

Demosaicing project

Demosaicing is an essential part of image processing: in order to obtain a colored image, the photons captured by the captor need to be counted as levels of red, green and blue (RGB). However, a photon does not carry this information intrinsically. That is why filters are applied at the entry of the captor, allowing each pixel to contain one information of either the red, green or blue captor. Demosaicing consists of retrieving the missing information in order to obtain a fully-colored RGB image.

First of all, we will further explain the task at hand, then offer a demosaicing solution to this issue, then show the results obtained with this method and explain the pros and cons of the proposed solution.

Table des matières

1. Task explanation.....	1
2. Another solution : the Superpixel method.....	2
3. Results.....	2
4. Conclusion.....	5

1. Task explanation

The issue is as follows: at first, we only have gray-scale images, with each pixel value corresponding to either the red, green or blue value retrieved by the sensor, depending on the following patterns (*Figure 1*). Here, we will only explore the bayer and quad-bayer pattern.

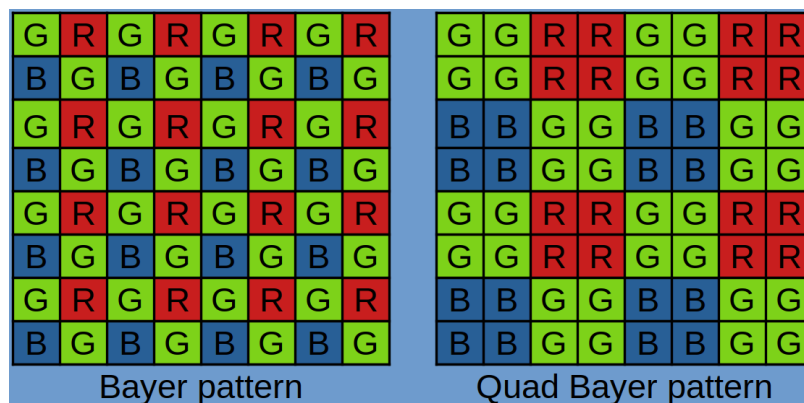


Figure 1 : image pattern

In order to obtain an RGB image, we need to obtain for each pixel levels of all red, green and blue channels, i.e. retrieving the missing two channels value for each pixel of the image. An already proposed solution is the naive interpolation between every pixel of the same channel. This method gives adequate results, but there is still room for improvement: when zooming in the image, it is possible to see false colors (i.e., bright colors in the eye).

2. Another solution : the Superpixel method

The superpixel method is a fast and very straightforward method that is useful when dealing with high-resolution (oversampled) sensors.

Each pixel is recreated by taking the value of the red, green and blue nearest pixels. As there are twice more green pixels in the initial image, the green channel is obtained by averaging the two nearest pixels. This method reduces the height and width of the image by two.

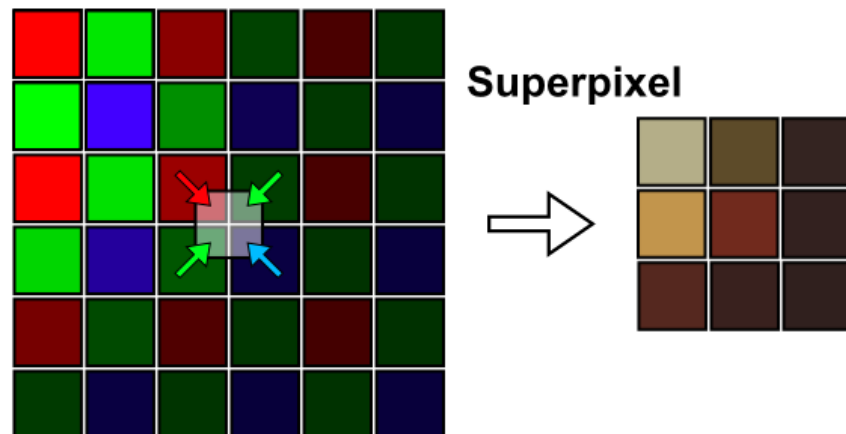


Figure 2 : Superpixel schematic (from pixinsight)

We implemented this algorithm by recreating a new image two times smaller than the initial image and then assigning the value of the red, green and blue nearest channels to every pixel. As said previously, this method is very straightforward and so is easily implementable.

3. Results

We tested the Superpixel method on both bayer (Figure 3) and quad_bayer pattern (Figure 4) on the four given images.

We obtained, because of this method, smaller images. As the initial images were quite big (1024x1024), the resulting images are adequate (512x512), but far from perfect. Furthermore, in order to compare the obtained results with the original image using the proposed methods (ssim and psnr), we had to enlarge the obtained image with the cv2 library functions, resulting in another loss of quality.

A lot of artifacts remain with this method. There are still false colors (in the detailed areas such as the eyes in image 1 or the hair in image 4), the image lost its texture (especially visible, because they are paintings), as well as its sharpness (because the resolution is smaller than the original). Still, when looking at the image from afar, the results seem correct: the image is correctly restituted.

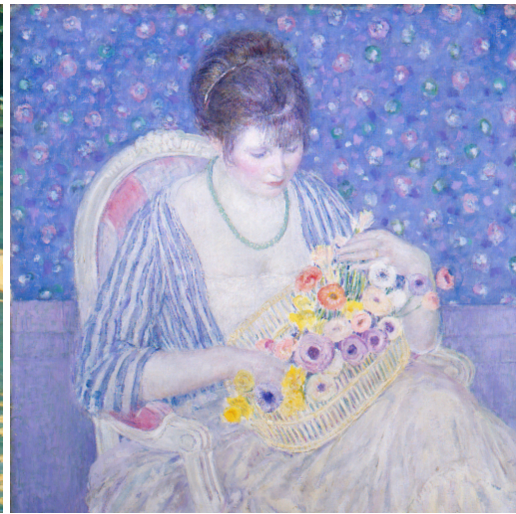
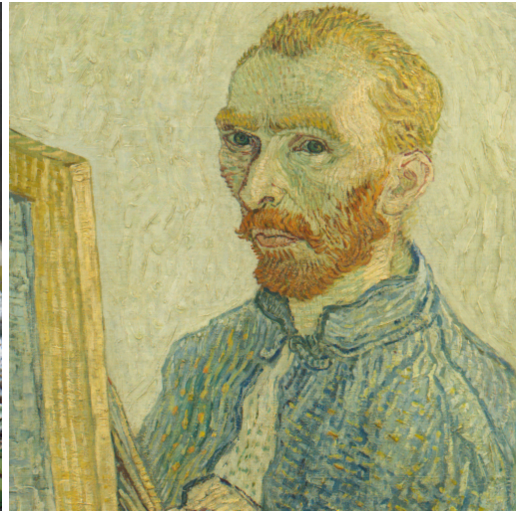


Figure 3 : "Demosaicked" images (bayer pattern)



Figure 4 : "Demosaicked" images (quad_bayer pattern)

When looking at the results, the bayer pattern showed better results, both qualitatively and quantitatively: the PNSR ratio and SSIM measure were around 30dB and 0,8 for the bayer pattern; and 20dB and 0,5 for the quad_bayer pattern respectively. Furthermore, when zooming in the image, we can see that the quad_bayer pattern adds another artifact, a grid effect, that does not show as much in the simple bayer pattern. The quad_bayer pattern also loses some detail (such as the nose in *Figure 5*) and adds false blue color (such as over the mouth in *Figure 5*).



Figure 5 : side-by-side comparison of bayer (left) and quad_bayer (right)

4. Conclusion

To sum up, the Superpixel method is correctly demosaicing images as we obtain seemingly good results. Still, when looking at the images in more detail, we can see a lot of artifacts such as false colors, blurring or a grid effect. The resolution of the obtained image is also worse than the original.

The big advantage of this method is that it is really fast and straightforward, which can be a plus when dealing with real-time processes where images need to be quickly “demosaicked”. Furthermore, the green channel is less subject to noise as we take for each pixel the average of two measures. It may also be useful for oversampled images where the resolution loss is not too impactful.

Other methods, such as variable number of gradients (VNG, or its improved version VNG4) can be more robust, but in order to implement them we would have needed more time, which sadly we did not have.