Vehicle Detection Using CNN Models

MD Al Arman Sorker (200101059), Abu Shadat Shaikat (200101057),
Monisha Bakshi (200101055) and Aznur Azam (200101037)

Course Code: CSE 4132 Course Title: Artificial Neural Networks and Fuzzy Systems Sessional
Semester: Winter 2023
*Department of Computer Science and Engineering, Bangladesh Army University of Science and Technology (BAUST)

Abstract

In this report we are using different CNN models for vehicle detection and analysis the result among the models. Here we shows how we work with the dataset in the methodology and showing the performance of the models in Result & Performance Analysi.

1. Introduction

Vehicle detection is an important task in computer vision with various practical applications such as traffic monitoring, autonomous driving, and security systems. In recent years, deep learning techniques have achieved improved performance in vehicle detection tasks. Convolutional Neural Networks (CNNs) are particularly effective at learning features from images and making predictions based on those features.

In this report, we present our work on vehicle detection using CNN models. Our goal is to create a model that can accurately classify an image as a vehicle or not. To achieve this goal, we first collected a dataset of non-vehicular images and vehicular images. We then trained and evaluated several CNN models on this dataset to determine the most efficient architecture for our task.

2. Literature Review

Vehicle detection is a crucial task in the field of computer vision with numerous applications such as surveillance, autonomous driving, and traffic control. In recent years, Convolutional Neural Networks (CNNs) have shown remarkable success in object detection tasks, including vehicle detection.

One of the early works in vehicle detection using CNNs is the paper by Wang et al. (2016) [1], where a deep CNN model was trained to detect vehicles in aerial images. The authors achieved a high accuracy of 97.29% using their proposed model. Another study by Karimian et al. (2018) [2] proposed a vehicle detection system that combines a deep CNN model with a clustering algorithm. The authors showed that their proposed method outperforms traditional vehicle detection methods.
Recently, transfer learning, which involves using pre-trained CNN models for vehicle detection, has become a popular approach. In a study by Kim et al. (2018) [3], the authors

used a pre-trained VGG16 model for vehicle detection in traffic camera images. The authors fine-tuned the last few layers of the VGG16 model and achieved a high accuracy of 97.6%. A similar approach was used by Chen et al. (2020) [4], where a pre-trained ResNet50 model was fine-tuned for vehicle detection in surveillance videos.

In summary, CNN-based models have shown great potential for vehicle detection, and transfer learning has emerged as a popular approach for this task. However, there is still room for improvement in terms of accuracy and speed, and future research could focus on developing more efficient CNN architectures and incorporating advanced techniques such as attention mechanisms and reinforcement learning.

3. Dataset Description

The Vehicle-Detection dataset is a collection of images used for training and testing of machine learning models for vehicle detection. The dataset contains three sets: training set, test set, and validation set. The training set consists of 12,971 images, the test set has 2,774 images, and the validation set contains 2,025 images. Each image in the dataset is labeled as either a 'vehicle' or 'non-vehicle', making it a binary classification problem. The images have a resolution of 64x64 pixels and are in RGB color format. This dataset can be used for training, testing and evaluating machine learning models for vehicle detection tasks.

4. Methodology

We use CNN models with different pre-trained techniques (Baseline, Transfer Learning and Fine Tuning) in DenseNet121, ResNet50, VGG16 and MobileNet CNN models.

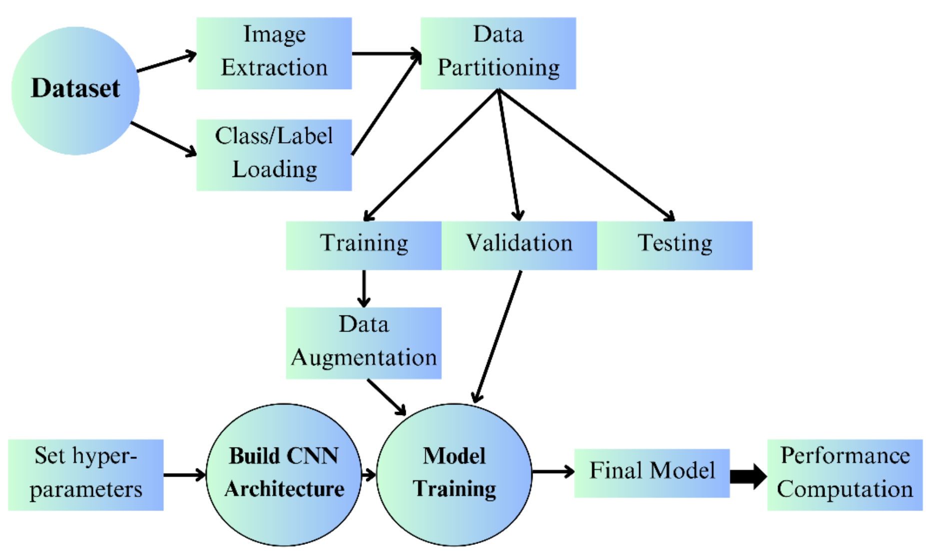


Figure 4.1: Methodology

5. Result and Analysis

We use four CNN Models with baseline, transfer learning and fine tuning techniques and they give different results and performance.

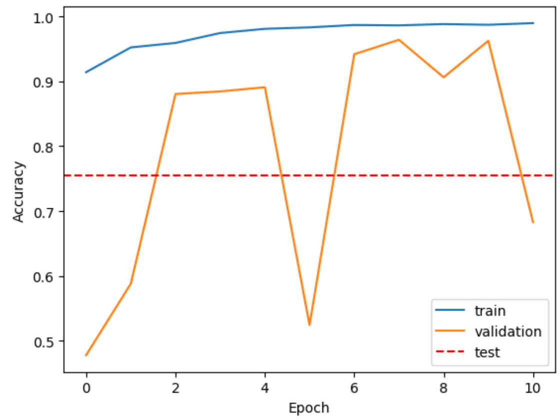
<table>
<tr><th>Model</th><th>Techniques</th><th>Performance</th></tr>
<tr><td rowspan="3">DenseNet</td><td>Baseline</td><td>

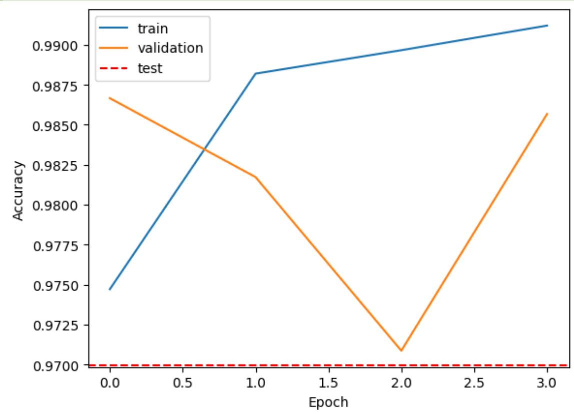
Figure 5.1: DenseNet Baseline Model</td></tr>
<tr><td>Transfer Learning</td><td>

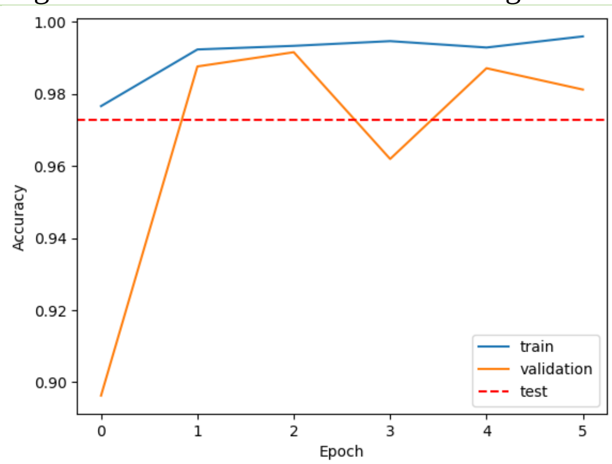
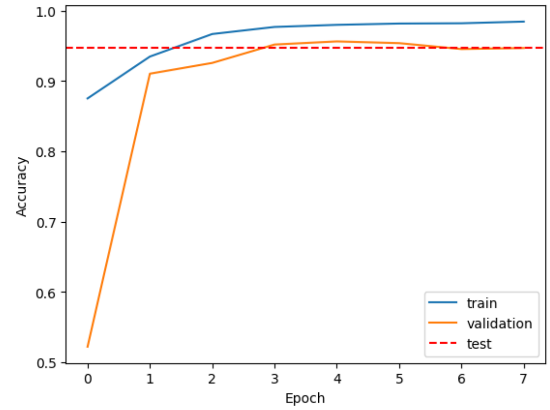
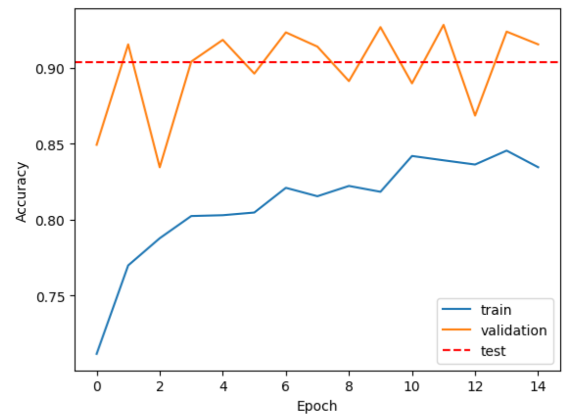
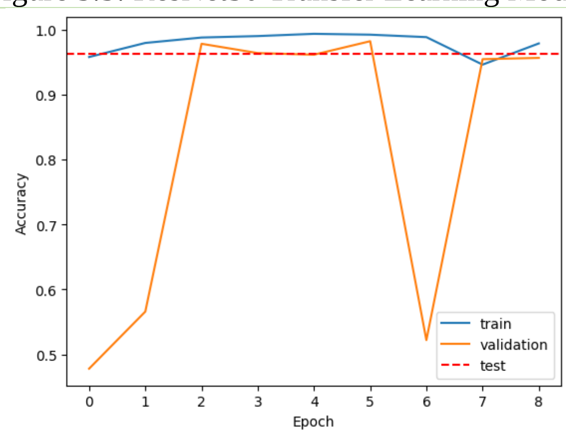
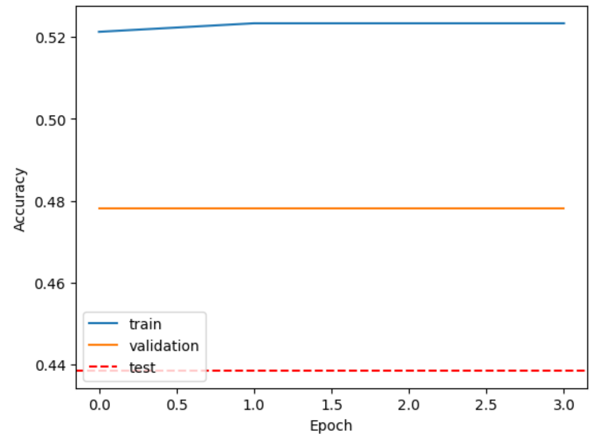
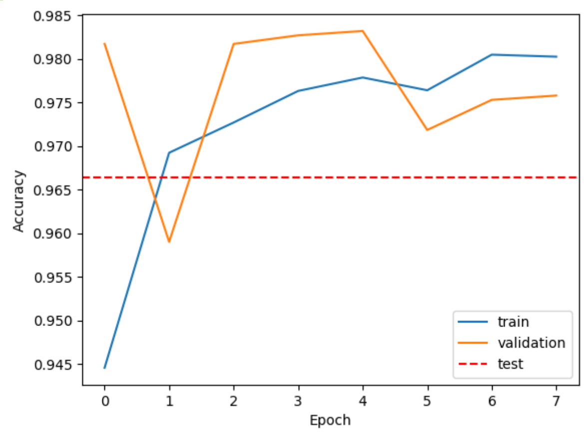
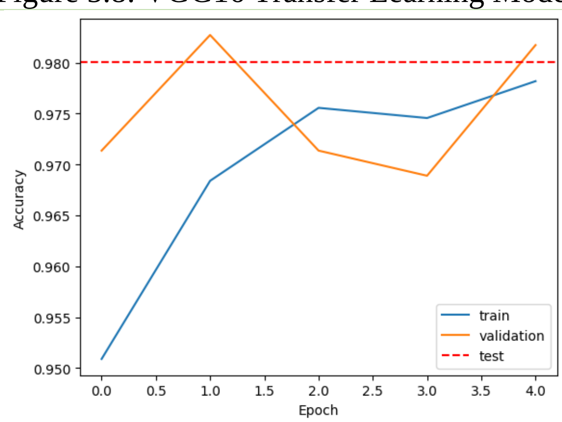
Figure 5.2: DenseNet Transfer Learning Model</td></tr>
<tr><td>Fine Tuning</td><td>

Figure 5.3: DenseNet Fine Tuning Model</td></tr>
<tr><td colspan="3">The DenseNet Transfer Learning gives better performance among the DenseNet Baseline, Transfer Learning and Fine Tuning techniques.</td></tr>
</table>

ResNet50	Baseline	 Figure 5.4: ResNet50 Baseline Model
	Transfer Learning	 Figure 5.5: ResNet50 Transfer Learning Model
	Fine Tuning	 Figure 5.6: ResNet50 Fine Tuning Model
The ResNet50 Baseline gives better performance among the ResNet50 Baseline, Transfer Learning and Fine Tuning techniques.		

VGG16	Baseline	 Figure 5.7: VGG16 Baseline Model
	Transfer Learning	 Figure 5.8: VGG16 Transfer Learning Model
	Fine Tuning	 Figure 5.9: VGG16 Fine Tuning Model
The VGG16 Transfer Learning gives better performance among the VGG16 Baseline, Transfer Learning and Fine Tuning techniques.		

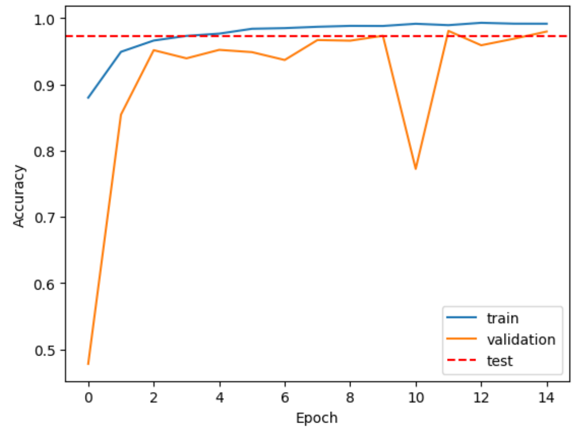
<table>
<tr><td>MobileNet</td><td>Baseline</td><td>

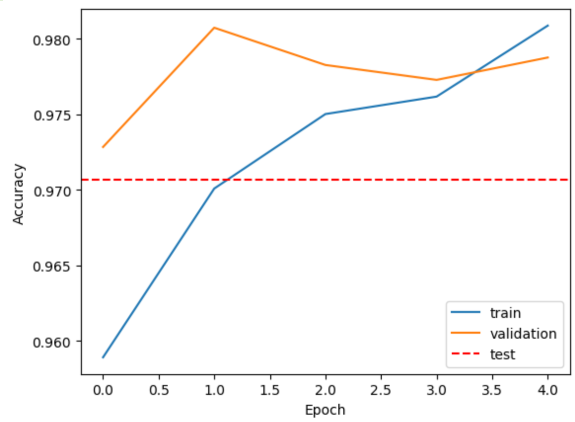
Figure 5.10: MobileNet Baseline Model</td></tr>
<tr><td></td><td>Transfer Learning</td><td>

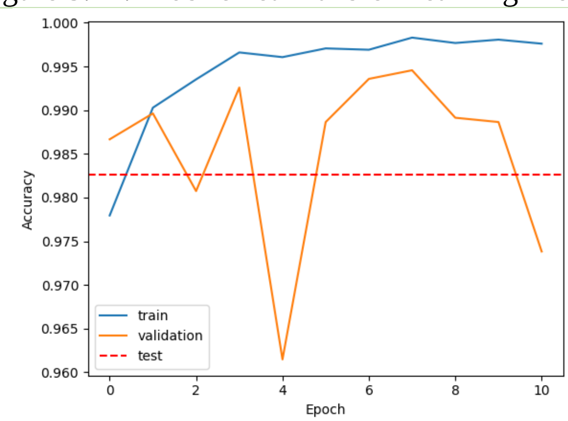
Figure 5.11: MobileNet Transfer Learning Model</td></tr>
<tr><td></td><td>Fine Tuning</td><td>

Figure 5.12: MobileNet Fine Tuning Model</td></tr>
<tr><td colspan="3">The MobileNet Baseline gives better performance among the MobileNet Baseline, Transfer Learning and Fine Tuning techniques.</td></tr>
</table>

The VGG16 Transfer Learning Model gives best performance than DenseNet, ResNet50 and MobileNet and gives best result.

6. Stakeholders

Some stakeholders for our project could be

1. Traffic monitoring department,
2. Autonomous driving company,
3. Security systems department

7. Issues Encountered

We face some issue by working in this project. The dataset didn't have train, test, validation set. So, we need to split the data into train, test and validation. But after splitting the data into train, test, validation we were facing problem in data augmentation while using flow() function instead of flow_from_directory() from keras. After so many attempt, we failed to use the flow() function for data augmentation. So, we manually make a new dataset using the old one which have train, test and validation folder and for this we are able to use flow_from_directory() function for data augmentation.

8. Conclusion, Limitations and Future Recommendations

In conclusion the Convolutional Neural Networks is very useful to extract features from images and analysis the images. In this project we are trying to learn different CNN models and use it in our project to see which can give us better result.

We also have some limitation in our project:

1. The dataset doesn't have much images of vehicles, so if there are new vehicles the model will give wrong result.
2. The model can only tell is there any vehicle or not. It cannot tell which vehicle is there.

Future Work:

1. Realtime images from CCTV could be take and test the result.
2. Make a better dataset with more and details images for training, testing and validation

References

[1] Wang, Y., Huang, J., Yuan, Z., & Ding, X. (2016). Aerial vehicle detection from UAV imagery using deep convolutional neural networks. Remote Sensing, 8(4), 342.

[2] Karimian, N., Karimian, A., Sattarivand, M., & Rahimi, M. (2018). A clustering-based approach for vehicle detection in aerial images using convolutional neural network. In 2018 5th International Conference on Control, Decision and Information Technologies (CoDIT) (pp. 583-588). IEEE.

[3] Kim, M., Kim, Y., & Kim, M. (2018). Vehicle detection in traffic camera images using a pre-trained deep neural network. Sensors, 18(4), 1234.
[4] Chen, Y., Li, K., Peng, Q., & Yang, J. (2020). Real-time vehicle detection using deep learning with GPU acceleration for traffic surveillance systems. Sensors, 20(7), 2011.
[5] Vehicle-Detection dataset - https://www.kaggle.com/datasets/mdalarmansorker/vehicle-detection

Appendix

Attainment of Complex Engineering Problem (CP)

S.L.	CP No.	Attainment	Remarks
1.	P1: Depth of Knowledge Required		K3 (Engineering Fundamentals): Require knowledge of Machine Learning.
			K4 (Engineering Specialization): Require knowledge of different machine learning models and how to use them.
			K5 (Design): Flow diagram shows the methodology.
			K6 (Technology): Google Colab
			K8 (Research): Studied related application to find limitation.
2.	P2: Range of Conflicting Requirements	Yes	Machine Learning Models
3.	P3: Depth of Analysis Required	Yes	Read papers for research.
4.	P4: Familiarity of Issues	Yes	Working in this field as CSE Student
5.	P5: Extent of Applicable Codes	No	
6.	P6: Extent of Stakeholder Involvement and Conflicting Requirements	No	
7.	P7: Interdependence	No	

Mapping of Complex Engineering Activities (CA)

S.L.	CA No.	Attainment	Remarks
1.	A1: Range of resources	No	

2.	A2: Level of interaction	Yes	Understand machine learning models working procedures and then apply them.
3.	A3: Innovation	No	
4.	A4: Consequences for Society and the Environment	Yes	Can use in traffic monitoring system and help the traffic department to reduce the traffic.
5.	A5: Familiarity	Yes	Working in this field as a CSE student.