CS Capstone Design Document

December 2, 2016

AI Gaming

Prepared for

NVIDIA

Mark Ebersole _____ _____
Signature *Date*

Prepared by

Group 14

Deep Learning on Embedded Platform

Christopher Johnson _____ _____
Signature *Date*

Gabe Morey _____ _____
Signature *Date*

Luay Alshawi _____ _____
Signature *Date*

Abstract

In this document we describe the design of a deep learning system developed to learn how to play the arcade game Galaga. Galaga is an arcade shooter which was released in 1981 by Namco [1]. This system is designed with the ultimate goal of being turned into a course for the NVIDIA Deep Learning Institute. The documentation will outline exactly what hardware will used, how the system will be put together, and what methods will be used in a way that allows others to recreate this project.

CONTENTS

1 OVERVIEW

1.1 Purpose

The main aim of this document is to give others a thorough enough understanding of the project and its design to replicate the project. The reader of this document should walk away able to understand how the system works and why it was designed this way. The document must be thorough enough for NVIDIA to recreate the project and package the design into a training course on deep learning.

This project is to be used to produce a training course for NVIDIA's Deep Learning Institute. After the project is complete it will be handed off to NVIDIA for the purposes of accomplishing this goal. To that end the documentation has been designed to provide readers a clear path towards building this system themselves.

1.2 Scope

This document outlines the design of a system for teaching a deep learning neural network how to play the game Galaga. The sections of this document will outline the system details, setup, and all things necessary to understand how and why the project is designed. Hardware required, API's to be used, and setup will be detailed as part of the project design.

1.3 Context

This system is not designed for an average computer user. This system is a showcase of possibilities when working with Neural Networks. As such, no real UI will be developed for the purposes of this project. Average people may see the result of the final project by viewing the network play Galaga.

This system is designed for developers. Developers should be able to do a project like this as an introduction to deep learning neural networks. The documentation provided here reflects this focus and is written with developers in mind.

1.4 Summary

The neural network created for this project will be able to play Galaga. Its design is based on GoogLeNet, a convolutional neural network designed in coordination with Google Inc., University of North Carolina, Chapel Hill, University of Michigan, and Magic Leap Inc. [2]. This neural network will undergo two main phases of training. One training phase will be done with just still images, while the second will involve live image feed from the game Galaga. The system will run inference on the Jetson TX1 developer kit, which will interface between a seperate computer via a controller. The Jetson will be wired to the controller to control the game on the seperate computer. Cloud computing GPUs will be used to handle the training. In total, the system should be able to learn how to play Galaga, and then continually improve as it plays.

1.5 References

[1] F. Rojas, "Galaga (Namco)," in gaminghistory101.com, Gaming History 101, 2012. [Online]. Available: https://gaminghistory101.co
[2] C. Szegedy et al., "Going Deeper with Convolutions," in cs.unc.edu. [Online]. Available: https://www.cs.unc.edu/ wli-u/papers/GoogLeNet.pdf.

1.6 Definitions

Neural Network: A multilayered graph structure containing nodes, referred to as neurons, and weights that affect how likely the node is to be chosen when confronted with a decision. The weights of nodes are decided through training the network, and a traversal of the graph acts as a large chain of decisions as the neural network is confronted with situations.

Deep Learning: Using a layered neural network constructed by GPU processing to teach a computer how to accomplish a specific task.

Convolutional Neural Network: A neural network specifically designed to take images as input. It uses convolutional layers to process the image into a more manageable chunk of data and extracts the information relevant to the network.

Caffe: A code framework that allows for complex mathematical operations that are required for training a neural network.

API: Application Programming Interface

Jetson TX1: Quad core embedded system designed for power efficiency and deep learning projects

OS: Operating system

OpenCV: (Open Source Computer Vision) is a library that can be used with different languages(C, C++, Java, Python, etc.). It provides standard functionalities such as image capture, Faces recognition, Gesture recognition, Motion tracking, Mobile robotics, Object identification and Image manipulation.

Galaga: Galaga is an arcade video game released in 1981 by Namco [1]. The player takes control of a small space ship and must shoot at and dodge waves of alien space ships. Victory in a level is achieved by destroying all enemy ships.

2 DESIGN

2.1 Introduction

This section lays out several viewpoints from which we have approached the design of this project. Each component will be matched under an appropriate viewpoint that gives the reader a clear idea of how the design was apparoached. Each subsection details a different set of components, many of which will be lumped together, which constitute a major piece of the final design. There are three main viewpoints from which we have approached the design:

- Structure Viewpoint: Design elements approached from a structural level.
- Information Viewpoint: Design elements approached from a perspective of data flow operations and data processing.
- Interaction Viewpoint: Design elements relevant to getting systems to integrate with each other. Particularly focused on hardware setups required for the rest of the process.

Each viewpoint is covered in more depth in its respective section.

2.2 Structure viewpoint

This section covers integral structural elements of the system and how they come together. Tools used will be covered to give readers an idea of how we set up the system structure. The heart of this structure is our neural network set-up.

2.2.1 Neural network setup

The neural network set-up itself is a multi-layered combination of components. At the basic level of design, our neural network follows the architecture known as GoogLeNet. GoogLeNet is designed, as a convolutional neural network, specifically to process images and categorize them. At the architectural level it has 22 layers, each consisting of special modules [2]. These modules have a structure similar to that outlined in figure 1.

The network set-up provided by GoogLeNet needs to be accessed with a framework, as well as modified slightly for this project's needs. For the framework this project uses Caffe. Caffe allows for command line instructions to be sent to define the network. To set up the network and Caffe properly we are using a program called DIGITS. DIGITS is a neural network set-up application designed and released by NVIDIA. It gives an easy interface for setting up the exact parameters for learning as well as building the network up. The exact parameters we need for the set-up of this net are unknown at this time and will be added when available.

Training of the neural network will be done on NVIDIA GPUs on the Amazon Web Services cloud. The network will