Alejandrina Patron Lopez
apl7@gatech.edu
GT ID 903075226
Project 1

Project 1 Write-up

What it does/Purpose

The main purpose of my project is to manipulate any square image to obtain a creative and unique output. My idea for this project was to create some sort of pattern or puzzle by working with an input image, while still obtaining an artistically pleasing output to look at.

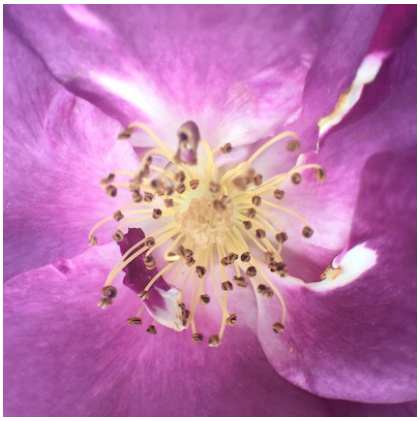
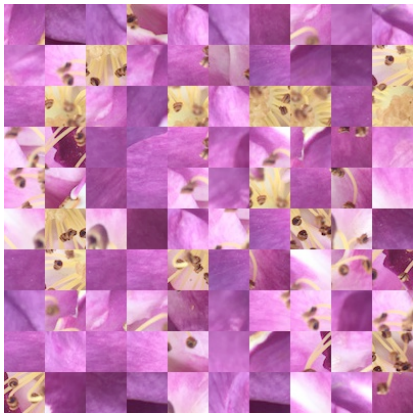
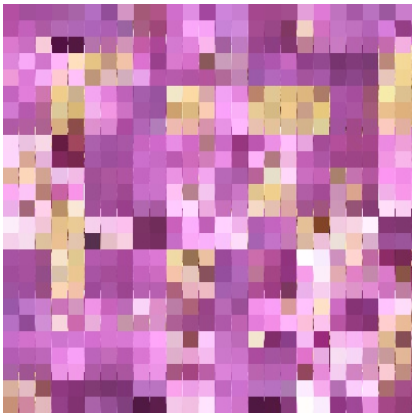
The code I wrote for this assignment computes the following tasks:

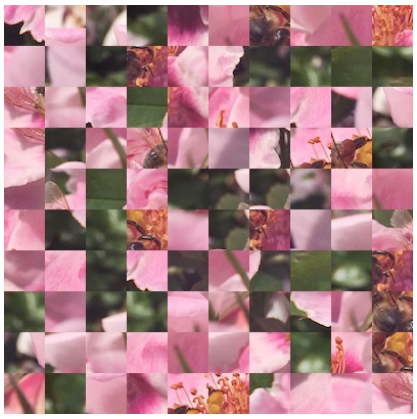
1. Turn the input image into a quilt
2. Manipulate the "quilt" obtained from step 1 to produce a pixel mosaic.

In fewer words, my project can be described as a "quilt generator" and a "pixelator".

Example Input/Output (Photos taken by me)

Input Output



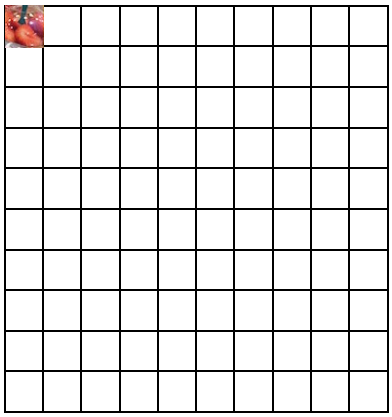

Alejandrina Patron Lopez
apl7@gatech.edu
GT ID 903075226
Project 1

Workflow

1. Input- One 400 x 400 image.

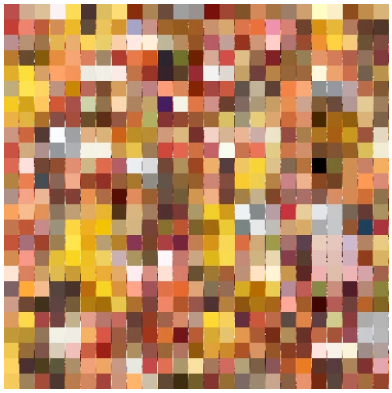
2. Output for Function quilt(inputIMG)

- This function generates random coordinates of the original image, creates a 40 x 40 square starting at that point, and adds it to the grid of the output image.
- The function iterates through every square in the grid, assigning a random "patch" of the input image on each square until the whole grid is filled.
- If the input image is larger, or smaller, than 400 x 400, the cropImage(inputIMG) function resizes it using cv2.resize().



Grid with patch on the first square.

Alejandrina Patron Lopez
apl7@gatech.edu
GT ID 903075226
Project 1



3. Output for Function pixelate(inputIMG)

- This function generates a “pixel mosaic” from an input image of size 400 x 400.
- It divides the input image into squares of 16 x 16, and assigns the color value in the middle of each square to the corresponding area in the output image (square at the same coordinates).
- Within the pixelate function, getBlue(point), getGreen(point), and getRed(point) are called to get the BGR values of a pixel.
- If the input image is larger, or smaller, than 400 x 400, the cropImage(inputIMG) function resizes it using cv2.resize().

Artistic/Technical Breakdown

- Artistic: 30%
- Technical: 30%