Algorithmic Approaches to Wastewater Categorization

UNIVERSITY of WISCONSIN UWMILWAUKEE

Nathan Kohls, Supervised by Dr. J. Rudi Strickler (Biological Sciences), previously by Dr. Peter Hinow (Mathematical Sciences)

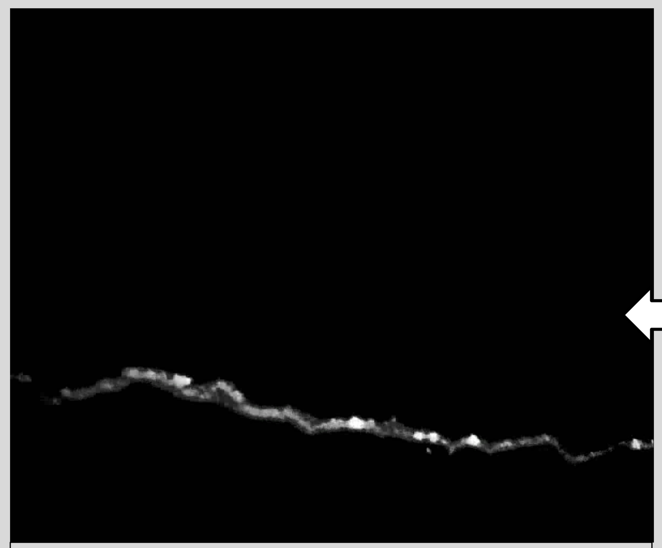
Introduction

Machine Learning and Image Recognition are powerful tools with myriad uses. One underappreciated application is the ‘simple’ task of image classification in specialized fields, like laboratory science or wastewater treatment. This project aims to elegantly automate the repetitive, time consuming process of testing the appropriate polymer levels for use on an industrial gravity belt. To do this, we shine a laser into the sludge on the belt, a process which culminates in a continuous stream of black and white images, depicting a waveform. These ‘waves’ can be sorted into two categories; the first is relatively ordered and low in the frame, while the second is both more chaotic and higher than the first. It is important for the machine operators to be able to tell where the character of the water changes from producing waves which consistently fall into our first category to those which fall into the second. When done in real time, this allows for the quick identification of causal factors for the shift.

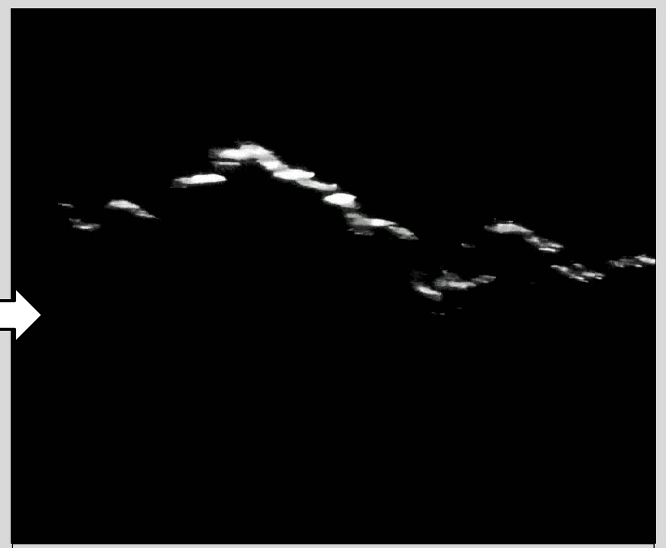
Figure 1

How would you sort the top image?

What qualities would you look for to sort it?



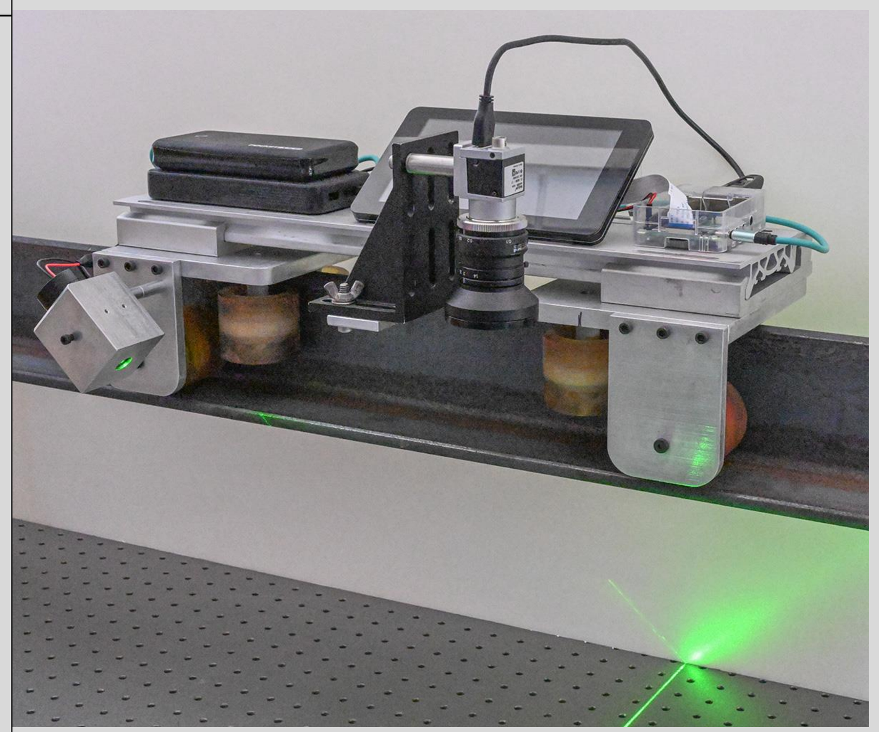
Group 1



Group 2

Figure 2

A picture of our current working classifier model. This model uses a raspberry pi 4, a Basler camera, and a horizontal green beam laser, all running off of two battery packs. The pi uses the camera to take a picture, classifies the picture, and uses the result to communicate with a controller.



Methods

To create this project, I used python 3.7 as the base programming language and relied on multiple open source python libraries such as TensorFlow and NumPy. My approach was to do all network training and testing was done in a jupyter notebook using a conda environment to make repeating and distributing the work as painless as possible. Training consists of showing a network a pre-classified image, seeing its classification, and then forcing the network to adjust its internal variables to more closely match the true classification of the image. The precise variables to change and the degree to which they should be changed is determined using methods from multivariable calculus. I trained multiple neural networks with different architectures, varying not only the number and size of the network layers but also the patterns of connections between these layers. These characteristics determine how the network behaves, and the options allowed me to chose the best one for this project, which I then optimized for use on a small computer. The end result is such that the computer running the program can, in real time, determine whether individual frames of continuous video are an example of the first group or the second, and use that information to determine when the character of the water has completely shifted. Due to the nature of machine learning and the variance in the output, I had to create the program so that it not only sorts the frames but also establishes criteria for when the character truly changes, while resisting the inevitable occasional false positive. My approach to this so far has been to use the average value of the last 10 captures.

Results

I found that a network with two convolutional layers works best. By the time this project is finished, we will have a model for use in the Jones Island Water Treatment Center. A sample of preliminary classification results are shown in Figure 3, and my proof of concept model and machine has shown positive results. With how the project has progressed up to this point, the final product will likely be ready for use by the middle of summer. I already have a classifier rigged up, as well as a controller for the classifier to talk to. What now remains is more data gathering and using that data to train a new network. This can be done as soon as the stay at home order is lifted. After that, all that’s left to do is real world testing!

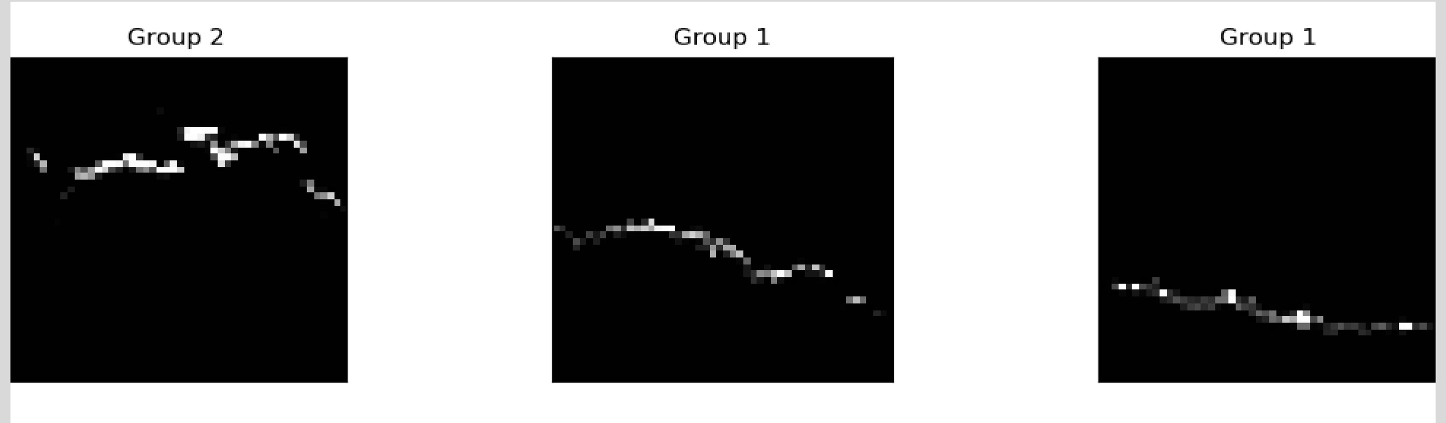


Figure 3

A set of sample classifications made by a prototype of the program and approved as correct by human classification.

This QR code will bring you to a YouTube video of a data gathering session using an old camera. (https://www.youtube.com/watch?v=c_-XlOIYw90)

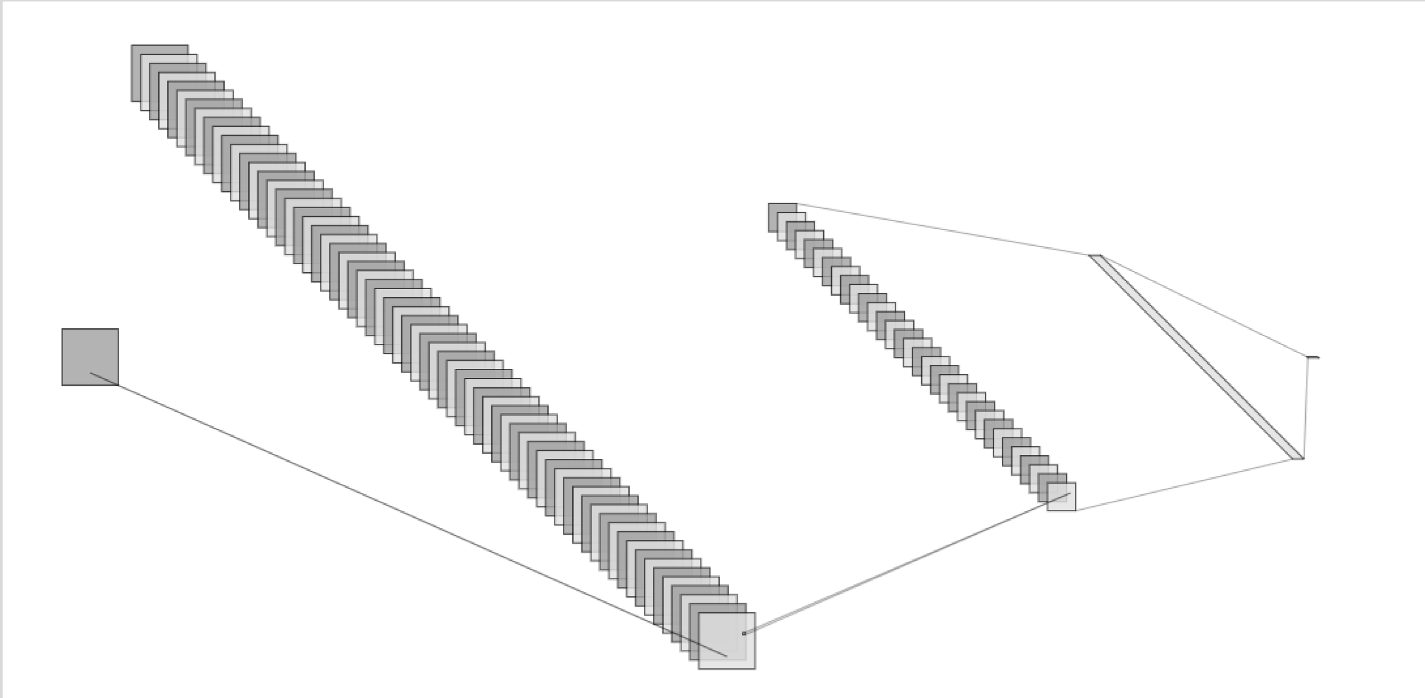


Figure 4

A diagram of the my current convolutional neural network made using the online tool at http://alexlenail.me/NN-SVG/LeNet.html

python powered TensorFlow