Rowan Lavelle

Final project proposal

For my first project idea I would like to implement handwriting to text detection using image segmentation and a convolutional neural network. For this project, I will narrow the scope of what is considered writing in text. I will stray away from taking images like stop signs, and pictures "out in the wild" that have writing in them, and turning those into text. I would like to stick to strictly pencil on paper writing, where its only blocks of writing as inputs. I will also be staying away from cursive writing, It creates a whole new problem because the letters are connected, and majorly slanted. From a top down point of view I will be following along with a research paper by Manoj Sonkusare and Narendra Sahu, which can be found at the following link. The basic idea is to start at line segmentation, either using a CNN or by using image segmentation techniques. This will separate the text line by line (if there is multiple lines) and allow us to process the text line by line. Steps to do this can be found at the following link from medium. After we have the text separate by line, we can then continue on to separate the lines into words, this part is word segmentation. There are a few ways to go about this, We can either again use a CNN, or use image segmentation techniques that assume similar distances between words and characters. This way we can look for the blanks and divide the line up by word. These steps for line segmentation and word segmentation are also discussed in a paper by Stanford which can be found at the following link. After this comes segmenting words into their individual characters, we can do this part either once again using image segmentation to detect gaps between characters in words, or we can use connected components to select all the pixel regions that are connected, which in this case will be the individual characters themselves. From here we have broken it down to the point where we can classify characters individually using a CNN. From reading the papers, it seems like using feature extraction, and a CNN is the best way to classify the characters. I have found the MNIST data set to do this with, which can be found at the following link. I would like to try and tackle this project In python, and use tensorflow as the 3rd party library to help create a CNN that classifies the individual characters. I plan on training the network on silo, which will free up computing power on my personal computer and allow me to leave the model training overnight. So as an overview, We preform line segmentation on a paragraph, then word segmentation on the individual lines, and finally feeding the segmented characters through a CNN to classify the letter and build the words and lines back up.

The second project idea that I had would be to implement JPEG compression from scratch by following along with a research paper that can be found at the following link. The first step is to compute the luminance and the two chrominance's. Next I would implement applying a discrete cosine transformation which outputs 8x8 blocks over the hole image. Next we quantize thee image which wipes out the less important DCT coefficients. This creates loss in the image, but is not noticeable to the human eye. Next we reduce the 0,0 value of each of the blocks by replacing it with the amount it differed from the corresponding element in the previous block. Finally we linearize the elements in a zig zag pattern, and see Huffman encoding to code the image. The reverse of this process is pretty easy also, we just take each of the steps and reverse them. We un encode the Huffman tree, reverse the linearization zig zagging, apply the reverse discrete cosine transformation and finally apply the inverse equations of luminance and chrominance.

Im keeping this as a backup project, and would really like a shot at implementing the first project that I described.