

UNDERGRADUATE PROJECT PROPOSAL

|  |  |
| --- | --- |
| **Project Title:** | **A Convolutional Neural network Approach for the Recognition of Traffic Signs** |
| **Surname:** | **Wu** |
| **First Name:** | **Yile** |
| **Student Number:** | **201918020102** |
| **Supervisor Name:** | **Happy Monday** |
| **Module Code:** | **CHC 6096** |
| **Module Name:** | **Project** |
| **Date Submitted:** |  |

**Table of Contents**

[1 Introduction 3](#_Toc117467119)

[1.1 Background (overview of topic and motivation) 3](#_Toc117467120)

[1.2 Aim 3](#_Toc117467121)

[1.3 Objectives 3](#_Toc117467122)

[1.4 Product Overview 3](#_Toc117467123)

[1.4.1 Scope (What will it do? How will it work?) 3](#_Toc117467124)

[1.4.2 Audience (Who is it for?) 4](#_Toc117467125)

[2 Background Review 4](#_Toc117467126)

[2.1 Summary of existing approaches (e.g., Competitive analysis, if appropriate) 4](#_Toc117467127)

[2.2 Brief summary of related literature (e.g., Annotated bibliography or initial literature review, with a brief summary of sources) 5](#_Toc117467128)

[3 Methodology 6](#_Toc117467129)

[3.1 Approach (Description of the research and development methodology, e.g., Software development model, requirement gathering method, test, and evaluation process) 6](#_Toc117467130)

[3.2 Technology (Implementation tools & resources such as hardware and software) 6](#_Toc117467131)

[3.3 Version management plan (e.g., Git repository or shared drive) 6](#_Toc117467132)

[4 Project Management 6](#_Toc117467133)

[4.1 Activities: tasks required to complete each objective 6](#_Toc117467134)

[4.2 Schedule, i.e., Gantt or other, showing activities, deadlines 8](#_Toc117467135)

[4.3 Data management plan (e.g., Google folder for project logs, reports, literature, etc.) 9](#_Toc117467136)

[4.4 Deliverables 9](#_Toc117467137)

[5 References 9](#_Toc117467138)

[5.1 (Citations and references adhering to University guidelines or IEEE) 9](#_Toc117467139)

[5.2 Formatting Requirements: 9](#_Toc117467140)

[Your written assignments must be presented in the following format: 9](#_Toc117467141)

[5.3 Written Presentation 10](#_Toc117467142)

# Introduction

## Background (overview of topic and motivation)

Traffic Sign Recognition is becoming more and more promising since the recent interested development of unman driving cars. Moreover, the human error made in traffic sign has always been a risk in driving cars, there were a large amount of people died of this in China ever year. However, Traffic Sign Recognition is of great help since it is able to remind drivers of the signs, and automobile as well. The motivation for the development of system is to save lives from car accident and improve accuracy of identify traffic signs. The abbreviation could be used in the following sectors are Convolutional Neural Network (CNN) and Rectified Linear Unit (ReLU). The rest of the paper is organized as follows: section 1.2, covers the aim of the product and 1.3 explain the objectives behind it. Section 2.1 and 2.2 covers the competitive analysis and a brief related literature review. Section 3.1 to 3.3, correspondingly details the development methodology, technique and version control plan. Section 4.1 to 4.4, explains the project plan and lists the important activities with deadline. Section 5, shows the reference.

## Aim

Develop a Traffic Sign Recognition system that based on a lightweight Convolutional Neural Network model.

## Objectives

The objectives are as follows.

Ob1: Finish a background review of the existing CNN technology.

Ob2: Design the CNN architecture.

Ob3: Train and test the CNN architecture with a public dataset named Chinese Traffic Sign using appropriate technology.

Ob4: Evaluate the model.

Ob5: Explain the work to related audience.

## Product Overview

### Scope (What will it do? How will it works?)

Traffic Sign Recognition is able to identify and classify traffic signs, which would greatly benefit in traffic management, traffic sector and road safety. The model realizes traffic sign identify through 3 stages: image pre-processing, detection and classification. The architecture would apply CNN as the classifier.

### Audience (Who is it for?)

The proposed product would bring a large benefit to government agency that controls traffic, as well as the automobile developers.

# Background Review

## Summary of existing approaches (e.g., Competitive analysis, if appropriate)

First, Haque et al. (2021) describe a light-weight CNN model for traffic sign recognition. The architecture is consist of four convolutional layers, two overlapping max-pooling layers and then followed by one fully-connected layer. And the proposal model achieve 98.97% accuracy as the authors claimed. Secondly, Vennelakanti et al. (2019) propose a CNN Ensemble to identify traffic signs, which uses a feed-forward network with six convolutional layers, 3 max pooling layers and 2 fully connected layers. and three CNNs in total as the final output. the accuracy is 98.11% for triangular traffic signs and 99.18% for the circles. Thirdly, Shustanov and Yakimov (2017) suggest a three CNN layers, one full-connected layer model with soft max as a classifier. The authors claimed the accuracy of 99.94% is achieved. Moreover, Sun, Ge and Liu (2019) propose a light-weight CNN classifier which is consist of two convolutional layers, two pooling layers and two full connected layers and the test accuracy for the model is 98.2%. Lastly, Alghmgham et al. (2019) suggest a CNN network with two convolutional layers, two max pooling layers, one dropout layer and 3 dense layers. The author claims achieve the accuracy of 100%. The following table 1 shows the difference between the models.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Convolutional layer | Pooling layers | Fully-connected layers | Soft max | accuracy |
| Haque et al. (2021) | 4 | 2 | 1 | √ | 98.97% |
| Vennelakanti et al. (2019) | 6 | 3 | 2 | √ | 98.11% for triangle, 99.18% for circle |
| Shustanov and Yakimov (2017) | 3 | 0 | 1 | √ | 99.94% |
| Sun, Ge and Liu (2019) | 2 | 2 | 2 | √ | 98.2% |
| Alghmgham et al. (2019) | 2 | 2 | 0 | √ | 100% |

Table 1. Competitive Analysis between different CNN architecture.

## Brief summary of related literature (e.g., Annotated bibliography, or initial literature review, with a brief summary of sources)

One of common problems in CNN model design is to find the balance between accuracy and the network depth since large network with small data would cause an overfitted model, but one single layer is not enough for the accuracy. In this paper, Haque et al. (2021) implement overlapping max pooling and sparsely used stride convolution made training faster and reduced overfitting issue. The benefit of a light-weight structure is it would lower the energy usage, and the architecture is deep at the same time since it uses 4 convolutional layers. And Vennelakanti et al. (2019) suggest an ensemble of CNNs. There are 3 CNNs to aggregate the output, which is of great help in improving the accuracy. It implements a fully connected hidden layer and drop out in between to avoid overfitting problem. Shustanov and Yakimov (2017) design 3 models to find the balance between data and layers, final model is with one soft max layer, which is used to normalize the output of previous layers, enable to classify 16 kinds of traffic sign. Sun, Ge and Liu (2019) use ReLU as the activation function to All CNN layers, enable to learn complex features. At the same time, to prevent overfitting, dropout layer is added. Alghmgham et al. (2019) proposed a model that uses a large number of layers, it added 2 pooling layers to prevent overfitting problem by decreasing the dimension of image. It also imply the ReLU as the activation function in CNN. However, the creativity about the model is it uses leaky ReLU since it overcomes the problem of dead neurons that is faced if the original ReLU is used. Basically, leaky ReLU does not output zero when the input values are less than zero, instead it outputs negative value. So after each convolution layer, and before the pooling layer, there is a leaky ReLU activation function used.

# Methodology

## Approach (Description of the research and development methodology, e.g., Software development model, requirement gathering method, test, and evaluation process)

This software would be developed in Agile mode and use case diagram and sequence diagram to requirement gathering. The test method is dynamic testing.

## Technology (Implementation tools & resources such as hardware and software)

The product would be developed with the help of Google’s open-source machine learning framework, TensorFlow.

## Version management plan (e.g., Git repository or shared drive)

GitHub will be used to store the progress of the program.

# Project Management

## Activities: tasks required to complete each objective

Ob1 Finish a background review of the existing CNN technology.

A1.1 Conduct a systematic research of current CNN models.

A1.2 Complete a feature competitive analysis.

A1.3 Complete a literature review.

A1.4 Conduct the requirement gathering.

Ob2 Design the CNN architecture.

A2.1 Draw a scratch of the Traffic Sign Recognition model including details the CNN architecture.

A2.2 Study the dataset and summary features.

A2.3 Complete GUI of the proposed product.

Ob3: Train and test the CNN architecture with a public dataset named Chinese Traffic Sign using appropriate technology.

A3.1 Pre-process the images from dataset.

A3.2 Realize the CNN architecture with code according to the scratch.

A3.3 Train the CNN model with dataset.

A3.4 Test the CNN model with dataset.

Ob4: Evaluate the model.

A4.1 Get accuracy of the proposed model.

A4.2 Reflect the model.

Ob5: Explain the work to related audience.

A5.1 Complete the report.

A5.2 Conduct the presentation.

## Schedule i.e., Gantt or other, showing activities, deadlines



## Data management plan (e.g., Google folder for project logs, reports, literature etc.)

All the related documentation would be upload to this link <https://github.com/Haleywuyile/Traffic-Sign-Recognition> on GitHub.

## Deliverables

The following are deliverables: weekly meeting logs includes progress, next steps and supervisor comments, requirements gathering, Agile model, testing documentation, reports includes proposal, interim and final, ethics forms, and useful links of literature.

# References

## (Citations and references adhering to University guidelines or IEEE)

Alghmgham, D. et al. (2019) ‘Autonomous Traffic Sign (ATSR) Detection and Recognition using Deep CNN’, *Procedia Computer Science,* 163(12), pp.264-274. Available at: <https://www.sciencedirect.com/science/article/pii/S1877050919321477> (Accessed: 6 November 2022).

Shustanoy, A. and Yakimov, P. (2017) ‘CNN Design for Real-Time Traffic Sign Recognition’, *Procedia Engineering,* 201(9), pp.718-725. Available at: <https://www.sciencedirect.com/science/article/pii/S1877705817341231> (Accessed: 6 November 2022).

Sun, Y. and Ge, P. and Liu D. (2020) ‘Traffic Sign Detection and Recognition Based on Convolutional Neural Network’, *2019 Chinese Automation Congress (CAC)*, 2019, pp. 2851-2854. Available at: <https://ieeexplore.ieee.org/document/8997240/references> (Accessed: 6 November 2022).

Vennelakanti, A. et al. (2019) ‘Traffic Sign Detection and Recognition using a CNN Ensemble’, *2019 IEEE International Conference on Consumer Electronics (ICCE)*, 2019, pp. 1-4. Available at: <https://ieeexplore.ieee.org/document/8662019/references> (Accessed: 6 November 2022).

Wasif Arman, H. et al. (2021) ‘DeepThin: A novel lightweight CNN architecture for traffic sign recognition without GPU requirements’, *Expert Systems with Applications*, 168 (5), pp.114-127. Available at: <https://www.sciencedirect.com/science/article/pii/S0957417420311283> (Accessed: 6 November 2022).

## Formatting Requirements:

## Your written assignments must be presented in the following format:

* It must be word-processed in 11-point Arial font
* It must be black text on a white or ivory background
* All pages must be numbered
* Margins must be as follows: Top: 1 inch, Bottom: 1 inch (2.5 cm), Left: 1.25 inches, Right:
* 1.25 inches (3.2 cm)
* Use a line spacing of 1.5
* Numbers and captions to figures and tables should be at the bottom of the figure or table. If the figure or table is mounted sideways into the report, then its bottom is on the right-hand side of the report.
* The report should not normally contain more than 80 tables/figures.

## Written Presentation

* The project proposal must have a concise written presentation and referencing style.
* It should also have a clear & logical presentation.