Image Processing for Learning the Rules of Video Games

Erik Culberson, Neil Ferman, and Viljo Wagner

Abstract—The abstract goes here. (DO WE NEED THIS?)

1 Introduction

There are many examples of machine learning algorithms being able to play video games when the algorithms are integrated directly with the game. This is a costly process and needs hours of development work as well as needing more access to resources such as the source code of the video game. This approach also give algorithms more in-game knowledge then a normal player would have. By removing this advantage from the algorithms we will be able to see how well the algorithms can compare to human players given an equal playing field. Given the same inputs as a human player, a screen image and a set of key controls, can a computer learn to play a video game.

2 Problem

3 Algorithm Structure

There are several approaches to solving this problem. For our purposes we will focus on two possible solutions. First solution requires a human to play the video game and capture the correct inputs to the corresponding image. In this first solution there is more of a requirement on the human to provide accurate image to key combinations. This method is more of a learn by watching scenario where a computer would watch a human play to learn the proper image to key combinations. The expectation of this approach is that given the correct inputs to image an algorithm can learn what to do in an unknown case where the algorithm has not seen a particular part of the game.

In the second solution we will look at use a genetic algorithm to teach the image processor the correct weights to use to get the corresponding correct outputs. This method is a learn by doing method. By giving the algorithm a fitness score of how well the algorithm did each generation the algorithm will keep the best scoring weights while getting rid of the less successful weights. The expectation of this approach is that the more the algorithm plays the game the better it will be. This will also allow the computer in unknown situations to keep trying until it finds the best, not necessarily the most optimal, solution.

It is worth noting that the two algorithm structure stated above are not the only solutions that would fit this problem but for the purpose of this project we will focus just on these two algorithm structures to start.

4 Algorithm Options

There are many image processing algorithms out there currently but for our problem we need an algorithm that can fit two primary criteria. First we need an algorithm that can learn fast and second the algorithm must be able to process images at the necessary speed to get all the information from the video game, usually 30-60 frames per second. There are two algorithms that are candidates for this problem. It is worth noting that just like the algorithm structure that there are plenty more solutions that would fit this problem but for the purpose of this project we will focus just on these two algorithms to start. First algorithm is called a Dynamic Routing between Capsules which is detailed in the following paper [1]. The Second algorithm is called a Mask R-CNN which is

detailed in the following paper [2]. Both of these algorithms seem to fit the criteria and both will be looked at more thoroughly as a possible algorithm solution to are problem.

5 Solution

6 Conclusion

The conclusion goes here.

References

[1] S. Sabour, N. Frosst, and G. E. Hinton, Dynamic Routing Between Capsules, Neural Information Processing Systems, 2017.

[2]