

# Assignment 1 by Jesse Nayak

jdn4ae

Developed on Mac OS X 10.10.5 with Sublime Text 2, using g++ compiler.

1 day late

## Introduction

This project was completed for [CS 4810: Introduction to Computer Graphics](#) at the [Univeristy of Virginia](#). For this assignment, I created a C++ command line application to perform several image processing tasks.

## Feature List:

- [AddRandomNoise](#)
- [Brighten](#)
- [Luminance](#)
- [Contrast](#)
- [Saturate](#)
- [Crop](#)
- [Quantize](#)
- [RandomDither](#)
- [OrderedDither2x2](#)
- [FloydSteinbergDither](#)
- [Blur3x3](#)
- [EdgeDetect3x3](#)
- [NearestSample](#)
- [BilinearSample](#)
- [GaussianSample](#)
- [ScaleNearest](#)
- [ScaleBilinear](#)
- [ScaleGaussian](#)
- [RotateNearest](#)
- [RotateBilinear](#)
- [RotateGaussian](#)
- [SetAlpha](#)
- [BeierNeelyMorphing](#)

## Features:

### AddRandomNoise

This tool adds a random amount of grey-scale noise to each pixel in the image. This means that, for each pixel, the same amount is added the r,g, and b components. The parameter determines the range of noise  $[-p*255, p*255]$  added to each pixel and is a value from 0 to 1.

Parameters: 0.5



Original image

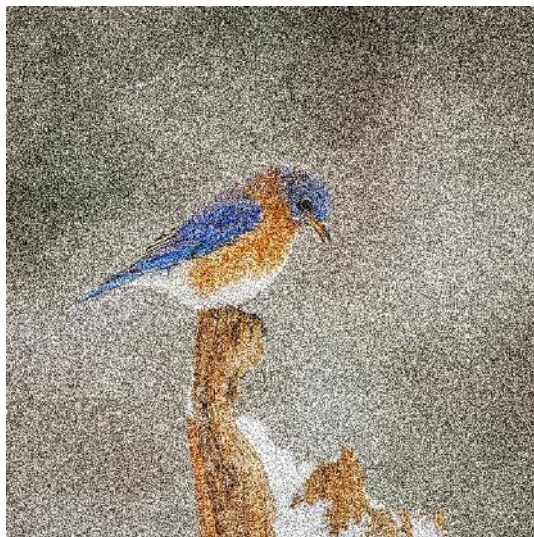


Image with random noise

### Brighten

This tool brightens or darkens the image by scaling the r,g, and b components of each pixel by the amount specified by the parameter.

Parameters: 1.5



Original image



Brightened image

### Grey

This tool converts an image to greyscale. An image appears grey if the r,g, and b values are all equal for each pixel. Thus, a greyscale image can be specified using just one component or channel. This channel is the luminance of the pixel.

Parameters: N/A



Original image



Grey image

### Contrast

This tool increases or decreases the contrast of the image by scaling the deviation of the luminance of each pixel from the average luminance the image.

Parameters: 2.0



Original image



Image with increased contrast

### Saturate

This tool increases or decreases the saturation of the image by scaling the deviation of the r,g, and b components from the luminance of each pixel.



Parameters: 100



Original image

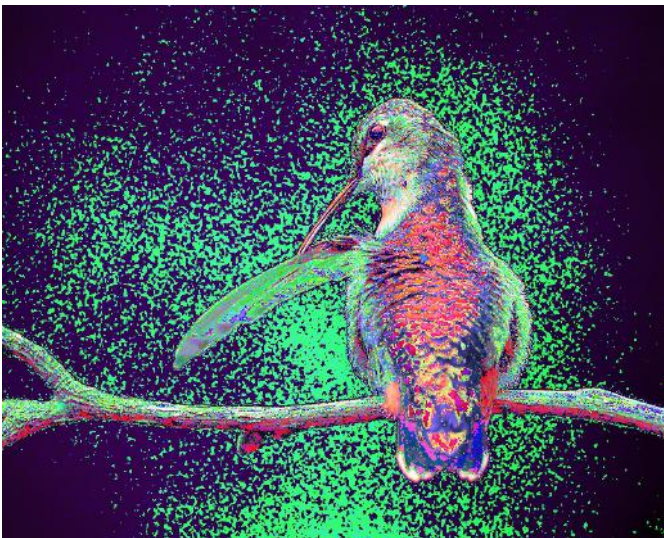


Image with increase saturation

## Crop

This tool writes the rectangular portion of the image specified by the parameters to a new file.

Parameters: 200 100 649 600



Original image



Cropped image

## Quantize

This tool reduces the number of bits the r,g, and b components of each pixel is represented with. This causes the color gamut to decrease, as fewer colors can be represented. A typical image uses 8 bits per r,g and b component.

Parameters: 3



Original image



Quantized image

### RandomDither

This tool adds random noise to the image, then quantizes it to the number of bits specified by the parameter. The amount of noise added to the image is  $0.5/2^{(bits-1)}$ . The random noise is added to help disguise the quantization errors.

Parameters: 2





Original image



Image with random dither

### OrderedDither2x2

This tool quantizes the image, then applies a filter where, for each component of a pixel, if the quantization error for the component is greater than the amount specified in the filter, the component is to be rounded up to the nearest value representable by the number of bits the image was quantized to. Otherwise, the component is to be rounded down to the nearest representable number by the number of bits.

Parameters: 1



Original image



Image with ordered dither

### FloydSteinbergDither

This tool quantizes the image, then implements an algorithm developed by Robert Floyd and Louis Steinberg that diffuses the quantization error among neighboring pixels.

Parameters: 1



Original image

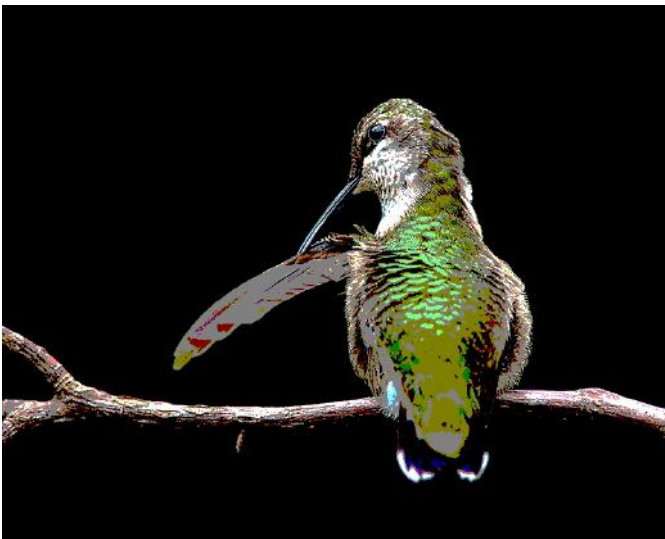


Image with Floyd Steinberg Dither

### Blur3x3

This tool uses a filter to blur the image. Each pixel is a weighted average of itself and its neighboring pixels.

Parameters: none



Original image



Blurred image

### EdgeDetect3x3

This tool detects edges in an image by using a filter that causes a pixel to darken if its neighboring pixels are similar and brighten if the neighboring pixels are very different.

Parameters: none



Original image



Image with edge detection

### NearestSample

This tool takes in floating point x and y coordinates and will return the nearest pixel in the image to those coordinates.



**BilinearSample**

This tool takes in floating point x and y coordinates and will return a pixel whose components are a weighted average of the neighboring pixels based on their distance from the input coordinates.

**GaussianSample**

This tool take in floating point x and y coordinates and will return a pixel whose components are a weighted average of the neighboring pixels within the radius based on the gaussian function (a 3-Dimensional function with normal distribution). This tool is not functioning properly.

**ScaleNearest**

This tool scales an image up or down using nearest neighbor sampling.

Parameters: .5 then 2



Original image



Scaled down image using nearest neighbor sampling



Scaled up image using nearest neighbor sampling

### ScaleBilinear

This tool scales an image up or down using bilinear sampling.

Parameters: .5 then 2



Original image



Scaled down image using bilinear sampling



Scaled up image using bilinear sampling

### ScaleGaussian

Parameters:





Original image



Scaled down image using Gaussian sampling



Scaled up image using Gaussian sampling

### **RotateNearest**

This tool rotates an image about its center pixel using nearest neighbor sampling.

Parameters: 30



Original image



Image rotated 30 degrees with nearest neighbor sampling

### **RotateBilinear**

This tool rotates an image about its center pixel using bilinear sampling.

Parameters: 30



Original image



Image rotated 30 degrees with bilinear sampling

### **RotateGaussian**

Gaussian sample not functioning properly.

Parameters:





Original image



Image rotated 30 degrees with gaussian sampling

## Composite

This tool is used to blend two images together using the alpha channel. Since JPEG and BMP files do not have an alpha channel, a matte must be used to represent the alpha channel.

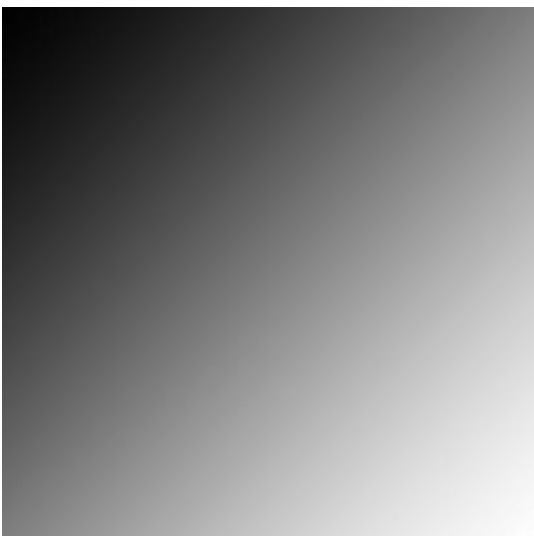
Parameters:



Original image 1



Original image 2



Matte used



Composed image

## FunFilter

This tool causes objects near the center of the image to buldge.

Parameters: none



Original image



Image with fun filter

### Beier-Neely Morphing

Not implemented

Parameters:



Original image 1



Original image 2



Morphed image

### Acknowledgements:

- [Kathrin Swoboda Photography:](#)

All sample images courtesy of Kathrin Swoboda Photography. Used with permission.

- [Code Skeleton:](#)

I used this code skeleton with command line interface, jpeg and bitmap libraries.