Hardhat Object Detection Using Convolutional Neural Networks

Jordan O'Donovan
School of Computing
National College of Ireland
Dublin, Ireland
x19372016@student.ncirl.ie

The Problem

For my project I will be utilising object detection through Convolutional Neural Networks (CNN) to detect whether one's head in an image is just their head or if it has a hard hat placed atop of it along with the accuracy of the axis-aligned box containing the object**.** Some of the images should be less complicated than others, as they'll be a single person standing close to the camera in a well-lit area, with the hard hat providing a juxtaposition to the surrounding environment. For the other images however, they may have multiple people, only some of whom will be wearing hardhats, in dark environments.

Ideas/Aims

I propose to create a CNN model which will detect both heads and hardhats, as well as their locations, with the highest accuracy achievable. Given how difficult some of the images will predictably be, if this aim proves to be too difficult the project could theoretically be scaled down to just detect if there's a hard hat in the image at all or if the image contains hard hats and/or heads without measuring the accuracy of their location.

Implementation

For my cleaning and preprocessing, I will utilise Pandas for the CSV files as well as Seaborn and Matplotlib for any visualisations. I will use Tensorflow to create my model, most likely inside of a Jupyter Notebook, which is also where I'll do any necessary cleaning and processing on the data. In my CNN, I expect to use multiple layers with a Rectified Linear Unit (ReLU) as my activation function to prevent any vanishing gradient issues I may face. I will also use pooling layers (max and/or average but probably max) in my model to reduce the spatial size of the convolved features thus decreasing any computational power needed when processing the data. Pooling will also extract dominant features which can be utilised with a fully connected layer to efficiently learn any non-linear functions present. Towards the end of my model, I'll flatten a given image into a column vector which will be fed into a feed-forward neural network and will have back-propagation applied before a SoftMax activation function is applied to classify each image for my output.

Dataset

The dataset I've chosen for this project contains 7,041 images of a person/people who may or may not be wearing hard hats. These images are all of the JPEG format, and their dimensions range from as low as 167 x 154 to 640 x 959. This dataset was created in 2019 by Liangbin Xie at the Northeastern University located in Shenyang, China and can now be found on Harvard's "Dataverse" website [1]. The dataset has been pre-split into train and test sets (75-25) and has a CSV file for each group containing each image's filename, along with the width and height of each image. These values can be duplicated however, as each row represents one helmet/head in each image, along with the minimum and maximum X and Y coordinates for said object.

GitHub

The GitHub repository for this project can be found at https://github.com/thejodster/neural_networks_project. This is where I will upload all of the work done for this project.

References

Harvard Dataverse (2019) *Hardhat-wearing.* Available at: https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/7CBGOS [Accessed November 12 2022].