SOLENT UNIVERSITY
FACULTY OF BUSINESS LAW AND DIGITAL TECHNOLOGIES

MSc Applied AI and Data Science
Academic Year 2022-2023

ANON

HUMAN EMOTION RECOGNITION USING DEEP LEARNING

Supervisor:

January 2023

This report is submitted in partial fulfilment of the requirements of Solent University for the degree of MSc Artificial Intelligence and Data Science

ACKNOWLEDGEMENTS

ABSTRACT

As computer processing capability has risen and enormous data sets has continued to arise, machine learning techniques have evolved fast in recent years. Automatic emotion recognition using facial gesture is an intriguing research topic that has been discussed and applied in a range of domains, like safety, health, and interactions between people. Researchers in this field are interested in developing ways for understanding, coding, and extracting facial expressions to better computer prediction. Because of deep learning's exceptional success, many designs of this concept are being applied to get greater performance. This study describes the development of an AI system that can recognize emotions using facial expressions. It describes the emotion detection technique, which consists of three primary steps: face detection, feature extraction, then emotion classification. There are seven universal face expressions: neutral, pleased, sad, angry, disgusted, fearful, and surprised. As a result, detecting these emotions on the face is critical. Pre-trained models such as MobileNetv2, VGG19, and a custom CNN model were developed for this study. Furthermore, this study will investigate the differences between several advanced models utilising both conventional and modern machine learning approaches. The suggested method's performance is assessed using facial emotion recognition (FER-2013). Furthermore, it will thoroughly investigate a range of methods in modeling process and provide a hopeful conclusion for emotion identification.

Keywords: emotion detection, facial expressions, facial emotion recognition, deep learning, MobileNetV2, VGG19, CNN, machine learning, artificial intelligence

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS

S.NO.	Abbreviation	Full Form
1.	IT	Information Technology
2.	CNN	Convolutional Neural Network
3.	AI	Artificial intelligence
4.	CV	Computer Vision
5.	DL	Deep Learning
6.	RaFD	Radboud Faces Database
7.	ML	Machine Learning
8.	EDA	Exploratory Data Analysis
9.	GS	Gaussian Noise
10.	VGG	Visual Geometry Group model
11.	SVM	Support Vector Machine
12.	LP-SVM	Linear Programming Support Vector Machine
13.	DCNN	Deep Convolutional Neural Network
14.	WMDNN	Weighted Mixture Deep Neural Network
15.	FER2013	Facial Expression Recognition 2013
16.	DBN	Deep Belief Network
17.	JAFFE	Japanese Female Facial Expressions Database
18.	CK+	Cohn-Kanade Dataset
19.	CUHK	China University of Hong Kong
20.	ORL	Olivetti Research Laboratory
21.	CONN	CONnectivity-based parcellation
22.	RELU	REctified Linear Unit
23.	ILSVRC	ImageNet Large Scale Visual Recognition Challenge
24.	ROC	Receiver Operating Characteristic
25.	TL	Transfer Learning
26.	FACS	Facial Action Coding System

CHAPTER 1 – INTRODUCTION & BACKGROUND

There's no art to find the mind's construction in the face.

He was a gentleman on whom I built an absolute trust.

Shakespeare, Macbeth (1.4.14-5, 2010/1699) [1]

King Duncan was cheated by Thane of Cawdor in Shakespeare's well-known play Macbeth (2010/1699), and as a result, he came to the conclusion that one cannot tell a person's thoughts or level of trustworthiness just by looking at them. This proverb can still be relevant today, especially in cases where communication is involved. However, it is also clear that humans are perfectly suited to sense other people's emotions from their face as well as speech. The speed at which you make a decision might also be impacted by emotional decision-making. Anger can breed impatience and hasty judgement. When you're thrilled, you could take emotional decisions without thinking through the repercussions as you ride the surf of your sense of optimism and assurance for the future.

The sciences of computer vision and machine learning have advanced significantly during the last few decades. Computers can now digest information and comprehend their surroundings more effectively than ever thanks to advancements in these domains. Convolutional neural networks have made it easier to process images and videos and recognize different types of content. One of the most intriguing challenges that the field of deep learning appears capable of solving is the automated recognition of human emotion. Since emotion detection is a necessary talent for an "intelligent" entity and then we can create systems work on this problem like a person does, it stands as a significant challenge for researchers as computers become more humanoid creatures every day.

Alongside the advancement of information technology, applications are being created to simplify human life and work. The current trend in contemporary information technology is the development of AI technology, or generally "artificial intelligence" (AI). Innovative technologies have changed in the contemporary age, like today, due to the adoption of a new manner of interaction, such as the use of icons, tracking displays, touch screens, voice assistants that produce public work, as well as in implements like security control systems. This is also the first step in recognizing human emotions.

To convey their feelings, people speak, make gestures, and make facial expressions[2]. Knowing a user's facial expressions is among the most crucial things beneficial for mutual communication, according to psychological study, which has established that in human interaction, the message via facial movements stands for much more than quarter of all info [3]. Computers may be used to record facial changes and expressions as well as to analyze how people's emotions are changing. This process is known as facial expression analysis. Face analysis is the corner stone of the affective computing system and the foundation of human-machine interaction. Facial detection has several potential applications in virtual reality, video advertising effect analysis, video conferences, and bionics, particularly with the advancement of computer vision technology in recent years.

Figure 1: All facial expressions emotions recognition

The field of artificial intelligence known as Computer Vision(CV) focuses on the comprehension fully automated images Understanding how computers see and recognize images is the goal of computer vision research. Systems and computers can extract some of the information with the use of computer vision from digital images, videos, and other visually inputs and to take actions or make recommendations in response to that knowledge. Understanding facial expressions can help you identify emotions. Making it possible for computers to comprehend the imagery is one of the most popular uses of AI.

Figure 2: Facial movements for recognize emotions

Detecting images of dogs and cats and intelligent image categorization are two common computer vision applications.

Computer vision can be divided into three main categories: segmentation, detection, and classification.

The precise shape of the object within a picture is provided by segmentation models. In contrast to Classification models, which classify the contents of an image, and Detection models, which draw a bounding box around particular objects, the pixel-by-pixel details for a given object are provided.

A computer vision technique called object detection focuses on identifying different items in an image or video. Although it is similar to classification, it identifies more precise things by applying classification to individual objects in an image or video and utilising bounding boxes to show us where each object is in the image or video. One type of object detection is face detection.

In order to answer the question "What is in this image/video?," classification is a sort of labelling in which an image or video is given one or more concepts. [4]

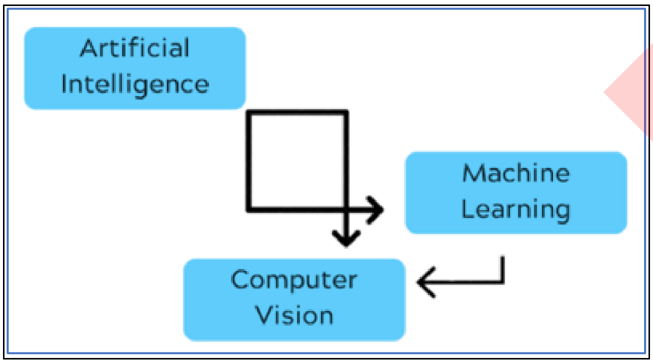


Figure 3: CV relation with AI

DL is used in combination to create machines with the necessary intelligence make decisions based relating to visual data. Here's an explanation that highlights the deep learning field

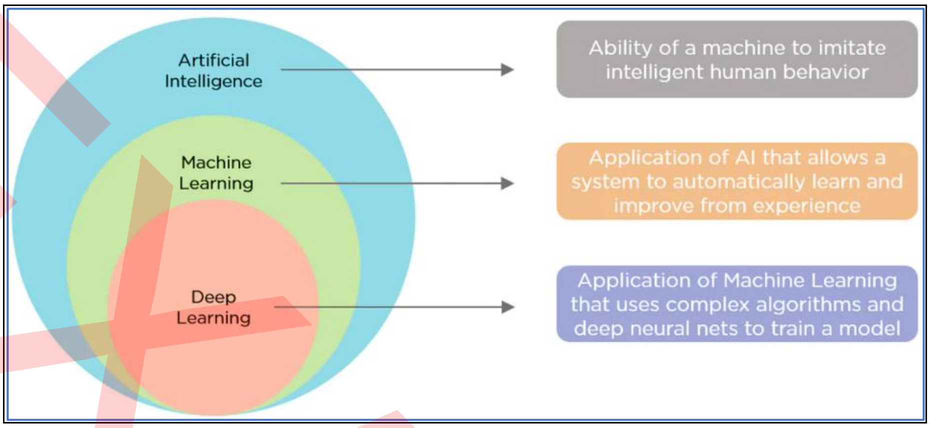


Figure 4: Comparison of Deep Learning

1.1. Aim(s)

- To create a deep learning-based intelligent model for accurately identifying human emotions from images.
- Using Deep Learning algorithm model based on CNNs trained using facial expression dataset, design, develop, verify, and examine an automatic system that accurately detects seven facial expressions (surprise, happy, sad, fear, anger, disgust, and neutral).

1.2. Objectives

These are the project's key objectives:

i. Look into and comprehend why DCNN models are superior to other deep learning models?

ii. Investigate the benefits and drawbacks of using deep learning algorithms in the field of face expression photos.

iii. Look into and comprehend how face expressions are recognized from images.

iv. To deploy deep learning model through Interactive GUI (a complete system with intuitive interface) which would detect the emotions through image or webcam of the user.

1.3. Research Questions

i. What are the most recent cutting-edge deep learning models for detecting human emotions in photos, and how does they compete in terms of performance?

ii. How can we use deep learning techniques to accurately classify emotions in human faces captured in images?

iii. Can pre-training deep learning models on large datasets of facial expressions improve their performance on smaller datasets of emotion detection?

1.4. Ethical Considerations

Discussing ethics is important, particularly when conducting effective and error-free research. In order to conduct successful big data research, the UK Data Center Department gives guidance that address ethical challenges. These standards serve as the backbone for the ethical conduct of this research. The main ethical issue with facial recognition is that these methods are commonly utilized without authorization or notification. Even if you have access to surveillance footage or live video feeds of individuals who are employees, clients, or members of the public, it's never a smart idea to use that information without first consulting those people.

1.5. Project Philosophy

The goal of the study is to identify human emotions from images of face expressions. This is a classification challenge since the grading basic human emotions are classified into 7 classes implementing these elements; this research's concept is derived from computer vision, deep learning with artificial intelligence. Images from the dataset shall be fed into our workflow, converted into an array by approaches of computer vision, and then intelligent categorization using DL will be performed to these images.

CHAPTER 2 – LITERATURE REVIEW

The background information for this project is described in this chapter. It describes the general methodology for recognising facial expressions. I made sure to look over the relevant literature in order to ensure that emotion recognition could be added to Deep Learning models. After reviewing the material, I ought to have known the process I would use to come up with my solution.

- **DeepEmo: Real-world Facial Expression Analysis via Deep Learning**

Weihong Deng, Jiani Hu, Shuo Zhang, Jun Guo in 2015 [5], they study investigates the features machine learning techniques, feature representations, and the training set required for a system to perform consistently in a more practical circumstances. RAF-DB, a new database with over 30,000 extremely varied face photos from social networks, is introduced. According to crowdsourcing findings, the unequal multi-label classification problem that characterizes the actual emotion detection problem is typical, and the stable datasets presently Used in literature may be causing investigation of algorithmic updates to be misinterpreted. In order to tackle the actual-world problem of emotion identification, a deep learning architecture called DeepEmo is developed. It works through the study of high-level extracted features, which are very good at differentiating between actual face expressions. Their accuracy is 65.1% after applying SVM weights applied to the DeepEmo feature vector. The top three are Softmax, LP-SVM, and SVM + Softmax. They next assess the effectiveness of the LPSVM in combination also with DeepEmo feature space that also scores 68.2% accuracy, 7.6% better than the handmade features' top performance.

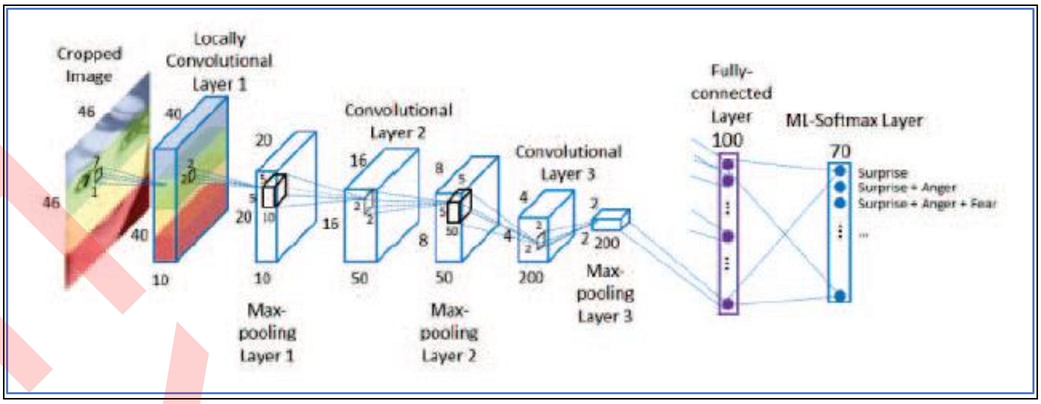


Figure 5: The DCNN architecture for feature learning

- **Development of Deep Learning-based Facial Expression Recognition System**

Heechul Jung and Injae Lee et al., 2015[6] attempt to use deep learning methods to identify facial expressions that correspond to various human emotions. Their face expression recognition technology works in the following way: First, using Haar-like characteristics, the face is found in the input image. Second, utilising faces that have been discovered, the deep system is utilized to identify facial expression. Two distinct deep networks, including DCNN as well CNN, can be employed at this stage. Convolutional neural networks performed better than deep neural networks when they experimentally evaluated the two types of deep networks.

They experimented using the CK+ database. The 327 picture sequences in the CK+ database contain 7 different moods. They located the face in the image, cropped it, and scaled it to 6464. Afterward, the 327 face10 groups of photos were created, with one group used for training and 9 groups utilised for testing. For the face detection implementation, they used the OpenCV library .From the input image, these discovered faces are cropped and normalised to a predetermined size. These facial picture inputs are used by CNN or DNN. Final findings for facial expression recognitions are the neural network's output. They used DNN and CNN for facial expressions and when they compare they found out that CNN had better performance than DNN.

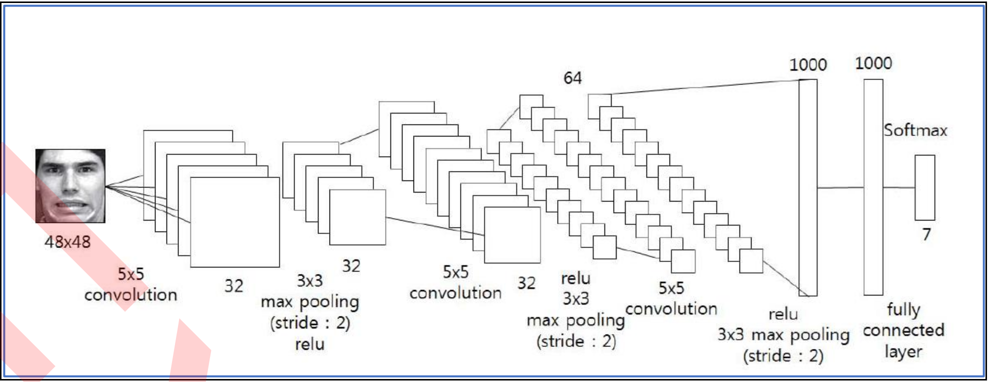


Figure 6: CNN model for facial expression recognition

- **Emotion Recognition Method and Its Application using Face Image**

Hyeon-Jung Lee and Kwang-Seok Hong in 2017[7] studies, they developed models utilizing facial expressions based on emotions to identify emotions using DL technology. Applications that represent six reactions already exist, but not seven feelings, positives, with negatives shown as graphs and percentages. Thus, they identified seven expressions, including rage, disgust, fear, happiness, sadness, and surprise, and they additionally split the computed feelings values into positive, negative, and neutral emotions. After that, we developed an app that rates seven emotions and displays both good and negative feelings for the user. Therefore, when they implemented such detection techniques into applications, the application execution rate for seven emotions was 50.7%, and for positive and negative emotions it was 72.3%. By modifying some aspects of the deep-learning system and introducing more emotional datasets from now on, they hope to increase the recognition rate. They were able to determine also that final recognition rate was 51.2% by applying deep learning. Positive, negative, and neutral sentiments were used to assess the identification of emotions. The 75.92% outcome.

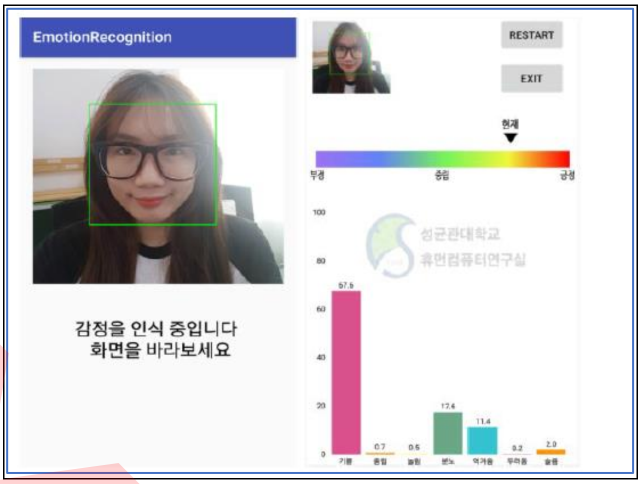


Figure 7: Emotion Recognition App

- **Facial Expression Recognition via Deep Learning**

They introduce a new architecture in this paper(Abir Fathallah, Lotfi Abdi, Ali Douik et al., 2017) [8]. network for facial expression recognition that is based on CNN. They great architectural design using the VGG to enhance outcomes. They tested their architecture to assess it. it has numerous, generally open databases (such as CK+, MUG, and RAFD). Results obtained demonstrate how very successful the CNN strategy is in the identification of facial expressions in various open databases, which increase the ability to analyse facial expressions. Results and identification rates show that our approach outperforms cutting-edge approaches. To train the model for this research, they used pictures that showed face was fixed in one place. They attained accuracy rates of 93.33% for RaFD Dataset, 71.04% for CK+ Dataset, and 87.65% for MUG Dataset.

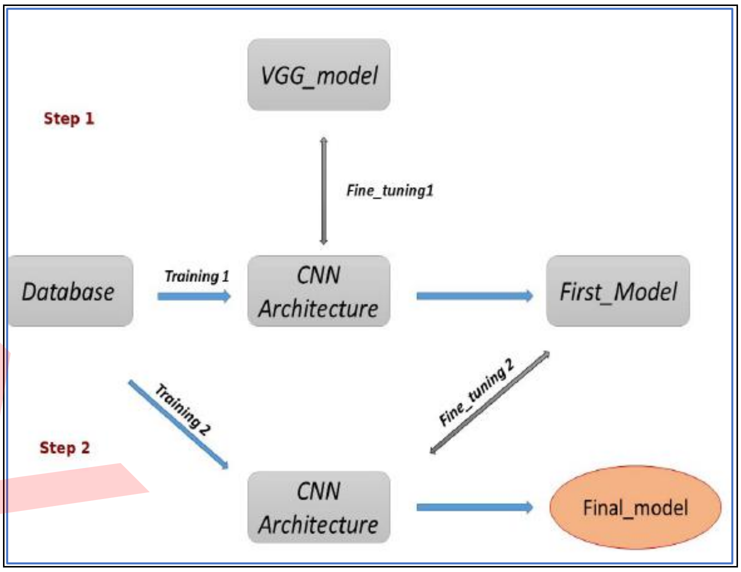


Figure 8: Proposed approach

- **Facial Expression Recognition Using Weighted Mixture Deep Neural Network Based on Double-Channel Facial Images**

In this paper, (Biao yang 1, Jinmeng cao1, Rongrong ni2, Yuyu zhang1 et al., 2017)[9]A WMDNN is suggested in this research as a method for automatically extracting the characteristics that are useful for FER tasks. To limit the regions for FER, a number of pre-processing techniques are used, including face detection, rotation rectication, and data augmentation. WMDNN processes two channels of face pictures, including local binary pattern (LBP) facial images and facial grayscale images. By fine-tuning a partial VGG16 network, the parameters of which are initialised using VGG16 model trained on ImageNet database, expression-related properties of facial grayscale images are retrieved. A shallow convolutional neural network (CNN) based on DeepID is used to extract features from LBP facial images. Both channels' results are combined in a weighted way. Softmax classification is used to calculate the outcome of final recognition. According to test findings, the suggested algorithm can accurately identify the six most common facial expressions (joy, sorrow, anger, disgust, fear, and surprise). The benchmarking data sets "CKC," "JAFFE," and "Oulu-CASIA" had average recognition accuracies of 0.970, 0.922, and 0.923, respectively. If sufficient data cannot be acquired, fine-tuning to FER tasks with a well-trained model is useful.

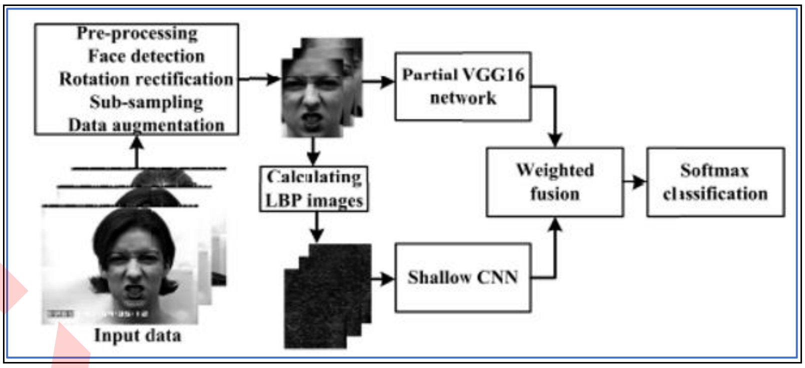


Figure 9: Pipeline for the planned WMDNN-based FER method

- **Detection and Recognition of Human Emotion using Neural Network**

The major goal of this research(J. Jayapradha Soumya Sharma and Yash Dugar et al. 2018)[10] is to create a reliable system that can identify human emotion from live broadcast. A few feelings, such as anger, sadness, joy, surprise, fear, disgust, and neutrality, are shared by all people. The FER2013 dataset was used. The facial detection process is accomplished by extracting the Haar Cascade characteristics from a face using the Viola Jones algorithm, and the emotion is then confirmed and recognised using a deep neural network.

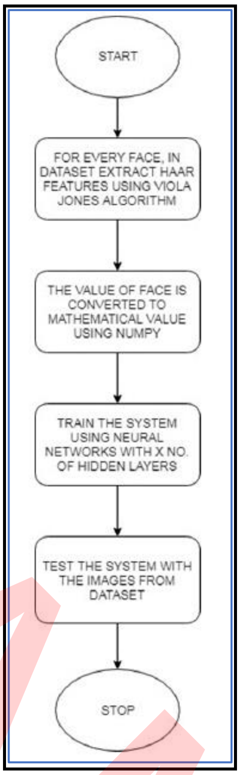


Figure 10: Initial workflow of training the model

- **Convolutional Neural Networks Models for Facial Expression Recognition**

The happy, disappointed, angry, and natural emotion expressions related to customer service were compared between two configurations(Burhanudin Ramdhani, Esmeralda C. Djamal*, Ridwan Ilyas, 2018)[11] using batch sizes of 8 and 128 with two datasets: FER-2013, self-created dataset, and cross dataset in this study to build an image recognition system of emotion expression using CNN. The top results on the researchers' dataset were 73.98% and 58.25% respectively, while the top results on the researchers' setup with batch size 8 were higher results. While the best accuracy obtained by the prior research setup utilising the FER-2013 dataset when batch size 128 is used is 69.10%. Whether using training data or test data, the configuration developed by the researcher consistently beat the settings from earlier study in each test.

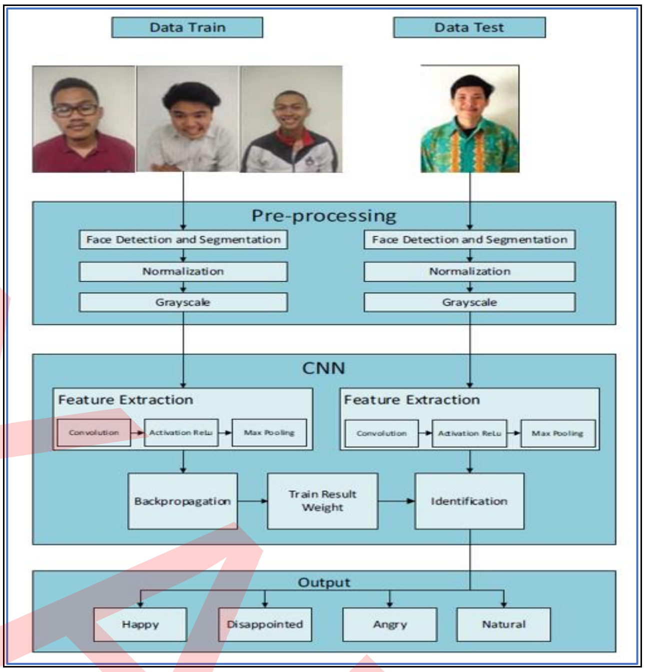


Figure 11: Process diagram

- **Research on Facial Expression Recognition Based on Deep Learning**

In this study(Xiaofang Jin, Ying Xu et al., 2019)[12] four distinct models are built on the CNN to realise the explanation and measurement of face expressions in images, so images can be predicted by the network to describe what types of emotions. Finally, the model's output is assessed by comparing the accuracy and loss scores of the four models. By contrasting the four neural network models for facial expression recognition, they can observe that adding more network layers might not improve accuracy. CNN filters will enhance the model's performance by adding layers to the same number of network layers. They conducted the training using the FER2013 database. The maximum accuracy in this study can reach 63.07%.

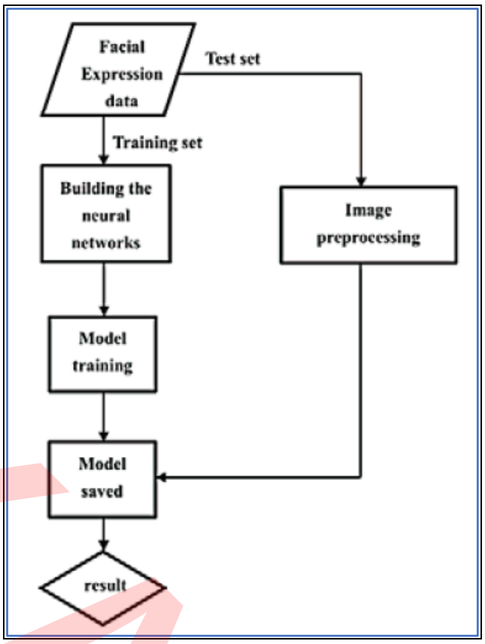


Figure 12: Experimental flow chart

- **Deep Learning Methods for Facial Expression Recognition**

FER has utilised a variety of DL algorithms, including CNN and DBN. They(Chowdhury Mohammad Masum Refat, Norsinnira Zainul Azlan et al.,2019)[13] examine numerous deep learning techniques and their outcomes in this research. They have determined that DCNN is the best method for identifying and categorising facial expressions. Using datasets from the Anaconda software's JAFFE, they tested the method in their study. With JAFFE datasets, the DCNN have an accuracy rate of about 97.01%. In order to train this CNN network, supervised backpropagation was used. For the best accuracy, the confusion matrix of facial recognition accuracy for various expressions employed training weights data.

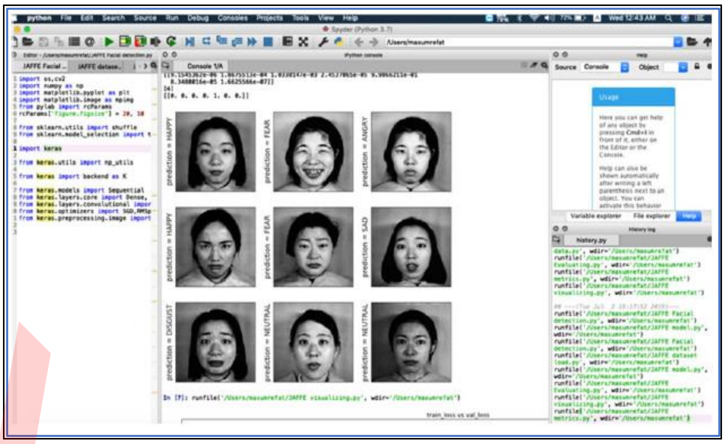


Figure 13: Facial expression classification by anaconda

- **Emotion Detection using Deep Facial Features**

Three crucial factors are studied in relation to face emotion recognition techniques: Feature extraction, pre-processing, and classification techniques are a few examples. This study(Hari Kishan Kondaveeti, Mogili Vishal Goud, 2020)[14] compares the deep learning architectures offered by Keras for detecting emotions in photos using Deep Facial Features and TL from well-known pre-trained models including VGG-16, ResNet152V2, InceptionV3, and Xception attributes for their input photos that bottleneck. Based on a dataset that combines the CK+ and JAFFE, these models' performance is assessed. The accuracy values obtained for the aforementioned designs are 83.16%, 82.15%, both 77.1% and 78.11%.

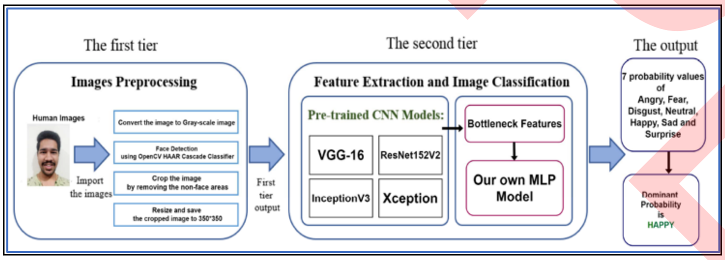


Figure 14: Proposed system for emotion detection

- **A Deep Learning Model for Face Expression Detection**

They(Sneha Lalitha K and Aishwarya J et al. 2021)[15] describe a deep learning method for classifying facial expressions in this work. The model is created to identify seven many emotions, including joy, sadness, anger, fear, disgust, surprise, and neutrality. Face detection was performed utilizing the Har cascade classifier. FER2013 dataset was used for training the model, and the outcomes were inspected. The accuracy of the SVM model is 57.17%, for the Emotion classifier employing Wiki Art is 52.9%, and that of a different CNN model utilizing a various datasets and layering scheme is 58% As per CNN, FER's CNN accuracy for FER2013 was 64.24%.Picture from gadget, image collection using a camera, and recording videos using a webcam are the three types of input available. The model's outputs are a voice output, an emoticon connected to the attitude, and a picture with the identified expression.

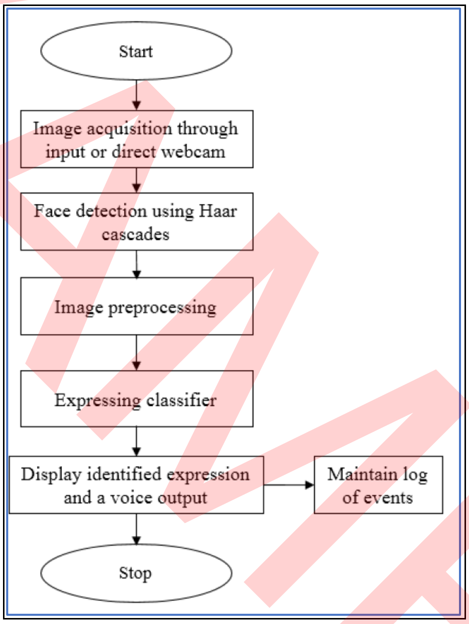


Figure 15: Steps performed during classification flowchart design

- **An Advanced Facial Expression Detection using Deep Neural Network**

Mr. Arnold Sachith A Hans, Mr. Mohit Bansal, Dr. Smitha Rao in 2021[16],The category of non-verbal communication covered by this research includes facial expressions, which are crucial in determining a person's emotional state. Dataset used: RAVDEES Dataset: There are 7356 files altogether in the dataset, making up 24.8GB of data. There exists 24 famous actors engaged, 12 of them are men and 12 of whom are women. The dataset contains 8 distinct emotions. The ability to read emotions via facial expressions can be applied in a variety of industries, including healthcare, election campaigns, and revealing a candidate's behavior during a job interview or in a classroom. They used ANN-LSTM model for detection. In addition to the photos supplied here to neural nets, the Open Face tool is utilised to remove the subject's facial expression units from the dataset, which helps in the training of the neural network. To accurately categorize the emotions based on facial expressions, they have created a model-based method. The information is trained using a Neural Network Based on Causal Relations. With a 90.95% accuracy rate, that model produced state-of-the-art outcomes for identifying facial emotions.

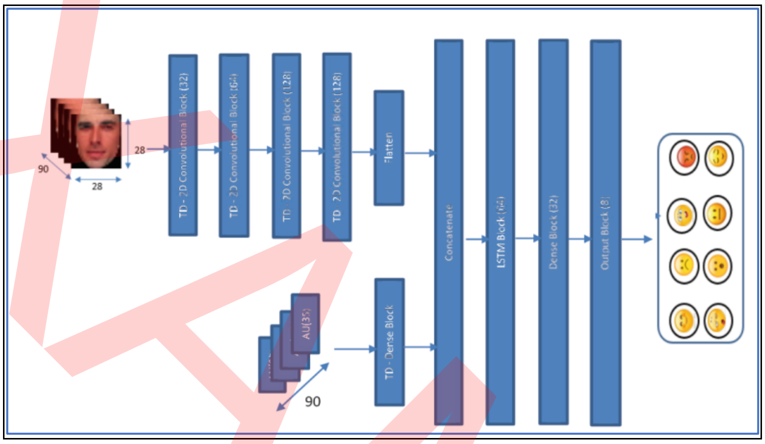


Figure 16: CNN and Facial Action Units in a LSTM-ANN architecture

- **Recognition Of Facial Expressions Using A Deep Neural Network**

This field is also quite popular since it has a wide variety of applications and demonstrates that CNN outperforms conventional methods(Vipan Verma, Dr. Rajneesh Rani, 2021)[17]. As a result, the worked toward creating a Deep CNN architecture that can handle real-world photos with a variety of resolutions, angles, poses, illumination, and brightness, among other factors. As a result, they used the Kaggle challenge dataset FER-2013 to deploy our CNN architecture and train the model to recognise the fundamental seven phrases. The inputs are stored in a CSV file as pixels. Given that we were able to attain a validation accuracy of 70.15 percent, the suggested strategy appears to be successful. This strategy can be used not only for other datasets but also in real-world applications.

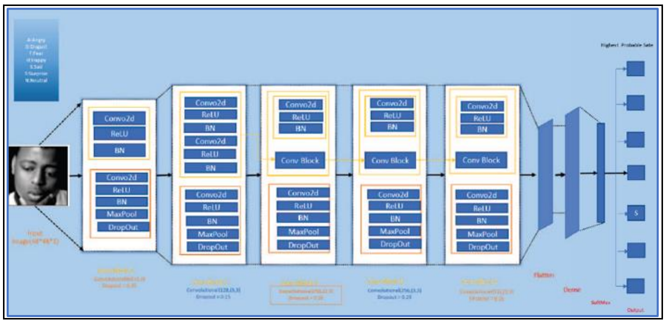


Figure 17: Model architecture

For this system's training, 9000 samples from the FER-2013 data will be used, along with 1000 additional new samples for validation. This has applications in the areas of robotics, surveillance, and security, among others.

- **Facial Expression Recognition Using Xception And DenseNet Architecture**

In this paper, the majority of research centered on machine learning method as opposed to in-depth knowledge and feeling Classes are also more narrowly defined(Hannatassja Hardjadinata, Raymond Sunardi Oetama, Iwan Prasetiawan, 2021)[18]. face expression identification can be put into practise utilising a deep learning strategy. The style of architecture that is popular and regarded as the greatest in CNN is used to classify images. Therefore, This paper creates a model of a CNN using DenseNet architecture and Xception. the model that was developed and trained using the RAF-DB dataset. The two are accurate When models are compared, Xception obtained a 70% accuracy rating and 79% went to DenseNet. The accuracy of DenseNet and Xception will next be analyzed to decide which model is the best fit for this investigation.

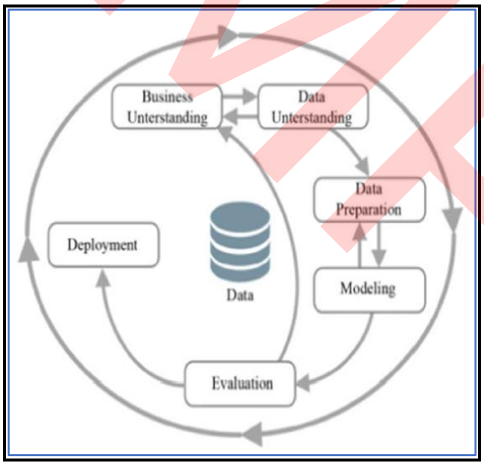


Figure 18: CRISP-DM Methodology

- **A Robust Face Recognition model using Deep Transfer Metric Learning built on AlexNet Convolutional Neural Network**

To expand the facial feature space on the limited face, they provide an AlexNet-CNN architecture-based TL system in this research(Gurukumar Lokku, Dr. G. Harinatha Reddy, Dr. M. N. Giri Prasad et al., 2021)[19]. The suggested framework is equipped with an input layer, convolutional layers, activation layers, pooling layers, and a fully connected layer connected to a classifier layer. It operates on the standard CUHK and ORL databases in a standard frontal pose with faces, under an ordinary illumination state, and with unbiased input with great accuracy in facial recognition tasks due to expressiveness. For frontal facial images, subjects with limitations like position, illumination, and expression are crucial. Ignoring the parameters throughout the testing and training phases will cause a sharp decline in recognition rate and have a permanent influence on accuracy. Biased classifier in deep CNN can result from training the network with incorrect lighting facial data base. The experimental outcomes show that the CNN model, which is built on AlexNet, can achieve an accuracy rate of 98% and deep learning on facial datasets And 99% for the testing phase for the ORL and CUHK datasets.

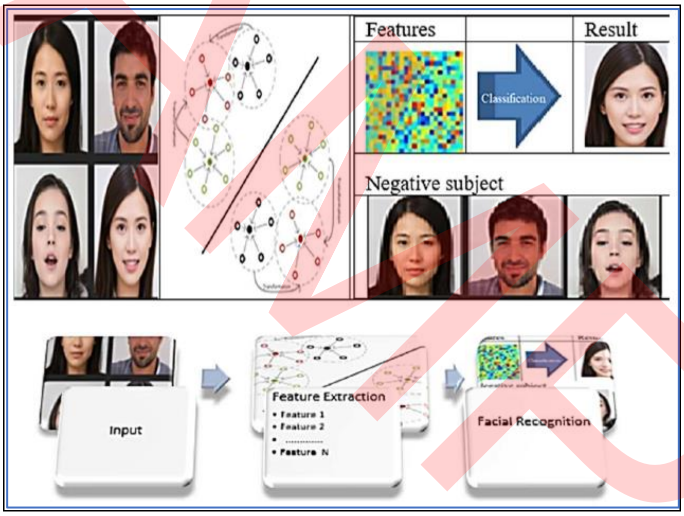


Figure 19: Face recognition framework

- **Facial Emotion Recognition Using Transfer Learning of AlexNet**

As it depends on the manually derived features(Sarmela A/P Raja Sekaran, Chin Poo Lee, Kian Ming Lim et al., 2021)[20], bias may be present domain expertise of the researcher. Contrarily, picture categorization is a task that DL approaches, particularly CNN. The drawback of DL techniques is that they need a lot of data to effectively train and execute recognition. Therefore, they suggest a deep learning strategy using the pre-trained AlexNet architecture for FER and transfer learning. Using emotion datasets, they complete entire model finetuning on the Alexnet, which was previously trained on the Imagenet dataset. Two prevalent face expressions are utilised to train and test the proposed model datasets, namely the FER dataset and the extended CK+ dataset. By obtaining accuracy of 99.44% and 70.52% for the CK+ dataset and the FER dataset, respectively, the proposed framework exceeds the current state-of-the-art approaches in facial emotion recognition.

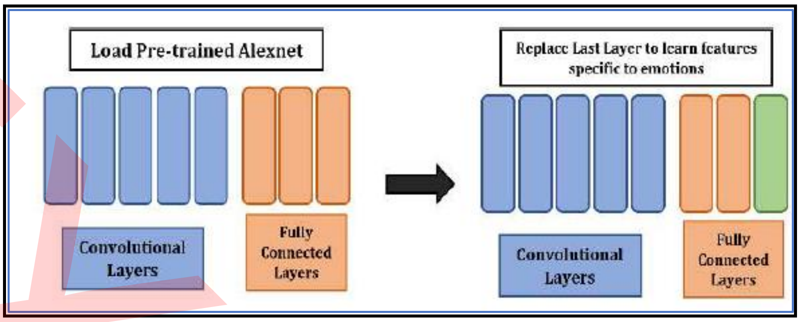


Figure 20: Transfer learning of Alexnet

- **A-MobileNet: An approach of facial expression Recognition**

Real-time performance and lightweight networking are becoming more and more important in the mobile Internet era. This study(Yahui Nan a,b, Jianguo Ju a,*, Qingyi Hua a, Haoming Zhang a, Bo Wang a, 2021)[21] suggests a simplified A-MobileNet model. The MobileNetV1 model first incorporates the attention module to improve local extraction of features of facial emotions. The model parameters are then optimised to decrease intra-class distance and increase inter-class distance by combining the centre loss and softmax loss. Their strategy considerably increases recognition accuracy without adding more model parameters when compared to the original MobileNet series models. According to experimental findings, the suggested A-MobileNet model can significantly increase the expression classification accuracy of FER on the FERPlus and RAFDB datasets without adding more model parameters. On the RAF-DB and FERPlus, mobilenet's recognition accuracy is 84.49% and 88.11%, respectively.

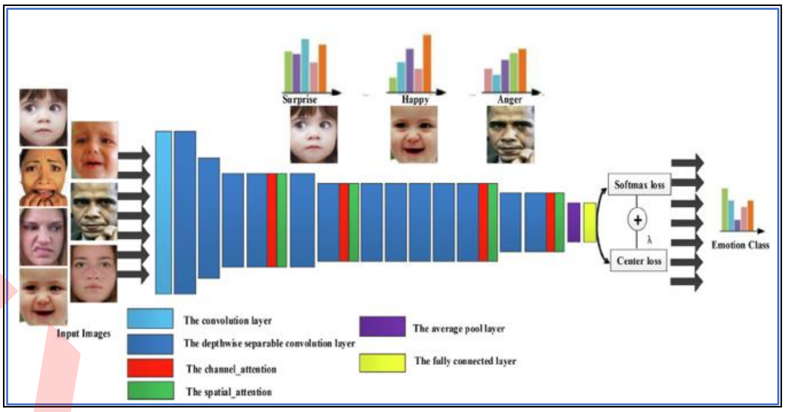


Figure 21: Framework of A-Mobilenet network model

- **Design and Implementation of Face Emotion Recognition System Based on CNN Mini_Xception Frameworks**

A smart, real-time, and generalizable UI interface of facial expression(Lili Sun1, Chenhao Ge1 and Yuanchang Zhong1et al., 2021)[22] must also be designed and implemented because people use computer software on a daily basis. As a result, the FER2013 database was used to train the facial expression recognition model in this paper, which was based on the Mini Xception architecture of CNN. Second, PyQT5, OpenCV, Keras, and other libraries are used in the design and implementation of the system UI interface. The system's overall impact is amazing, and the final results demonstrate that it is built on an algorithm model. In addition to recognising emotions in saved photos, the system's UI can also do so in real time while using a camera.

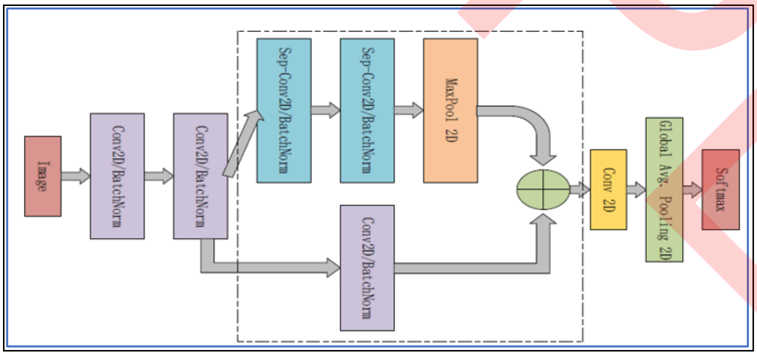


Figure 22: Framework of mini_XCEPTION

- **Deep learning algorithm for Facial Expression Classification**

Bin He in 2022 research primarily examines[23] how various depth networks affect face expression recognition ability. For comparison analysis, they employ ResNet, VGG, and

AlexNet. The FER2013 dataset studies demonstrate that varying spatial networks perform differently for recognizing facial expressions, not those deeper networks perform better. They selected three layers with varying depths, namely Alex Net, VGG, and ResNet, for training on the dataset of Fer2013 facial expressions. They plan to contrast how different depth networks affect facial emotion recognition tasks' accuracy. The accuracy percentage for human eye recognition in this dataset is 655%. They find that the larger ResNet is only 61.9% accurate in this dataset, whereas the VGG accuracy reaches 66.4%.

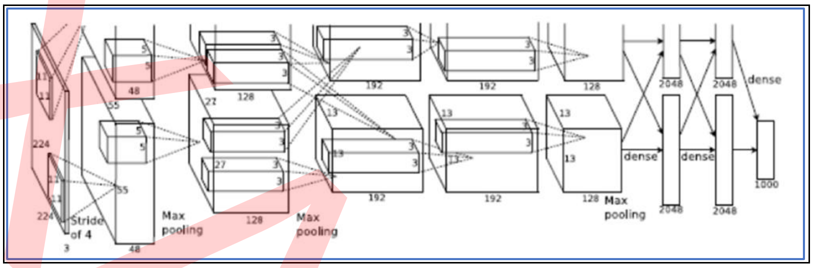


Figure 23: Alexnet Network Structure

- **Facial Emotional Expression Recognition Using Hybrid Deep Learning Algorithm**

Phasook Phattarasooksirot and Adna Sento in 2022[24], they study the Convolutional Neural Network (CNN)-based face emotional expression identification system. They use FER2013, CK+ and JAFFE dataset. Utilizing cutting-edge technologies like Inception Net, ResNet, and VGG, which were created excellence in their particular feature extraction technique, is the most trustworthy strategy. A widespread use model, like Convolutional Auto Encode, which again is able of highly effective noise mitigation, appears in addition to the aforementioned models.

Table 1: Comparison of models with accuracy

Model	FER2013	CK+	JAFFE
VGG [16], [23]	72.7 %	86.2 %	94.7 %
Inception [16], [23]	71.6 %	76.5 %	75.9 %
ResNet [16]	72.4 %	-	-
AlexNet [7]	61.1 %	92.2 %	-
U-Net	59.8 %	70.7 %	72.3 %
Proposed Model	88.0 %	93.5 %	83.4 %

Then, based on the idea of the original models that are modern, several more sophisticated models were created, such as U-Net, which does high-performance picture segmentation utilizing convolution layer and transposed convolution approach. And used the key characteristics from the state-of-the-arts listed above, a hybrid deep learning method using CNN and CAE is built in this study. The updated CNN model employs two combinations to forecast emotional state in people. The experimental outcome demonstrates the suggested model's ability to make predictions Accuracy of 88%

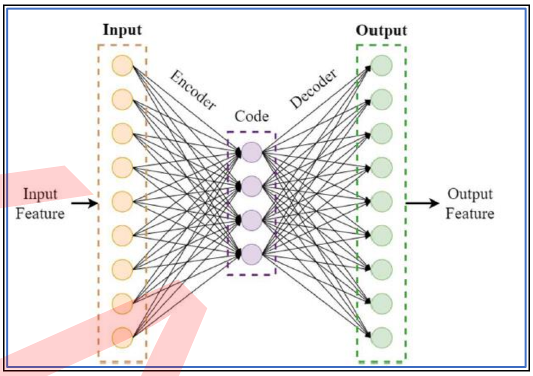


Figure 24: Autoencoder architecture

- **Face Detection and its Features Extraction using Convolution Neural Network Model**

In this work(Yenumaladoddi Jayasimha, Venkatesha M, R. Venkata Siva Reddy et al., 2022)[25], a CNN-based prototype for age estimation and gender classification of identified faces coupled with their emotions is given. The suggested model is built in three steps, starting with a hybrid feature extraction method that uses an SVM classifier to recognise facial expressions. For distinguishing facial expressions, a model based on SIFT and deep learning is described. The performance of facial emotion recognition is optimised using the whale optimization method. This model provides a CNN and proposes a novel method for categorising facial emotions, predicting age, and differentiating between men and women. The WideResNet architecture and Caffe models are used to generate the dataset for the investigation. An empirical analysis shows that the suggested model greatly improves emotion detection when evaluated against a benchmark dataset for static emotion identification. The reported accuracy is 98.79%, and it is contrasted with the current techniques. The use of developing technologies AI and ML has led to the stated model being discovered to be suited for numerous applications that require human computer interaction.

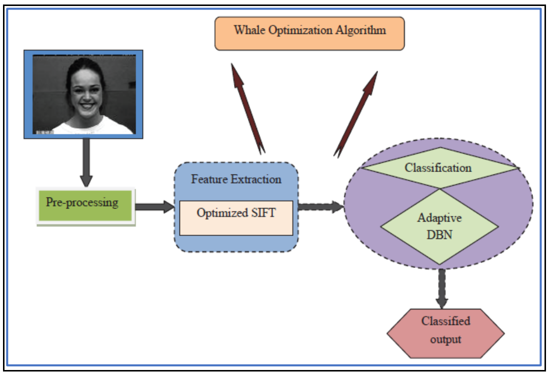


Figure 25: Design of suggested WOA method

- **Facial expression recognition using CNN**

This article's methodology is based on CNN. In this study(1st and 2nd Fahreddin M. SADIKOG˘ LU, Mohamed Idle Mohamed et al., 2022)[26], Alexnet structure is examined using Deep Learning CNN. Applying the transfer learning strategy and changing the fully connected layer with the SVM classifier allowed for improvements. The system was successful since it produced acceptable results on the icv-the MEFED dataset. Around 64.29% of recognition rates for the classification of the chosen expressions were attained by improved models. The outcomes were respectable and similar to the Ideas and a background for additional enhancements are provided by relevant systems in the literature. The Alex Net model and SVM classifiers, which they further improved, achieved the highest recognition accuracy 64.29%, followed by the Alex Net model and SoftMax classifiers.

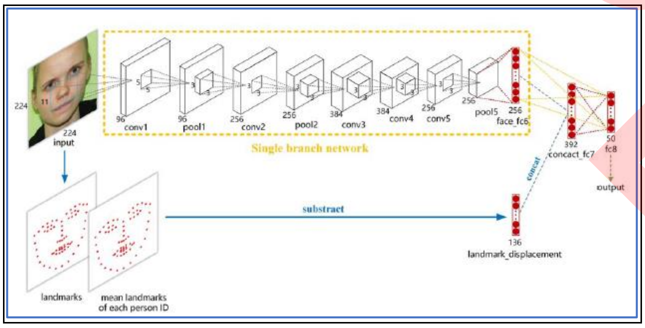


Figure 26: Alexnet architecture

- **A GoogleNet architecture based Facial emotions recognition using EEG data for future applications**

Under this study(Talliboina Ramu, Dr. A. Muthukumar et al., 2022)[27], the GoogleNet-7 deep learning algorithm (RDL) is recommended for precise facial recognition and emotion estimation in a range of conditions, such as dim lighting, sunglasses, long hair, or other substantially obscuring facial objects, and low resolution based facial photographs. Feature extraction for FERS is constrained by pre-processing skills including face identification, rotation, and data collection. The construction of the GoogleNet-7 has enhanced testing, training, and scalability while processing facial images using 165 convolutional layers. Automatic monitoring of buried face features is possible with the GoogleNet-7 architecture, which is based on Python 3.7.8. The recommended methodology can identify facial expressions (sadness, happiness, fear, surprise, disgust) with increased accuracy and calculates classification accuracy using a soft max mechanism. This model exceeds the approach and competes with contemporary CONN models, achieving 99.12% accuracy, 99.51 recall, 0.94 F1 score, and 98.11 sensitivity.

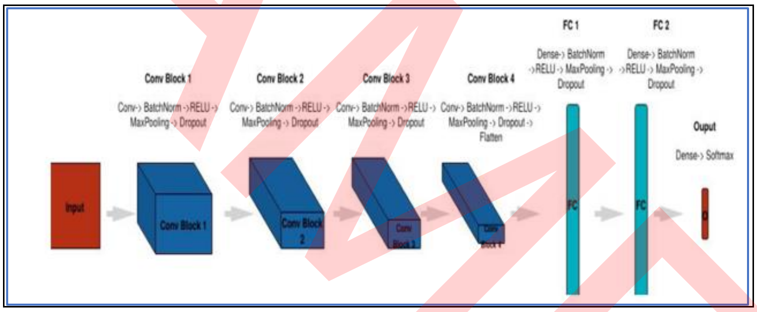


Figure 27: Googlenet-7 architecture

Table 2: Deep learning algorithm comparison from different research

Paper	**Year**	**Headline**	**Accuracy(%)**	**Algorithms Used**
Weihong Deng, Jiani Hu, Shuo Zhang, Jun Guo	2015	DeepEmo : Real-world Facial Expression analysis via Deep Learning	68.2	SVM + Softmax and SVM+ML-KNN
Heechul Jung, Injane Lee	2015	Development of Deep Learning - based Facial Expression Recognition System	-	DNN and CNN

Hyeon-Jung Lee and Kwang-Seok Hong	2017	Emotion Recognition Method and it's application using face image	51.2	Histogram of Oriented Gradient using SVM
Abir Fathallah, Lotfi Abdi, Ali Douik	2017	Facial Expression Recognition via Deep Learning	RAFD – 93.33 CK+ - 71.04 MUG – 87.65	VGG
Biao yang 1, Jinmeng cao1, Rongrong ni2, Yuyu zhang1	2017	Facial Expression Recognition Using Weighted Mixture Deep Neural Network Based on Double-Channel Facial Images	CKC – 0.970 JAFFE – 0.922 Olu Casia – 0.923	VGG16
J. Jayapradha Soumya Sharma and Yash Dugar	2018	Detection and Recognition of Human Emotion using Neural Network	-	Viola Jones algorithm
Burhanudin Ramdhani, Esmeralda C. Djamal*, Ridwan Ilyas	2018	Convolutional Neural Networks Models for Facial Expression Recognition	Batch size 8 - 73.98 Batch size 128 - 69.10	CNN
Xiaofang Jin, Ying Xu	2019	Research on Facial Expression Recognition Based on Deep Learning	63.07	CNN
Chowdhury Mohammad Masum Refat, Norsinnira Zainul Azlan	2019	Deep Learning Methods for Facial Expression Recognition	97.01	CNN
Hari Kishan Kondaveeti, Mogili Vishal Goud	2020	Emotion Detection using Deep Facial Features	VGG16 - 83.16%, REsNet152V2 - 82.15%, Inception V3 – 77.1% and Xception - 78.11%.	VGG-16, ResNet152V2, InceptionV3, and Xception
Sneha Lalitha K and Aishwarya J	2021	A deep learning model for Face	67	CNN

		Expression Detection		
Mr. Arnold Sachith A Hans, Mr. Mohit Bansal, Dr. Smitha Rao	2021	An advanced Facial Expression Detection using Deep Neural Network	90.95	CNN-LSTM
Vipan Verma, Dr. Rajneesh Rani	2021	Recognition Of Facial Expressions Using A Deep Neural Network	70.15	CNN
Hannatassja Hardjadinata, Raymond Sunardi Oetama, Iwan Prasetiawan	2021	Facial Expression Recognition Using Xception And DenseNet Architecture	Xception - 70 DenseNet - 79	Xception DenseNet
Gurukumar Lokku, Dr. G. Harinatha Reddy, Dr. M. N. Giri Prasad	2021	A Robust Face Recognition model using Deep Transfer Metric Learning built on AlexNet Convolutional Neural Network	0.98	AlexNet - CNN
Sarmela A/P Raja Sekaran, Chin Poo Lee, Kian Ming Lim	2021	Facial Emotion Recognition Using Transfer Learning of AlexNet	CK+ - 99.44 FER - 70.52	AlexNet CNN
Yahui Nan a,b, Jianguo Ju a,*, Qingyi Hua a, Haoming Zhang a, Bo Wang a	2021	A-MobileNet: An approach of facial expression Recognition	RAFDB - 84.49 FER - 88.11	MobileNetV1
Lili Sun1, Chenhao Ge1 and Yuanchang Zhong1et	2021	Design and Implementation of Face Emotion Recognition System Based on CNN Mini_Xception Frameworks	-	Mini Xception CNN

Bin He	2022	Deep Learning algorithm for Facial Expression Classification	Alex Net – 65.3 VGG – 66.4 ResNet – 61.9	AlexNet, VGG, ResNet
Phasook Phattarasooksirot, Adna Sento	2022	Facial Emotional Expression Recognition Using Hybrid Deep Learning Algorithm	88	Inception Net, ResNet, and VGG
Yenumaladoddi Jayasimha, Venkatesha M, R. Venkata Siva Reddy	2022	Face Detection and its Features Extraction using Convolution Neural Network Model	98.79	WideResNet
1st and 2nd Fahreddin M. SADIKOG˘ LU, Mohamed Idle Mohamed	2022	Facial expression recognition using CNN	64.29	AlexNet with SVM
Talliboina Ramu, Dr. A. Muthukumar	2022	A GoogleNet architecture based Facial emotions recognition using EEG data for future applications	99.12	GoogleNet -7

CHAPTER 3 – RESEARCH METHODOLOGY

The applications of data science aim to collect data and views from it. The usefulness of the revealed concepts is highlighted. The influence of a project, on the other hand, may be considerably lessened if the outcomes are not adequately conveyed. This section discusses the many stages of the research process that was followed for this project. It is divided into sub-sections namely: Data collection, Pre-processing, model design, development, training and testing and analyze & result.

The project's phases are briefly discussed in this section.

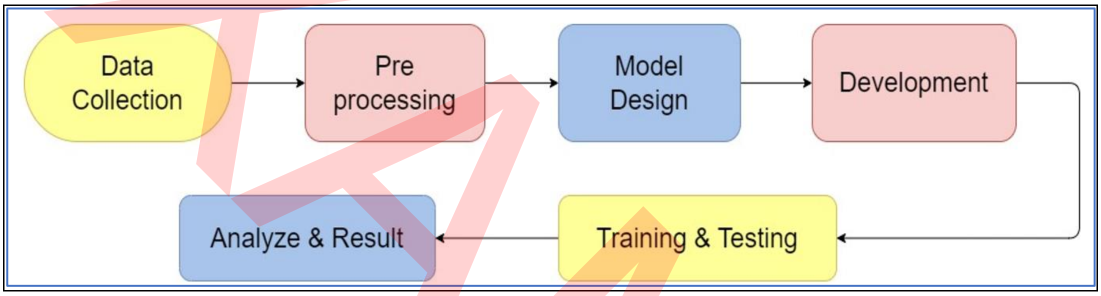


Figure 28: Project phases

3.1 Data Collection

Data collection is the systematic collecting and measurement of information on variables of interest in order to respond to well-defined research issues, test hypotheses, and assess outcomes. This stage was important for the development process because all ml algorithms are data-driven, and choosing the wrong dataset would have complicated the project. Because the major goal of this research is not only to differentiate faces, as datasets with unclear backgrounds would impose an unneeded strain on the face recognition system. For emotion identification, I used dataset: FER2013.

3.2 Data Pre-processing

Data pre-processing relates to the actions required to alter or catalogue data so that it can be interpreted by a machine. The algorithm should quickly comprehend the data's attributes in order for a model to be efficient and exact in predictions.

The data was cleaned and transformed using the techniques outlined below. Firstly, Face detection, which creates bounding boxes around the detected face, Secondly, Feature extraction, which extracts face characteristics from a face picture, and lastly, face recognition, which analyses facial expressions from both static photographs and movies to disclose information about one's emotional condition.

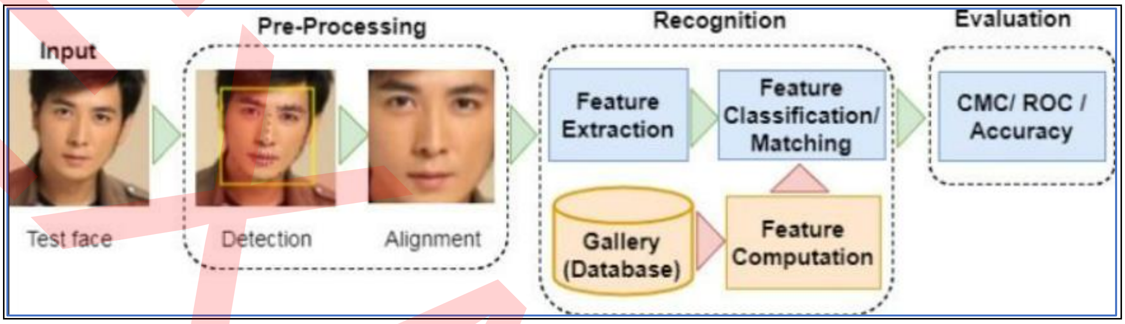


Figure 29: Process Diagram

3.3 Model Development

The primary goal of machine learning is to create a model which can make predictions. This Phase uses various deep learning models such as MobileNetV2, CNN, Yolov5.

3.3.1 Why I use pre trained model?

The model contains the weights and biases that characterise the characteristics of the dataset(s) used to train the pre-trained model. Learned characteristics are often relevant to a wide variety of data. A model trained on a big dataset of bird images, for example, can include learnt components like borders or horizontal lines that are transferable to your dataset.

Pre-trained models can help us in a variety of ways. Using a pre-trained model can save you time. Your model will almost certainly profit from the use of someone else's time and computing resources to learn a large number of features.

3.3.2 MobileNetV2

Since AlexNet, which won the 2012[28] ImageNet competition by utilising a CNN to categorise images. However, as the network grows, the tremendous storage and processing needs imposed by computation of a model have begun to limit the breadth of applications for DL models. Since 2017, Google has recommended MobileNetV1, MobileNetV2, and

MobileNet3 in that sequence, all of which may be utilised with mobile and embedded devices.

We carefully analysed the MobileNet V2 design. The MobileNet V2 model has 53 convolution layers and 1 AvgPool and has about 350 GFLOP. It is divided into two primary sections: Inverted Residual Block, Remaining Bottleneck Block. There appear to be two types of Convolution layers in the MobileNet V2 design: Convolution Convolution 1x1, Depthwise Convolution 3x3.

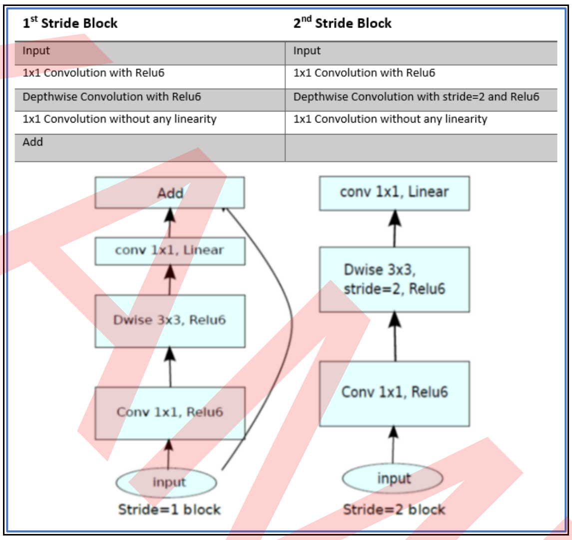


Figure 30: MobileNetV2 convolution layers

The parameters of each Convolution layer in order are:

Figure 31: Convolution layers parameters

3.3.3 13 Layer CNN Model

CNNs are a subset of deep, feed-forward ANNs that are commonly employed for visual data analysis. This type of DL model has lately gained a lot of appeal as a result of major breakthroughs in algorithms and computer capacity.

The network design is depicted in the figure below, which consists of 13 convolutional layers, eight fully connected layers, and three pooling layers [29]. Convolutional kernels in convolutional layers are 3*3 in size, with stride set to 1. The kernel of the pool layers is 2 * 2 with a step size of 2. In convolutional layers, the ReLU serves as the activation function. The sequential network receives 3 channel, 224 224 pixel image as input. The first segment has 2 convolutional layers, which are followed by a pooling layer. Each kernel in these 64 convolution layers is 224 × 224 pixels in size.

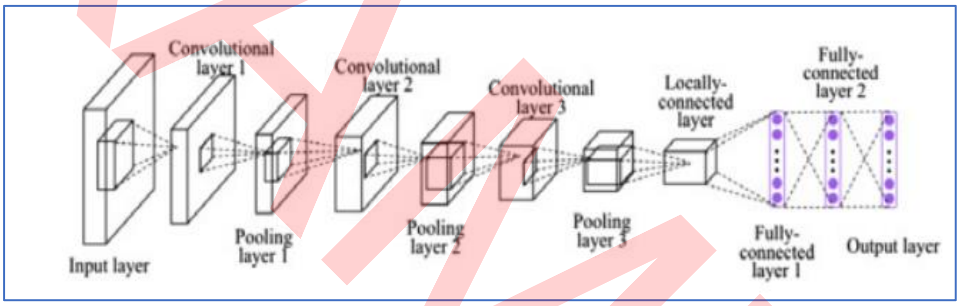


Figure 32: : CNN Architecture

3.3.4 VGG19 Model

VGG is a successor to AlexNet, however it was established by a separate group at Oxford called the Visual Geometry Group, thus the name VGG[30]. It builds on and improves on previous ideas, and it employs deep Convolution layers to boost accuracy. In 2014, VGG surpassed other cutting-edge models and is still preferred for a wide range of tough circumstances. The VGG19 model has the following layers:

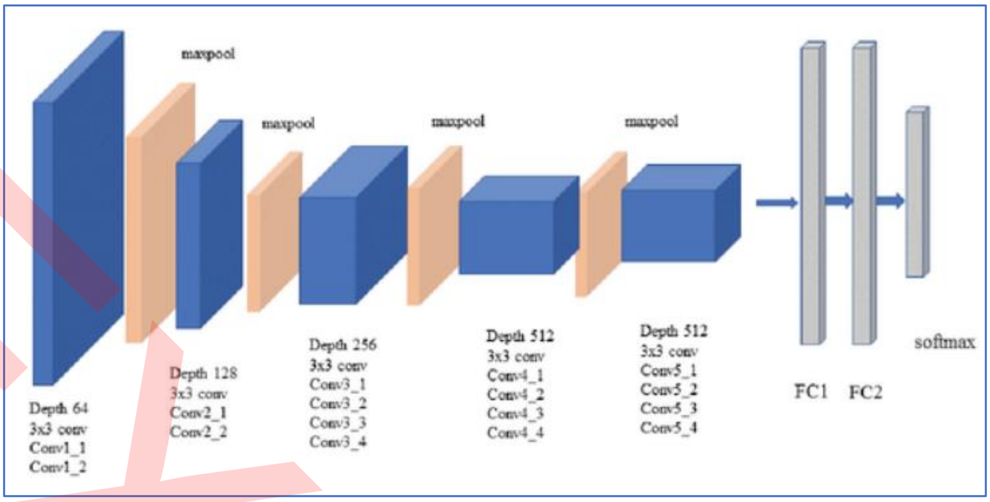


Figure 33: VGG-19 Architecture

- This network was transmitted a fixed size (224 * 224) RGB picture as input, implying that now the matrix has been of shape (224,224,3).
- The only preprocessing done was to remove the average Transformation function from each pixel across the entire training set.
- They use kernels of (3 * 3) size with such a stride size of 1 pixel to cover the entire visual concept.
- To keep the image's spatial resolution, spatial padding was applied.
- Sride 2 was used to conduct max pooling across a 2 * 2 pixel window.
- This was followed by the ReLu to bring non-linearity into the model to enhance classification and computing speed, while prior models employed tanh or sigmoid functions, which proved significantly better than those.
- Three fully linked layers were built, the first two of which were 4096 in size, followed by a layer of 1000 units for 1000-way ILSVRC classification, and the last layer is a softmax function.

3.4 Evaluation Metrics

The goal of evaluating a model is to offer a reliable evaluation of its predictive value; nevertheless, performance indicators must account for the specificity of calculations; the most generally used performance indicators are the disparities between actual and prediction function values. The efficacy of both the implemented model and the implemented model is assessed by how well the system predicts the intended label or variable in the test dataset. In this study, the system must determine if a particular news report is true or incorrect. To compare predicted and actual numbers, the confusion matrix is used. Model success may be quantified in terms of Accuracy, Precision, Recall, and F1-score. The ROC Curve is shown to assess the model's efficiency.

3.5 Model Deployment

The process of deploying a finalised machine learning model into a live environment where it may be utilised for its intended purpose is known as machine learning model deployment. For model deployment, we used the streamlit python library to create a user-interactive graphical user interface in which the user would turn on his webcam, the interface would automatically take a snapshot, and the user would then select which machine learning model to use, and the interface would predict the type of emotion the person was experiencing at the time.

CHAPTER 4 – IMPLEMENTATION

Implementing a machine learning algorithm offers a comprehensive grasp of how the algorithm operates. Several micro-decisions must be taken while implementing a machine learning algorithm, and these decisions are typically ignored from theoretical algorithm formulations. This section of the research goes on system design and implementation. The most challenging aspect in this project was developing and implementing an emotion recognition module that could work in real-time. The design has to be broken into two stages in order for them to obtain an accurate result: the Emotion detecting module and the test on a live webcam.

4.1 Workflow

The complete strategy is discussed here, including the full methods used to develop this deep learning model. This assignment has previously been described in detail, and in the next part, we will go through each component of this work step by step.

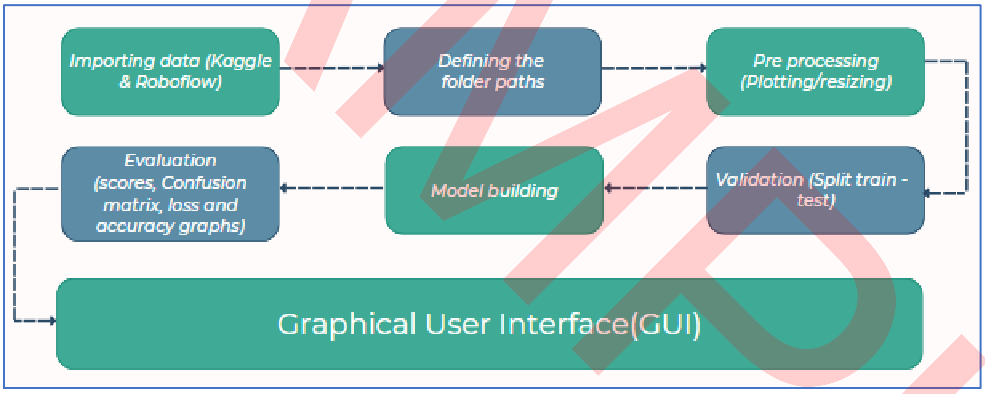


Figure 34: Workflow of Human Emotion Recognition

4.2 Data Collection

Data collecting is one of the most important aspects in doing research. Data collection is a difficult task that requires thorough planning, persistence, and other attributes to be accomplished effectively. The dataset for the models, MobileNetV2, VGG-19 and CNN, was obtained from Kaggle. Importing the libraries and data loading.

4.2.1 FER2013 Database

The Expressions Faces (FER2013) database contains pictures that were developed by categorising training 28709 grayscale samples and testing 7178 48x48 pixel photographs of faces to assess their different facial expressions. Images are classified based on the emotion expressed in their facial expressions (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral).[31]

Figure 35: FER Dataset explanation

- **Defining the folder pathways**:

We determined the locations of our datasets at this stage after determining which datasets to utilise. The dataset is stored on Google Drive. We constructed a base directory that contains the complete dataset's region. Training photographs may be found in the Train directory. Similarly, the photographs used for testing and validation in the given dataset are included in the testing and validation folders. There are seven classes in this dataset. It comes with the necessary code. (Code is available in the appendix A)

Various libraries were used for the system implementation:

- Numpy
- Pandas
- Os
- Matplotlib
- Seaborn
- Skimage

- Cv2
- Glob
- Random
- collections
- Keras
- Tensorflow
- Itertools

4.3 Data Analization

To get a satisfying result, data must be evaluated. Python's easy instructions and syntax make it an effective open-source substitute for established methodology and applications. EDA is a method of summarising data by identifying significant attributes and displaying them using appropriate drawings. EDA was conducted on the dataset once it was collected and stored in the right location. Several graphs were created, and photos were loaded.

Plotting sample images for each class was completed. Since we have a total of seven classes, we plotted five random photos from each class to see if the system is reacting and plotting correctly. (Code is available in the appendix B)

Figure 36: Plotting of images from 7 classes

Various distributions were presented for the train test datasets to assess the existence and number of photos in all classes using the matplotlib library's bar chart.

Furthermore, the number of pictures in the train and test datasets was evaluated, and it was discovered that the training dataset includes 28709 photographs whereas the testing dataset has 7178 photos divided into 7 groups.

- **Imbalanced Dataset labels**

Because we discovered the dataset has an uneven distribution of labels after showing the bar charts of class distribution, class weights were applied to address this problem. We used class weights to balance the labels in order to achieve correct results. (Code is available in the APPENDIX C)

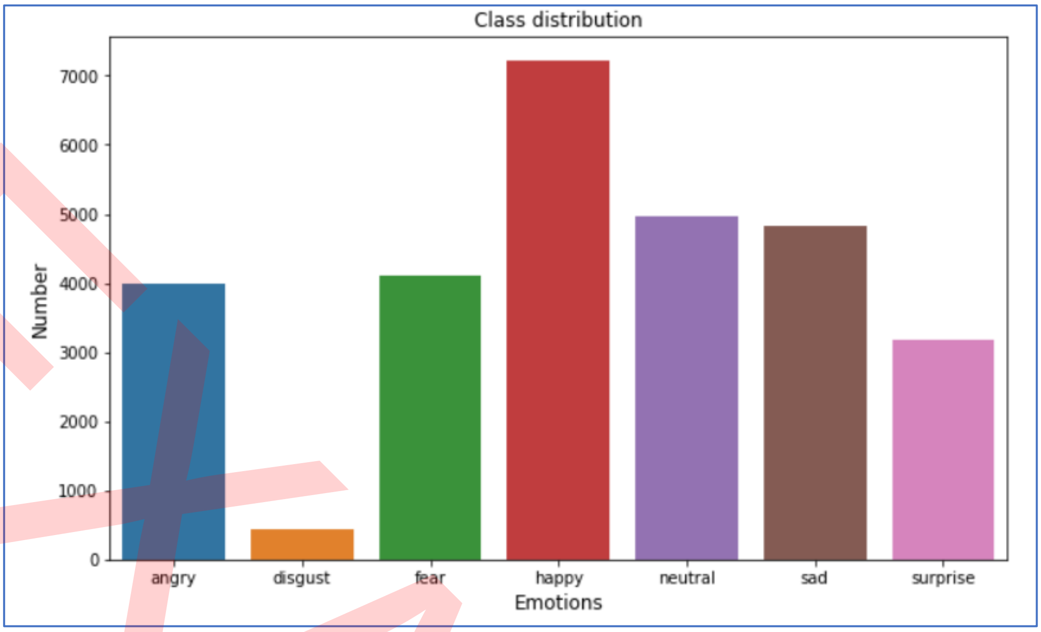


Figure 37: Check imbalanced distribution of data

- **Image size verification:**

We confirmed the overall size of all the photographs in the dataset for further processing during this phase. The typical image size is (48, 48, 3). (Code is available in the APPENDIX D)

- **Data Pre-processing using ImageDataGenerator:**

The information gathered is often disorganised and originates from a variety of sources. Prior to feeding them into the machine learning model, they must be cleaned and structured. Pre-processing is commonly used to minimise the complexity and enhance the precision of a given procedure. Because there isn't a specific method that can be built for every instance in which a picture is obtained, we usually convert it to a format that a generic algorithm can use to solve it when we acquire an image. I used ImageDataGenerator's weights to establish a pipeline in pre-processing to equalise the imbalance classes(Code is available in the APPENDIX E).

The ImageDataGenerator module from the keras.preprocessing package was utilised. This module generates batches of tensor image data while augmenting the data in real time. It utilised to take the original data as inputs and then randomly transform it, providing the final resultant containing just the newly altered data.

4.4 Model building

The Machine Learning development phase includes procedures such as discovering source data, constructing models, deploying them, and maintaining them. Model development and model operations are the two key aspects of the activity [57]. We developed three deep learning models during this phase: 13 Layers CNN, MobileNetV2, and VGG19. We then imported all of the models into the GUI, where the user could choose his or her preferred deep learning model to forecast with. Before we get into the models, let's go through some of the key terms used in developing a deep learning model.

4.4.1 Model Terminologies

Table 3: Used terms for model building

Terminology	**Description**
BatchNormalization()[34]	Supervised learning technique that converts neural network interlayer outputs into a standardised format.
GaussianNoise()	This aids in reducing overfitting. GS is a logical choice for a damage process with real-number inputs.
GlobalAveragePooling2D()[33]	Pooling approach designed to replace completely connected layers in traditional CNNs.
Dropout()[32]	To minimise overfitting, the Dropout layer sets input units to 0 at random at a frequency of rate at each step during training. Non-zero inputs are scaled up by 1/ to preserve the total of all inputs (1 - rate).
Dense()	Used to categorise pictures based on the findings of the convolutional layers.

Flatten()	Used to combine all of the 2-D arrays produced by pooling layer maps into a single long continuous linear vector
Layer weight regularizers	Regularizers allow you to penalise layer parameters or layer activity while optimising. Types: kernel regularizer is used to penalise the layer's kernel, whereas bias regularizer is for layer's bias.
Early stopping()	Regularisation used to prevent overfitting when a learner is being trained iteratively, like with gradient descent.
ReduceLROnPLateau()	Callback function used to improve model learning and to slow learning if the model stops improving.
ModelCheckpoint()	Callback class allows you to specify how, where and when to checkpoint the model weights, name the file,
Freezing Layer()	Model's layer weights do not change when employed on a later downstream mission

- **Activation Functions**

An activation function determines a neuron's activity status. It will identify the extent to which the neuron's input to the network is relevant all across the prediction process by applying simplified mathematical approaches.

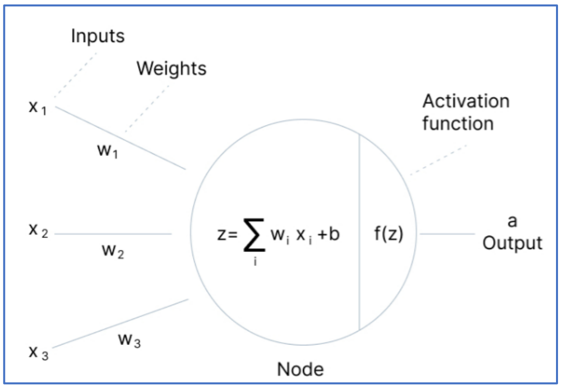


Figure 38: Activation function

I have used 2 activation functions in the deep learning models:

Table 4: Different Activation Functions

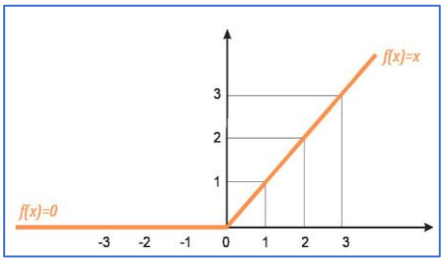
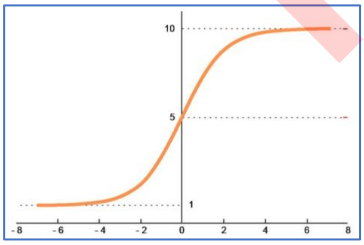
Relu	Softmax
ReLu is a non-linear or piecewise linear function that, if the input is positive, outputs the input directly; if not, it outputs zero.[35] In neural networks, particularly CNNs and multilayer perceptrons, it is the most often employed activation function. Mathematic representation : f(x) = max(0,x) 	With the ability to return the "confidence score" for each class, the function is excellent for classification issues, particularly when faced with multi-class classification issues.[36] The results of the softmax function would add up to 1 because we are dealing with probabilities here. Mathematical representation: $\frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$ 

Figure 39: Relu and Softtmax Activation functions

4.4.2 Different Deep Learning Models

Following a review of deep learning model terms. We constructed three models using various deep learning techniques and their combinations. All of these models received the identical inputs.

4.4.2.1 MobileNetV2

Mobilenetv2 is based on an inverted residual topology, with residual connections connecting the bottleneck levels. Lightweight depth wise convolutions are employed as a source of nonlinearity in the intermediate expansion layer to filter features. With roughly 350 GFLOP, the MobileNet V2[37] model comprises 53 convolution layers and 1 AvgPool. We visualized the model using the plot model module of the tensorflow,keras.utils package as indicated in the figure below.

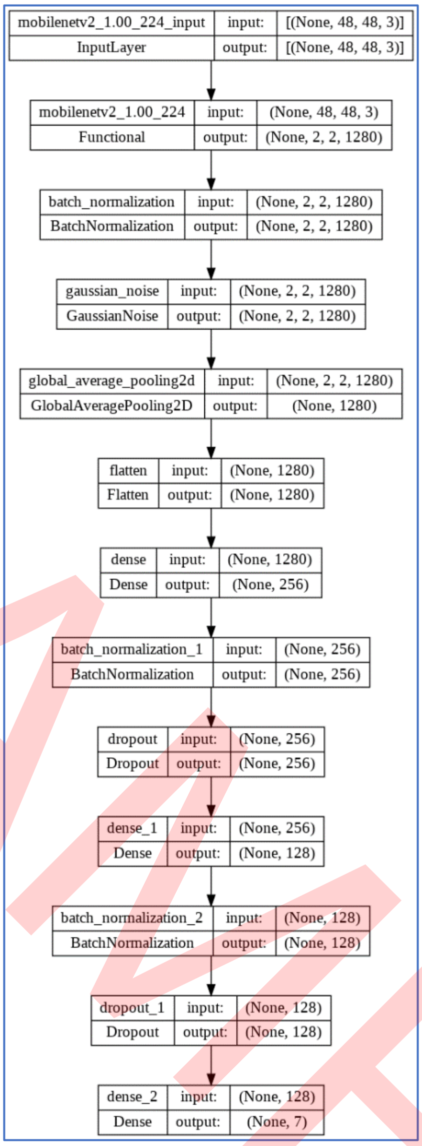


Figure 40: MobileNetV2 model

We built a sequential neural network with 12 layers, 2 dropout layers, and 3 batchnormalization layers using mobilenetv2 as the base model. The total number of params generated by the network was 2,626,375(Code is available in the APPENDIX F).

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 mobilenetv2_1.00_224 (Funct  (None, 2, 2, 1280)       2257984
 ional)

 batch_normalization (BatchN  (None, 2, 2, 1280)       5120
 ormalization)

 gaussian_noise (GaussianNoi  (None, 2, 2, 1280)       0
 se)

 global_average_pooling2d (G  (None, 1280)             0
 lobalAveragePooling2D)

 flatten (Flatten)           (None, 1280)              0

 dense (Dense)               (None, 256)               327936

 batch_normalization_1 (Batc  (None, 256)              1024
 hNormalization)

 dropout (Dropout)           (None, 256)               0

 dense_1 (Dense)             (None, 128)               32896

 batch_normalization_2 (Batc  (None, 128)              512
 hNormalization)

 dropout_1 (Dropout)         (None, 128)               0

 dense_2 (Dense)             (None, 7)                 903

=================================================================
Total params: 2,626,375
Trainable params: 1,086,983
Non-trainable params: 1,539,392
_________________________________________________________________
```

Figure 41: Mobilenetv2 model summary

Table 5: Layers of MobilenetV2 Model

Layer	Description
BactchNormalization	BactchNormalization()
GaussianNoise	Gaussian_noise :0.01
GlobalAveragePooling2D	GlobalAveragePooling2D()
Flatten ()	Flatten ()
Dense	Units: 256, activation='relu', kernel_regularizer=regularizes. l2(0.001), bias_regularizer=regularizers.l2(0.001)
BactchNormalization	BactchNormalization ()
Dropout	Dropout =0.5
Dense	Dense ()
BactchNormalization	BactchNormalization ()
Dropout	Dropout =0.5
Dense	Units: 7, activation="softmax"

4.4.2.2 13 layer CNN Model Specification

CNN[38] is undoubtedly the most well-known family of neural networks when it comes to picture data. The RGB channel of an image is what neural networks employ to process image data.

We visualized the model using the plot model module of the tensorflow,keras.utils package as indicated in the figure below.

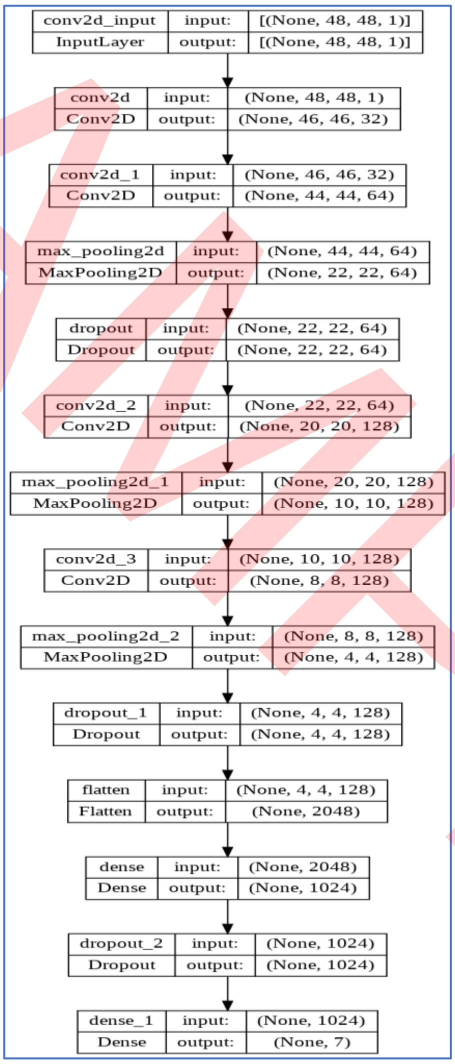


Figure 42: CNN model

In this model, we built a 13-layer basic sequential neural network with three dropout layers, four Conv2D layers, and three MaxPooling2D layers. The total number of params generated by the network was 2,345,607(Code is available in the APPENDIX G).

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 46, 46, 32)        320

 conv2d_1 (Conv2D)           (None, 44, 44, 64)        18496

 max_pooling2d (MaxPooling2D  (None, 22, 22, 64)       0
 )

 dropout (Dropout)           (None, 22, 22, 64)        0

 conv2d_2 (Conv2D)           (None, 20, 20, 128)       73856

 max_pooling2d_1 (MaxPooling  (None, 10, 10, 128)      0
 2D)

 conv2d_3 (Conv2D)           (None, 8, 8, 128)         147584

 max_pooling2d_2 (MaxPooling  (None, 4, 4, 128)        0
 2D)

 dropout_1 (Dropout)         (None, 4, 4, 128)         0

 flatten (Flatten)           (None, 2048)              0

 dense (Dense)               (None, 1024)              2098176

 dropout_2 (Dropout)         (None, 1024)              0

 dense_1 (Dense)             (None, 7)                 7175

=================================================================
Total params: 2,345,607
Trainable params: 2,345,607
Non-trainable params: 0
_________________________________________________________________
```

Figure 43: CNN model summary

Table 6: Layers of CNN Model

Layer	Description
CONV2D	Units: 32, kernel_size= (3, 3), activation='relu', input_shape=(48,48,1)
CONV2D	Units: 64, kernel_size= (3, 3), activation='relu'
MAXPOOLING2D	pool_size= (2, 2)
DROPOUT	Dropout_rate = 0.25
CONV2D	Units: 128, kernel_size= (3, 3), activation='relu'
MAXPOOLING2D	pool_size= (2, 2)
CONV2D	Units: 128, kernel_size= (3, 3), activation='relu'
MAXPOOLING2D	pool_size= (2, 2)
DROPOUT	Dropout_rate = 0.25
FLATTEN	FLattten ()
DENSE	Units: 1024, activation='relu'
DROPOUT	Dropout_rate = 0.25
DENSE	Units: 7, activation='softmax'

4.4.2.3 VGG19 Model

VGG19 is a powerful CNN with pre-trained layers and a strong grasp of what constitutes an image in terms of form, colour, and structure. We visualized the model using the plot model module of the tensorflow,keras.utils package as indicated in the figure below.

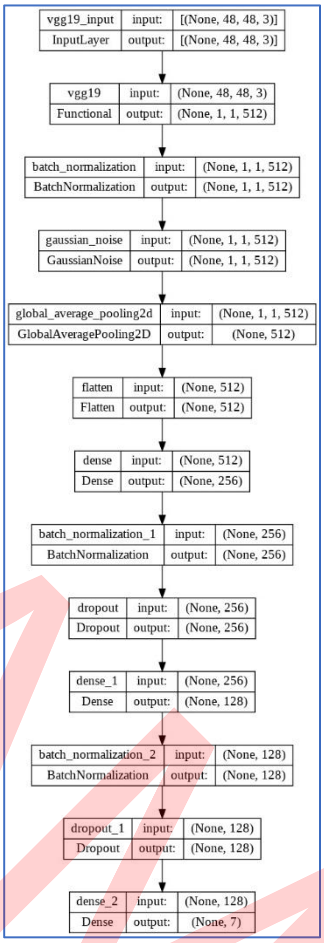


Figure 44: VGG19 Model

In this model, we built a 12-layer basic sequential neural network with base model as VGG19 and two dropout layers, three BatchNormalization layers. The total number of params generated by the network was 20,193,095(Code is available in the APPENDIX H).

```
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 vgg19 (Functional)          (None, 1, 1, 512)         20024384

 batch_normalization (BatchN  (None, 1, 1, 512)        2048
 ormalization)

 gaussian_noise (GaussianNoi  (None, 1, 1, 512)        0
 se)

 global_average_pooling2d (G  (None, 512)              0
 lobalAveragePooling2D)

 flatten (Flatten)           (None, 512)               0

 dense (Dense)               (None, 256)               131328

 batch_normalization_1 (Batc  (None, 256)              1024
 hNormalization)

 dropout (Dropout)           (None, 256)               0

 dense_1 (Dense)             (None, 128)               32896

 batch_normalization_2 (Batc  (None, 128)              512
 hNormalization)

 dropout_1 (Dropout)         (None, 128)               0

 dense_2 (Dense)             (None, 7)                 903

=================================================================
Total params: 20,193,095
Trainable params: 11,965,959
Non-trainable params: 8,227,136
_________________________________________________________________
```

Figure 45:VGG19 Model summary

Table 7: Layers of VGG-19 Model

Layer	Description
BactchNormalization	BactchNormalization()
GaussianNoise	Gaussian_noise :0.01
GlobalAveragePooling2D	GlobalAveragePooling2D()
Flatten ()	Flatten ()
Dense	Units: 256, activation='relu', kernel_regularizer=regularizes. l2(0.001), bias_regularizer=regularizers.l2(0.001)
BactchNormalization	BactchNormalization ()
Dropout	Dropout =0.5
Dense	Dense ()
BactchNormalization	BactchNormalization ()
Dropout	Dropout =0.5
Dense	Units: 7, activation="softmax"

4.5 Evaluation

When addressing predictive model stability, the efficacy of a model must be calculated, applying a wide range of performance criteria to achieve a highly accurate model. The assessment metrics used to assess the effectiveness of the models utilised are listed below.

Confusion Matrix: The confusion matrix is a 2 x 2 matrix that indicates the model's performance in terms of true positive, true negative, false positive, and false negative.

True Positive: When a trained model predicts that the genuine news is true. In those other words, it predicted what was expected.

False Positive: A false positive occurs when a trained machine incorrectly predicts phoney news as true. In other words, the model predicted incorrectly.

True Negative: When a trained machine predicts that phoney news is fake. In other words, it predicted what was expected.

False Negative: When a trained machine incorrectly forecasts true news as phoney. In other words, the model predicted incorrectly.

Accuracy: The fraction of news discovered properly (false or true) over the entire number of news is defined as accuracy.

$$\boxed{\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{False Positive} + \text{True Negative} + \text{False Negative}}} \quad (1)$$

Precision : Precision indicates the percentage of news that is accurately forecasted as true news.

$$\boxed{\text{Precision} = \frac{\textit{True Positive}}{\text{True Positive} + \text{False Positive}}} \quad (2)$$

Recall : Recall is defined as the amount of news recognised as true despite the fact that part of the news is false.

$$\boxed{\text{Recall} = \frac{\textit{True Positive}}{\text{True Positive} + \text{False Negative}}} \quad (3)$$

F1-score: The F1-score is an able to categorize of model accuracy. It's defined as the harmonic mean of Recall and Precision.

$$\boxed{\text{F1 Score} = 2 * \frac{Recall * Precision}{Recall + Precision}} \quad (4)$$

Receiver Operator Characteristic Curve : Characteristics of a ROC curve is a visual diagram used in binary classifiers to demonstrate diagnostic capabilities. The ROC curve is created by graphing the True Positive Rate in the y-axis and the False Positive Rates in the x-axis at various threshold values.

The **True Positive rate** is defined as the percentage of news that the model properly predicts and classifies as Real news.

$$\boxed{True\ Positive\ Rate = \frac{True\ Positive}{True\ Positive + False\ Negative}} \quad (5)$$

The **False Positive Rate** is defined as the portion of news that the model wrongly predicts and classifies as Real news.

$$\boxed{False\ Positive\ Rate = \frac{False\ Positive}{False\ Positive + True\ Negative}} \quad (6)$$

4.6 Graphical User Interface

We designed a user-interactive GUI. The GUI is built using the Python module streamlit. There are three types or displays in the interface. The first screen is the home screen, from which the user may choose a screen using the dropdown menu on the left. The three displays are home, Image Face Detection, and Webcam Face Detection.

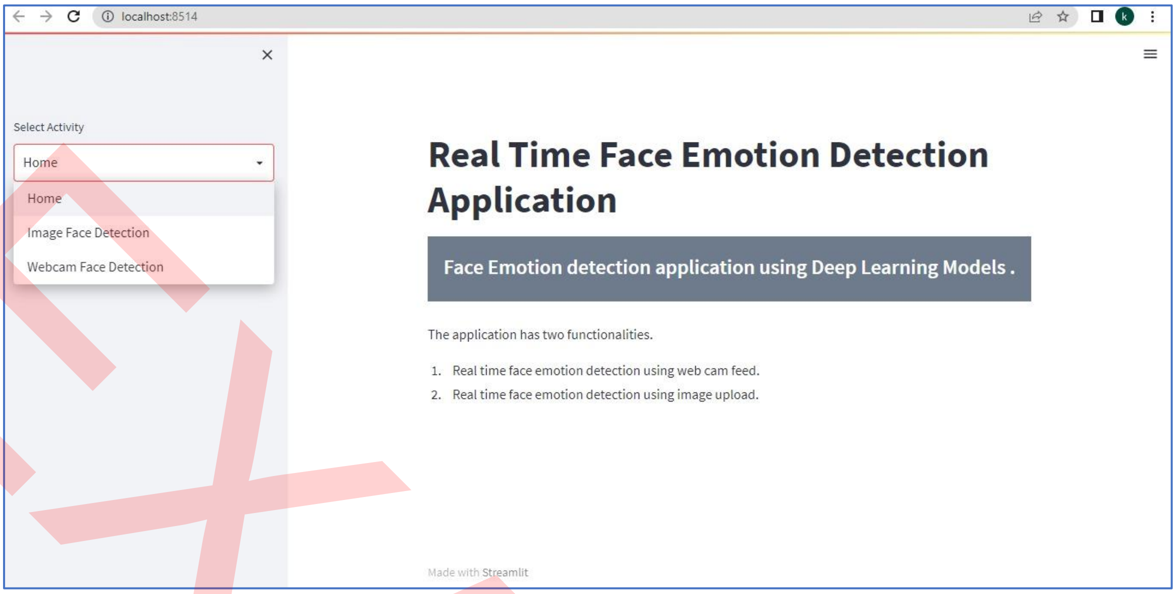


Figure 46: GUI Home Page

In the Image Face Detection page, the user would submit the image they want to know the emotion represented by, the uploaded image after pre-processing would be sent to the model, the model would then determine based on its learning and report the findings on the screen. While on the other screen of web cam face detection, the user enables the system to record the webcam, where the webcam video is played using javascript, the video is sent through the model, and the model recognises and displays the emotions.

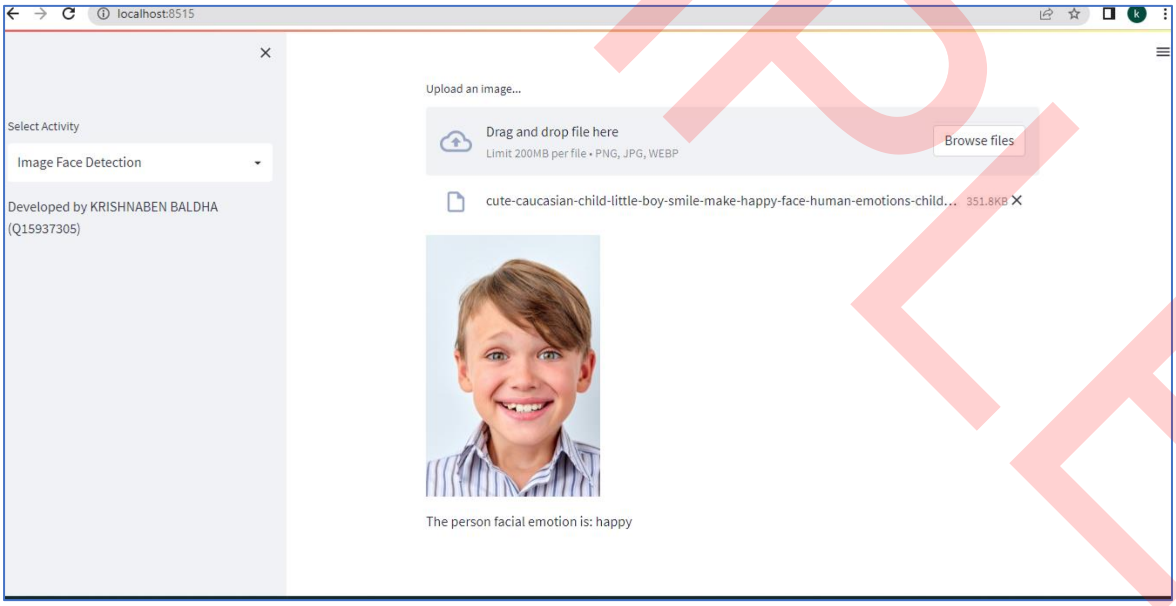


Figure 47: GUI for Image Face Detection

CHAPTER 5 – RESULTS

This section follows the previous chapter in delivering the results. The key conclusions of the study, which were founded on the research's implementation section, are summarised in this section.

With the Three Models constructed, several trainings are carried out. Each workout started with train, test, split.

5.1 Model1: MobileNetV2

The accuracy at various iterations and epochs was used to calculate performance on MobileNetV2 using the monitored epoch for each validation. The table displays the training's period accuracy.

Table 8: Number of epochs for Mobilenetv2

Model Name	Epochs	Accuracy
MobileNetV2	1	24%
	25	51%
	50	85%

Loss Vs Accuracy Graph :

Accuracy in Mobilenetv2 mode is seen in the graph below:

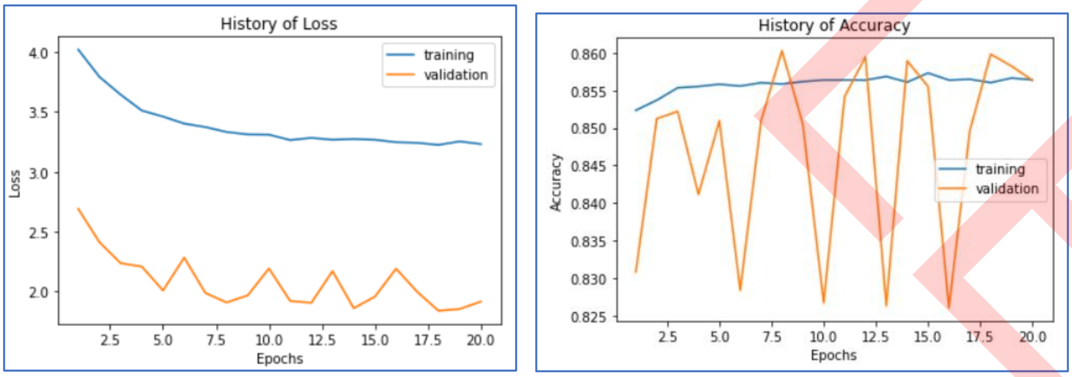


Figure 48: History of loss and accuracy for mobilenetv2

Model Accuracy

- The training loss is shown by the blue line, while the testing/validation loss is represented by the orange line.
- The number of epochs is shown on the x-axis, while the model loss is shown on the y-axis.
- It has been noted that the training loss began at 0.852 and finally came to a conclusion at 0.858 with some ups and downs.
- It has been noted that the testing/validation loss began at 0.831 and ended with some ups and downs at 0.857 after the training loss.
- The y-axis interval is 0.05 while the x-axis interval is 5.

Model Loss

- The training accuracy is shown by the blue line, while the testing/validation loss is represented by the orange line.
- The number of epochs is shown on the x-axis, while the model loss is shown on the y-axis.
- It has been noted that the training loss began at 4.1 and finally came to a conclusion at 3.2 with some ups and downs.
- It has been noted that the testing/validation accuracy began at 2.8 and ended with some ups and downs at 1.9 after the training loss.
- The y-axis interval is 0.05 while the x-axis interval is 5.

Confusion Matrix of MobilenetV2

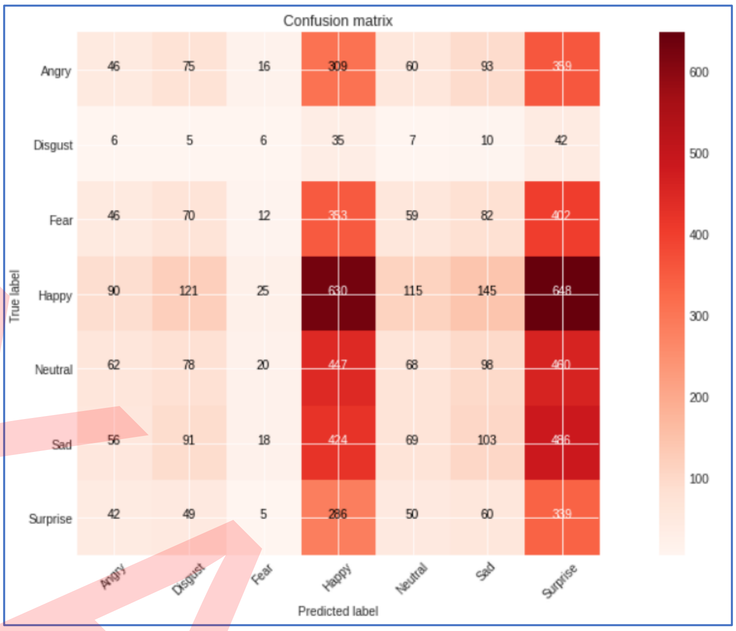


Figure 49: Confusion matrix for mobilenetv2

Compared to angry, disgust and surprise, Model 1 shows superior memory for the emotions of Fear , Happy, Neutral and Sad.

• It frequently (28%) mislabels happy, which might lead to confusion among humans.

• It also occasionally (17%) referred to fear as sadness, which may cause misunderstanding because people tend to feel sad when they are afraid.

• Because it links a good feeling to a bad emotion, it does view anger as calm (7%), fear as happy (10%), happy as fear (7%) and sad (7%) which is unusual.

5.2 Model2: 13 Layer CNN Model

Performance was calculated based on accuracy at different iterations and epochs using each validation's monitored epoch on CNN model. The training's accuracy with epochs are shown in the table.

Table 9: Number of epochs for CNN Model

Model Name	Epochs	Accuracy
CNN Model	1	26%
	50	72%
	100	93%

Loss Vs Accuracy Graph :

Accuracy in CNN mode is seen in the graph below:

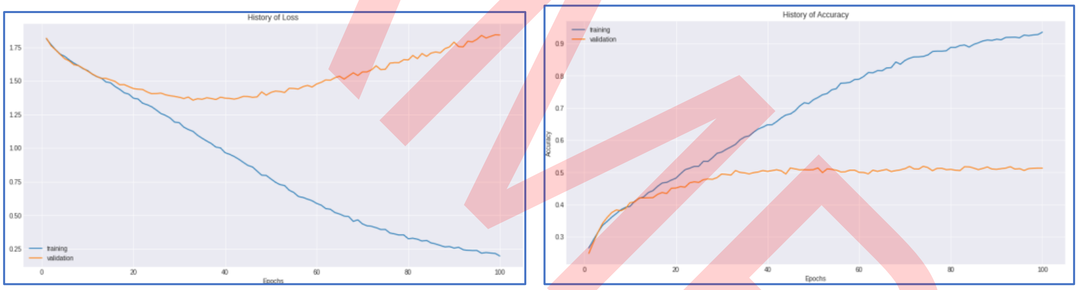


Figure 50: History of loss & accuracy for CNN

Model Loss

- The training loss is shown by the blue line, while the testing/validation loss is represented by the orange line.
- The number of epochs is shown on the x-axis, while the model loss is shown on the y-axis.
- It has been noted that the training loss began at 1.76 and finally came to a conclusion at 0.22 with some ups and downs.
- It has been noted that the testing/validation loss began at 1.77 and ended with some ups and downs at 1.79 after the training loss.
- The y-axis interval is 0.05 while the x-axis interval is 20.

Model Accuracy

- The training accuracy is shown by the blue line, while the testing/validation loss is represented by the orange line.
- The number of epochs is shown on the x-axis, while the model loss is shown on the y-axis.
- It has been noted that the training accuracy began at 0.2 and finally came to a conclusion at 0.11 with some ups and downs.
- It has been noted that the testing/validation accuracy began at 0.1 and ended with some ups and downs at 0.6 after the training loss.
- The y-axis interval is 0.1 while the x-axis interval is 5.

Confusion Matrix for CNN :

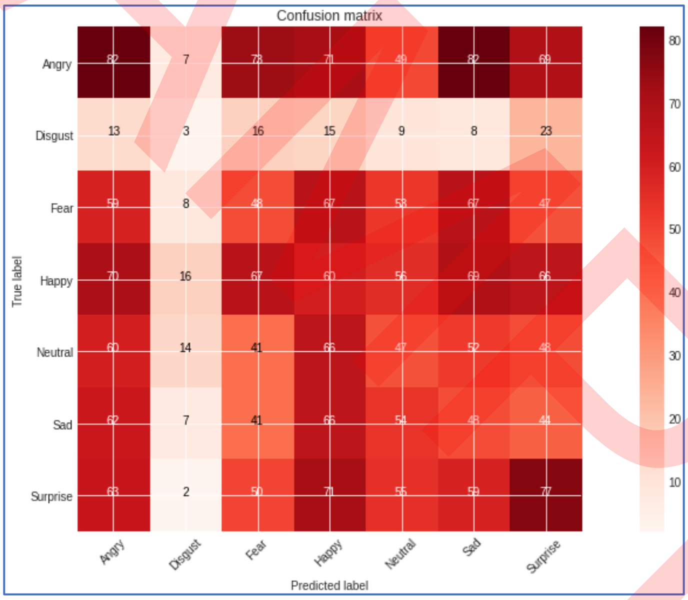


Figure 51: Confusion matrix for CNN

Compared to sad and surprise, Model 1 shows superior memory for the emotions of Angry, Fear , Happy and Neutral.

• It frequently (70%) mislabels happy, which might lead to confusion among humans.

• It also occasionally (59%) referred to fear as sadness, which may cause misunderstanding because people tend to feel sad when they are afraid.

• Because it links a good feeling to a bad emotion, it does view anger as calm (7%), fear as happy (10%), happy as fear (7%) and sad (7%) which is unusual.

5.3 Model3: VGG19

Performance was calculated based on accuracy at different iterations and epochs using each validation's monitored epoch on VGG19 model. The training's accuracy with epochs are shown in the table.

Table 10: Number of epochs for VGG19

Model Name	Epochs	Accuracy
VGG19	15	86%
	25	86%
	50	87.58%

Loss Vs Accuracy Graph:

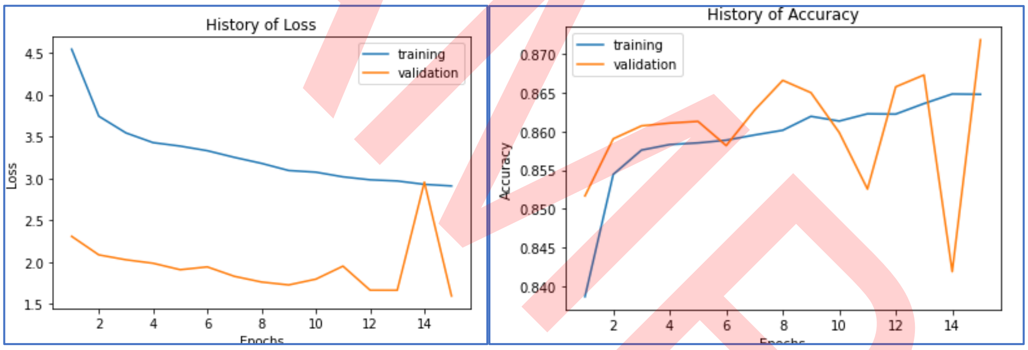


Figure 52: History of Loss & Accuracy for VGG19

Model Loss

- The training loss is shown by the blue line, while the testing/validation loss is represented by the orange line.
- The number of epochs is shown on the x-axis, while the model loss is shown on the y-axis.
- It has been noted that the training loss began at 4.63 and finally came to a conclusion at 2.921 with some ups and downs.
- It has been noted that the testing/validation loss began at 2.32 and ended with some ups and downs at 1.532 after the training loss.
- The y-axis interval is 0.05 while the x-axis interval is 2.

Model Accuracy

- The training accuracy is shown by the blue line, while the testing/validation loss is represented by the orange line.
- The number of epochs is shown on the x-axis, while the model loss is shown on the y-axis.
- It has been noted that the training accuracy began at 0.838 and finally came to a conclusion at 0.863 with some ups and downs.
- It has been noted that the testing/validation accuracy began at 0.852 and ended with some ups and downs at 0.873 after the training accuracy.
- The y-axis interval is 0.005 while the x-axis interval is 2.

Confusion Matrix of VGG-19 :

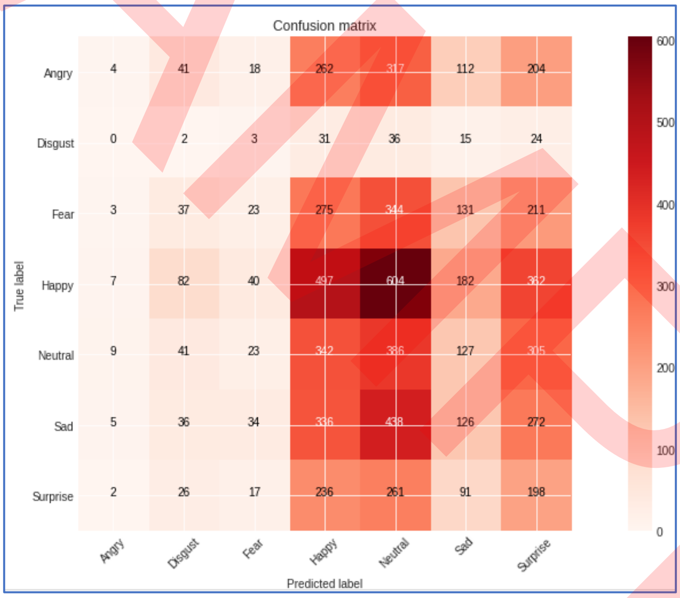


Figure 53: Confusion matrix for VGG19

Compared to Fear, angry and disgust, Model 3 shows superior memory for the emotions of Surprise , Happy, Neutral and Sad.

• It frequently (49%) mislabels happy, which might lead to confusion among humans.

• It also occasionally (27%) referred to fear as sadness, which may cause misunderstanding because people tend to feel sad when they are afraid.

5.4 Emotion Detection with webcam

I use my webcam to check for an image that contains emotion, and I display the results in a bar graph with emotion. I can check how well emotions are recognised with this.

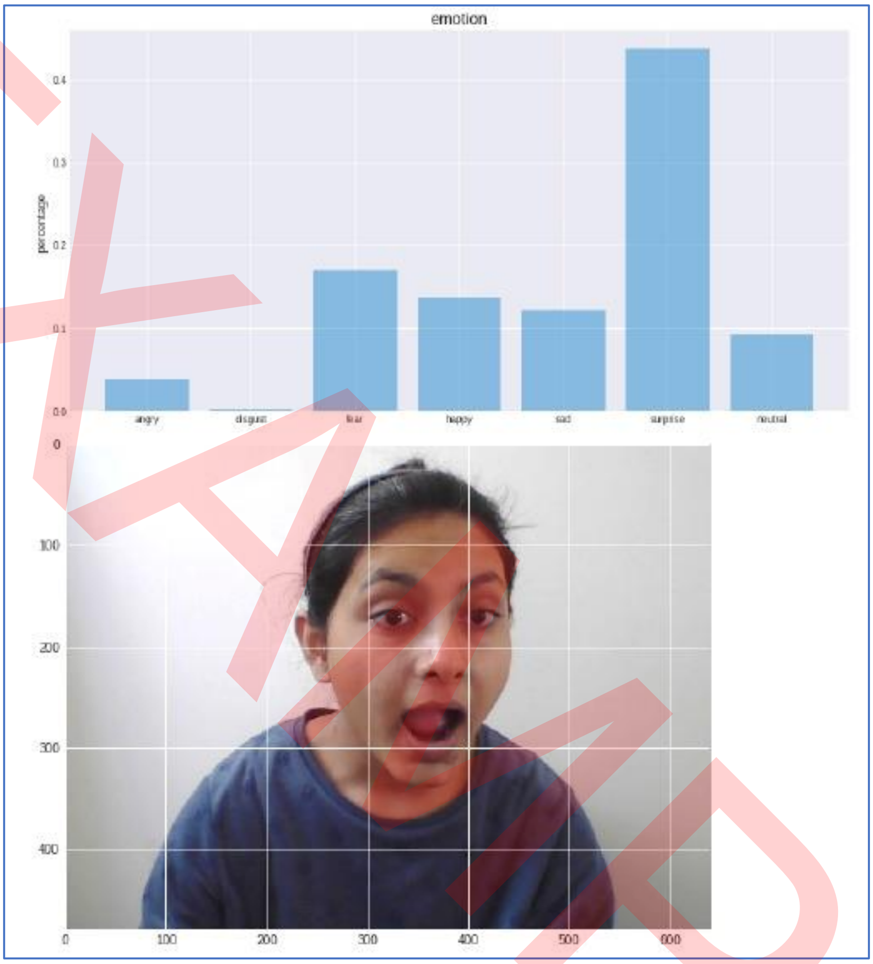


Figure 54: Emotions detection using webcam

As per the graph we can see that the model has predicted that the emotion would be surprise , as the bar of surprise is the highest in the bar chart. Moreover when you re-run it again , the code will open the webcam and take image after which it will be passed through model and graph is displayed with the emotion detected.

CHAPTER 6 – DISCUSSIONS

Three models, MobileNetV2, VGG19, and CNN, were run with varied epochs and their accuracy was measured; we discovered that having more epochs boosted the models' accuracy. The maximum accuracy of MobileNetV2 was 85% with 50 epochs, 93% with 100 epochs for CNN, and 87.58% with 50 epochs for VGG19. The CNN model outperformed the other models in terms of epochs accuracy, with the maximum accuracy of 93%. The orange line in the graphs in the results section shows validation, while the blue line represents training. The number of epochs is indicated on the x-axis, while accuracy/loss is shown on the y-axis. The loss history plot for models reveals that as the epochs rose, the loss dropped for training, whereas there were repeated ups and downs for validation. While for the accuracy graph, as epochs grew, the accuracy improved for training, and there were repeated ups and downs for validation. In terms of confusion matrices for the models, MobileNetV2 was able to predict two kinds of people: joyful and surprised. Concerning the CNN model, it was capable of predicting all classes but was unable to predict distaste. While the VGG19 model can accurately predict happiness, neutrality, and surprise. According to the confusion matrix, the CNN model correctly predicted the real labels of the classes. Thus, when the graphs, accuracy, and evaluation matrix are considered, the CNN model outperformed the others.

6.1 LIMITATIONS AND FUTURE SCOPE

The constraint of this approach is that the large amount of image data sets necessitates a high configuration system. If you do not have enough resources to run the code, it may collapse. Extensive topic expertise is required to finish the job. I faced execution troubles due to limited cloud resources (memory, CPU). The downsides of using facial expressions to assess emotions include the fact that most facial expression working frameworks rely on the FACS system, which has typically been utilized to classify only the six basic emotions, and are extremely labour-intensive if performed by trained human programmers rather than software. Furthermore, I attempted emotion detection using Yolo, but there were several issues linked to code execution that I was unable to discover.

However, there remains space for learning and improvement and because the scope was constrained, there is plethora of future possibility for this research, such as further research investigating the effect of new independent factors, testing additional external factors, and obtaining more data. In the future, the work can be enhanced by working with video datasets to obtain more precise findings. The number of emotions included may potentially be extended, and the model can be generalised for other languages and accents. Transfer learning and reinforcement learning are further alternatives that might be investigated and utilised. Eventually, I want to create a self-contained Emotion Recognition system for real-world applications. By adjusting the hyperparameters, the models' performance may be significantly improved, outperforming competing models. Because of its effective use of computer resources and recent improvements in intelligent control processing units, it may be employed in a wide range of applications.

6.2 CONCLUSION

After extensive research and experimentation, 3 deep learning models using 13 layer CNN model, MobileNetV2, yolov5 are built, trained, and evaluated on FER2013 dataset. Before model designing, the data pre-processing methods are explored and implemented on the dataset. The accuracy and confusion matrices are used to evaluate and compare each model. The final model's accuracy 80.1% on a holdout test dataset and was able to predict 7 out of 10 real data correctly. It also has a 73% to 91% F1-score for all five emotions considered. CNNs driven by TL will be used for image classification, keeping in mind the nature of the training data (pictures).

6.3 PROJECT TIMELINE

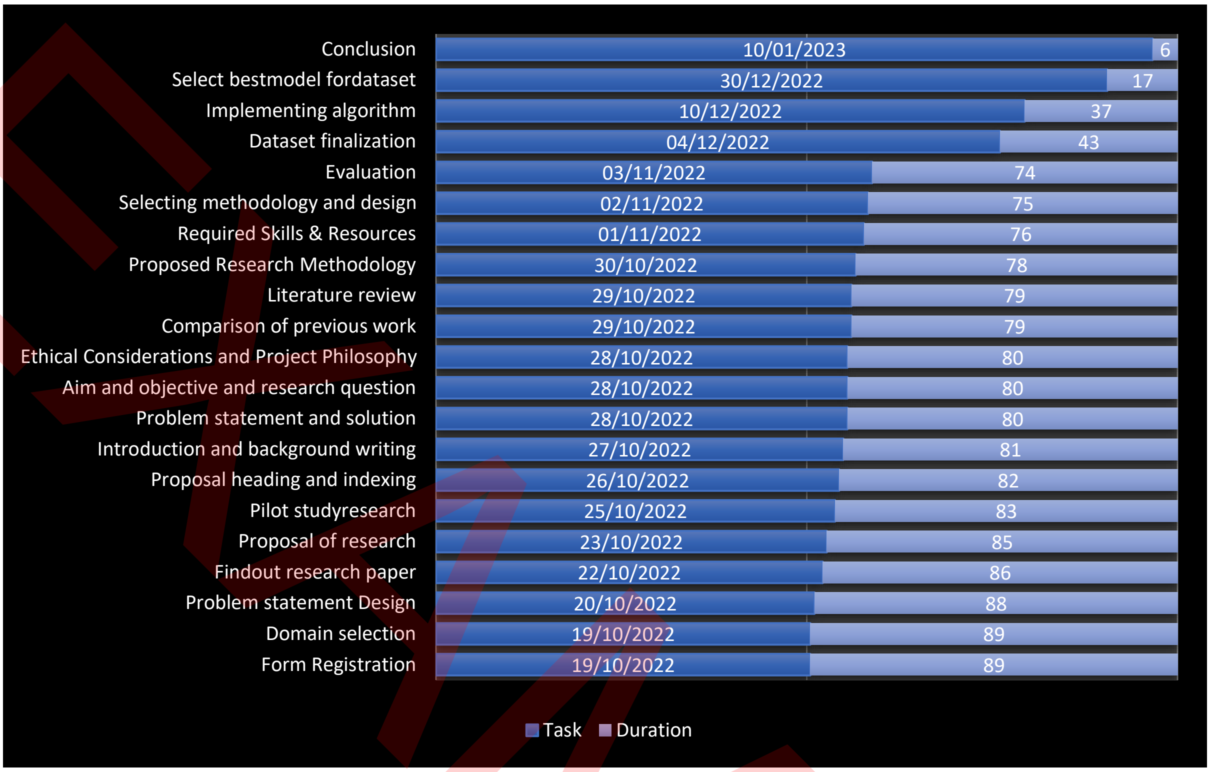


Figure 55: Project Timeline

REFERENCES

1. Shakespeare, W. (2010). Macbeth. Boston, MA: English Play Press. (Original work published 1699)
2. Gamboa Valero, H. (n.d.). "Automatic Facial Expression Recognition. [online] Available at: https://studentnet.cs.manchester.ac.uk/resources/library/thesis_abstracts/MSc16/FullText/GamboaValero-Hugo-diss.pdf [Accessed 25 Nov. 2022].
3. Kuhn, L. (2014). EMOTION RECOGNITION IN THE HUMAN FACE AND VOICE. [online] Available at: https://bura.brunel.ac.uk/bitstream/2438/11216/1/FulltextThesis.pdf.
4. Fletcher, N. (n.d.). *Classification vs. Detection vs. Segmentation Models*. [online] www.clarifai.com. Available at: https://www.clarifai.com/blog/classification-vs-detection-vs-segmentation-models-the-differences-between-them-and-how-each-impact-your-results#:~:text=Segmentation%20models%20provide%20the%20exact.
5. Deng, W., Hu, J., Zhang, S. and Guo, J., 2015, December. DeepEmo: Real-world facial expression analysis via deep learning. In 2015 Visual Communications and Image Processing (VCIP) (pp. 1-4). IEEE.
6. Jung, H., Lee, S., Park, S., Kim, B., Kim, J., Lee, I. and Ahn, C., 2015, January. Development of deep learning-based facial expression recognition system. In 2015 21st Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV) (pp. 1-4). IEEE.
7. Lee, H.J. and Hong, K.S., 2017, October. A study on emotion recognition method and its application using face image. In 2017 International Conference on Information and Communication Technology Convergence (ICTC) (pp. 370-372). IEEE.
8. Fathallah, A., Abdi, L. and Douik, A., 2017, October. Facial expression recognition via deep learning. In 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA) (pp. 745-750). IEEE.
9. Yang, B., Cao, J., Ni, R. and Zhang, Y., 2017. Facial expression recognition using weighted mixture deep neural network based on double-channel facial images. IEEE access, 6, pp.4630-4640.
10. Sharma, J.J.S. and Dugar, Y., 2018. Detection and recognition of human emotion using neural network. Int J Appl Eng Res, 13(8), pp.6472-6477.
11. Ramdhani, B., Djamal, E.C. and Ilyas, R., 2018, August. Convolutional neural networks models for facial expression recognition. In 2018 International Symposium on Advanced Intelligent Informatics (SAIN) (pp. 96-101). IEEE.
12. Jin, X. and Xu, Y., 2019, October. Research on Facial Expression Recognition Based on Deep Learning. In 2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE) (pp. 1144-1147). IEEE.
13. Refat, C.M.M. and Azlan, N.Z., 2019, October. Deep learning methods for facial expression recognition. In 2019 7th International Conference on Mechatronics Engineering (ICOM) (pp. 1-6). IEEE.
14. Kondaveeti, H.K. and Goud, M.V., 2020, December. Emotion detection using deep facial features. In 2020 IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation (ICATMRI) (pp. 1-8). IEEE.
15. Lalitha, S.K., Aishwarya, J., Shivakumar, N., Srilekha, T. and Kartheek, G.C.R., 2021, August. A Deep Learning Model for Face Expression Detection. In 2021 International

Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT) (pp. 647-650). IEEE.

16. Hans, A.S.A., Bansal, M. and Rao, S., 2021, July. An Advanced Facial Expression Detection using Deep Neural Network. In 2021 8th International Conference on Smart Computing and Communications (ICSCC) (pp. 204-208). IEEE.
17. Verma, V. and Rani, R., 2021, August. Recognition Of Facial Expressions Using A Deep Neural Network. In 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 585-590). IEEE.
18. Hardjadinata, H., Oetama, R.S. and Prasetiawan, I., 2021, October. Facial Expression Recognition Using Xception And DenseNet Architecture. In 2021 6th International Conference on New Media Studies (CONMEDIA) (pp. 60-65). IEEE.
19. Lokku, G., Reddy, G.H. and Prasad, M.G., 2021, June. A Robust Face Recognition model using Deep Transfer Metric Learning built on AlexNet Convolutional Neural Network. In 2021 International Conference on Communication, Control and Information Sciences (ICCISc) (Vol. 1, pp. 1-6). IEEE.
20. Sekaran, S.A.R., Lee, C.P. and Lim, K.M., 2021, August. Facial emotion recognition using transfer learning of AlexNet. In 2021 9th International Conference on Information and Communication Technology (ICoICT) (pp. 170-174). IEEE.
21. Nan, Y., Ju, J., Hua, Q., Zhang, H. and Wang, B., 2022. A-MobileNet: An approach of facial expression recognition. Alexandria Engineering Journal, 61(6), pp.4435-4444.
22. Sun, L., Ge, C. and Zhong, Y., 2021, September. Design and Implementation of Face Emotion Recognition System Based on CNN Mini_Xception Frameworks. In Journal of Physics: Conference Series (Vol. 2010, No. 1, p. 012123). IOP Publishing.
23. He, B., 2022, May. Deep learning algorithm for Facial Expression Classification. In 2022 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL & ICCEA) (pp. 386-391). IEEE.
24. Phattarasooksirot, P. and Sento, A., 2022, May. Facial Emotional Expression Recognition Using Hybrid Deep Learning Algorithm. In 2022 7th International Conference on Business and Industrial Research (ICBIR) (pp. 323-329). IEEE.
25. Jayasimha, Y., Venkatesha, M. and Reddy, R.V.S., 2022, May. Face Detection and its Features Extraction using Convolution Neural Network Model. In 2022 3rd International Conference for Emerging Technology (INCET) (pp. 1-6). IEEE.
26. Sadikoğlu, F.M. and Mohamed, M.I., 2022, August. Facial Expression Recognition Using CNN. In 2022 International Conference on Artificial Intelligence in Everything (AIE) (pp. 95-99). IEEE.
27. Ramu, T. and Muthukumar, A., 2022, January. A GoogleNet architecture based Facial emotions recognition using EEG data for future applications. In 2022 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-7). IEEE.
28. OpenGenus IQ: Computing Expertise & Legacy. (n.d.). *MobileNetV2 architecture*. [online] Available at: https://iq.opengenus.org/mobilenetv2-architecture/.
29. www.sciencedirect.com. (n.d.). *Pooling Layer - an overview | ScienceDirect Topics*. [online] Available at: https://www.sciencedirect.com/topics/mathematics/pooling-layer.
30. Sec, D.I. (2021). *VGG-19 Convolutional Neural Network*. [online] All about Machine Learning. Available at: https://freenext.com/vgg-19-convolutional-neural-network/ [Accessed 18 Jan. 2023].

31. www.kaggle.com. (n.d.). *FER-2013*. [online] Available at: https://www.kaggle.com/datasets/msambare/fer2013.
32. Team, K. (n.d.). *Keras documentation: Dropout layer*. [online] keras.io. Available at: https://keras.io/api/layers/regularization_layers/dropout/#:~:text=The%20Dropout%20layer%20randomly%20sets.
33. paperswithcode.com. (n.d.). *Papers with Code - Global Average Pooling Explained*. [online] Available at: https://paperswithcode.com/method/global-average-pooling.
34. DeepAI. (2019). *Batch Normalization*. [online] Available at: https://deepai.org/machine-learning-glossary-and-terms/batch-normalization#:~:text=Batch%20Normalization%20is%20a%20supervised%20learning%20technique%20that.
35. OpenGenus IQ: Computing Expertise & Legacy. (2021). *ReLU (Rectified Linear Unit) Activation Function*. [online] Available at: https://iq.opengenus.org/relu-activation/.
36. Radečić, D. (2020). *Softmax Activation Function Explained*. [online] Medium. Available at: https://towardsdatascience.com/softmax-activation-function-explained-a7e1bc3ad60.
37. paperswithcode.com. (n.d.). *Papers with Code - MobileNetV2 Explained*. [online] Available at: https://paperswithcode.com/method/mobilenetv2.
38. Uzila, A. (2022). *5 Popular CNN Architectures Clearly Explained and Visualized*. [online] Medium. Available at: https://towardsdatascience.com/5-most-well-known-cnn-architectures-visualized-af76f1f0065e.
39. Cybiant (n.d.). An Introduction to Computer Vision. [online] Cybiant. Available at: https://www.cybiant.com/resources/an-introduction-to-computer-vision/ [Accessed 25 Nov. 2022].
40. OpenCV. (n.d.). Home. [online] Available at: http://opencv.org/ [Accessed 25 Nov. 2022].
41. Tversky, A., & Kahneman, D. (1981). The framing of decisions and the psychology of choice. *Science*, *211*(4481), 453–458. https://doi.org/10.1126/science.7455683
42. Card, S.K. (Ed.). (1983). The Psychology of Human-Computer Interaction (1st ed.). CRC Press. https://doi.org/10.1201/9780203736166
43. P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar and I. Matthews, "The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression," *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, 2010, pp. 94-101, doi: 10.1109/CVPRW.2010.5543262.
44. www.v7labs.com. (n.d.). A Newbie-Friendly Guide to Transfer Learning: Everything You Need to Know. [online] Available at: https://www.v7labs.com/blog/transfer-learning-guide.
45. Li, S. and Deng, W. (2020). Deep Facial Expression Recognition: A Survey. IEEE Transactions on Affective Computing, pp.1–1. doi:10.1109/taffc.2020.2981446.
46. Song, Z. (2021). Facial Expression Emotion Recognition Model Integrating Philosophy and Machine Learning Theory. Frontiers in Psychology, 12. doi:10.3389/fpsyg.2021.759485.

10.1 Figure References

1. Maxwell. (n.d.). Emotional Intelligence. [online] Available at: https://www.maxwell.cz/training-solutions/personal-effectiveness/emotional-intelligence/ [Accessed 25 Nov. 2022].
2. Kumar, A. (2019). *Facial Expression Recognition*. [online] Medium. Available at: https://medium.com/@abhayakumar169/facial-expression-recognition-cf6a61f8045f.
3. Cybiant (n.d.). *An Introduction to Computer Vision*. [online] Cybiant. Available at: https://www.cybiant.com/resources/an-introduction-to-computer-vision/.
4. Scott, A. (2022). AI, ML, or DL – learn what it means. [online] Data Science Central. Available at: https://www.datasciencecentral.com/ai-ml-or-dl-learn-what-it-means/ [Accessed 25 Nov. 2022].
5. Deng, W., Hu, J., Zhang, S. and Guo, J., 2015, December. DeepEmo: Real-world facial expression analysis via deep learning. In 2015 Visual Communications and Image Processing (VCIP) (pp. 1-4). IEEE.
6. Jung, H., Lee, S., Park, S., Kim, B., Kim, J., Lee, I. and Ahn, C., 2015, January. Development of deep learning-based facial expression recognition system. In 2015 21st Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV) (pp. 1-4). IEEE.
7. Lee, H.J. and Hong, K.S., 2017, October. A study on emotion recognition method and its application using face image. In 2017 International Conference on Information and Communication Technology Convergence (ICTC) (pp. 370-372). IEEE.
8. Fathallah, A., Abdi, L. and Douik, A., 2017, October. Facial expression recognition via deep learning. In 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA) (pp. 745-750). IEEE.
9. Yang, B., Cao, J., Ni, R. and Zhang, Y., 2017. Facial expression recognition using weighted mixture deep neural network based on double-channel facial images. IEEE access, 6, pp.4630-4640.
10. Sharma, J.J.S. and Dugar, Y., 2018. Detection and recognition of human emotion using neural network. Int J Appl Eng Res, 13(8), pp.6472-6477.
11. Ramdhani, B., Djamal, E.C. and Ilyas, R., 2018, August. Convolutional neural networks models for facial expression recognition. In 2018 International Symposium on Advanced Intelligent Informatics (SAIN) (pp. 96-101). IEEE.
12. Jin, X. and Xu, Y., 2019, October. Research on Facial Expression Recognition Based on Deep Learning. In 2019 3rd International Conference on Electronic Information Technology and Computer Engineering (EITCE) (pp. 1144-1147). IEEE.
13. Refat, C.M.M. and Azlan, N.Z., 2019, October. Deep learning methods for facial expression recognition. In 2019 7th International Conference on Mechatronics Engineering (ICOM) (pp. 1-6). IEEE.
14. Kondaveeti, H.K. and Goud, M.V., 2020, December. Emotion detection using deep facial features. In 2020 IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation (ICATMRI) (pp. 1-8). IEEE.
15. Lalitha, S.K., Aishwarya, J., Shivakumar, N., Srilekha, T. and Kartheek, G.C.R., 2021, August. A Deep Learning Model for Face Expression Detection. In 2021 International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT) (pp. 647-650). IEEE.
16. Hans, A.S.A., Bansal, M. and Rao, S., 2021, July. An Advanced Facial Expression Detection using Deep Neural Network. In 2021 8th International Conference on Smart Computing and Communications (ICSCC) (pp. 204-208). IEEE.

17. Verma, V. and Rani, R., 2021, August. Recognition Of Facial Expressions Using A Deep Neural Network. In 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN) (pp. 585-590). IEEE.
18. Hardjadinata, H., Oetama, R.S. and Prasetiawan, I., 2021, October. Facial Expression Recognition Using Xception And DenseNet Architecture. In 2021 6th International Conference on New Media Studies (CONMEDIA) (pp. 60-65). IEEE.
19. Lokku, G., Reddy, G.H. and Prasad, M.G., 2021, June. A Robust Face Recognition model using Deep Transfer Metric Learning built on AlexNet Convolutional Neural Network. In 2021 International Conference on Communication, Control and Information Sciences (ICCISc) (Vol. 1, pp. 1-6). IEEE.
20. Sekaran, S.A.R., Lee, C.P. and Lim, K.M., 2021, August. Facial emotion recognition using transfer learning of AlexNet. In 2021 9th International Conference on Information and Communication Technology (ICoICT) (pp. 170-174). IEEE.
21. Nan, Y., Ju, J., Hua, Q., Zhang, H. and Wang, B., 2022. A-MobileNet: An approach of facial expression recognition. Alexandria Engineering Journal, 61(6), pp.4435-4444.
22. Sun, L., Ge, C. and Zhong, Y., 2021, September. Design and Implementation of Face Emotion Recognition System Based on CNN Mini_Xception Frameworks. In Journal of Physics: Conference Series (Vol. 2010, No. 1, p. 012123). IOP Publishing.
23. He, B., 2022, May. Deep learning algorithm for Facial Expression Classification. In 2022 3rd International Conference on Computer Vision, Image and Deep Learning & International Conference on Computer Engineering and Applications (CVIDL & ICCEA) (pp. 386-391). IEEE.
24. Phattarasooksirot, P. and Sento, A., 2022, May. Facial Emotional Expression Recognition Using Hybrid Deep Learning Algorithm. In 2022 7th International Conference on Business and Industrial Research (ICBIR) (pp. 323-329). IEEE.
25. Jayasimha, Y., Venkatesha, M. and Reddy, R.V.S., 2022, May. Face Detection and its Features Extraction using Convolution Neural Network Model. In 2022 3rd International Conference for Emerging Technology (INCET) (pp. 1-6). IEEE.
26. Sadikoğlu, F.M. and Mohamed, M.I., 2022, August. Facial Expression Recognition Using CNN. In 2022 International Conference on Artificial Intelligence in Everything (AIE) (pp. 95-99). IEEE.
27. Ramu, T. and Muthukumar, A., 2022, January. A GoogleNet architecture based Facial emotions recognition using EEG data for future applications. In 2022 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-7). IEEE.

28. www.turing.com. (n.d.). *How to Use Deep Learning for Face Detection and Recognition Systems?* [online] Available at: https://www.turing.com/kb/using-deep-learning-to-design-face-detection-and-recognition-systems.
29. www.researchgate.net. Sentence-Level Classification Using Parallel Fuzzy Deep Learning Classifier? [online] Available at: https://www.researchgate.net/publication/348703101_Sentence-Level_Classification_Using_Parallel_Fuzzy_Deep_Learning_Classifier

APPENDIX

APPENDIX A: Define folders path

Importing necessary library

```
# importing various libraries
import numpy as np
import pandas as pd
import os
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
import seaborn as sns
import skimage.io
import cv2
import glob
import random
from collections import Counter
import keras.backend as K
import tensorflow as tf
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from keras.applications.mobilenet import MobileNet, preprocess_input
from tensorflow.keras.layers import GaussianNoise
from tensorflow.keras.layers import Dense, Flatten, Dropout,BatchNormalization ,Activation, MaxPooling2D
from tensorflow.keras import regularizers
from tensorflow.keras.models import Model, Sequential
from keras.layers import Conv2D
from keras.applications.nasnet import NASNetLarge
from tensorflow.keras.callbacks import ReduceLROnPlateau, ModelCheckpoint, EarlyStopping
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.preprocessing import image
from sklearn.metrics import classification_report, confusion_matrix
from tensorflow.keras.preprocessing.image import ImageDataGenerator,load_img
from keras.models import load_model
from tensorflow.keras.preprocessing import image
from tensorflow.keras.preprocessing.image import ImageDataGenerator,load_img
import itertools
```

Define the folder paths

```
[ ] #defining the folder paths
    base_dir = "/content/drive/My Drive/FER2013"
    train_dir = os.path.join(base_dir,'train')
    test_dir = os.path.join(base_dir, 'test')
```

APPENDIX B: Data Pre-Processing/ Data Analysis

```
# plotting some images from each emotions of the classes
fig, ax = plt.subplots(7,5, figsize=(18,18))

for class_id in range(7):
    folder = os.path.join(train_dir,str(class_id))
    os.chdir(folder)
    samples = random.sample(os.listdir(folder), 7)

    if class_id == 0:
      mystr1 = 'angry'
    elif class_id == 1:
      mystr1 = 'disgust'
    elif class_id == 2:
      mystr1 = 'fear'
    elif class_id == 3:
      mystr1 = 'happy'
    elif class_id == 4:
      mystr1 = 'neutral'
    elif class_id == 5:
      mystr1 = 'sad'
    else :
      mystr1 = 'surprise'

    for col in range(5):
        image = cv2.imread(samples[col])
        ax[class_id, col].imshow(image)
        ax[class_id, col].set_title("class_" + str(class_id)+mystr1)
        ax[class_id, col].set_axis_off()
```

APPENDIX C: Imbalanced Dataset labels

```
#defining a function to visualise the distribution
def show_distribution(folder):


    x11 = []
    x12 = []
    emotion = ['angry','disgust','fear','happy','neutral','sad','surprise']


    datagen = ImageDataGenerator()
    generator = datagen.flow_from_directory(folder)

    counter = Counter(generator.classes)
    distribution = [(class_id,round(num_images,2)) for class_id, num_images in counter.items()]

    for class_id, percentage in distribution:

      x11.append(percentage)
      class_id.astype(str)

      type(class_id)
      class_id1 = emotion[class_id]
      x12.append(class_id1)

    plt.figure(figsize=(10,6))
    sns.barplot(x=emotion, y=x11)
    plt.title('Class distribution')
    plt.ylabel('Number', fontsize=12)
    plt.xlabel('Emotions', fontsize=12)
    plt.show()
# plotting the distribution of training data
print("The distribution for the training data is :\n")
show_distribution(train_dir)
```

APPENDIX D: Image Size Verification

```
#checking the size of the images
folder = os.path.join(train_dir,'0')
os.chdir(folder)
samples = random.sample(os.listdir(folder), 7)

for filename in samples:
    image = cv2.imread(filename)
    print(image.shape)
```

APPENDIX E: ImageDataGenerator function for train the data

```
# creating variables for model
NUM_CLASSES = 8
IMAGE_SIZE=[48, 48,3]
BATCH_SIZE=32
classes=['Angry','Disgust','Fear','Happy','Neutral','Sad','Surprise']
```

```
#using imageDataGenerator function for training
train_datagen = ImageDataGenerator(rescale = 1./255,
                                   validation_split = 0.2,

        rotation_range=5,
        width_shift_range=0.2,
        height_shift_range=0.2,
        shear_range=0.2,
        #zoom_range=0.2,
        horizontal_flip=True,
        vertical_flip=True,
        fill_mode='nearest')
#using imageDataGenerator function for testing
test_datagen  = ImageDataGenerator(rescale = 1./255
                                  )
```

```
#calling the train datagen for training dataset
train_dataset  = train_datagen.flow_from_directory(directory = "/content/drive/My Drive/FER2013/train",
                                                   target_size = (48,48),
                                                   class_mode = 'categorical',
                                                   subset = 'training',
                                                   batch_size = 32)
```

```
Found 22970 images belonging to 7 classes.
```

```
#calling the test datagen for testing dataset
test_dataset = test_datagen.flow_from_directory(directory = "/content/drive/My Drive/FER2013/test",
                                                  target_size = (48,48),
                                                  class_mode = 'categorical',
                                                  batch_size = 32)
```

```
Found 7178 images belonging to 7 classes.
```

APPENDIX F: MobileNetV2 model building

```
# Building Model
model=Sequential()
model.add(base_model)
model.add(BatchNormalization())
model.add(GaussianNoise(0.01))
model.add(GlobalAveragePooling2D())

model.add(Flatten())
model.add(Dense(256, activation='relu',kernel_regularizer=regularizers.l2(0.001),bias_regularizer=regularizers.l2(0.001)))
model.add(BatchNormalization())
model.add(Dropout(0.5))

model.add(Dense(128, activation='relu',kernel_regularizer=regularizers.l2(0.001),bias_regularizer=regularizers.l2(0.001)))
model.add(BatchNormalization())
model.add(Dropout(0.5))

model.add(Dense(7, activation="softmax"))

model.summary()
```

APPENDIX G: 13 layer CNN Model Building

```
emotion_model = Sequential()
emotion_model.add(Conv2D(32, kernel_size=(3, 3), activation='relu', input_shape=(48,48,1)))
emotion_model.add(Conv2D(64, kernel_size=(3, 3), activation='relu'))
emotion_model.add(MaxPooling2D(pool_size=(2, 2)))
emotion_model.add(Dropout(0.25))
emotion_model.add(Conv2D(128, kernel_size=(3, 3), activation='relu'))
emotion_model.add(MaxPooling2D(pool_size=(2, 2)))
emotion_model.add(Conv2D(128, kernel_size=(3, 3), activation='relu'))
emotion_model.add(MaxPooling2D(pool_size=(2, 2)))
emotion_model.add(Dropout(0.25))
emotion_model.add(Flatten())
emotion_model.add(Dense(1024, activation='relu'))
emotion_model.add(Dropout(0.5))
emotion_model.add(Dense(7, activation='softmax'))

emotion_model.summary()
```

APPENDIX H: VGG-19 Model Building

```
# Building Model
model=Sequential()
model.add(base_model)
model.add(BatchNormalization())
model.add(GaussianNoise(0.01))
model.add(GlobalAveragePooling2D())

model.add(Flatten())
model.add(Dense(256, activation='relu',kernel_regularizer=regularizers.l2(0.001),bias_regularizer=regularizers.l2(0.001)))
model.add(BatchNormalization())
model.add(Dropout(0.5))

model.add(Dense(128, activation='relu',kernel_regularizer=regularizers.l2(0.001),bias_regularizer=regularizers.l2(0.001)))
model.add(BatchNormalization())
model.add(Dropout(0.5))

model.add(Dense(7, activation="softmax"))

model.summary()
```

APPENDIX J: Ethics Form

Ethics.pdf

Ethical clearance for research and innovation projects

Project status

Status

Approved

Actions

Date	Who	Action	Comments
18:52:00 18 January 2023		Supervisor approved	
18:34:00 18 January 2023		Principal investigator submitted	

Get Help

Ethics release checklist (ERC)

Project details

Project name: Human Emotion Recognition Using Deep Learning

Principal investigator:

Faculty: Faculty of Business, Law and Digital Technologies

Level: Postgraduate

Course: MSc Applied AI & Data Science

Unit code: com726

Supervisor name:

Other investigators:

Checklist

Question	Yes	No
Q1. Will the project involve human participants other than the investigator(s)?	○	◉
Q1a. Will the project involve **vulnerable participants** such as children, young people, disabled people, the elderly, people with declared mental health issues, prisoners, people in health or social care settings, addicts, or those with learning difficulties or cognitive impairment either contacted directly or via a **gatekeeper** (for example a professional who runs an organisation through which participants are accessed; a service provider; a care-giver; a relative or a guardian)?	○	◉
Q1b. Will the project involve the use of **control groups** or the **use of deception**?	○	◉
Q1c. Will the project involve any **risk to the participants' health** (e.g. intrusive intervention such as the administration of drugs or other substances, or vigorous physical exercise), or involve psychological stress, anxiety, humiliation, physical pain or discomfort to the investigator(s) and/or the participants?	○	◉
Q1d. Will the project involve **financial inducement** offered to participants other than reasonable expenses and compensation for time?	○	◉
Q1e. Will the project be carried out by individuals unconnected with the University but who wish to use staff and/or students of the University as participants?	○	◉
Q2. Will the project involve sensitive materials or topics that might be considered offensive, distressing, politically or socially sensitive, deeply personal or in breach of the law (for example criminal activities, sexual behaviour, ethnic status, personal appearance, experience of violence, addiction, religion, or financial circumstances)?	○	◉
Q3. Will the project have detrimental impact on the environment, habitat or species?	○	◉
Q4. Will the project involve living animal subjects?	○	◉
Q5. Will the project involve the development for export of 'controlled' goods regulated by the Export Control Organisation (ECO)? (This specifically means military goods, so called dual-use goods (which are civilian goods but with a potential military use or application), products used for torture and repression, radioactive sources.) Further information from the Export Control Organisation	○	◉
Q6. Does your research involve: the storage of records on a computer, electronic transmissions, or visits to websites, which are associated with terrorist or extreme groups or other security sensitive material? Further information from the Information Commissioners Office	○	◉

Declarations

I/we, the investigator(s), confirm that:

☑ The information contained in this checklist is correct.

☑ I/we have assessed the ethical considerations in relation to the project in line with the University Ethics Policy.

☑ I/we understand that the ethical considerations of the project will need to be re-assessed if there are any changes to it.

☑ I/we will endeavor to preserve the reputation of the University and protect the health and safety of all those involved when conducting this research/enterprise project.

☑ If personal data is to be collected as part of my project, I confirm that my project and I, as Principal Investigator, will adhere to the General Data Protection Regulation (GDPR) and the Data Protection Act 2018. I also confirm that I will seek advice on the DPA, as necessary, by referring to the Information Commissioner's Office further guidance on DPA and/or by contacting information.rights@solent.ac.uk. By Personal data, I understand any data that I will collect as part of my project that can identify an individual, whether in personal or family life, business or profession.

☑ I/we have read the prevent agenda.