. SHTI, proposed by Cok [6] is one of the first methods used in the commercial digital cameras. Modifications of this method are still currently used yet. Algorithms as *Kimmel* [11] and *optimal recovery* [12] algorithms are developed on its base. In Cok the interpolation of the red and green color, using the green color can be found with some assumptions about the correlation between the separated color planes. One of the possible assumptions is that the ratio between the chromatic color components (red and blue) and the luminance (green) remains equal in the particular objects in the image (this is true in most of the cases, where there is no an abrupt transition). That's way the idea here is that instead of independent interpolation of both chromatic channels could be interpolated the relationships red/green and blue/green. This leads to much better results in comparison to bilinear interpolation, since the green component of PRB filter is with higher sampling frequency and can be restored much more precisely with the use of chromatic components after that.

In short the implementation of SHTI to PRB pattern is as follows: first, an interpolation of green pixels is done, using bilinear interpolation (note that this rule is binding to be done before the interpolation of the red and blue ones). Second, the red/blue pixels upon the green and blue/red position are interpolated by using Cok's algorithm [6], which is adjusted to the PRB, and both *linear* and *logarithmic* exposure space are realized.

C. Edge sensing interpolation

A human visible system is sensitive to the image edges and to the non-adaptive (described above) algorithms “falls” around the edge since they are not able to find them. The goal of edge sensing algorithms is to cope with the edge using an adaptive interpolation, based on evaluation of vertical and horizontal gradients.

1) *Hibbard's algorithm*: This method is proposed by Hibbard [7] and it is as follows: in order to find the edge direction are calculated horizontal ΔH and vertical ΔV luminance gradients (11), using the neighbours of the

interpolated on a blue/red position green pixel, accepting the edge direction of this pixel will coincide with that of the smaller gradient. After this step the value of the interpolated pixel is calculated as average value of both pixels in the edge direction. If such gradient is missing the four adjacent pixels are taken. The condition of blue and red pixel for arbitrary type is $(i + j) \bmod 2 = 0$.

After the green color plane interpolation, the missing red/blue pixels are interpolated by using the logarithmic exposure space, as previously described SHTI algorithm [6].

Lets it be necessary to calculate the value of the green pixel 25 (Fig. 1).The gradients and edge direction are given with following estimates (11):

$$\begin{aligned} \Delta H &= |G24 - G26| & \Delta V &= |G18 - G32| \\ \text{if } \Delta H > \Delta V & G25 = (G18 + G32)/2 & \text{else} & \\ \text{if } \Delta H < \Delta V & G25 = (G24 + G26)/2 & \text{else} & \end{aligned} \quad (11)$$

$$G25 = (G18 + G24 + G26 + G32)/2$$

3) *Laroche-Prescott's algorithm*: The method of Laroche-Prescott [8] is used in Kodak DSC200 digital camera. The gradients evaluation at the interpolation of the green pixels (Fig. 1), in contrast to the Hibbard's algorithm is performed through differences in the chromatic components of the adjacent pixels (12), as the edge direction is found as in Hibbard. With regard to the specific pixel location at the PRB filter such interpolation can be applied solely for evaluation of green pixels, located on blue/red positions of Type 2. An interpolation to all the rest green pixels, based on the Hibbard method is applied.

$$\begin{aligned} \Delta H &= |(R23 + R27)/2 - R25| & \Delta V &= |(R11 + R39)/2 - R25| \\ \text{if } \Delta H > \Delta V & G25 = (G18 + G32)/2 & \text{else} & \\ \text{if } \Delta H < \Delta V & G25 = (G24 + G26)/2 & \text{else} & \end{aligned} \quad (12)$$

$$G25 = (G18 + G24 + G26 + G32)/2$$

After the interpolation of the green pixels it follows with that of the red and blue ones, according to the method, described in [8], as for the particular type of pixels it is adapted to PRB filter.

D. Adaptive color plane interpolation

In [9] Hamilton and Adams consider area 5×5 pixels around each pixel and accept that the difference $G_i - R_i$ is approximately constant for such an area.

Here, first a reconstruction of the green color plane is performed, as by analogy to Laroche-Prescott the original demosaicing algorithm [9] is applied to the green pixels which are founded on the blue/red positions of Type 2. The Hibbard's method is used to all the rest green pixels.

The algorithm begins with calculation of the so called *red/blue "correction" terms*, which are actually Laplacian second derivative operators or first derivative operators. On the next step, these correction terms are compared with

previously defined threshold values, after which the green color value for the evaluated pixel is found. By using these correction terms the aliasing in the final image is reduced. After the end of green pixels interpolation, the red pixels and blue ones are interpolated as described in [9]. This method is adapted to the PRB filter for a particular type of pixels.

An example for red correction terms (see Fig.1): (13) is Laplacian second derivative operator, and (14) – Laplacian first derivative operator.

$$G25 = (G24 + G26)/2 + (2R25 - R23 - R27)/4 \quad (13)$$

$$G25 = G24 + (R25 - R23)/2 \quad (14)$$

E. Pattern recognition interpolation

The idea of Cok's method [10] is to define appropriate type pattern by evaluating the values of the four adjacent green pixels around every red/blue pixel. To PRB filter the Cok's algorithm is applicable to all pixels, where the green color interpolates. Let H denote a green pixel, which has a value that is greater than the mean value $m = (G1 + G2 + G3 + G4)/4$ of all adjacent green pixels around the interpolated pixel, and let L denote a green pixel, whose value is smaller than m . The four green pixels can form the following patterns (Fig.3):

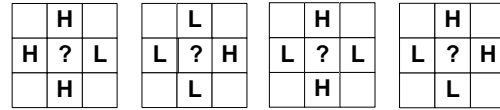


Fig. 3. Pattern types: (a), (b) – Edge Pattern, (c) – Stripe Pattern; (d) – Corner Pattern

After the type of the pattern is found, the corresponding green value on red/blue position is calculated. By *Edge Pattern* and *Corner Pattern*, in order to precise the missing value is made an extension in the area of the adjacent green pixels [10]. For *Stripe Patterns* this is made as follow (an example for Fig. 1): $G = \text{median}\{G18, G24, G26, G32\}$. An interpolation of the red/blue pixels on the green and blue/red position is implemented by using bilinear interpolation.

III. TEST IMAGES FOR COMPARISON OF DEMOSAICING METHODS

A. Type I Test Images: Synthetic vector images

The images of this type (Fig. 4) are created with Adobe Illustrator, after which they are transformed to raster images with Adobe Photoshop. Test images_{1,2} are used for evaluation of the algorithms for reproducing a smoothing transitions. Test image₃ contains a lots of high frequency pattern in the form of black and white sharp edges in different angles, divided in several semicircles for evaluation of the resolution. Each wedge is 2 degree wide, so the black and white rays are with a period of 4 degree. Test image₄ is created on by analogy with "Dot Distortion Target" of Sine Patterns [13], here imitating the regular structure of Bayer filter and the pixels in images (dot diameter and spacing = 10 pixels). Test

image₅ and Test image₆ are selected to display artifacts, caused by different black and white stripes.

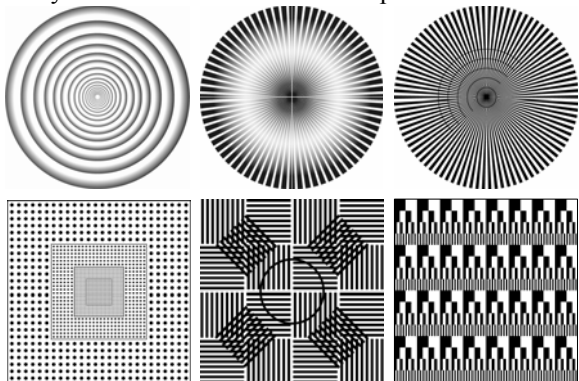


Fig. 4. Synthetic test images (images are numbered from 1 to 6 in the order of left-to-right and top-to-bottom) Test images_{1,2} for estimation of smooth transitions; (c) Test image₃ for estimation of sharp edges in different directions; (d) Test image₄ for estimation of regular structure; (e) Test image₅ is with lines with constant spatial frequency; (f) Test image₆ is with stripes with varying thickness.

B. Type II Test Images: Photographic images

To compare the demosaicing algorithms 14 photographic images are generally used: 7 of them are from the Kodak testbed (Test images₇₋₁₃ in that order: LightHouse, LightHome, RedRidingHood, Parrots, RomanStatue, Sailboats and Flowers (all are with 768x512 pixels resolution) and 7 represent high quality pictures (Test images₁₄₋₂₀ – Fig. 5). These images are selected so as to contain abrupt transitions as well as smooth ones, pastels and saturated colors, details with high spatial frequency.



Fig. 5. Collection of 7 autor's test images (images are numbered from 14 to 20 in the order of left-to-right and top-to-bottom). Test image₁₄ is “UnknownSoldierMemorial”, Test image₁₅ – “Nikolai’sPainting”, Test image₁₆ – “BachkovoMonastery”, Test image₁₇ – “Anastasia”, Test image₁₈ – “LondonEye”, Test image₁₉ – “Goblin”, Test image₂₀ – “SunClock”

Test image₁₄ contains white stripes over a red background (guard uniforms) and Test image₁₅ (iconography) is in dominant blue pastel hues. Test image₁₆ has multiple details with abrupt transitions with increasing spatial frequency,

while at Test image₁₇ contains basically smooth transitions. Test image₁₈ contains a great number of small details (London Eye cabins), Test image₁₉ – white figure and stripes and Test image₂₀ - abrupt transitions.

IV. EXPERIMENTAL RESULTS

Results from the described in Section II demosaicing algorithms for PRB filter, applied over both types test images are showed in Figs. 6-12. For objective metric mean squared error (MSE) is used, that gives the differences between colors similarly to Euclidean space. Although, this approach is not directly connected with the human visual system, it is widely used because of its simplicity and interpretation of results. In this case is used normalized MSE (normalization towards minimal value of MSE for red, green or blue), because the relative performance of algorithms is important for their estimation. Normalized MSE for 10 test images are presented in Table 1.

During the investigation of the demosaicing algorithms for PRB CFA we receive results that are similar to those given in [3] for Bayer filter. Artifacts, described in the literature for Bayer filter were observed, as well as some, which result from the structure of PRB CFA. In Fig. 6 some of the basic shortcomings of demosaicing algorithms are shown (Test image₆): *zipper effect* and *blurring*. Zipper effect (fringe artifact) appears when it is interpolated around edges, where the color leap is abrupt. In this case, the edges look like zipper or colored fringes of a carpet. The effect is most visible in bilinear interpolation, but it can be found in the Cok’s methods as well. In bilinear interpolation, as well strongest blurring (smoothing) of image is observed. Laroche-Prescott interpolates precisely the vertical and horizontal lines, but introduces a few false colors in the angles. Hamilton-Adams’ method gives similar results, but there is a quite weak zipper effect. The quality of both algorithms is one of the best, because for estimation of missing information they use the other channels (chrominance channels for interpolation of luminance and vice versa).

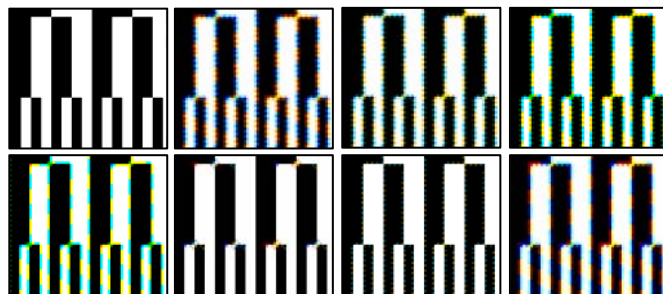


Fig. 6 Images are signed from (a) to (h) in the order of left-to-right and top-to-bottom: (a) – Region of interest (ROI) on test image₆ ; (b) – bilinear; (c) – SHTI in linear space; (d) - SHTI in logarithmic space; (e) – Hibbard; (f) – Laroche-Prescott; (g) – Hamilton-Adams; (h) – pattern recognition interpolations.

Since the diagonal top-left to bottom-right in PRB filter is

not well balanced with blue and red color, theoretically the number of artifacts there should be maximal. Exactly the same effect (green-blue diagonal), it observes in Test image₁ due to structure of PRB pattern (Fig. 7). The effect is strongly expressed visually as well as a metric in pattern recognition interpolation, but it observed in the rest of the algorithms. In Hamilton-Adams, this effect is missing almost.



Fig. 7. From left-to-right: (a) – ROI on test image₁; (b) – SHTI in logarithmic space; (c) – Hamilton-Adams; (d) – pattern recognition interpolations.

For Test image₄ a comparison between the execution of Laroche-Prescott's and Hamilton-Adams' algorithms for PRB and Bayer filters is made in order to establish how the regular structure transfers. In such image, PRB pattern works better than a Bayer CFA (Fig. 8)

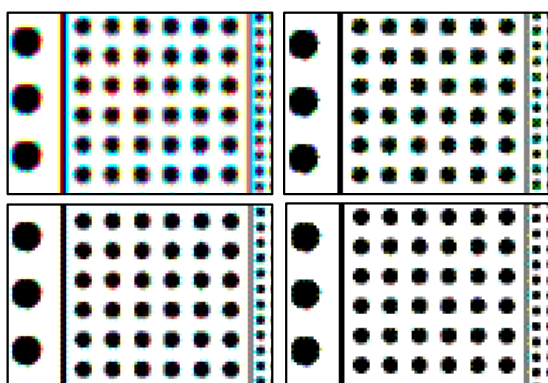


Fig. 8. ROI on test image₄ Left column: Bayer CFA; right column: PRB CFA; top row: Laroche-Prescott interpolation; bottom row: Hamilton-Adams interpolation.

When there is a smoothing transitions (Test image₂) most of the algorithms, except bilinear and pattern recognition, interpolate very well (see Table 1). In the center of the star (Fig. 9), where there is a high spatial frequency, all algorithms introduce another demosaicing effect – *aliasing*. Between the individual parts of the star, zipper effect in SHTI-s is observed. Such effect is missing in Hibbard and Laroche-Prescott. In Hamilton-Adams it is weakly visible, only in isolated spots, nevertheless it gives the smallest MSE.

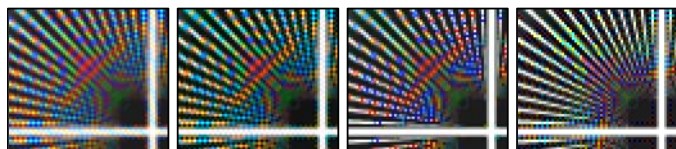


Fig. 9. Aliasing and zipper effect – From left-to-right: (a) – bilinear; (b) – SHTI in logarithmic space; (c) – Hibbard; (d) – Hamilton-Adams interpolations.

In images that contain sharp edges in random directions

(Test image₃), again much better results are received by the methods of Laroche-Prescott and especially Hamilton-Adams (Fig. 10), as once again bilinear interpolation introduces blurring and fringes. The reason is that this interpolation is low pass filter process and cannot “capture” correctly an image edges.

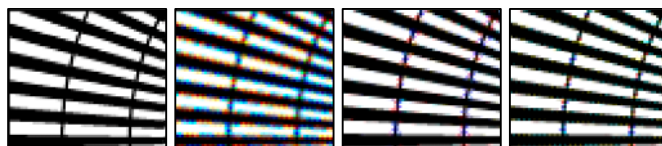


Fig. 10. From left-to-right: (a) – ROI on test image₃; (b) – bilinear; (c) – Laroche-Prescott; (d) – Hamilton-Adams interpolations.

When the images contains in certain areas some of both colors with low level spatial frequency (red or blue), most of the algorithms “fall through” in their reconstruction (see MSE results for Test image₉ in Table 1). The same problems occur in Test image₁₄ as well (Fig. 11) where in SHTI-s “corrode” effect is observed – bright points surrounded by dark neighborhood. Near the letters, which are parts with high spatial frequency, in all algorithms aliasing is observed. This is most weakly expressed again in Laroche-Prescott and Hamilton-Adams.



Fig. 11. Aliasing and “corrode” effect – From left-to-right: (a) – ROI on test image₁₄; (b) – SHTI in logarithmic exposure space; (c) – Laroche-Prescott; (d) – Hamilton-Adams interpolations.

Test image₁₀ (Parrots) notices the same shortcomings in image as these described in [3] for Bayer CFA. Confetti type of error [3] and aliasing are most remarkable in bilinear, Hibbard and pattern recognition interpolations, while in SHTI-s they are weakly expressed. Laroche-Prescott's, especially Hamilton-Adams' interpolations visibly reconstruct the region of interest (ROI) on Test image₁₀ (Fig. 12) precisely, but it introduces a few false color spots. In practice, these errors can prove almost unremarkable in print process.

In Table 1 are presented the experimental results obtained from calculation the normalized MSE in RGB color space, to 10 of 20 of the test images. Our main idea was to provide a sample of the obtained experimental results that is representative according to the evaluated demosaicing algorithms.

With the boldface numbers are presented the minimum values in the corresponding image, which give us an idea about which algorithm performs best for a given image.

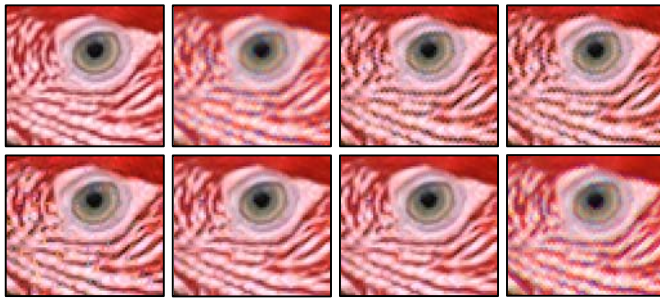


Fig. 12. Images are signed from (a) to (h) in the order of left-to-right and top-to-bottom: (a) – Region of interest (ROI) on test image₁₀; (b) – bilinear; (c) – SHTI in linear space; (d) – SHTI in logarithmic space; (e) – Hibbard; (f) – Laroche-Prescott; (g) – Hamilton-Adams; (h) – pattern recognition interpolations.

TABLE I
NORMALIZED MEAN SQUARED ERROR (MSE)

TEST IMAGES	BI	SHTI 1	SHTI 2	ESI1	ESI2	AI	PRI	
1	R	7.91	2.86	3.55	1.45	1.01	1.26	7.92
	G	2.88	2.86	2.88	1.23	1.47	1.35	6.62
	B	8.13	2.87	3.51	1.42	1.00	1.28	8.13
2	R	6.41	2.38	2.86	1.48	1.02	1.01	6.41
	G	2.35	2.32	2.35	1.38	1.32	1.00	4.90
	B	6.52	2.39	2.87	1.46	1.02	1.01	6.52
4	R	13.21	7.32	25.43	25.40	2.28	1.02	13.21
	G	4.88	4.88	4.88	3.15	3.12	1.29	8.48
	B	13.87	6.72	24.50	24.57	2.26	1.00	13.87
6	R	54.26	33.17	104.3	104.3	1.00	3.20	54.27
	G	13.51	13.51	13.51	1.35	1.35	6.38	4.80
	B	54.23	33.14	104.6	104.6	1.00	3.20	54.23
8	R	7.49	3.06	3.16	2.15	1.20	1.31	7.50
	G	2.21	2.21	2.21	1.90	1.00	1.16	2.88
	B	6.36	3.11	3.66	2.28	1.16	1.25	6.36
9	R	6.65	10.26	11.09	14.21	2.83	1.84	6.65
	G	2.08	2.08	2.08	4.71	1.50	1.00	4.00
	B	5.49	2.66	3.50	3.77	1.72	1.19	5.49
13	R	6.54	3.41	4.19	4.23	1.69	1.17	6.54
	G	2.37	2.37	2.37	4.54	1.43	1.00	5.30
	B	5.87	3.17	4.77	4.53	1.53	1.09	5.87
16	R	2.32	2.30	2.29	2.86	1.67	1.44	2.33
	G	1.27	1.27	1.27	1.54	1.00	1.01	3.23
	B	4.50	2.41	2.49	2.60	2.13	1.84	4.50
17	R	2.92	3.57	3.58	5.05	2.08	1.89	2.92
	G	1.20	1.20	1.20	3.58	1.00	1.13	2.78
	B	4.59	3.40	3.43	4.13	3.19	2.87	4.59
20	R	3.07	7.56	6.60	8.34	2.10	1.93	3.07
	G	1.38	1.38	1.38	1.58	1.00	1.12	3.91
	B	4.20	5.61	5.13	5.88	3.29	2.90	4.20

BI - bilinear interpolation; SHTI1 – Cok interpolation in linear exposure space; SHTI2 – Cok interpolation in logarithmic exposure space; ESI1 – Hibbard interpolation; ESI2 – Laroche-Prescott interpolation; AI – Hamilton-Adams interpolation; PRI – pattern recognition interpolation.

V. CONCLUSION

A comparative study of several demosaicing algorithms was present. Its goal was the investigation of the applicability of the PRB filter for capturing visual images.

Experimentally, it has been found that the Laroche-Prescott's and Hamilton-Adams' algorithms are best suited for all images, especially these with sharp edges.

We observe two types of specific error effects - "corrode" effect, which is result of the missing chromatic component during its interpolation (the problem of "missing" pixels) and "green-blue diagonal" effect that is due to the specific filter structure.

Our results show that despite the unbalanced structure along one of the diagonals of PRB filter and the most complicated developing approach, it is able to reconstruct the images as better as the original Bayer filter. In some images, PRB pattern works better than a Bayer CFA. Consequently, when using algorithms, especially created to take into account the specifics of this filter's structure, as well as in modifying others well known demosaicing algorithms, such filter could be used successfully in real practice.

REFERENCES

- [1] B. Bayer, "Color imaging array", U.S. Patent No.: 3,971,065, Eastman Kodak Company, July 20, 1976.
- [2] FillFactory, "The Color Filter Array FAQ", [Online], Available: <http://www.fillfactory.com/htm/technology/htm/rgbfaq.htm>.
- [3] R. Ramanath, W.E. Snyder, G.L. Bilbro, and W.A. Sander III, "Demosaicing methods for Bayer color arrays", *J. Electron. Imaging*, vol. 11, no. 3, pp. 306-315, July 2002.
- [4] B.K. Gunturk, J. Glotzbach, Y. Altunbasak, R.W. Schafer, and R.M. Mersereau, "Demosaicing: Color filter array plane interpolation", *IEEE Signal Processing Mag.*, vol. 22, no. 1, pp. 44-54, Jan. 2005.
- [5] G.S. Zapryanov, and I.N. Nikolova, "Image sensor for visual information", *Automatics & Informatics Journal*, 2005, No.2 (Abstract in English, text in Bulgarian).
- [6] D. Cok, "Signal processing method and apparatus for producing interpolated chrominance values in a sampled color image signal", U.S. Patent No.: 4,642,678, Eastman Kodak Company, Feb. 10, 1987.
- [7] R. Hibbard, "Apparatus and method for adaptively interpolating a full color image utilizing luminance gradients", U.S. Patent No.: 5,382,976, Eastman Kodak Company, Jan. 17, 1995.
- [8] C. Laroche and M. Prescott, "Apparatus and method for adaptively interpolating a full color image utilizing chrominance gradients", U.S. Patent No.: 5,373,322, Eastman Kodak Company, Dec. 13, 1994.
- [9] J. Adams and J. Hamilton, "Adaptive color plane interpolation in single sensor color electronic camera", U.S. Patent No.: 5,506,619, Eastman Kodak Company, April 9, 1996.
- [10] D. Cok, "Signal processing method and apparatus for sampled image signals", U.S. Patent No.: 4,630,307, Eastman Kodak Company, Dec. 16, 1986.
- [11] R. Kimmel, "Demosaicing: image reconstruction from CCD samples", *IEEE Trans. Image Processing*, vol. 8, no. 9, pp 1221-1228, 1999.
- [12] D. D. Muresan and T. W. Parks, "Optimal recovery approach to image interpolation", *IEEE Proc. ICIP*, vol. 3, pp 7-10, 2001
- [13] Sine Patterns – Standard Charts, [Online], Available: http://www.sinepattern.com/i_Stdrrds.htm.