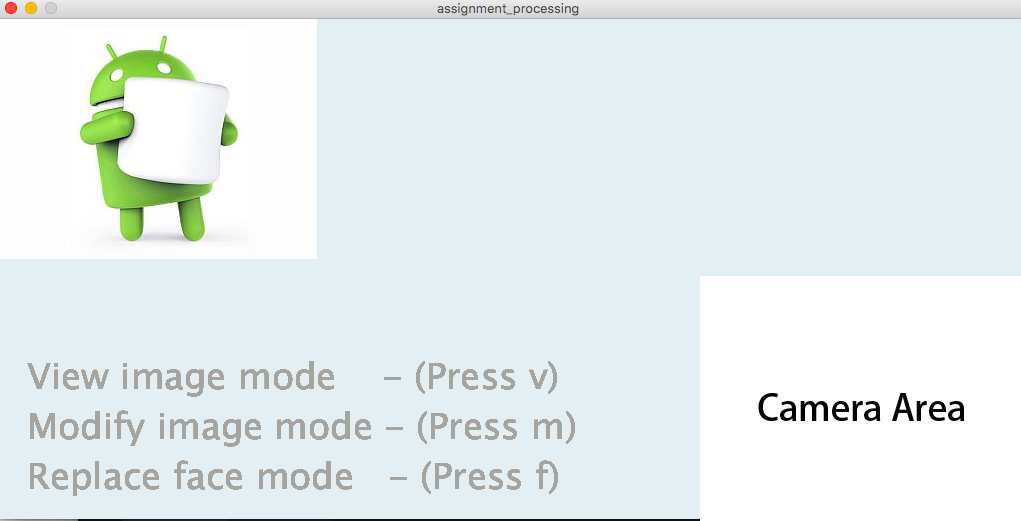
This is the personal assignment report about a Processing to develop an interface for viewing and interacting with an image. And this report will illustrate the design of the different sections of the Processing program with screen captures. And according to the guideline, it will have several limitations and possible improvements of the program.

1. Initial Setup

First of all, when user open and enter the program, the program will first do an initial setup – the image (pexels-photo-medium.jpg) will be show on the top-left corner, and the camera view will be display on the bottom-right corner of the window. The window size is fixed – in order to make most of users have space to interact with the interface between camera view and image, as well as not to exceed the computer screen size, so the screen size of this Processing program is 1024x 500.

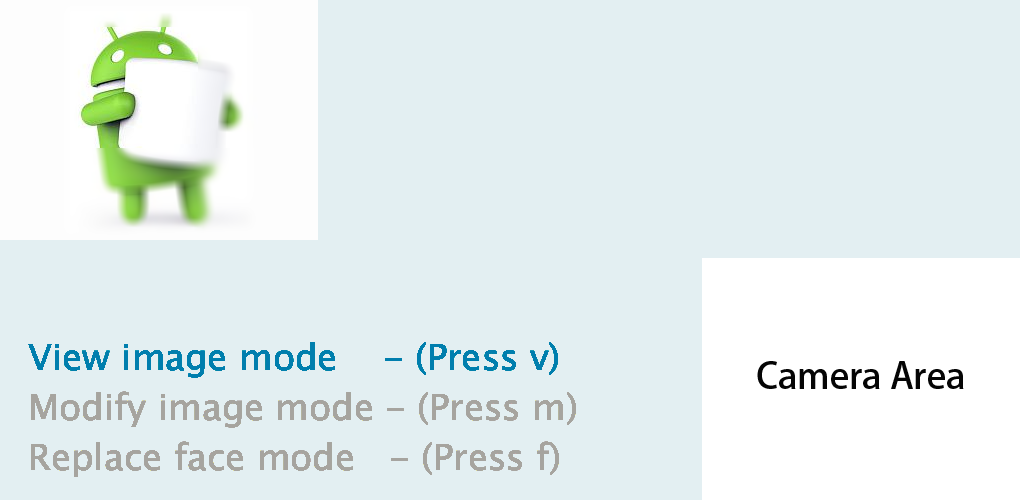


2. Face detection

About the image in the bottom-right corner of the window, it will keep captured image from the web camera and show the rectangle region on user face and keep following user face movement.

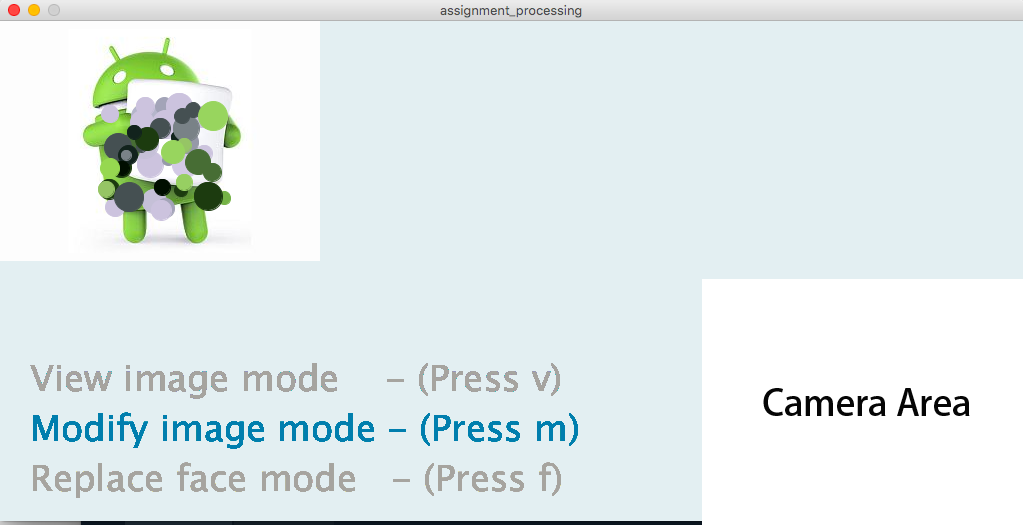
3. Viewing the image

Second, in the view image mode, a whole blurred version of the image (pexels-photo-medium.jpg) displayed at the very first time. The Android in the image is apply blur effect. When user face is moving across the image via the web camera, a rectangular area display the corresponding region from the original image without blurring in pexels-photo-medium.jpg. At the same time, the screen’s text statement “View image mode - (Press v)” will be highlighted with blue color, and other statement will keep remain in grey color.



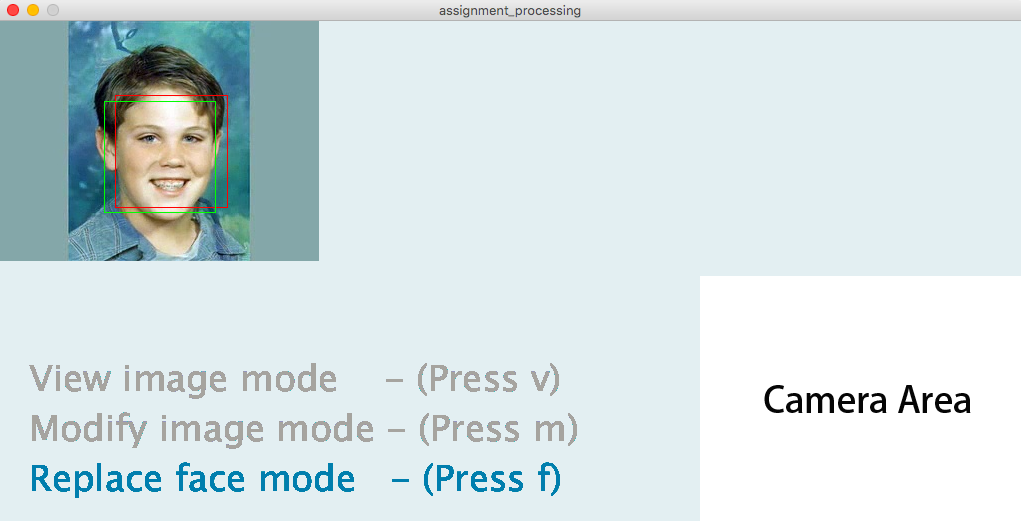
4. Modifying the image

Third, in the modify image mode, the Android image (pexels-photo-medium.jpg) displayed on the top-left corner. The interaction method between image and user face is same with the view image mode, but the effect is different; when user face is moving across the image, a small circle will be drawn at the position indicator. The small circle filled with a random color, when user moves away, the small circle will be retained.



5. Replacing the faces

When user press and release ‘f’ key, replace face, the interface enter the replace face mode. Since Android image is not a human face so I use another image (human.jpg) to show on the top left corner. When user face move across the image and user’s face is close to the human’s face in the image, user face captured by the camera will be replace with the human’s face on the top-left side image. When user’s face is move away from the human face for a certain distance, the human’s original face in the image will be re-displayed.



6. Exiting the view image/modify image/replace face mode

When the ‘e’ key is pressed and released, the interface will exit from the view image/modify image/replace face mode. The top-left corner’s image will be re-display the original image without any effect.

Due to the user able to change different mode in this program, and user may not aware which mode he/she’s in. There are region on the bottom-left corner that indicate which mode the user currently in- there are three statement represent the mode with a notice about press which key so as to go into the mode he/she want.

Moreover, in order to let user to switch mode easily, user no need to exit mode first and then press key to switch the correspond mode; he/she can just press key to switch the correspond mode and no need to press ‘e’ key, this approach can make user spend less time to switch the mode he/she want.

Limitations and possible improvements of the program

There are several limitations and still have room of improvements in this program. And those points are listed below:

·The window size is hardcode which is 1024x500. Which is still acceptable on most of the computer and laptop. But the best practice of this setting is change it to a resizable one – user can adjust the size they want, the loaded image and camera view should be keep adjust base on the screen window, at least all image and text size can be automatically scale according to the size of the current window.

·The program does not handle the error so well. There is possible to occur some errors which is inevitable. For example, the program run in a computer that does not equip any camera, so at the end the console from processing just pop up some red text warning indicate the error while user just keep waiting for the white blank screen. I think it should have a toast notification or a dialog to let user know the program is encounter problem.

·The image loaded in the top left corner can have further improvement – since some time some mode there is a need to pick up a image that contain human face, but sometimes not. So there should have a button to let user change the image he/she want.

·The rectangle draw by the OpenCV is not smooth enough – when detecting the face of user, the rectangle can be drawn, but it keep shaking while user face is hold still. In order to make it smoother, a function can be apply that first get a destination position and then move one point to another point gradually and keep adjust the change period of time base on the looping speed in Processing.