Tecnológico de Monterrey

Juan Pablo Aboytes Novoa A01701249

Image filter

Programming Languages

28/05/20

Index

Context of the problem

It has happened to all of us that we went out with a friend or family or your partner and we took a photo to remember, but there is a problem, even if it is a nice memory the photo did not go very well, something that could be solved with a program that will apply filters.

Solution

As I said before, the solution is to develop a program that can change the photo to make it look better, a simple program in sight and easy to use.

Image filter (non-technical description).

It is a program that occupies the size of your screen, within this a small section of buttons is shown on the left, within these is the button to open and save as and others that are hidden until you have a loaded image, (this it will be explained in more detail in the technical part of the report) these buttons are those that will apply the filters to the image loaded in the central part of the program.

Organization.

From here begins a more oriented language for people with programming skills.

The project is divided into three classes:

1.- Window class: This is the class that extends JFrame and where the interface part is defined, i used JButtons in their properties like if they are visible or not in the JPanels (i used 2), also i used an instance of Image_displayed class in order to show de image and this class is where i create the ForkJoinPool, something that it is explain better in the “Explanation of the code” section. It has the project main that initialize the window class.

2.- Image_displayed class: This is the class that extends canvas and process the image, is where depending which filter button you clicked it will prepare the image to be change it.

3.- Filter class: This class is in Image_displayed.java, it extends Recursiveaction and is where all the filters area applied depending which filter button you clicked.

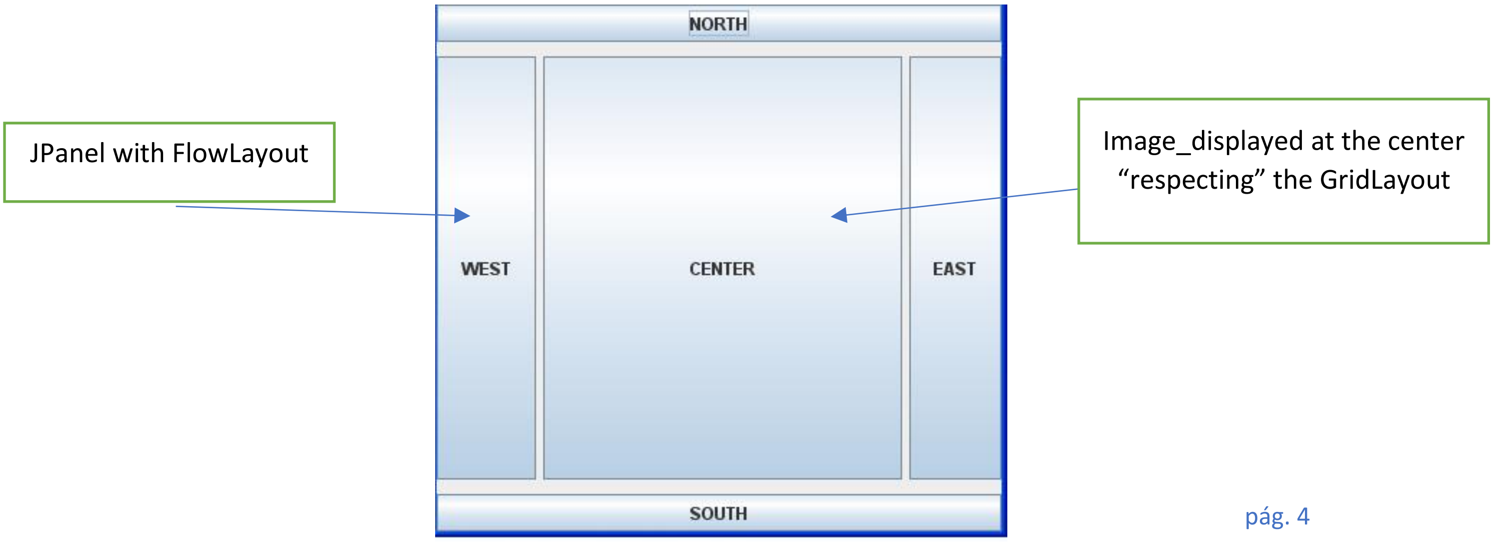
Explanation of the code

Window class:

Library or interface used**:**

- **Swing**: This is a graphical library, I used this library in order to create the JFrame that contains buttons that at the same time contains methods that make possible the use of the Image_displayed methods.
- **AWT**: I used this library in order to organize the graphical part, I used containers, color, and layouts in order to have a control of where the buttons and the image must appear.

As a said before is where i have the main, it instance the window class and it creates a Container that contains a Image_displayed instance and a JPanel inside another JPanel, the first JPanel have a GridLayout and the second a FlowLayout in order to have the buttons at the left and the Image_displayed instance at the center of the first JPanel (Image 1.0).

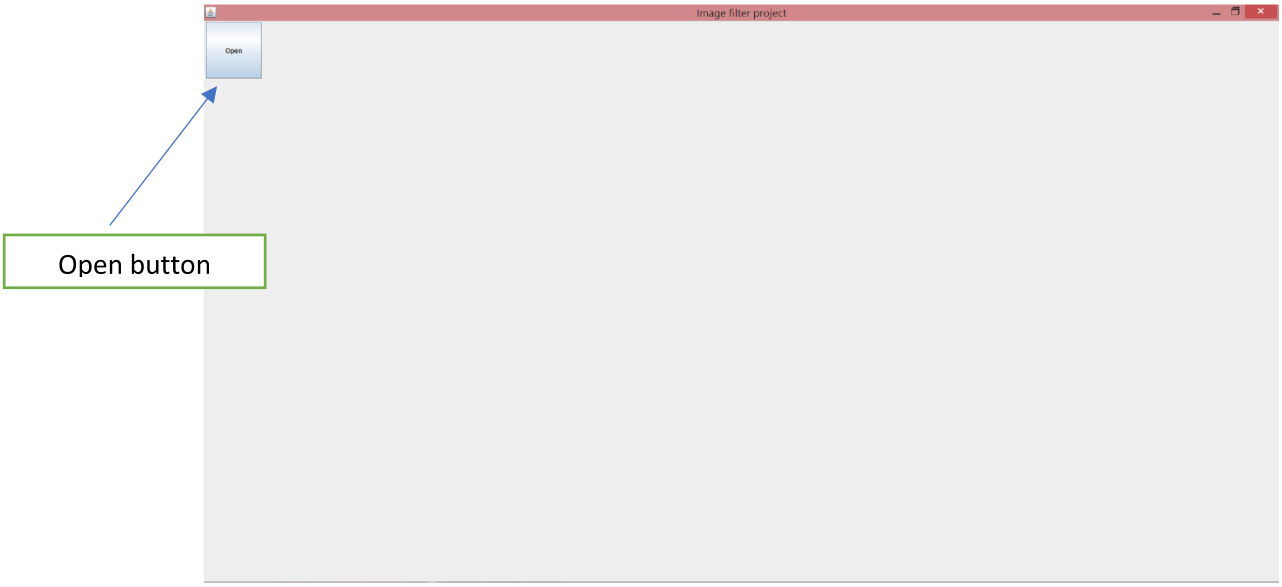


JPanel with FlowLayout

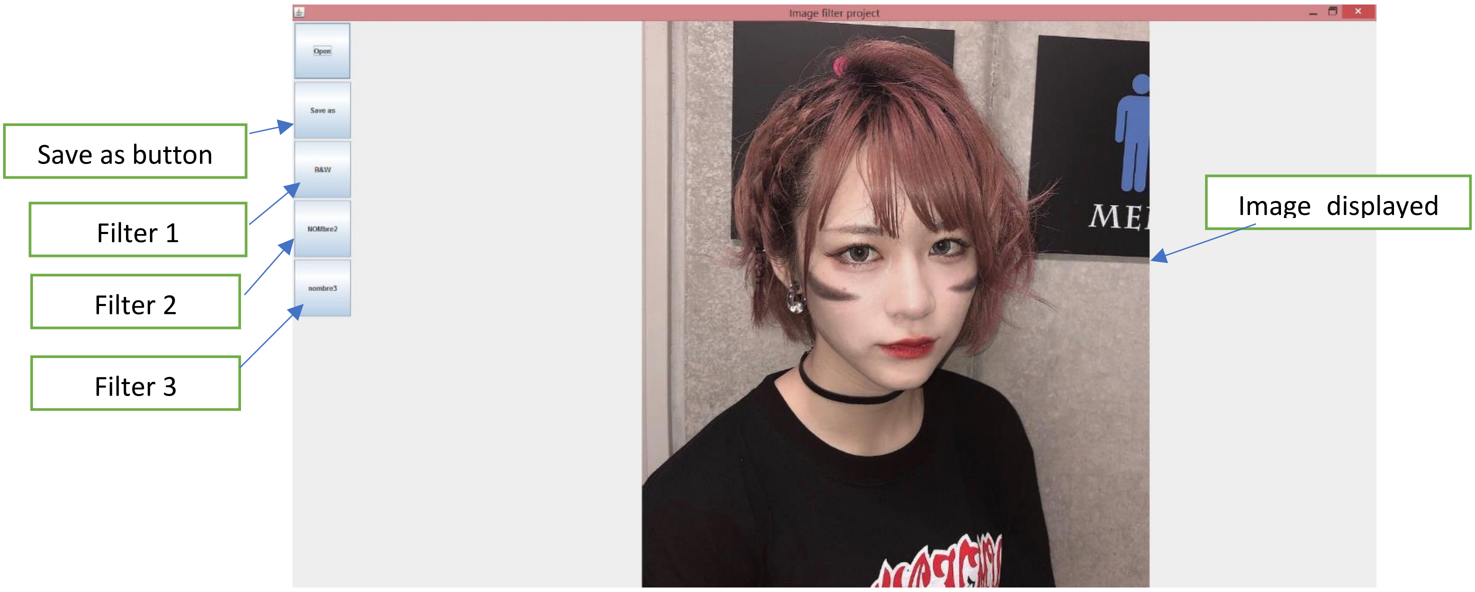
Image_displayed at the center “respecting” the GridLayout

(Image 1.0). Graphic organization

The graphic part contains 5 buttons, 1 of them are visible since the beginning (Open button), the Save as it can not be use until there is an image loaded. After an image is loaded the Save as button can be use and the other 3 buttons will appear below the Save as button, they are the filters buttons, you can click a button and then it will call a function in Image_displayed class that will change de image (Image 2.0 and 2.1).



(Image 2.0). Before uploading the image



(Image 2.1). After uploading the image

Image_displayed class:

Library or interface used**:**

- **IO**: As the name says “InputOutput”, I used this library in order to let the program access to the documents and create a new document.
- **AWT**: I used this library in order to organize the graphical part, I used containers, color, and layouts in order to have a control of where the buttons and the image must appear.

This class paints the image on the screen, using the function Paint() or repaint(), it also has a function that let you save the image (Image 3.0).

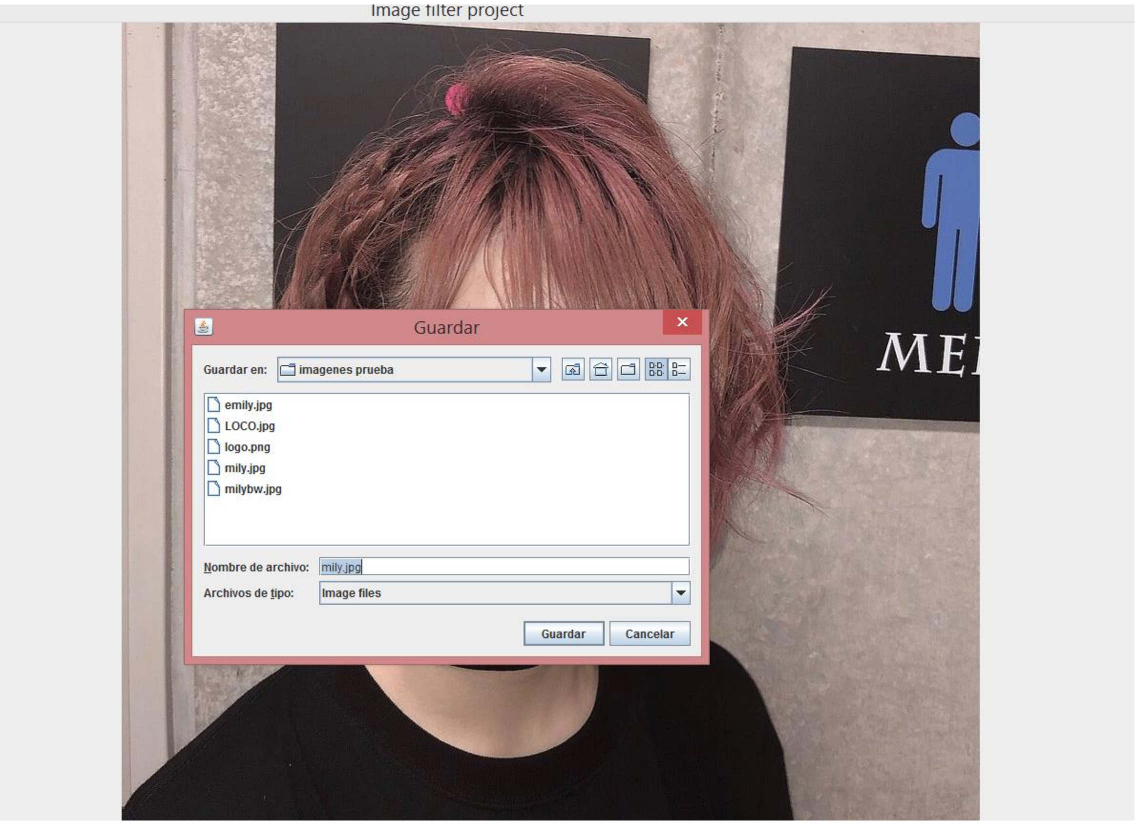


Image filter project

Guardar

Guardar en: imagenes prueba

emily.jpg
LOCO.jpg
logo.png
mily.jpg
milybw.jpg

Nombre de archivo: mily.jpg

Archivos de tipo: Image files

(Image 3.0). Save image.

Filter class:

Library or interface used**:**

- **RecursiveAction**: To make the threads execute the same task, is one way to use thread pool, it can divide the original problem into smaller parts and each small part executes the same code.

This class divides the input image and divide it in threads to execute the same code but with different parts of the image at the same time. The color of each pixel is obtained, is divided in channels (red, green, blue), its gray scale equivalence is obtained and the same value is assigned to both red, blue and green, finally the new value is assigned to the pixel. Let see an example(Image 4.0).

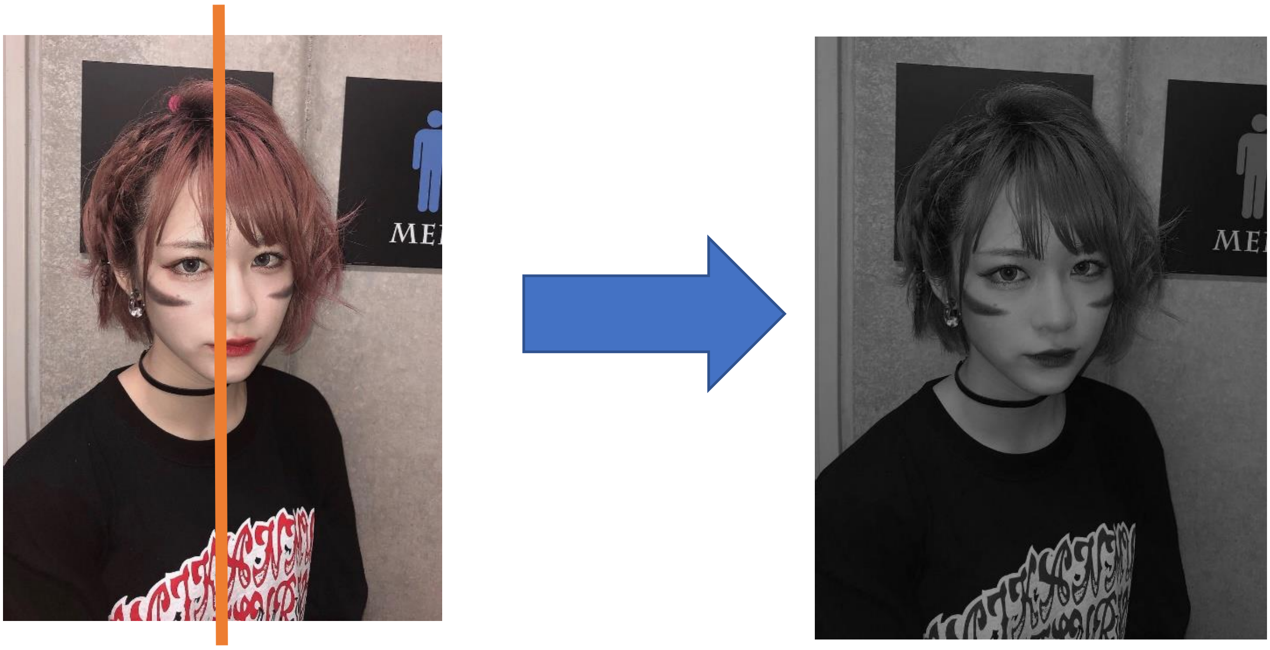


Image 4.0: How it is divided and applied the B&W filter

```
int heigth=img.getHeight();
   if (filter==1) {
        for (int i = mStart; i < mStart + mLength; i++) {
            for (int j = 0; j < heigth; j++) {
                int rgb = img.getRGB(i, j);
                int red = rgb & 0xFF;
                int green = (rgb >> 8) & 0xFF;
                int blue = (rgb >> 16) & 0xFF;
                float L = (float) (0.2126 * (float) red + 0.7152 * (float) green + 0.0722 * (float) blue);
                int color;
                color = 153 * (int) L / 255;
                color = (color << 8) | 153 * (int) L / 255;
                color = (color << 8) | 153 * (int) L / 255;
                dst.setRGB(i, j, color);
}}
```

Setup Instructions

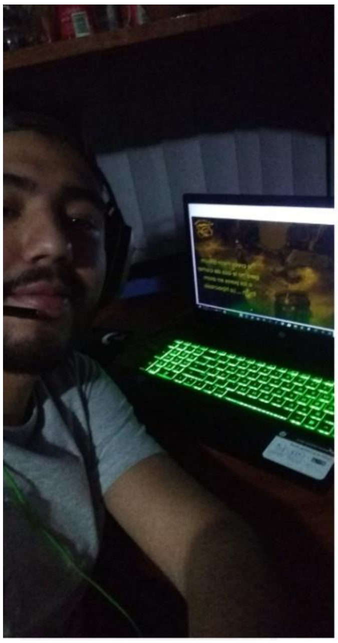
The code can be found at the following link:

https://github.com/JuanPabloAboytes/ImageFilterProject_A01701249

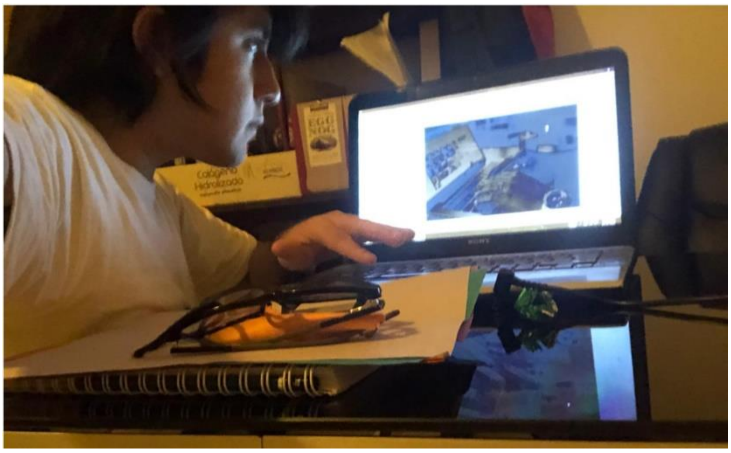
Right click on “Window” class and run the Project.

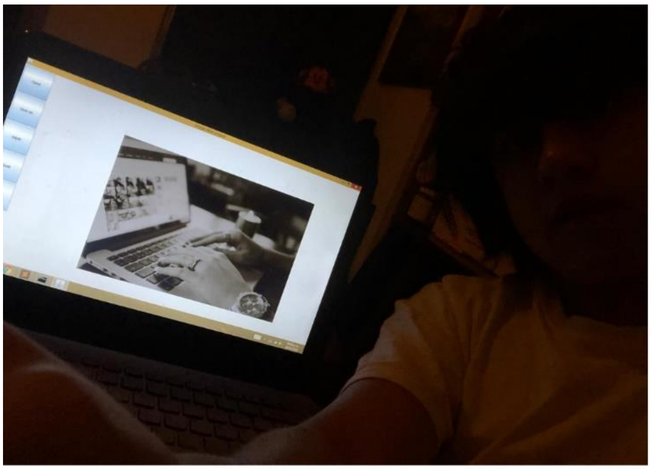
Tests and evidence.

The following images are the final version of the Project, i also put some tests that Friends made in order to get feedback.



Juan Carlos tested the Project and he said that he would like to have more filters because he did not like the filters i put.





Romel rested the project and he said that he would like to have an option to resize the image and change the brightness

I tested with a small image with a medium and a image in order to see that it works.

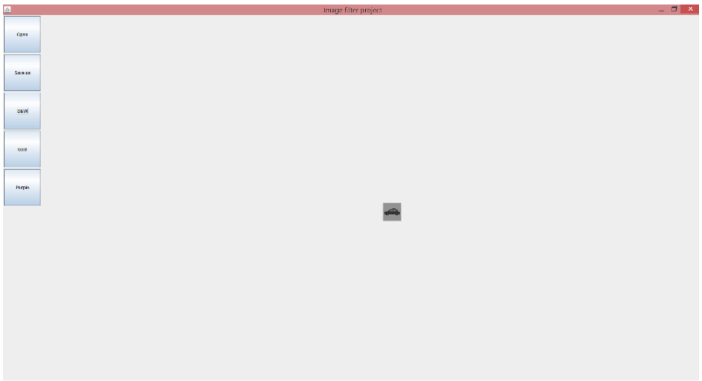


Image filter project
Open
Save as
B&W
Gold
Purple

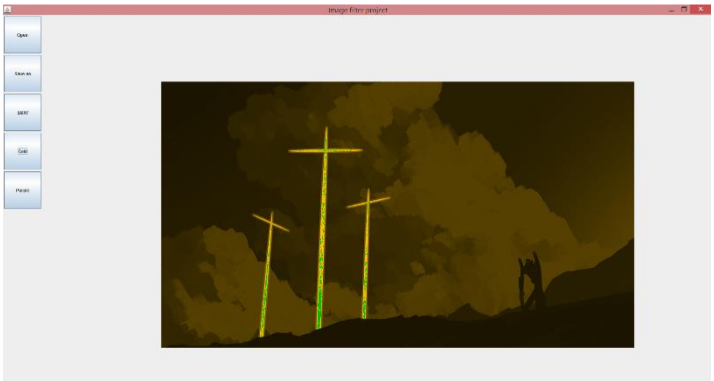


Image filter project
Open
Save as
B&W
Gold
Purple

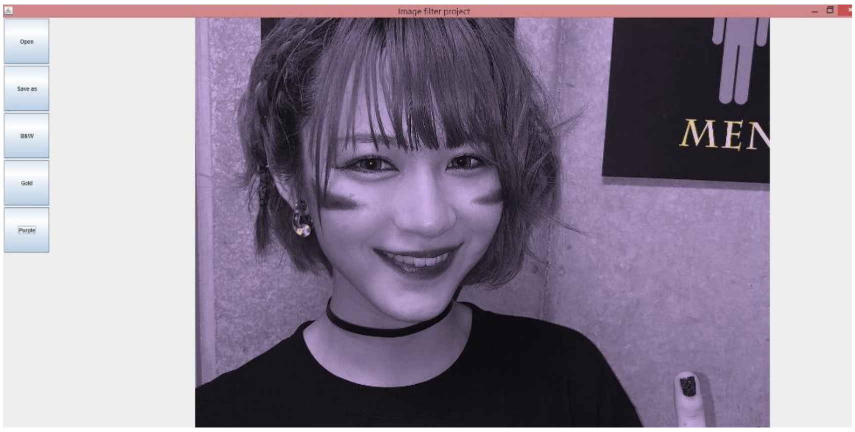


Image filter project
Open
Save as
B&W
Gold
Purple
MEN