Benjamin Northrop
5/13/24
CS5310
Summer 24
Homework 1: What is an image?

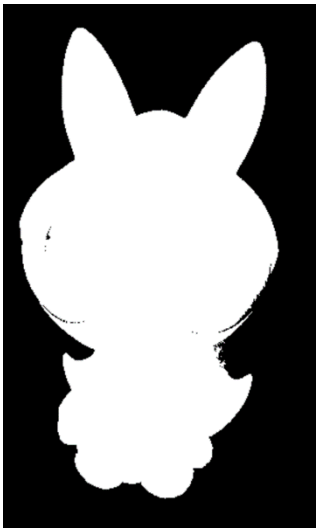
Introduction

This project was an intro to working with images in C as well as the concept of blending. It also included the use of green or blue screen behind a foreground image, and then incorporating a separate background to replace the green screen pixels. For the first portion, we rotated the color channels in the image so that the *values* of the RGB channel resulted in (in terms of the original) GBR. This resulted in the flowers being blue instead of red, and the green turned into more of a brown.

(*All images included in the report are screenshots of the original .ppm. Please see .zip file for original images)*

Original (geraniums.ppm)	Rotated (rotate.ppm)
	

For the second images, we created a mask out of the provided example powerpuff girl, and blended it with the scene depicting "The Council of Elrond" from Lord of the Rings: The Fellowship of the Ring.



Alpha Mask, all of the black was considered green screen

Projects

For this project, we started with creating a mask of the Blossom, and evaluating each pixel for green screen. Because of the shadows, we could not use simply (0, 255, 0) as a criteria. Our approach was to have a threshold, so that if the ratio of green:other was 1.3 or greater and the green value was >50, it was considered green screen and the alpha value would be 0. Otherwise, alpha was 1. There are still a couple areas where spots are misidentified as green screen, so we'd have to do a slight adjustment, perhaps blending of two pixels in a small gap between pixels.

Source Images	
	

One Puff to rule them all, One Puff to find them,
One Puff to bring them all, and in the Townsville bind them

To create the blended image, the first step was importing the 2 images and the mask successfully. Then, we scaled the mask and foreground images down to 25% the original form. This results in each dimension reduced 50%. This was the way to scale, as each pixel in the result was the average of a 4 pixel square. Once the scaling was complete, we transformed the foreground so that it was placed “on top” of the pedestal.

The extension requested is for the scaling function. The current implementation is quite basic, and not very robust. It can only scale to 25%. In future refactorings, the intent is to create a

function that can scale based on an input by the user, which will calculate the ratio of output to input.

Reflection

I came across 2 main challenges in this project, both dealing with pixel index math. The first challenge stemmed from having 2 different dimensioned images. However, using the rows * columns + index helped create a “simulated” 2D array for the images. This did not help doing the scaling though. I think overall, it seems like it would be easier to do the scale using 2D arrays so that we could use some combination of ceiling and floor to determine the index of the pixel we need data from as well as the percentage of “extra” to determine the weight of each pixel.

Acknowledgements

References for this project came from the textbook, canvas videos, and lecture notes on the home page.