National College of Ireland

A Deep Learning Approach to Vehicle Make and Model Recognition with Specification Matching

MSc Research Project
Data Analytics

Samuel Biwei Tanga
Student ID: x20167784

School of Computing
National College of Ireland

Supervisor: Dr. Martin Alain

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Samuel Biwei Tanga
Student ID:	20167784
Programme:	Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Dr. Martin Alain
Submission Due Date:	31/1/2022
Project Title:	A Deep Learning Approach to Vehicle Make and Model Recognition with Specification Matching
Word Count:	8142
Page Count	23

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Samuel Biwei Tanga
Date:	16/12/2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

A Deep Learning Approach to Vehicle Make and Model Recognition with Specification Matching

Samuel Biwei Tanga
X20167784

Abstract

Vehicle Make and Model Recognition (VMMR) has risen to become a highly significant research area within the automobile sector in recent years. Specifically, it is beneficial in traffic analysis, vehicle analysis, and detection of crimes associated with vehicles, among other applications. A VMMR system with great accuracy and the ability to create dynamic real-time results helps save resources. For this recognition and classification task, a system that is sufficiently capable to handle the ambiguity and multiplicity that occurs between various makes and models needs to be implemented. This project aims to develop a vehicle model recognition system that can appropriately recognise and perform classification of vehicles into appropriate make and model classes they belong to with an attempt to match the accurately recognized vehicles with their standardized specifications. The author proposes three different models in order to address this challenge and develop a system that is more adaptive and responsive than previously proposed methods. The system is developed with these models (MobileNet-v2, ResNet50, VGG16) by training them with the train sets and evaluating the models by their accuracy and computational time. ResNet-50 outperforms the other models considering the accuracy and minimal trade-off with computational time. The ResNet-50 model is then incorporated into a GUI application which can be used for real life applications, this to displays the importance of the system to the automobile industry and intelligent transport systems.

1 Introduction

Vehicle make and model recognition (VMMR) is one of the trending areas of research in the computer vision field because of the complexities involved. VMMR is applied in many real-world scenarios such as in toll payment systems, traffic monitoring systems, identification of vehicles in crime scenes, autonomous vehicles, intelligent parking systems and many more applications. The idea behind VMMR especially in crime and surveillance system is identification therefore, the accuracy rate of the VMMR system must be very high. There are many factors that could affect the proper recognition of vehicles, such as lighting conditions and surrounding objects in the environment. However, the most significant problem that affects the proper recognition of vehicle make and model is the high inter-class and intra-class similarity of the different makes and models of vehicles. Day by day there is an increase in the different vehicle makes and differentiating these vehicle makes from one another is one problem. Even though there is wide variation, there is also a close similarity, it gets more difficult when identifying models of cars from the same manufacturer (i.e., having the same make) because of the very high similarity between the features of the various models. These differences cannot even be made out by human beings who do not have a prior knowledge in

the domain. Thus, to classify intraclass entities, fine-grained vehicle identification requires more innovative and discriminating features. As a result, there have been a number of studies on fine-grained vehicle identification in intelligent transport systems because of this intraclass variation.

There are two main techniques to identifying the make and model of a vehicle. It is possible to determine a vehicle's make and model by detecting the vehicle's logo (Wah et al, 2011). When there is no prior information on the car, this is a useful technique to determine the make. Furthermore, there may be times when a vehicle's logo is missing or has faded. Because the logo is just a tiny part of the car, it is indeed possible that the images acquired don't display it well. This strategy is not the most effective in these kinds of circumstances. A vehicle's make and manufacturer are all that can be established using this technique, even if it has a conspicuous logo. To identify the vehicle make and model, the alternative method makes use of the back-view or front-view images of the vehicle. It may sometimes include extracting particular vehicle parts and identifying the vehicle's make and model in reference to the area from which the components were taken. Most of the time, the components to be extracted are picked manually, and afterwards the detection algorithm is trained using supervised learning. This may be a huge problem since the accessibility and collecting of this highly precise data is costly. This reduces the practical application of this to real-life large-scale problems. Furthermore, only the make and model of a vehicle may be determined using any of these methods. The ability to match generic vehicle parameters, such as year of manufacture to particular vehicle makes and models is necessary because a potential vehicle buyer may see a car on the street and want to know the make, model and year of manufacture without asking around or doing so much research. This serves as an introduction to the research issue that will be addressed in this study. This issue being it is possible for a deep learning model to accurately recognize and classify vehicles based on make and model and also if there is a possibility of matching these classified vehicles with their generic specifications

In literature, VMMR approaches are divided into three primary categories: feature-based, appearance-based, and model-based. Vehicle models may be identified using global or local invariant features. Thus, the dependability of features is critical to their performance. Appearance-based approaches define automobiles based on its intrinsic features, such as their shape, geometry and dimensions. The adaptive model, the approximation model, and the 3D model are all examples of vehicle model-based recognition (Jamil et al., 2020). In this study, the author has used a feature-based technique that is both accurate and has computational efficiency. There are three main steps in this process for a VMMR system: vehicle detection, feature extraction and classification. This is carried out using a Convolution Neural Network (CNN). The input dataset consists of different 2-dimensional images. These input images are then passed on to the convolution layers that contain several filters. For every filter, a 2-D feature matrix is created. The convolution and pooling activities are where the extraction and representation occurs. Then lastly, the fully connected layer is where classification occurs to produce a prediction in the output layer. These results are then evaluated and compared against existing research work. The final model is incorporated into a mobile application and the computational efficiency is compared against other mobile models.

This report comprises of five sections and is structured as follows; the introduction, which is chapter one gives an overview of the whole report. It introduces the research project, the background, the research question and also the proffered solution. Chapter two - literature review, delves into similar research work done by other researchers to acknowledge the work done so far, while also highlighting gaps that this project needs to fill. Chapter three - the research methodology details what data analytics methods the research has taken to implement the solution. Chapter four – the design specification discusses the system architecture and also the steps taken to achieve the design. Chapter five – Implementation entails discussion of the different models that were built and justification as to why these models were chosen and also parameters and settings used. Chapter six - show cases the results of the various experiments carried out while also interpreting the results and also discusses the results compared to previously critiqued research. Chapter seven – Conclusion gives a summary of the research, highlighting major points in the research and also makes suggestions for future work.

2 Related Work

The identification of vehicle types has been the subject of previous research. When compared to the purpose of the VMMR, they are inadequate in terms of providing specific vehicle information and are restricted in terms of providing suitable accuracy for particular applications, among other limitations. The fine-grained classification challenge of recognizing the make and model of a car has been addressed by several studies as a means of extracting more detailed and precise data from vehicles. VMMR has been the subject of a great deal of research, but it remains a difficult issue to solve because of the enormous range of visual appearances, as well as the variations in lighting conditions and viewing angles, among other factors. The foundation for VMMR was laid by the work of (Petrovic and Cootes, 2004). To accomplish this goal, they used the front-view images of vehicles in their research. With the help of a nearest neighbour classifier, the final classification process was carried out based on the size and placement of the registration plate. The purpose of this literature review is to examine prior studies in this area and highlight the advancement of research done in this domain. Additionally, this study seeks for areas that have been overlooked in earlier studies in order to determine gaps to be filled. Research work in this field have been broadly divided into three areas; vehicle detection, vehicle type recognition and vehicle make and model recognition.

2.1 Vehicle Detection

Detecting vehicles is the reference point for issues concerning vehicle classification. Vehicle detection identifies the existence of a vehicle in an image and extracts the area of interest from the image in order to exclude the surrounding environment from consideration. There are times when using the whole vehicle as an input to the model is inefficient, therefore information is removed and utilized from just the required regions (such as the rear and front lights, the bumper, the registration plate, and so on). Enhancing the effectiveness of the vehicle classification system is achieved by removing undesired or unnecessary car parts.

Background elimination was used by Faro, Giordano, and Spampinato (2011) in order to identify probable parts in images. This technique was used to remove any object that did not resemble a car. However, Chen et al. (2015) suggest that the extraction of features using the approach of background elimination is quite unreliable, even when other background images and surrounding elements are removed from consideration. Despite the fact that it may enhance accuracy, it is incapable of dealing with noise, rapid illumination alterations, or rapid movements occurring in the background. A further disadvantage of extracting is that the precision of the extraction declines with delayed and quick motions and may be incapable of obtaining all critical motion components entirely. Among several difficulties in background elimination is its failure to handle sophisticated situations involving a large number of objects that move continuously. As a result, they proposed employing a Speed-Up Robust Features (SURF) for both the original and mirrored images, as well as computing correlations among SURF features in order to determine horizontally symmetrical points. In this step, a midline is identified, and each group of symmetrical SURF points together with the midline indicates a potential vehicle. This strategy resolves the basic problem of the background elimination technique. This produces far more precise and comprehensive points of extraction, which is particularly beneficial in circumstances with constant motion or imagery with a great deal of background noise. The downside of this method is the high level of computational requirements (Chen, Zhang and Zhao 2020).

2.2 Vehicle Type Recognition

Efforts on Vehicle Type Recognition (VTR) are aimed at classifying vehicles into high-level groups such as vans and mini-vans, trucks, sedans, buses, minibuses, and other vehicles, among others. There is also no way to tell the make and models of the vehicles in this type of classification. The use of an autonomous VTR system is beneficial in operations like automated toll booths, traffic evaluations and analysis, and other intelligent transport system tasks. Research into vision-based VTR methods has attracted a great deal of interest in recent years, thanks to the advancement of computer vision capabilities and the proliferation of data from traffic surveillance systems. For the purpose of vehicle type recognition, different researchers have classified vehicles into different classes dependent on the approach they used. Researchers like Chen, Ellis and Velastin (2016) combined Random Forest (RF) and Support Vector machines to classify vehicles into four broad categories: vans, buses, cars, and cycles. This was a big shift from what had been earlier implemented by Ma, Eric and Grimson (2005) who represented vehicles using edge points and adjusted SIFT descriptors, and afterwards customized a cluster model which classified the vehicles into two groups: cars vs minivans and sedans against taxis, respectively. Counting applications benefit from these high-level classifications, while security systems are severely hampered by it. In order to gather more specific vehicle details, Kumar and Sivanandam (2012) employed Haar cascade classifiers to identify vehicle logo location and grouped cars into four classes with the use of a SVM classifier. With the use of mesh lines, Leotta and Mundy (2011) established a more extensive modelled recognition for vehicles, rather than the classification of recognized vehicles into four distinct classes as described by the previous authors. This method has the capability of detecting distinct parts of vehicles and drawing mesh-lines over

the recognized parts in order to provide a structurally visualised in a meshed form. The displayed structure served as the foundation for the classification of the vehicles.

The researchers have essentially employed the same form of modelling representation, since both models entail visual depiction of a vehicle using lines. This assists in the identification process because the approaches involve utilisation of geometry, this is a highly beneficial factor for deciding what class a vehicle falls under. In contrast to a sedan, the range of measurements for a van is remarkably distinguishable. In comparison to the robust classification system of Leotta and Mundy (2011), other researchers classified vehicles into four categories, which is not a suitable indication of VTR, especially because vehicles can be classified into much more than the four groups used by Chen, Ellis, and Velastin (2016). In this respect, Leotta and Mundy (2011) provide a highly dynamic classification approach, given the fact that there is flexibility to classify as well as label classes which is fully reliant on exclusive parts, the overall dimension, as well as the distances between the regions of interest. When it comes to VTR, this model-based method has been frequently adopted. For example, a Bayesian Network (BN) was built by Kafai and Bhanu (2012) for extracting vehicle parts from region of interest, so as to address the problem of recognising and the appropriate classification of different vehicle types. Additionally, the model takes into account the vehicle's height, width, and length. Several additional measurements, such as the distance between taillights, were also recorded. A 0.96 accuracy rate was achieved when the model was evaluated on a small data sample. The model effectively recognised vehicle types across four groups of vehicles: sedans, trucks, buses, and vans. Wang et al. (2014) developed a voting method that classified cars into three categories based on Euclidean edge distance and geometric factors. 0.91 of the time, the algorithm was accurate.

Overall, at the stage of VTR the observation is that vehicles are only weakly classified into vague groupings that do not precisely distinguish them by makes or models, and that this classification type is inadequate. However, although model-based approaches are useful for recognising vehicle types, the need to classify cars in a way that goes beyond the broad classification centred upon their geometry and dimensions. Vehicles cannot be distinguished by their make and model because the three-dimensional models created at this stage are not intelligent enough and built to that level of sophistication. There is also a considerable degree of similarity across the various vehicle types, which should be resolved by the ability to identify vehicles according in respect to specific make and model.

2.3 Vehicle Make and Model Recognition

In the automobile sector, Vehicle Make and Model Recognition (VMMR) is a highly regarded subject matter because of its importance in intelligent transport systems. In comparison to the previously discussed concepts of vehicle detection and type identification, this concept delves much deeper since it focuses greater emphasis to subtle distinctions within and across vehicle classes. It is capable of classifying vehicles based on their make and model, among other things. Given its significance, it is a topic that requires further investigation, since there has only been a minimal amount of research conducted in this field. Within the VMMR domain, existing study have been grouped into three, each of which is dependent on the technique utilised in order to solve the VMMR problem. These approaches

include feature-based, appearance-based and model-based, while there have also been innovative methods like deep learning and part-based methods which have been proposed in over the last few years to contribute to addressing the VMMR problem. According to Gu and Lee, identifying the model of a vehicle is improved by its structural features. As validation for this assertion, Daza et al. (2014) used this technique to a collection of 1350 vehicle images and were 0.94 accurate on the results. When it comes to classifying vehicles by make and model, some researchers contend that relying just on the structural representation of vehicles is insufficient. These researchers suggest a vehicle should be grouped according to specific physical attributes. For example, a model created by Cootes and Petrovic (2004) was able to identify vehicles by extracting features from their frontal images and mapping them to the dataset images. They were able to classify 1130 vehicle photos from 77 classes with 0.93 accuracy on their model. Furthermore addition, Manzoor, Morgan, and Bias (2019) created a model to extract features from a specified ROI. In order to differentiate one vehicle from another, ROI had distinctive features. Features and pixels were extracted using Histogram of Oriented Gradient and GIST respectively. The accuracy of the classification was 0.97. Another school of thought holds that just classifying vehicles by their appearance and features is insufficient; rather, in order for a vehicle to have an appropriate classification, it needs some modelling and then classification should be done according to the resulting model. This may be observed in the work of Betke, Haritaoglu, and Davis (2014), who built a framework for the 3D modelling of vehicles through identifying and extraction of relevant points. The classification algorithm had a 0.91 accuracy rate, which was excellent. Testing was done on a collection of about 440 vehicle images representing 40 different vehicle classes.

Nevertheless, it is worth emphasizing that these algorithms generate excellent performance while only dealing with a comparatively low dataset volume of vehicles, indicating that their usefulness is limited (Llorca et al. 2014). In a realistic scenario when there are wide range of vehicles of varying makes and models, these approaches may not possess the capacity to functional performance. A major part of the research also employ a considerable amount of poorly defined attributes that do not show considerable variation across several lower-level classes with similar appearances, despite the fact that they are used extensively in the research.

It has become more popular for researchers to apply deep learning to computer vision thanks to rapid advances in that discipline (LeCun, Bengio, and Hinton, 2015). Deep learning is founded on the notion that actual representations of output values is the product of a multi-layered method to extrapolating it. A greater degree of information is gained sequentially, with the lowest-level features serving as the reference point for the abstraction process. There is a lot of success with the deep learning algorithms since they can uncover structural complexity in very complex and huge datasets (Krizhevsky, Sutskever, and Hinton 2017). One of the most prominent achievements computer vision and deep learning alogorithms is the DeepFace by Facebook (Taigman et al. 2014). In order to better understand the application of deep learning on VMMR, Fang et al. (2016) modelled a convolution neural network (CNN) for this purpose. Using feature mapping, the model was able to locate granular parts. In a study conducted on a dataset that included 290 different makes and models of vehicles, the model was shown to be accurate to 0.98. The method proposed by Yu

et al. (2017) combines a CNN with a Baysian Network for vehicle classification and extraction of metadata. A dataset of 200 vehicle classes were examined with the combination of the algorithms and the accuracy was 0.88. In order to train a neural network for good classification, it is necessary to feed it a large number of images with substantial amounts of variance. The authors were successful in demonstrating the application of deep learning and its benefits in computer vision, notably in the area of vehicle make and model recognition. Past research, on the other hand, have indeed utilised relatively limited variety of vehicle image datasets for this particular subject matter. There is still a gap as to if a deep learning model can accurately classify vehicles based on their make and model recognition if there is an enormous amount of data provided. Additionally, Yu et al. (2017) demonstrates vaguely the idea of merging deep learning with Bayesian networks to extract metadata; this is a technique that may be used in the extraction of specifications of vehicles and match them appropriately with the vehicles that have been classified. There is no definitive answer to the issue of how precisely vehicle specifications can be combined with the results of an appropriate vehicle make and model recognition classifier.

In conclusion, there is evidence that the domain of computer vision proves to be a domain that attracts a lot of research work, with increased interest in subjects such as object recognition. Recognition and classification of images by machines are essential topics since they allow machines to recognise and classify images in identical way humans do. The advantages of using computers to do image recognition tasks rather than humans are considerable a lot. According to the different literature that have been evaluated, vehicle detection, VTR, and VMMR have all emerged as extremely significant study subjects in the field of image classification. The necessity for proper vehicle classification originates from a variety of perspectives, including buyers' knowledge, vehicle accessibility, traffic management and security. Existing study has shown the advancement of the subject, the many techniques attempted to address the issue, the advantages of certain methodologies over others, as well as the gaps which ought to be addressed by further investigation into the subject matter. In past studies, there has been a key gap which is the availability of sizeable datasets that contain a large variety of classes and vehicle images, most models discussed have only been evaluated on limited datasets, which is insufficient to determine if these same accuracies can be achieved on large datasets. In addition, there is a lack of emphasis in the subject of retrieving metadata, which is a critical component of data analytics. It is imperative that information retrieval should be incorporated in VMMR such that when a vehicle is correctly classified information about its make, model, year of manufacture and other information about the vehicle is displayed. This research aims to address these problems that have been discovered. The methods and processes that will be used to address these shortcomings are detailed in the next chapter of this project.

3 Methodology

The methodology used for this research is based off the generic CRISP-DM approach but has been redesigned to fit into the aims and objectives of this research. Figure 1 below shows a graphical illustration of the methodology with the actions taken in each step of the process.

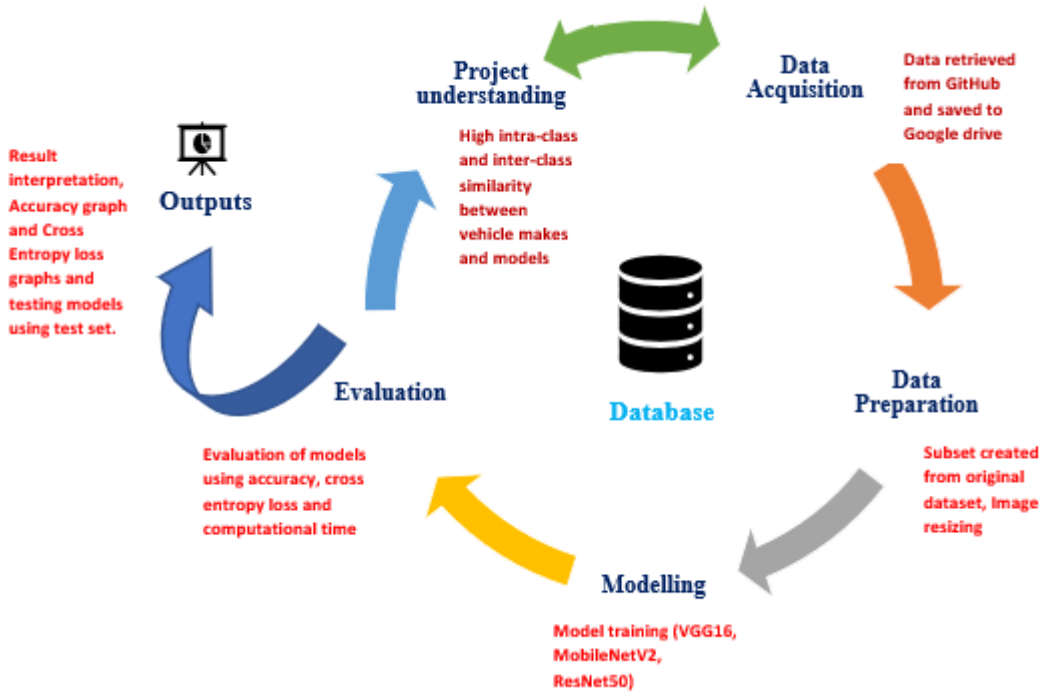

Figure 1: Vehicle make and model recognition methodology

3.1 Project Understanding

Understanding the problem of the VMMR problem was key to this project being successful. Reviews on previous research was extensively done to find out where the problem lies and what gaps could be filled. The identified problem for this project was the high interclass similarity of vehicles of different makes and the high intraclass similarity between vehicles of the same make but different models. The objective is to find out how well deep learning models can help in solving this problem.

3.2 Data Acquisition

Based on previous research and the limited accessibility and use of a large dataset, finding a large dataset was crucial to this project. The choice of data is very crucial, the appropriate data has to be chosen in order for the project to make sense. There are various factors that needed to be considered in acquiring data such as ethical issues, bias, data source, variance, volume and viability of the data all needed to be considered. Data acquired also helped the researcher have a better understanding of the project. The dataset used is the VMMRDB[1] dataset. The VMMRDB dataset contains 291,000 images and 9170 classes. The dataset contains information about the make, model and year of manufacture of the vehicles. The range of years in the two datasets span from 2002 to 2016.

[1] https://github.com/faezetta/VMMRdb

3.3 Data Preparation

Most time data in its raw state contains a lot of noise or other defects which may affect the viability of the data. In other cases, just as is with the case of the VMMRDB, the dataset may be too voluminous and may need subsets to be created and other cleaning tasks. In order to create a balanced subset from the VMMRDB, a python script was written to randomly select 203 classes with a maximum of 27 vehicle images and minimum of 26 images each. This generated a dataset containing 5368 vehicle images with 203 classes. It was observed that there were discrepancies in the heights of the various vehicle images, in order to rectify this, image cropping was done to normalise the images. In order to determine the maximum and minimum values for the vertical and horizontal axis of the images, a bounding box was used on each image to detect and extract vehicles. In order to minimise over-edging and distortion, padding was applied to the cropped image.

3.4 Modelling

Several modelling algorithms were used to model the dataset for vehicle make and model recognition and accurate classification of vehicle images. It was important to do research and find out which models work best for the image classification problem and in particular the VMMR problem. The use of pretrained models as compared to the other models discussed in chapter is one that is highly effective. Pretrained do not need a lot of effort to execute as compared to building a model from the base, which will required extensive training over a large dataset. Hence, the choice to use pretrained models. ResNet-50 was chosen as one of the models because it is one that is widely used in image classification. It has a 50 layers depth and was trained on over 14 million images with a wide range of classes. It contains over 23 million trainable parameters; with this kind of architecture it is definitely a good fit for image classification tasks. The models that were developed are only mentioned in this section, however the implementation of these models will be discussed in detail in the implementation phase. The models used were the ResNet50, MobileNet-v2 and VGG-16 models. The MobileNet-v2 model is a lightweight model and when compared to other models it has lesser calculations and parameters. This makes the image classification time faster. The final aim of the project is to develop an application, therefore the decision to use the MobileNet-v2 because of its viability for application purposes. The VGG-16 model has been classified as one of the top 5 achieving models on the ImageNet dataset, which makes it suitable for image classification tasks. All of these models are chosen for the reasons stated above, but the aim is to find the best performing model for this dataset in terms of classification as well as computational time.

3.5 Evaluation

To get a sense of performances of the models, there is a need for evaluation. The accuracy and the loss function of this model are both taken into account when assessing its performance. The two metrics (cross-entropy loss function and accuracy) are calculated on both the training and testing sets. To split the dataset into training and testing sets the number

of images in the original dataset folders are calculated and then they are split according to the split ratio that was set which was 80:20. 80 going for the training set and 20 for the test set. The test data is saved in a different folder. The results of these metrics determines the performance of the model on both sets. The accuracy is a measure of how the predictive accuracy of the model is in comparison to the real data. Another performance evaluation metric used is the computational time for each implemented model. The accuracy illustrates how well the model performs across all classes while the loss function illustrates how far off classifications are from the actual value.

3.6 Results

The output or the results generated by the model need to be made sense of, in order to do this, the results are interpreted and also shown through different graphical illustrations and images. The different plots used are the training accuracy, validation accuracy, loss function and validation loss function plots. Also, each model is tested over 12 randomly selected images and a plot containing these vehicle images are presented indicating if they have been accurately classified or not.

4 Design Specification

The Figure 2 below shows the overall architecture of the VMMR system. It depicts the different layers and the processes involved. It is a 2-tier architecture. The first tier being the logic while the second tier being the client tier.

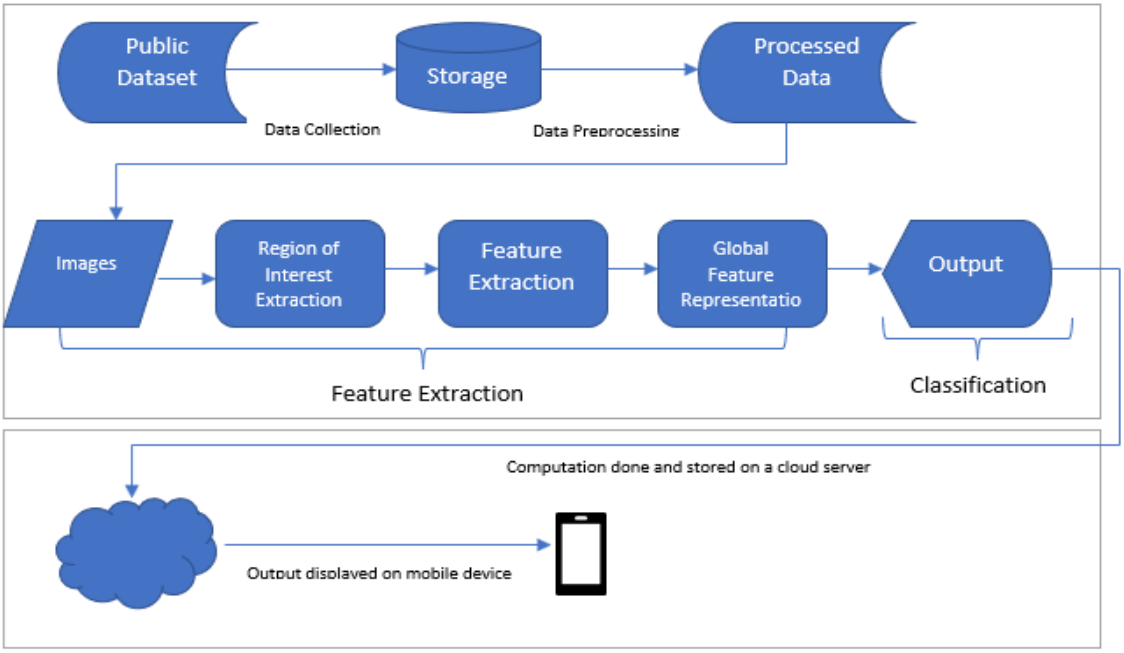

Figure 2: System Architecture for the VMMR application

4.1 The ResNet-50 Model

ResNet-50, also known as Residual Networks, is another common deep learning model in use today. The model is made up of 50 layers which makes it more extensive than the previous models implemented. Unlike the previous models, this model has normalisation layers in addition to ReLu and convolution layers. The benefits of adopting ResNet50 are that it increases the exploitation of features, strengthens feature propagation, and reduces parameter usage by a significant amount. Since the identity connection used by ResNet avoids connecting across layers and instead passes output from one layer to the next, it enables it to build models that are both more complex and more detailed (He et al., 2016).

The flattened outputs from the ResNet-50 are passed through the dropout layer and then for classification through the softmax classifier.

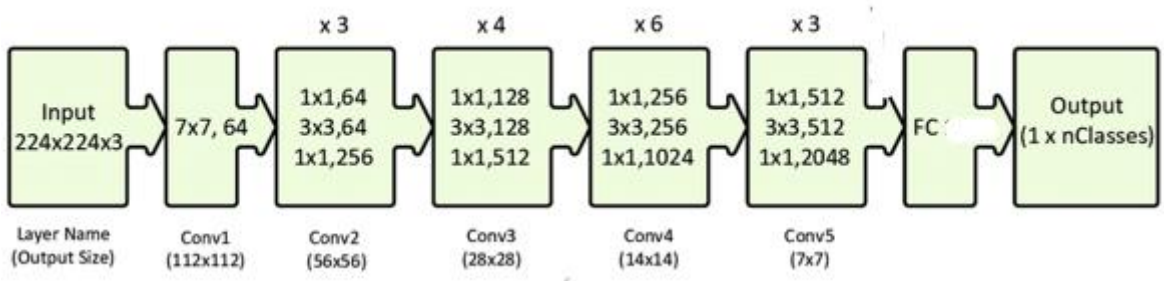

Figure 3: ResNet-50 Architecture

4.2The MobileNet-V2 Model

MobileNet-V2 is a deep neural networks framework that has been designed to operate efficiently on mobile devices such as smartphones and tablets. It is founded on the principle of the inverted residual structure, in which between the bottleneck layers lies the residual connections. A lightweight depthwise convolution filter is used in the intermediate expansion layer for filtering out features, which serves as a source of non-linearity. The residual block has a stride of 1 while the second block with a stride of 2 is used to downsize, each block has 3 layers. The first being a 1x1 with a Relu6 and the third also a 1x1 and in the intermediary is the depthwise convolution filter.

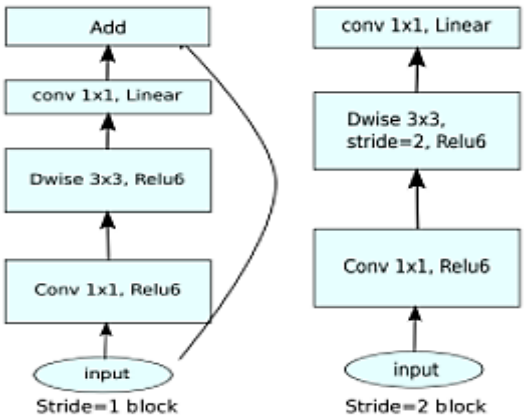

Figure 4: MobileNetV2 Architecture

4.3The VGG-16 Model

The VGG-16 model is one of the most used models in deep learning, it is a transfer learning model. The architecture is such that the layers are in blocks. It has a total of five (5) blocks. The initiating 2 blocks have two (2) convolution layers and one (1) maxpooling layer while the remaining blocks each have 3 convolution layers and a maxpooling layer attached to them. This makes up the 13 convolution layers, these then have 2 fully connected layers attached to them with a softmax classifier for classification which makes a total of 16 layers, hence the name VGG16. The architecture is such that it contains 16 layers. 13 of these layers are convolution layers, it also has 2 fully connected layers and the last layer for classification is the softmax classifier. The ImageNet dataset is what was the model was trained on. It is one model that has been used extensively for image classification (Deng and Liu, 2015).

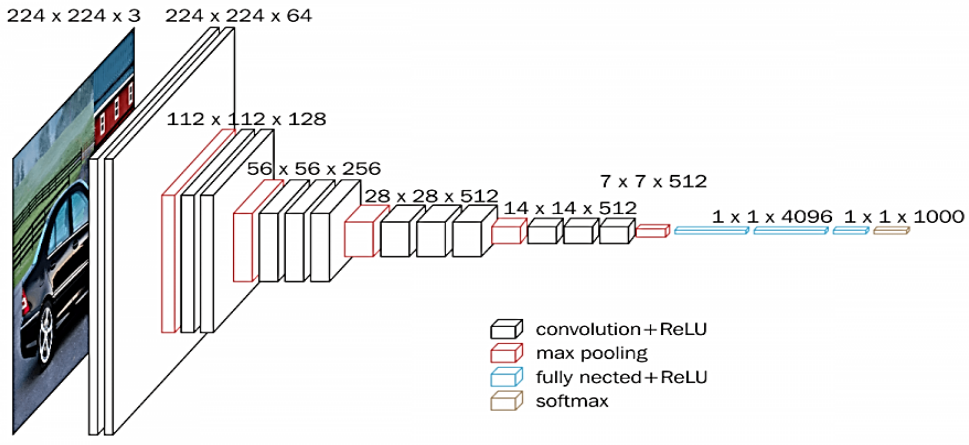

Figure 5: VGG-16 Architecture

5 Implementation

Data preparation, definition of model architectures and parameters, training and testing of the different models, evaluation of results produced as well as cross-validation are all processes associated with the implementation phase. In order to get accurate outcomes, the model must be validated for both underfitting and overfitting, and the parameters for the model must be optimized accordingly. However, although the models that have been developed are centred on neural network models, the architectures of the models vary. With the aid of scikit-learn package, the vehicle image dataset was divided into both training and testing sets in an 80/20 split. The stratify parameter enabled the classes to have equal amounts of data when the dataset was being split. The validation accuracy is obtained at the completion of each epoch by passing a 0.25 fraction of the training set through as validation data. As a result of running a lot of epochs, the model can often end up having an overfitting problem, it was imperative to use early stopping and setting a patience value. Training is terminated once a set number of epochs have been completed. After each epoch, the trained model is monitored for

improvement in validation loss. A non-improvement in the validation loss value means training has finished.

5.1 Models

For optimization of every model the Adam optimizer is used. The batch size was set to 32, image size was set at 224 x 224. The patience value was set at 5 and the number of epochs was set at 100, even though all models finished learning before getting to 100 epochs. Categorical Cross-Entropy was used for the loss function. The split between training and testing was 80/20 with a validation split of the training set at 0.25. The outputs and results of the model are discussed in the evaluation chapter. The models were trained using the pre-trained weights. A call-back utility was defined during the training of all the models such as early stopping in order to prevent inconsistent model training and to maximize computational resources. A tensorboard[2] is deployed to aid in visualizing the outputs of accuracy and loss. The tensorboard is a tool that provides all the visualizations and measurements that machine learning needs.

5.2 Mobile Application

A GUI is created using application using KIVY python framework. This application is a set of 5 pages altogether. Home page containing 2 buttons: one for opening the camera device to capture image and the second to browse files and pick up an image. Files page where the user browses the files and picks up an image. Camera page where the user can capture an image. Prediction page where the predictions on the image are displayed. Error page that shows up when an error occurred during predictions phase. The GUI application submits the image to predict to a remote server that makes predictions and send back the top 3 classes to the GUI application for display. The server is set up using python FastAi library. It supplies an endpoint that the mobile application uses to send the captured image and get predictions back.

6 Evaluation

The results and the evaluation of the three models implemented in the preceding chapter are presented in this chapter. The results of each model are evaluated based on their accuracy, loss function and computational time and these metrics are used to compare the performance of each model.

6.1 Experiment 1: MobileNet-V2 Model

The model performed excellently. There was no underfitting and overfitting this is evident in the fact that the gap between the train and validation accuracy continuously decreases while the learning rate is excellent as seen from the train and validation loss. The model achieved a

[2] www.tensorflow.org/tensorboard/get_started

training accuracy of 0.993 while the test accuracy was 0.971. Computational time for training was 49 minutes 22 seconds, running over 5 epochs. Figure 6 below shows the results of the model.

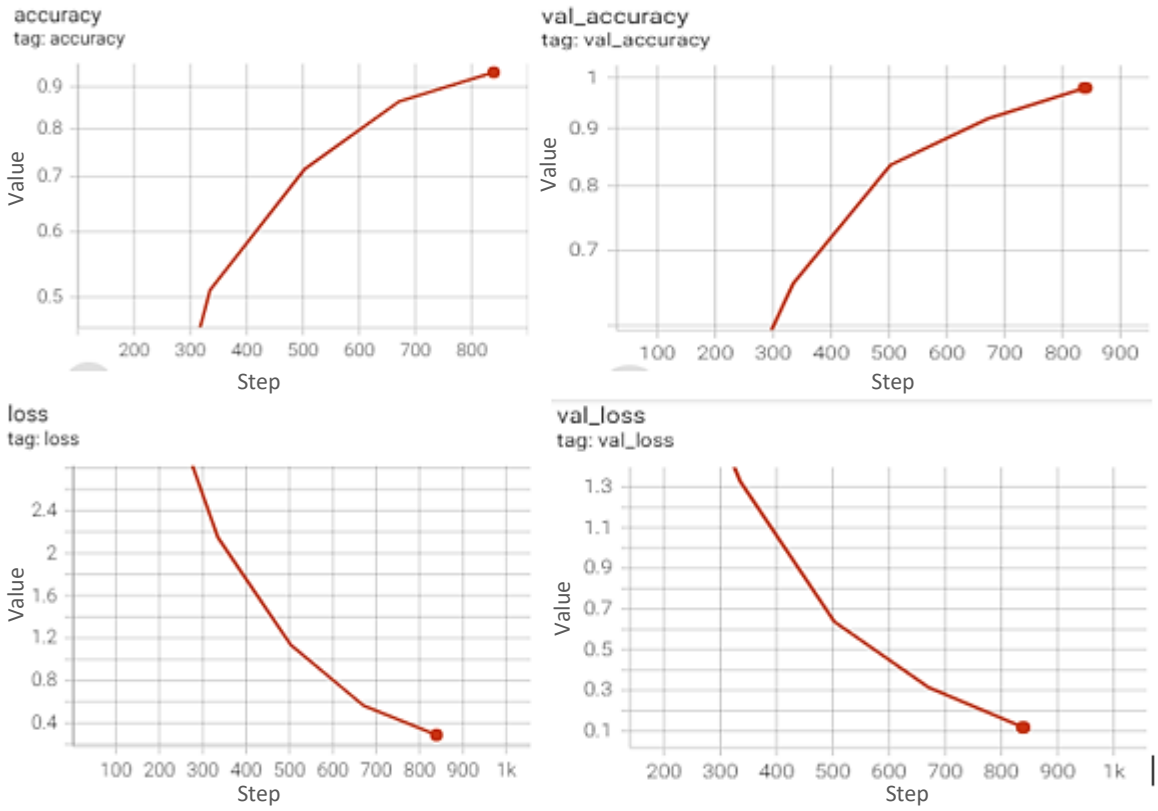

Figure 6: Model accuracy and loss graphs of the MobileNet-v2 Model

Figure 7: Testing predictions of MobileNet-v2 Model
(Green indicating correct classification)

In order to have a clearer visualisation of the text written on the testing predictions, Figure 8 shows a cross-section of only 6 images of the test predictions with a higher resolution. The figure shows both the real vehicle name and the predicted vehicle name.

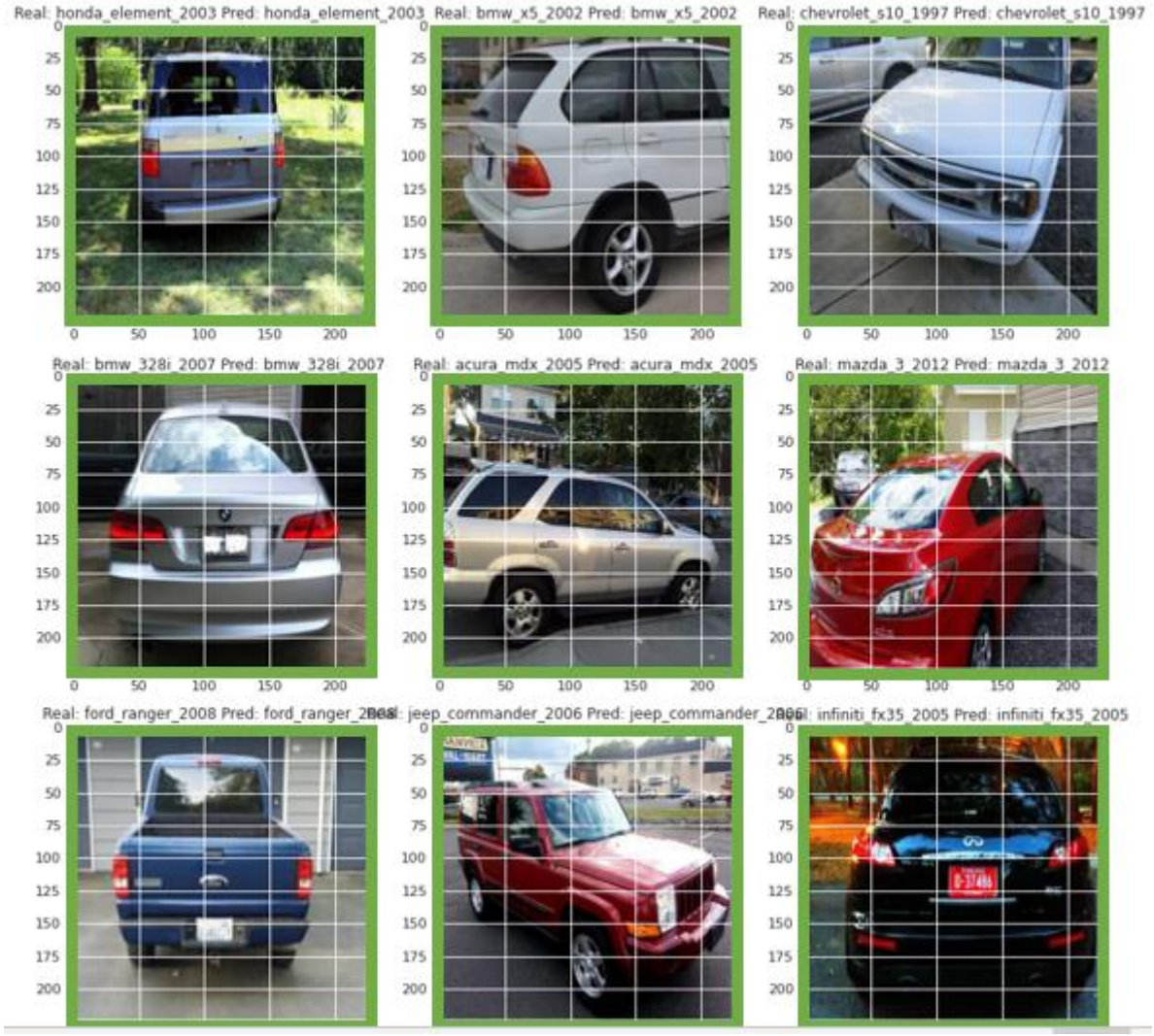

Figure 8: Cross-section of testing predictions
(Green indicating correct classification)

6.2 Experiment 2: The ResNet-50 Model

The model seems to also perform excellently. Just as with the CNN model, there is no underfitting and overfitting this is evident in Figure 9 below from the plots. The model achieved a training accuracy of 0.985 while the test accuracy was 0.977. Computational time for training was 51 minutes 44 seconds running over 5 epochs. The author intentionally interrupted the training at epoch 5 when training accuracy was at 0.96 and test accuracy was at 0.90 and tested the model. Figure 10 below shows the test scenario of the model. The model was only able to predict 8 out of the test images accurately while after training for 5 epochs with test accuracy at 0.977 the model predicts all 12 vehicles appropriately as shown in Figure 11. This shows that the model has a good learning rate and was constantly learning after each epoch. Figure 9 shows the accuracy plots of the model.

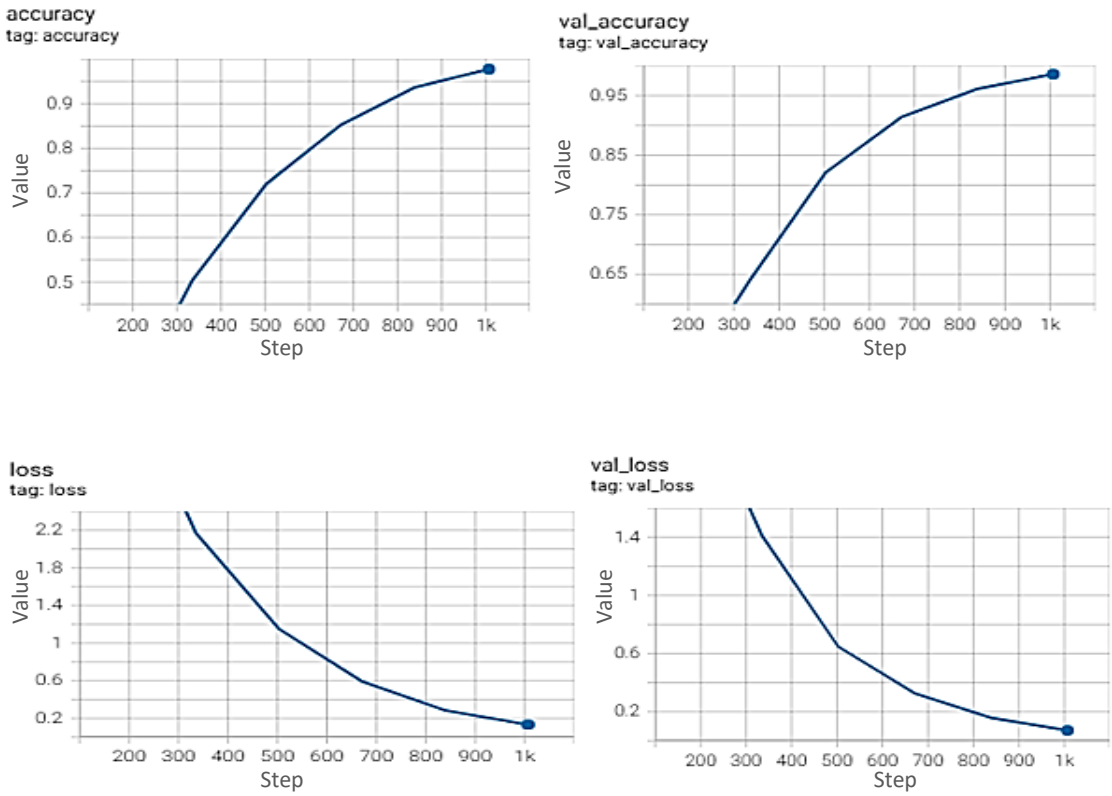

Figure 9: Model accuracy and loss graphs of the ResNet50 Model

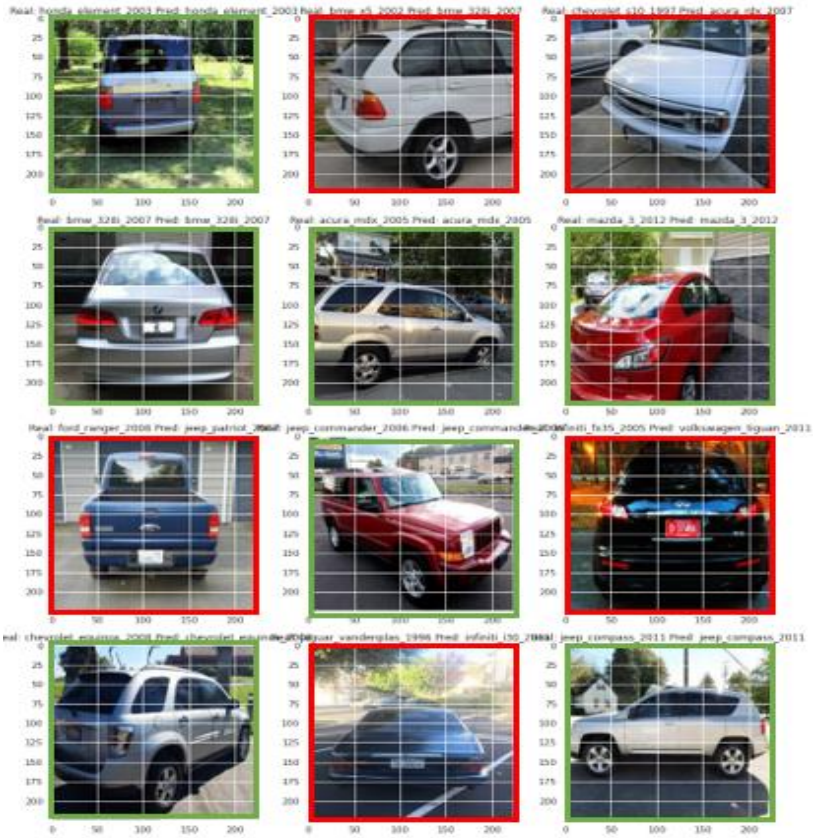

Figure 10: Testing predictions at 0.9 accuracy

(Green indicating correct classification and red incorrect classification)

Figure 11: Testing predictions at 0.977 accuracy

(Green indicating correct classification)

6.3 Experiment 3: The VGG-16 Model

The model had a training time of 3 hrs 27 minutes with a train accuracy of 0.992 and a test accuracy of 0.92. The Figure 12 below shows that the validation set is not capable of providing sufficient information to evaluate the capability of the model to generalize.

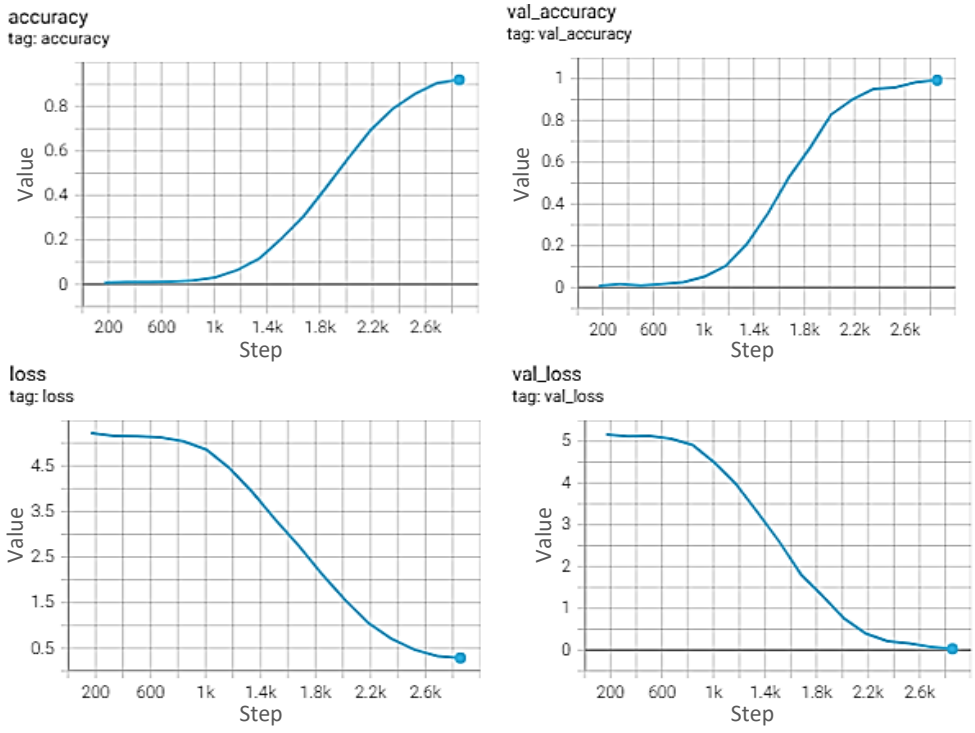

Figure 12: Model accuracy and loss graphs of the VGG-16 Model

Figure 13: Testing predictions of VGG-16 Model
(Green indicating correct classification)

6.4 Experiment 4: The GUI Application

The ResNet-Model is incorporated into a GUI application and is used to test the prediction capability of the model by uploading images through it. The application returns the top three predictions for the image passed through it. The result shows that even by passing a distorted image through the model, the model was still capable of identifying the vehicle. Figure 14 shows the output.

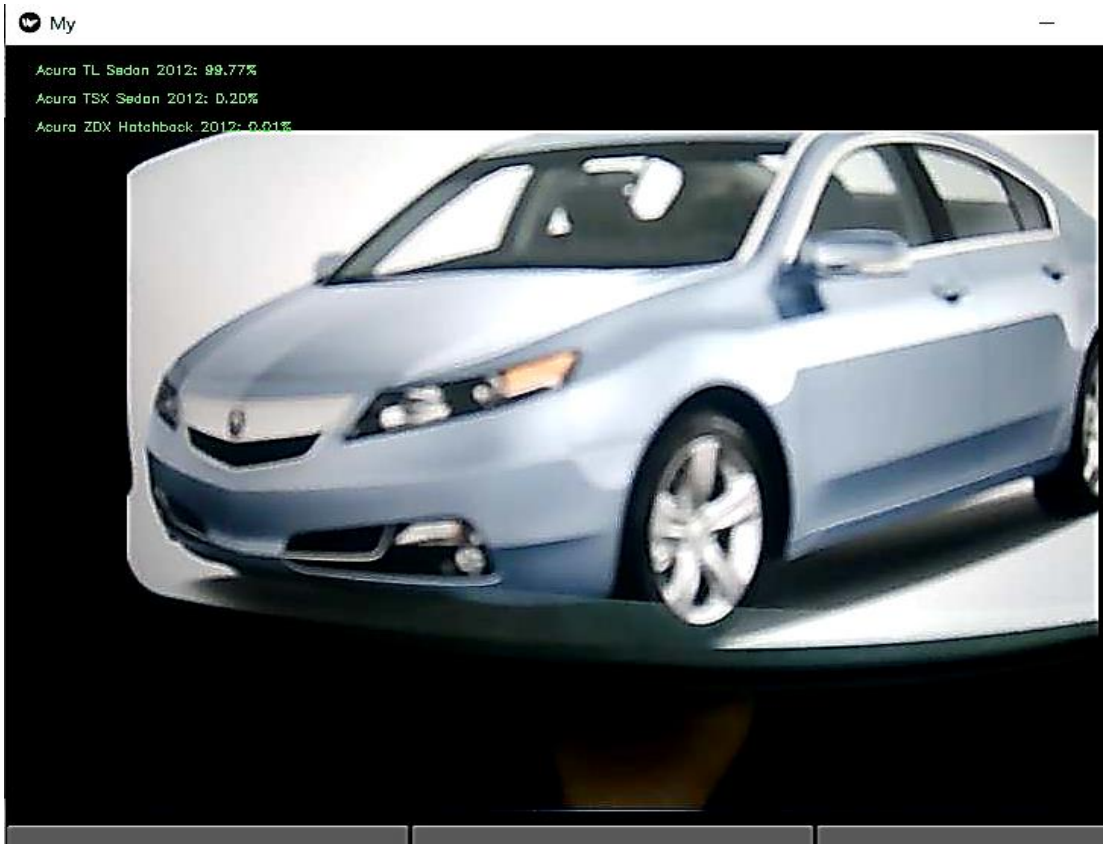

Figure 14: Test result of GUI application

6.5 Discussion

The accuracy of all models during training and testing was remarkably similar, with just a little difference between the ±0.1 range. Each model was evaluated by randomly picking 12 images from the test set in order to determine its overall performance. As shown in Table 1, the MobileNet-V2 model required the least amount of time to execute when compared to the other models, making it the most efficient model in respect to computational resources. From the second evaluation metrics of the study which is accuracy. ResNet-50 achieves a higher accuracy that the other models. When comparing the implemented models, it is evident that based on model performance taking into cognisance the trade-off with computational resources (time) ResNet-50 outperforms the two other models as shown in Table 1. When the models were tested against a random selection of 12 images, ResNet-50 appropriately recognized and classified all the 12 images, MobileNet-V2 also appropriately recognized and classified only all 12 images and the VGG-16 also appropriately recognized and classified all 12 images. The best perform model was used to then used incorporated into the GUI application. For every image passed into the GUI application, the application displays the top 3 predictions based on the images.

Table 1 Comparison of Model Performance

Model	Train Accuracy	Test Accuracy	Computation Time
MobileNet-v2	0.993	0.971	49 mins 22 secs
ResNet-50	0.985	0.977	51 mins 44 secs
VGG-16	0.992	0.92	3 hrs 27 mins

Furthermore, when the observed results are compared to the findings of previous models by other authors as shown in (Table 2), the generated model outperforms most of them by a considerably small margin, especially when considering the larger number of classes that the dataset contained. The diversity of vehicles it is capable of recognising and classifying is impressive. Furthermore, the generated models address the difficulties that previous research had with high inter-class and intra-class similarities. However, while this was tested on a considerable larger number of classes, there was still a limitation in the number of vehicle images due to low computational resources. Further work can be done on the number of images while keeping in mind the approach the author has used in limiting the number of vehicles in each class so as to achieve a balance and uniformity in vehicle images and classes.

Table 2 Comparison of previous Research and Implemented System

Author	Number of images and classes	Classification Method	Accuracy
Llorca et al. (2014)	1342 vehicles, 52 classes	Hand-crafted feature representation	0.940
He et al. (2015)	1196 vehicle, 30 classes	ANN	0.924
Chen et al. (2015)	6639 vehicles, 29 classes	Sparse representation and hamming distance	0.911
Siddiqui et al. (2016)	6601 vehicles, 29 classes	Bag of Surf Features	0.948
Fang et al. (2017)	44,481 vehicles, 281 classes	CNN	0.986
Samuel Tanga (2021)	5364 vehicles, 203 classes	ResNet-50	0.977

7 Conclusion

The aim of this research was to use deep learning approaches in solving the vehicle make and model recognition (VMMR) problem. In order to achieve this, three different deep learning models were employed, the MobileNet-V2 model, the ResNet-50 Model and the VGG-16 Model. The experiments carried out using these models were done on the VMMRDB a public dataset containing over 291,000 vehicle images with 9170 classes. To run the experiment, a subset of this dataset was created, which contained 5368 vehicle images and 203 classes. Each model was evaluated based on model accuracy, model loss and computational time. After experimentation and evaluation, it was observed that the ResNet-50 model outperformed the other two models with an accuracy of 0.977 and a training computational time of 51.44 minutes. This model was then incorporated into a mobile application as the final product where users can upload saved images or use the camera to capture a new vehicle image. The system was able to only retrieve the vehicle make, model and manufacture year as these was the only metadata associated with the vehicle images, further research can be done on merging more metadata containing more specific details about the vehicles to their images. Also, more research can be done in the aspect of experimenting across a wider range of datasets, as this was not possible due to insufficient computational resources. The research output will assist the intended user group in quickly identifying vehicles that they were previously unaware of, as well as relieve the stress of having to continue looking for a vehicle over the web. The system has the potential to be commercialised and made available to the general public. The research project has been able to make a contribution to the field of computer vision by solving the vehicle make and model recognition and classification problem.

Acknowledgement

I want to use this medium to express my gratitude to my supervisor Dr. Martin Alain for the huge role he has played in my ability to complete this research project. My appreciation also goes to John Kelly and every other lecturer that have given me the knowledge to boost my career goals through this data analytics programme. I also thank my family for all the support given me. Thank you all.

References

Betke, M., Haritaoglu, E. and Davis, L., 2000. Real-time multiple vehicle detection and tracking from a moving vehicle. *Machine Vision and Applications*, 12(2), pp.69-83.

Chen, L., Hsieh, J., Yan, Y. and Chen, D., 2015. Vehicle make and model recognition using sparse representation and symmetrical SURFs. *Pattern Recognition*, 48(6), pp.1979-1998.

Chen, Z., Ellis, T. and Velastin, S., 2016. Vision-based traffic surveys in urban environments. *Journal of Electronic Imaging*, 25(5), p.051206.

Everingham, M., Eslami, S., Gool, L., Williams, C., Winn, J. and Zisserman, A., 2015. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1), pp.98-163.

Fang, J., Zhou, Y., Yu, Y. and Du, S., 2017. Fine-Grained Vehicle Model Recognition Using A Coarse-to-Fine Convolutional Neural Network Architecture. *IEEE Transactions on Intelligent Transportation Systems*, 18(7), pp.1782-1792.

Faro, A., Giordano, D. and Spampinato, C., 2011. Adaptive Background Modeling Integrated with Luminosity Sensors and Occlusion Processing for Reliable Vehicle Detection. *IEEE Transactions on Intelligent Transportation Systems*, 12(4), pp.1398-1412.

Hsiao, E., Sinha, S., Ramnath, K., Baker, S., Zitnick, L. and Szeliski, R., 2014. Car make and model recognition using 3d curve alignment. *Applications of Computer Vision (WACV), IEEE Winter Conference*, p.1.

Kafai, M. and Bhanu, B., 2012. Dynamic Bayesian Networks for Vehicle Classification in Video. *IEEE Transactions on Industrial Informatics*, 8(1), pp.100-109.

Krizhevsky, A., Sutskever, I. and Hinton, G., 2017. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), pp.84-90.

LeCun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. *Nature*, 521(7553), pp.436-444.

Leotta, M. and Mundy, J., 2011. Vehicle Surveillance with a Generic, Adaptive, 3D Vehicle Model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(7), pp.1457-1469.

Li, M., Kwok, J. and Lu, B., 2010. Online multiple instance learning with no regret. *IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pp.1395-1401.

Li, Y., Gu, L. and Kanade, T., 2011. Robustly Aligning a Shape Model and Its Application to Car Alignment of Unknown Pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(9), pp.1860-1876.

Llorca, D., Colas, D., Daza, I., Parra, I. and Sotelo, M., 2014. Vehicle model recognition using geometry and appearance of car emblems from rear view images. *IEEE Intelligent Transportation Systems Magazine*, pp.3094-3099.

Manzoor, M., Morgan, Y. and Bais, A., 2019. Real-Time Vehicle Make and Model Recognition System. *Machine Learning and Knowledge Extraction*, 1(2), pp.611-629.

Nilsback, M. and Zisserman, A., 2006. A visual vocabulary for flower classification. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference*, 2, pp.1447-1454.

Petrovic, V. and Cootes, T., 2004. Analysis of features for rigid structure vehicle type recognition. *British Machine Vision Conference*, pp.1-10.

Taigman, Y., Yang, M., Ranzato, M. and Wolf, L., 2014. Deepface: Closing the gap to human-level performance in face verification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.1701-1708.

Teoh, S. and Bräunl, T., 2011. Symmetry-based monocular vehicle detection system. *Machine Vision and Applications*, 23(5), pp.831-842.

Wah, C., Branson, S., Perona, P. and Belongie, S., 2011. Multiclass recognition and part localization with humans in the loop. *Computer Vision (ICCV), IEEE International Conference*, pp.2524-2531.

Wang, R., Zhang, L., Xiao, K., Sun, R. and Cui, L., 2014. EasiSee: Real-Time Vehicle Classification and Counting via Low-Cost Collaborative Sensing. *IEEE Transactions on Intelligent Transportation Systems*, 15(1), pp.414-424.

Yang, L., Luo, P., Loy, C. and Tang, X., 2015. A large-scale car dataset for fine-grained categorization and verification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp.3973-3981.

Yu, S., Wu, Y., Li, W., Song, Z. and Zeng, W., 2017. A model for fine-grained vehicle classification based on deep learning. *Neurocomputing*, 257, pp.97-103.

Zhang, T., Zhao, R. and Chen, Z., 2020. Application of Migration Image Registration Algorithm Based on Improved SURF in Remote Sensing Image Mosaic. *IEEE Access*, 8, pp.163637-163645.