Advanced Interactive Media Design Ⅱ
Luiz Sampaio – 1351209

Image Filter

The aim of this exercise was to create a "beautiful" image filter. I was inspired and Andy Warhol and intended to create a filter that could "imitate" his artwork style, repeating in the same image copies of a portrait with different colors, resulting in a "cool" effect.

The algorithm is relatively simple, and I could achieve the desired effect in few steps.

1. *main()*

The main function is just a start point, as usual. It loads an image (in this case, hardcoded to make the tests more simple), and calls two functions `makePanel` and `makeWharol`. At the end, the algorithm presents the resulting image and saves in the disk.

2. *makePanel*

It is part of "Warhol" style to repeat the same portrait several times on the same "canvas". To make the algorithm run faster, I decided to repeat the image only four times. For that, first I had resize the original image by its half. Then I applied a blur filter, in order to remove some "noise" from the image.

Then, a loop calls the `makeWharol` function for times, one for each image. The resulting images are placed in one quarter of the final image.

3. *makeWharol*

This function is the "core" of the whole algorithm. One of main task is segment the picture in 2 color blocks. For that, I am using the `kmeans` function from OpenCV. However, before call this function, I have to provide "samples" of all parts of image, in order to allow the kmeans function to segment the color blocks properly.

The `kmeans` function also accept many different parameters that will change how the algorithm will perform the color segmentation. I cannot explain exactly how each of them influences in the result, so I decided to use how it was shown in the documentation. Besides these parameters for fine-tuning, this function receives the matrix of samples and the integer that represent the number of "color clusters" that we want to have.

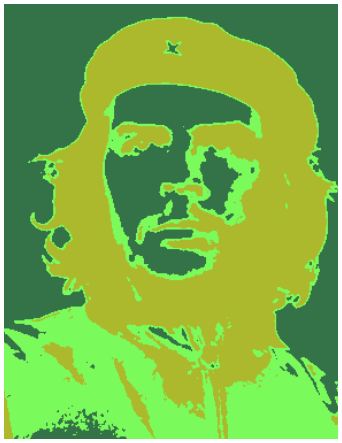

Figure 1

After that, we need to "paint" the image, according the result of the `kmeans` function. In order to achieve the intended style, the color are chosen randomly. Most of Warhol's artworks use three colors in each portrait; however, if we set the function to create three clusters of images, usually the result is not "cool" (fig 1) as expected, because the face should have a different color than the other parts.

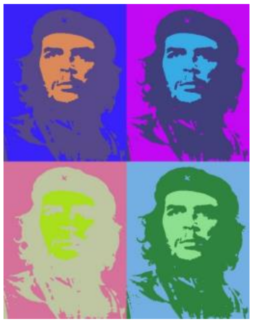

Figure 2

In order to solve this issue, I decided to segment the image in only two colors, and then apply another method to obtain the third color on the portrait's face. Then, my approach was using the `FloodFill` function to paint all the pixels around the center of image (usually the portrait's nose).

Then, to use the `FloodFill` function I just needed to pass the image, as the starting point for the "flood" and the desired color. The other parameters seemed to have almost no impact on the final result. The Figure 2 shows that the result is very close with the idea I had before write the code.