**Intro to Machine Learning**

* It is meant to mimic what an organic brain does
* Consists of neurons, each of which can take on any real number between 0 to 1.0 (organic neurons are discrete, meaning it can take on either 0 or 1)
* One or more layer(s) of neurons between the input and output layers
  + Each neuron takes in inputs from all neurons in the previous layer and sends its outputs to all neurons in the next layer
  + In the case of image processing, input layer consists of every single pixel in the image
  + Output layer is the desired outcome

Example:

Objective: To label an image of a handwritten digit with the correct digit that it represents

Input: A 20x20 pixels grayscale image (i.e. input layer consists of 400 pixel values)

Output: Numbers 0-9 (i.e. output layer consists of 10 possible outcomes)

* Each neuron-to-neuron connection is assigned a weight and a bias (
  + The sum of all inputs to any neuron is then normalised (e.g. Sigmoid, tanh, etc.)
* Before training stage, the weights and biases are assigned arbitrary values
* During training, training data (in this case images) is entered as inputs and ran through the neural network
* The values in the output layer is compared with the correct output
  + In the example above, imagine the input of handwritten digit ‘3’
  + This should correspond to a value of 1 for the label ‘3’ and value of 0 for all other labels in the output layer
  + The actual answer will likely be very different – the error is computed (e.g. using RMS) and this error is known as the cost function
* The objective of the neural network is essentially a linear algebra problem of reducing the cost function
* This is done by calculating the gradient of the cost function and taking steps in the opposite direction

**What I plan to do**

1. Compile a database of images – all will be standardised in terms of size and grayscale
2. Create an output image of equal size for each input image – this output image will contain black pixels where a correct edge is and white everywhere else
3. Create the neural network
4. Train and test the model

**Major challenges**

* Acquiring enough images to train the machine (usually in the order of 104 cases)
  + Google only yielded ~100 images
* Creating an efficient way to create a correct image for each corresponding input image
  + Otherwise I will need to manually draw ~104 images

**How to move forward:**

1. Look for sources of images\*
2. Create a tool to make tracing edges easier and faster\*
3. Create a ML model (readily available from Tensorflow)

\* Can be outsourced to Amazon Mechanical Turk for finding and labelling images – will require funding