Eric Hayes and Cooper Frederick

CSCI 250

Wendy Fisher

4/26/2020

Toddler Simulator Final Report

**Project Description:**

Our project is a machine learning program implemented on the Raspberry Pi to be able to recognize handwritten digits in an image. We use a camera hooked up to the Pi to take a still 920x920 resolution picture of our digit. This image is then processed, where it is turned to grayscale and compressed to 28x28 resolution. The resulting image is now black and white and the correct size for our neural net to be able to process it. However, even after this processing, the image usually still has a large amount of noise, primarily from shadows. To reduce this noise, we compare each pixel in the image to a threshold grayscale value. This value represents what level of darkness we will allow to remain in the image. If the pixel being compared is a higher value than the threshold then that means that it is a lighter gray than we want to remain in the image, so we set that pixel to white. This allows us to make our pictures basically a black digit on a white background which is much easier for the neural net to understand. This final image is then sent to the neural net which tries to determine what digit is in the picture. The output from the neural net is then printed to an LCD screen.

/To operate this project, run the program from the Raspberry Pi terminal. It will show a preview on the screen of what the camera can see and will prompt you to press a button on the breadboard. Ideally, the handwritten digit should take up much of the image. Once it detects button input, it takes a still image which is then processed by the neural net. The result of the neural net is automatically output to the screen and the LCD. The threshold value can also be set using a potentiometer on the breadboard which allows the threshold to be any value between 80-180 since the effectiveness of this image manipulation is based on the average light level in the picture. /

**Operation:**

Our project uses some non-standard python libraries including Pillow and TensorFlow [more info here].

* Run program through Raspberry Pi terminal.
* Program will show a preview of the camera and will prompt you to hit the button on the breadboard.
* Line up the camera with your handwritten digit so that the digit takes up most of the image.
* Press button when ready to take the picture.
* Wait for the program to output the results!
* Note: The threshold value used in removing shadows from the image can be set using a potentiometer on the breadboard with a range of 80-180.

**Hardware:**

* Push Button: used to tell the camera when to take the image, wired according to Jupyter Notebook
* LCD Screen: used to display output from neural net, also requires a potentiometer to be wired up, wired according to this guide [INSERT LINK]
* Potentiometer: used to set the threshold value for image manipulation, wired via ADC according to Jupyter Notebooks
* Camera: standard Raspberry Pi camera

**Data Collection and Analysis:**

* Data is collected by the camera, push button, and potentiometer.
* The camera only takes a picture once the push button is pushed.
* The potentiometer returns a value between 0-3.3. This value is manipulated to become a value between 80-180
* Camera data is processed to be black and white, 28x28, and to have reduced background noise.
* The modified camera data is then processed by the neural net [EXPLAIN FURTHER?]
* The result of the neural net is an array of values. The largest value in this array is what digit the neural net thinks is in the image.
* The output of the neural net is then printed to the LCD screen

**Code:**