LED Light Stick

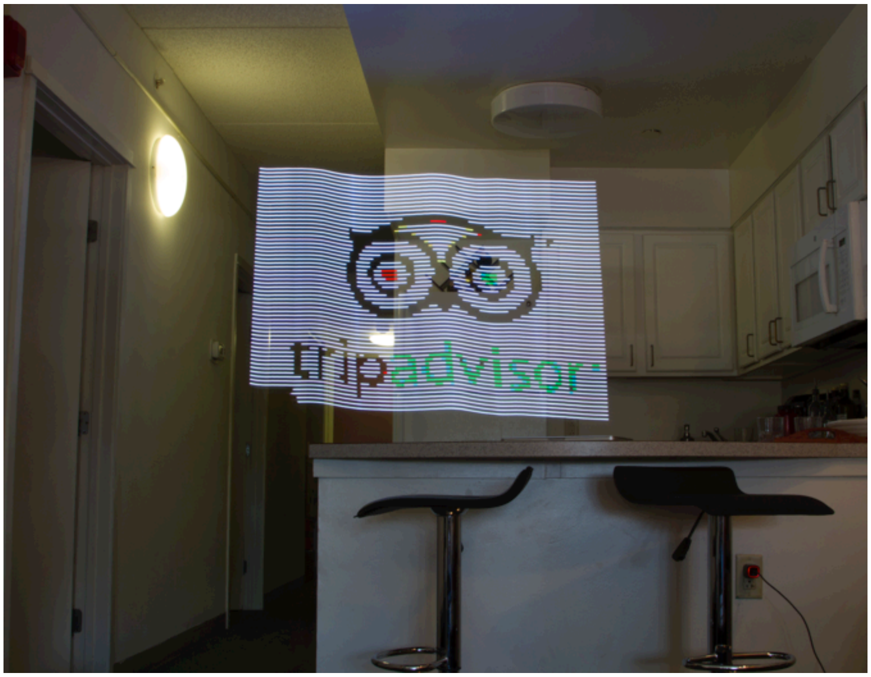
ECE 4999 Independent Study
Summer 2016

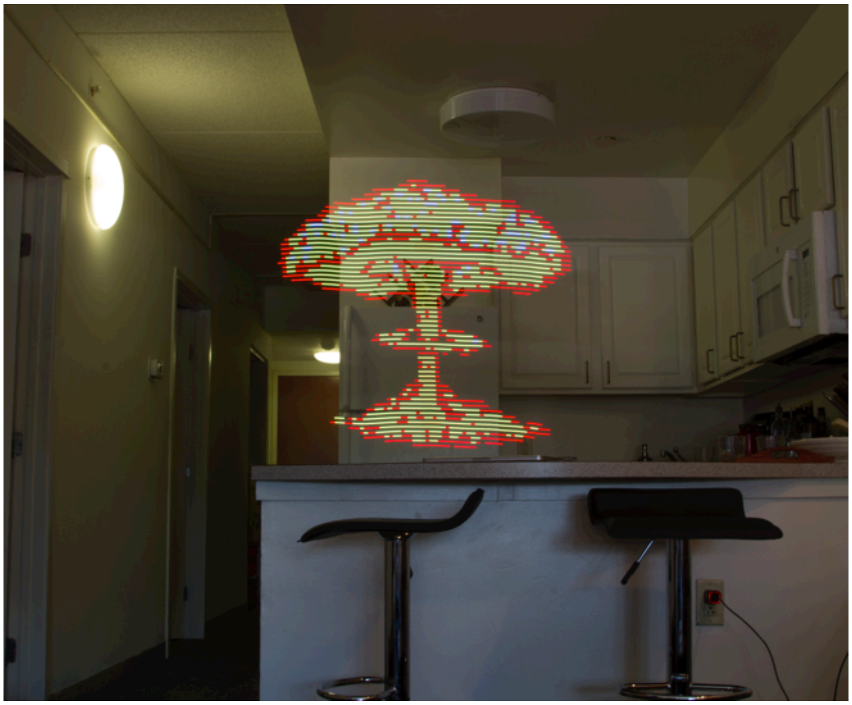
Author: Stefano Barbier
Advisor: Bruce Land

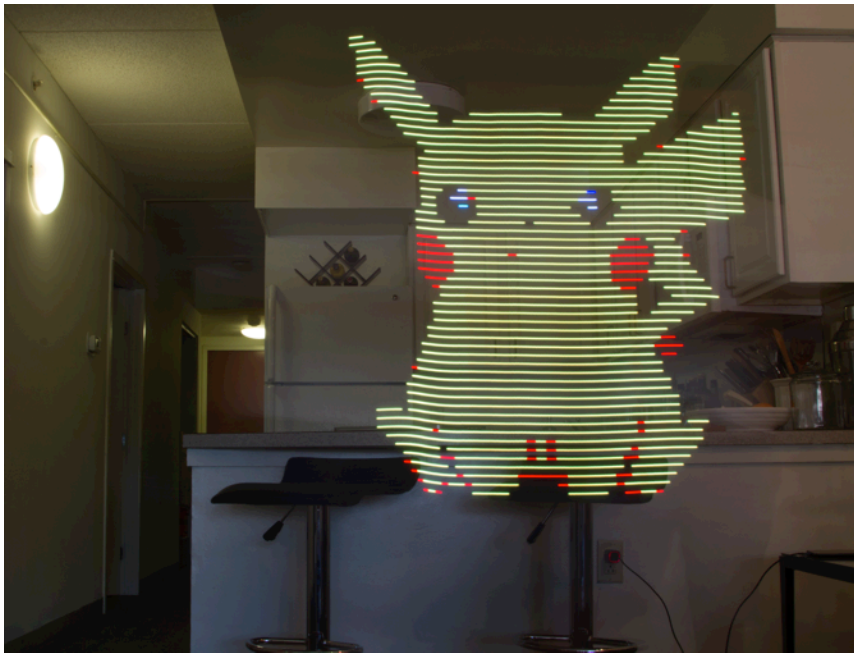
Introduction:

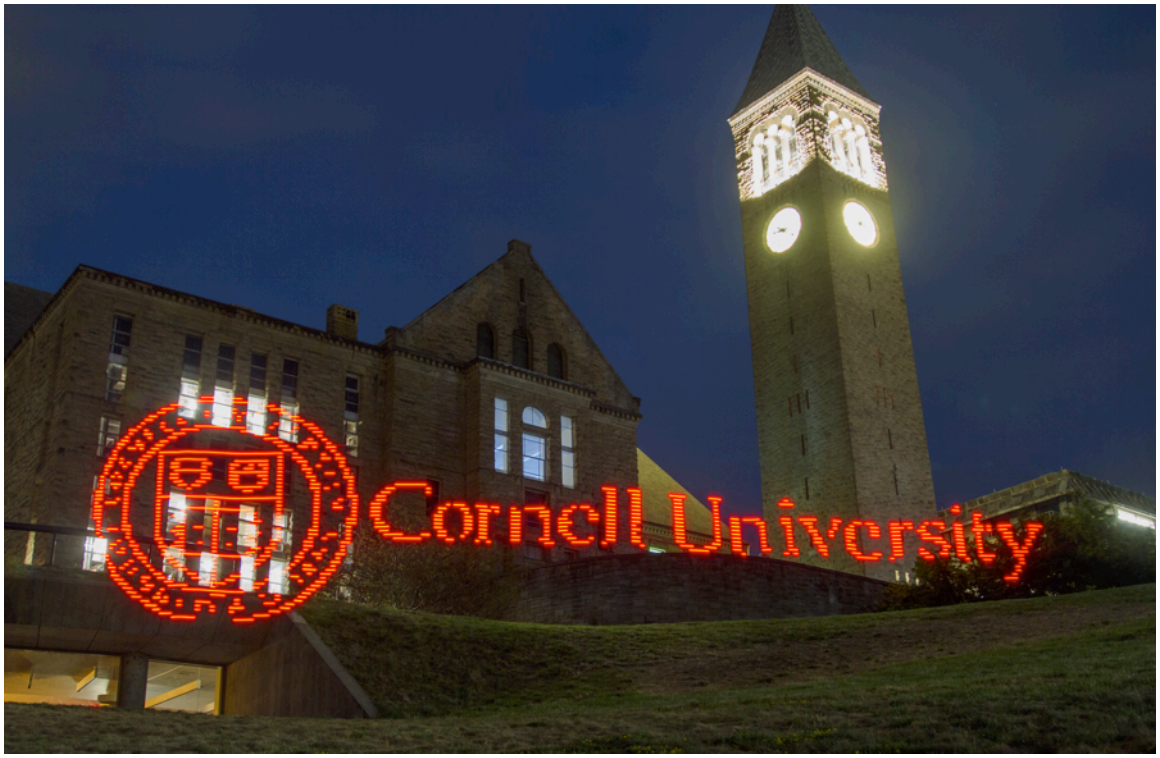
A one meter long strip of 60 neo-pixels is connected to an Arduino Uno and a portable battery to recreate images with a long exposure camera. Images can either be downloaded online or created in Adobe Photoshop. Images are compressed to 60 pixels high, while maintaining proportional horizontal dimensions, and exported as a JPEG. This JPEG is read in a Java program (written in Eclipse Neon), which exports a text file with each pixel's color as one element in a long list. Each pixel in the JPEG image is converted to its' binary RGB representation and then color mapped to the rainbow colors (ROYGBIV). The outputted list is copy and pasted into an Arduino IDE script and saved in flash memory. The script reads the list in 60 element chunks and flashes them to the LED stick. A camera on a tripod is set up with an exposure time of around ten seconds. A user walks across the frame while holding up the LED stick during the duration of the exposure.

Example Images:







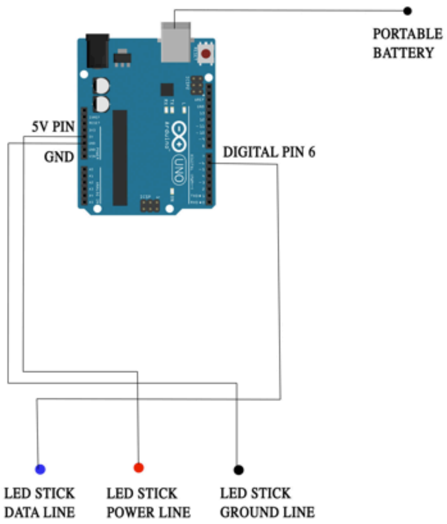


Materials:

- 1 one meter long flat wooden stick (Home Depot)
- 1 neo pixel strip (Adafruit.com)
- 1 Arduino Uno
- Electrical wires
- Electrical tape
- 2 Zip ties
- 1 portable iPhone battery

Construction:

1. Peel off the back of the LED strip and attach to the wood stick with the provided 3M adhesive.
2. Reinforce top and bottom by wrapping with a zip tie.
3. Attach the Arduino Uno to the back of the wooden strip.
4. Connect LED data line to a digital port on the Arduino Uno.
5. Connect LED ground and power to ground and 5V on the Arduino Uno.
6. Attach portable battery to wood stick.
7. Connect portable battery to the Arduino via the power jack.



How to take a Photo:

1. Set up a DSLR on a tripod and use test shots to find a 10 second exposure that is very dark but light enough to just see the foreground and background.
2. Download the Arduino script to the Arduino Uno, unplug from the computer and plug into the portable battery.
3. Set the camera on a ten second timer and press the shutter button.
4. Listen to the ten beeps from the camera and between the 8th and 9th beep press the rest button on the Arduino.
5. At the same time begin walking across the frame perpendicular to the direction the camera is pointing while holding up the LED stick and maintain a constant velocity, height, and distance from the camera.
6. In Photoshop raise the shadows and darks so that the background is more visible.

Image Processing:

Images chosen from the Internet must be somewhat simple in design so when compressed to 60 pixels high the graphic is still recognizable and also must have a uniform background color. In Adobe Photoshop Replace Color is used to change the background color to black (by default the LED stick does not turn on a pixel when it is sent black color). The image is then compressed to 60 pixels in height while maintaining proportions. The maximum width the Arduino can store is about 180 pixels. The image is then saved as a JPEG in the highest quality.

In Java the JPEG is read and each pixel is converted to its binary representation. The binary representation is then converted to an instance of the class Colors, which has three integer fields for R, G, and B. The distance formula is used to determine which color from the rainbow colors (ROGYBIV) each pixel is closest to. The R, G, and B are used as x, y, and z would be when calculating the distance between two points. The script then creates a list where each element is an integer that represents each new pixel's rainbow color. The list reads from the top right of the image, down the first left hand column to the bottom and then jumps up to the top of the second column etc… The script then exports a text file with this list.

The user copy and pastes this list into the Arduino script. This list must be stored in the flash memory (32k bytes), as the SRAM is only 2k bytes. An LED library must be downloaded from the manufacturer's website and added to the Arduino program files. This provides functions such as setPixelColor, which allows you to directly address each pixel with a known RGB value. In loop the script reads one column at a time and translates each integer back to its color RGB representation. These three integers are used to set each pixel. At the end of setting the entire column the script says strip.show(), which turns the LED strip on with all the assigned colors. At the end of the display the script delays 30 milliseconds and then returns to the list and sets the second column (pixels 60-119). At the end of the list the LED strip turns off.