

## Technical Documentation (Fiat Visio)

### 1. Introduction

#### 1.1 Purpose

This document is to provide a technical overview of the functioning and structure of our code for our project and how to use the provided code to train and test a pre-trained convolutional neural network on MATLAB. The document was first drafted on May 1, 2017 and finalized on May 18, 2017 and was written by Daniel Kim and Jason Ligon.

#### 1.2 Definitions, Acronyms or Abbreviations

There will be some use of acronyms and abbreviations used throughout the document in order to provide brevity and more concise reading. The following acronyms and abbreviations are:

- CNN - convolutional neural network
- NN - artificial neural network
- R-CNN - region-based convolutional neural network
- MOT - Multiple Object Tracking database
  - MOT16 - Multiple Object Tracking database from year 2016

### 2. Technical Documentation

Our code provides an easily editable framework that allows for it to be used on different types of object detection and classification. The code was written in MATLAB version R2016b and uses the Computer Vision System Toolbox and Neural Network Toolbox that usually comes standard with a full copy of MATLAB. Earlier versions of MATLAB may require that you download and install these toolboxes separately. Important directories such as the location of training and testing images are hardcoded into the code and therefore must be altered should the code be used on a different system that it was coded on. Currently, our code is written to work with training and testing images that are aimed to classify and detect traffic lights.

#### 2.1 Important Files

There are two main files of code that our project consists of and they are: "ROICreate.m" and "RCNN.m". The "ROICreate.m" file uses the given training data to create ROI labels for the R-CNN to use for learning to detect the target object, which is in this case traffic lights, which need to be manually selected. The "RCNN.m" file contains the main meat of the project and

code and handles the NN constructions, training, and testing. The “ROICreate.m” file is, in our setup, run first and then the “RCNN.m” file is ran afterwards.

## *2.2 Source Code Documentation*

- a) ROICreate.m - programmed and written by: Daniel Kim

Traverses through the training images and asks for the user to draw an ROI around the target object for the detection and classification task. The code will save all of the file paths and ROI information in a new table which must be saved as a .m file to be used for the rest of the code and project.

- b) RCNN.m - programmed and written by: Daniel Kim, Jason Ligon

Constructs and pre-trains a normal NN with the CIFAR-10 dataset. The code will then load the .m file created from the ROICreate.m file and use the information to train the R-CNN using the pre-trained network. The code will then traverse through the test images and run them through the R-CNN and display the image and draw a ROI around any detected traffic lights.