Image Analysis
Demosaicing

Marty Benjamin - SICOM Ense3 - 2023/2024

Contents

Introduction

As a human, we can see the world thanks to our eyes, which are sensitive to light. This light is composed of several wavelengths. For humans, we are sensitive to wavelengths between 400 and 700 nm. Usually, we can represent the scene that we see with our eyes in an image of 3 dimensions, each dimension coding a specific wavelength, the red, green and blue. But, a single sensor cannot take measurements for multiple values of wavelength, so how can we have colorful images when we take a picture ?
It comes from the conception of the cell. We alternate every sensor so that we get a red value, then a green, then another one and so on. Following a pattern, we are able to know exactly what is the value and the wavelength associated with a single pixel. We then have to use an algorithm to regenerate the missing values. This is the goal of this project, to create an algorithm that creates the 3D image from a Bayern pattern.

Method chosen

I have chosen to code two methods. The first one (file high_quality_interpolation.py) is a better interpolation than the naive one. It uses bigger kernels and is inspired by [1]. The second method (file mdwi.py) is inspired from [2] and is also an interpolation but it is also using the gradient at some point, which is why it is called a Multidirectional Weighted Interpolation (MDWI). I did not code the whole process because I had some issues understanding the paper.
I wanted to pick up a simple method because this problem can be solved very easily with good results (example given) but trying to do a little state-of-the-art of this problem was something I wanted to do, so I coded the method 2.

The theory behind the first method is basically saying that the missing pixel is likely the same as his neighbors, so we will extract their information and use it to estimate the missing pixel.

The theory behind the second method is a bit more complex. First, we have the same principle that neighbors give information, but they also consider that the gradient between pixels of the same color gives information. As a result, we must compute the gradient in eight directions (N, S, W, E, NE, NW, SE, SW) and use it to predict the missing one. We first apply this method for the green channel because it contains twice more pixels (bayer pattern), and then we apply a mean-like method, with gradient information as well, to predict the missing red and green plans.

Results

We will first explore the results of both methods on the bayer pattern, then we will see how it goes on the quad-bayer pattern. These results come from my methods applied on the four images given, enumerated as it is in the folder.

BAYER PATTERN :

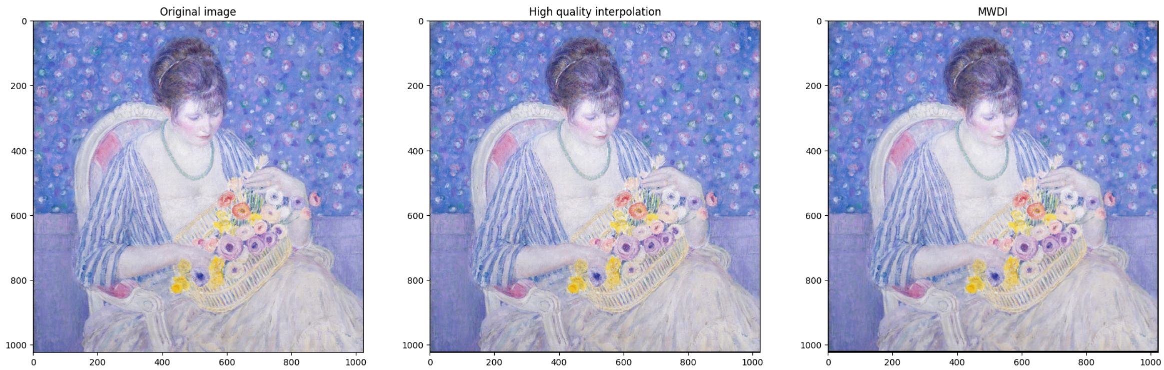
PSNR :

Image	1	2	3	4
Method 1	39.20	34.67	35.44	33.63
Method 2	36.69	32.20	33.70	31.48

SSIM :

Image	1	2	3	4
Method 1	0.98	0.94	0.95	0.92
Method 2	0.96	0.89	0.92	0.85

If we take a look at the images, we can see the reconstruction works pretty well for both method:



QUAD-BAYER PATTERN :

To process the quad-bayer pattern, I decided to downsample the quad-bayer image into a bayer one. Basically I compute the mean between each cluster of 4 pixels, so in the end I get a bayer pattern. I then apply the same methods described earlier and after the 3D image is computed, I upsample it and apply a filter to try to refine the image. Here are the results :

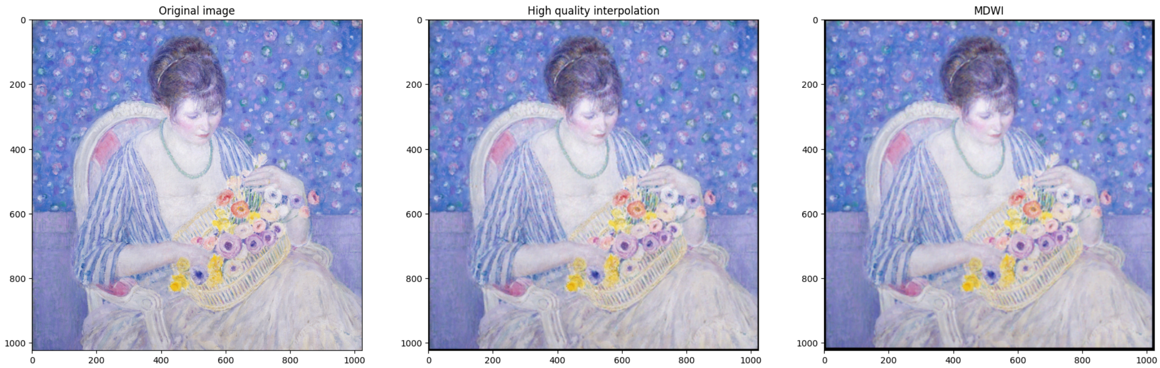
PSNR :

Image	1	2	3	4
Method 1	31.57	27.06	28.78	27.53
Method 2	27.46	21.93	24.33	21.98

SSIM :

Image	1	2	3	4
Method 1	0.91	0.73	0.81	0.71
Method 2	0.88	0.67	0.77	0.66

As we could expect, they are lower than for the Bayer pattern. It comes from the down and up sampling process, plus the mean filter which is applied right after the up-sampling.



As we can see, we lack some details on both methods compared to the Bayer pattern. It may come from the last operation, the refinement.

Here we can consider that the best method is the simplest one. We have better results on both patterns and both indicators. But it comes from the fact that I was not able to code the whole process from [2]. Normally, we would expect better results with method 2.

Conclusion

Regarding method 1, I do not think this method can be improved by taking a bigger kernel. I am scared that taking a bigger one would just make a too big low pass filter and we will lose edge information. Maybe we could change the value inside the kernel but it could overfit some images and this method will not be applicable for every problem.
But for method 2, there are things to do. As I said, I did not code the whole process because of some misunderstanding. They presented a post-processing method that is supposed to enhance the correlation over color channels in demosaicing methods [2].
I would like to add that I did not explore all theoretical methods, such as residual interpolation, and I did not explore deep learning methods.

Also, regarding the quad-bayer pattern, there must be a theoretical method but I only found deep learning ones.

Source

[1] : Henrique S. Malvar, Li-wei He, and Ross Cutler, “HIGH-QUALITY LINEAR INTERPOLATION FOR DEMOSAICING OF BAYER-PATTERNED COLOR IMAGES”, 2004

[2] : Xiangdong Chen, Liwen He, Gwanggil Jeon, Jechang Jeong, Member, “Multidirectional Weighted Interpolation and Refinement Method for Bayer Pattern CFA Demosaicking”, AUGUST 2015