Aaron Borger, Nathan Mwiinga, Tyler Shea

Machine Learning

5 May 2020

Dragon-Vision

Abstract

The following report describes and analyzes the Dragon-Vision project, including its intent, methodology, and results. This project came about as a result of Northwest Nazarene University's campus game, "Humans vs. Zombies" (HVZ). The program's objective is to detect people who are wearing red headbands. The program uses a neural network classifier to classify images as either having regular people or "zombies" (people wearing red headbands). The input data for this program comes from a variety of pictures, some that have people in them and some that do not. Some of the input data was used to train the classifier on what to look for, then the remaining data was used for testing and verification. This report explains the details of how the program was built, as well as its results and analysis.

Introduction

Project Dragon-Vision is a machine learning project designed to detect people from an image. Image classification is a very important and popular application of machine learning, and can be used in many situations. The general motivation behind Dragon-Vision is to build a classifier that can learn to locate a person in a picture. The specific motivation comes from a game called "Humans vs. Zombies," which is played on the campus of Northwest Nazarene University. However, Dragon-Vision can also be utilized in a much broader scope, as it is a program that can learn to pick out a specific object from a picture, which is a process that has various applications and uses.

Background

As aforementioned, the driving purpose for Dragon-Vision was the “Humans vs. Zombies” game. In this game, some students are designated “zombies” while other students are “humans.” The humans must survive against the zombies by defending themselves with nerf guns and socks. The zombies are required to wear a red headband in plain view to show that they are a zombie. They attempt to tag the humans to convert them into zombies. The motive of the Dragon-Vision project is to use pictures taken from a drone, and classify whether there are zombies in the image. As a result, the program can be used to detect any zombies from a drone, and can be implemented in a targeting system used by a drone with a built in nerf gun. The classifier for this project is a special type of neural network, called a Faster Regional Convolutional Neural Network, which will be explained in great detail later on. In short, this network is able to do the arduous and time-consuming process of classifying an image in a much faster time frame than a simple neural network.

Methodology

There were many parts that needed to be accounted for in order for this project to take shape. The first step is to accumulate the right data. The data used to train the machine came from hundreds of pictures with people in them. Some of the pictures included people who had red headbands on, while other pictures were simply of normal people. The next step was to import all of the pictures into an online tool called VGG Image Annotator from Oxford

University. This tool was used to manually draw rectangular boxes around each of the people in the picture. Then, for each rectangle, the person was labeled as “person” or “zombie.” The image annotator was then able to export a .csv file that included the four pixel coordinates for the rectangles as well as the label that went along with each person. From there, the data became the input into the classifier to be used for training.

The classifier used is an FRCNN (Faster Regional Convolutional Neural Network). Specifically, the one cloned from kbardool’s keras-frcnn on github. In order to understand what the FRCNN does, it is important to understand what a neural network does. A neural network has some input nodes, some output nodes, and several layers of hidden nodes in the middle. If the neural network is being looked at like a map with input nodes on the left and output nodes on the right, the connections connect all the inputs to the farthest left layer of hidden nodes. That layer of hidden nodes is then connected to the next and so on until the output nodes are connected. Figure 1 shows an example of this.

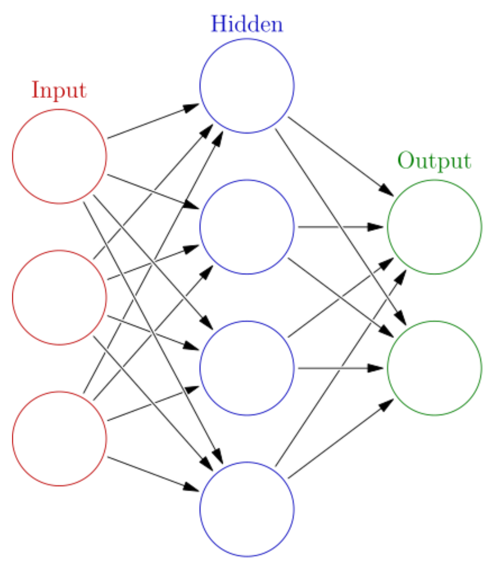

Figure 1 (https://en.wikipedia.org/wiki/Artificial_neural_network)

The input nodes provide a value. In the case of Dragon-Vision, these values would be an RGB value. The connections are given a weight, which is multiplied by the value given by the input and passed to the next node. That node sums all of the products of the connections that are being inputted into it and adds a bias to it. That sum is passed through an activation function to keep the value between a certain range. Zero and one is a common range, and a simple example of an activation function would be the sine function. When the network is being trained, it takes the input and provides an output; the output is compared to the output the network should have gotten based on that input. In this case, that is either person or zombie. The typical way of training the network uses gradient descent to calculate how wrong the output was, and then back propagation to little by little change the values of the connection weights and node biases until the correct output is received.

The convolutional part of the FRCNN means the network uses the surrounding area of a pixel and then pools all those areas together before it feeds them into the network. The regional portion of the FRCNN means the algorithm looks at areas of interest that are then fed into the network instead of the entire image. Therefore, the classifier can look for many objects in the same picture. The algorithm finds these areas of interest by looking for lots of different pixel colors in a small area. If a large area of the image is the same object there most likely is not an object there. The typical R-CNN finds 2,000 different areas of interest before feeding them into the convolutional neural network. The algorithm uses the border drawn around the objects in the annotations to more accurately find regions of interest based on our data. The faster part in the FRCNN algorithm means that it does not find 2,000 different regions as that takes a long time. Instead, the algorithm passes the image through a convolution layer to take surrounding areas into account, and then has a separate neural network to decide where to place the region proposals that are sent to the CNN.

Results

Upon analysis of the FRCNN at work, various things became apparent through the collection, training, and testing of the Dragon Vision. On the surface, the neural network operates in its intended way. It is able to recognize persons within a picture, and determine if they need be classified as a human or zombie. It was noted early on that training would take an extensive amount of time to run because of the number of epochs originally envisioned for training. To tackle this, the use of a GPU would be advisable; however, there is a slightly different formatting of the python modules that is needed to go into use. Although it was successful in running, there are a couple of ways that could improve the classification process. One such method includes training the program to look for headbands primarily as opposed to persons, and then determining if it is attached to a person.

Conclusion

The results of this project show that the Dragon Vision development has certain potential to be used in the Humans vs Zombies Game in the coming years. Moving forward with the project, there are a couple areas in which alterations could improve the performance. The primary modification would be to utilize a GPU as opposed to a CPU to train data. This would allow for more models to be trained, and would refine the classification process. Also, implementing the identification of red bands primarily is worth exploring as it may be a way to not avoid people that don’t have as much light reflected off them. Furthermore, the application of this program outside of Humans vs Zombies can extend to detecting any identifiable object. Coupled with another program to target and manipulate, this can be used in agricultural implementations, and also in the overarching Dragon Fly project to fire nerf darts at targets.