# Report of a Image Classification Project based on Convolution Neural Networks

Vishrut Kaushik, Derrick Liu, Lucas Lymburner, Hansen Qin, Ruiyang Wang

**Abstract**—The perception task 1 of the MECHENG599/ROB535 course asks for a image classification of vehicles in snapshots from a game engine, and assign them to three labels accordingly. We tackled this problem by feeding the images into our trained Convolution Neural Network, and the final prediction accuracy in the provided test data set is 52%. This report explains the over-all structure of our method, and also includes the work distribution among our team members.

**Index Terms**—Image Classification, Convolution Neural Networks (CNN), work distribution

## 1 Team Work Distribution

Because there are two projects in the MECHENG599/ ROB535 course: Control and Perception, we split our team into two sub-groups. Group 1 includes Lucas Lymburner and Hansen Qin and focuses on the control project, and Group 2 includes the rest of the team and works on the perception project. In Group 2, Ruiyang Wang works on finding bounding boxes of the object based on a convolution neural network (CNN) and cropping images accordingly, as well as writing of the report, while Vishrut Kaushik and Derrick Liu are responsible for implementing and training the CNN for our image classification. our code for perception task is available here [1]

## 2 Training Process

Our training data is a set of images with bounding boxes of the interested object in each image and their corresponding labels. The first step involves cropping and resizing the images based on their bounding boxes, and the second part trains the neural network based on the processed data. We will explain each step in more details in this section.

### 2.1 Image Processing

#### *2.1.1 Training*

Because vehicles are small in overall field of view in most of the images of the training data, there is a need to crop those images according to the bounding boxes provided so that the neural network is not polluted by picking up all the irreverent details in the image. In addition to that, vehicles are not necessarily have a constant size in each image, so a resizing of cropped images into a constant size is needed for later neural network training. This part is accomplished using MatLab and the final result can be illustrated as in Fig.1.

*The authors are with the University of Michigan, Ann Arbor.* {vishrutk,dsliu,llymburn,qinh,ruiyangw}@umich.edu

1. https://github.com/vishrutskaushik/ROB-535-Percpetion-Project

#### *2.1.2 Testing*

Because there is no data for bounding boxes given for test images, we trained a CNN based on ImageNet from Keras [1] by replacing its head fully-connected layers with four fully-connected dense layers to output the predicted bounding box coordinates. Then test images are cropped with MatLab based on the same procedure in 2.1.1 section mentioned above.

(a) Original image from the training data (b) Cropped and resized image focuses on the vehicle targets

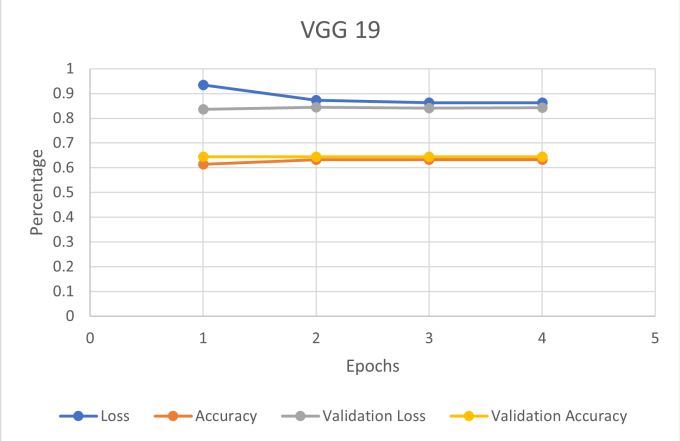
Fig. 1: Effect of the Image Processing step

### 2.2 Image Classification

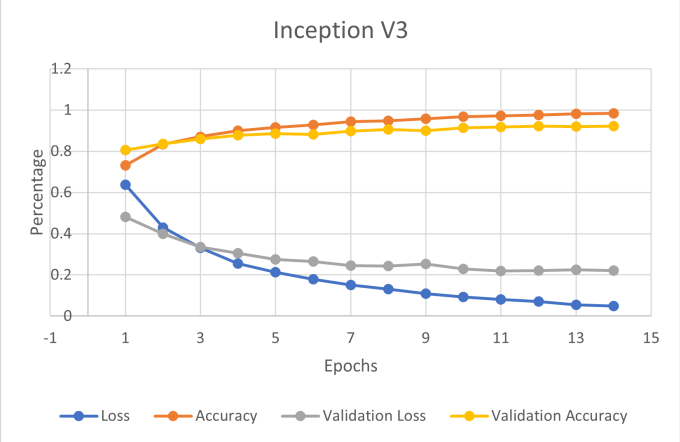
To classify images with vehicles inside into three categories, we combined VGG, Inception and Resnet defined in Keras [1] by letting each model vote simultaneously and take the maximal voted value as our final result. The performance of the combined convolution neural network is summarized in Fig.2

## 3 Conclusion

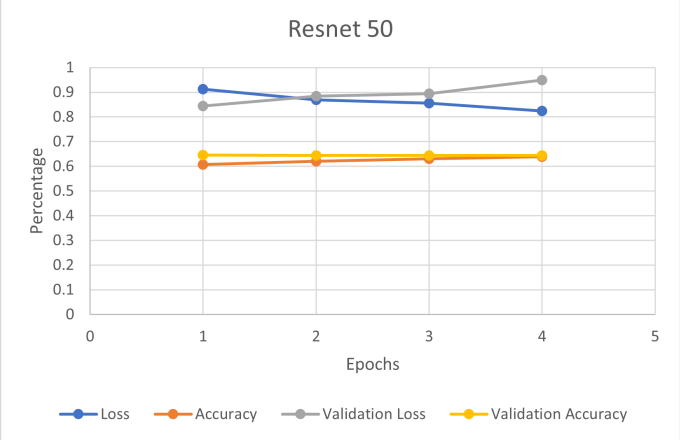
We designed a combined convolution neural network based on the existing VGG, Inception and Resnet model in Keras to classify images into three categories based on vehicles contained in them. But vehicles appear small in most of our training images, so that we have to first identify object in each image with a convolution neural network based on Imagenet and resize the image to focus on the interested




(a) Performance of the VGG model




(b) Performance of the Inception model targets




(c) Performance of the Resnet model targets

Fig. 2: Performance of different CNN models implemented in image classification

object in order to train and test the each CNN model we used without letting it be distracted into other parts of the image. The final accuracy of the modified neural network is 52% in the test data set, and we are confident that it can beat the baseline.

## REFERENCES

[1] F. Chollet *et al.* (2015) Keras. [Online]. Available: https://github.com/fchollet/keras