Requirements Document for HIVENet

Enrique Ferndandez, William Wodrich, Andrew Davis

CS 461 Oregon State University

Fall 2018

Abstract

Facial recognition employed by neural networks requires large amounts of computational power and large data sets to accurately identify images and the subjects therein. These restrictions drastically affect the ability to utilize neural networks on many small-scale devices. An approach to fix this is a decentralized network that allows multiple devices to train, and share data.

However, in the cases where data sets are being collected on multiple distributed edge devices, it is both cumbersome and time consuming to transmit all the relevant data over a network. This project aims to optimize machine power, of data by using a decentralized architecture that will manage the various distributed devices. Each edge device will train an artificial neural network from its subset of data, to reduce the size of the data set transmitted only the coefficients of the Artificial Neural Network are given to the remaining edge devices. The edge devices will integrate the transmitted coefficients of the localized ANN into the ANN of the core device, this will make the core ANN engine the most effective aggregate of the localized devices. This decentralized system is aimed to perform continuous recognition while registering and training on new users.

CONTENTS

1 UPDATES FROM PREVIOUS DOCUMENT

The following table outlines changes in made in this document since its initial creation.

Section	Original	New
• 2.1 • 2.2 • 3.2 • 3.3	• Does not reference GUI for user interaction • Mentions client-server model • Some functional requirements not outlined • Some performance requirements not outlined	• Added reference to GUI • Changed model to ad-hoc/peer-to-peer • Added functional requirements: 2, 4, 8 • Added performance requirements: 4, 5, 6

2 INTRODUCTION

This section gives an overview of the content of this software requirements specification (SRS) document. The purpose and scope of the document are explained, as well as the definition of some relevant terms and abbreviations.

2.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the Hierarchical Information Variant Exchanging Network (HIVENet). This document will explain the purpose and development of the system as well as provide a high-level understanding of HIVENet's infrastructure and restraints.

2.2 Scope

The HIVENet facial recognition software application is designed to demonstrate a method to optimize neural network training and illustrate methods to enable communication between neural networks. The tool will be built to interface with Google's Deep Learning application, TensorFlow™(pending).

2.3 Glossary

- **HIVENet:** "Hierarchical Information Variant Exchanging Network." Used throughout the document to refer to the project application.
- **Edge-Device:** A computer connected Via LAN to the router. Each computer is equipped with a camera to receive video input.
- **Edge-Process:** The tandem structure of the artificial neural network and inference engine housed on the edge devices.
- **Core-Device:** A central computer wirelessly connected to all edge-devices.
- **Core-Process:** The HIVENet application manager, housed on the each edge device. It periodically receives updates from edge-processes and handles the variant exchange process.

- **Packet:** A unit of data packaged to be transmitted over a network. Edge processes will send packets of their NN coefficients to the Edge-Devices through a wired network. Each Edge-Device will on network will send instructions of how an edge-process should modify its coefficients to match the new information variant.
- **Neural Network:** "NN." A computational framework that employs several learning algorithms to process complex data input. This framework is generally implemented as directed weighted graphs of artificial neurons.
- **Artificial Neurons:** The most basic component of a neural network. They mirror neurons in the brain, acting as aids in the overall decision process of NNs [1].
- **Coefficients/Weights:** The values inside of a NN that define and modify its behavior. They are associated with the artificial neurons and edges that connect them.
- **Training:** The act of providing a NN with a dataset to operate on and apply its learning algorithms to.
- **Inference Engine:** A component of the edge-process that applies logical functions to a knowledge base to deduce information. In our implementation the knowledge base is provided by the weights of the neural network.
- **Fitness Level:** An enumeration of an edge-process' competency. As the NNs train themselves on a dataset, it will modify itself to increase accuracy, raising the fitness level.
- **Dataset:** A collection of information for a neural network to process,labeled such that the NN can grade its performance. For this project datasets will be comprised of images of individuals, labeled with the measurements of their facial features.
- **Upstream:** An established communication protocol from a Edge-Device to the Core-Device.
- **Downstream:** An established communication protocol from the Core-Node to the Edge-Node.

2.4 Overview

The following sections will provide an overview of HIVENet's infrastructure and system functionality. Section two will cover user interaction, as well as the assumptions and constraints of the application. Sections three and four will provides the required specifications of the application, detailed in specific benchmarks the application and its components must meet.

3 OVERALL DESCRIPTION

This section will give a high-level HIVENet software infrastructure and front-end usability. This software will be explained in a matter of how the HIVENet infrastructure will be setup, how users will interact with HIVENet, and what data will be passed. By the end, assumptions and constraints for the system will be presented.

3.1 Product Perspective

This application will consist primarily of underlying code of hierarchical NN stack that will pass coefficients, and alter edge-devices inference engines that will allow for those same Edge-Devices to be able to identify the users that are using this product across the entire network of Edge-Devices. The user will interact with an authentication GUI web page.

3.2 Product Functions

HIVENet is designed to register and recognize any user who purposely interacts with this application. It uses a hierarchical, convolutional NN that self-trains its edge-devices. Each Edge-Device will have a peer-to-peer communication channel to every other Edge-Device. This peer-to-peer channel will be the primary communication to share NN coefficients to all other Edge-Devices.

Each Edge device is responsible to reconfigure their own inference engine edge-devices to further optimize the success rate of the facial recognition software.

3.3 User Characteristics

With HIVENet, the application manager can register physical facial characteristics as a form of subject identification. The subject will then be identifiable via the Edge-Devices.

4 SPECIFIC REQUIREMENTS

4.1 Interfaces

4.1.1 User Interfaces

This will provide visual feedback to the user so as to demonstrate the success of the facial recognition across multiple devices. Doing this will require webcam functionality, and an interface to get user input upon initial entry into the system (entering their name).

4.2 Functional Requirements

This section includes the requirements that specify the fundamental actions of the software system, and the necessary outcomes that are expected by the client.

ID: FR1
Title: GUI Registration platform
Description: HIVEnet shall have an interactive GUI that allows users to register themselves into HIVEnet for identification on all Edge-Devices.

ID: FR2
Title: GUI Identification platform
Description: HIVEnet shall have a GUI that displays users identified, as well of all the users previously identified.

ID: FR3
Title: Synchronized Edge-Device Inference Engines
Description: Each Edge-Device should have the most up-to-date Inference Engine and assembled by the weights of all Edge-Device Artificial Neurons in the HIVEnet network.

ID: FR4
Title: Edge-Device ID's
Description: Each Edge-Devices should be uniquely identified using Unique Identification Numbers.

ID: FR5
Title:Re-configurable Neural Network on Edge-Device
Description: Edge-Devices should be able to incorporate new weights from other Edge-devices into each Edge-Device's Inference Engine.
Depends on: FR8

ID: FR6
Title: Edge-Device Fitness Level
Description: Edge-Devices should hold a qualifying fitness level of 85 percent when identifying a person
Depends on: FR7, FR8

ID: FR7

Title: Self-training Edge-Device

Description: Each Edge-Device should be able to self-train its own neural network.

ID: FR8

Title: Upstream and Downstream communication

Description:all Edge-Devices should have two-way communication with all other Edge-Devices in the HIVEnet network.

4.3 Performance Requirements

ID: PR1

Title: User Identification

Description: Every updated neural inference engine on a localized device will be able to correctly identify A registered user with 80% accuracy.

ID: PR2

Title: Small packet size sent from localized device to update other devices.

Description: When the localized device finishes training on a data set of size A, the size of the packet to be sent to All other Edge-Devices should be at most size A/5.

ID: PR3

Title: Small packet size sent from localized device to update other devices.

Description: When the localized device finishes training on a data set of size A, the size of the packet to be sent to the core device to update the core NN should be at most size A/5.

ID: PR4

Title: Inference Engine recognition.

Description: The Inference Engine shall be used for facial recognition, except in the cases where a new user is being registered. Otherwise, the recognition portion will be triggered by user input.

ID: PR5

Title: Downstream Packet transfer rate.

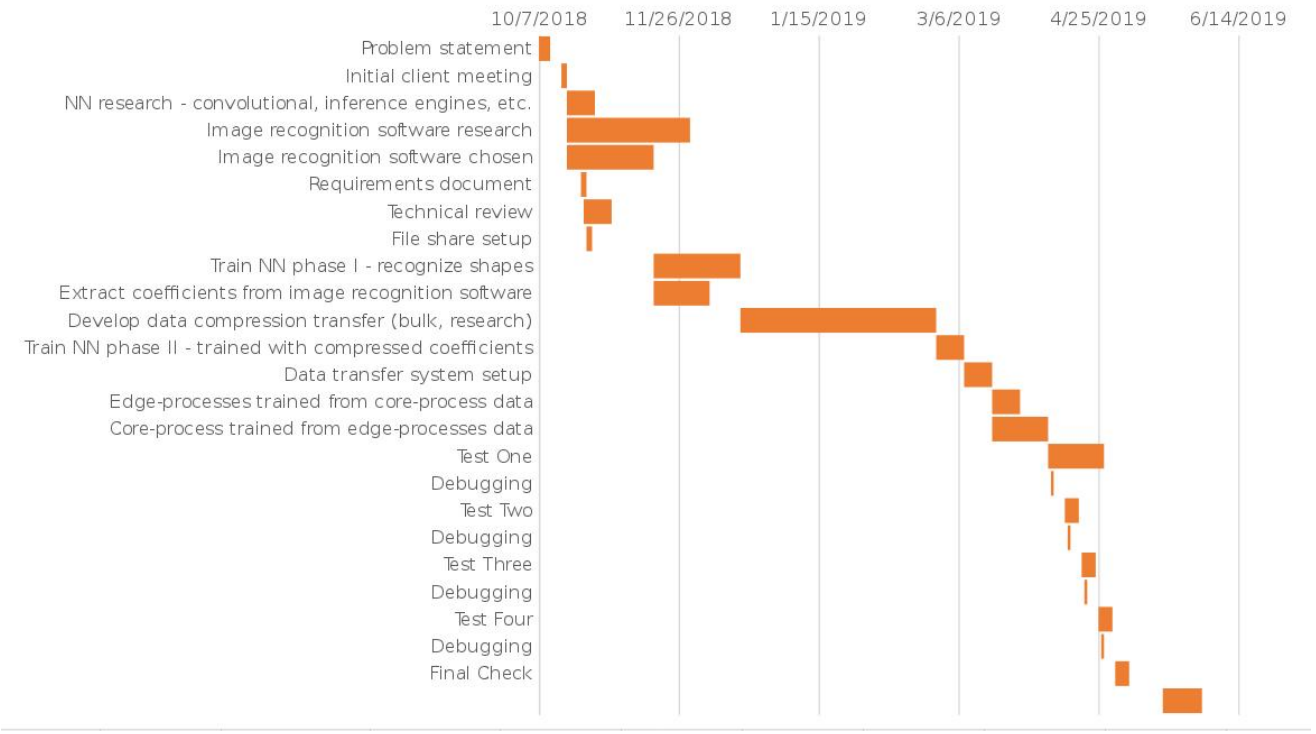
Description: 95% of packets transmitted between all Edge-Devices shall be transmitted in less than 1 second.

ID: PR6

Title: Downstream Packet transfer rate.

Description: 95% of packets transmitted between the core device and an edge device should be transmitted in less than 1 second.

5 GANTT CHART



[1] Eyal Reingold, "Artificial Neural Networks Technology", *University of Toronto Mississauga*, [Online]. Available: *http://www.psych.utoronto.ca/users/reingold/courses/ai/cache/neural_ToC.html*. [Accessed October 31 2018]

[2] Marcel van Gerven, "Artificial Neural Networks as models of Neural Information Processing", *Frontiers in Computational Neuroscience*, [Online]. Available: *https://www.frontiersin.org/research-topics/4817/artificial-neural-networks-as-models-of-neural-information-processing#authors*. [Accessed October 31 2018]

[3] "Definition - Data Packet", *techopedia*, [Online]. Available: *https://www.techopedia.com/definition/6751/data-packet*. [Accessed November 1 2018]

[4] "What is a Neural Network", *DeepAI.org*, [Online]. Available: *https://deepai.org/machine-learning-glossary-and-terms/neural-network* [Accessed November 1 2018]