International Journal of Applied Engineering Research ISSN 0973-4562 Volume 15, Number 4 (2020) pp. 372-376

Efficiency - Optimized Approach - Vehicle Classification Features Transfer Learning and Data Augmentation Utilizing Deep Convolutional Neural Networks

V.Sowmya*[1], R. Radha[2]

[1]*Research Scholar, Research Dept. of Computer Science, SDNBV College for Women, University of Madras, Chrompet, Chennai-600044, India*

[2]*Associate Professor, Research Dept. of Computer Science, SDNBV College for Women, Chrompet, Chennai-600044, India.*

**Corresponding Author (V.Sowmya)*

Abstract

Vehicle Classification represents an essential function in the Traffic Management System. In recent years, predominantly, the Deep Convolutional Neural Network algorithms are widely adopted for object classification and detection. Accordingly, in this paper, transfer learning-based vehicle classification exercising pre-trained Deep Convolutional models such as VGG16, InceptionV3 are proposed. To reduce the over-fitting problem of Deep Convolutional Neural Networks, on minimum-capacity datasets Transfer Learning and Data Augmentation methods are enabled in this proposed system. The performance of the model is tested, premised on the experiments on the custom dataset of vehicle images. In this study, the classification and detection algorithm seeks three classes of vehicles such as bus, truck, and motorcycle. Consequently, the experimental outcomes reveal that compared with the VGG16 model, the classification accuracy of the pre-trained model is higher by implementation of the InceptionV3 model. The InceptionV3 model with an optimized approach achieves classification accuracy of 99.33% for the training set and 98.87% for the validation set, which governs improvement in accuracy of detection.

Keywords: Vehicle Classification, Deep Convolutional Neural Network, Transfer Learning, Data Augmentation.

1. INTRODUCTION

Image Classification is the cynosure of the Image Processing technique; In real-time applications, Preponderantly, the Deep Learning models are successfully implemented. Nowadays, Deep Learning systems models a tremendous impact on the improvement of classification problems. In Deep Learning, Convolutional Neural Networks are inclined to the efficient analysis of images through the elimination of the manual technique of feature extraction, and directly extract features from raw data. This automatic feature extraction is an accurate learning model to classify the objects.

In much recent research, the deployment of Image Classification has the application of various techniques. In recent years, the Convolutional Neural Network (CNN)[1] evolved as the primary method of various Computer Vision tasks [2]. Mengying shu [3] proposes the system to pre-trained the deep model with proper modification and can be used to fit the model into small dataset without severe over fitting. The experiment results show that the classification of accuracy of 96% is achieved.

Haijian ye [4] et al. proposes a image recognition method on the pre-trained model optimizes the fully connected layer and replaces the softmax classifier and tested them on the self-expanding dataset of Vegetable Pest Images. The experiment proves that the test accuracy of 99.99%. K.S. Anand [5] et al. proposed a system on the Transfer Learning-Based Machine Learning classification system, they leverage the rich features in CNN and propagate into an artificial neural network using Transfer Learning. The experiment result determines that the accuracy of 72% mAP. Muthukrishnan Ramprasath [6] et al. proposes a model using the MNIST benchmark dataset for classifying images based on CNN. The experiment shows that the model has obtained 98% of accuracy. Sajja Tulasi Krishna [7] et al. provides a brief survey on the Deep Learning model. The proposed system is configured and analyzed with the most popular dataset. This survey concludes the dependency on the GPU of the system for the performance of the Deep Learning techniques. Srikanth Tammina [8] proposed a system that uses single pre-trained model VGG16 and compared this with basic CNN using Transfer Learning and Data Augmentation. The experiment shows that VGG16 as obtained 95.40% of accuracy. Manali Shaha [9] et al. proposes a system that make use of Transfer Learning to fine-tune the parameters of pre-trained VGG19 for Image Classification tasks, and the two databases GHIM10K and CalTech256 are used for robust feature extraction. Experiment results show that fine-tuned VGG19 architecture outperforms the other CNN, and Hybrid Learning approach for Image Classification tasks. Mahbub Hussain [10] et al. proposes a study on Image Classification using CNN. For the experiment, the two benchmarked datasets CIFAR-10 and CalTech are structured to train on the CNN framework, and the results are compared with state-of-the-art approaches.

This paper presents a Vehicle Classification method based on pre-trained Deep Models such as VGG16, InceptionV3. The principal objective of this proposed system is to Pre-Train the Deep Model with adequate modification of efficiency-oriented system parameters, and to enhance the probability of the utilization of the model into minimum-capacity datasets enabled with the optimization of existing limitations of over-fitting. This proposed model uses the Image Augmentation and Transfer Learning techniques in pre-trained VGG16 and InceptionV3 to classify images. In previous papers, we have identified the VGG16 achieves higher accuracy, but in this proposed system InceptionV3 achieves higher accuracy compared to VGG16, because the parameters are fine-tuned, and by applying the Image Augmentation the Over-Fitting is reduced, which leads to improvement in the accuracy. Finally, Image Recognition results by using the Computer Vision technique. In Section 2, Material and Methods are explained. Experimental Analysis and Results are given in Section 3, and Section 4 concludes the paper.

2. MATERIALS AND METHODS

2.1 Convolutional Neural Networks (CNN)

Convolutional Neural Network (CNN) is a special kind of Multi-Layer Neural Networks (MLNN), and a version of the Back-Propagation algorithm is configured to the MLNN like any other Neural Network. This trained Neural Network is designed to recognize Visual Patterns directly from pixel images with minimal pre-processing. CNN reduces dimensionality and extracts features to perform image classification with the help of Multi-Layers. The CNN architecture model is shown in Figure1

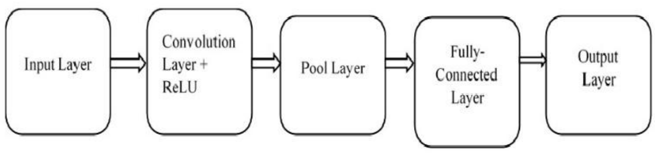


Figure 1: A CNN architecture model

A CNN architecture contains the following layers: Input Layer, Convolution Layer, Pool Layer, Fully Connected Layer, and Output Layer. The first three phases are called Feature Extraction phases, and the last two phases are called Classification Phases.The image processing begins with the Input Layer and completes in Output Layer. The Input Layer is populated with raw images and is propagated to subsequent layers for features extraction. A number of filters are tested on images for finding their features in the Convolutional Layer. These captured features direct the measurement of match at the testing stage. ReLU (Rectified-Linear Unit) layer applies the ReLU function as activation function max (0, x), which turns negative values to zeros. This layer does not change the size of the volume, and there are no Hyper Parameters. This activation function is very useful in CNN.Pool Layer perform a function to reduce the spatial dimensions of the input and the computational complexity of the model. Also, it controls overfitting. There are various functions such as average pooling, max pooling, or L2-norm Pooling. However, Max pooling is the most used type of pooling, which only takes the most important part (the value of the brightest pixel) of the input volume. The Fully-connected layer is the final layer, which captures high-level filtered images and converts them into labels with categories. The last fully-connected layer uses a SoftMax activation function for classifying the generated features of the input image into several classes based on the training class. It provides the decimal probabilities for each class. The decimal probabilities are between 0 and 1.

2.2 Transfer learning

Transfer Learning is a prominent method in Machine Learning and Computer Vision, which uses extant knowledge to solve problems differently, but not effective with the associated domain. Its goal is to complete the transformation of different knowledge between their associated domains. Transfer Learning is the effective application of "knowledge" trained on existing datasets in advanced fields.

In Deep Learning, Transfer Learning can be successfully trained with pre-trained models. A pre-trained model is a model that has been trained with a large benchmark dataset along with their weights. Several pre-trained models such as VGG16, Resnet50, InceptionV3 are used in Transfer Learning are based on Convolutional Neural Networks. Therefore, train the new network with pre-trained models along with their weights by using the following ways in the proposed system. There are two ways to customize the pre-trained model using Transfer Learning:

(i) Feature Extraction: Extract features from new datasets by previous base Convolutional Neural networks for classifying images. A new classifier is added on the top of the pre-trained model which has been trained from scratch. Therefore, the feature maps are reused from previously learned for the dataset.

(ii) Fine-Tuning: A few top layers of a frozen model base should unfreeze and jointly train the newly-added classifier layers as well as the final layers of the base model. This help us to "fine-tune" the representations of the higher-order features in the base model to make them more appropriate for the specific task.

2.3 Data Augmentation

Image data augmentation is a popular technique for improving performance and preventing overfitting. The Keras Deep Learning Neural Network library provides the capability to fit models using Image Data Augmentation through the Image Data Generator class. Image Data Generator converts image files to pre-processed tensors, which can be directly fed into a training dataset to train the model, by using various parameters such as rotating, shifting, zooming, shearing and flipping images.

2.4 VGG16

The VGG16 architecture was first introduced by Simonyan, Zisserman in the year 2014. VGG16 model is composed of 13 convolutional layers along with the max pooling, the two fully connected layers, and a SoftMax classifier. The large size kernels are replaced with the multiple numbers of the 3x3 filters in this network to extract complex features at a low cost.

2.5 InceptionV3

The InceptionV3 architecture was first introduced by Szegedy et al. in the year 2015. The original paper "Rethinking the InceptionV3 architecture for Computer Vision" laid the foundation for the advancement of this model. This Model is composed of 42 Convolutional Layers deeper with fully connected layers, and a SoftMax classifier. The InceptionV3 model act as a "Multi-Level Feature Extractor" by computing 1x1, 3x3, and 5x5 convolutions, increased when moving to higher layers.

2.6 Dataset

The custom vehicle dataset contains 5500 images of bus, truck, and motorcycle. The vehicle dataset acquires the images in icrawlers, is in use in this system. Also, some of the images are downloaded from the internet (not containing copyright issues) to test varying conditions. For Training, Validation, and Testing, the Allocation of Images is 3000, 1500, and 1000 respectively. ImageNet, a benchmarked large dataset, is adopted. VGG16 and InceptionV3 are the pre-trained models are implemented in this proposed system to qualify the smaller vehicle dataset.

3. EXPERIMENT ANALYSIS AND RESULTS

The experiment is carried out in Anaconda3, which uses the open source deep learning framework Keras as the development environment. In this paper, the Loss curve and the accuracy curve of the experimental results are drawn by using Matplot library to analyse the convergence of convolutional neural network.

3.1 Basic Convolutional Neural Network

Initially, we create a basic convolutional network to train the minimum-capacity vehicle dataset and evaluate the

model. The basic architecture of the Convolutional Neural Network involves two Convolutional Layers with the Kernel, size 3x3, stride size 1x1, and Activation Function as ReLU. These layers combined with 2D Maxpooling of size 2x2. The initial two Convolutional Layers serve as Feature Extraction. The output is fed to a dense Fully-Connected layer with a 0.5 dropout, then add the classifier on top of these layers, and categorical cross-entropy is used as an optimizer to make the final prediction of vehicles. From Table 1, the basic CNN performance is evident that it is sub-optimal since it achieves only 85.90% validation accuracy and 90.43% training accuracy leading to overfitting. Inorder to overcome these challenges the proposed system utilizes pre-trained models to improve accuracy for the minimum-capacity vehicle dataset.

Based on a pre-trained convolutional neural network, a model for identifying three types of vehicle images is developed as shown in Figure 2.

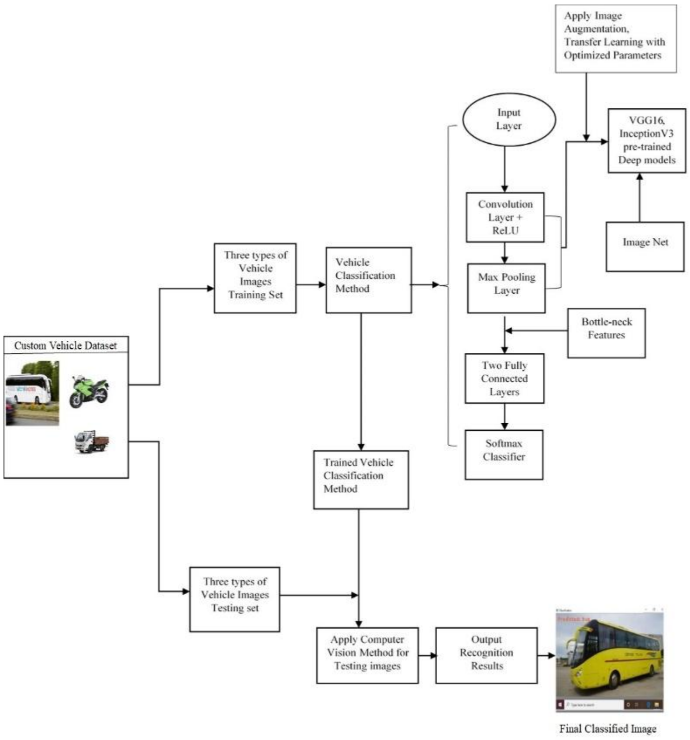


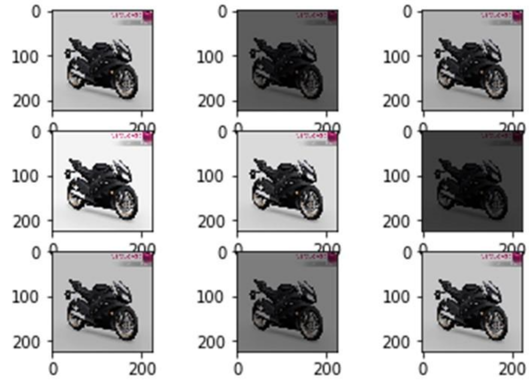
Figure 2: Vehicle-Classification Framework

3.2 Pre-trained models with Feature Extraction

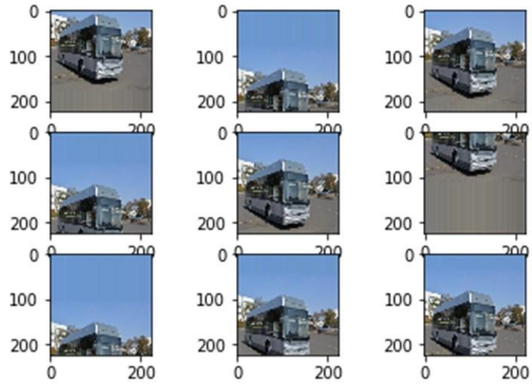
The pre-trained models such as VGG16, InceptionV3 are loaded from the ImageNet database with their corresponding weights. The pre-trained models are trained on 14 million images with 1000 categories to recognize the image. To achieve reasonably high accuracy with only a few training images, a technique called "Bottleneck features" is applied on minimum-capacity vehicle dataset . Bottleneck Features are the last Activation Maps that will be added before the Fully-Connected layer. The Input size of the image is 224x224x3, and the final feature map for VGG16 and InceptionV3 are 512x7x7 and 2048x5x5. The summary is displayed for each model to see the layers and feature maps. Then add two densely connected classifiers with dropout 0.5 on top of these layers, and categorical cross-entropy is used as an optimizer. Finally, train the custom vehicle training set from scratch, and the optimal model parameters of the model are obtained as the learning rate is 0.0001, the number of epochs is 20, the batch size is 20. The Feature Map is used as input in the Fully-Connection layer to get the classification results. The performance of these pre-trained models is better than basic Convolutional Neural Network but overfitting is not reduced. Thus, image Data Augmentation and Fine-Tuning parameters are necessary for these models.

3.3 Pre-trained models with Image Data Augmentation and Fine-Tuning

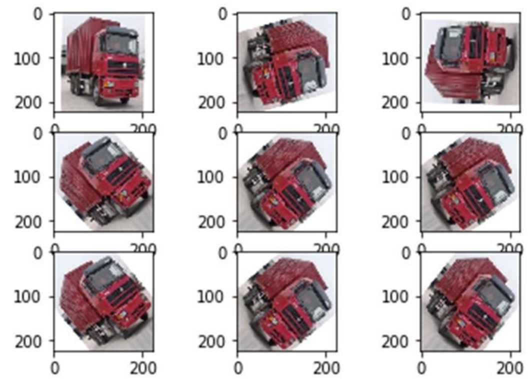
To enhance the vehicle classification further and to reduce overfitting, the training data is augmented with pre-trained models which is shown in Figure 3. For VGG16 model, the last four layers are unfreeze containing convolution layers, max pooling layers and used for feature extraction. Then, train these layers with two fully densely connected classifier layer with dropout 0.5% and categorical cross entropy is used as an optimizer. For Inception V3, mixed8 and mixed9 layer is chosen with many layers are used for feature extraction. Train these layers with two fully densely connected classifier layer with dropout 0.5% and categorical cross-entropy is used as an optimizer. Thus, the performance is improved and overfitting is reduced.



(a)



(b)



(c)

Figure 3: Image Data Augmentation with various parameters (a) Brightness (b) Height flip (c) Rotation

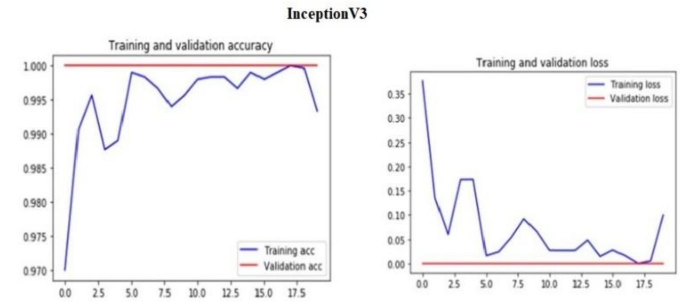


Figure 4: Accuracy and Loss function metric plot of InceptionV3 model

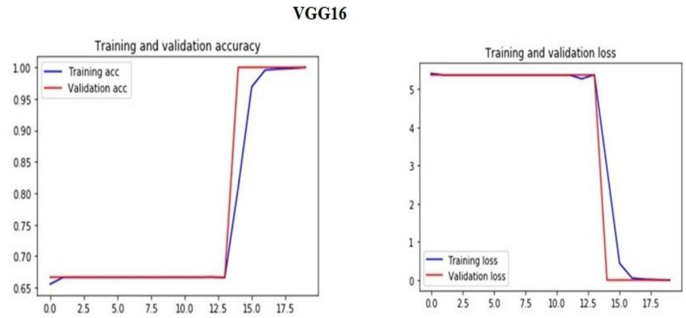


Figure 5: Accuracy and Loss function metric plot of VGG16 model

Table 1: Performance Comparison of Deep-Learning Models

Models	Training Accuracy %	Validation Accuracy %	Training Loss	Validation Loss
Basic Convolutional Neural Networks	90.43%	85.90%	5.2629	7.1875
Fine-tuning with VGG16 model and Image Augmentation	97.91%	96.77%	0.1733	1.1921
Fine-tuning with Inceptionv3 model and Image Augmentation	99.33%	98.87%	0.0098	0.1852

From Table 1, Figure 4 and Figure 5 shows the training and validation accuracy and loss function for different neural network models. The initial model is built using the Convolutional Neural Network, which gives training, and

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 15, Number 4 (2020) pp. 372-376

validation accuracy of 90.43%, and 85.90%. The pre-trained models such as VGG-16 and InceptionV3 are trained on a minimum-capacity dataset of images and fine-tuned with image augmentation to achieve accuracy of 97.91% and 99.33% respectively. The experimental results showed that compared with VGG16, the classification accuracy of InceptionV3 is higher.

Finally, the Computer vision technique is used for testing the classification model. The experiment shows that the proposed method is tested for variable input, of images, which may not be included in the dataset and the prediction results of bus, truck and motor-bike is shown in Figure 6.

Figure 6: Various Vehicle-Classification Results

4. CONCLUSION AND FUTURE WORK

In this study, Deep Models like VGG16, Inception v3 that is configured for handling smaller size custom datasets. All of these models are pre-trained on ImageNet. According to the experiments, Data Augmentation, dropout, and Transfer Learning techniques are applied to these models to improve accuracy and reduce overfitting. The proposed model is fine-tuned with the number of layers, and their weights achieve a greater classification accuracy of 99.33% for the trainining set and 98.87% for the validation set. The Experiments on various deep-learning models such as YOLO, Faster RCNN with specific modifications can be performed in the future. Also, to check whether YOLO, Faster RCNN models can be fit in a smaller dataset or not and comparing them with benchmark dataset in the future.

REFERENCES

[1] Nikhil Buduma, Nicholas Lacascio. Fundamentals of Deep Learning. 2017.

[2] Michael Beyeler. Machine Learning for OpenCV. July 2017.

[3] Mengying Shu. Deep Learning for image classification on very small datasets using transfer learning. 2019. https://lib.dr.iastate.edu/cgi/viewcontent.cgi?article=1374&context=creativecomponents.

[4] Haijian Ye, Hang Han, Linna Zhu and Qingling Duan. Vegetable Pest Image Recognition Method Based on Improved VGG Convolutional Neural Network. IOP Conf.Series: Journal of Physics.Volume 1237. Issue 3. December 2019.

[5] K.S. Anand, M. Poonkodi, C. G. Rohith Kumar Gupta, Raj Kshatriya Kshaktawat. Transfer Learning Based Multi Label Classification of Images. *International Journal for Research in Applied Science & Engineering Technology (IJRASET).* Volume 7 Issue III. Mar 2019.pp 2642-2646.

[6] Muthukrishnan Ramprasath, M. Vijay Anand, Shanmugasundaram Hariharan. Image Classification using Convolutional Neural Networks. *International Journal of Pure and Applied Mathematics*. Volume 119 No. 17 2018. pp 1307-1319.

[7] Sajja Tulasi Krishna, Hemantha Kumar Kalluri. Deep Learning and Transfer Learning Approaches for Image Classification. *International Journal of Recent Technology and Engineering (IJRTE).* Volume-7, Issue-5S4, February 2019. pp 427-432.

[8] Srikanth Tammina. Transfer Learning using VGG 16 with Deep Convolutional Neural Network for Classifying Images. *International Journal of Scientific and Research Publications*. Volume 9, Issue 10, October 2019. pp 143-150.

[9] Manali Shaha, Meenakshi Pawar. Transfer Learning for Image Classification. *Proceedings of the 2nd International conference on Electronics, Communication and Aerospace Technology (ICECA 2018).* March 2018. pp 656-660.

[10] Mahbub Hussain, Jordan J. Bird, and Diego R. Faria. A Study on CNN transfer learning for image classification. UK Workshop on Computational Intelligence systems - Nottingham, United Kingdom. Volume 840, Springer, September 2018. pp 191-202.