

# AI Project 3

Matthew Wheeler

May 4, 2016

## Approach

I wanted to have a program that could be sent to another location and have the ability to train the SVM from an array. Doing this would not require the user of the program to have all the images. This would make the program more lightweight and easier to use. I succeeded in doing this by creating a program that would scrape all the images in the Data folder and package them into one long array. Each image of the array is a flattened 2D bitmap. Since each image was 100x100 pixels, each element of the array has 10000 dimensions. While adding the flattened pictures to the data array, I simultaneously made a label array in order to identify the data's corresponding label. The data and label relationship is supported by the index of each array. I then uniformly shuffled both these arrays in order to keep the 1 to 1 relationship preserved. Finally I saved these arrays to an output file named data.npy and labels.npy respectfully. These 2 files are the ones that will train the SVM at runtime in order to classify new images. I created a new program called SVM.py that holds the code in order to classify images. It takes one argument: an image file. From this it verifies the file exists, creates the SVM and teaches it with the 2 previously created arrays, and then classifies the user inputted image. In order to test my program, prior to making the arrays I took 10 photos out of each folder in order to have a set of images the machine learning algorithm hasn't seen before. I then fed these images into the SVM and the results were perfect. All 50 images were classified correctly.

## Accuracy

Accuracy is pretty high. Every time I tested I got 100% with the test images I took out of the dataset so I can safely assume an accuracy of over 90%.