# Project 2

*The Neon Image Converter*

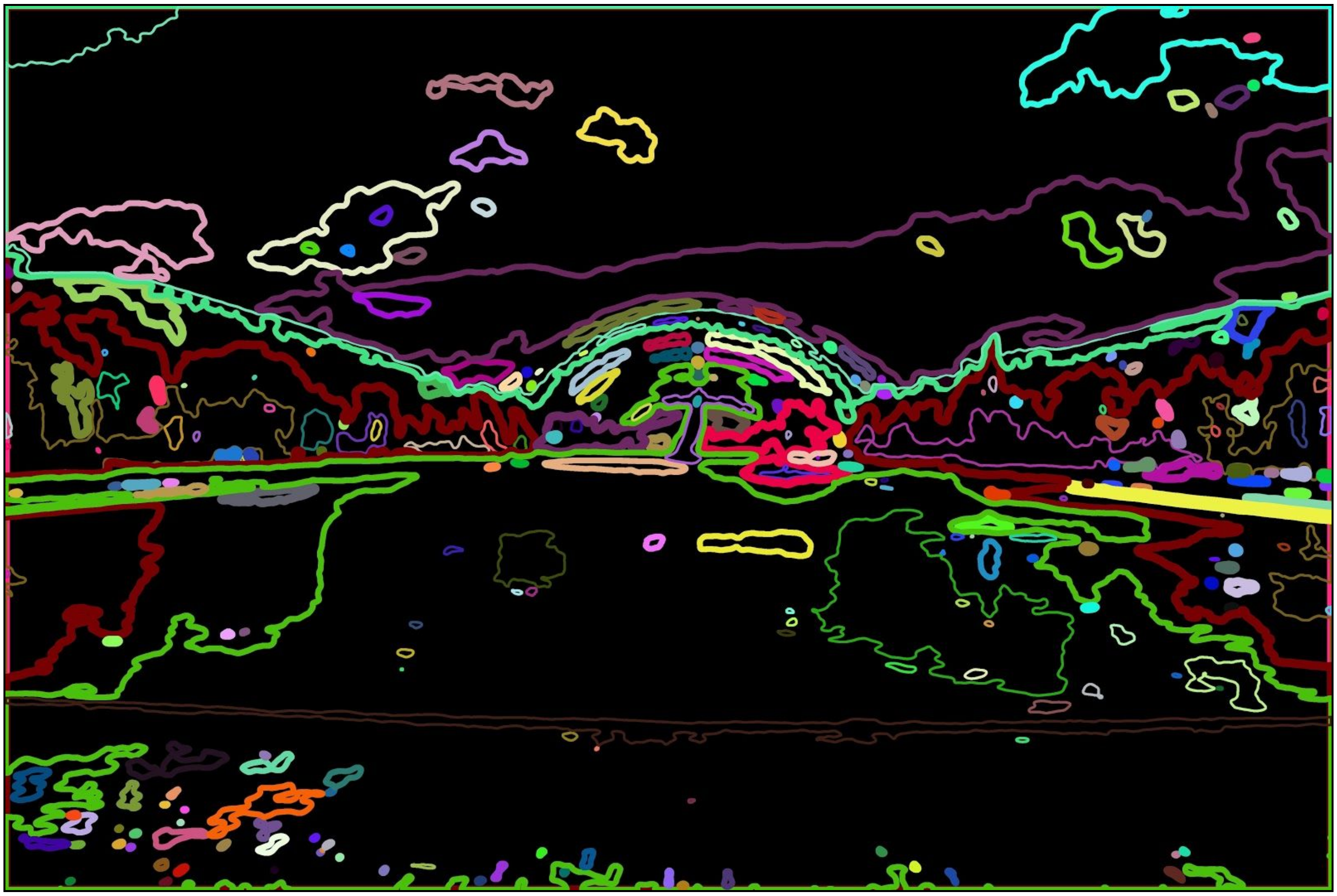
Jones, Jonathan

jonathan.jones@gatech.edu
jjones646@gmail.com

# Introduction

This program takes in an image and transforms it into an abstract neon image according to the image's contours. The resulting image will be different for 2 consecuitive runs with the program since many random numbers are used in generating the final output.

After running the program with the provided example image, you will get a result that is similar to the one below (but not exactly the same).



The source code for this program is also hosted on GitHub at the following link:
https://github.com/jjones646/mkneon

## Workflow

- Convert to grayscale
- Randomly threshold image multiple times
- Clean up thresholded image with erode & dilate
- Use OpenCV's findContours API
- Iterate over each set of contours & apply a random stroke width and a random color

## Credits

Extracting the contour lines out of an image is heavily based on snippets of code from Adrian Rosebrock's blog post that can be found here:
http://www.pyimagesearch.com/2016/04/11/finding-extreme-points-in-contours-with-opencv

# Artistic/Technical Emphasis

| | |
|---|---|
| *Artistic:* | 35% |
| *Technical:* | 15% |

While this project contains some technical details about how the output image is generated, most of the technical details in doing so is just a few OpenCV calls. The creative part of this program is how it uses the contour data to construct a neon-type image. For this reason, I base most of the credit on the artistic side.