A DISSERTATION

ON

EMOTION DETECTION USING SEPARABLE CONVOLUTIONS

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

MASTER OF TECHNOLOGY

IN

INFORMATION SYSTEMS

Submitted by:

AARTI SONI

2K21/ISY/01

Under the supervision of

DR. BINDU VERMA

Assistant Professor

Department of Information Technology

INFORMATION TECHNOLOGY

DELHI TECHNOLOGY UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

MAY, 2023

CANDIDATE'S DECLERATION

I, Aarti Soni 2k21/ISY/01 of M.Tech (Information System), hereby declare that the dissertation Report titled "EMOTION DETECTION USING SEPARABLE CONVOLUTION" which is submitted by me to the department of Information System, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is the original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi Aarti Soni

Date:

CERTIFICATE

I, Aarti Soni, 2k21/ISY/01 of M.Tech (Information System), hereby declare that the project Report titled "EMOTION DETECTION USING SEPARABLE CONVOLUTION" which is submitted by me to the department of Information System, Delhi Technological University, Delhi in partial fulfilment of the requirement for the award of the degree of Master of Technology, is the original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi

Date:

Dr. Bindu Verma

Assistant Professor

ACKNOWLEDGEMENT

I want to thank my Mentor, Dr Bindu Verma, for her constant support, patience, and trust in me, as well as for providing a good environment and giving me with helpful comments to me to research in this area. It has been a totally new area to work upon, but mam had generously supported us to research on this area.

ABSTRACT

Emotion detection is a tremendously booming field that aims to identify and analyze human emotions using different manner such as text, speech, and facial features. This thesis provides a detailed point of the state of the art methods, applications, challenges in emotion detection. It begins with an introduction to emotions with their importance in human-computer interaction, followed by a review of the most embossed methods employed in emotion recognition. The thesis then goes into the applications of emotion detection in different field, such as mental health, education and marketing and discusses the ethical considerations and challenges faced by the field. Finally, it proposes future research ways to further advance the field of emotion detection and improve its impact in the ordinary life of human.

Table of Contents:

LIST OF FIGURES

List of Symbols, Abbreviations and Nomenclature

CNN – Convolutional Neural Network

AI –Artificial Intelligence

HCI – Human Computer Interaction

NLP- Natural Language Processing

RNN- Recurrent Neural Network

CHAPTER 1

1.1Introduction

Emotions plays a pivotal role in human communication and interaction. They are the underlying force that drives our behaviours, decision-making processes, and relationships. As human beings, we possess the innate ability to perceive, interpret and respond to emotions expressed by others into various forms. This ability is essential for social functioning and has been a subject of intense scientific inquiry for centuries. With the rapid advancement in technology, there has been a growing curiosity in developing computational models and systems capable of detecting emotions in recent years. This thesis aims to explore the multifaceted aspects of emotion detection, including its underlying theories, methodologies, applications, and challenges.

The field of emotions is a different field that spans psychology, neuroscience, sociology, and anthropology. It is in close relation to artificial intelligence, machine learning, and human-computer interaction. The importance of emotion detection lies in its capability to a deeper understanding of human emotional experiences, which can then be used to improve various aspects of our lives, such as mental health, education, marketing, and entertainment.

This thesis will first go into the theory of emotions and how they are graded. The fundamental emotion theory the dimensional theory and the appraisal theory are a few of the well-known theories of emotion that we will look upon. This part will also cover the various emotional categories, ranging from basic feelings like happy, sorrow, angry, and fear to more subtle and deep ones like envy, remorse, and shame. To create projects that can effectively recognize and interpret human emotions, it is essential to understand the complex part that makes up these feelings.

We will also go through the various techniques and approaches used in emotion recognition. Physiological cues, body language, voice cues, facial expressions, and language content make up this list. These signals can be extracted and analysed using a variety of approaches developed by researchers, and each method provides unique information on an individual's emotional state. We will discuss the possibilities of applying multimodal methods for improved emotion recognition as well as the advantages and disadvantages of each method.

In addition to understanding the methodologies for emotion detection, it is crucial to study the ethical implications and potential biases that may arise during the deployment of these systems. This thesis will dedicate a section to the discussion of ethical concerns, such as data privacy, informed consent, and the potential for discrimination or misinterpretation of emotions. We will also address the challenges associated with cultural, linguistic, and individual differences in emotional expression and perception, which may impact the generalizability and accuracy of emotion detection systems.

The interconnection of emotions is represented by Plutchik's wheel of emotions, a two-dimensional model based on the two main variables of arousal and valence. Arousal, which ranges from low to high, is represented by the horizontal axis and refers to the strength or level of activation of an emotion. Valence, which ranges from positive to negative on the vertical axis, shows the degree of pleasantness or unpleasantness of a feeling. This model shows how various emotions relate to one another and how they can be grouped according to where they are located on the wheel.

The wheel consists of multiple layers of circles, with each circle representing a specific emotion or a combination of emotions. The innermost layer contains derivatives of the eight basic emotions, which are considered to be primary or fundamental emotions. These basic emotions include joy, sadness, anger, fear, surprise, disgust, trust, and anticipation. The next layer of the wheel consists of the eight basic emotions themselves. Each basic emotion is positioned next to its closest related emotions based on their similarities and differences in terms of Arousal and Valence. For example, emotions with similar Arousal and Valence values will be placed closer to each other on the wheel.

Finally, the outermost layer of the wheel represents combinations of the principal emotions. These combinations arise when two or more basic emotions are experienced simultaneously or in close succession. Plutchik proposed that these combinations could give rise to more complex emotions.

By using this wheel of emotions, one can visually understand the relationship between different emotions based on their positions on the wheel. It provides a framework for analyzing and categorizing emotions, taking into account their Arousal and Valence dimensions and their proximity to each other on the wheel.

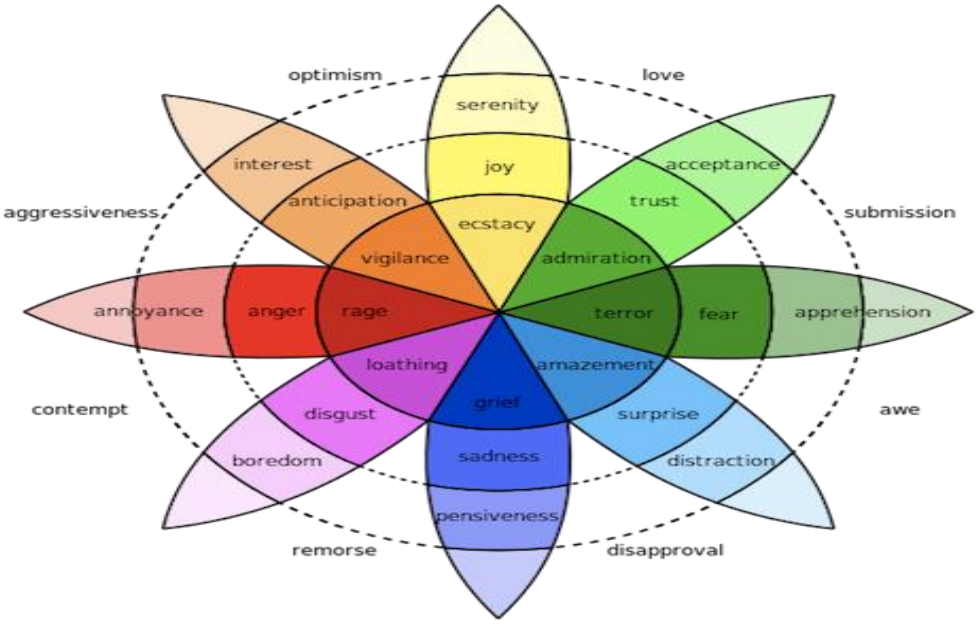

Fig 1.1.1: Plutchik wheel of emotion

Another important aspect of emotion detection is its practical applications. This thesis will explore a variety of domains where emotion detection technology can be beneficial. These include mental health care, where emotion detection systems can help clinicians monitor patients' emotional well-being and provide targeted interventions; education, where recognizing students' emotions can lead to personalized learning experiences; marketing, where understanding consumers' emotional responses can inform advertising strategies; and entertainment, where emotion-aware systems can enhance user experiences in gaming and virtual reality environments.

1.2 Motivation and Significance

It is essential to create systems that can react to users' emotional states as technology becomes more ingrained in daily life, boosting their experiences and general wellbeing. By enabling robots to offer more individualised and sympathetic answers, emotion detection systems have the ability to enhance the effectiveness of human-machine interactions. Applications for emotion detection can be found in many different fields, including: -

Mental Health: Therapists, counsellors, and psychiatrists can benefit from the knowledge that emotion detection systems can provide by monitoring patients' emotional states. Based on a person's emotional requirements, these systems can also be utilised to create customised interventions and treatment plans.

Education: By utilising emotion detection technologies, educators can modify their instructional methods in order to better comprehend the emotional conditions of their

students. Educators can understand more about the emotions that students are experiencing while they are studying by using emotion detection technologies. By adjusting their teaching strategies to each student's unique needs, they may improve the learning environment and make it more enjoyable and effective.

Customer Service: By recognizing customers' emotions, businesses can provide personalized and empathetic customer service, leading to higher customer satisfaction and retention rates.

Entertainment: By altering material based on consumers' emotional responses, emotion detection can be used to improve the user experience in gaming, virtual reality, and other entertainment platforms.

Marketing: Businesses can make more educated marketing decisions and increase the overall efficacy of their campaigns by better understanding consumers' emotional responses to commercials or products.

It is crucial to keep developing the field of emotion detection and researching new methodologies, strategies, and algorithms for enhanced performance and adaptability given the wide range of applications and the potential to revolutionise human-machine interactions.

The multidisciplinary field of affective computing, often known as emotion detection, combines computer science, psychology, and cognitive science to create technology that can identify, comprehend, and react to human emotions. Rosalind Picard introduced the idea of building emotionally intelligent machines in her seminal book "Affective Computing," released in 1997. Since then, this discipline has advanced significantly, and algorithms for detecting emotions are currently used in a variety of fields like marketing and human-computer interaction.

It is impossible to overestimate the significance of emotion recognition because emotions are fundamental to human communication, decision-making, and general well-being. Multiple means, body language, voice, physiological markers, are used to communicate emotions. To effectively identify a user's emotional state, an emotion detection system must be able to gather and analyze data from these many sources. Such systems need to be developed, and they need to be developed using cutting-edge computational techniques.

The lack of a generally acknowledged taxonomy of emotions is one of the main obstacles to emotion detection. While there are various theories about emotions, there is disagreement about the number or character of the fundamental emotions. In Ekman's paradigm of fundamental emotions, there are six universal emotions—happiness, sorrow, anger, fear, surprise, and disgust—is the basis for the majority of emotion recognition technologies. This approach has drawn criticism for its oversimplification and disregard for more complex emotional experiences. The circumplex model, which contends that emotions can be represented in a continuous two-dimensional space delineated by valence and arousal dimensions, is one of the different emotion models that researchers have investigated as a result.

1.3 Goal

Finally, we will conclude by discussing the limitations and future directions of emotion detection research. While significant progress has been made in recent years, there remain numerous challenges to overcome. These include improving the accuracy and reliability of emotion recognition systems, addressing ethical concerns, and developing culturally sensitive and inclusive models.

In the era of ever-evolving technology and artificial intelligence, human-computer interaction (HCI) has become an essential aspect of modern life. As humans, we often rely on emotional cues to understand and interpret others, which is a critical component of effective communication. This need for emotional understanding extends to our interactions with machines and artificial intelligence systems, as well. The capability of machines to recognize, interpret, and respond to human emotions has become a fundamental research area, and the development of emotion detection systems aims to bridge the gap between humans and machines, leading to more seamless and natural interactions.

1.4 Scope

Deep learning methods, in particular, have revolutionized the field of emotion recognition in past period. Deep learning algorithms have displayed great performance in tasks like emotion classification based on physiological cues, voice emotion detection, and facial expression identification. These algorithms are highly suited for addressing the complexity and variety included in human emotional responses because they can automatically discover relevant characteristics from unstructured input.

However, despite the advancements in emotion detection, several challenges remain. One of the most significant issues is the absence of large, diverse, and well-annotated datasets for training and assessing emotion detection models. Image dataset is not easy to make up of Many existing datasets suffer from biases in terms of age, sex, and cultural background of the participants, which can limit the capacity of the trained models. Additionally, the process of emotion annotation is often subjective and prone to inconsistencies, as different annotators may interpret the same emotional expressions differently.

Handling individual variances in emotional expression and perception presents another difficulty in emotion identification. People differ greatly in their emotional expression and interpretation, which can make it challenging for a one-size-fits-all model to correctly identify emotions across various users. It has been suggested that a potential solution to this issue is to personalise emotion recognition models based on individual factors, such as personality traits or cultural background.

CHAPTER 2

LITERATURE REVIEW

Emotion detection, an essential aspect of (HCI) and artificial intelligence (AI) has got much attention in the years. Emotion detection, which involves the identification and classification of emotions from various techniques (e.g., text, speech, and facial expressions), has numerous applications, including mental health monitoring, customer service, and entertainment. The field of emotion detection has significantly evolved, embracing a variety of approaches, encountering difficulties, and finding applications in a variety of fields. The goal of this literature review is to give readers a thorough overview of the state of the area right now.

2.1. Evolution of Emotion Detection

Early Work and Theories (Ekman, 1992; Plutchik, 1980)

Researchers like Ekman and Friesen were pioneers in the field of emotion recognition in the 1960s. They established this branch of study with their research on face expression recognition. Happiness, sadness, wrath, fear, disgust, and surprise were six unique emotions that Ekman distinguished as being recognisable in all cultures [1].

Plutchik suggested a more thorough model known as the "Wheel of Emotions" in order to build on Ekman's work. Eight basic emotions were included in this model: rage, excitement, joy, trust, fear, surprise, sadness, and disgust. The circular design of this model, which symbolised the interconnections and complexity of human emotions, was its defining characteristic [2].

Emotion Representation Models (Russell, 1980; Mehrabian, 1996)

Emotion representation models have evolved to include the circumplex model [3], which posits that emotions can be denoted as points in a two-D space (valence and arousal). Mehrabian [4] later developed the PAD model, adding a third dimension: dominance.

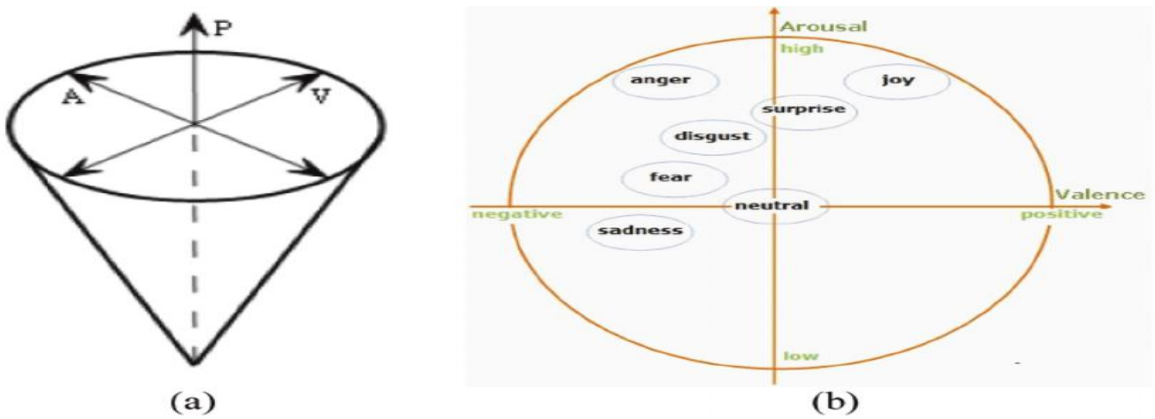

Figure 2.1.1: 3D Emotion Space (Valence, Arousal, and Power) Images taken from (Jin & Wang, 2005) and (Breazeal, 2003) respectively.

2.2 Methodologies in Emotion Detection

Text-based Emotion Detection (Pang and Lee, 2008; Liu, 2012)

Text-based emotion detection relies on natural language processing (NLP) techniques to extract emotion-related information from text. Early methods focused on sentiment analysis, determining the (positive, negative, or neutral) of a given text. More advanced techniques have emerged, such as deep learning and word embedding [6], which better capture the details of emotion in text.

Speech-based Emotion Detection (Schuller et al., 2013; Lee and Narayanan, 2005)

Signal processing and machine learning techniques are used to understand parameters like pitch and energy within vocal expressions for emotion detection through speech. Significant progress has been made in this area throughout time, particularly with the use of complex deep learning systems. Recurrent neural networks (RNNs) and convolutional neural networks (CNNs) are two examples of these that have been shown to be efficient at determining emotional states from speech patterns [7].

Visual-based Emotion Detection (Tian et al., 2001; Pantic and Bartlett, 2007)

The study of facial expressions, posture, and gestures is known as "visual-based emotion detection," and it aims to recognise and comprehend human emotions. Early studies in this field analysed and categorised facial expressions using manual coding techniques like the Facial Action Coding System (FACS).

However, more recent work has moved towards automated methods for emotion identification as a result of developments in computer vision techniques and machine learning algorithms. These methods automate [9].

Multimodal Emotion Detection [10]

The development of multimodal emotion detection has given emotion detection a more thorough and accurate makeover. This method combines many techniques, such as text analysis, speech processing, and visual signals, to offer a comprehensive knowledge of emotional states. Strategies like feature-level fusion and decision-level fusion have been devised to efficiently combine data from these various modalities. To improve the precision of emotion identification, these techniques combine numerous data points [11].

2.3 Emotion Detection Approaches

There are three major approaches to emotion detection: behavioral, physiological, and provisional. Each approach has its pros and cons and so the combination of this can lead to accurate and good detection system.

Physiological Approaches

Physiological approaches to emotion detection focus on the bodily changes associated with emotional experiences, such as heart beat rate, skin conductance and brain waves.

Autonomic Nervous System (ANS) Measures

Emotions are known to elicit changes in ANS activity, including heart rate changes [12] and skin feeling responses [13]. ANS measures have been used for emotion detection in different places, including affective computing and mental health research.

Neurophysiological Measures

Many neurophysiological methods have been used to investigate the neural basis of emotion, including Electroencephalography (EEG) and Functional Magnetic Resonance Imaging (fMRI) [14]. Furthering our understanding of this intricate interplay, the use of machine learning to these neurophysiological data sets has shown promise in the area of emotion detection [15].

2.4 Behavioural Approaches

Behavioural approaches to emotion detection focus on observable expressions, including facial expressions, vocal cues, and body language.

Facial Expression Analysis

Automatic facial expression recognition (AFER) systems analyse facial movements to detect emotions. Methods like the Facial Action Coding System [16] and machine learning algorithms have been employed to recognize emotions from facial expressions [17].

Recent advancements in computer vision and machine learning have facilitated automatic facial expression analysis. Kanade, Cohn, and Tian [18] introduced the first automated FACS-based facial expression recognition system. Since then, several approaches have been proposed, such as Active Appearance Models (AAM) [19] in particular, have demonstrated significant improvements in emotion detection accuracy, thanks to their ability to learn hierarchical features from raw pixel data.

Speech Analysis

Speech is another widely used modality for emotion detection. Early studies in this area focused on prosodic features, such as pitch, intensity, and duration [20]. Later research identified additional acoustic features, such as spectral characteristics, which could be used to discriminate between different emotions [21].

Body Language Analysis

Body language, or nonverbal behavior, is a crucial component of human communication and emotion expression. Early studies in this area relied on manual coding of body language cues, such as the Body Action and Posture Coding System (BAP) [22]. However, manual coding is labor-intensive and subjective, prompting researchers to develop automated methods.

Kinect, a motion sensing device developed by Microsoft, has facilitated the automatic extraction of body language features [23]. Deep learning techniques, such as 3D convolutional neural networks (3D-CNN) and pose-based CNNs, have also been proposed for body language-based emotion detection [24].

2.5 Types of Emotion Detection:

Text-Based Emotion Detection

Sentiment Analysis: The strategy is to evaluate the tone of the material and categorise it as either positive, negative, or neutral. Natural language processing (NLP), text analysis,

and computational linguistics are frequently combined to extract subjective information from sources. For this, a variety of machine learning models can be used, including Naive Bayes, Logistic Regression, and sophisticated Deep Learning models like Recurrent Neural Networks (RNN).

Emotion Lexicons: The established emotion lexicons found in published works of literature are frequently used for emotion detection in text. These lexicons are generally produced by posts on social media sites like Twitter that use emotion-related hashtags like #happy or #sad. This allows for a more accurate assessment of the emotional content of written language. These lexicons are then primarily related to emotional descriptors including wrath, fear, anticipation, surprise, happiness, and trust.

Deep Learning: A form of machine learning called deep learning (DL) allows programmes to learn by experiencing and comprehending the hierarchy of concepts, where each concept is explained in terms of how it relates to simpler concepts. By constructing complicated ideas on the foundation of simpler ones, this technique aids a program's learning [25]. DL model is referred to as long short-term memory (LSTM) in numerous research studies. An RNN (recurrent neural network) with long-term dependency management skills makes up the LSTM. The common problem of disappearing or exploding gradient in RNNs is resolved by LSTM.

Speech-Based Emotion Detection

Rhythmic Feature Analysis: This involves analyzing the pitch, volume, and speed of speech, which can be indicators of certain emotions.

MFCC (Mel-frequency cepstral coefficients): A speaker's emotion can be detected from their speech using the Mel Frequency Cepstral Coefficient (MFCC) approach. The developed system's efficacy was determined to be roughly 80% when it was validated for happiness, sadness, and rage.

Image-Based Emotion Detection

Facial Expression Analysis: This involves the use of computer vision techniques to detect facial features that correspond to certain emotions.

Deep Learning: Techniques like CNN can be used for facial expression recognition. More complex models such as Capsule Networks (CapsNets) can also be used.

Eye Tracking: This involves tracking eye movements and pupil dilation, which can be indicative of certain emotional states.

Physiological-Based Emotion Detection

GSR (Galvanic Skin Response): Changes in skin sweat gland can indicate emotional arousal.

Heart Rate Variability: Changes in heart rate can be linked to different emotional states**.**

EEG (Electroencephalogram): Brainwaves measured by EEG can be used to detect emotions.

The best method for emotion detection depends on the dedicated uses and the availability of data. We can also combine different methods for the most accurate results. It should be noted that all these methods have limitations and may not always be 100% accurate. Emotion is a complex and subjective occurrence, and what works well for one person may not work as well for another.

CHAPTER 3

PROPOSED WORK

A detailed description of the proposed model along with the methodology involved is mentioned in the below sections.

3.1 OVERVIEW:

A brief block diagram of the entire work done in this project is shown in Fig 3.1.1.
The workflow diagram step by step is shown in Figure 3.1.1.

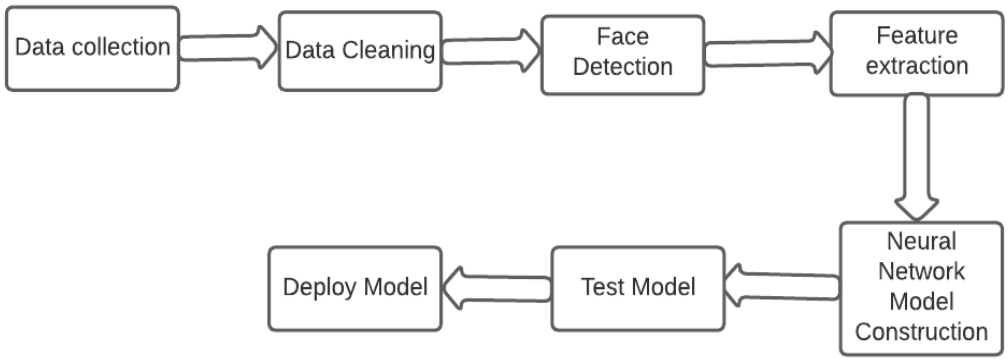

Fig 3.1.1 Emotion Detection Flowchart

There are basically 7 steps taking place in this project:

1. Data collection
2. Data Cleaning
3. Face Detection
4. Feature extraction
5. Neural Network Model Construction
6. Test Model
7. Deploy Model

Artificial intelligence (AI) models that analyse and interpret human emotions based on facial expressions are known as facial emotion recognition (FER) systems, sometimes known as emotion detection systems utilising pictures. Since it directly affects the system's accuracy and dependability, the data collection method for these systems is vital. The procedure for gathering data for such systems is described in the following steps. Images of human faces showing different emotions serve as the main source of data for these systems. Images can be collected in several ways:

1.Data Collection

Existing Databases: The Extended Cohn-Kanade (CK+) dataset, the Japanese Female Facial Expression (JAFFE) dataset, and the Facial Expression detection (FER2013) dataset are just a few examples of the numerous publicly available datasets that support emotion detection research. Pre-labeled photos from these sets make for priceless training and testing materials for artificial intelligence algorithms.

Real-time Capture: In this method, images are captured in real-time using a camera. In this approach, we will have more diverse images which have different skin tones, different lighting conditions and various expressions. However, these images will need to be manually labeled.

2. **Data cleaning** is an essential process in any machine learning project. In the context of an emotion detection system using images, it involves removing or correcting erroneous, incomplete, or irrelevant data from the dataset. This process is crucial to improvize performance and accuracy of the machine learning model used for emotion detection.

Here are some detailed steps on how data cleaning might be performed in an emotion detection system using images:

Data Formatting: The data we collect from different points and could be in different formats. The first steps which is in data cleaning is to make the data in the standard format for image data, this could involve converting all images into the same file format (like JPEG or PNG), size, and color scheme which the model can understand.

Missing Data: Missing data will lead to accuracy issue in the project and it is difficult to fill missing values and instances. In the case of images, it is not easy to fill missing images and label them. It may need another model to fill up the space.

Duplicate Data: Duplicate images can bias the model towards certain emotions. Hence, duplicates should be identified and removed. Duplicates might be exact similar images or slightly changes version of the images and so different models might be needed to identify them.

Distorted Images: Distorted images can drastically affect the performance of your model. In the context of emotion detection, the images that don't contain faces, images with multiple faces, or images with distorted or obscured faces. These should be identified and handled appropriately, either by removing them or by developing methods to process them correctly.

Data Augmentation: We develop modified datasets in this process. By artificially expanding your dataset, data augmentation can assist in improving the resilience of your model. Flipping, rotating, zooming, or adding noise to your photographs are some examples of how to do this. This step can help your model respond to new data more effectively.

Data Verification: After all the steps, check the quality of the data. You can randomly take a sample of data and manually check it to ensure that it meets your expectations. It is a critical step that can significantly impact the performance of your emotion detection system. It can also be a very iterative process, where you might need to revisit earlier steps based on findings from later steps. It is an important procedure so don’t rush through it.

3. Face Detection

Detecting the face in an image is the initial step in an emotion detection system. The Haar Cascade classifier, Histogram of Oriented Gradients (HOG), or more sophisticated deep learning techniques like Convolutional Neural Networks (CNNs) or Multi-task Cascaded Convolutional Networks (MTCNNs) can all be used to do this.

For instance, the Haar Cascade classifier uses machine learning and trains a cascade function using a large number of positive faces and negative images. The faces in the image are then found using it.

On the other hand, techniques like MTCNN, is useful for both face alignment and face detection. It depends on landmark locations of face such as eyebrows, nose on the image.

4. Feature extraction

Emotion detection using images is a difficult task that requires advanced techniques in machine learning and image processing. The extracted features serve as inputs to machine learning algorithms that classify the emotion present in the image. This step involved extracting of crucial information that is useful in detection.

Facial Landmark Detection: This is an important part of feature extraction in emotion detection systems. Facial landmarks are points within a face. These include features such as the eyebrows, nose, mouth, and jawline. These landmarks can be used to identify expressions that correspond to different emotions. For instance, a raised eyebrow or a smile can indicate certain emotions. There are several neural models such as OpenCV's Haar cascades, or MTCNN (Multi-task Cascaded Convolutional Networks) that can be used for facial landmark detection.

Geometric Features: These geometric features can capture the spatial dimension of facial components, which is important for emotion recognition. When facial features are detected, geometric features can be derived. For example, the distance between the eyebrows and top of the nose could be geometric feature. The angle during smiling is another example.

Appearance Features: Different techniques, such as conventional ones like Gabor filters and Local Binary Patterns (LBP) and more sophisticated deep learning techniques like Convolutional Neural Networks (CNNs), can be used to extract facial appearance features, such as wrinkles and changes in skin texture. These methods allow for the identification and capture of particular properties related to the texture and contour of the face when analysing facial photographs.

Deep Learning-Based Feature Extraction: Deep learning models have shown a great boost in the detection of images from images. Convolutional Neural Networks are capable of performing great when it comes to extracting a ladder of features from pixel values. As the layer deepens it is able to detect much more complex features like shapes and more complex features while lower level works with simple features like color corners.

After feature are extracted, the next step is feature selection, where most synonym features are selected for emotion classification. Then, these features are input to a machine learning or deep learning model for training.

5. Convolutional Neural Network (CNN) is useful for image-based models because it performs strongly in structural ranking and features.

The model typically starts with layers that involves convolutional and max-pooling layers. Convolutional layers learn local patterns through small windows that slide over the input while pooling layers reduce the (width, height) of the input to make the model more computationally efficient.

After several convolutional and pooling layers, the model typically has a few fully connected (dense) layers which operates on 1-dimension array and every input connects to every neuron.

Final layer of the network is a softmax layer which works when we have multiple classes in a model and each class will be given probability that adds upto 1.

A typical structure might be:

- Convolutional Layer (involves ReLU activation)
- Max Pooling Layer
- Dropout Layer (for regularization)
- Repeat these layers a few times, each time increase number of filters in the convolutional layers.
- Flatten Layer
- Dense Layer (ReLU or another activation function)
- Dropout Layer (for regularization)
- Output Layer (Softmax activation for multi-class classification)

Testing a model in an emotion detection system using images involves various steps which takes into account the model's performance, accuracy, and abstraction capacities. Here's a detailed approach:

Model Training and Validation:

After preprocessing the images (include grayscale conversion, normalization, and resizing), they are used to train the model. In the training model tries to link features of image to dedicated emotions. After this a part of dataset is used for validation while training to help in tuning of the model and keep away from overfitting of model.

When the model is trained it goes for testing. Testing is generally performed on a separate dataset that the model has not visited so that we can unsure a non-partial evaluation of the model.

Testing involves feeding these images in the model and comparing the model's predictions to actual labels. This comparison allows for the calculation of performance metrics.

Performance Metrics:

Various measures are used to undertake the model's performance. These may include:

Accuracy: The percentage of total output that was correct. (i.e., some emotions are represented more than others). It is not the best measure to evaluate a model because the test data can be imbalanced. It can only give a general idea.

Confusion Matrix: It is a table that allows the realization of the model's performance It is helpful in determining the types of errors which is made by model, such as which emotion is confused with others.

Precision, Recall, and F1 Score: In particular for binary classification tasks, precision, recall, and the F1 score are significant metrics used to assess the performance of a model. They shed light on the model's accuracy in detecting positive cases and effectiveness in reducing false positives and false negatives.A composite metric called the F1 score combines precision and recall to provide a more accurate assessment of a model's performance.

ROC and AUC: A graphical representation of the performance of a binary classification model as its discrimination threshold is changed is called the Receiver Operating Characteristic (ROC) curve. Applying integral calculus throughout the range from 0 to 1, the Area Under the Curve (AUC) quantifies the full two-dimensional space underneath the complete ROC curve. With regard to all feasible classification thresholds, this statistic provides an extensive evaluation of model performance.

Fine-Tuning and Model Iteration:

Based on the results, the model might need to be fine-tuned. Fine tuning relates to transfer learning it means getting knowledge from one thing and applying to another thing. The updated model would then go through the same training and testing process until satisfactory results are achieved.

In conclusion, testing an emotion detection model involves running the model on unseen data and using specific metrics to evaluate its performance. The process is iterative and may require several rounds of adjustments to the model to achieve optimal results.

7. Deploying model

When the model is optimized and converted to the correct format, it can be deployed. This involves mixing up the model into a larger system or application. For example, you might integrate your emotion detection model into a video conferencing application to give real-time feedback on participants' emotions.

Facial Emotion Recognition

Emotion detection using facial recognition in images is a significant field within computer vision and artificial intelligence. This technique employs machine learning algorithms to analyse human facial features and deduce the emotional state that person in the image is experiencing.

Detected face coordinates are fed as input to the trained model, the output bounds the face with a box and displays the emotion attached to it in the form of text.

CHAPTER 4

AlGORITHM FOR DETECTION

One of the most popular techniques for recognizing facial emotions is the Separable Convolutional Neural Network (CNN). Before sending the image to the Separable Convolutional Neural Network, it needs to be pre-processed. Convolutional neural networks (CNNs) can be implemented more effectively using separable convolution neural networks, commonly referred to as depth-wise separable convolutions. Architectures like Mobile Nets have helped to make this technology particularly well-known. In order to reduce computational complexity and increase network efficiency while retaining competitive performance, they divide convolutions into two distinct operations: depth-wise convolutions and pointwise convolutions. Convolutional neural networks (CNNs), are potent machine learning models that can automatically and adaptively learn spatial hierarchies of characteristics.

Depth-wise Convolutions: In a standard convolution where each filter is applied to all input channels, depth-wise convolutions apply a single filter one by one to all input channel. This way, it captures spatial information for each channel singly, reducing the computational cost.

Pointwise Convolutions: Many convolutional neural network designs go to a 1x1 convolution, often referred to as a pointwise convolution, after conducting depth-wise convolution, which applies distinct filters to each input channel. By calculating linear combinations of the input channels, this procedure aids in the construction of new features. A convolutional process known as a "1x1 convolution" employs a filter/kernel size of 1x1.

The purpose of spatially separable convolution is one of the recent advancements in CNNs to make them more computationally effective. To understand the concept, let's first understand what convolution operation is in the context of CNNs. Convolution operation in CNN involves applying a filter on the input image If we consider a 3D image (width, height, and depth as channels), for instance, the convolution function which requires a 3D kernel is not cost-effective for large images and large datasets.

In spatially separable convolution, instead of using a 3D kernel, the convolution operation is separated into two line-by-line operations: first across the height of the image (or feature map) using a 2D kernel (height and channels), and then across the width using another 2D kernel (width and channels). This results in the same output as the 3D convolution, but at a much lower computational cost.

Now, let us discuss how spatially separable convolution works in emotion detection using images:

Pre-processing: The input images are pre-processed to standardize their sizes and normalize their pixel values. Some systems may also use face detection algorithms to crop the faces from the images, as facial expressions are the main sign for emotion detection.

Feature Extraction: The pre-processed images are then loaded into a Convolutional neural network with spatially separable convolution layers for feature detection. The CNN can automatically learn dimensional ranking of features that are effective for emotion detection, such as the size of the eyebrows, the curves of the mouth, and the wrinkles around the eyes. The use of spatially separable convolution makes the CNN more statically efficient, allowing it to handle larger images and deeper architectures.

Classification: The features extracted by the CNN are then used for emotion classification. This can be done using a FC layer followed by a softmax layer, which detects the positibility of the images belonging to different emotion classes, such as happy, sad, angry, surprised, neutral, etc.

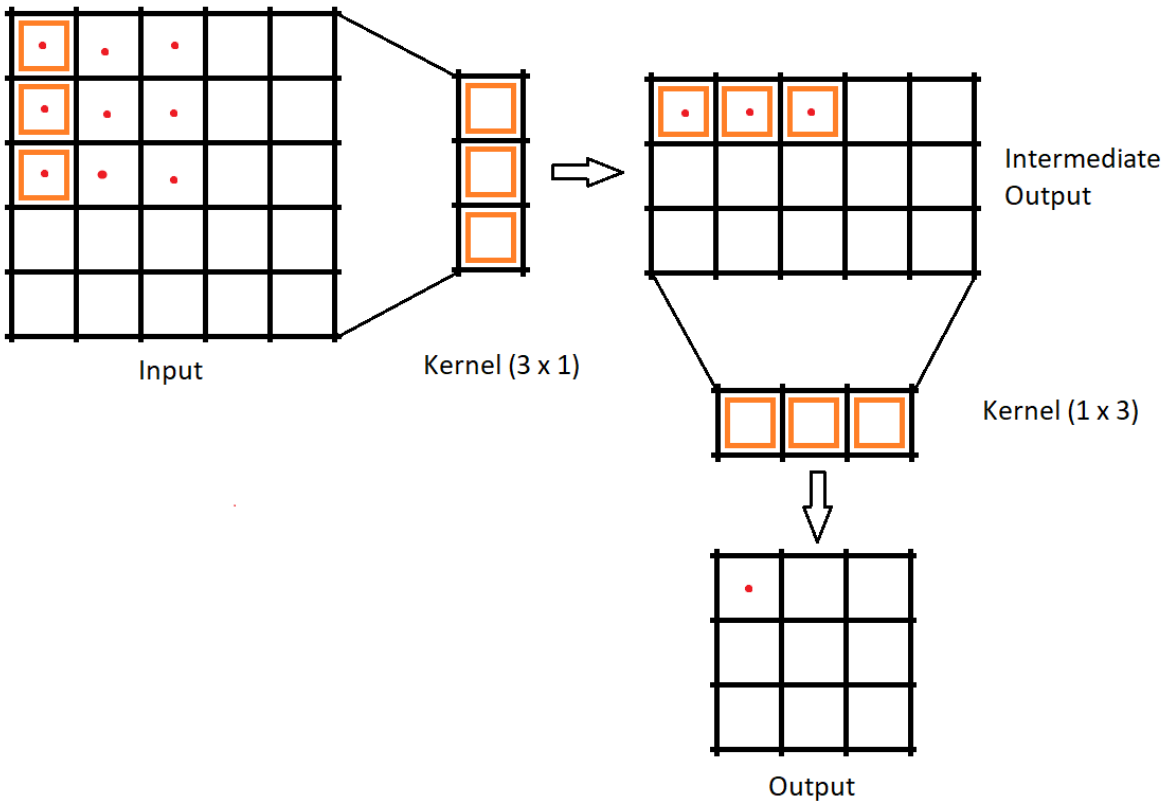

Fig 4.1 Convolutions with spatially separable kernels

Given its capacity to autonomously acquire useful characteristics and its high computational efficiency, a spatially separable convolutional neural network can undoubtedly be a potent tool for emotion detection using photos. It's crucial to remember that the effectiveness of such a system depends on a number of variables, including the calibre and diversity of the dataset, the CNN's architecture, and the complexity of the emotions being identified.

The model should be trained on a sizable dataset of labelled images in order to get good performance. There should be a label for each image in the dataset that describes the emotion it conveys. The network improves its capacity to reliably detect emotions in hidden images during training by learning to minimise the discrepancy between its predictions and the true labels.

It's also worth noting that separable convolutions aren't always the best choice for every task. They are mainly used in scenarios where computational efficiency is a priority, such as on mobile devices or in real-time applications. For tasks where computational resources are not as limited, regular convolutions might yield better performance.

4.1 Layers in the Separable Convolutional Neural Network (CNN):

Now, let's discuss an example architecture for emotion detection using images. This would be a Convolutional Neural Network (CNN) model, potentially using separable convolutions, with an architecture something like this:

Input Layer: The input would be your images. If these are color images, they might have three channels (red, green, blue) and be of a size like 48x48 pixels.

Convolutional Layers: These layers apply many filters to the image to create a set of feature maps. Using separable convolutions, the first step is applying depthwise convolutions, followed by pointwise convolutions. These layers are typically followed by a normalization layer (like batch normalization) and a non-linear activation function (like ReLU).

Pooling Layers: The computation needed is decreased by these layers' downsampling of the feature maps, which also helps to make the model insensitive to little translations of the input image.

Fully Connected Layers: After several rounds of convolutional and pooling layers, the high-level features extracted by these layers are flattened into a 1D vector and passed through one or more fully connected layers to make the final prediction.

Output Layer: A softmax layer with one node for each emotion you're attempting to forecast would be the top layer. The final outputs are transformed into probabilities via the softmax function, which also ensures that they are positive and add to 1.

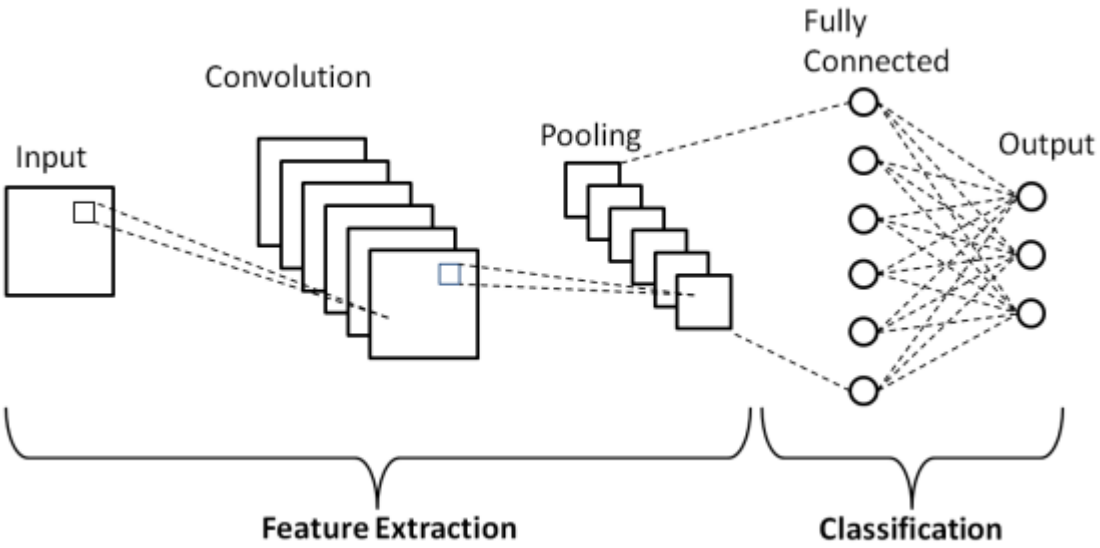

Fig 4.1.1 CNN layers shown in the CNN architecture diagram

The entire network is trained end-to-end using backpropagation and gradient descent (or a variant thereof) to update the weights in all layers to minimize the difference between the model's predictions and the true labels.

This is a broad overview of a typical emotion detection network. It's also worth noting that the exact architecture can vary significantly depending on the specifics of your problem and dataset.

4.2 ARCHITECTURE OF PROPOSED MODEL

There are basically two steps in this algorithm. The first is the training phase, and the second is the testing phase.

Training phase :- During training phase of any deep learning model, the algorithm (model), looks for patterns in the dataset. There are some few steps which takes place in this process these are mentioned below:

i. upload the dataset

ii. create train /test dataset split

iii. define the embeddings neural network

iv. train the constructed model

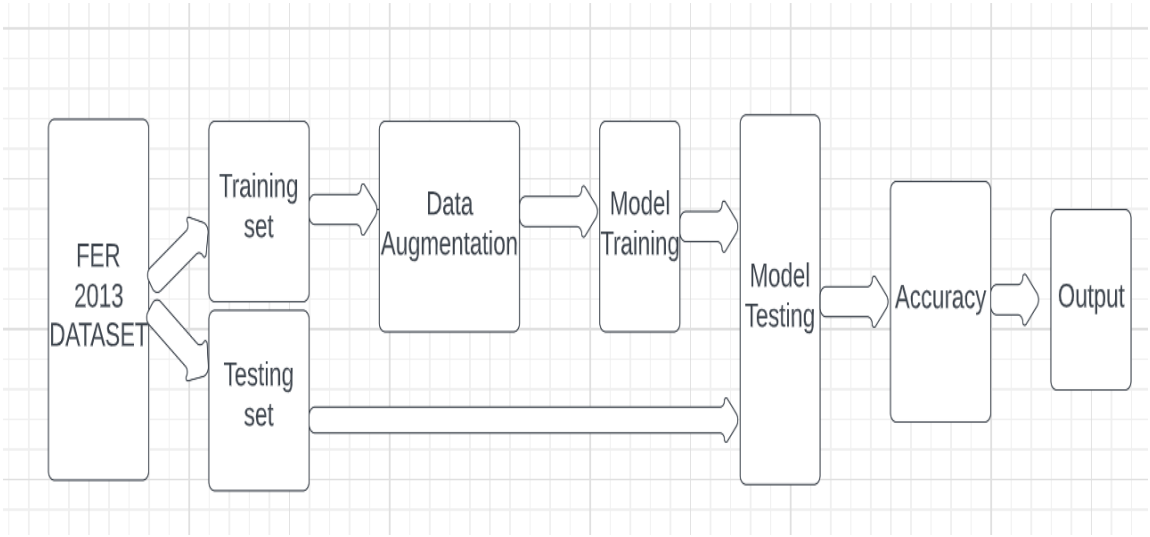

Fig 4.2.1 Block diagram of the proposed CNN architecture

All of the photos supplied into this architecture are grayscale and have a size of 48*48. The first layer is composed of a convolution layer with a kernel size of 3*3 and 32 filters; this layer is followed by batch normalisation, which is again followed by ReLU activation function; finally, the convolution layers with 64 filters of size 3*3 are followed by this layer. Batch normalisation comes next, and after that comes the ReLU function. Following Batch Normalisation, which is again followed by separable convolution layers with 128 filters of size 3*1, the input is then fed into the convolution layer with a kernel size of 1*1 and 128 filters. Then comes Batch Normalisation, and after that comes ReLU. After Batch Normalisation and Max Pooling, the output is once more fed into a separable convolution layer with a kernel size of 3*3 and some 128 filters, then Concatenate is used, then convolution layer, then batch. After normalisation, the previous phase is repeated six times, and the data is then transformed into a linear vector and supplied to dense or completely connected layers. Dropout layers are employed in this case to prevent the model from fitting too closely to the data. Once more, the actual classification is done using the fully connected layers. The SoftMax layer is ultimately used to display the findings of each category and group into any one of the seven fundamental emotions.

The model structure of the proposed model:

```
__________________
Layer (type)                 Output Shape              Param #     Connected to
==================================================================================
input_1 (InputLayer)         [(None, 64, 64, 3)]       0
__________________________________________________________________________________
conv2d (Conv2D)              (None, 62, 62, 32)        896         input_1[0][0]
__________________________________________________________________________________
batch_normalization (BatchNorma (None, 62, 62, 32)     128         conv2d[0][0]
__________________________________________________________________________________
tf.nn.relu (TFOpLambda)      (None, 62, 62, 32)        0
batch_normalization[0][0]
__________________________________________________________________________________
conv2d_1 (Conv2D)            (None, 60, 60, 64)        18496
tf.nn.relu[0][0]
__________________________________________________________________________________
batch_normalization_1 (BatchNor (None, 60, 60, 64)     256         conv2d_1[0][0]
__________________________________________________________________________________
tf.nn.relu_1 (TFOpLambda)    (None, 60, 60, 64)        0
batch_normalization_1[0][0]
__________________________________________________________________________________


__________________
separable_conv2d_1 (SeparableCo (None, 60, 60, 128)  17664
tf.nn.relu_2[0][0]
__________________________________________________________________________________
__________________
batch_normalization_4 (BatchNor (None, 60, 60, 128)  512
separable_conv2d_1[0][0]
__________________________________________________________________________________
__________________
conv2d_2 (Conv2D)            (None, 30, 30, 128)  8320
tf.nn.relu_1[0][0]
__________________________________________________________________________________
__________________
max_pooling2d (MaxPooling2D)    (None, 30, 30, 128)  0
batch_normalization_4[0][0]
__________________________________________________________________________________
__________________
batch_normalization_2 (BatchNor (None, 30, 30, 128)  512         conv2d_2[0][0]
__________________________________________________________________________________
__________________
first (Concatenate)          (None, 30, 30, 256)  0
max_pooling2d[0][0]
batch_normalization_2[0][0]
__________________________________________________________________________________
__________________
separable_conv2d_2 (SeparableCo (None, 30, 30, 128)  35200       first[0][0]
__________________________________________________________________________________
__________________
batch_normalization_6 (BatchNor (None, 30, 30, 128)  512
separable_conv2d_2[0][0]
__________________________________________________________________________________
```

```
__________________
tf.nn.relu_3 (TFOpLambda)       (None, 30, 30, 128)  0
batch_normalization_6[0][0]
__________________________________________________________________________________
__________________
separable_conv2d_3 (SeparableCo (None, 30, 30, 128)  17664
tf.nn.relu_3[0][0]
__________________________________________________________________________________
__________________
batch_normalization_7 (BatchNor (None, 30, 30, 128)  512
separable_conv2d_3[0][0]
__________________________________________________________________________________
__________________
conv2d_3 (Conv2D)               (None, 15, 15, 128)  32896       first[0][0]
__________________________________________________________________________________
__________________
```



```
max_pooling2d_1 (MaxPooling2D)  (None, 15, 15, 128)  0
batch_normalization_7[0][0]
__________________________________________________________________________________
__________________
batch_normalization_5 (BatchNor (None, 15, 15, 128)  512         conv2d_3[0][0]
__________________________________________________________________________________
__________________
second (Concatenate)            (None, 15, 15, 256)  0
max_pooling2d_1[0][0]
batch_normalization_5[0][0]
__________________________________________________________________________________
__________________
separable_conv2d_4 (SeparableCo (None, 15, 15, 256)  68096       second[0][0]
__________________________________________________________________________________
__________________
batch_normalization_9 (BatchNor (None, 15, 15, 256)  1024
separable_conv2d_4[0][0]
__________________________________________________________________________________
__________________
tf.nn.relu_4 (TFOpLambda)       (None, 15, 15, 256)  0
batch_normalization_9[0][0]
__________________________________________________________________________________
__________________
separable_conv2d_5 (SeparableCo (None, 15, 15, 256)  68096
tf.nn.relu_4[0][0]
__________________________________________________________________________________
__________________
conv2d_4 (Conv2D)               (None, 15, 15, 128)  32896       second[0][0]
__________________________________________________________________________________
__________________
batch_normalization_10 (BatchNo (None, 15, 15, 256)  1024
separable_conv2d_5[0][0]
```

```
__________________________________________________________________________________________________
batch_normalization_8 (BatchNor (None, 15, 15, 128)  512         conv2d_4[0][0]
__________________________________________________________________________________________________
max_pooling2d_3 (MaxPooling2D)  (None, 7, 7, 256)    0
batch_normalization_10[0][0]
__________________________________________________________________________________________________
max_pooling2d_2 (MaxPooling2D)  (None, 7, 7, 128)    0
batch_normalization_8[0][0]
__________________________________________________________________________________________________
third (Concatenate)             (None, 7, 7, 384)    0
max_pooling2d_3[0][0]
max_pooling2d_2[0][0]
```



```
__________________________________________________________________________________________________
tf.nn.relu_5 (TFOpLambda)       (None, 7, 7, 384)    0           third[0][0]
__________________________________________________________________________________________________
separable_conv2d_6 (SeparableCo (None, 7, 7, 512)    200576
tf.nn.relu_5[0][0]
__________________________________________________________________________________________________
batch_normalization_11 (BatchNo (None, 7, 7, 512)    2048
separable_conv2d_6[0][0]
__________________________________________________________________________________________________
tf.nn.relu_6 (TFOpLambda)       (None, 7, 7, 512)    0
batch_normalization_11[0][0]
__________________________________________________________________________________________________
separable_conv2d_7 (SeparableCo (None, 7, 7, 512)    267264
tf.nn.relu_6[0][0]
__________________________________________________________________________________________________
batch_normalization_12 (BatchNo (None, 7, 7, 512)    2048
separable_conv2d_7[0][0]
__________________________________________________________________________________________________
tf.nn.relu_7 (TFOpLambda)       (None, 7, 7, 512)    0
batch_normalization_12[0][0]
__________________________________________________________________________________________________
separable_conv2d_8 (SeparableCo (None, 7, 7, 512)    267264
tf.nn.relu_7[0][0]
__________________________________________________________________________________________________
```

```
__________________
fourth (Concatenate)            (None, 7, 7, 896)    0
batch_normalization_13[0][0]
                                                                 third[0][0]
__________________________________________________________________________________
__________________
tf.nn.relu_8 (TFOpLambda)       (None, 7, 7, 896)    0           fourth[0][0]
__________________________________________________________________________________
__________________
separable_conv2d_9 (SeparableCo (None, 7, 7, 512)    467328
tf.nn.relu_8[0][0]
__________________________________________________________________________________
```



```
__________________
batch_normalization_14 (BatchNo (None, 7, 7, 512)    2048
separable_conv2d_9[0][0]
__________________________________________________________________________________
__________________
tf.nn.relu_9 (TFOpLambda)       (None, 7, 7, 512)    0
batch_normalization_14[0][0]
__________________________________________________________________________________
__________________
separable_conv2d_10 (SeparableC (None, 7, 7, 256)    135936
tf.nn.relu_9[0][0]
__________________________________________________________________________________
__________________
batch_normalization_15 (BatchNo (None, 7, 7, 256)    1024
separable_conv2d_10[0][0]
__________________________________________________________________________________
__________________
tf.nn.relu_10 (TFOpLambda)      (None, 7, 7, 256)    0
batch_normalization_15[0][0]
__________________________________________________________________________________
__________________
separable_conv2d_11 (SeparableC (None, 7, 7, 256)    68096
tf.nn.relu_10[0][0]
__________________________________________________________________________________
__________________
batch_normalization_16 (BatchNo (None, 7, 7, 256)    1024
separable_conv2d_11[0][0]
__________________________________________________________________________________
__________________
fifth (Concatenate)             (None, 7, 7, 1152)   0
batch_normalization_16[0][0]
                                                                 fourth[0][0]
__________________________________________________________________________________
__________________
tf.nn.relu_11 (TFOpLambda)      (None, 7, 7, 1152)   0           fifth[0][0]
__________________________________________________________________________________
__________________
separable_conv2d_12 (SeparableC (None, 7, 7, 512)    600704
tf.nn.relu_11[0][0]
```

```
------------------
batch_normalization_18 (BatchNo (None, 7, 7, 512)    2048
separable_conv2d_12[0][0]
------------------------------------------------------------------------------
------------------
tf.nn.relu_12 (TFOpLambda)      (None, 7, 7, 512)    0
batch_normalization_18[0][0]
------------------------------------------------------------------------------
```



```
------------------
separable_conv2d_13 (SeparableC (None, 7, 7, 512)    267264
tf.nn.relu_12[0][0]
------------------------------------------------------------------------------
------------------
conv2d_5 (Conv2D)               (None, 7, 7, 256)    295168      fifth[0][0]
------------------------------------------------------------------------------
------------------
batch_normalization_19 (BatchNo (None, 7, 7, 512)    2048
separable_conv2d_13[0][0]
------------------------------------------------------------------------------
------------------
batch_normalization_17 (BatchNo (None, 7, 7, 256)    1024        conv2d_5[0][0]
------------------------------------------------------------------------------
------------------
sixth (Concatenate)             (None, 7, 7, 768)    0
batch_normalization_19[0][0]
batch_normalization_17[0][0]
------------------------------------------------------------------------------
------------------
separable_conv2d_14 (SeparableC (None, 7, 7, 256)    203776      sixth[0][0]
------------------------------------------------------------------------------
------------------
batch_normalization_20 (BatchNo (None, 7, 7, 256)    1024
separable_conv2d_14[0][0]
------------------------------------------------------------------------------
------------------
tf.nn.relu_13 (TFOpLambda)      (None, 7, 7, 256)    0
batch_normalization_20[0][0]
------------------------------------------------------------------------------
------------------
separable_conv2d_15 (SeparableC (None, 7, 7, 256)    68096
tf.nn.relu_13[0][0]
------------------------------------------------------------------------------
------------------
batch_normalization_21 (BatchNo (None, 7, 7, 256)    1024
separable_conv2d_15[0][0]
------------------------------------------------------------------------------
------------------
tf.nn.relu_14 (TFOpLambda)      (None, 7, 7, 256)    0
batch_normalization_21[0][0]
```

```
global_average_pooling2d (Globa (None, 256)          0
tf.nn.relu_14[0][0]
__________________________________________________________________________________
__________________
dense (Dense)                   (None, 128)          32896
```

11

```
global_average_pooling2d[0][0]
__________________________________________________________________________________
__________________
dropout (Dropout)               (None, 128)          0           dense[0][0]
__________________________________________________________________________________
__________________
dense_1 (Dense)                 (None, 64)           8256        dropout[0][0]
__________________________________________________________________________________
__________________
dropout_1 (Dropout)             (None, 64)           0           dense_1[0][0]
__________________________________________________________________________________
__________________
dense_2 (Dense)                 (None, 7)            455         dropout_1[0][0]
__________________________________________________________________________________
```

4.3 SOFTWARE AND HARDWARE REQUIREMENTS

The processor, RAM, platform, operating system, and the language used are specified in table below. The proposed algorithm is implemented using the following experimental setup:

Processor Required	Intel Core i5-1035G1 CPU @ 1.00GHz
RAM	8 GB
Platform	Kaggle
Operating System	Windows 10
Language Used	Python3

Table 4.3.1: System Configuration

These are the model parameters and the corresponding values used in the deep neural network in the table 4.3.2.

Model Parameters	**Values**
Activation	ReLU
Learning Rate(lr)	0.0001
Epoch	64
Optimizer	Adam
Loss function used(lf)	Categorical cross entropy
Batch size	64

Table 4.3.2: Model Parameters

CHAPTER 5

DATASET

The Facial Expression Recognition (FER2013) dataset is a sizable collection of 48x48 pixel grayscale facial photos that was unveiled during the ICML 2013 Challenges in Representation Learning. One of seven categories has been assigned to each image in this dataset. Each category has a separate human facial expression associated with it, giving computational analysis access to a wide range of emotional circumstances.

0 for Angry

1 for Disgust

2 for Fear

3 for Happy

4 for Sad

5 for Surprise

6 for Neutral

The dataset is split up into three subsets: a training set consists of 28709 examples , while both the public and private test sets contain 3,589 images each. This division serves two main purposes: preventing overfitting and establishing a fair basis for comparing different models. This dataset has been tremendously used for training machine learning and deep learning models for emotion recognition from facial expressions. The training and testing data is split up in the ratio of 80:20.

CHAPTER 6

RESULTS AND ANALYSIS

It's clear that Convolutional Neural Networks (CNNs) have revolutionized the field of image analytics. They have become an indispensable tool in addressing image-related problems and are integrated into numerous architectural designs like ResNet and Google Net, which have accomplished remarkable precision in image categorization tasks. However, like most things, they do have their drawbacks, one significant one being the substantial time they require to train on extensive datasets. This is where Separable Convolutions come into play, offering a solution to this particular challenge.

The snapshots for each steps in the projects are:

1) Training the model:

Fig 6.1 shows the accuracy and loss corresponding to each epoch. The accuracy of our model is improving slowly with each epoch and also the loss is being slowly decreased.

Model-1

```
Epoch 1/60
359/359 [==============================] - 452s 1s/step - loss: 1.8209 -
accuracy: 0.2640 - val_loss: 1.9006 - val_accuracy: 0.1739

Epoch 00001: val_accuracy improved from -inf to 0.17388, saving model to
best_model.h5
Epoch 2/60
359/359 [==============================] - 422s 1s/step - loss: 1.6266 -
accuracy: 0.3624 - val_loss: 1.6015 - val_accuracy: 0.3959

Epoch 00002: val_accuracy improved from 0.17388 to 0.39595, saving model to
best_model.h5
Epoch 3/60
359/359 [==============================] - 423s 1s/step - loss: 1.5082 -
accuracy: 0.4173 - val_loss: 1.4819 - val_accuracy: 0.4309

Epoch 00003: val_accuracy improved from 0.39595 to 0.43087, saving model to
best_model.h5
Epoch 4/60
359/359 [==============================] - 427s 1s/step - loss: 1.4206 -
accuracy: 0.4598 - val_loss: 1.4435 - val_accuracy: 0.4344

Epoch 00004: val_accuracy improved from 0.43087 to 0.43436, saving model to
best_model.h5
Epoch 5/60
359/359 [==============================] - 423s 1s/step - loss: 1.3585 -
accuracy: 0.4886 - val_loss: 1.2796 - val_accuracy: 0.5244

Epoch 00005: val_accuracy improved from 0.43436 to 0.52444, saving model to
best_model.h5
Epoch 6/60
```

```
Epoch 00034: val_accuracy did not improve from 0.64385
Epoch 35/60
359/359 [==============================] - 424s 1s/step - loss: 0.7913 -
accuracy: 0.7167 - val_loss: 1.2166 - val_accuracy: 0.5950

Epoch 00035: val_accuracy did not improve from 0.64385
Epoch 36/60
359/359 [==============================] - 421s 1s/step - loss: 0.7840 -
accuracy: 0.7226 - val_loss: 1.0722 - val_accuracy: 0.6229
```

Fig 6.1 The epochs of the proposed model

2) Accuracy graphs:

Fig 6.2.1 shows the validation as well as the training accuracy of the for the proposed model. It is evident from the graph 6.2.2 that the accuracies are increasing at some constant rate.

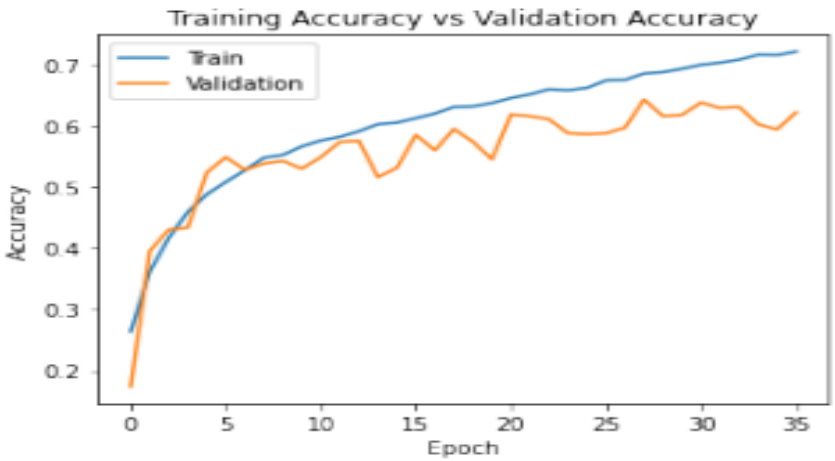

Fig 6.2.1 epochs vs accuracy for the proposed model

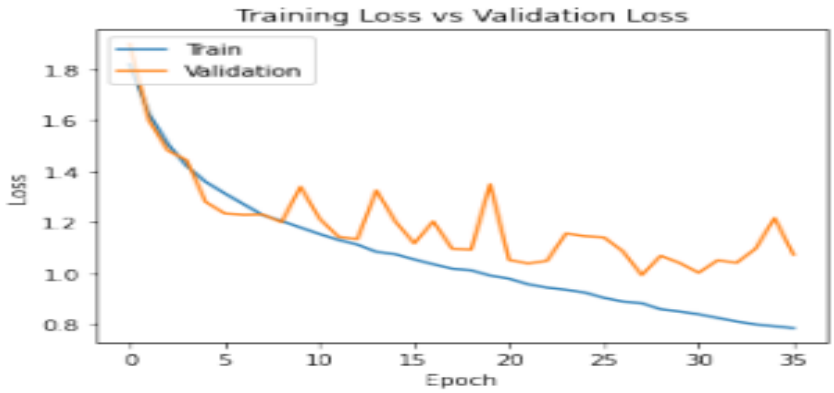

Fig 6.2.2 epochs vs loss value for the proposed model

3) Confusion matrix:

The confusion matrix draws a table which represent the predicted label vs actual values stating which are correctly predicted and which are not correctly predicted.

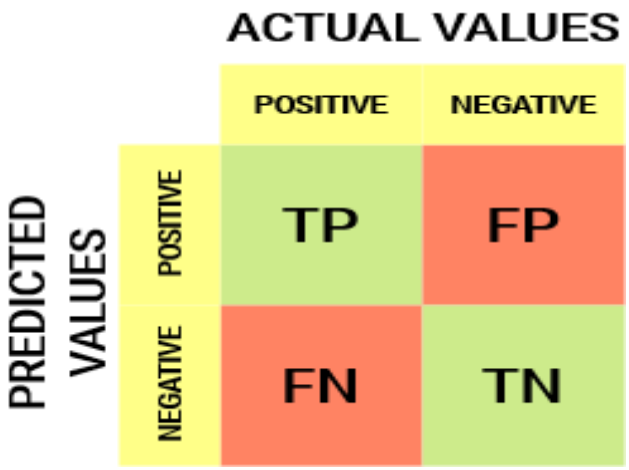

Fig6.4.1 The confusion matrix parameters

The confusion matrix after training the model is displayed in Fig 6.4.2. It's basically a simple 2D matrix which is also used to calculate the accuracy of prediction of output.

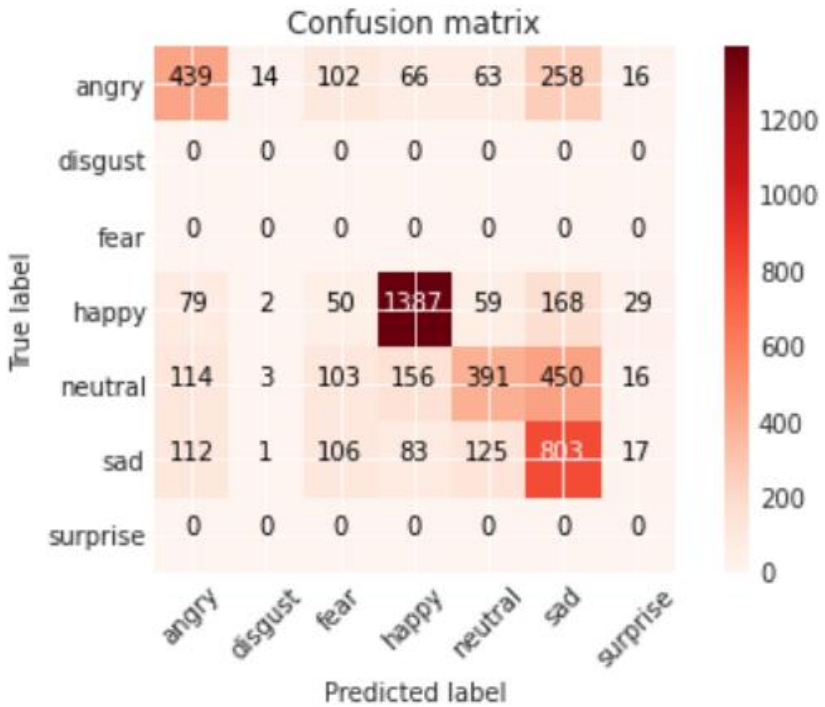

Fig 6.4.2 confusion matrix after training the model

4) Report Test:

The precision and recall can be calculated from the confusion matrix. The

calculation of precision and recall is shown in Fig 6.4.1

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

The precision in the context of a classification problem in machine learning is defined as:

Precision = TP / (TP + FP)

Here:

TP represents True Positives: the number of positive cases correctly identified as positive.

FP represents False Positives: the number of negative cases incorrectly identified as positive.

Precision gives us a measure of the accuracy of positive predictions.

The recall in the context of a classification problem is a measure that shows the completeness of the true positive predictions. In other words, it demonstrates how well a model can identify the positive cases. It can be calculated using the following formula:

Recall = True Positives / (True Positives + False Negatives)

In this formula:

True Positives (TP) is the number of positive cases that the model correctly identified as positive.

False Negatives (FN) is the number of positive cases that the model incorrectly identified as negative.

The precision, recall and the F1-score of the model is given in the below Fig 6.4.1.

	precision	recall	f1-score	support
angry	0.10	0.14	0.12	191
disgust	0.00	0.00	0.00	22
fear	0.10	0.07	0.09	204
happy	0.22	0.21	0.21	354
neutral	0.15	0.14	0.14	246
sad	0.17	0.20	0.18	249
surprise	0.12	0.11	0.12	166
accuracy			0.15	1432
macro avg	0.12	0.12	0.12	1432
weighted avg	0.15	0.15	0.15	1432

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

Our model achieves a testing accuracy of 72.26%. The wrong classification occurs due to highly common images in training dataset, especially in fear and neutral, sad and neutral also because of skew classes such as disgust class in FER-2013 dataset which contains very less images in that particular training dataset.

7.2 FUTURE WORK

This research could be expanded to identify changes in emotion utilising additional, unstudied images and video clips. These could then be used in situations that happen in real time, such as feedback analysis and others. The system could also be connected to other electrical components for more dynamic control.

It's important to remember that incorrect classification of some emotions, including fear and sadness, can affect the precision of emotion detection. The model learned indistinguishable features for several categories since the training dataset currently being utilised comprises photos that are quite similar to one another. The accuracy of categorisation within these groupings has been impacted by this.

Future work will involve locating and using higher quality datasets to help reduce this problem. We intend to increase the precision of emotion classification and enhance the model's overall performance by training it with these fresh data.

REFERENCES

[1]. Ekman P. Basic emotions. Handbook Cognit Emot. 1999;98(45-60):16.

[2]. Plutchik R. A general psychoevolutionary theory of emotion. Amsterdam, Netherlands: Elsevier; 1980 (pp. 3–33).

[3]. Russell JA. A circumplex model of affect. J Pers Soc Psychol. 1980;39(6):1161.

[4]. Mehrabian, A. (1996) Pleasure-Arousal-Dominance: A General Framework for Describing and Measuring Individual Differences in Temperament. Current Psychology, 14, 261-292.

[5]. Pang, B. and Lee, L. (2008) Opinion Mining and Sentiment Analysis. Foundations and Trends® in Information Retrieval, 2, 1-135.

[6]. Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. "Efficient estimation of word representations in vector space." *arXiv preprint arXiv:1301.3781* (2013).

[7]. Trigeorgis, George, Fabien Ringeval, Raymond Brueckner, Erik Marchi, Mihalis A. Nicolaou, Björn Schuller, and Stefanos Zafeiriou. "Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network." In *2016 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 5200-5204. IEEE, 2016.

[8]. Pantic, Maja, and Marian Stewart Bartlett. *Machine analysis of facial expressions*. INTECH Open Access Publisher, 2007.

[9]. Lopes, Nuno, André Silva, Salik Ram Khanal, Arsênio Reis, João Barroso, Vitor Filipe, and Jaime Sampaio. "Facial emotion recognition in the elderly using a SVM classifier." In *2018 2nd International Conference on Technology and Innovation in Sports, Health and Wellbeing (TISHW)*, pp. 1-5. IEEE, 2018.

[10].Baltrušaitis, Tadas, Chaitanya Ahuja, and Louis-Philippe Morency. "Multimodal machine learning: A survey and taxonomy." *IEEE transactions on pattern analysis and machine intelligence* 41, no. 2 (2018): 423-443.

[11]. Atrey, Pradeep K., M. Anwar Hossain, Abdulmotaleb El Saddik, and Mohan S. Kankanhalli. "Multimodal fusion for multimedia analysis: a survey." *Multimedia systems* 16 (2010): 345-379.

[12]. Kreibig, Sylvia D. "Autonomic nervous system activity in emotion: A review." *Biological psychology* 84, no. 3 (2010): 394-421.

[13]. Roy, Jean-Claude, Wolfram Boucsein, Don C. Fowles, and John Gruzelier, eds. *Progress in electrodermal research*. Vol. 249. Springer Science & Business Media, 2012.

[14]. Lindquist, Kristen A., Tor D. Wager, Hedy Kober, Eliza Bliss-Moreau, and Lisa Feldman Barrett. "The brain basis of emotion: a meta-analytic review." *Behavioral and brain sciences* 35, no. 3 (2012): 121-143.

[15]. Mühl, Christian, Brendan Allison, Anton Nijholt, and Guillaume Chanel. "A survey of affective brain computer interfaces: principles, state-of-the-art, and challenges." *Brain-Computer Interfaces* 1, no. 2 (2014): 66-84.

[16]. Ekman, Paul, and Wallace V. Friesen. "Facial action coding system." *Environmental Psychology & Nonverbal Behavior* (1978).

[17]. Zheng, Wenming, Hao Tang, Zhouchen Lin, and Thomas S. Huang. "Emotion recognition from arbitrary view facial images." In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part VI 11*, pp. 490-503. Springer Berlin Heidelberg, 2010.

[18]. Lucey, Patrick, Jeffrey F. Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression." In *2010 ieee computer society conference on computer vision and pattern recognition-workshops*, pp. 94-101. IEEE, 2010.

[19]. Cootes, Timothy F., Gareth J. Edwards, and Christopher J. Taylor. "Active appearance models." *IEEE Transactions on pattern analysis and machine intelligence* 23, no. 6 (2001): 681-685.

[20]. Banse, Rainer, and Klaus R. Scherer. "Acoustic profiles in vocal emotion expression." *Journal of personality and social psychology* 70, no. 3 (1996): 614.

[21]. Schuller, Björn, Gerhard Rigoll, and Manfred Lang. "Hidden Markov model-based speech emotion recognition." In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03).*, vol. 2, pp. II-1. Ieee, 2003.

[22]. Dael, Nele, Marcello Mortillaro, and Klaus R. Scherer. "Emotion expression in body action and posture." *Emotion* 12, no. 5 (2012): 1085.

[23]. Biswas, Kanad K., and Saurav Kumar Basu. "Gesture recognition using microsoft kinect®." In *The 5th international conference on automation, robotics and applications*, pp. 100-103. IEEE, 2011.

[24]. Lin, Yuan-Pin, Tzyy-Ping Jung, and Yijun Wang and Julie Onton. "Toward Affective Brain–Computer Interface: Fundamentals and Analysis of EEG-Based Emotion Classification." *Emotion Recognition: A Pattern Analysis Approach* (2015): 315-341.

[25]. Goodfellow I, Bengio Y, Courville A (2016) Deep learning. MIT Press, Cambridge

PAPER NAME

aarti thesis.pdf

AUTHOR

aarti thesis

WORD COUNT

8126 Words

CHARACTER COUNT

46525 Characters

PAGE COUNT

37 Pages

FILE SIZE

1.7MB

SUBMISSION DATE

May 29, 2023 7:29 PM GMT+5:30

REPORT DATE

May 29, 2023 7:30 PM GMT+5:30

8% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 4% Internet database
- 3% Publications database
- Crossref database
- Crossref Posted Content database
- 6% Submitted Works database

Excluded from Similarity Report

- Bibliographic material
- Small Matches (Less then 9 words)