Detection of Driver Distraction Using Deep Learning

MSc Research Project
MSc. Data Analytics

Aishwarya Ratnakar Shetty
Student ID: x19237740

School of Computing
National College of Ireland

Supervisor: Dr. Catherine Mulwa

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Aishwarya Ratnakar Shetty
Student ID:	x19237740
Programme:	MSc. Data Analytics	**Year:** 2021-2022
Module:	MSc. Research Project
Supervisor:	Dr. Catherine Mulwa
Submission Due Date:	15th August 2022
Project Title:	Detection of Driver Distraction Using Deep Learning
Word Count:	7615 **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: Aishwarya Ratnakar Shetty

Date: 15th August 2022

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):	

Detection of Driver Distraction Using Deep Learning

Aishwarya Ratnakar Shetty
x19237740

Abstract

Driver distraction is one of the core aspects that now regularly causes fatal car crashes and may impair traffic safety. Even though there are many rules and regulations, there is no development. To further identify these distractions, which may have been caused by a driver engaging in other activities like texting, chatting on the phone, eating, etc., several research have been undertaken and technologies have been created. Using deep learning, which is normally used for image categorization, this research aims to develop a system that aids in assessing whether the driver is distracted or not. CNN employs transfer learning, which aids in lowering costs and increasing productivity. In this study, the pre-trained models ResNet50, VGG16, and VGG19 are employed. In this study, the idea of data augmentation is applied. According to the results, CNN accurately predicted the distraction with a 94.5% accuracy rate with second best model ResNet50 with 93.90%. Computational time taken for ResNet50 was less compared to CNN.

1. Introduction

Automobile is a major source of transportation in one's day-to-day life. And one of the main causes of road crashes is driver distraction. According to the World Health Organization (WHO), approximately 1.3 million people die each year as a result of road accidents, which are highly dangerous for both drivers and pedestrians. Distracted driving is one of the main causes of these accidents. Drivers who are distracted are those who are not paying attention to the roads while driving. They may be using a cell phone, listening to music, or becoming side-tracked by other road users or objects. Distracted driving can result in a crash. Driving when fatigued is likewise regarded as a distraction. Distractions can be manual (like being side-tracked by a text or call or engaging in other activities like drinking or eating), auditory (like being side-tracked by an unwanted voice), visual (like being side-tracked by nearby objects or people and not paying attention to the driving lane), and cognitive (like being side-tracked by thoughts or ideas) (it means driver not mentally being present or day dreaming). (Baheti, Gajre and Talbar, n.d.) Driver fatigue, which can result from a variety of factors, is also one of the causes of driver distraction, which accounts for 20% of all traffic accidents. It is possible to prevent these distractions, which might result to fewer crashes brought on by driver distractions. Driver, passengers, and other road users all face a serious risk of death as a result of driver distractions. Driver distraction can be identified utilizing embedded sensor devices in the vehicle. There have been numerous early studies that use behavioural methods to identify driver distraction. In this study, application of deep learning techniques is used to concentrate on finding drivers using various features Support vector machines and deep neural networks, which were the most often utilized techniques, have been used in several research studies. Although SVM is faster and more efficient when it comes to computing cost, because it is believed to offer the greatest accuracy when compared to other machine learning approaches, CNN is still the most often used methodology. Deep neural networks are also good at categorizing photos and objects. (Feng and Yue, n.d.)

Deep learning is a popular application which is the bottom area of machine learning which uses various non-linear process unit which is used for deep learning, feature extraction and conversion. It has many layers. Transfer learning models are commonly use where pre-trained models are transmitted to the model for training. Transfer learning which is used to reduce cost and to improve efficiency.

Taking into account the above said studies and research papers, it can be concluded that using different deep learning models predict driver distraction using various deep learning techniques in addition to transfer learning, which will assist in determining whether the driver is distracted or not.

This system can be used in future in manually driven vehicle as a safety measures just by adding a camera at driving position which will help to detect the driver distraction which will help to reduce car crash and promote safe driving.

1.1 Research Question

Many studies have been conducted on driver distraction detection with the goal of determining whether the driver is distracted by utilizing a variety of variables, including lane-keeping, physical behaviour, facial behaviour, and others. Finding drivers who are distracted will help save many lives, which will also assist to prevent accidents and other traffic-related problems. Deep learning is employed in the proposed study to determine whether or not the driver is distracted.

RQ: "How effectively can Deep Learning techniques (CNN, VGG16, VGG19, and ResNet50) using transfer learning to detect whether a driver is distracted or not?"

The following sections detail the research. The literature review in Section 2 contains a variety of studies and examines the methods and results, which will aid in identifying earlier approaches. In section 3, the recommended methodology will be described. The implementation approach for this study will follow the specifications in Section 4. The paper's impact and the methodology's shortcomings will be examined. The conclusion of the paper is in Section 5.

1.2 Research Motivation

One such example is well-known comedian and actor Tracy Morgan, who was seriously injured in a crash that was caused by drowsy driving. A Walmart-operated truck and the actor's car were both involved, and the truck's driver had reportedly been awake for 24 hours. Three additional people were also in the actor's vehicle, one of who was killed. Numerous cases show how harmful and sometimes fatal such distractions may be[1]. In other cases, reported, a young boy lost his life when the driver following him failed to pay attention and slammed into his car, which was then hit by an approaching truck, instantly killing him. The boy's family also organized a charity to oppose distracted driving through awareness; eventually, The National Distracted Driving Coalition was created when this group collaborated with 2 dozen activists.[2]

The purpose of this study is to categorize images of normal and distracted drivers engaging in side activities such texting, chatting on the phone, drinking, etc. and

[1] https://sleepeducation.org/tracy-morgan-crash-avoided-sufficient-sleep/

[2] https://eu.usatoday.com/story/opinion/voices/2021/09/17/distracted-driving-killed-my-son-gmpresident/8362401002/

determine whether or not the activity is distracting. Therefore, to develop and implement a model that would assist in detecting distracted driving using various deep neural models as well as transfer learning, which will aid in early detection of such distraction to address and prevent such incidents, as well as to develop models that will be more accurate and useful.

1.3 Research Objective and Contribution

The major goal of this research is to develop a system that will benefit in determining whether or not the driver is distracted. In the proposed study, deep learning models alongside transfer learning models such as CNN, ResNet50, VGG16, and VGG19 were used. Below are the objectives that the paper will stick to

Table 1: Objectives

Objective	**Description**	**Methods**
Obj.1	Reviewing of literature that are related to Detection of Driver distraction that uses various methods	
Obj.2	Methodology: Where the steps followed for the paper is discussed.	CRISP-DM
Obj.3	Implementation of the proposed models	CNN, ResNet50, VGG16 and VGG19
Obj.3.1	Preliminary data exploratory, Data Pre-processing and Data augmentation is Discussed.	
Obj.3.2	Implementation, Evaluation and results of Convolutional Neural Network	
Obj.3.3	Implementation, Evaluation and Results of Residual Network 50 Model (ResNet50	
Obj.3.4	Implementation, Evaluation and Results for VGG16	

Obj.3.5	Implementation, Evaluation and Results for VGG19	
Obj.4	Comparison of developed models, Existing models and Discussion	
Obj.4.1	Comparison of Developed Models and Discussion	
Obj.4.2	Comparison of Developed Models vs Existing Models	
Obj.5	Conclusion and Future Work	

Contribution

The research makes a contribution by employing 4 models on 10 class which has normal and distracted driver images which for determining whether a driver is distracted or not. The models developed can identify the distracted drivers with all the models performing well. The aim of the study was to develop and put into effect a method that would help identify the distractions that increase safety and decrease accidents.

2 Related Work

Driver distraction may be complicated, and numerous researchers have experimented with various techniques in an effort to develop better performing models that can be used to determine whether or not a driver is distracted. Driver drowsiness can be complicated and provide a number of difficulties. The analysis of numerous studies that attempted to resolve these problems by utilizing various features, methodologies, etc. is presented here in the connected work. The work in this study is provided beginning with the year (DATE). Depending on the chosen model, which is detailed below, the section is separated into three sections.

2.1 Detection of driver distraction using Deep learning

In order to identify driver distraction using a camera and apply deep convolutional neural networks, (Tran et al., 2018) conducted research with the goal of reducing accidents and improving transportation. In addition, they built a conversational (voice-alert) system that alerts the driver in real-time and demonstrated better performance than the baseline system, which had 256 neurons. Due to a shortage of hardware tools, they used a specially created assisted driving testbed simulator and their own data to conduct their research. For each action, they recorded a 5-minute video and chose one image. Data augmentation was also used to lessen overfitting. Here, ResNet had higher accuracy but was the slowest and took the longest, whereas VGG-16 had better frequency but less accuracy. They conclude the research with future work that hopes to solve the misclassification problems that occur due to similar posture in behaviours which gives misleading classification.

In this paper, they (Majdi, Ram, Gill and Rodr´ıguez, 2020) present Drive-Net, an automatic supervised learning technique for detecting driver detection. Drive-Net uses a combination of a convolutional neural network and a random forest to classify images of drivers. They compared the proposed Drive Net's efficacy to that of a recurrent neural network and a multi-layer perceptron, two other well-known machine learning techniques. They tested the techniques on a currently available database of photographs taken in a controlled setting, which contains approximately 22425 images that have been meticulously annotated by an analyst. The findings show that Drive-Net has a detection performance of 95%, which is 2% higher than the best results obtained using other techniques on the same database. They demonstrate that their Drive-Net beats the driver fatigue algorithms submitted in the competition when applied to a publicly accessible dataset of photos used in a Kaggle challenge. They compared Drive-Net against RNN and MLP, two alternative neural network approaches, using a comparable dataset. The results show that Drive-Net outperforms the other two algorithms in terms of detecting performance.

(Kumar, Sangwan and Sangwan, 2021) conducted study on currently existing techniques for driver posture detection in order to enhance the driver detection system employing genetic approach with six advanced deep neural networks using two datasets where they both got higher results than the prior system. They used an ensemble to modify the hierarchical variation of DenseNet-201. Additionally, they have an exploratory analysis. Their experiment included two datasets, namely AUC and State Farm, and it performed well in both datasets when compared to the existing state-of-the-art. They conclude that GA ensemble may be deployed on devices and performance can be monitored in real-time as future work.

The research proposed studies the driver distraction using the posture of driver is done. the research tried to build a system which had higher accuracy (Abouelnaga, Eraqi and Moustaf, 2018) the proposed research used an ensemble CNN which is a genetically weighted method along with genetic algorithm. The research also studied the effects of visual distractions. The system build works as visual based that detects driver's posture. The research used face and hand detectors on which the models where applied. Genetically weighted ensemble CNN resulted in better accuracy of 95%. The research also concluded that using one simple model like AlexNet can also work better maintaining the accuracy and can be used in real time. As future work, a better device can be used to detect the face and hand movements where manual labelling might be needed for the proposal to train and use them to improve the localization of hand and face.

2.2 **Detection of driver distraction through Machine learning**

In the proposed research a method for detection of driver errors and also to warn the driver of modern vehicles using CNN feature extraction-based classification model along with machine learning techniques. Initially, SqueezeNet CNN was used with transfer learning to extract features of images before the classification using machine learning was applied. The main goal of the research was to detect driver's errors using the images. The classification model used in the study were k-NN, SVM and RF which was applied to 1000 features provided in an image. Cross validation was used to measure the performances of each model. Confusion metrics was also performed for each model which was used to calculate the performance of the model. Among the models used, k-NN yielded better results than others along with faster classification time. Lastly, the mention of model success was described which will be also used for real time driver errors where the success of the model can be increased with the increase in number of images. (AL-DOORI, TASPINAR and KOKLU, 2021)

(Alkinani, Khan, Arshad and Raza, 2022) developed a hybrid model scheme called HSDDD for detecting driver distraction combining aggregate handmade and Deep CNN features, which works on three-tier co-ordination, concatenation, and classification, and to compare their research's current approaches. At the coordination tier, they utilized a four deep CNN approach to extract DCNN, which were then merged with HOG at the concatenation tier after they had first obtained HOG using handmade methods. They also employed PCA for feature selection, which improves results by removing undesirable data in addition to extracting features. Additionally, they seek to strengthen deep learning models, get beyond its flaws, and evaluate it against other approaches. They also aim to overcome the limitations of deep learning models and make it more robust and to compare with existing methods. They conducted an experiment in which they chose 100, 200, and 500 characteristics that gave them a prediction model with SVM and KNN that had equivalent accuracy. Despite getting strong results, there were some accuracy limits

Researchers (Aksjonov et al., 2019) performed research with the goal of detecting distracted drivers utilizing hardware and software environments where the regular driving model (to forecast drivers' lane holding and speed limit on the road) is taken into consideration. They conducted a driver in loop experiment with 18 individuals engaging in secondary activity, which allowed them to identify the activity. They used machine learning techniques to assess distraction and judge whether it was safe or not. These systems used nonlinear regression and fuzzy logic. The suggested research has an advantage over other forms of driver distraction that rely on behavioural and psychological qualities. It also has the benefit of not requiring additional tools like a camera or neuro-scan. This study has certain restrictions, such as the absence of statistical analysis, which they hope to add in the future and test using many additional factors that contribute to distracted driving.

2.3 Detection of driver distraction using transfer learning

It is important to understand drivers' behaviour. The research done by (Xing et al., 2018) aims to build an end-to-end system using CNN that identifies driver related issues where 7 activities where taken which is divided as normal and distracted driving. The research used Kinect which is a consumer camera that captures images and which are segmented using Gaussian mixture model (GMM) as images are raw. 2 methods were performed one using GMM and with raw images. The research used AlexNet along with transfer learning which gave an accuracy of 79% which is evaluated using leave-one-out (LOO) cross validation. The accuracy achieved without segmentation was 50.3%. The right mirror checking gets the best accuracy. The study also proposed a comparison between transfer learning and feature extraction which according to the researchers doesn't identify the behaviour of drivers accurately using CNN or HOG. The research also studied CNN model as binary classifier which had an accuracy of 94%. The proposed research had limitations where it is cannot detect some behaviours accurately. The future work also suggests on using other CNN models.

The research proposed by (Ugli et al., 2022) developed a system for driver distraction using transfer learning and fine-tuning techniques using Mobile Net, VGG16 and ResNet50. Using ImageNet, the dataset was pre-trained. 3 various fine-tuning techniques where applied which freezes some layer to compare and increase accuracy. The research claims better prediction with 495 right predictions out of 500 images using the test set. MobileNet that uses 80 frozen layers achieves the highest accuracy on both train and validation sets which researchers claim that this model can be used in real-time detection.

In the proposed research, the aim is to build a system which is robust that helps in detection of driver distractions. Using VGG16 which is modified according to the piece of work and CNN, the system is built which also implies regularization methods which is ReLU to yield better accuracy. The experimental result yielded accuracy of 82.5% with 240 images

/sec on GPU. Adam optimizer was used and with 4096 channels of fully connected layers was replaced by 512 convolutional layers. The modified model used in the research has image resized to 224x224 with RGB planes that subtracts from each pixel from the images. The evaluation is done using the heatmap. The future work suggests to lower the number of values and the computational time. It also suggests to incorporate a temporary context that will help reduce the errors that will increase the accuracy. Using genetic algorithm is suggested to improve the accuracy. (Oberoi, 2020)

A study was conducted by (Wang, Wu, Li and Zhang, 2020) for the purpose of detecting driver distraction on the operation area using data augmentation method. They combined the DOA dataset and original AUS to create a large dataset, and they used class activation method to extract key factors of behaviour analysis and R-CNN to detect the features and areas. Later, convolutional neural network classification was implemented to detect the actual dataset and operation area dataset, and this is where the research comes into its own. The use of CNN was made for feature extraction. To improve accuracy, data augmentation and transfer learning were used. they claimed that real-time detection can be performed using YOLOV3 and SSD. With respect to accuracy, R-CNN outperforms grad-CAM. They claim that the DOA dataset has a high level of distraction detection. But the model's accuracy improved when both were integrated. But when combined both, the model yields better accuracy. Additionally, they employed a wide-angle dataset to test their methodology. Their study is ended with a future perspective in which the distraction may be seen from several angles at night. Additionally, they state in their conclusion that more features should be taken into account when employing the YOLO detector to identify driver attention.

Using the driver' s behaviours involved in different activities, the research builds a system using CNN where 7 common activities are involved which are divided into 2 parts normal and distracted class. Image segmentation was done using Gaussian Mixture Model (GMM) which extract the driver images. Transfer learning was applied to pre-trained CNN. AlexNet, GoogleNet and ResNet50 was implemented that deals with binary and multiple classification. The experimental research consisted of 4 parts which were effect using image segmentation, CNN model, binary classification and comparison of performance with other models. It was observed that segmented images yield better result that the other models. Transfer learning and other extractions methods were compared too. CNN with binary classification gave better results of 91%. Using automated vehicles can be used for the driver distraction as a future work. (Xing et al., 2019)

2.4 **Detection of driver distraction using IoT**

This study (Khunpisuth, Chotchinasri, Koschakosai and Hnoohom, 2016) attempted to solve the problem by designing an experiment to assess sleepiness levels. The use of a Raspberry Pi Webcam as well as a Raspberry Pi 3 component capable of calculating the amount of sleepiness in operators was required for this study. The frequency with which a driver tilted his head or blinked his eyes was used to determine more than just whether he was tired. In a test with ten subjects, the accuracy of face and eye identification reached 99.59 percent. The focus of this article was on sleepy drivers and their proclivity to cause vehicle accidents. As a result, the authors created an embedded system that could notify drivers when they were sufficiently sleepy using computer vision and C++ programming code. The embedded system can estimate the driver's sleepiness level using only a Raspberry Pi 3 B Model and a Raspberry Pi Webcam. The study was able to detect a person using 99.85 percent of the entire frame.

After reviewing the research studies, it was observed that the studies were done using different features like driver behaviour, manual distraction, facial distraction, lane keep distractions, etc. Many different techniques and applications were used for detection of whether or not the driver is distracted which uses deep neural, machine learning, etc. this research tries to investigate deep learning techniques to detect the whether or not the driver is distracted or not.
This section achieves the objective that is mentioned in table 1 chapter 1.

3 Research Methodology

The research is conducted using the CRISP-DM approach. Cross Industry Standard Process for Data Mining (CRISP-DM), which contains a number of process, is the name of the method used for project deployment.

3.1 Driver Distraction Detection Methodology

The CRISP-DM approach is being used for this project. For data mining projects that provide a framework, CRISP-DM is frequently employed. Technology and industry both are independent to this methodology. Project/business understanding is a step in the CRISP-DM process that aids in gaining a better understanding of the projects. Following are the processes that are included: a) Understanding the business/project; b) Understanding the data; c) Preparing the data; d) Modeling; e) Evaluating; and f) Deploying. (Wirth and Hipp, n.d.)

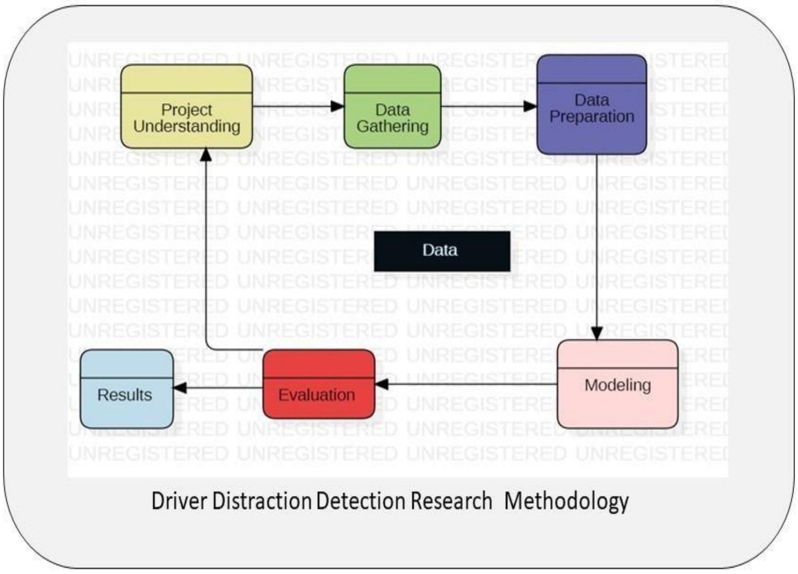


Fig 1 Driver distraction detection research methodology

Project Understanding

In this step, the emphasis is on obtaining data in order to fully understand the project's requirements and goals, which will be utilized to direct the study. This project's primary objective is to identify whether or not the driver is distracted which will assist to avoid accidents. A problem statement and objectives are created through the gathering of information, and a structure is needed to carry them out.

Data Understanding

Once the requirements are clear, the next crucial phase is data understanding, during which data is gathered and gain insights into whether any values are missing, whether any modifications or deletions are required, etc. State Farm Distracted Driver Detection data from Kaggle, an open-source platform, was used for the study.

Data Preparation

The next stage is to prepare the data for model implementation after it has been acquired. Data pre-processing takes place here. The pre-processed data is divided into train and test groups. In this case, the data is divided 80:20 ratio. Then, data augmentation was carried out.

Modelling

The models are put to use after the data preparation is finished. The models utilized have been explored in detail below. Following data pre-processing, which also includes data augmentation, and after the data has been divided into train and test sets using validation sets, these models are utilized. The models' performance is then evaluated using test data. The details of the models that have been used are shown below.

Convolutional Neural Network (CNN)

Convolutional neural networks, or CNN, are a type of ANNs which is Artificial Neural Network, which are most frequently employed for image detection. CNNs are made up of self-improving neurons that have an input layer where processing will occur to produce the final output. CNN separates pictures based on a single scoring mechanism called weight. The height, breadth, and depth of space are the three spatial dimensions that make up the 3D organization of neurons that make up the CNN architecture. Convolutional, pooling, and fully connected layers are the three layers they have. (O'Shea and Nash, 2015)

Residual Network-50 (ResNet50)

Residual Network 50 is what RestNet50 is. According to what its name implies, ResNet50 is a convolutional network that uses 48 convolutional networks across 50 layers of a neural network, one maxPool, and one average pool. Deep neural networks called ResNet-50 are used to resolve the vanishing gradient problem. Additionally, it makes an effort to address the degradation problems that convolutional networks commonly experience. Despite having 50 layers, it only contains 23 million more trainable values, which is regarded to be a tiny number compared to the architectures now in use. (Mandal, Okeukwu and Thei, 2021)

Visual Geometry Group (VGG16)

VGG16, or the Visual Geometry Group, is a convolutional neural network with 16 total layers, of which 13 are convolutional layers, 2 are fully connected layers, and 1 is a SoftMax function. The VGG16 architecture was first introduced by Karen and Andrew in 2014. Containing merely 3x3 convolutional layers layered on top of one another, they created a 16layer network with convolutional and fully connected layers. They have five layers: the first two are composed of two convolutional layers and one MaxPool layer, while the latter three are composed of three convolutional layers and one MaxPool layer. The model is trained on the training set using ImageNet. (Tammina, 2019)

Visual Geometry Group (VGG19)

VGG19, also known as the Visual Geometric Group, is a convolutional neural network of 19 layers, of which 16 are convolutional layers and 3 are fully connected layers that aid in classifying the images. To train on a dataset, VGG19 also leverages ImageNet. The usage of 3x3 filters in each layer makes VGG19 popular.

Evaluation

The performance of developed models was evaluated using confusion matrix. Below are some terms described in detail that are used in evaluation metrics. (Çınar, Yıldırım and Eroğlu, 2021)

Accuracy

It is used to evaluated the model performances of all the classes especially helpful when the classes are of equal importance. Accuracy gives right prediction to the total of prediction.

$$\boxed{\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}}$$

Precision

Ration of correct samples of positive to classify to the total number of samples used (it may be correct or it may be incorrect)

$$\boxed{\text{Precision} = \frac{TP}{TP+FP}}$$

Recall

It is used to detect the model’s ability to detect correct values. Higher is the recall value, more is the correct value detected.

$$\boxed{\text{Recall} = \frac{TP}{TP+FN}}$$

F1 score

It is nothing but the average of precision and recall. It is more useful when there is uneven classes.

$$\text{F1score} = \frac{2 * (\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}}$$

Sensitivity

Sensitivity is used to detect right class correctly. It is calculated as follows

$$\text{Sensitivity} = \frac{TP}{TP+FN}$$

Specificity

It is nothing but it gives the percentage of negative prediction correctly

$$\text{Specificity} = \frac{TN}{TN+FP}$$

Results

Result is the last stage of the research methodology where the interpretation of the models results are gained. The results are presented using visualizations and graphs.

3.2 Design Specification

Below diagram represents the research flow that is used for carrying out this research and it implementations

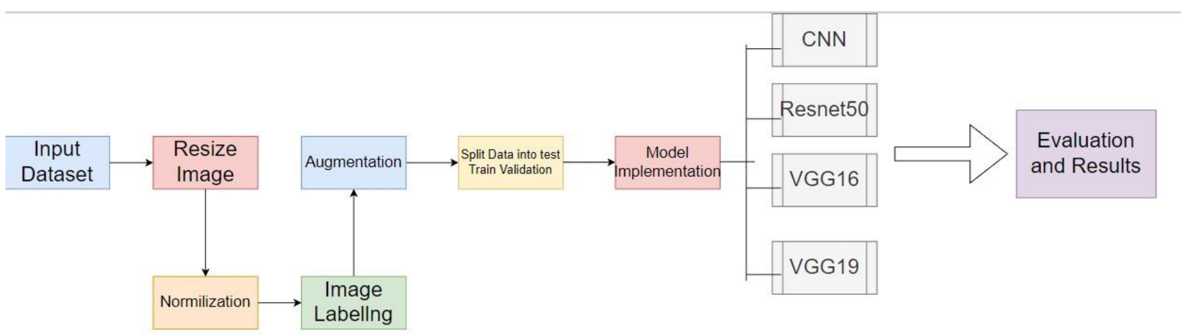


Figure 2: Design Specification

4. Implementation, Evaluation and Results for Detection of Driver Distraction

4.1 Introduction

Here in this chapter, the proposed methodology is explained. It is an extension of the previous chapter's explanation of the model's implementation, outcomes, and evaluation, which satisfies the objectives mentioned in Table 1

4.2 Data Gathering

The State Farm Distraction Driver Detection dataset, which is gathered from the open-source platform Kaggle and is used to identify distracted drivers. Since the dataset couldn't be loaded by the system, which was causing it to crash, only 7000 data were selected, with 700 images taken from each class and the rest manually deleted. 22,424 images made up the original dataset. The dataset as a whole therefore consists of 7000 images. The. Images from the 2D dashboard were photographed as the photographs were being taken. In order to be utilized for the next implementation, all necessary packages and libraries were installed.

```
import os
len(os.listdir('/content/drive/MyDrive/Distracted_Driver_Detection/Data/imgs/train/c5'))

    700
```

Figure 3: Count if total images.

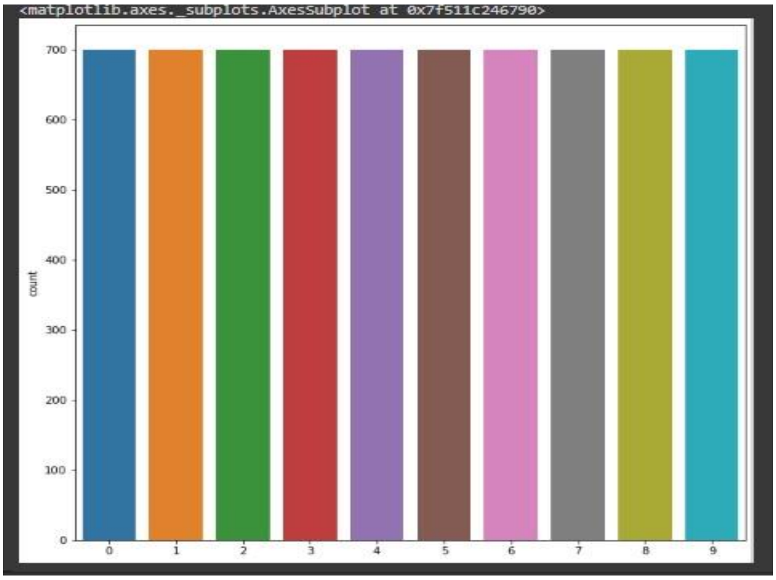


Figure 4: Bar plot showing total count in each class

4.3 Preliminary Data Exploratory, Pre-processing and Data

Augmentation

Once the data is loaded and after all the packages have been installed and imported, preliminary data exploratory analysis is done. There are 10 image class. Where the images are classified where the images are read in Gray and RGB (Red, Green, Blue) colour. Where 1 is grayscale image and 3 is RGB and default is kept as 3 and returned the images in this format. Then to read the images in each class folder, first an empty list is declared where the images in array and label class will be stored. Looping in train folder will go through all the class and will store the label class and then image is read and then it is image and class are appended. In data pre-processing step, the images in the dataset where of various size, so image where resize to 64x64 size and along normalization. train and test where given shape. The images in the dataset were labelled as c0, c1, c2 and so on. So, the images were mapped accordingly where c0 class was assigned as normal images and rest 9 classes were labelled according to the image (figure 4 & 5). The dataset, which was previously described, consisted of 10 classes, nine of which contained pictures of distracted drivers. Normal and distracted driving were segregated into separate classes. All the 10 classes were used for prediction. The train images underwent data enhancement. In order to acquire a fresh perspective and improve the robustness of a model, data augmentation is employed to obtain numerous copies of the same photos (figure 6). For image augmentation, the ImageDataGenerator from the Keras package is utilized because it is quick and simple. Resize, zoom range, width and height shift, and shear range are all used in this instance. The batch sized used for implementation is 32. After data augmentation, it augmented data is split into train and validation set.

```
batch_size = 32
train_datagen = ImageDataGenerator( rescale=1 / 255.0,
                                    zoom_range=0.05,
                                    width_shift_range=0.05,
                                    height_shift_range=0.05,
                                    shear_range=0.05,
                                    fill_mode="nearest")
```

Figure 5: Data Augmentation

```
activity_map = {'c0': 'Safe driving',
                'c1': 'Texting - right',
                'c2': 'Talking on the phone - right',
                'c3': 'Texting - left',
                'c4': 'Talking on the phone - left',
                'c5': 'Operating the radio',
                'c6': 'Drinking',
                'c7': 'Reaching behind',
                'c8': 'Hair and makeup',
                'c9': 'Talking to passenger'}
```

Figure 6: Image Name Labelling

Figure 7: Images after Image labelling

4.4 Implementation, Evaluation and results of Convolutional Neural Network

One of the most popular and well-known deep learning network applications, CNN, has numerous layers, including convolutional, non-linear, pooling, and linked layers. One of the many areas where CNN has excelled is image processing. When analysing an input image, CNN assigns weight and biases to various elements so they can stand out from one another.

First, the fundamental Convolutional Neural Network was put into practice. In this study, a classification model with numerous layers was created using the CNN approach. A 2D convolutional layer as large as 32 and 64 was engaged. With a kernel size of 2, padding of "same," and the activation function ReLu, there are 5 convolutional layers applied. Also carried out was batch normalization which is used to improve the efficiency of the network. The layer depth affects filter size, and the variation is seen through the next layers. The application MaxPooling2D was used, and its 2x2 pooling size improves in maintaining the characteristics of the images as well as preventing overfitting in each convolutional layer**.** Rectified linear units, or ReLUs, are used. It has a method called activation that helps turn any potential negative pixel values into zero. The result of the dropout layer is given to a fully connected dense layer with 512 values with ReLu utilized as the activation function. The flattening layer turns the generated 2D arrays from the pooling features into a single dimensional linear vector. SoftMax function activation is used to classify the output images and determine whether or not the driver is distracted. It captures the value from the layers and gives the possible value for the process.

After using data augmentation, the model was applied to test and training sets of data. A dataset with 10 training epochs was used to train the model. The computational time of

239s in 7^{th} epoch with 94.5% accuracy after 7^{th} epoch, the val-acc did not improve and with test accuracy 89.70% with total of around 35.96 minutes of computational time to run all the epochs. This accuracy was attained by choosing the images and applying the model to prediction. CNN was the best performing model among all the models with good accuracy and good computational time as well.

Adam optimizer was used and categorical cross entropy loss and accuracy as metrics was used. Adam optimizer is used for gradient which works efficiently when working on data with many parameters. Categorical cross entropy is used for multi class classification.

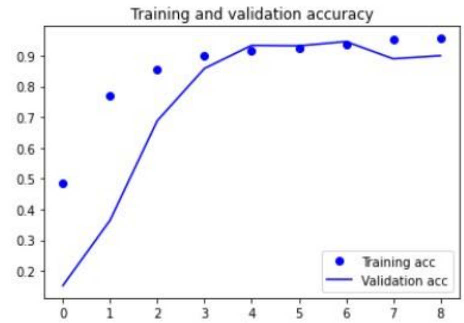


Figure 8: training and validation accuracy CNN

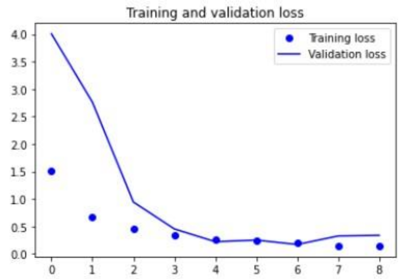


Figure 9: training and validation loss CNN

As seen the above graph, the gap between the accuracy and loss going near stating that the model is good.

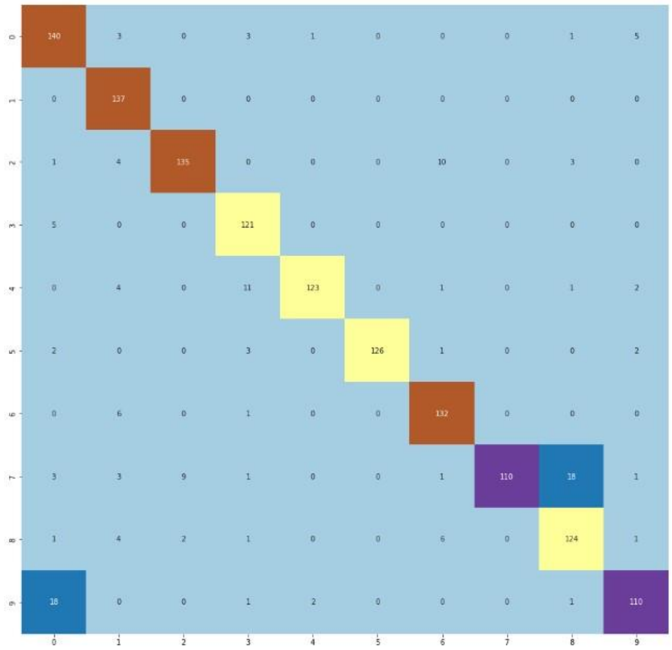
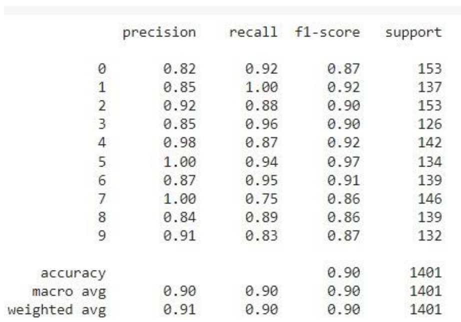


Figure 10: Heatmap CNN



	precision	recall	f1-score	support
0	0.82	0.92	0.87	153
1	0.85	1.00	0.92	137
2	0.92	0.88	0.90	153
3	0.85	0.96	0.90	126
4	0.98	0.87	0.92	142
5	1.00	0.94	0.97	134
6	0.87	0.95	0.91	139
7	1.00	0.75	0.86	146
8	0.84	0.89	0.86	139
9	0.91	0.83	0.87	132
accuracy			0.90	1401
macro avg	0.90	0.90	0.90	1401
weighted avg	0.91	0.90	0.90	1401

Figure11: Confusion Matrix CNN

The above figure tells the accuracy of each class, such like 140 were correct prediction and 13 were incorrect prediction for class0 which is safe driving and so is for the rest of the class. The diagonal line represents the true value means correct prediction.

4.5 Implementation, Evaluation and Results of Residual Network 50 Model (ResNet50)

ResNet50 is a CNN that is 50 layers deep. ResNet stands for Residual Network. Here weights ImageNet was used with and addition of 3 more layers in the model. Like CNN here too there are 5 convolutional layers that is applied with kernel size 2, padding "same" and an activation function was used which is ReLu. BatchNormalization was also performed. Filter size changes according to the layer depth and the difference can be seen through the successive layers. MaxPooling2D was applied which had 2x2 pooling size which helps to retain the features of the images and also by preventing overfitting in each convolutional layer. ReLU was used as an activation function. The outcome from the dropout layer uses Flattening layer which converts the resultant 2D arrays from the pooling features into single dimensional linear vector and which is passed to fully connected dense layer that has 512 values with ReLu used as activation function. The final layer matches to the 10 classes that are present in the dataset and by using SoftMax function activation to classify the output images to tell whether the driver is distracted or not. Categorical cross entropy and the Adam optimizer were applied. The model was applied to train and test data that is used after applying data augmentation. The model used trained dataset and the number of epochs used were 10 where the maximum validation accuracy obtained was 93.30% and with the test data the accuracy achieved was 93.86%. The plot shows close correctness which states that the model has good performances. With validation didn't get better with the next epoch. The evaluation is done using the confusion matrix and by accuracy table. The time taken to complete 10 epoch was around 34.91 minutes.

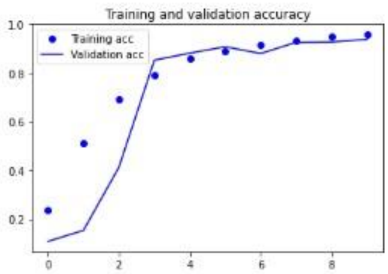


figure 12: Training and validation accuracy ResNet50

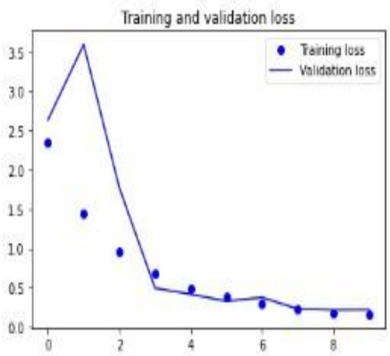


figure 13: Training and validation accuracy ResNet50

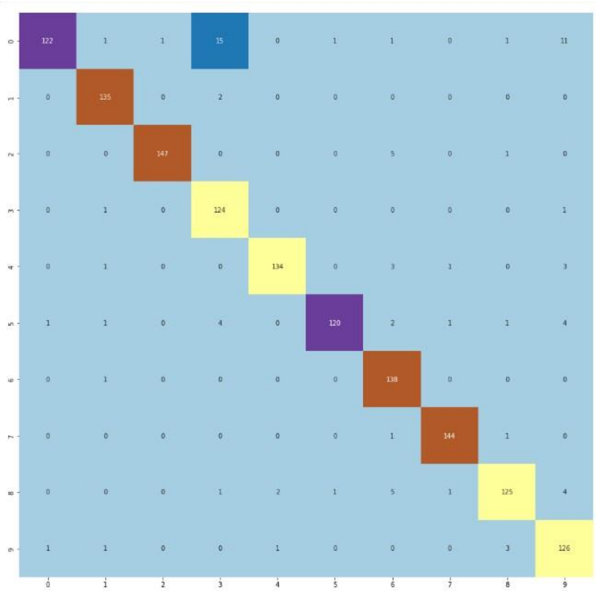


Figure14: Heatmap

```
print(classification_report(ytest_labels,classes_x))
```

	precision	recall	f1-score	support
0	0.98	0.80	0.88	153
1	0.96	0.99	0.97	137
2	0.99	0.96	0.98	153
3	0.85	0.98	0.91	126
4	0.98	0.94	0.96	142
5	0.98	0.90	0.94	134
6	0.89	0.99	0.94	139
7	0.98	0.99	0.98	146
8	0.95	0.90	0.92	139
9	0.85	0.95	0.90	132
accuracy			0.94	1401
macro avg	0.94	0.94	0.94	1401
weighted avg	0.94	0.94	0.94	1401

Figure 15: Confusion Matrix

ResNet50 gave the highest accuracy among the rest models. Though the computational time it took was little longer CNN, it performed better. It was able to predict correctly where for c0 it was able to predict 122 correct with 41 incorrect prediction and so on for the rest classes where it was able to predict correctly.

4.6 Implementation, Evaluation and Results for VGG16

VGG16 is one of the most known transfer learning models used in deep learning. The 16 in VGG16 refer to as 16 layers where 13 are convolutional layer, 2 are fully connected layers and 1 is the SoftMax. Weighting is done using a pre-trained model that draws on ImageNet, and transfer learning is employed. The VGG16 has 3x3 filter and has 5 convolutional layer which is divided by MaxPooling in each layer with ReLu used as an activation function. GlobalAveragePooling was applied at last layer. Pre-trained model is used. Apart from filters, rest parameters have kept none. The 'layer.trainable' which is used in VGG16 is kept false. Rmsprop optimizer is used here and for loss function categorical_croosentropy is used. The size of the batch is kept 32 and epochs 10 This model too was applied on augmented data. The maximum validation accuracy obtained was 73.63% with early stopping at 6^{th} epoch as the accuracy didn't not improve with the next epoch and with the test data the accuracy achieved was 73.66%. The plot shows close correctness which states that the model has good performances. The evaluation is done using the confusion matrix The time taken to complete the training for 6 epoch was around 31.3 minutes.

Figure 16: train and validation loss VGG16

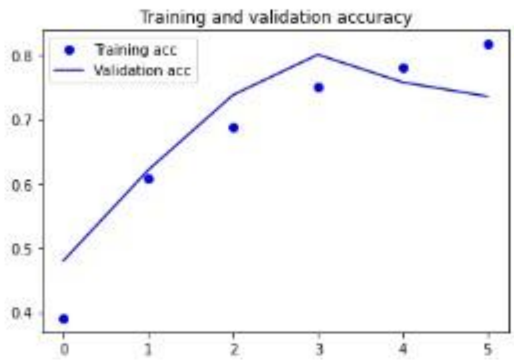


Figure 17: train and validation accuracy VGG16

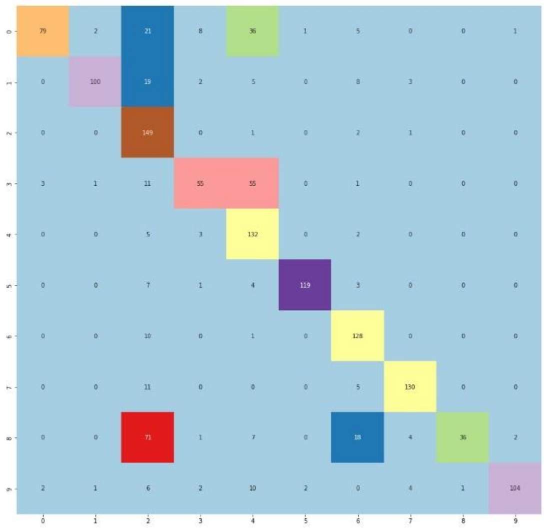


Figure 18: Heatmap VGG16

```
print(classification_report(ytest_labels,classes_x))

              precision    recall  f1-score   support

           0       0.94      0.52      0.67       153
           1       0.96      0.73      0.83       137
           2       0.48      0.97      0.64       153
           3       0.76      0.44      0.56       126
           4       0.53      0.93      0.67       142
           5       0.98      0.89      0.93       134
           6       0.74      0.92      0.82       139
           7       0.92      0.89      0.90       146
           8       0.97      0.26      0.41       139
           9       0.97      0.79      0.87       132

    accuracy                           0.74      1401
   macro avg       0.83      0.73      0.73      1401
weighted avg       0.82      0.74      0.73      1401
```

Figure19: Confusion Matrix CGG16

VGG16 did not perform that well but the accuracy achieved is acceptable.
Considering c0 looking at the confusion matrix, it was able to predict 79 correctly and 74 incorrectly.

4.7 Implementation, Evaluation and Results for VGG19

VGG19, also known as the Visual Geometric Group, is a convolutional neural network of 19 layers, of which 16 are convolutional layers and 3 are fully connected layers that aid in classifying the images.
Here pre-trained model is used. It has 3 layers. Apart from filters, rest parameters have kept none. the output result is passed through each layer to get the resultant outcome using SoftMax activation. Here in the model, the layer.trainable which is used in VGG19 is kept false. Rmsprop optimizer is used here and for loss function categorical_croosentropy is used. This model too was applied on augmented data. The maximum validation accuracy obtained was 72.46% and with the test data the accuracy achieved was 72.73%. With validation didn't get better with the next epoch, early stopping was at epoch 5. The evaluation is done using the confusion matrix and by accuracy table. The time taken to complete the 5 epochs was around 32.7 minutes.

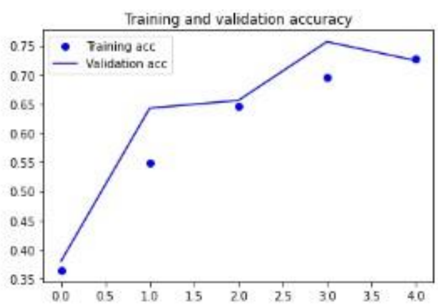


Figure 20: Train and validation accuracy VGG19

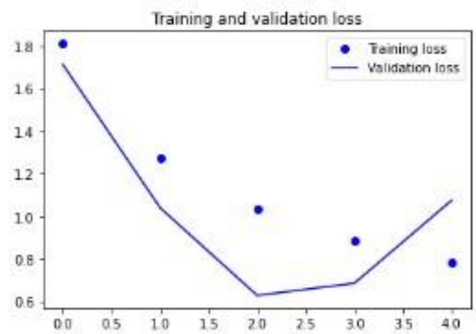


Figure 21: Train and validation accuracy VGG19

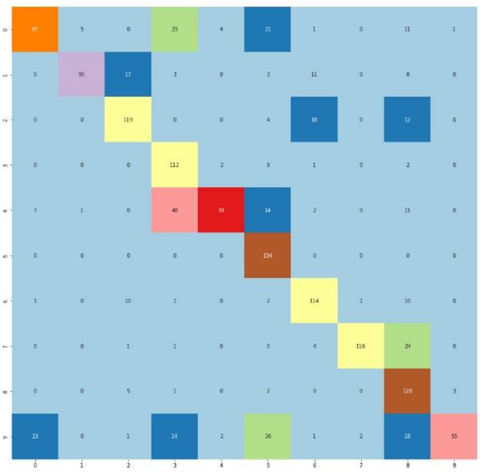


Figure 22: Heatmap VGG19

```
print(classification_report(ytest_labels,classes_x))
```

	precision	recall	f1-score	support
0	0.81	0.57	0.67	153
1	0.94	0.69	0.80	137
2	0.78	0.78	0.78	153
3	0.55	0.89	0.68	126
4	0.88	0.42	0.56	142
5	0.62	1.00	0.77	134
6	0.75	0.82	0.78	139
7	0.97	0.79	0.88	146
8	0.57	0.92	0.71	139
9	0.93	0.42	0.58	132
accuracy			0.73	1401
macro avg	0.78	0.73	0.72	1401
weighted avg	0.78	0.73	0.72	1401

Figure 23: Confusion Matrix VGG19

VGG19 model was the poor performing model among the four. Though the result obtained cannot be termed bad. The computational time too was long with around 32.7 minutes to run 5 epochs. It was able to detect 87 correct and 66 incorrect for c0.

5. Comparison of developed models, Existing models and Discussion

In this part of the section, overall research analysis will be discussed. The first part contains a discussion where comparison on developed models used for carrying out research will be discussed, its outcomes, limitations, benefits and so on. In the second part comparison of already existing research models versus the developed models will be discussed

5.1 Comparison of developed models and Discussion

In this section, the overall summarization of the results of the developed models. Below table. No discuss the same. Many research done have combined the classes of distracted driving into one class and segregated them into 2 groups namely safe driving and distracted driving. In this paper a different approach was used and the classes where not combined. All the 10 class went through classification model which were applied to the pre-trained dataset. Here, as the dataset was reduced by manually deleting images from each class and making the final dataset of 7000 images. The images went through various steps which was formed in the initial stage. Deep leaning and transfer learning was applied. After reviewing the research work done previously, the model's selection was done. transfer learning was applied. The models selected for this research yielded better accuracy and was able to do the prediction od driver distraction correctly. The performance every model can be considered good. The model that achieved the high accuracy was CNN with a accuracy of 94.55% the result for the test set was also good at 89.79%. the second-best model was ResNet50 with 93.90% and 93.86% respectively for validation and test but the computational time taken by ResNet50 was comparatively less than CNN.

Limitation of this research is that it used only some part of the images from the dataset. It used 7000 images out of the 22,424 where images were randomly selected from each class and rest being deleted.

Table 2: Model Comparison

MODELS	AUGMENTED IMAGE
CNN	Train Accuracy: 94.55% Test Accuracy: 89.79%
ResNet50	Train Accuracy: 93.90% Test Accuracy: 93.86%
VGG16	Train Accuracy: 73.63 % Test Accuracy: 73.66%
VGG19	Train Accuracy: 72.46% Test Accuracy: 72.27%

5.2 comparison of Developed models vs existing models

The section describes the results of developed model which is compared with already existing models(state-of-art) and is presented in the table below.

Table 3 Developed model vs Existing Model

Author Name	Year	Models Used	Accuracy
(Xing et al., 2019)	2019	CNN. AlexNet, GoogleNet and ResNet50	CNN-91%
(Oberoi, 2020)	2020	VGG16	82.5%
(AL-DOORI, TASPINAR and KOKLU, 2021)	2021	CNN for feature extraction, k-NN, SVM and RF	k-NN-98.1%
Ugli et al., 2022)	2022	Mobile Net, VGG16 and ResNet50	MobileNet-99%

Aishwarya Shetty	2022	CNN, ResNet50, VGG16, VGG19	CNN-94.55% & ResNet50- 93.90%

6. Conclusion and Future Work

Here in this research deep learning techniques were used to detect whether or not the driver is distracted. After reviewing various research, the model selection was done. The models used CNN, ResNet50, VGG16, VGG19 which used transfer learning approach. The main goal of the research is to develop a system that helps to detect whether or not the driver is distracted or not. The detection was done using the body language of the driver which tells whether or not the driver is distracted or not. The dataset used gathered from Kaggle that had 10 different groups of safe driving and distracted driving Concept of data augmentation using rescaling, width and height range, zoom range, shear range was applied. The augmented data is divided into train and test sets in order to assess the model's performance. The developed models were applied which yielded better accuracy for all the models. The model that performed best was CNN which had the highest accuracy of 94.5%. ResNet50 was the second-best model with 93.90% accuracy but the computational time it took was less compared to CNN for 10 epochs. VGG19 gave the least accuracy with 72.46% which also had the highest computational time of 32.7 minutes for 5 epochs.

The research had few limitations where the dataset used is less due to system configuration which didn’t allow the data to load so out of 22,424 images,7000 images used. As future work, the research can be conducted on larger dataset with better performing system. Also, different dataset can be used to perform the analysis with exploring other models of CNN.

Acknowledgement

The project would not have been accomplished without the help of my supervisor, Dr. Catherine Mulwa, who supervised me throughout the study. I want to express my appreciation to my family and my friends for giving me the motivation and encouragement I needed to achieve and accomplish my goals.

Reference:

Abouelnaga, Y., Eraqi, H. and Moustaf, M., 2018. *Real-time Distracted Driver Posture Classification*. [online] Arxiv.org. Available at: <https://arxiv.org/pdf/1706.09498.pdf>.

AL-DOORI, S., TASPINAR, Y. and KOKLU, M., 2021. Distracted Driving Detection with Machine Learning Methods by CNN Based Feature Extraction. *International Journal of Applied Mathematics Electronics and Computers*, 9(4), pp.116-121.

Alkinani, M., Khan, W., Arshad, Q. and Raza, M., 2022. HSDDD: A Hybrid Scheme for the Detection of Distracted Driving through Fusion of Deep Learning and Handcrafted Features. *Sensors*, 22(5), p.1864.

Baheti, B., Gajre, S. and Talbar, S., n.d. [online] Openaccess.thecvf.com. Available at: <https://openaccess.thecvf.com/content_cvpr_2018_workshops/papers/w14/Baheti_Detection_of_Distracted_CVPR_2018_paper.pdf>

Çınar, A., Yıldırım, M. and Eroğlu, Y., 2021. Classification of Pneumonia Cell Images Using Improved ResNet50 Model. *Traitement du Signal*, 38(1), pp.165-173.

Feng, D. and Yue, Y., n.d. [online] Cs229.stanford.edu. Available at: <http://cs229.stanford.edu/proj2019spr/report/24.pdf>

Khunpisuth, O., Chotchinasri, T., Koschakosai, V. and Hnoohom, N., 2016. Driver Drowsiness Detection Using Eye-Closeness Detection. *2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*,.

Kumar, A., Sangwan, K. and Sangwan, D., 2021. *A Computer Vision Based Approach for DriverDistraction Recognition using Deep Learningand Genetic Algorithm Based Ensemble*. [online] Available at: <https://www.researchgate.net/publication/353544363_A_Computer_VisionBased_Approach_for_Driver_Distraction_Recognition_using_Deep_Learning_and_Genetic_Algorithm_Based_Ensemble>.

Majdi, M., Ram, S., Gill, J. and Rodr´ıguez, J., 2020. [online] Arxiv.org. Available at: <https://arxiv.org/pdf/2006.12586.pdf>.

Mandal, B., Okeukwu, A. and Thei, Y., 2021. *Masked Face Recognition using ResNet-50*. [online] Arxiv.org. Available at: <https://arxiv.org/pdf/2104.08997.pdf>

Oberoi, M., 2020. Driver Distraction Detection using Transfer Learning. *International Journal of Engineering Research and*, V9(05).

O'Shea, K. and Nash, R., 2015. [online] Arxiv.org. Available at: <https://arxiv.org/pdf/1511.08458.pdf>

Tammina, S., 2019. Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images. *International Journal of Scientific and Research Publications (IJSRP)*, 9(10), p.p9420.

Tran, D., Manh Do, H., Sheng, W., Bai, H. and Chowdhary, G., 2018. Real-time detection of distracted driving based on deep learning. *IET Intelligent Transport Systems*, 12(10), pp.1210-1219.

Ugli, I., Hussain, A., Kim, B., Aich, S. and Kim, H., 2022. A Transfer Learning Approach for Identification of Distracted Driving. *2022 24th International Conference on Advanced Communication Technology (ICACT),*

Wang, J., Wu, Z., Li, F. and Zhang, J., 2020. A Data Augmentation Approach to Distracted Driving Detection. *Future Internet*, 13(1), p.1.

Wirth, R. and Hipp, J., n.d. [online] Cs.unibo.it. Available at: <http://www.cs.unibo.it/~danilo.montesi/CBD/Beatriz/10.1.1.198.5133.pdf>

Xing, Y., Lv, C., Wang, H., Cao, D., Velenis, E. and Wang, F., 2019. Driver Activity Recognition for Intelligent Vehicles: A Deep Learning Approach. *IEEE Transactions on Vehicular Technology*, 68(6), pp.5379-5390.

Xing, Y., Tang, J., Liu, H., Lv, C., Cao, D., Velenis, E. and Wang, F., 2018. End-to-End Driving Activities and Secondary Tasks Recognition Using Deep Convolutional Neural Network and Transfer Learning. *2018 IEEE Intelligent Vehicles Symposium (IV)*,