Master's Programme in Mechanical Engineering

Detecting Trash and Valuables with Machine Vision in Passenger Vehicles

Nilusha Jayawickrama

Aalto University

MASTER'S THESIS

Aalto University
MASTER'S THESIS 2020

Detecting Trash and Valuables with Machine Vision in Passenger Vehicles

Nilusha Jayawickrama

Thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Technology.
Otaniemi, 30 July 2020

Supervisor:	professor Kari Tammi
Advisor:	Klaus Kivekas

Aalto University
School of Engineering
Master's Programme in Mechanical Engineering

Aalto University

Abstract

Aalto University, P.O. Box 11000, FI-00076 Aalto www.aalto.fi

Author
Nilusha Jayawickrama

Title
Detecting Trash and Valuables with Machine Vision in Passenger Vehicles

School School of Engineering

Master's programme Mechanical Engineering

Major Mechanical **Code** MEN.thes

Supervisor professor Kari Tammi

Advisor Klaus Kivekas

Level Master's thesis **Date** 30 July 2020 **Pages** 52 **Language** English

Abstract

The research conducted here will determine the possibility of implementing a machine vision based detection system to identify the presence of trash or valuables in passenger vehicles using a custom designed in-car camera module. The detection system was implemented to capture images of the rear seating compartment of a car intended to be used in shared vehicle fleets. Onboard processing of the image was done by a Raspberry Pi computer while the image classification was done by a remote server. Two vision based algorithmic models were created for the purpose of classifying the images: a convolutional neural network (CNN) and a background subtraction model. The CNN was a fine-tuned VGG16 model and it produced a final prediction accuracy of 91.43% on a batch of 140 test images. For the output analysis, a confusion matrix was used to identify the correlation between correct and false predictions, and the certainties of the three classes for each classified image were examined as well. The estimated execution time of the system from image capture to displaying the results ranged between 5.7 seconds and 11.5 seconds. The background subtraction model failed for the application here due to its inability to form a stable background estimate. The incorrect classifications of the CNN were evident due to the external sources of variation in the images such as extreme shadows and lack of contrast between the objects and its neighbouring background. Improvements in changing the camera location and expanding the training image set were proposed as possible future research.

Keywords convolutional neural network, CNN, background subtraction, classification, prediction, camera module, image dataset, confusion matrix, cycle time

urn http://urn.fi/URN:NBN:fi:aalto-201712188201

Acknowledgements

I would like to begin by thanking Professor Kari Tammi for making me a part of his research team in the mechanical engineering department of the Aalto University and for giving me the opportunity to become a collaborator of the Anti-Trash Project within which my thesis was based upon. I would also like to thank my advisor Klaus Kivekäs for the continued support and guidance rendered with respect to the project planning as well as for the suggestions and recommendations given with regard to the documentation material.

Next, I would like to thank two of my fellow research workers. The first is Risto Ojala for sharing his expertise in machine vision and for the advice and direction given in the implementation stages of the algorithmic models. Then I also thank Jesse Pirhonen for the help given with respect to the technical aspects especially with regard to the connections and the setup of the camera module in the vehicle. I would also like to extend my gratitude to my entire research team in Aalto, guided by Professor Kari Tammi and to other personals who attended my workshop session and made their contribution towards creating the image dataset.

Additionally, I thank the Anti-Trash project partners in Hamburg for hosting me and allowing me to be a part of their workshop sessions which proved invaluable in obtaining ideas and incentives in order to analyze and accomplish the project requirements. My final gratitude goes to another Anti-Trash project member who serves as a computer vision engineer in CARNET Barcelona, Esteve Mascaró, for providing useful suggestions during the creation of the algorithms.

Contents

List of Acronyms and Abbreviations

ANN	Artificial neural network
AP	Average precision
CNN	Convolutional neural network
GPIO	General purpose input/output
IoU	Intersection over union
KL	Karhunen–Loève
mAP	Mean average precision
R-CNN	Region-based convolutional neural network
ReLu	Rectified linear unit
RPN	Region proposal network
SCP	Secure copy protocol
SGD	Stochastic gradient descent
SIFT	Scale-invariant feature transform
SSD	Single shot detector
SSH	Secure shell
SVM	Support vector machine

List of Figures

List of Tables

1. Introduction

1.1 Background

In the modern day, much focus is being directed towards public transportation and shared vehicles over private modes of transportation primarily to reduce traffic flow and vehicle emissions. Driver-less taxis are a developing concept in the area of smart mobility. However, in order to encourage customers to use the different means of public transportation, it is important that the vehicle interiors are maintained and kept clean. With usage, customers tend to leave behind certain trash items and even their belongings on the seating area. The leftovers would then be exposed to the next customer who would be travelling in the vehicle and that is far from ideal.

Henceforth, the project undertaken here is focused on detecting trash and other valuables left in the seating area of relevant vehicle applications using machine vision in order to retain customer trust in using public transportation modes. More specifically, the possibility of using such an integrated detection system in a shared passenger vehicle will be examined by the creation of an algorithmic model and analysing the results in order to prove its viability.

Generally, there has been a lot of research related to trash detection and classification. Much of it has been focused on ensuring appropriate disposal and recycling at waste disposal sites autonomously. Initiatives have also been taken to reduce littering in public places by means of mobile surveillance robots capable of detecting and alerting somebody around it once it detects trash items on the floor. Visual detection of underwater trash has also been looked into by various research groups in an attempt to explore and extract trash using underwater autonomous vehicles.

Yet, there has been no significant prior initiatives taken to ensure cleanliness in public transportation or shared vehicles by autonomous means. At this point, our research steps in. We propose to use an in-vehicle camera system to take scheduled pictures of the vehicle interior seating area in order to detect and output the cleanliness level of the vehicle. Within our scope, a convolutional neural network (CNN) would be trained to recognize any trash or customer valuables which might have been left behind in the vehicle. The detection should yield a good level of accuracy under varying lighting conditions as well since the interior lighting of the vehicle will change.

During the algorithm implementation phase, a top down approach will be used to test and optimize the classification results. In the first phase the CNN will simply have a binary output criterion where it would be expected to simply identify if the seat is empty or not. Thereafter, the classification would be extended to three possible output classes:

1. An empty or clean seat
2. A dirty seat (a seat with one or more trash items)
3. A seat with one or more valuables left behind. Valuables could include items such as keys, mobiles, wallets, bags, etc..

Finally, in order to get a measure of the cleanliness level, the algorithm will be developed to be able to identify a dirty seat and predict the level of cleanliness as an output once it detects trash.

Henceforth, the key challenges of the study conducted here are as follows:

1. Obtaining the dataset for the training of the CNN would be extensive. It would have to consist of a multitude of test images comprising of a good distribution of clean and messy/dirty backseats under varying lighting conditions. Randomness in the images with respect to certain criteria such as ambient light changes are important as well in order to train the model well. Moreover, the results of the prediction model should be unaffected by such forms of variation.
2. It is essential to optimize the model to obtain detection results of high accuracy. In order to do the optimization, multiple training epochs are required while varying the hyper parameter values until their best combination is obtained.
3. The camera itself should not be in operation continuously as it is redundant in terms of energy consumption and computation. Hence it is important to determine how and when the camera would be capturing image data. Efficient operation of the in-car camera module in terms of power consumption and computation should be evident in the final demonstration.

1.2 Objectives

1. Produce a working demonstration of the operation of the in-car camera detection in the rear seating area of the research vehicle.
2. The CNN will ultimately be expected to obtain a classification accuracy of up to 85 % if not more.
3. The developed CNN should have the ability to differentiate between an empty seat, a seat containing trash and a seat containing valuables.

4. A custom dataset will be created and compiled in order for it to be made available to be reused or optimized for extended neural network functionality in relation to in-car cleanliness projects.

1.3 Scope

The CNN developed will classify if the image from the camera contains trash or valuables or if it is simply an empty seating area free from any form of unknown object. It will not be expected to perform object detection to identify and classify individual items on the seat. Therefore no information about the orientation or pose of such items will be available either. Moreover, the CNN will be trained with image data consisting of either trash or valuables and hence, a scenario in which a customer leaves both trash and valuables will be effectively neglected at least for the initial implementation phase.

The developed in-car camera system will be limited to the vehicle backseat and its detection range is limited to the foot rest area and the rear seat itself. The CNN will be trained for both uniform lighting and shadow existing scenarios within the scope of this project and hence optimization for it will be done collectively and not for each individual case. Any possible obstructions in the view of the camera will not be considered for the training of the CNN. Therefore the seating area will be assumed to be visible at all times

1.4 Thesis structure

Now that the report has presented an overview of the focus criteria of the research and what is to be expected upon completion, it is useful to identify how the rest of the report unravels with respect to its structure. In chapter 2, the state of the art which revolves around the project focus will be elaborated. It will primarily examine details involving any existing datasets which would be utilized for the project at hand. Additionally, the results and corresponding elements of discussion of related work with respect to the algorithms and neural network architectures will be elaborated. Moreover, a research gap identification will also be presented here. Afterwards, in chapter 3, the methodology utilized in this research which would present details of the proposed system architecture along with the on-site connections as well as an in-depth procedure into how the algorithmic models will be trained in sequential phases. In chapter 4, the results of the training process over the course of the project will be presented and analyzed in order to realize the detection accuracy. In addition, the performance of the presented fine-tuned models is compared with existing trash classification models. The results are then discussed alongside an analysis of the computational efficiency of the overall system in a quantitative manner. The discussion points included here would also highlight the pros and cons of the developed model. Finally, in chapter 5, the conclusion will indicate how the performance of the model relate to the original objectives and desired outputs, a summary of its current limitations as well as future implementations in developing the model to be an integral part of the shared vehicle fleet of a city.

2. State of the art

Based on the description of what is expected with respect to the objectives of the research mentioned in the previous section, it is now necessary to explore existing models, data sets and techniques. The purpose in doing so would be to obtain a clear and comprehensive understanding of the types of machine vision based approaches utilized for similar classification purposes, its current state of development in research and applications, and what additional measures are being implemented in the modern day to monitor the interior of automobiles to ensure high levels of interior cleanliness as well as damage free maintenance. Furthermore, it would also be useful to determine the means by which communication plays its role in data transfer of machine vision based trash detection applications. More importantly, the entire analysis of the current state of art presented in this chapter will be the basis upon which research gap will be identified with respect to our task at hand and how we can work on making any existing platforms better.

2.1 Relevant image databases

With regards to trash images, the most popular database utilized by most of the research applications for classification purposes and model training is the trash image dataset developed and made open source by Thung and Yang (2016). The dataset was released during their development of a trash classification model to separate trash into six classes which included plastic, glass, cardboard, metal, paper and general trash, totalling up to 400-500 raw images.

Research conducted by Fulton et al. (2019) included the creation of a substantial dataset of underwater trash obtained using autonomous underwater vehicles (AUVs). The dataset was made available to the public after doing careful image annotations in order to aid in the training of convulutional neural networks so that they would be capable of performing similar object detection. Note that the dataset in the research was sourced from the originally created J-EDI dataset of marine debris (Japan Agency for Marine Earth Science and Technology).

Another image database used over the years for a majority of the pretrained neural network models was the ImageNet database (Deng et al. 2009). The database is widely used in cases where the initial weights are not completely randomized but alternatively the model would have the trained weights from the Imagenet model at the commencement of training the model.

2.2 Vision based trash detection and classification applications

Following their own trash image database creation, Thung and Yang (2016) tested out two techniques by which they could perform the classification into one of six different categories of waste. The models used here were support vector machines (SVMs) with scale invariant feature transform (SIFT) features and also their own CNN. In comparison to their work with the image database, the results of the algorithms were less accurate in both models where the SVM model yielded an accuracy of 63% and the CNN could only produce an accuracy of 22%. Hence their conclusion was merely that their experiments proved SVM with SIFT to be the better option along with an additional argument that they were unable to find the optimal hyper-parameters of the CNN due to a lack of a sufficiently large enough data source (Thung and Yang 2016).

More recently, Adedeji and Wang (2019) worked on a proposal of an intelligent waste material classification system. The model was developed using a pretrained ResNet-50 CNN model, which served as a machine learning tool and an extractor, for the purpose of recognition. It was used together with a multi-class support vector machine to classify the images from a precompiled trash image dataset (Thung and Yang 2016) which are used as input to the model. The expected outcome was to separate a given image containing waste into one of four different classes; glass, metal, paper or plastic. The model was able to yield an accuracy of 87% on the trash image dataset as depicted in figure 2.1 below.

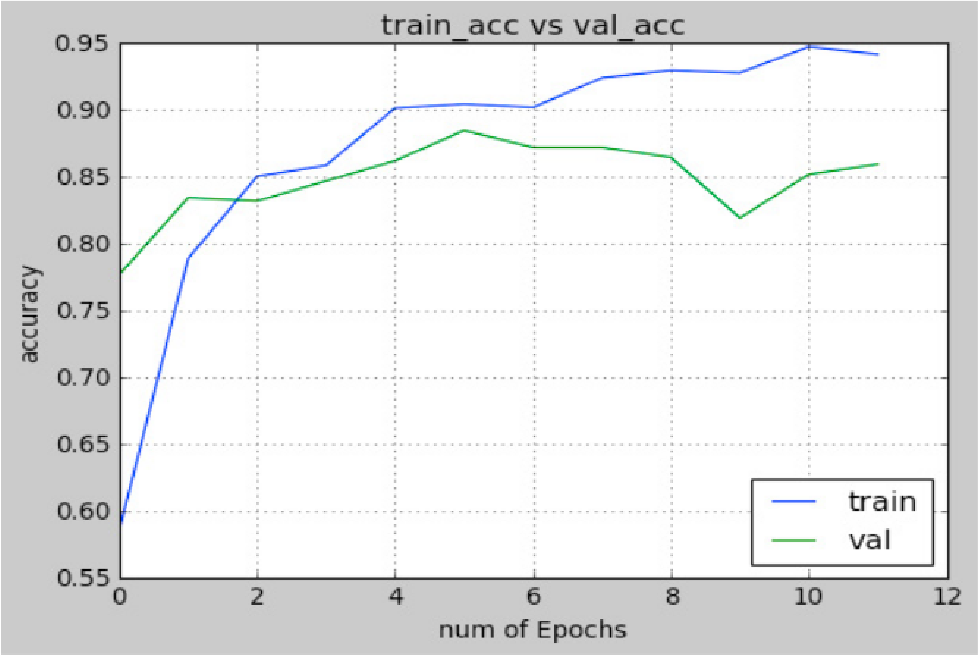

Figure 2.1. Training and Validation accuracy. Reprinted from Intelligent Waste Classification System Using Deep Learning Convolutional Neural Network (Adedeji and Wang 2019).

The material type is also a variable which could determine the type of substance or object which exists in an image, and consequently classify it to be trash or otherwise. Related work based on the theory here was done by Donovan (2016) who developed a vision integrated trashcan equipped with a camera connected Raspberry Pi. The system could automatically sort items based on the type using a custom image recognition model built using Google's Tensorflow AI. A constraint in the project however was that the material type was limited to compost (Donovan 2016).

Another real time trash detection project was undertaken by Salimi et al. (2018) where a trash detection plus classification system was developed on a social trash bin robot which would be found in public places with considerable crowds. In the implementation here, more traditional computer vision approaches were used in the following manner:

1. Haar-Cascade: For the detection of objects which might be laying on the floor
2. Viola-Jones object detection: A descriptor to obtain the features of the images obtained by the mounted camera. The descriptor essentially breaks the object detected to its elementary characteristics which are primarily its shape and texture.
3. SVM: Classification of the images to the types of organic waste, non organic waste and non waste.

The robot will autonomously traverse while detecting for trash and once it does, it will calculate the distance to it. After moving close to it, the robot will produce a sound to get people's attention so that they would come and throw the trash to the bin mounted on the robot. A Kalman based tracker was used as a measure to improve the detection stability of the frames captured by the mounted stereo camera. The technique here could be useful in the case where the in-vehicle camera of this project is subject to any vibrations which may distort the input images. Testing was performed for a total of 1000 images by means of 5-fold cross validation and the average prediction accuracy was revealed as 82.7%. However, testing was done offline and subsequently, the system was not implemented to the actual robot (Salimi et al. 2018).

The vision system developed by Yasuhiro et al. (2005) was focused on implementing an outdoor service robot to collect trash. Although the authors primarily discussed the accuracy of different types of mechanical sensors, they also provided discussion with respect to the algorithmic technique utilized for the detection itself. The research makes use of template matching for detecting an object in an image. Hence robust matching was done using multiple template images which were compressed using Karhunen-Loeve (KL) expansion, which is essentially a data compression tool, and then getting a normalized cross correlation. The vision system implemented however was at an introductory stage and therefore its accuracy was not yet measured in real time application (Yasuhiro et al. 2005).

A similar vision based approach was used by Zhihong et al. (2017) where a robotic grasping system capable of autonomously sorting garbage was designed. Object recognition and classification for this project was done by Regional Proposal Generation (RPN) and a VGG16 model which combine as sub nets to a Fast Region-CNN or otherwise known as R-CNN (Girshick 2015). RPN and VGG16 are both examples of existing CNN model architectures. Yet, it is important to note that the actual backbone of the R-CNN is the RPN which has complete image convolutional features to predict both object bounds as well as the prediction scores at every position (Girshick 2015). Hence, R-CNNs are described as being capable of achieving near real time speeds by the employment of deep neural networks. Thereby, the authors compared the use of their R-CNN to previous models showing that it is superior in terms of both training rates as well as the detection accuracy. The computation time of the model in practice was recorded as 220 ms and the rates of missed detection and false detection were obtained as 3% and 9% respectively (Zhihong et al. 2017).

Just a year later, Bai et al. (2018) proposed the implementation of another autonomous garbage pickup robot whose perception module was supported by a popular CNN architecture known as ResNet. The model used for the pickup robot utilized 34 layers and hence the ResNet-34 architecture which was developed by Zhang et al. (2016). The classification errors for both garbage and non garbage are depicted in tables 2.1 and 2.2 respectively below. The results demonstrated that powerful deep learning based neural networks work well for real time detection applications for classification of both garbage as well as valuables (Bai et al. 2018).

Table 2.1. Classification error on the test test. Reprinted from Deep Learning Based Robot for Automatically Picking Up Garbage on the Grass (Bai et al. 2018).

Category	Error (%)
Bottle	8.13
Can	9.89
Carton	9.06
Plastic bag	14.32
Waste paper	22.3

Table 2.2. Classification error for valuables. Reprinted from Deep Learning Based Robot for Automatically Picking Up Garbage on the Grass (Bai et al. 2018).

Category	Bottle	Can	Carton	Plastic bag	Waste paper
Cup	0.153	0.184	0.012	0.009	0.003
Book	0.002	0.010	0.136	0.005	0.012
Shoes	0.005	0.023	0.038	0.009	0.003
Phone	0.007	0.011	0.065	0.004	0.008
Bag	0.007	0.013	0.009	0.032	0.004
Wallet	0.010	0.023	0.089	0.012	0.009

However important it may or may not be, it is interesting to determine how vision systems perform under more unusual conditions with respect to the input images where the background would not necessarily be so ordinary anymore. Fulton et al. (2019) tested a set of deep learning models and evaluated them to find out the accuracy of detecting marine litter by utilizing underwater trash images from AUVs. The classification was intended to be threefold: plastic, man made objects and bio material. The project chose four successful and well proven network architectures to be trained for detection using four NVIDIA GTX 1080s. The architectures were YOLOv2 (Redmon and Farhadi 2016), Tiny-YOLO (Redmon et al. 2016), Faster RCNN with Inception (Ren et al. 2015) and single shot detector (SSD) with MobileNet (Liu et al. 2016; Sandler et al. 2018). They mostly utilized fine tuning with pretrained weights from the COCO dataset (Lin et al. 2014). The quantitative results following the experiments are as shown in tables 2.3 and 2.4 and the preferred architecture model from an accuracy standpoint was determined as the Faster R-CNN. YOLOv2 proved to provide a fair balance between the rate of computation and accuracy while SSD proved to be the fastest (Fulton et al. 2019). Note that in table 2.3 each columnar value per network is a unique type of accuracy metric in object detection with mAP being mean average precision (therefore AP is average precision) while IoU corresponding to intersection over union.

Table 2.3. Detection metrics in mAP, IoU and AP. Reprinted from Robotic Detection of Marine Litter Using Deep Visual Detection Models (Fulton et el. 2019).

Network	mAP	Avg. IoU	plastic AP	bio AP	rov AP
YOLOv2	47.9	54.7	**82.3**	9.5	52.1
Tiny-YOLO	31.6	49.8	**70.3**	4.2	20.5
Faster R-CNN	81.0	60.6	**83.3**	73.2	71.3
SSD	67.4	53.0	**69.8**	6.2	55.9

Table 2.4. Performance metrics in frames per second for different types of processors and their associated graphics. Reprinted from Robotic Detection of Marine Litter Using Deep Visual Detection Models (Fulton et el. 2019).

Network	1080	TX2	CPU
YOLOv2	74	6.2	0.11
Tiny-YOLO	**205**	**20.5**	0.52
Faster R-CNN	18.75	5.66	0.97
SSD	25.2	11.25	**3.19**

2.3 Model type evaluation

Neural networks have been a popular choice among researchers in the creation of trash detection and classification models. The models are seldom created from scratch especially for demanding machine vision applications. Therefore, the commonly used approach is to import a prebuilt model architecture, such as AlexNet, and then utilize model training techniques such as transfer learning and fine tuning where new inputs are used to train the parameters of a prebuilt neural network architecture. Afterwards, relevant adjustments could be made in order to enhance its performance for the new prediction model. The most common types are as follows (Pai 2020):

1. Artificial Neural Network (ANN): A type of feed forward neural network where the inputs are always processed in the forward direction of the network. The type of model here was used by Batinic et al.(2011) to predict quantities along with the compositions of waste in Serbia.
2. Convolutional Neural Network (CNN): CNNs are probably the most common type of neural network models which are mostly used for image processing applications. The network is composed of blocks which are essentially filters or kernels capable of extracting features from input images by means of convolution. The initial layers identify simple features such as edges and simple shapes while the deeper layers capture more complex patterns which may represent certain objects for instance (Zeiler and Fergus 2013). A majority of the trash detection implementations utilize CNNs for prediction and some examples include the work done by Adeji and Wang (2019) in their intelligent waste material classification model and

also the research conducted by Zhihong et al. (2017) where a deeper CNN was integrated for detection purposes in a robotic grasping system for garbage sorting.

The possibility of using existing objection detection network architectures such as YOLO were also evident in the research on detecting marine trash (Fulton et al. 2019). However the algorithms described here require a high amount of computational resources and may therefore suffer from redundant use of memory and time consumption when we consider their application to a specific and custom dataset as we do in this project since such networks have been trained for the identification of several hundred output classes.

Moving on to the more traditional computer vision approaches which are still being utilized abundantly in the modern day, there have been many techniques in this regard which were employed for trash detection as elaborated in the previous subsection. Work done by Thung and Yang (2016) for instance employed SIFT features to detect patterns in the images and concluded that such an approach would be beneficial over a CNN in the case where it would be difficult to tune the CNN to obtain its optimal parameters due to limitations in the data source. However, the resulting detection accuracy with SIFT features would saturate at a lower maximum compared to what it would be with an optimally tuned CNN when we consider the complexity of trash related images. Other computer vision based approaches similar to the one described here were the Viola-Jones detection which was used by Salimi et al. (2018) in their real time trash detection project to obtain the shape and texture of the images to be detected (the accuracy of the predictions were close to 90% here) and the Karhunen-Loeve expansion technique in the research done by Yasuhiro et al. (2005) to detect trash in outdoor terrains.

Despite the variation in detection techniques, classification has almost exclusively done by SVM since it is a straightforward and well established approach. Research papers which have utilized neural networks found it optional to use SVM for the final classification stage. Alternatively, a classification layer has been used in the form of an activation function at the end of the network such as a softmax or a sigmoid function.

2.4 Sensor based cabin state monitoring in automobiles

The section here presents a brief study on the current state of autonomous monitoring inside passenger vehicles (outside the norm of the more commonly integrated sensors such as door lock checks) in order to ensure cleanliness within the cabin. The various vision possibilities in vehicles have been widely studied in the past 20 years with the inclusion of looking-in vision sensors to observe the passenger and environment conditions within the cabin. The sensors here have been focused towards being able to accomplish tasks such as tracking occupants (and their mannerisms) in vehicles, controlling the cabin odor and ensuring good cleanliness levels within it (Intelligent Vehicle Vision Systems 2007).

Ensuring cleanliness by way of odor control in vehicles has been an area focused on by researchers in the recent past. Such autonomous odor assessment was carried out by Li et al. (2016) in an attempt to replace human panels to inspect vehicle cabin smells. Consequently their predictive model yielded correlation values between 0.67 and 0.84 for the different odor parameters with respect to the ground truth obtained from human experts after testing on ten different vehicles (Li et al. 2016). Sensor based air quality control within the vehicle was also looked into by Ramya et al. (2012) where gas detection sensors were utilized to detect and alert the user alongside producing suitable ventilation once the sensor readings exceed a predefined threshold. Although the research is relevant to cabin cleanliness, it was primarily focused on the elimination of toxic gases such as carbon monoxide in the vehicle interiors (Ramya et al. 2012)

Installation of a camera in passenger vehicle cabins has rarely been done by in previous studies. One of the oldest publications in this regard was in 2003 when Fritzsche et al. (2003) proposed a camera implementation to monitor the internal space of a vehicle with a minimum of one camera. The system was intended for general purpose actions in vehicles such as deactivating airbags when the seats are not occupied or detecting unused seat belts (both of which could be accomplished in the modern day without the requirement of an in-car camera). A more recent patent released by Schofield et al. (2017) demonstrated the use of an overhead camera together with a video display screen purely for the purpose of enabling the driver to monitor the rear part of the cabin in their vehicle.

Studies directly addressing object and cleanliness detection inside vehicles have been conducted. Piirainen et al. (2009) experiments the possibility of detecting the presence of objects on vehicle seats by utilizing a combination of wave type transducers and electric field sensors. The interior monitoring system demonstrated the capability of being able to detect the object type as well as its pose up to a certain extent by examining the wave signals from the receivers. Furthermore, Hyundai, Total, NUMA and GreenTropism (a startup) carried out a study (first of its kind) which presented results from their own in-car cleanliness study recently in 2019 with a focus of ensuring passenger comfort in shared vehicles. Although it is said that the initiative was taken to obtain data about stains and ambient air, the specifics of the system implementation are not publicly available (Hyundai Motor to present results of in-car cleanliness study n.d.).

2.5 Research gap

It is clear that the current state of the art with respect to vision algorithms is in a well established state and the options are sufficient. Hence it is not a requirement to create such algorithms from scratch but alternatively import and modify their parameters along with suitable additions to the code in order to suit our in car trash detection application. Autonomous detection of trash itself has been an area of high focus over the past decade or so and several studies proved to be effective in achieving it with good levels of accuracy. In automobiles however, the integration of a certain level of automation in order to maintain interior cleanliness has been recognized as a significant research topic only recently and hence development is still in progress.

By combining the set of facts explained in the preceding paragraph, the research gap lies in utilizing a suitable machine vision based algorithmic technique to address the issue of cleanliness and seat occupancy with respect to customer belongings in passenger vehicles utilized for shared purposes when there is nobody inside. Subsequently, the aim would be to detect trash and valuables in the rear seating compartment of car cabins using an appropriate vision based technique and cross reference the results with known labelled data in order to determine the accuracy of the system. Once it is achieved, the system would be possible to create a suitable alert system in future work when a detection has been made so that the vehicle interior would be vacant and trash free when the next passenger gets inside.

3. Methods

3.1 On-site functionality and electrical design

3.1.1 Research Vehicle

A single four door passenger vehicle was used to train the vision based prediction model since the intuition was that all shared vehicles released by a single company will be of the same model. The model of the vehicle used was a Ford Focus (2017 model) which is dedicated for research work in the mechanical engineering department of Aalto University. Figure 3.1 below shows the actual vehicle which was used for this project.

Figure 3.1. Aalto research vehicle which was utilized to implement the in-car detection system

3.1.2 Component description

For the setup inside the vehicle, the following components were used:

1. **Raspberry Pi 3 Model B**

   Figure 3.2 presents the model of onboard controller used in this study. The Raspberry Pi is a single board computer which was used as the on-site processor in order to run the image capture algorithm and then transfer the captured image to the remote server. The module is Wi-Fi compatible and therefore is capable of image transfer once it is connected to a network. It requires a 5V power supply for operation and draws a maximum current of up to 2.5A. The Raspberry Pi could also support a display since it includes an HDMI input in addition to its four USB ports. However since the the addition of a display would congest the vehicle in terms of space, the preferred option was to control it remotely from a laptop computer. The means by which it was done is explained in section 3.2.

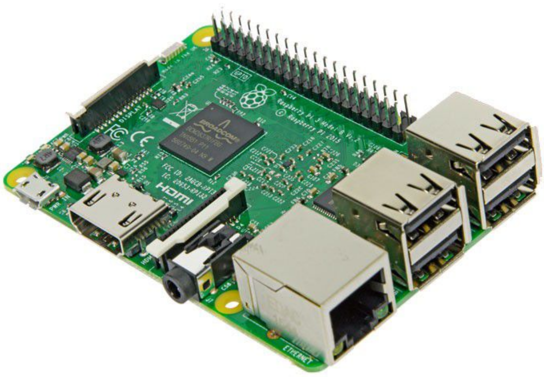

Figure 3.2. Raspberry Pi 3 Model B. Single board computer which was used for onboard processing of the image (Raspberry Pi 3 B n.d.)

2. **ELP 170 Degree Fisheye Camera Module**

   The objective was to capture the seat state using a single camera module. Moreover, its actual placement in the vehicle cabin was also constrained primarily because the vehicle had a sunroof. Therefore it was not a possibility to place it over the rear seating area on the roof midway along the width of the vehicle. Consequently it was placed adjacent to the rear roof handle on the left in place of an already existing light in the vehicle. Due to its placement in the corner, a fish-eye camera had to be used in order to capture the entirety of the rear seat and the rear floor area.

   In accordance to the requirement here, the camera choice was an ELP 170 degree wide angle fisheye camera shown in figure 3.3 which supported autofocus and had a USB connection. The image resolution for each captured frame was set to 1920 by 1080 pixels.

Figure 3.3. USB camera which was installed in the vehicle for capturing the state of the rear seating area (ELP Webcam Wide angle 170 degree Fisheye Lens Camera USB Full HD 1080P 30fps Web Camera for Linux/Windows/android 2015)

3. **LED light strip**

   Since the system here is a vision based image capture application such, it is necessary to ultimately capture an image at any time of the day under varying light conditions. The LED installation was primarily to support image capture at low light conditions such as obtaining the seat state during the night when it is dark outside. The light module was primarily set up for demonstration purposes and was therefore not intended as a final solution. The light draws a total current of up to 1.2A.

   Henceforth, a single coloured LED strip similar to the one shown in figure 3.4 was purchased and cut to the required length (about 900 cm) and integrated to an enclosure made of a plastic casing, a metal strip and a light diffuser. The installed light module was attached on the two ends to the rear roof handles of the vehicle using 3D printed tight fit attachments.

Figure 3.4. LED strip (Uusi LED Strip 5m 24V IP20 White 2500K CRI98 n.d.)

4. **N Channel MOSFET**

   It was necessary to implement a two state switch (ON/OFF) which could be logically controlled within the image capture program of the Raspberry Pi. The reason here was to control the LED strip in such a way that it only turns on when an image has to be taken from the camera. Hence it technically acts in a similar way that a flash light acts in any photography camera. The light in this case would turn on whenever an image has to be taken and not specifically when the light condition is low as implementing such a mechanism would be tedious and moreover, redundant since the light would anyway have no effect when there is a lot of ambient light.

   Therefore, an N-channel MOSFET of type FDP7030bl was chosen. The main requirement for the MOSFET was its maximum gate threshold voltage which in this case was 3V in accordance to the voltage output of the I/O pins of the Raspberry Pi.

5. **12V Battery Supply** The maximum voltage required within the on-site setup is 12V which is for the LED strip. Hence a 12V rechargeable battery pack was used to obtain the voltage. Voltage from the supply was stepped down to 5V as well to power up the Raspberry Pi.

With the exception of the camera and the LED module, the rest of the components were located in the trunk of the research vehicle.

3.2 Connections and communication

Figure 3.5 below presents a circuit schematic of the setup inside the research vehicle. The step down block has been included here to make it clear that the same power supply was used to power up all the components of the onboard setup.

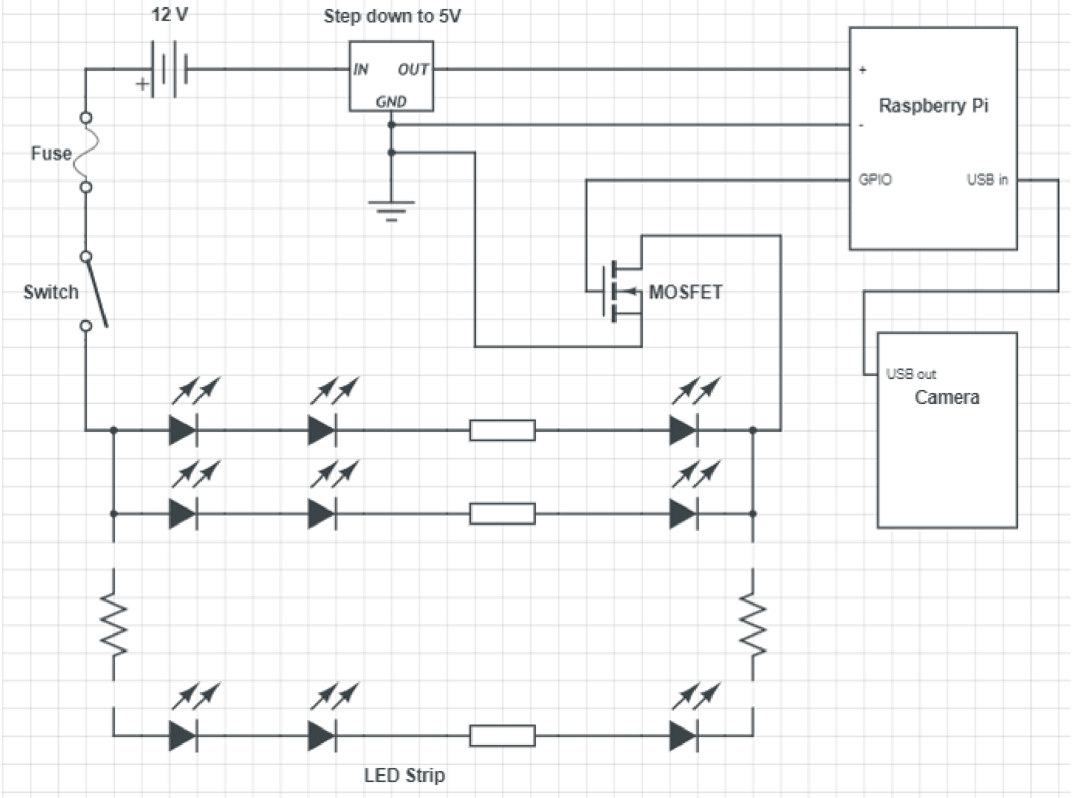

Figure 3.5. Schematic of the proposed on-site circuit design

The following list summarizes the main connections depicted in figure 3.5 above of the on-site setup:

- In addition to the components described in the previous section, there is also a fuse in series with a switch between the connection of the positive terminal of the battery and the LED strip for safety reasons.
- The camera is connected to the Raspberry Pi via a USB cable

- The MOSFET pin connections are as follows:
  - Source: Grounded
  - Drain: Negative end of the LED strip
  - Gate: a GPIO pin of the Raspberry Pi

Although it is not evident in the schematic shown in figure 3.5, the Raspberry Pi is connected to a 4G network via Wi-Fi which enables both access to the Raspberry Pi in order to execute its program, as well as a connection to the remote server for the transmission of the captured image for processing. For remote use of the raspberry pi a communication protocol known as secure shell (SSH) was used. Therefore, SSH enabled remote operation of the raspberry pi by a laptop. Additionally, another network protocol known as secure copy protocol (SCP) was utilized for the transmission of captured images from the Raspberry Pi to the remote server.

Using the setup described, the image captured by the camera is sent to the Raspberry Pi and stored temporarily until it is transmitted to the remote server where the image will be subject to the prediction model. Figure 3.6 illustrates the process flow of the detection system.

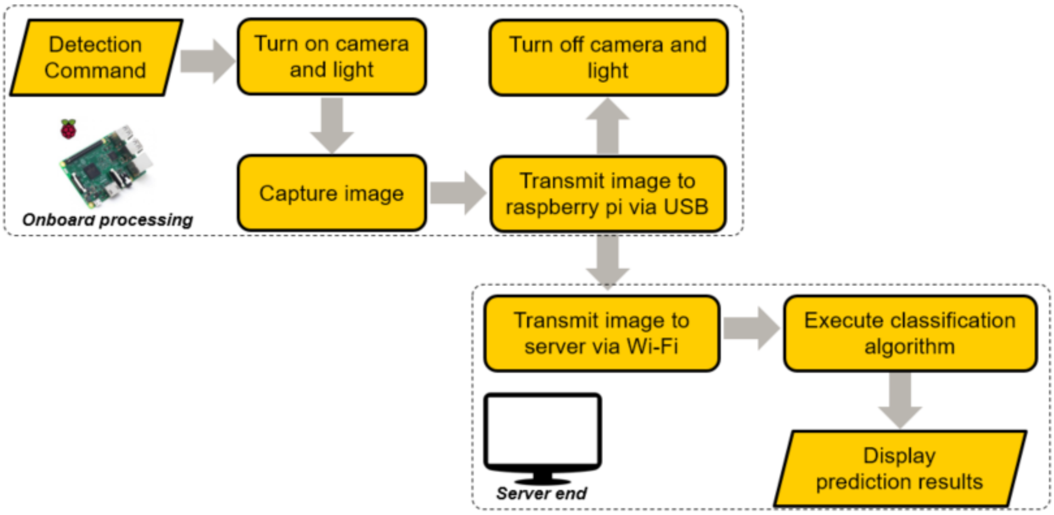
Detection Command
Turn on camera and light
Turn off camera and light
Onboard processing
Capture image
Transmit image to raspberry pi via USB
Transmit image to server via Wi-Fi
Execute classification algorithm
Server end
Display prediction results

Figure 3.6. Flowchart depicting the system functionality from image capture to result display

Furthermore, all of the images captured were expected to be annotated and compiled in order to create an image dataset which will be made available on an open source platform.

3.3 Final Setup

Figure 3.7 depicts how the final setup (which corresponds to the schematic in figure 3.5)appeared in the research vehicle. It is important to highlight that the vehicle here is not solely dedicated to this project and it has been used for other research work. Hence, although the setup appears elaborate, components such as the battery pack and the switch board circuit were already in place prior to the commencement of the project. The circuit box is the main addition to the setup in the trunk which contains the major portion of the circuit depicted in figure 3.5 including the Raspberry Pi. Hence the connections to the module here were:

- The USB extension from the camera to the Raspberry Pi
- Power cable for the Raspberry Pi (5V and ground)
- Power connections from the battery pack (12V and ground)
- Power connections to the LED module (12V and ground)

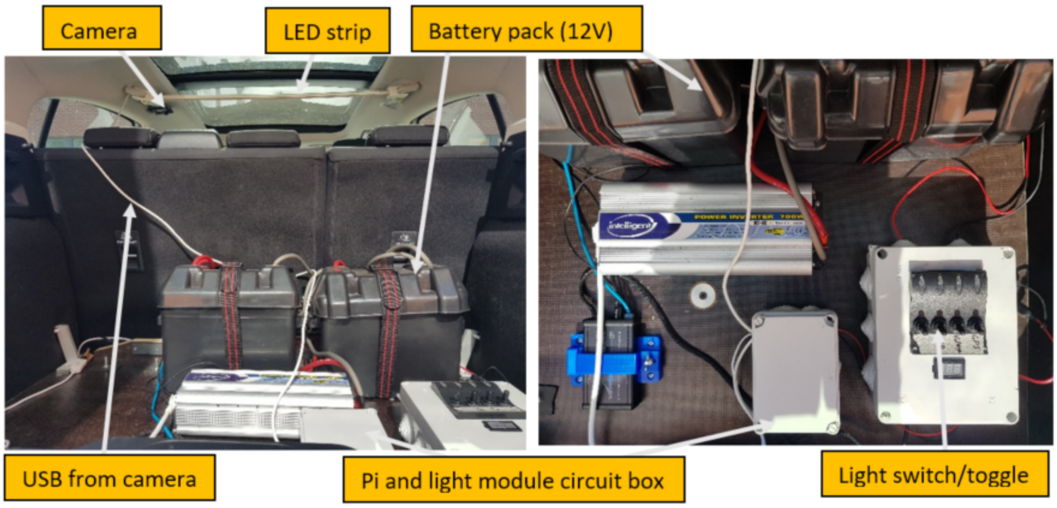

Figure 3.7. Finalized onboard setup of the research vehicle

Figure 3.8 presents close up images of the light module installation while figure 3.9 presents the difference of capturing an image frame from the camera module during the night using the existing vehicle interior lights (image on the left) and the integrated LED module (image on the right)

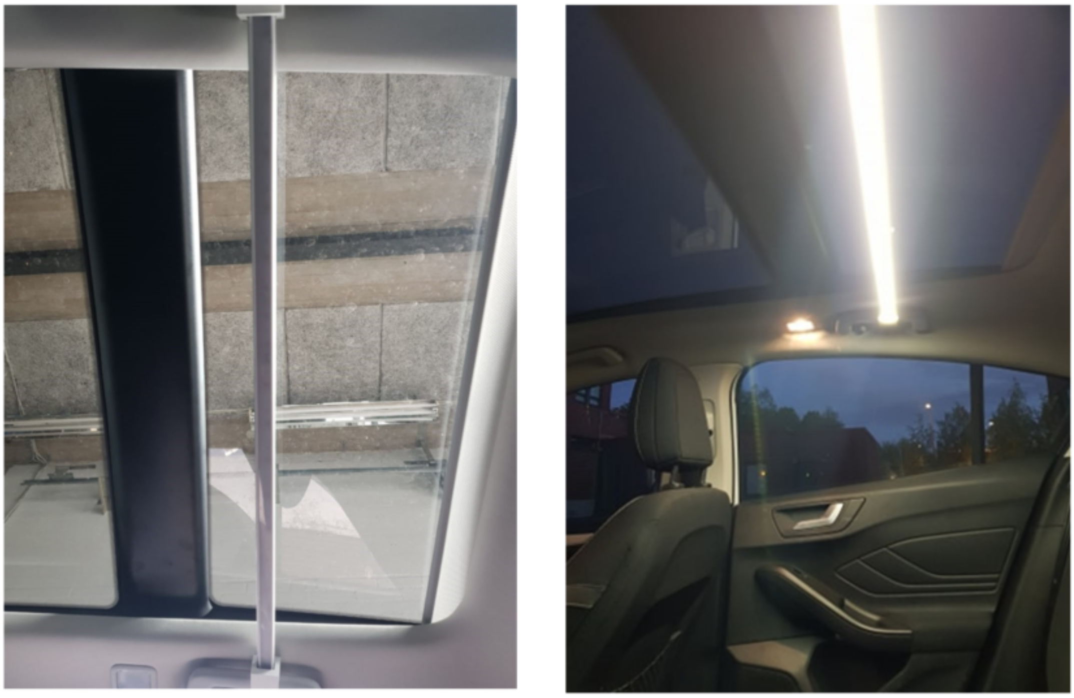

Figure 3.8. Installed light module which traverses over the rear seat in its off and on states

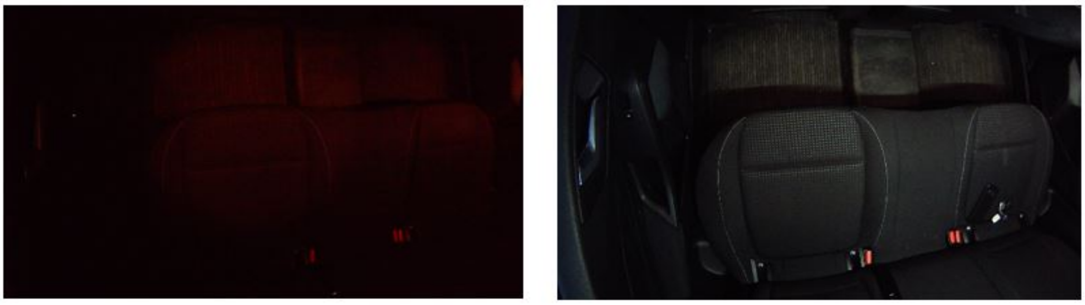

Figure 3.9. Sample images taken with the existing vehicle interior lights (left image) and with the installed LED strip

3.4 Data gathering

As formerly mentioned, there are three final output classifications of the prediction model which correspond to an empty seating area, a seating area containing trash and a seating area containing valuables. Hence the models had to be trained with such types of images. Moreover, it was important to ensure that external variation especially with respect to shadows and ambient lighting were distributed evenly across all classes and doing so was the challenging part of image gathering.

In order to collect images with a clean seating area (free from any unknown objects) with a good distribution of external variation, the proposed method was to execute a script on a daily basis which would run throughout a day in order to capture an image every periodic interval (for instance, every 30 minutes) so that the lighting levels and any possible shadows would be different across the images. The method here provides the advantage of requiring minimal human supervision since the image capture script could be left to run for as long as needed while the vehicle is parked outside (preferably on different spots on different days to have more lighting and shadow variation).

Collecting images with objects in the seating area is comparatively more tedious since it requires manual placement of different kinds of trash and valuables in the vehicle for each image that has to be taken. Doing so under varying lighting conditions is the biggest challenge here. Additionally, to collect images containing valuables, three workshops were scheduled (one during the winter and two during the spring and summer period). The idea here was to invite fellow research workers, students and other individuals to come with their day to day valuables to the research vehicle at allocated time slots so that their items could be placed one after the other at random locations of the seating area. Each time, one or more valuables are placed, an image would be taken and added to the dataset.

3.5 Proposed algorithms

Based on the evaluation of the models done in studying the state of the art, two algorithmic techniques were proposed to create the prediction model. The algorithms were designed as follows:

1. In the first phase, the model will only be expected to identify the difference between a vacant backseat of a vehicle and a seat with valuables or trash on it; hence a binary classification. Since the training here is not extensive, up to 200 images were planned to be captured from the in-car camera module and used for training.
2. In the second phase, the model had to be trained to categorize a given image scene into trash or valuable classes; in essence it should be able to identify if the items detected are trash or customer valuables (in addition to the empty seat classification). Therefore there will be three possible classifications. Training becomes more extensive at this point and hence to obtain a good classification accuracy the model will expect more training data. Hence, for this second phase, image augmentation was used to expand the dataset in addition to manually capturing more frames from the camera module. For this study, images were augmented by creating new images based on the existing ones by executing certain transformations such as horizontal flips, orthogonal translations, zoom levels and brightness changes in order to increase the dataset size and enable the CNN to be trained better.

3.5.1 Convolutional neural network (CNN)

Since the research here corresponds to an image processing application which is highly case specific, it was decided that one approach to accomplish classification would be to create a fine tuned CNN model. Hence the idea here was to import an existing CNN architecture from a neural network library and train it by adjusting its hyper-parameters.

The prebuilt CNN model selected was VGG16. The model here was initially proposed by K.Simonyan and A.Zisserman (2015). At the time, it achieved a test accuracy of 92% in ImageNet over 1000 classes. Figure 3.10 depicts the default architecture of the VGG16 network. As shown in the figure, it comprises of 3x3 kernel sized filters consecutively along the model. Since the first convolutional layer accepts images of a fixed size of 224 x 224, the original image resolution is shrunk accordingly (from 1920 x 1080). There is a total of five max pooling layers which are required for the purpose of progressively reducing the spatial magnitude of the input in order to avoid having a redundantly large amount of parameters and thereby to enhance computation. Since the final layer is a softmax layer, the probabilities for each output class per image will add up to 100%. The equation given below shows how the softmax function works in converting a vector of values to its corresponding probabilities:

$P(y_i) = \frac{e^{y_i}}{\sum\limits_{j} e^{y_j}}$

where:

y = Input value from vector

P (y) = Probability of the input value

i, j = Indexes of the vector and sum respectively

In addition, each hidden layer comprises of rectification through the use of rectified linear units (ReLu) which is a type of activation that corresponds to the graph in figure 3.11.

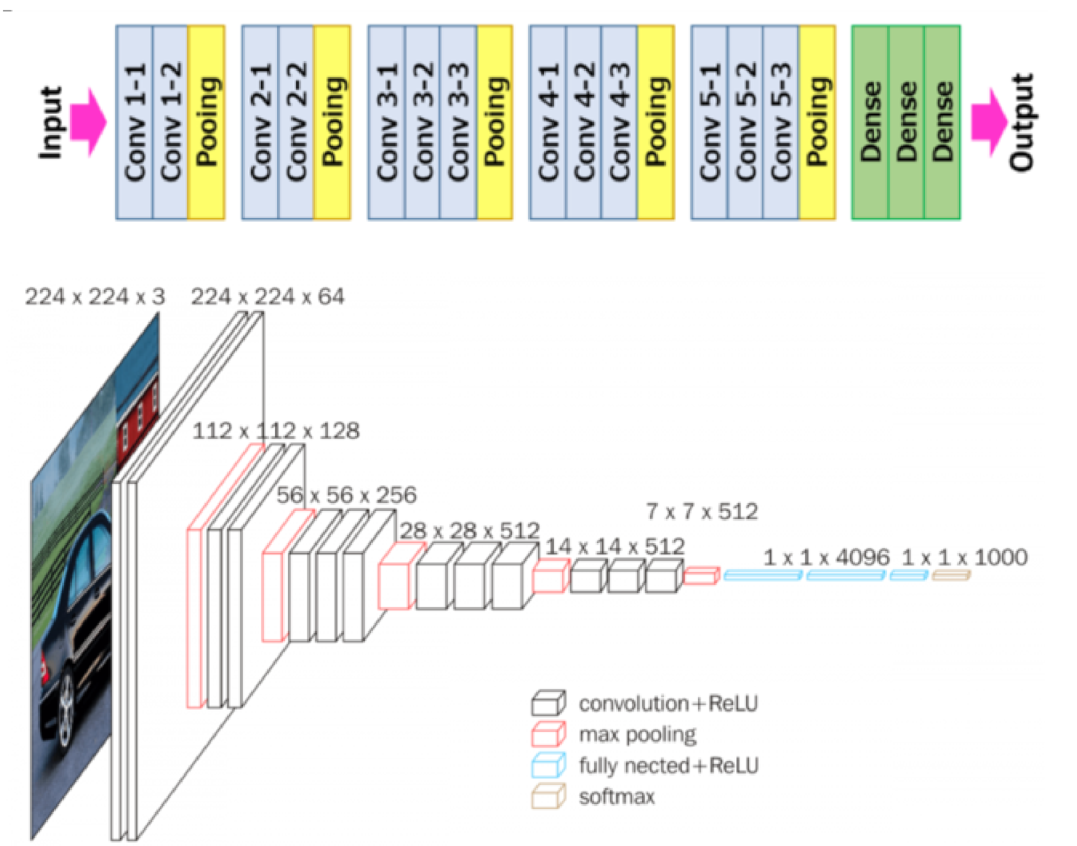

Figure 3.10. The default architecture of the VGG16 CNN model. Reprinted from VGG16 – Convolutional Network for Classification and Detection (Muneeb Hassan, 2018).

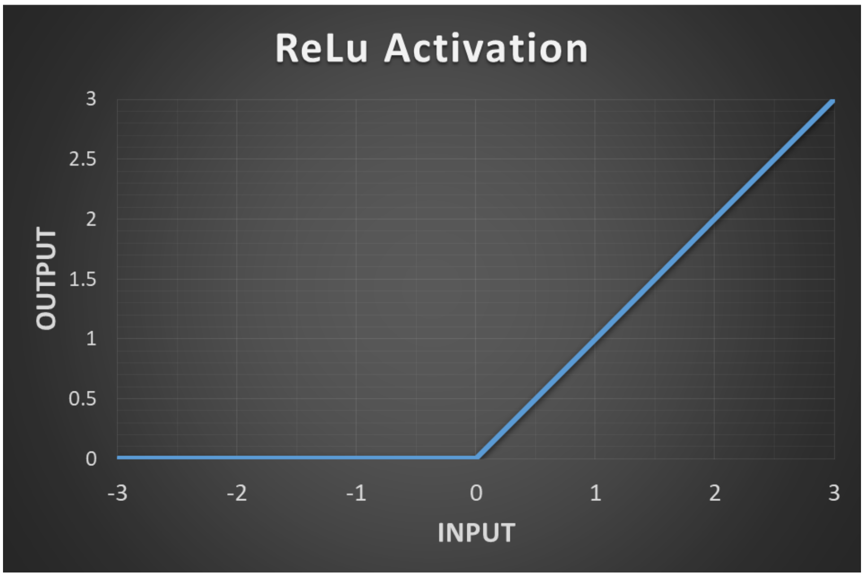

Figure 3.11. Graph corresponding to the activation function ReLu

For the creation and execution of the VGG16 model, the following software and libraries were employed:

- Python: Programming language used throughout the creation of the program scripts.
- Keras: Neural network library from which the model was imported and edited. It is written in python and is an open source library.

- Tensorflow: An end to end machine learning platform which in this case is the backend upon which Keras was run.
- OpenCV: Another library which was primarily used for image manipulation and preprocessing purposes. It is specifically created for computer vision applications and is therefore an ideal choice of utilization here.

In addition to the software given above, other general purpose and output visualization libraries such as numpy and matplotlib were also used.

Note that for the CNN implementation procedure described above, the image data were categorized into training, validation and testing sets in order to train and obtain the predictions. The training set contains the raw image data which the model will directly use for training itself by learning the features of the images over each epoch. In technical terms, we will use the data here to fit the model. The validation set, in essence, contains brand new data which the model does not use for training directly, but rather to predict on and evaluate the performance during each training. The purpose of employing validation during the training phase is to prevent overfitting which is when the CNN memorizes the training data rather than actually training itself for classification and thereby it will perform well on the training data, but will fail to yield good accuracies for new data. An indication of not overfitting is if the results of the training loss and training accuracy have a similar trend to the validation loss and validation accuracy across the epochs. Finally once the training is complete, the test set will be used for the predictions and the results here will indicate how good the model is in performance. Moreover, the results of the prediction model during the training process were checked using tools such as confusion matrices, trial comparison tables and manual validation (these will be discussed further under results in chapter 4).

3.5.2 Background subtraction

Background subtraction is a simpler and more traditional machine vision based detection algorithm. As the name suggests, the idea would be to initially form a background of the image which would thereby be the reference frame. In this case therefore, the background would essentially be an image which does not contain any trash or valuables (a clean and empty car seat). Once the background model is developed, new images can be fed and compared to it in order to detect any applicable foreground. The foreground here would hence be any form of unknown object on the rear car seat or its adjacent floor area. In theory, the background model would be an estimate since there is still some variation between different empty car seat images in terms of sources such as lighting changes and shadows. The model should subsequently be robust to such changes. The difference between new images with the background model will be obtained essentially by subtracting the developed estimate of the background from the new image and then applying a suitable threshold which would separate its classification to an empty image or otherwise. Figure 3.12 below presents a clear illustration of the main steps which correspond to the execution of background subtraction in this study.

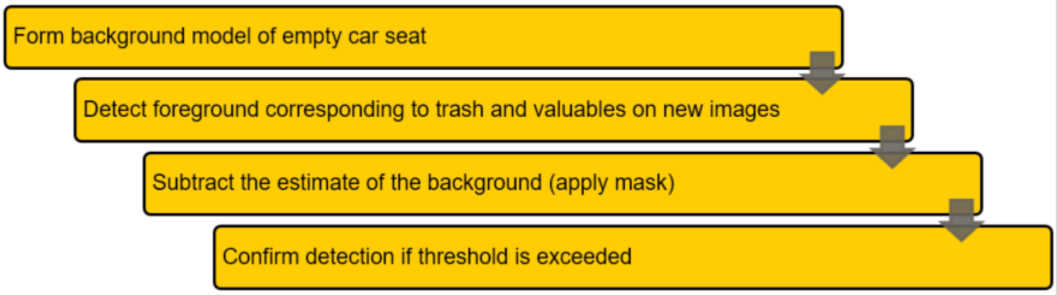
Form background model of empty car seat
Detect foreground corresponding to trash and valuables on new images
Subtract the estimate of the background (apply mask)
Confirm detection if threshold is exceeded

Figure 3.12. Main steps in developing and executing the background subtraction model for this study

Similar to the CNN model creation explained previously, the background subtraction model too was not developed from scratch since there is already a good availability of libraries capable of assisting us with image processing. In this thesis, OpenCV was used to create and train the background detection model.

4. Results and Discussion

4.1 Image dataset

Figure 4.1 depicts two sample images taken from the camera module of the research vehicle. Since this is a static camera, it can be observed that the boundaries for all of the images are the same. It is evident that the proposed areas of detection which include the rear seat and floor area are well covered by the fish eye camera. It also detects portions of rear doors and therefore any image could tell us if the rear doors are closed or not. The state of the rear doors therefore was considered as a source of external variation in addition to the lighting levels and shadows. Additionally, the back portion of the front seats may or may not be present in the captured image depending on the actual location of the front seats. For instance, if the driver seat is more frontward, then it may not be too visible in the image. If it is pushed further back however, a comparatively bigger portion of the seat will be visible in the image. Figure 4.1 also depicts the scenario here where the driver seat in the right image is actually more visible than it is on left image.

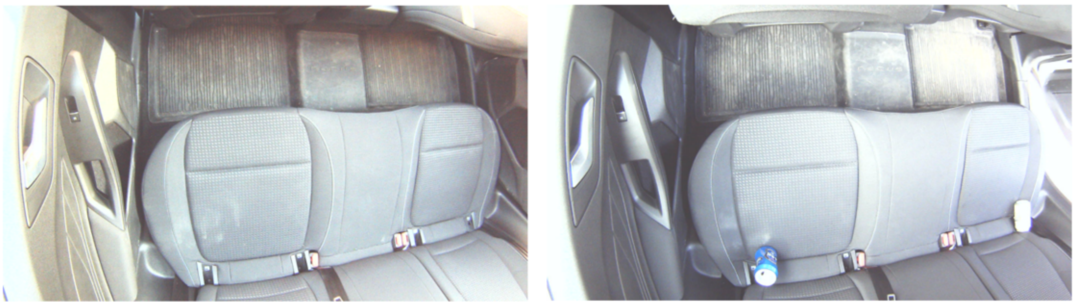

Figure 4.1. Sample images taken with the existing vehicle interior lights (left image) and with the installed LED strip

Table 4.1 summarizes the composition and categories of the compiled images taken from the on-site camera setup. All the images were of resolution 1920 x 1080 pixels and will be available on the open source platform with no further changes. Note however that the dataset also contains the augmented images which were used as input for the training process of the prediction model and the images therefore were reduced to a size of 224 x 224 pixels. Figure 4.2 presents four augmented versions of the images from figure 4.1. The parameters which were changed in the image augmentation process are given in table

4.2. The parameters here were chosen in such a way so that the image augmentation did not cut off any trash or valuable items on the seating area.

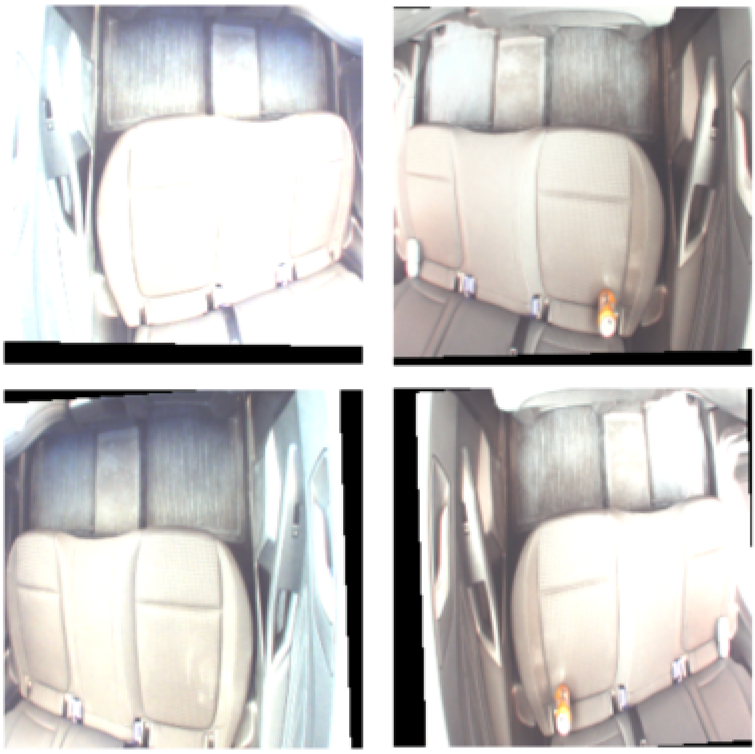

Figure 4.2. Augmented version of the sample images from figure 4.1. Images on the left depict two augmented versions of the empty seat while the images on the right depict the augmented versions of the seat containing cans

Table 4.1. Image categories and their respective compositions with respect to the image dataset. Note that the total for prediction model corresponds to the added tally with the exception of the original training images since the images here were augmented.

Image Category	Empty	Trash	Valuables	Total split
Training	160	151	181	492
Validation	192	120	120	432
Testing	60	40	40	140
Total original	412	311	341	**1064**
Augmented training	466	448	529	1443
Total for prediction model	718	608	689	**2015**

Table 4.2. Parameter types and their corresponding quantities which were applied for image augmentation

Parameter type	Parameter value	Description
Brightness	0.2 - 1.2	Range of values within which the brightness would be adjusted. The value represents the fraction of the original brightness. Hence values less than 1 will make the augmented image darker and vice versa
Rotation	2	Maximum degree by which an image will be rotated. Hence in this case an augmented image would be rotated by a value between 0 and 2 degrees
Height shift	0.05	Maximum percentage of the total image height which will be shifted. Hence in this case an augmented image would be shifted between 0 % and 0.05% of its original height (1080 pixels)
Width shift	0.05	Maximum percentage of the total image width which will be shifted. Hence in this case an image would be shifted between 0 % and 0.05% of its original width (1920 pixels)
Shear	2	Maximum number of degrees by which the image would be slanted. Hence in this case an augmented image would be slanted by a value between 0 degrees and 2 degrees
Horizontal flip	True	Allows horizontal flips for augmented images and hence an image may or may not be flipped horizontally upon augmentation
Zoom	0.1	Maximum percentage by which an image could be zoomed in. Hence in this case, an augmented image could be zoomed between 0% and 0.1% of its original size

It is also interesting to point out the different types of objects which are available in the dataset for images containing trash or valuables. In this study, only cans and bottles were included for the trash category of images. Hence in the image dataset, you will only see two types of trash. However, the bottles and cans were of different shapes (some were crushed), sizes and material. On the other hand, there were no restrictions in the types of valuables which were fed into the images. The most common valuables included were mobile phones, wallets, backpacks, and keys while the rare ones included laptops, gloves, earphones, spectacles and beanies.

Figure 4.3 depicts four images randomly picked out from each category of the three classes. Note however that the images containing valuables have, more or less, constant levels of lighting across all images. In the methodology section, it was explained that the image gathering for valuables was planned by having three workshop sessions. The reason behind the lack of variation in the valuables was that it was possible to have only one workshop and that was the one in the winter which is the time of the year when there is a lack of sunlight.

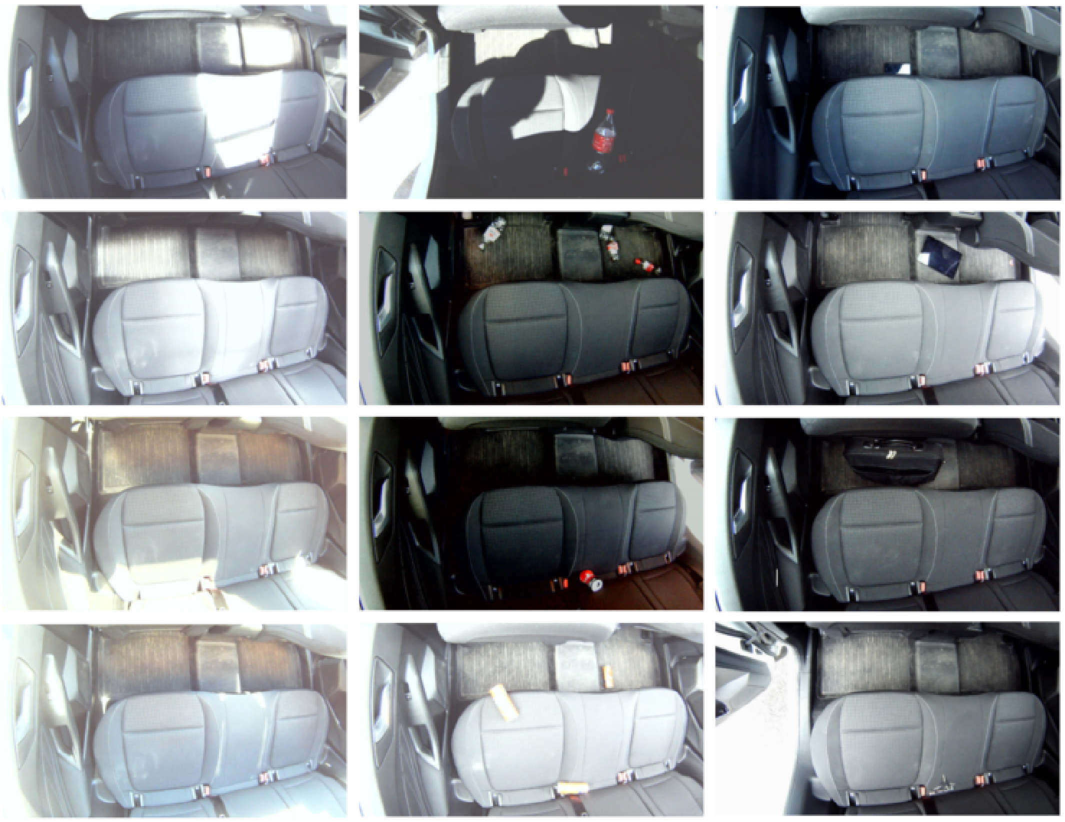

Figure 4.3. A series of randomly picked out images from the image dataset belonging to the three classes of empty, trash and valuables respectively (left to right)

4.2 CNN Results

4.2.1 Final architecture

During the training phase of the VGG16 model, a range of hyper-parameter combinations were tested out. For each combination of hyper-parameters, the model was tested from scratch and its results archived for the purpose of comparison with other trained versions of the model. Although techniques exist to minimize error based on the results of the model, there is a high amount of trial and error that is required in order to tune the network.

With respect to the architecture of the model, the final classification layer was fixed to be a softmax activation function for all trials. Since there is a total of three possible output classifications, a categorical cross entropy loss function was used. The loss function in other words is the objective function whose scalar value has to be minimized in order to yield a prediction model of high accuracy.

Since the value of the loss function depends on the weights of the VGG16 CNN, it is imperative to update and optimize the weights iteratively in order to minimize the loss. In order to do this, an optimizer was used in the CNN. There were two types of optimizers which were tested in this study and they were Adam optimization and Stochastic Gradient Descent (SGD) with Momentum. Upon result comparison, SGD was chosen as the final optimizer for the model since it gave better trends in updating the weights across training epochs. Additionally, in order to ensure that the loss function did not spiral out of control, regularization (of type l2) was integrated to the VGG16 model. The type of regularization used here is also known as ridge regression where a squared coefficient is added to the loss function. Using regularization proved useful in optimizing the model to generalize better on unseen data and therefore played a crucial role in improving the validation accuracy.

During training, the training accuracy and loss as well as the validation accuracy and loss were displayed for each training epoch in order to observe its variation. For a well trained model, both the training accuracy as well as the validation accuracy should increase gradually. Figure 4.4 shows how the values changed with the number of training epochs. In order to ensure that the rate of learning was well controlled during training, a learning rate scheduler was used with a step decay function which would reduce the learning rate by a factor after every predetermined number of training epochs. Hence, the equation given below was used to calculate and update the value of the learning rate during the training process:

$lr = initlr * drop^{\frac{1+epoch}{epochsteps}}$

where:

lr = Learning rate

initlr = Initial learning rate

drop = fraction by which the learning rate should reduce

epoch = current epoch

epochsteps = number of epochs after which the learning rate should be changed

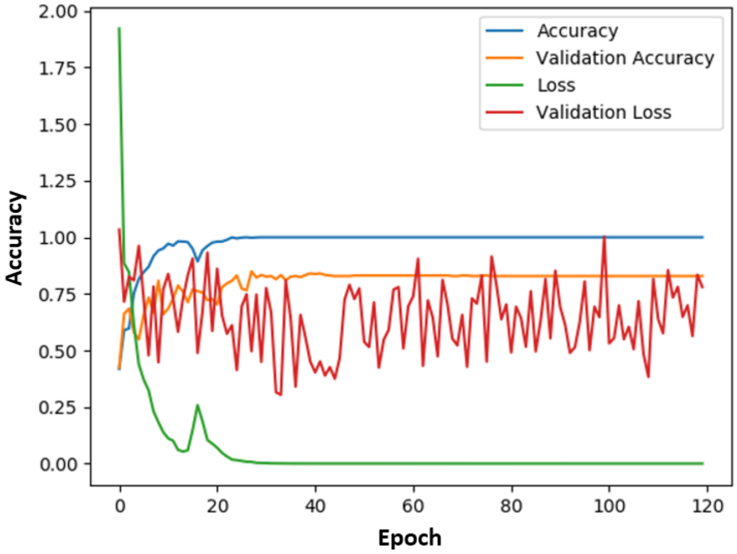

Figure 4.4. Change in the accuracy and loss of the model during the training phase. Compared to the plots, the validation loss (as indicated by the red curve) was inconsistent with heavy fluctuation

4.2.2 Confusion matrix and prediction certainties

The ultimate objective of the training model was to maximize the accuracy it yielded on the test batch of image for the model created here. Figure 4.5 depicts the final confusion matrix that the prediction model produced for the batch of test images. Interpreting the matrix is straightforward as it is a simple correlation between the actual class and the predicted class for all the images. Hence, adding up all the diagonal values from left to right will give the total number of correct predictions while the addition of the remaining values will equal to the number of false predictions. The final accuracy for the test set was therefore 91.43%. The composition of the images in the test set is given in table 4.1.

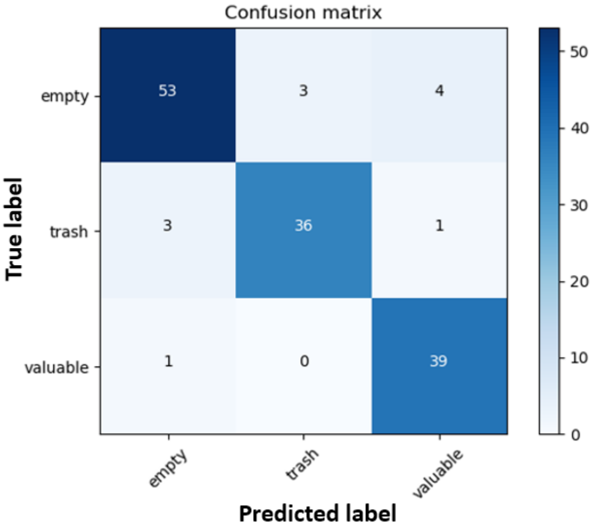

Figure 4.5. Confusion matrix for the batch of test images indicating the correlation between the true labels and the predicted labels of the images

Although the above analysis tool can be used to get a sense of how accurate the model is, it does not provide any quantitative values with respect to how the uncertainties of the predictions were for each of the test images. Knowing the certainties is useful in order to identify the confidence by which each of the predictions were made and in the case of false predictions, what went wrong. Therefore, in order to obtain the incorrect classifications, the program was written to generate two files both of which contained the certainty or probability value for each class in every image. Two files were used to separate correct predictions from the incorrect ones. Table 4.3 depicts the structure of such a file. Appendix A and Appendix B presents the full form of the files which correspond to the correct predictions and the false predictions respectively. Each index of the files given in the appendix corresponds to an image in the batch of test images. The index is used to identify the image and each of the indexes therefore is associated with three probabilities values corresponding to the certainty of the three output classes (Empty, Trash and Valuable). Note that all probability values for a single image add up to 1 (100%) since we use softmax activation.

Table 4.3. Prediction certainties of the three classes for the first ten images of the test batch reproduced from the file of correct predictions generated by the algorithm once the prediction model completes the classification

Index	Empty	Trash	Valuable
0	0.955	0.0113	0.0338
1	0.0611	0	0.9388
2	0.2498	0.0003	0.7499
4	0.3206	0.0003	0.6792
5	0.4347	0.5652	0.0001
6	0.065	0.8861	0.0489
7	0.1853	0.0002	0.8145
8	0.1583	0.0002	0.8415
10	0.1423	0.0002	0.8575

4.2.3 Analysis of the output classes

The classified images were also written as image files and stored in the remote server for graphical visualization. As seen from the images in figure 4.6 each image when written was annotated with the final classification as well as the certainty by which it was predicted.

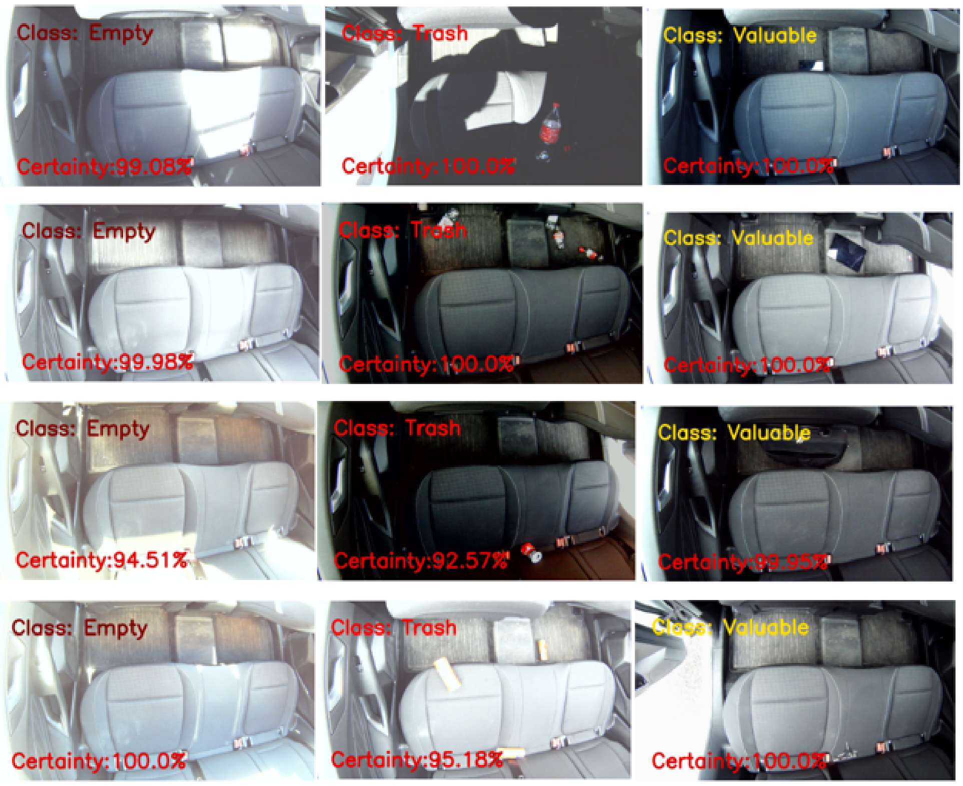

Figure 4.6. Images from figure 4.6 annotated with the predicted classification and certainty.

Across all classes, the images responded well to changes in rear door states and therefore the certainties were not particularly affected whether the rear doors were closed, opened or even partially opened. The scenario with the rear door states is evident in figures 4.6, as none of the images show any noticeable trends in the certainty regardless of how the rear doors are.

One common source of error across all image classes was that when the driver seat is pushed all the way to its rearmost position, the prediction model abnormally categorises it as trash image. Figure 4.7 depicts such an image. Note the circled section of the headrest of the driver seat which becomes visible only when the seat is in the position depicted in the figure. The main reason behind the form of error here is the lack of training data which included images of the driver seat position depicted in the image. Including more images with the driver seat pulled all the way back will therefore allow the model to generalize better and eliminate this form of random variation.

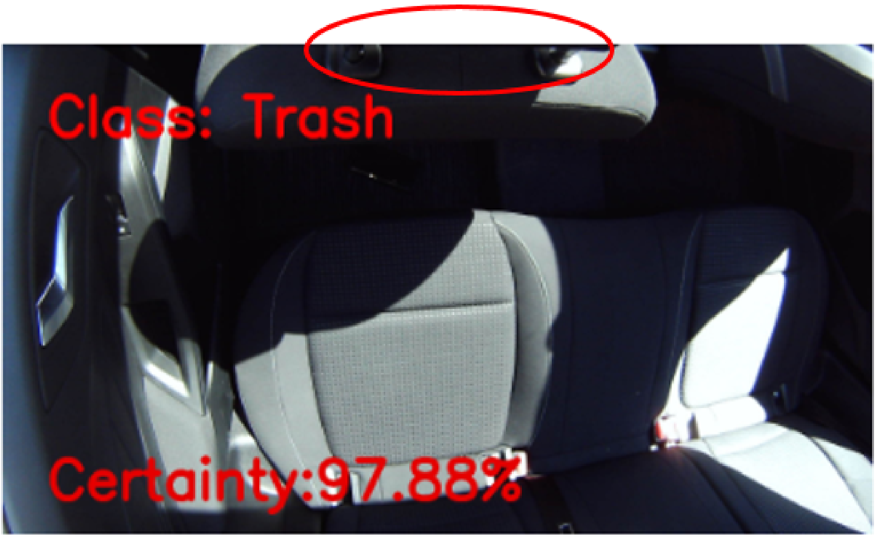

Figure 4.7. An empty seat is classified as trash due to the driver seat being pushed all the way back to its rearmost position

Apart from the driver seat case mentioned in the previous paragraph, the final classification is not affected by any other combination of front seat positions. However, the certainty of classification across all three classes gets slightly lower on occasion when the front passenger seat is pushed all the way back. Note the difference in figure 4.6 where the certainty between the third image from the top of the trash category is comparatively slightly lower. The case here however is not necessarily a source of error since the final classification category is not affected by the slight reduction of certainty.

The most influential sources of error with respect to the different forms of external variation were the changes in lighting and extreme shadows. Figure 4.8 depicts how a misclassification occurs when an object with a glare (a face up mobile phone in this case) is placed in an extremely bright spot of the rear seating area. The classification is incorrect because the camera focus in the location here is hindered due to the existence of shadows and the object is therefore not visible enough to be detected by the model. Furthermore, objects which have a lack of contrast with their neighbouring background are also not detectable from an image perspective and therefore the model usually classifies such images as being empty. Figure 4.9 presents such a scenario where a black mobile phone is placed on the far end of the floor area and therefore is left undetected. Possible ways to overcome the problem here would be the installation of a better camera or from an algorithmic perspective, applying contrast enhancement filters to obtain different forms of the same image and finally taking an average classification across all the forms of images.

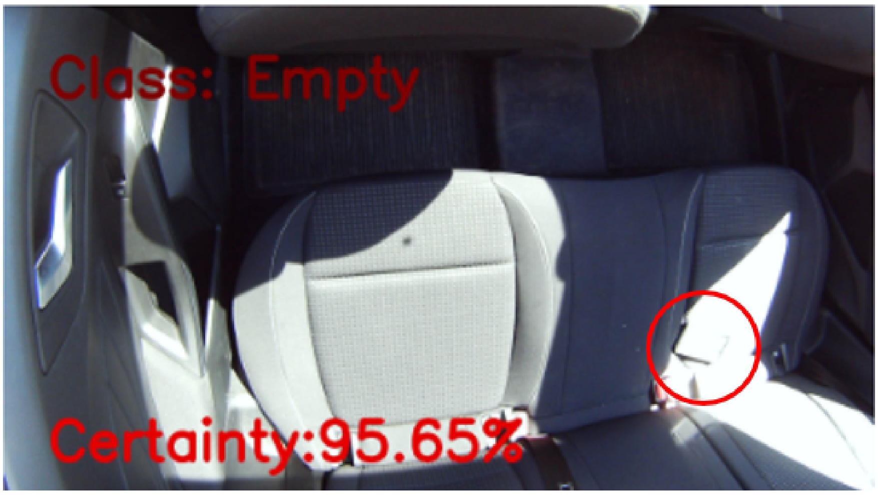

Figure 4.8. A seat containing a mobile phone is classified as empty due to its placement in a bright spot which hinders its detection

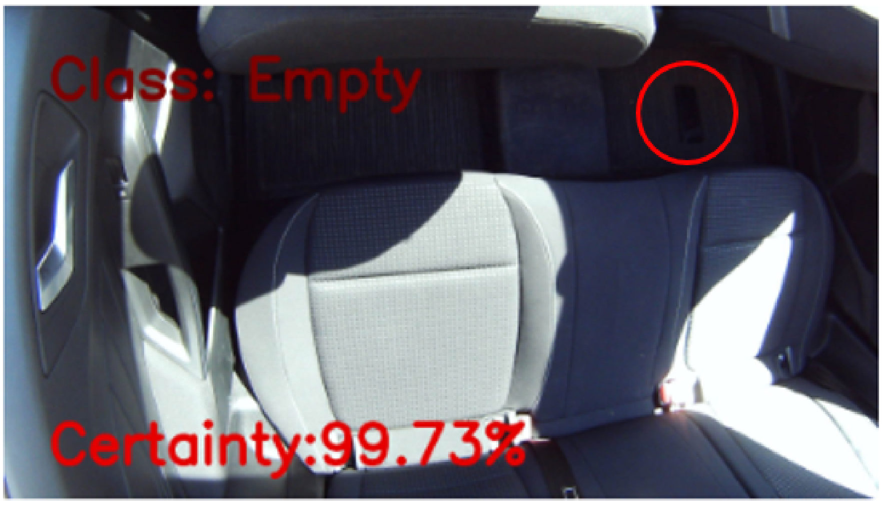

Figure 4.9. A black mobile phone placed on the far end of the floor area is undetected due to its lack of contrast with the neighbouring background. Hence its classified as empty

When an object is placed at a location corresponding to one of the furthest points of the camera where there is a lower contrast, marginal misclassifications were observed between the trash and valuable images. What it means is that although the model detects an object, it sometimes misclassifies it between the two image classes of trash and valuables. Figure 4.10 presents a good example of such a scenario where a small trash can placed far from the camera at a point with low object contrast has been marginally classified incorrect as a valuable. The certainty by which the model interpreted the image here to be trash was roughly around 48%. The error is one of the tougher ones to overcome. The best possible form of rectification would be to expand the image dataset (for training the model) further and proceeding with extensive training.

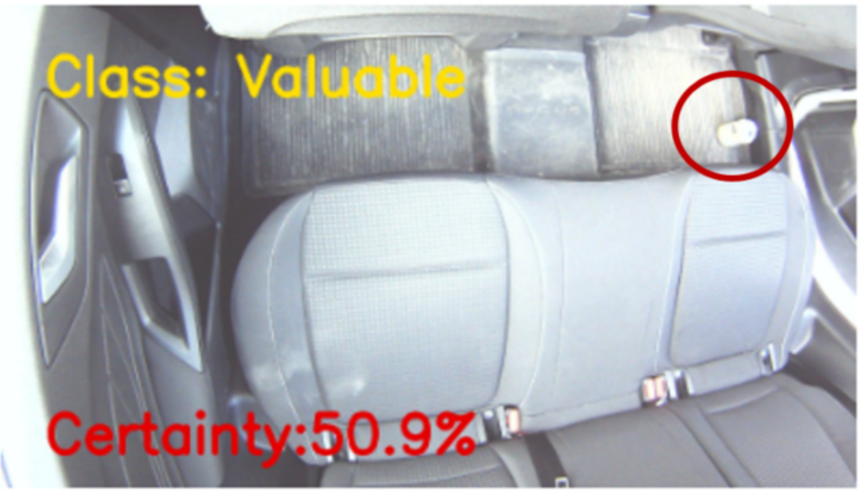

Figure 4.10. A trash can placed on the far end of the rear seating compartment marginally misclassified as a valuable

As an overview of the performance of the model, figure 4.11 shows how the accuracy drops with respect to the external sources of variation. The resulting classified images of the valuables in figure 4.6 depict how uniform lighting with less effects of shadows and clearly visible objects produce extremely high prediction certainties. On the other hand a collective analysis from all the sources of error explained primarily with regard to figures 4.7, 4.8, 4.9 and 4.10 show how the accuracy tends to drop due to the undesirable sources of error. Since figure 4.10 has been incorrectly classified marginally (in comparison to the other errors) we can deem the image here to be the error which degrades the accuracy the lowest in comparison to the other three main sources explained in the previous section. Similarly, the three forms of error at the lowest end of the arrow in figure 4.11 is arranged based on the certainty values from figures 4.7, 4.8 and 4.9 with the highest certainty of the false prediction at the bottom.

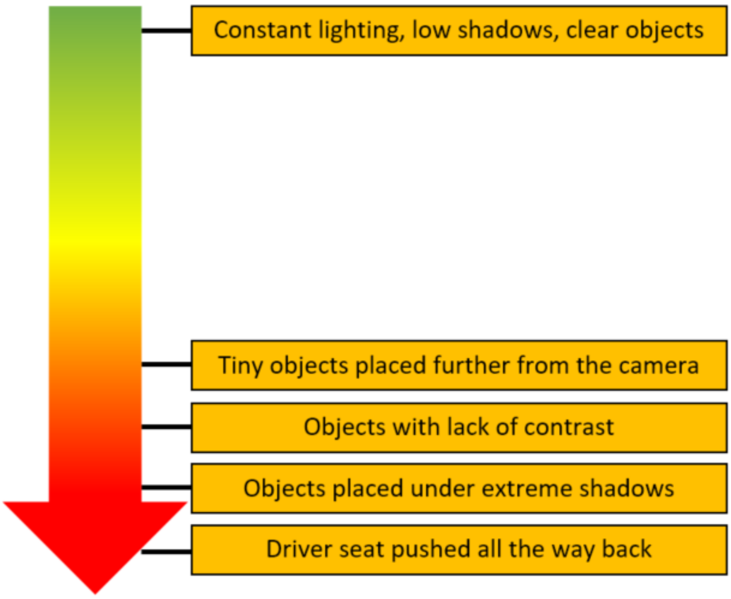
Constant lighting, low shadows, clear objects
Tiny objects placed further from the camera
Objects with lack of contrast
Objects placed under extreme shadows
Driver seat pushed all the way back

Figure 4.11. Illustration of how the accuracy drops with different sources of undesirable variation and image scenes. Note that the diagram here is not drawn to scale. However, the lower the item is down the length of the arrow, the higher its effect is in degrading the classification accuracy

4.3 Background Subtraction Results

As elaborated in the Methods chapter, the first step here was to form an estimate of the static background model of the rear seating area of the vehicle. A total of up to 117 empty car seat images were used to train and create the background model. The parameters which were tested out for the model include the background ratio, number of gaussian mixtures, the variance threshold and boolean parameter corresponding to shadow detection. Trial and error was used in adjusting the parameters and comparing the results for each combination.

However, it was unable to develop a good estimate of the background. The main reason behind this failure was the variation in terms of the seat positions and rear door states which vary from image to image. The model therefore was unable to generalize well. An attempt was made to feed images where the doors were cropped out. Yet, the accuracy suffered badly since the front seats were still a significant source of variance across the images of the empty class. Since the background model was poor, it was difficult to set a threshold that would separate clean images from those containing some form of unknown material. Therefore, the technique of background subtraction failed at the binary classification stage itself. Figure 4.12 depicts the result when the background mask was applied to two test images, one of which was empty while the other contained two drinking cans. The graph in figure 4.13 below illustrates the distribution of the scalar values obtained for all test images which had to be examined in order to apply the threshold. However, since there was no clear cut separation between the empty images and the rest of the images in the test batch, the method failed.

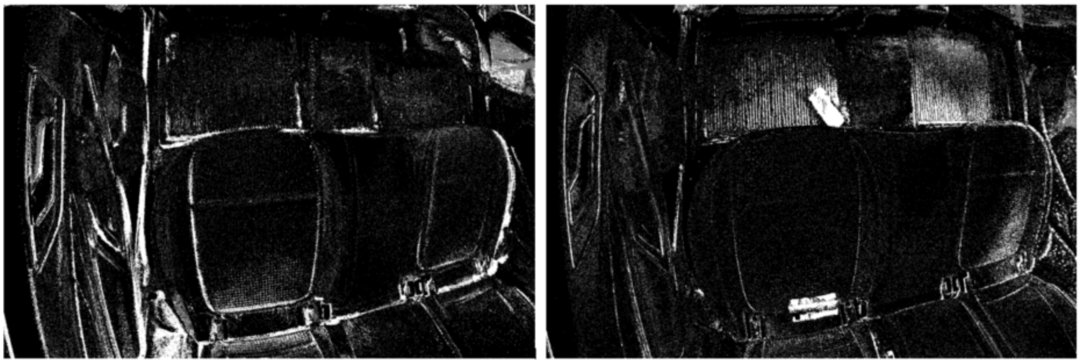

Figure 4.12. Results of background subtraction for a clean car seat image (left) and an image containing trash (right)

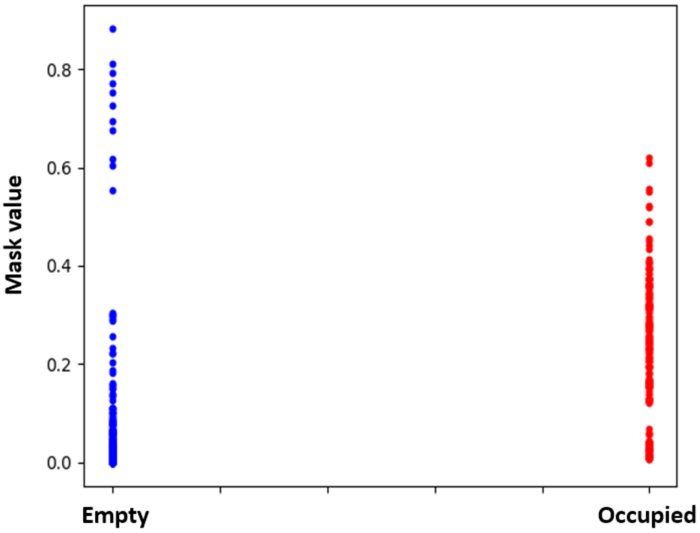

Figure 4.13. Scatter of the scalar values used for thresholding the images from the empty and non empty test set (binary classification). Note that there is no clear separation of the points along the vertical which therefore indicates that there it has failed to create a good estimate of the background

4.4 Execution speeds

Table 4.4 depicts the cycle times for different routines within the system functionality starting from the image capture from the camera module all the way up to the model classification stage. The values here were obtained during the testing phase and in some cases, were highly dependent on how good the network connection was. The graph in figure 4.14 illustrates the trends of the different execution times for a better visual comparison. Note that the extra time column in both the table and the graph refers to the additional time range which each process would take. Hence by adding this value to the minimum time we could obtain the maximum time that each process would consume while the system functions.

Table 4.4. Total cycle times for the main processes of the detection system. Note that the times here are given for the processing of a single image. The total cycle time is given for the classification done using the CNN.

Process	Min time(s)	Extra time(s)
Image capture and transmission to Raspberry Pi	0.46	0.52
Image transmission to server (via SCP)	0.2	4
CNN classifier	5	7
Total cycle time	5.66	11.52
Background subtraction	0.05	0.14

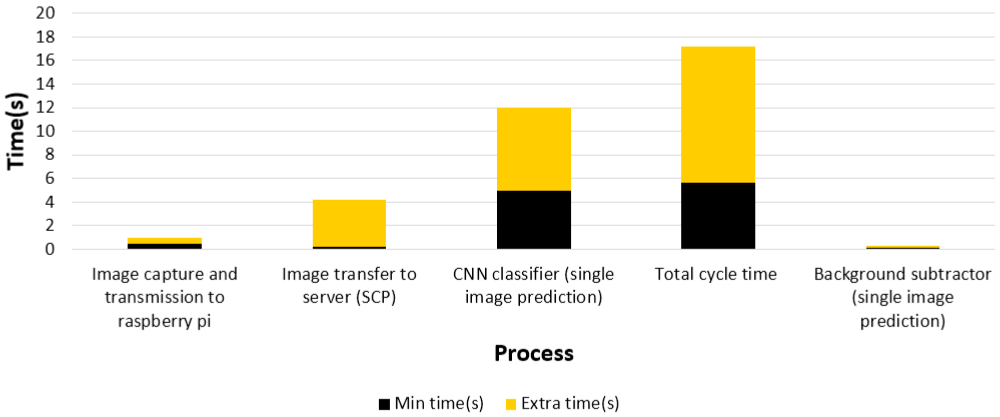

Figure 4.14. Comparison of the approximate execution times for the main processes of the detection system corresponding to the values from table 4.3

4.5 Comparison to previous studies

Going back to the state of the art which was explained in detailed in chapter 2, we saw that most of the research work done in the past was focused on applying modified vision based detection algorithms in order to identify and classify different types of trash. In comparison, the focus in the study here was slightly different as priority was given to identify where a detected object is trash or a customer belonging. The reason behind doing so is merely based upon the objective of the research which is to distinguish between the trash and valuable categories since the ultimate response intended from the detection system in application is dependent upon the categorization. Knowing the type of trash or even the type of valuable is therefore of lesser importance. In addition, one feature which is lacking in this study is the detection of the pose and position of identified objects as it merely does a classification. Once again however, its absence is not a crucial drawback given what the system is intended for primarily because the removal of any unknown object, once detected, will not be automated (it practically requires a manual response).

The trash dataset created by Thung and Yang(2016) was utilized in a number of previous research applications for the purpose of training their models. Since the focus area with respect to the ultimate objectives were similar in the related research studies, the dataset worked out well. However, the dataset here was not applicable to this study primarily because of the specificity of the application. Simply put, this research focused on a unique case of detecting trash and valuables in passenger vehicles whereas most of the previous studies were aimed at detecting different categories of trash for the purpose of separation and recycling. Hence the dataset created here could be specifically utilized in future research studies which involve autonomous trash detection in transportation media. One possible drawback here would be the lack of different types of trash in the images in comparison to the datasets of the previous studies in the field which explore a much wider range of trash categories both in terms of material and type. Furthermore, since the images of the previous studies were generalized, there was not particularly any form of bias compared to the image dataset in this study which had images containing items handpicked by me and placed inside the vehicle.

We observed that most of the models created for the purpose of trash detection in previous studies were also CNNs since it is based on image processing. In comparison however, the model used here is a simpler CNN since the VGG16 is not as deep and complex as the different ResNet CNNs for instance which were actually a popular choice among previous research models. The complexity in the neural network therefore may well be the reason as to why some models suffered to produce a good prediction accuracy because the tuning process becomes more intensive as the levels of the network expands. Moreover, the model was also able to perform under different sources of external variation which include lighting changes, shadows, rear door states of the vehicle and its front seat positions. Such variation was not evident in previous applications since the background was static and uniform in most cases.

In comparison to previous studies which were focused on creating trash classification models, the final accuracy obtained in this research was quite as the final prediction accuracy on the test set was 91.43%. Thung and Yang (2008) managed to obtain an accuracy of 22% using their CNN model and an accuracy of 63% using SIFT features across six categories of trash. Adedeji and Wang (2019) managed to obtain an increased accuracy of 63% using a fine-tuned ResNet-50 CNN model on the same trash dataset which was employed in the study conducted by Thung and Yang (2008). In addition, the research conducted by Salimi et al. (2018) utilized Viola-Jones detection for their trash classifier and obtained a final prediction accuracy of 82.7%. Bai et al. (2018) conducted a study to classify trash across five different categories using a ResNet-34 model and ended up with an average classification accuracy of 87.26%. The results from the previous studies are henceforth summarized in table 4.5 below

Table 4.5. Comparison of the classification accuracies of the trash classification models from previous studies

Authors (reference)	Type of prediction model	Classification accuracy
Thung and Yang (2008)	Artificial CNN	22%
Adedeji and Wang (2019)	ResNet-50	87%
Bai et al. (2018)	ResNet-34	87.26%
Thung and Yang (2008)	SIFT and SVM	63%
Salimi et al. (2018)	Haar-Cascade and Viola-Jones detection	82.7%

5. Conclusions

5.1 Summary of the study

The study conducted here was aimed at proving the viability of using a vision based detection system to detect any forms of trash and customer belongings which have been left behind in the rear seating area of shared vehicle cabins.

In order to capture the state of the seating area, an in-car camera module was developed and integrated to the existing power setup of the Aalto research vehicle. Although the setup was not a final solution, it proved to be compact, cheap and moreover, successful in accomplishing the task of efficiently capturing an image frame upon receiving the command to do so and then transmitting it to the remote server for classification.

Two algorithmic models were proposed to execute the function of image classification into one of three possible categories (empty, trash and valuables). The first one was a type of convolutional neural network (CNN) known as VGG16. The model, once fine tuned, turned out to produce a high accuracy of 91.43% on the batch of test images. However, multiple sources of error were identified primarily due to the presence of external variation such as extreme shadows and the driver seat position in addition to misclassifications which were evident due to a lack of contrast between an object and its neighbouring background. The second type of algorithm proposed was background subtraction, but the method failed due to its inability to form a stable background estimate due to the high variation present in the empty images especially with respect to the changes in front seat positions from one to another.

In addition, an expansive dataset was created and compiled as proposed in the objectives of this study. It was annotated and organized into train, test and validation categories and further into empty, trash and valuable classes. The dataset will be made available in an open source platform for related future research work in the field.

To conclude, the viability of utilizing machine vision to detect trash and valuables in shared vehicles was proven to work well for an individual type of vehicle model. Specifics were given on the patterns of certain false predictions, what causes the drop in accuracy for the images and what could be done to optimize and overcome the issues of misclassifications in further studies.

5.2 Other limitations

Apart from the effects of accuracy which were observed due to the presence of external forms of variation that were elaborated in the previous section as well as in the results, there were two other categories of limitation in this research. The first of which was with respect to the detection area of the camera module. It was clear that only the rear seating area of the vehicle cabin was captured by the camera in its current location. Hence, the front passenger seat as well as the driver seat were unaccounted for in this study. In addition, any trash item or customer valuable left behind under the front seats or the door pockets will not be detected by the in-car camera system.

The second type of limitation is related to the system functionality itself. First and foremost, the trash and valuable image classes were considered to be mutually exclusive. Thereby, the model was not trained for the scenario of an image having both trash and valuables. Furthermore, any abnormal forms of camera obstructions, such as a person's hand or head covering the vision of the camera, were also unaccounted for when training the model. In relation to the types of image data, it was already pointed out that the trash class only included images of cans and bottles. In addition, cases involving certain scratch marks on the seating area or even spills were ignored and therefore such an image if produced will yield an unpredictable classification from the prediction model in its current form.

5.3 Future work

Given the development of the vision based detection system in this research, it is evident that it is not yet ready for the use case application. Hence the section here will discuss briefly on the future work that needs to be considered in realizing the dream of having such a system in place for actual shared vehicle fleets.

First and foremost, it is necessary to obtain data from an actual shared vehicle in operation since such data would be more in line with the final practical application. Moreover, the image data obtained by such means will be more general, unbiased and will have a good spread of external variation especially with respect to various forms of lighting and shadows as opposed to the current batch of images which were taken by parking the vehicle in a limited number of parking spots. In order to obtain such image data, it would be useful to consider a change of camera brand to a higher quality one as well as its installation in better locations of the vehicle in order to provide a wider detection radius. Figure 5.1 depicts the current location of the camera installation. Figure 5.2 presents two possible configurations once the camera is installed into an actual shared vehicle in operation. Note that the locations of the camera (denoted with the red dots)in both figures 5.1 and 5.2 are approximated. Furthermore, the model of the car used for the figure does not represent the actual research car used for this study.

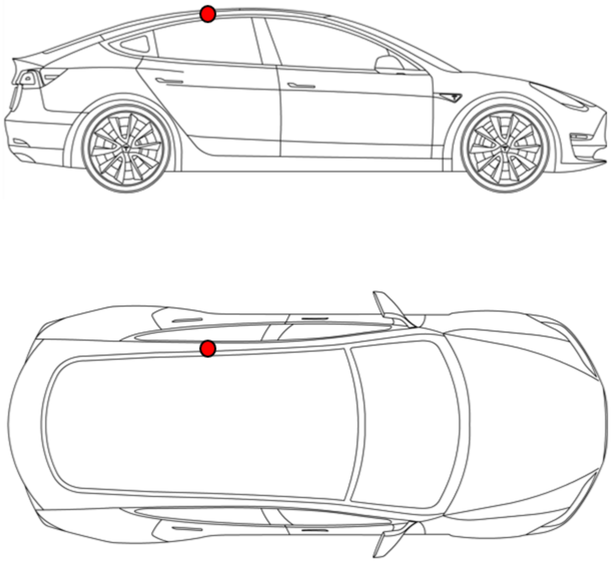

Figure 5.1. Approximate location of the current camera installation as denoted by the red dot.

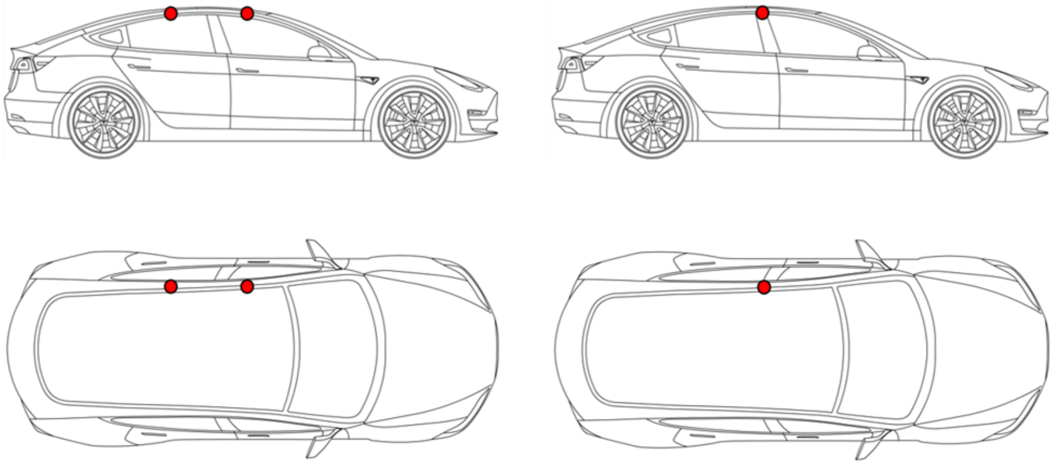

Figure 5.2. Two possible future camera installation locations in the vehicle as denoted by the red dots.

The algorithm development would be an area to be continuously worked on in order to yield higher accuracy values from the prediction model under different forms of external variation. Once the model is able to generalize more reliably between the output classes, it is necessary to develop the algorithm further to be able to produce a cleanliness level preferable on a predetermined scale in order to determine the type of response. Figure 5.3 illustrates an example of such a scale.

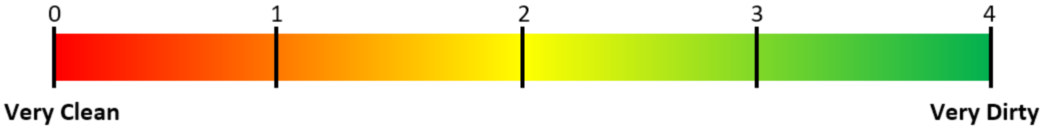

Figure 5.3. Possible option to rate the level of cleanliness in the vehicle. In the scale here, there would be five possible classifications which correspond to how clean or dirty the vehicle interior is

Last but not least, two more functionalities will be integrated to the vision based detection system. The first one would be to integrate a person detection algorithm. As the name suggests, the idea would be to determine if there is a person on the seat at the time of image capture. The purpose of such an addition would be to overcome privacy issues with regard to data transmission. Hence, if a person has been detected, the image will not be transmitted to the server and will instead be disposed of. The algorithm here would therefore be run within the onboard processor of the vehicle (the Raspberry Pi module in this case). The second integration would be the addition of an odour detection module to the detection system primarily for the purpose of achieving enhanced levels of cleanliness and consequently, customer attraction towards the use of shared vehicles.

Bibliography

Adedeji, Olugboja and Zenghui Wang (2019). "Intelligent Waste Classification System Using Deep Learning Convolutional Neural Network". In: *Procedia Manufacturing* 35, pp. 607–612. URL: http://www.sciencedirect.com/science/article/pii/S2351978919307231.

Bai, J. et al. (2018). "Deep Learning Based Robot for Automatically Picking Up Garbage on the Grass". In: *IEEE Transactions on Consumer Electronics* 64.3, pp. 382–389.

Balakrishnan, Ramalingam et al. (Dec. 2018). "Cascaded Machine-Learning Technique for Debris Classification in Floor-Cleaning Robot Application". In: *Applied Sciences* 8, p. 2649. DOI: 10.3390/app8122649.

Batinic, Bojan et al. (2011). "Using ANN model to determine future waste characteristics in order to achieve specific waste management targets-case study of Serbia". In: *Journal of Scientific & Industrial Research* 70, pp. 513–518.

Deng, J et al. (2009). "ImageNet: A Large-Scale Hierarchical Image Databases". In: *CVPR09*.

Donovan, Jay (Sept. 2016). *Auto-Trash sorts garbage automatically at the Techcrunch Disrupt Hackathon*. https://techcrunch.com/2016/09/13/auto-trash-sorts-garbage-automatically-at-the-techcrunch-disrupt-hackathon/.

ELP Webcam Wide angle 170 degree Fisheye Lens Camera USB Full HD 1080P 30fps Web Camera for Linux/Windows/android (Sept. 2015). https://www.amazon.co.uk/ELP-Webcam-Fisheye-Windows-android/dp/B015PCAVZG.

Fritzche, Martin et al. (Feb. 2003). *Process for monitoring the internal space of a vehicle, as well as a vehicle with at least one camera within the vehicle cabin.*

Fulton, Michael et al. (2018). "Robotic Detection of Marine Litter Using Deep Visual Detection Models". In: *CoRR* abs/1804.01079. arXiv: 1804.01079. URL: http://arxiv.org/abs/1804.01079.

Girshick, Ross B. (2015). "Fast R-CNN". In: *CoRR* abs/1504.08083. arXiv: 1504.08083. URL: http://arxiv.org/abs/1504.08083.

He, Kaiming et al. (2015). “Deep Residual Learning for Image Recognition”. In: *CoRR* abs/1512.03385. arXiv: 1512.03385. URL: http://arxiv.org/abs/1512.03385.

Hyundai Motor to present results of in-car cleanliness study (n.d.). https://www.hyundai.news/eu/brand/hyundai-motor-to-present-results-of-in-car-cleanliness-study/.

Japan Agency for Marine Earth Science and Technology,Deep-sea Debris Database. (N.d.). http://www.godac.jamstec.go.jp/catalog/dsdebris/e/.

Li, L and F Wang (2007). “Intelligent Vehicle Vision Systems”. In: *Advanced Motion Control and Sensing for Intelligent Vehicles*.

Lin, Tsung-Yi et al. (2014). “Microsoft COCO: Common Objects in Context”. In: *CoRR* abs/1405.0312. arXiv: 1405.0312. URL: http://arxiv.org/abs/1405.0312.

Liu, Wei et al. (2015). “SSD: Single Shot MultiBox Detector”. In: *CoRR* abs/1512.02325. arXiv: 1512.02325. URL: http://arxiv.org/abs/1512.02325.

Pai, Aravind (Feb. 2020). *CNN vs. RNN vs. ANN – Analyzing 3 Types of Neural Networks in Deep Learning*. https://www.analyticsvidhya.com/blog/2020/02/cnn-vs-rnn-vs-mlp-analyzing-3-types-of-neural-networks-in-deep-learning/.

Ramya, V and B Palanaippan (Apr. 2012). “Embedded Technology for vehicle cabin safety Monitoring and Alerting System”. In: *International Journal of Computer Science, Engineering and Applications (IJCSEA)* 2.

Raspberry Pi 3 B (n.d.). https://www.elektor.com/raspberry-pi-3-model-b.

Redmon, J. et al. (2016). “You Only Look Once: Unified, Real-Time Object Detection”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788.

Redmon, Joseph and Ali Farhadi (2016). “YOLO9000: Better, Faster, Stronger”. In: *CoRR* abs/1612.08242. arXiv: 1612.08242. URL: http://arxiv.org/abs/1612.08242.

Ren, S. et al. (2017). “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6, pp. 1137–1149.

Salimi, Irfan, Bima Sena Bayu Dewantara, and Iwan Kurnianto Wibowo (2018). “Visual-based trash detection and classification system for smart trash bin robot”. In: *International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*.

Sandler, Mark et al. (2018). “Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation”. In: *CoRR* abs/1801.04381. arXiv: 1801.04381. URL: http://arxiv.org/abs/1801.04381.

Schofield, Kenneth et al. (Dec. 2017). *Cabin monitoring system for a vehicle*.

Thung, Gary and Mingxiang Yang (2016). “Classification of trash for recyclability status”. In:

Uusi LED Strip 5m 24V IP20 White 2500K CRI98 (n.d.). https://fi.rsdelivers.com/product/powerled/b5-11-28-2-70-f8-20-98ra/led-strip-5m-24v-ip20-white-2500k-cri98/1845166.

Wang, Ying and Xu Zhang (Jan. 2018). "Autonomous garbage detection for intelligent urban management". In: *MATEC Web of Conferences* 232, p. 01056. DOI: 10.1051/matecconf/201823201056.

Yasuhiro, Fuchikawa et al. (2005). "Development of a Vision System for an Outdoor Service Robot to Collect Trash on Streets." In: pp. 100–105.

Zeiler, Matthew D. and Rob Fergus (2013). "Visualizing and Understanding Convolutional Networks". In: *CoRR* abs/1311.2901. arXiv: 1311.2901. URL: http://arxiv.org/abs/1311.2901.

Zhihong, Chen et al. (2017). "A Vision-based Robotic Grasping System Using Deep Learning for Garbage Sorting". In: *Beijing institute of precision mechanical and electrical control equipment*.

A. File of certainties of the test batch: correct predictions

Correct Predictions

Index	Empty	Trash	Valuable
0	0.001	0.000	0.999
1	0.000	0.001	0.999
2	0.000	0.926	0.074
3	0.000	0.000	1.000
4	0.000	0.000	1.000
5	0.000	1.000	0.000
6	0.992	0.001	0.007
7	0.486	0.424	0.090
8	1.000	0.000	0.000
9	0.000	0.000	1.000
10	0.000	0.996	0.004
11	1.000	0.000	0.000
13	0.000	0.000	1.000
14	1.000	0.000	0.000
15	0.000	1.000	0.000
16	0.000	1.000	0.000
17	1.000	0.000	0.000
18	0.000	1.000	0.000
19	0.000	1.000	0.000
20	0.997	0.003	0.000
21	0.003	0.000	0.997
22	0.000	1.000	0.000
23	0.000	0.000	1.000
24	0.000	0.000	1.000
25	0.000	1.000	0.000
26	0.000	1.000	0.000
27	0.997	0.001	0.003
30	0.000	1.000	0.000
31	0.000	0.000	1.000
32	0.128	0.000	0.872

33	0.000	1.000	0.000
34	0.000	0.000	0.999
35	0.003	0.000	0.997
37	1.000	0.000	0.000
38	0.000	0.000	1.000
40	0.000	0.000	1.000
41	0.000	0.981	0.018
42	1.000	0.000	0.000
43	1.000	0.000	0.000
44	0.000	0.000	1.000
45	0.948	0.000	0.052
46	0.982	0.018	0.000
48	0.000	0.000	1.000
49	0.975	0.024	0.000
51	1.000	0.000	0.000
52	0.329	0.654	0.017
53	0.520	0.479	0.001
54	0.018	0.000	0.982
55	1.000	0.000	0.000
56	1.000	0.000	0.000
57	1.000	0.000	0.000
58	1.000	0.000	0.000
59	1.000	0.000	0.000
60	1.000	0.000	0.000
61	0.025	0.975	0.000
62	0.000	0.884	0.116
63	0.000	0.000	1.000
64	0.010	0.974	0.016
65	1.000	0.000	0.000
66	0.003	0.007	0.990
67	0.000	0.996	0.004
68	0.998	0.002	0.001
69	0.000	0.799	0.201
70	1.000	0.000	0.000
71	0.003	0.979	0.018
72	0.060	0.009	0.931
73	0.000	0.001	0.999
74	0.001	0.000	0.999
75	0.140	0.860	0.000
76	0.004	0.989	0.007
77	1.000	0.000	0.000
78	1.000	0.000	0.000

79	0.001	0.003	0.995
81	1.000	0.000	0.000
83	0.512	0.488	0.000
84	0.000	0.000	1.000
85	1.000	0.000	0.000
86	0.000	0.000	1.000
87	0.000	0.976	0.024
88	0.000	1.000	0.000
89	1.000	0.000	0.000
90	1.000	0.000	0.000
91	0.986	0.014	0.000
92	0.000	0.000	1.000
93	1.000	0.000	0.000
94	1.000	0.000	0.000
95	1.000	0.000	0.000
97	1.000	0.000	0.000
98	0.993	0.006	0.002
99	1.000	0.000	0.000
100	0.000	0.000	1.000
102	0.999	0.001	0.000
103	0.000	0.000	1.000
104	1.000	0.000	0.000
105	0.000	1.000	0.000
106	0.000	1.000	0.000
107	0.000	1.000	0.000
108	0.000	1.000	0.000
109	1.000	0.000	0.000
110	0.000	1.000	0.000
111	1.000	0.000	0.000
112	0.999	0.001	0.000
113	0.000	1.000	0.000
114	1.000	0.000	0.000
115	1.000	0.000	0.000
116	0.004	0.764	0.232
117	1.000	0.000	0.000
118	0.000	0.000	1.000
119	0.001	0.596	0.403
120	1.000	0.000	0.000
121	1.000	0.000	0.000
122	0.000	0.000	1.000
123	0.981	0.019	0.000
124	0.013	0.006	0.981

125	0.000	0.000	1.000
126	0.000	0.000	1.000
127	0.000	0.000	1.000
128	0.000	1.000	0.000
129	0.000	1.000	0.000
130	0.944	0.056	0.000
131	0.000	0.000	1.000
132	0.000	0.987	0.013
133	1.000	0.000	0.000
135	0.000	1.000	0.000
136	0.024	0.000	0.976
137	0.000	0.000	1.000
138	1.000	0.000	0.000
139	0.000	0.000	1.000

B. File of certainties of the test batch: false predictions

False Predictions

Index	Empty	Trash	Valuable
12	0.057	0.932	0.011
28	0.003	0.488	0.509
29	0.000	0.520	0.480
36	0.985	0.015	0.000
39	0.343	0.000	0.657
47	0.199	0.000	0.801
50	0.100	0.000	0.900
80	0.999	0.001	0.000
82	0.000	0.065	0.935
96	0.003	0.000	0.997
101	0.059	0.941	0.000
134	0.963	0.037	0.000