Dzik Erwan January 2024

Report Image Analysis Project

In camera devices sensors are arranged in various dispositions to capture as much as possible of the visible light. Furthermore, due to physical limitation one sensor can only be sensitive to one of the RGB colors. The different arrangements are called Color Filters Array (CFA).

I. Problem statement

In this project we will be looking at 2 dispositions Bayer pattern and Quad-Bayer pattern displayed on the figure 1.

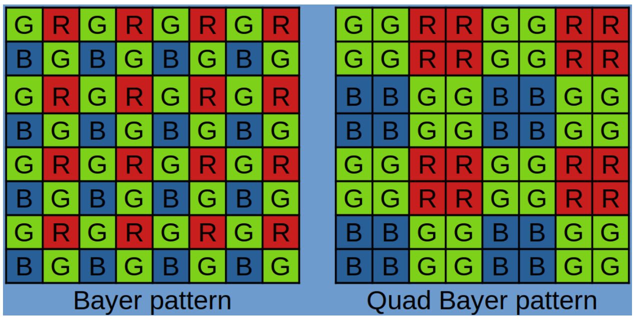

Figure 1: Pattern used in this project

The aim of this project is then to reconstruct the RGB image from the raw image from the sensor. This raw image can be seen as a "flat" RGB image which is thus a grayscale image. It is then an interpolation problem.

In order to evaluate the efficiency of the solution we will be using different metrics:

- Visual Analysis: Visual Comparison between the ground truth and the reconstructed image.
- PSNR : Peak Signal to Noise Ratio, Measure the proximity of the reconstructed image compared to the original one. It evaluates the distortion between the 2 images. The greater the value the better is the reconstruction.
- SSIM : Structural Similarity Index Measure, Measure the resemblance of the ground truth and the reconstructed image. This value is between 0 and 1, the closer it is to 1 the better is the similarity between the images.
- Computational cost/Duration: Considering that this process is embedded in a camera the computation cost/duration should be low in order to not slow the user (i.e the rendering shouldn't be noticeable by the user).

We will try to improve the results from the naive method proposed. These methods were:

- For the Bayer pattern: A simple bilinear interpolation. It uses only two dimensional convolutions
- For the Quad Bayer pattern: An adaptive filtering using adaptive kernels. As shown in chart 1, this method is very slow.

II. Proposed solution

For the Bayer pattern the first method is using the Malvar et al [1] algorithm. The proposed method is still a linear interpolation which is interesting considering our time limitation. This paper was initially published in order to improve the bilinear method. The idea is to correct the bilinear interpolation by adding a term which is the gradient of the 9 neighboring colors. It allows to enlight edges considering that they have a much stronger luminance than chrominance component. For example if we consider an initial R pixel (figure 1), we will estimate the green component $\hat{g}$ using (1):

$$\hat{g} = \hat{g}_{BI} + \alpha \Delta_R \tag{1}$$

where $\hat{g}_{BI}$ is the green component estimated using bilinear interpolation, Δ_R is the gradient of R for the R neighboring pixels and α is a correction term that monitors the intensity of the correction.
And so on, to estimate red at G pixel position we use

$$\hat{r} = \hat{r}_{BI} + \beta \Delta_G \tag{2}$$

To estimate red at B pixel position:

$$\hat{r} = \hat{r}_{BI} + \gamma \Delta_B \tag{3}$$

Then to determine the α, β, γ and thus the appropriate filter Malvar et al [1] used a Wiener approach. The resulting filters and their utilization are displayed in figure 2.

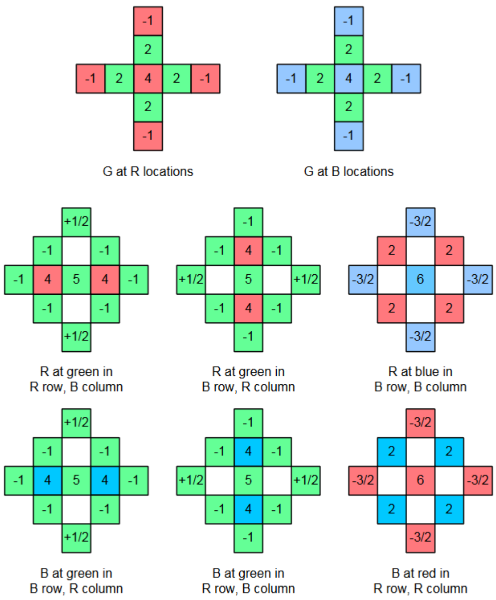

Figure 2: Filters used for the Bayer pattern proposed solution

We will now look to adapt this method for the Quad Bayer pattern. The idea is to modify the Quad Bayer disposition to a Bayer one. For that we use the swapping method explained by Pyxalis [2] in their user guide. The idea is just to rearrange the pixels. The algorithm is detailed in figure 3.

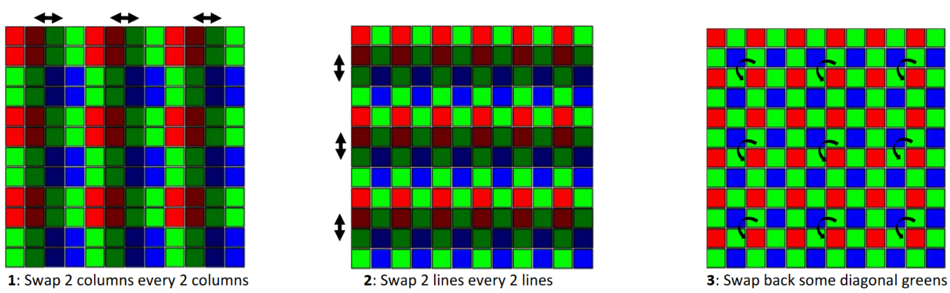

Figure 3: Quad bayer to bayer pattern by swapping method

Now that we have a Bayer pattern we just apply the Malvar et al [1] algorithm.

III. Results

This part is listing the results for the 4 test images from the dataset of the National Gallery of Art using the previously detailed methods. The chart 1 sums up all the metrics obtained.

	Quad Bayer (PSNR/SSIM [Computing time (s)])		Bayer (PSNR/SSIM [Computing time (s)])	
	Naive solution	**Proposed solution**	**Naive solution**	**Proposed solution**
`img1`	30.98/0.9108 [42.1]	30.21/0.9007 [0.6]	34.63/0.9502 [0.1]	39.14/0.9801 [0.2]
`img2`	26.96/0.7577 [45.0]	28.05/0.7650 [0.4]	30.31/0.8430 [0.1]	34.52/0.9443 [0.1]
`img3`	28.61/0.8280 [44.6]	28.55/0.8176 [0.4]	31.98/0.8941 [0.1]	35.31/0.9515 [0.1]
`img4`	26.65/0.7230 [42.7]	27.98/0.7379 [0.4]	29.88/0.8145 [0.1]	33.59/0.9228 [0.1]

Chart 1: Comparison between the proposed solution and the naive solution for the 4 images

For the Bayer pattern, we can see that the corrected algorithm improved significantly the results for all the images without degrading the computation duration. For the Quad Bayer the quality results are similar but the computation time has been reduced by a factor 100. From far away the image looks good and nothing seems wrong. But when we look a bit more closely on figure 4. In fact, for the Bayer pattern, the image is globally a bit blurry, the colors feels a bit more pale and it looks like the small spots on the bottom right are amplified and result in artifacts during the reconstruction.

Figure 4: Reconstructed, Bayer pattern, image (left), Reconstructed, Quad Bayer pattern, image (right) and the original image (bottom)

For the Quad Bayer pattern, the edges are not well reconstructed leaving artifacts and in general the image looks pixelated.

For other results we provide the output folder in which the reconstructed images are (Bayer and Quad Bayer)

IV. Conclusion

To conclude, we implemented two algorithms which improved, in the case of the Quad Bayer pattern, the computation time without degrading the reconstruction quality. And for the Bayer pattern enhancing the reconstruction quality without increasing the computation time. Moreover we saw that the reconstruction is far from perfect and there is still artifacts/damage after the reconstruction. In the case of preserving artwork the details of a painting are sometimes carrying important information. For example to evaluate which type of paint is used by looking at the cracks. Or to be more pragmatic, create a numerical image representative of the artworks without betraying it.

References

[1] H.S Malvar, L. He and R. Cutler, "High-quality linear interpolation for demosaicing of Bayer-patterned color images" Microsoft Research, 2004

[2] Pyxalis, "Image Viewer User Guide", 2022