

## Acknowledgments

Thank you to Miss Nicole for guiding us on our assignment, thank you Jessica Li on Kaggle for providing the dataset, and thanks to all those researchers who wrote their papers that we used on our related works.

## Abstract

The goal of this project is to utilize existing computer vision techniques to improve the accuracy and reliability of existing car classification models. Our group has extensively reviewed 8 papers on their approach to car identification and classification. We have proposed several modifications and improvements to enhance both the accuracy and effectiveness of currently accessible artificial neural networks, mainly Convolutional Neural Networks (CNNs). Based on our extensive research in the literature reviews, we found that CNN architectures such as ResNet and VGGNet are extremely useful in handling image processing tasks thanks to their capability in image identification and classification. For example, Ahmed S. Algamdi et al. (2023) achieved a whopping 99.7% accuracy using a VGG model, which is the best combination in terms of performance we had reviewed. We also discovered that transfer learning and deep learning can significantly increase the model's accuracy in classification while addressing problems such as gradient fading. Our study compares various CNN architectures, mainly ResNet and VGGNet to figure out which model is the best suited for car model classification. In this study, we hope to achieve improved accuracy and scalability in different circumstances by utilizing modern preprocessing methods. The model's implementation and results such as performance and accuracy metrics will be discussed and shown in the next assignment.

## **Table of contents**

<b>1.0 Introduction</b>	<b>7</b>
<b>1.1 Background</b>	<b>7</b>
<b>1.2 Research Goal</b>	<b>8</b>
<b>1.3 Research Objectives</b>	<b>8</b>
<b>2.0 Related Works</b>	<b>13</b>
<b>2.1 Gap Analysis</b>	<b>14</b>
<b>2.2 Scope of research</b>	<b>15</b>
<b>3.0 Methodology</b>	<b>16</b>
<b>3.1 Dataset</b>	<b>16</b>
<b>3.2 Data Preparation</b>	<b>16</b>
<b>3.2.1 Data Collection</b>	<b>16</b>
<b>3.2.2 Data Preprocessing</b>	<b>16</b>
<b>3.2.3 Image Representation and Feature Extraction</b>	<b>19</b>
<b>3.2.4 Data Exploration and Analysis</b>	<b>20</b>
<b>3.3 Models</b>	<b>21</b>
<b>3.4 Model Evaluation</b>	<b>25</b>
<b>3.5 Summary of methodology</b>	<b>27</b>
<b>6.0 Conclusion</b>	<b>27</b>
<b>7.0 References</b>	<b>29</b>

## **1.0 Introduction**

### **1.1 Background**

Due to the rapid growth and production of cars over the past years, the demand for car classification and identification models have increased. In recent years, car manufacturers are under pressure to create more cars than in the past as the automotive industry continues to grow and evolve in order to meet the growing demands of customers (Alexandros Kelaiditis, 2023). Furthermore, given that car industries differ significantly throughout nations, this results in the production of both a substantial number of cars and a wide range of distinctive car types. The problem arises due to the large amount of car models available, making it a difficult task for humans to accurately identify the make and model of a car. The process of classifying cars becomes substantially more difficult due to several factors such as camera angles, lighting conditions, etc. As a result, traditional car classification methods are deemed not suitable due to them being prone to flaws and human errors as well as their inability to scale. Thus, they often come up short of the strict standards. With recent advancements of computer vision however, Convolutional Neural Networks (CNNs) have become a liable solution for most image processing tasks. CNNs are particularly effective for tasks such as object identification and classification as they are good at identifying distinctive patterns and characteristics within the images. Additionally, the implementation of transfer learning and data augmentation methods will significantly improve the model's accuracy in classification while addressing problems such as gradient fading. With these advancements in computer vision, our group's aim for this project is to utilize existing computer vision techniques such as normalization and data augmentation to improve the accuracy and reliability of existing car classification models.

### **1.2 Research Goal**

The primary goal of this research is to develop and validate a robust computer vision model capable of accurately identifying the make, model, and year of vehicles from images captured in various environmental conditions. This model aims to leverage advanced deep learning

techniques, specifically convolutional neural networks(CNN), to process and analyze visual data, distinguishing between subtle differences in vehicle design and features.

### **1.3 Research Objectives**

The objective of this research is to modify and evaluate existing artificial neural networks such as convolutional neural networks capable of identifying the make, model, and year of vehicles from images fed into the model. This involves:

1. Data preprocessing:
  - a. Acquire a diverse and comprehensive dataset of vehicle images, including different makes, model, and years.
  - b. Implementing preprocessing techniques like unfreezing classification layer and fine tuning higher level features to enhance image quality, normalize data, and address challenges such as occlusions, shadows, and reflections
2. Model comparison:
  - a. Modify existing convolutional neural network (CNN) architecture using transfer learning in order to optimize the model for vehicle recognition tasks.
  - b. Implement advanced machine learning techniques to improve the model's ability to distinguish between visually similar vehicle models.
    - Choosing the most suitable performance metrics

## 2.0 Related Works

Reference	Domain	Dataset	Feature Selection	Model	Performance
(Buzzelli & Luca Segantin, 2021)	Image classification of car models	Stanford cars dataset containing 16,185 image classification pairs across 196 classes	Data augmentation- random rotations of 20 degrees from anchor, random translations and color jittering. Classification method-Two-Step Cascade: This approach splits the problem into two sub-tasks: First predicting the car type(12 classes), then directing it to a specific classifier for make, model, and year(MMY) classification.	Multiple CNN architectures were used for testing such as Resnet-50	The best performing model achieved a Top-1 accuracy of 91.05% and a Top-5 accuracy of 98.88% which was achieved by Resnet-50.
(Simoni et al., 2021)	Improving Car Model Classification through vehicle keypoint localization	Pascal3D+ dataset that includes 10 sub-categories of car models. This dataset contains 4081 training images and 1024 testing images	Leveraging both visual and pose information, leads to better results than using either models independently	Combining Stacked-Hourglass model and ResNeXt-101 model	By combining the two models together, the classification accuracy was improved by +1.3% over the baseline
(Chigateri et al., 2022)	System for detecting car models based on machine learning	Stanford cars dataset	<ul style="list-style-type: none"> <li>- Activation function: The activation function used is ReLU(Rectified Linear Unit)</li> <li>- Training Data: The network is trained on nearly 16,000 photos of cars from over 30</li> </ul>	Resnet-34 CNN model	The model achieved 80 percent test accuracy and reduction of training loss from 4.0 to less than one



			distinct automobile models and manufacturers		
(Alghamdi et al., 2023)	Vehicle Classification Using Deep Feature Fusion and Genetic Algorithms	Stanford car dataset that contains almost 196 cars and vehicles	<ul style="list-style-type: none"> <li>- Data augmentation like image flipping and standardizing image sizing with 8 classes each containing 1000 images after augmentation</li> <li>- Guided filter applied to improve colors of images</li> <li>- Feature extraction using 5 different combinations of convolutional layers with each combination containing 2-3 convolution layers.</li> <li>- Feature selection using Genetic Algorithm (GA) Optimization: Used to reduce feature complexity. GA prioritizes important features while discarding less relevant ones</li> <li>- Classification with SVM Kernels: The selected features are classified using various SVM kernels, including Linear SVM, Cubic SVM,</li> </ul>	VGG 16	Achieve accuracy of 99.7%

			Quadratic SVM, and Medium Gaussian SVM		
(Stjepan Ložnjak et al., 2020)	Automobile classification using transfer learning on Resnet Neural Network architecture	Stanford cars dataset	<ul style="list-style-type: none"> <li>- Image Features: Extracting relevant features from car images such as color, shape, texture, and edges.</li> <li>- Transfer Learning: Utilize pretrained models to extract high-level features</li> <li>- Dimensionality Reduction: Applying techniques such as Principal Component Analysis (PCA) or t-SNE to reduce feature dimensionality</li> </ul>	Resnet-152 Neural Network	The model didn't change much. Best achieved accuracy was 88.82% on the 15th epoch
(Hassan et al., 2021)	An Empirical Analysis of Deep Learning Architectures for Vehicle Make and Model Recognition	Stanford cars dataset & VMDB dataset	<ul style="list-style-type: none"> <li>- Data augmentation like random erasing, horizontal flip, random crop, resizing, sharpen, and rotation.</li> <li>- Texture Descriptors: Use texture features like Local Binary Patterns (LBP) or Haralick features to capture surface characteristics</li> <li>- Deep Learning Features: Utilize pre-trained deep</li> </ul>	ResNet-152, Inception-ResNet-v2, Xception, DenseNet-201, MobileNet-v1, DenseNet-121	The model that had the best accuracy was Xception was 92.45% and ResNet-152 was 92%



			learning models example CNN to extract high-level features from car images		
(Shi Hao Tan et al., 2020)	Spatially Recalibrated CNN for Vehicle Type Recognition	BIT-Vehicle Dataset. Consists of 9850 images of different viewpoints	<ul style="list-style-type: none"> <li>- T-distributed Stochastic Neighbor Embedding(TSNE) is used to project 4,096 dimensional features extracted from the penultimate fully connected layer into two dimensions for visual inspection</li> </ul>	CaffeNet	The model achieved an accuracy of 94.17%
(Lu et al., 2022)	A novel part-level feature extraction method for fine-grained vehicle recognition	Stanford Cars & CompCars	<ul style="list-style-type: none"> <li>- Data augmentation included standardizing image sizes and also horizontal flipping</li> <li>- Best results were derived from merging 4th and 5th convolutional layers</li> <li>- Feature grouping module to aggregate the strongly correlated part information of local regions</li> <li>- Feature fusion module to complement multi-scale information of part-level features</li> </ul>	ResNet-101	The proposed model achieved an accuracy of 97.7%

## 2.1 Gap Analysis

From these reviews we can observe several notable gaps in our computer vision-based car model prediction. To begin with, although a variety of feature selection techniques are utilized, there is a lack of exploration into more advanced methods such as automated machine learning (AutoML) for a more meta-learning approach. Such techniques will be necessary to integrate to possibly enhance our feature selection, thus improving the overall model performance. Training splits of the dataset could be introduced to generative models to augment them further.

Moreover, while high accuracy has been recorded in several models such as VGG 16 and DenseNet201, the lack of focus on other performance metrics such as model size, computational efficiency and inference time, may result in a less accurate model as it will not take into account the practical applicability of such models. Exploring lightweight models for deployment on edges could prove to be beneficial. Most of the research focused on broad vehicle recognition tasks instead of real world applications like real-time traffic monitoring. A model that is able to perform multiple related tasks at once could enhance the reliability of the results. Lastly, one prominent gap observed is the lack of ability for the model to recognise new or unseen car models without the aid of intensive additional training. To combat this the use of few-shot and zero-shot learning approaches should be taken into account in more research. By addressing these gaps, future research can significantly advance the field of computer vision-based car model recognition, pushing the boundaries of current methodologies and improving practical deployment in real-world scenarios.

In terms of the models we propose, VGG16 provides better performance in terms of accuracy for image classification tasks. This can be explained by the fact that it uses deeper architecture and more sophisticated feature extraction capabilities. For example, in a study done on a variety of different ImageNet datasets, the Top-5 Accuracy for the models VGG16 and AlexNet were 92.7% and 84.6% respectively. (Krizhevsky et al., 2012) This however, comes at a cost, VGG16 has a much higher computational complexity requirement resulting in longer training and inference times compared to AlexNet. Furthermore, studies that chose to do extensive preprocessing of the dataset like unfreezing classification layers or those who fine tuned higher

level features for better performance of classifiers, personalizing the model to focus on our owned defined classes, improving the overall accuracy of the results.

## **2.2 Scope of research**

By using VGG-16 as our benchmark index, we will assess the performance of both sophisticated neural networks models in this study, which are AlexNet and ResNet-101. In order to verify the validity and effectiveness of these models, we will be carrying out extensive testing and evaluation on a variety of datasets during the course of our assessment procedure. We will be applying several preprocessing methods such as image augmentation and normalization in order to guarantee a comprehensive evaluation. These preprocessing methods are used with the intention of enhancing the data input quality, examining precisely how the methods will affect AlexNet and ResNet-101's output performance in comparison to VGG-16. Besides that, we will look at the effects of various textual representation approaches, evaluating the way variations in data technique and arrangement might impact these models' results. We will be evaluating the effects of the preprocessing methods on AlexNet and ResNet-101 to determine which augmentation techniques work most effectively in enhancing these model's performance and also demonstrate how these methods improve the predictive capability of these models. This comparison of AlexNet and ResNet-101 to VGG-16 will assist in defining the pros and cons of each model, which will guide future studies on neural network architectures along with its applications.

## **3.0 Methodology**

### **3.1 Dataset**

The data set that is seen the most in the literature review above is the Stanford car dataset. Stanford Car dataset consists of 16,185 images with 196 classes of different car models and makes up to the year 2012. Compared to the hundreds of datasets available out there, stanford dataset has a great coverage of a wide range of car make and models, spanning from different conditions and angles to help in the training and testing of specific model architectures. By using

a dataset of this significance, the training and testing will be able to achieve better accuracy and consistency compared to datasets that have fewer images and variations for their images.

## **3.2 Data Preparation**

### **3.2.1 Data Collection**

The Stanford Cars Dataset above is available on multiple sources, including Kaggle, among others. It is ensured to be the right one without any additional sets that can hamper or interfere with our process unnecessarily.

### **3.2.2 Data Preprocessing**

Data preprocessing involves transforming raw data into a format that is suitable for analysis and modeling, and consists of several steps that might or might not be necessary depending on our goal, such as data loading and cleaning, data augmentation, transformation, reduction, and feature engineering (Markus, 2024). One of the reasons why data preprocessing is essential is to ensure that the dataset used in the process is of the highest quality and proper for further processing. Raw data collected at first can have many errors, inconsistencies, incompleteness, and other flaws that can result in improper and misleading results.

Furthermore, raw data collected can have too many dimensions and be too complex for our intended process. It is often high dimensional, noisy, and heterogenous, making it way harder to directly analyze in this stage. This is why data preprocessing can come in handy, by using preprocessing techniques called dimensionality reduction, or feature scaling, so the overall data can be simplified enough and sufficiently complex for proper processing by our model.

Moreover, since our goal in this research is to use computer vision and classification for primarily images, we need to perform the technique of image preprocessing before we carry on. The technique of image preprocessing itself refers to the manipulation and engineering of raw image data given and turning it into a workable and meaningful format for our model. This allows us to remove and get rid of unwanted, unneeded features and distortions, and emphasize on specific attributes and qualities that we want to focus on for our model to process. These are very crucial to be implemented before feeding the image data into the machine learning models.

For all our data preprocessing, The following steps were undertaken sequentially:

### **Data Loading**

- The intended dataset with available images is downloaded to a local machine and loaded into the environment by using appropriate deep learning libraries that are well known and well documented, such as PyTorch or TensorFlow. For example, we can import TensorFlow libraries by ‘import tensorflow’, then we can use ‘tf.keras.preprocessing.image\_dataset\_from\_directory’ and put it into a train dataset variable.
- The image dataset has a certain path in our directory that needs to be located accurately and passed down to the functions to be preprocessed. Also, labels refer to the attributes or classes contained in the dataset that the image belongs to. The function ‘tf.keras.preprocessing.image\_dataset\_from\_directory’ in TensorFlow automatically handles the mapping of the image and their corresponding labels based on the directory structure.

### **Data Augmentation:**

To enhance the robustness of the model, data augmentation techniques were applied to the training images. This included transformations such as:

- Rotation - Rotation of the image refers to turning the image around its center by some degree resulting in different angles. This will help the model to process the image of cars orientation invariantly.
- Horizontal and vertical flipping - Flipping of the image is very beneficial as it can horizontally and vertically flip and change the car’s position, helping the model to learn about the car regardless of side.
- Random cropping and scaling - Random cropping involves selecting a random portion of the image and resizing it back to the desired size. Scaling adjusts the size of the image randomly.

- Color jittering (adjusting brightness, contrast, saturation) - Color jittering involves randomly changing the brightness, contrast, saturation, and hue of the image to make the model robust to color variations.

These techniques help in increasing the diversity of the training data and prevent overfitting.

#### **Image Resizing:**

- All images were resized to a standard dimension (e.g., 224x224 pixels) to ensure uniformity across the dataset. This standardization is necessary for feeding images into convolutional neural networks (CNNs).

#### **Normalization:**

- Pixel values of images were normalized to a range of  $[0, 1]$  by dividing by 255. This helps in speeding up the convergence of the neural network training process.
- Additionally, mean subtraction and standard deviation scaling were applied based on the dataset statistics.

#### **Label Encoding:**

- Car class labels were encoded into numerical values using label encoding techniques. This is essential for training the machine learning model, which requires numerical inputs for categorical variables.

#### **Train-Test Split Verification:**

- The predefined train-test split was verified to ensure there is no data leakage. This ensures that the testing set remains completely unseen during the training process.

More importantly, in data preprocessing, we would have to unfreeze layers and this simply means to remove the classifications layer of pre-trained models as there are 196 car classes in this dataset and we will only need a selected number of classes of models in our assignment. Additionally, it is important to state that the dataset we have chosen have classes in the form of

MMY (model-make-year) but in our assignment we will only be classifying cars based on models, and so any car images that do not fall under the classes we have trained the model under will automatically be classified as “rejection” class and this idea was inspired from one of our literature reviews (Buzzelli & Luca Segantin, 2021).

We can also fine tune our feature extractor by removing the higher feature layers while keeping the pre-trained lower level feature layers so that we can train the model we have chosen to identify the specific car model classes we need for the assignment and this will optimize the performance and accuracy of our classifier which will allow for higher accuracy score of classifying car models.

### **3.2.3 Image Representation and Feature Extraction**

The technique of image representation involves taking the raw pixel values and transforming them into meaningful features that can be utilized by the machine learning models for training. To extract and get these meaningful features, this often can be achieved by techniques such as feature extraction and fine-tuning existing pre-trained models.

#### **Pre-trained Models and Transfer Learning:**

- Pre-trained models refers to machine learning models that are already existing and were trained for specific tasks on specific dataset, and are currently usable. These pre-trained models such as VGG16, ResNet50, InceptionV3 that were trained on large datasets like ImageNet, can then be utilized to extract features from the Cars dataset. This is possible because these pre-trained models have already learned rich feature representations.
- Transfer learning refers to the technique that involves taking a pre-trained model and utilizing it for adaptation to a new dataset. This technique involves removing the final classification layer and replacing it with the new layer adapted to the new dataset, which in this case is suited to the Cars dataset.

#### **Feature Extraction:**

- Feature extraction involves using the convolutional base of a pre-trained model to extract features from the images. These features are then used as input to a new classifier.
- This can be done by removing the top layer of the pre-trained model and using the output of the convolutional layers as feature representations.

### **3.2.4 Data Exploration and Analysis**

#### **Class Distribution:**

- We can analyze the distribution of the attributes or classes so that any imbalances in the dataset can be identified and corrected. The way to do this is to plot the number of images per class and ensure that the images in classes of the datasets are distributed equally.
- Some available techniques such as class-weight adjustment, undersampling, or oversampling can be considered if significant imbalances were detected.

#### **Sample Visualization:**

Random samples or top heads of the rows from the dataset are visualized and presented, so the quality and variety of the images can be inspected, along with their corresponding attributes.

This step is important for us so that we can improve our understanding of the dataset and identify any anomalies, errors, outliers, and hidden features.

## **3.3 Models**

The models selected for this assignment are VGG16 and AlexNet and will be compared to ResNet-50 as the best model from literature review as a benchmark. The reason for ResNet-50 being chosen as the benchmark is that although VGG16 technically has the highest accuracy, ResNet-50 has the most usage.

## **3.4 Model Evaluation**

This section compares and assesses the capabilities of AlexNet and VGG16, two well-known convolutional neural network (CNN) designs. These models have been chosen because of their



historical relevance and broad usage in the computer vision community. The foundational benchmark is AlexNet, which transformed picture classification through its creative application of dropout regularization and ReLU activations. VGG16 provides a more intricate method of feature extraction and is renowned for its deeper architecture and usage of smaller convolutional filters. The objective is to comprehend the different benefits and shortcomings of these architectures through a thorough comparative analysis, with a focus on accuracy, computing efficiency, and suitability for various image recognition applications.

## Introduction to AlexNet

AlexNet is a deep convolutional neural network (CNN) developed in 2012, by Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. It was a pioneering model that made major contributions to the field of computer vision. By obtaining a top-5 error rate of 17.0% and a top-1 error rate of 37.5%, which was a notable improvement over earlier techniques, the model won the 2012 ImageNet Large Scale Visual Recognition Challenge (ILSVRC), garnering major attention and recognition (Chen et al., 2022).

## Architecture of AlexNet

AlexNet is composed of three fully connected layers before five convolutional layers. Convolutional layers employ max pooling for spatial reduction and ReLU activation to extract features using filters with different sizes and strides (Singh et al., 2022). These features are processed for classification by the fully connected layers, which result in an output layer with softmax activation for 1000-class predictions (Chen et al., 2022),

No	Layer Type	Filters	Size/Stride	Output Size
1	Input	-	-	224x224x3
2	Convolutional	96	11x11 / 4 / 0	55x55x96

3	Max Pooling	-	3x3 / 2 / 0	27x27x96
4	Convolutional	256	5x5 / 1 / 2	27x27x256
5	Max Pooling	-	3x3 / 2 / 0	13x13x256
6	Convolutional	384	3x3 / 1 / 1	13x13x384
7	Convolutional	384	3x3 / 1 / 1	13x13x384
8	Convolutional	256	3x3 / 1 / 1	13x13x256
9	Max Pooling	-	3x3 / 2 / 0	6x6x256
10	Fully Connected	-	-	4096
11	dropout	-	-	4096
12	Fully Connected	-	-	4096
13	dropout	-	-	4096
14	Fully Connected	-	-	1000
15	Softmax	-	-	1000

*(Chen et al., 2022)*

## Introduction to VGG16

VGG16 is a popular convolutional neural network architecture used for image classification and feature extraction, developed by the Visual Geometry Group in 2014. Key features of VGG16 include deep architecture with a depth of 16 layers inclusive of the convolutional layers, max pooling layers, and the fully connected layers. VGG16 boasts a simple and uniform structure where all convolutional layers use a 3x3 filter (Albashish et al., 2021). High accuracy is achieved by VGG16 on classification tasks such as ImageNet, while being highly effective for feature extraction which is required for all computer vision tasks. VGG16 is widely used for computer vision tasks because it supports weight sharing or transfer learning with pretrained weights on the ImageNet dataset, allowing the model to adapt to new tasks.

## Architecture of VGG16

Diving deep into the architecture of VGG16 shows that VGG16 comprises 13 convolutional layers and 3 fully connected layers, organized into 5 blocks. Each block is followed by a max pooling layer, with the output for classification being softmax (Albashish et al., 2021).

No	Layer Type	Filters	Size/Stride	Output Size
1	Input	-	-	224 x 224 x 3
2	Convolutional	64	3 x 3 / 1	224 x 224 x 64
3	Convolutional	64	3 x 3 / 1	224 x 224 x 64
4	Max Pooling	-	2 x 2 / 2	112 x 112 x 64
5	Convolutional	128	3 x 3 / 1	112 x 112 x 128

6	Convolutional	128	3 x 3 / 1	112 x 112 x 128
7	Max Pooling	-	2 x 2 / 2	56 x 56 x 128
8	Convolutional	256	3 x 3 / 1	56 x 56 x 256
9	Convolutional	256	3 x 3 / 1	56 x 56 x 256
10	Convolutional	256	3 x 3 / 1	56 x 56 x 256
11	Max Pooling	-	2 x 2 / 2	28 x 28 x 256
12	Convolutional	512	3 x 3 / 1	28 x 28 x 256
13	Convolutional	512	3 x 3 / 1	28 x 28 x 256
14	Convolutional	512	3 x 3 / 1	28 x 28 x 256
15	Max Pooling	-	2 x 2 / 2	14 x 14 / 512
16	Convolutional	512	3 x 3 / 1	14 x 14 / 512
17	Convolutional	512	3 x 3 / 1	14 x 14 / 512
18	Convolutional	512	3 x 3 / 1	14 x 14 / 512
19	Max Pooling	-	2 x 2 / 2	7 x 7 x 512

20	Fully Connected	-	-	4096
21	Fully Connected	-	-	4096
22	Fully Connected	-	-	1000
23	Softmax	-	-	1000 (Classes)

*(Albashaish et al., 2021)*

### **Comparative Summary**

When comparing the performance and depth of both models, AlexNet boasts 8 (5 convolutional, 3 fully connected) layers whereas VGG16 uses 16 (13 convolutional, 3 fully connected) layers. This means that while AlexNet architecture is shallower, this allows the architecture to be simpler. VGG16 has a deeper, and more complex architecture due to the increased number of hidden layers in the network. These architectural differences cause AlexNet to have only basic hierarchical feature extraction capabilities while VGG16 is capable of more advanced hierarchical feature learning. Another performance impact is the accuracy, AlexNet is respectable for its time, but proves less effective on complex datasets, while VGG16 can provide a high accuracy with better performances on more complex datasets.

AlexNet has a lower parameter count than VGG16. It is ~80 million on AlexNet while VGG16 has a parameter count of ~138 million. This lower number of parameters in AlexNet allows for the model to be faster and require less powerful hardware to function. When looking at VGG16 it has a greater training time and requires more powerful hardware in comparison.

Usability of the model is an important aspect when choosing a model for computer vision tasks. AlexNet has effective transfer learning while with less generalizable features when compared with VGG16 which has highly effective transfer learning capabilities with robust feature extraction. Looking at data augmentation support of both models, AlexNet pioneered the use of data augmentation techniques, and VGG16 is commonly used in conjunction with data augmentation. AlexNet has strong historic support from the community and developers while VGG16 has extensive current support and resources.

In conclusion, the choice of VGG16 and AlexNet for this comparative analysis emphasizes the variety and development of convolutional neural network architectures. AlexNet is not as good for sophisticated jobs due to its limited hierarchical feature extraction capabilities and lesser accuracy on complicated datasets, even though its simpler, shallower structure gives advantages in speed and lower hardware requirements. On complicated datasets, VGG16 excels at providing sophisticated hierarchical feature learning and improved accuracy thanks to its deeper design of 16 layers. VGG16 is the recommended option for complex computer vision problems due to its strong feature extraction and excellent transfer learning capabilities, even with its higher processing requirements and longer training durations. Furthermore, the research community's broad support for it now and its compatibility with modern data augmentation approaches reinforce its selection. To end the performance metrics we can use to measure our model's accuracy we can use F1-score and confusion matrix besides our precision and accuracy score.

### **3.5 Summary of methodology**

To summarize our methodology, our model we've chosen is VGG16 and the reason we've chosen it is because it has a deep design of 16 layers and smaller filters which allow it to extract features at a more higher hierarchical level as compared to AlexNet and hence is generally more accurate in classifying classes. Next our data preprocessing process is very crucial since our chosen dataset, Stanford cars dataset has a size of 16,185 images and 196 classes and we need to carry out data augmentation as it helps prevent overfitting during training and helps the model have a more varied understanding of the existing car models in the dataset like for example we can provide cropped version of images at different parts of the car like it's headlights, bumper, car doors, logo or flipped versions and different degrees of rotation so that the model classifying

accuracy is optimized. Afterwards we need to unfreeze the classification layers and fine tune the higher level features so that we can train the model properly to classify.

## **6.0 Conclusion**

In conclusion, based on the Stanford Cars Dataset, we aim to do meticulous data preparation to ensure the best possible input for our machine learning model and address the gaps observed in previous similar research. By exploiting advanced data preprocessing techniques like data augmentation, normalization, label encoding and image resizing in order to make the dataset suitable for an effective train test split. The use of models like VGG16 for its superior feature extraction capabilities to achieve higher accuracy in classifying car models comparatively to its less complex counterpart AlexNet. The systematic approach we have taken in aspects in our methodology such as fine tuning and data selection is all for the purpose of achieving reliable and consistent outcomes in this computer vision task.

## 7.0 References

1. **Revisiting the CompCars Dataset for Hierarchical Car Classification: New Annotations, Experiments, and Results** By Marco Buzzelli, Luca Segantin  
Container: Sensors Publisher: Multidisciplinary Digital Publishing Institute Year: 2021 Volume: 21 Issue: 2 DOI: 10.3390/s21020596 URL: <https://www.mdpi.com/1424-8220/21/2/596>
2. **Improving Car Model Classification through Vehicle Keypoint Localization** By Alessandro Simoni, Andrea D'Eusanio, Stefano Pini, Guido Borghi, Roberto Vezzani  
Container: IRIS UNIMORE (University of Modena and Reggio Emilia) Publisher: University of Modena and Reggio Emilia Year: 2021 DOI: 10.5220/0010207803540361 URL: <https://iris.unimore.it/handle/11380/1229971?mode=complete>
3. **System for detecting car models based on machine learning** By Keerthana B Chigateri, Sujay Suryavamshi, Shrihari Rajendra  
Container: Materials today: proceedings Publisher: Elsevier BV Year: 2022 Volume: 52 DOI: 10.1016/j.matpr.2021.11.335 URL: [https://www.sciencedirect.com/science/article/abs/pii/S2214785321073478?casa\\_token=pqyn3myj4zsAAAAA%3AAanhQ7139N4tYOZZaPC9RmIFVunTpPXoLnIbeAYE3j0BYgC7G\\_5FkZzqVS8ZjTXLxkWFYQCQ06Wa5-oA](https://www.sciencedirect.com/science/article/abs/pii/S2214785321073478?casa_token=pqyn3myj4zsAAAAA%3AAanhQ7139N4tYOZZaPC9RmIFVunTpPXoLnIbeAYE3j0BYgC7G_5FkZzqVS8ZjTXLxkWFYQCQ06Wa5-oA)
4. **End-to-End Car Make and Model Classification using Compound Scaling and Transfer Learning** By Omar BOURJA, Abdelilah MAACH, Zineb ZANNOUTI, Hatim DERROUZ, Hamd AIT ABDELALI, Rachid OULAD HAJ THAMI,



**Francois BOURZEIX Container: International Journal of Advanced Computer Science and Applications Year: 2022 Volume: 13 Issue: 5 DOI: 10.14569/ijacsa.2022.01305111 URL:**

**[https://thesai.org/Downloads/Volume13No5/Paper\\_111-End\\_to\\_End\\_Car\\_Make\\_and\\_Model\\_Classification\\_using\\_Compound\\_Scaling.pdf](https://thesai.org/Downloads/Volume13No5/Paper_111-End_to_End_Car_Make_and_Model_Classification_using_Compound_Scaling.pdf)**

5. **Automobile classification using transfer learning on ResNet neural network architecture By Stjepan Ložnjak, Tin Kramberger, Ivan Cesar, Renata Kramberger Container: Polytechnic and design Year: 2020 Volume: 8 Issue: 01 DOI: 10.19279/tvz.pd.2020-8-1-18 URL: <https://hrcak.srce.hr/clanak/352672>**
6. **Vehicle Classification Using Deep Feature Fusion and Genetic Algorithms By Ahmed S Alghamdi, Ammar Saeed, Muhammad Kamran, Khalid T Mursi, Wafa Sulaiman Almukadi Container: Electronics Publisher: Multidisciplinary Digital Publishing Institute Year: 2023 Volume: 12 Issue: 2 DOI: 10.3390/electronics12020280 URL: <https://www.mdpi.com/2079-9292/12/2/280>**
7. **An Empirical Analysis of Deep Learning Architectures for Vehicle Make and Model Recognition By Aqsa Hassan, Mohsin Ali, Nouman M Durrani, Muhammad Atif Tahir Container: IEEE access Publisher: Institute of Electrical and Electronics Engineers Year: 2021 Volume: 9 DOI: 10.1109/access.2021.3090766 URL: <https://ieeexplore.ieee.org/document/9460843>**
8. **Hierarchical System for Car Make and Model Recognition on Image Using Neural Networks By Ivan Fomin, Ivan Nenahov, Aleksandr Bakhshiev Year: 2020 DOI: 10.1109/icieam48468.2020.9112026 URL: <https://ieeexplore.ieee.org/document/9112026>**

**9. A novel part-level feature extraction method for fine-grained vehicle recognition**

By Lei Lu, Ping Wang, Yijie Cao Container: Pattern recognition Publisher: Elsevier  
BV Year: 2022 Volume: 131 DOI: 10.1016/j.patcog.2022.108869 URL:  
[https://www.sciencedirect.com/science/article/abs/pii/S0031320322003508?casa\\_token=Pc9CuvTwBvIAAAAA%3AFfgIEwwyePcyuPuzbokRxs6tIIVlU8LTS\\_OuqpQH\\_WiTd0zuXXmCdIlgFkgW0HZwQP0JIFEDKIAOMgw](https://www.sciencedirect.com/science/article/abs/pii/S0031320322003508?casa_token=Pc9CuvTwBvIAAAAA%3AFfgIEwwyePcyuPuzbokRxs6tIIVlU8LTS_OuqpQH_WiTd0zuXXmCdIlgFkgW0HZwQP0JIFEDKIAOMgw)

**10. Learning to locate for fine-grained image recognition**

By Jiamin Chen, Jianguo Hu, Shiren Li Container: Computer vision and image  
understanding Publisher: Elsevier BV Year: 2021 Volume: 206 DOI:  
10.1016/j.cviu.2021.103184 URL:  
<https://www.sciencedirect.com/science/article/abs/pii/S107731422100028X>

**11. Hybrid quantum ResNet for car classification and its hyperparameter optimization**

By Asel Saginalieva, Mo Kordzanganeh, Andrii Kurkin, Artem Melnikov, Daniil  
Kuhmistrov, Michael Perelshtein, Alexey Melnikov, Andrea Skolik, David Von  
Dollen Container: Quantum Machine Intelligence/Quantum machine intelligence  
Publisher: Springer Science+Business Media Year: 2023 Volume: 5 Issue: 2 DOI:  
10.1007/s42484-023-00123-2 URL:  
<https://link.springer.com/article/10.1007/s42484-023-00123-2>

**12. Deep CNN Model based on VGG16 for Breast Cancer Classification**

By Dheeb Albashish, Rizik Al-Sayyed, Azizi Abdullah, Mohammad Hashem Ryalat,  
Nedaa Ahmad Almansour Year: 2021 DOI: 10.1109/icit52682.2021.9491631 URL:  
<https://ieeexplore.ieee.org/document/9491631>

13. AlexNet Convolutional Neural Network for Disease Detection and Classification of Tomato Leaf By Hsing-Chung Chen, Agung Mulyo Widodo, Andika Wisnujati, Mosiur Rahaman, Jerry Chun-Wei Lin, Liukui Chen, Chien-Erh Weng Container: Electronics Publisher: Multidisciplinary Digital Publishing Institute Year: 2022 Volume: 11 Issue: 6 DOI: 10.3390/electronics11060951 URL: <https://www.mdpi.com/2079-9292/11/6/951>
  
14. AlexNet architecture based convolutional neural network for toxic comments classification By Inderpreet Singh, Gulshan Goyal, Anmol Chandel Container: Journal of King Saud University. Computer and information sciences/Mağalaĩ ġam'aĩ al-malĩk Saud : ùlm al-ħasib wa al-ma'lumat Publisher: Elsevier BV Year: 2022 Volume: 34 Issue: 9 DOI: 10.1016/j.jksuci.2022.06.007 URL: <https://www.sciencedirect.com/science/article/pii/S1319157822002026?via%3Dihub>
  
15. Car make and model classification from image By Alexandros Kelaiditis Container: ResearchGate Publisher: unknown Year: 2023 URL: [https://www.researchgate.net/publication/372159544\\_Car\\_make\\_and\\_model\\_classification\\_from\\_image](https://www.researchgate.net/publication/372159544_Car_make_and_model_classification_from_image)
  
16. Why Data Preprocessing is Necessary in Data Science  
  
By Gideon Markus Container: Medium Publisher: Medium Year: 2024 URL: <https://medium.com/@gideonmarkus/why-data-preprocessing-is-necessary-in-data-science-546235345fdb#:~:text=One%20of%20the%20primary%20reasons,and%20lead%20to%20erroneous%20conclusions>
  
17. ImageNet Classification with Deep Convolutional Neural Networks  
  
By Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton Container: Communications of the ACM Year: 2012 Volume: 60 Issue: 6 DOI: 10.1145/3065386 URL:

[https://proceedings.neurips.cc/paper\\_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf](https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf)

18. Vehicle Type Recognition Algorithm Based on Improved Network in Network

By Erxi Zhu, Min Xu, De Chang Pi Container: Complexity Publisher: Hindawi  
Publishing Corporation Year: 2021 Volume: 2021 DOI: 10.1155/2021/6061939 URL:  
<https://onlinelibrary.wiley.com/doi/10.1155/2021/6061939>

19. How to Measure Model Performance in Computer Vision: A Comprehensive Guide

By Zoumana Keita Container: Encord.com Publisher: Encord Blog Year: 2023 URL:  
<https://encord.com/blog/measure-model-performance-computer-vision/#:~:text=Classification%20models%20can%20be%20evaluated,%2Dscore%2C%20and%20confusion%20matrix>