Behavior Change Analysis Due to Violent Video Gaming Using Deep Learning Models

by

Akhlak Ur Rahman
20101422
Fahad Khan Raj
20101250
Monthasir Delwar Afnan
20101247
Rakib Hasan Rahad
20101010
Md. Samir Uddin Ahmed
20101520

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
February 2024

Declaration

It is hereby declared that

1. Thesis submitted is original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Akhlak Ur Rahman
20101422

Fahad Khan Raj
20101250

Rakib Hasan Rahad
20101010

Monthasir Delwar Afnan
20101247

Md. Samir Uddin Ahmed
20101520

Approval

The thesis titled "Behavior Change Analysis Due to Violent Video Gaming Using Deep Learning Models" submitted by

1. Akhlak Ur Rahman (20101422)
2. Fahad Khan Raj (20101250)
3. Monthasir Delwar Afnan (20101247)
4. Rakib Hasan Rahad (20101010)
5. Md. Samir Uddin Ahmed (20101520)

Of Fall, 2023 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on February 4, 2024.

Examining Committee:

Supervisor:
(Member)

Dr. Md. Khalilur Rhaman
Professor
Department of Computer Science and Engineering
Brac University

Co-supervisor:
(Member)

Sayantan Roy Arko
Research Assistant
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Associate Professor and Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

In the 21st century, technology has advanced at such a wondrous rate that people resort to this developed medium to carry out most of their tasks, from household chores to daily necessities and almost everything else. People of almost all ages resort to technology to pass their leisure time, and gaming is prevalent among everyone. However, this gaming behavior, whether in online, multiplayer, or single-player mode, has significant behavioral changes, both negatively and positively. A significant amount of research was conducted from the early stage of video gaming. After the development of machine learning and deep learning, these techniques were used to predict emotions. Employing a unique approach, a vast amount of YouTube videos were collected from different online gaming streamers and then image and audio datasets comprising hundreds of those videos immersed in these intense gaming sessions were created. By using Facial Expression Recognition (FER) and Speech Emotion Recognition (SER) techniques, an approach was made to find a pattern of behavior change during gaming and over time. For FER, various models were used. Also, different models for SER were used. Some of the best models were used to perform prediction on the image and audio data that we had extracted from the videos. This research contributes significant insights into a player's emotional change while playing video games.

Keywords: Video games, Behavior analysis, Facial Expression Recognition, FER, Speech Emotion Recognition, SER, Machine Learning Algorithms, Deep Learning Algorithms

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background Information

Who doesn't like to spend their free time with friends playing video games? The leisurely pursuit of video gaming has evolved into a burgeoning career path, catalyzed by the seamless accessibility and technological advancements in the gaming industry. With 3.09 billion active video gamers around the world in 2023, the global online gaming market made about US$26.14 billion. This is an increase of 9.8 percent over the previous year [1][2]. Recent estimates place the percentage of people who play video games online in 2023 at about 40 percent. As fewer adults aged 16 and above spend their free time gaming, the newest online gaming research suggests that public opinion on gaming as a whole has evolved [3]. So, the young generation and professional gamers spend a lot of time playing online games. So, the one aspect of this fastest-growing sector that caught our attention was the behavioral changes due to long-term gaming. Researchers research on this topic from the long time and found both positive and negative impact of video game. Nowadays, because of the vast gaming industry, it become more debatable whether gaming is good or bad for the behavioral change. Researches showed the negative impact of video game. Video game addiction has been linked to negative traits including frustration, anger, restlessness, and impulsive [4]. Because gaming takes time away from friends and family, it may lead to isolation, rudeness, and poor self-esteem [5]. If a kid spends a lot of time playing video games at school, it will negatively affect his or her academic performance [6]. In fact, a recent research confirms that the setting of video games might affect students' performance in the classroom [7]. Adrenergic stimulation associated with the emotionally charged video game setting has been linked to cardiac arrhythmia in adolescents, according to several recent research [8]. On the other hand, there are researches that found video gaming to connect with positive impact. [9] has divide many popular games in sections and find out their positive impact on player. According to [10], gaming can also be beneficial similar to the courses taken aim to same skill. Besides, [11] categorised the gaming benefits in four sections and their impact on player. From the previous research it is unclear that whether the gaming impacts positively or negatively. In our study, we aimed to find the true impacts of video gaming with deep learning models. Besides, as the prevalence of gaming grows, so does the need of constant monitoring of any consequent changes in player behavior. In response to this need, the creation of an easy-to-use system for monitoring a player's actions over time becomes urgent. With the components

we used in our research, we also develop a model structure, which will be able to lead to the development of monitoring system in multi-platform.

1.2 Problem Statement

In this day and age, when playing video games has graduated from the realm of simple pastime to that of a prominent cultural phenomenon, there is a growing concern over the substantial behavioral changes that can occur in people who play video games for extended periods of time. The widespread occurrence of addiction to video games and the negative characteristics that are often highlighted as connected with it, such as frustration, anger, restlessness, and impulsiveness, constitute a significant problem. Because of that, the parents and seniors often blame the video game for all of the negative issue. Thus, we have aimed to find the truth behind of the video gaming impact.

Chapter 2

Research Objectives

The primary goal of this research is to comprehensively investigate the behavioral changes resulting from the engagement in video games. We have outlined specific research objectives:
Dataset Preparation: To curate two distinct datasets, one for facial expression recognition (FER) and another for speech emotion recognition (SER).
Model Evaluation: To rigorously assess the results and accuracy of models in the context of speech emotion recognition (SER) and facial expression recognition (FER).
Pattern Analysis: To identify and elucidate discernible patterns within the behavior of video gamers during online game sessions.
Without using any sophisticated medical equipment or advance technology, our research embraces a pragmatic approach to analyze the behavioral shifts of video gamers. As Webcam and microphone is an essential tool for a gamer, we harnessed the good amount of data that can be generated from these two tools. The purpose of our paper is to find out the positive and negative changes in a player's behavior during online video game sessions using speech emotion recognition (SER) and facial expression recognition (FER) leveraging deep learning and machine learning models like . The insights gleaned from our study hold the potential to empower gamers themselves, as well as parents and healthcare professionals, in monitoring and comprehending the behavioral dynamics of specific individuals.

2.1 Work Plan

Finding behavior change is a complex process. For this process, we had to work with two main structure to detect emotion. Facial expression recognition and speech emotion recognition were use as the main structure to figure out the emotion for each video. After collecting several months of video, we have detect a pattern of changing emotion. For our working procedure, first, we figured out which approaches would be better for us by studying research papers. Then, we have collected game recordings from various sources. We have divided our dataset to prepare work on two sectors. Facial expression recognition and speech expression recognition. For FER, we have convert the videos into images at 4 second interval. For SER, we also have convert the videos into audios and segmented them into 4 second interval. Then, we have annotated our images. After that, we have developed deep learning models and find the best model to predict our data. After the prediction, we find the pattern of emotion.

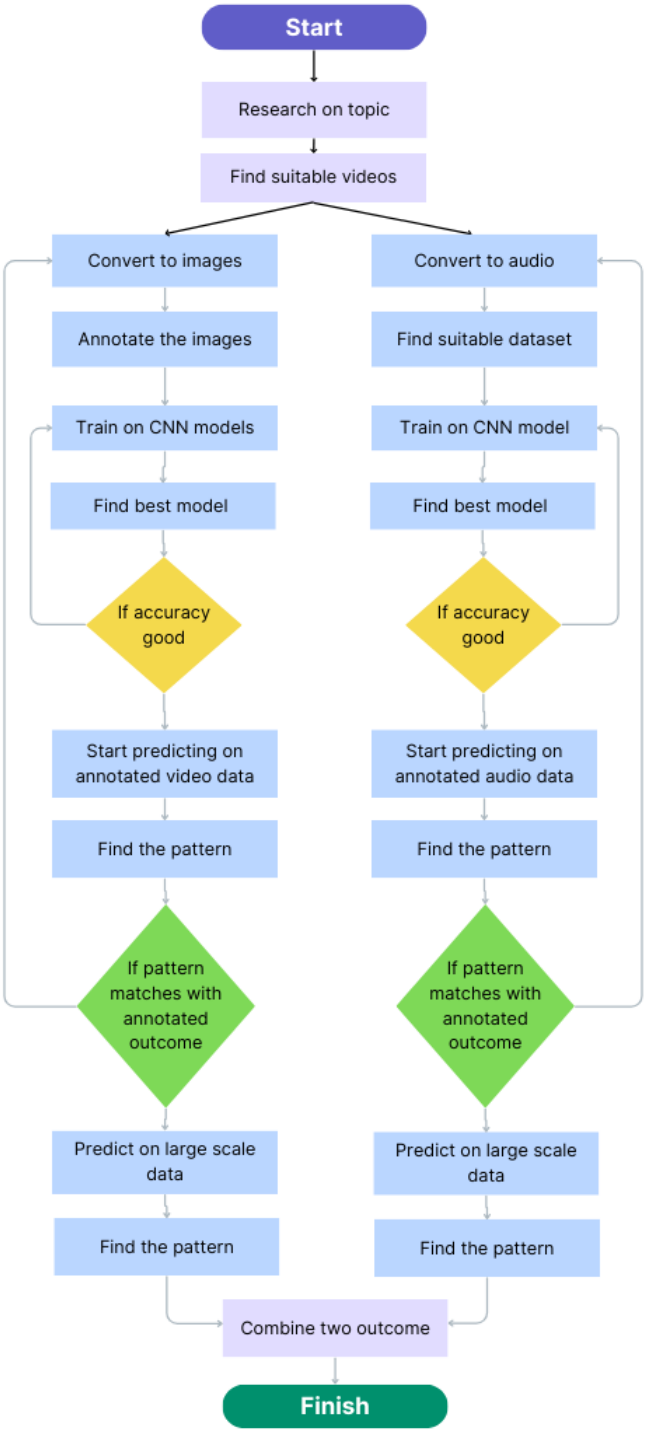


Figure 2.1: Flowchart of our work plan

Chapter 3

Literature Review

There has been much research published in recent years about the impact of video games on human life. Recent studies show that video games impact human life negatively and have a big reason for increasing aggression [12]. However, a few studies also tried to show the positive effects of playing video games. Most of the recently most played games contain violence. Thus, in this research, we tried to find out about violent gaming. The next subsection is divided into two parts. In the next one, we tried to show the studies on video gaming and behavior from psychological aspects. After that, we tried to show some research papers that helped us do our research from a machine-learning perspective.

3.1 Video games and behavior from psychological aspects

3.1.1 Positive impact of video games

Few studies have shown the benefits of playing video games. [9] have done a study on the benefits of video games. In a figure they categorized the games into a graph plot. In positive x-axis, Social, In negative x-axis, Non-social, in positive y-axis, Complex, in negative y-axis, they took Simple as parameter. Games such as MMORPG, multiplayer shooter games such as Halo 4, sports games such as fifa, strategy games such as Starcraft II, these were put on Complex-Social game zone. Games such as RPG games such as Final Fantasy XIII-2, solo shooter games such as Halo 4, these were put on Complex-Non Social game zone. Games such as Racing games such as Need for speed, Social Media games such as Farmville, party games such as Mario Party 9, these were put on Simple-Social game zones. Lastly, games like puzzle games such as Bejeweled, Platformer such as Super Mario Bros, these were put into Non social-Simple game zone. From these categorisation, that leads them to create another category of the benefits type according to the games type. The authors segment their research into four categories. cognitive, motivational, emotional, and social advantages. According to [10], the benefits of taking formal courses aimed at strengthening the same skills are comparable to those gained from playing commercially accessible shooter video games. In the study, [9] came to the conclusion that it might be possible that video game environments cultivate a persistent, optimistic

motivational style.

Here, [11] shows the benefits of gaming and its impact on games. For cognitive skills, shooter, strategic, role-playing game type were selected. They had several impacts on players. Such as it improves the attention allocation more faster and accurately in the brain. Besides that, it improves mental rotation ability, memorization, problem solving skills, analytical, collaboration skills and many more. [11] categorized strategic games for motivational benefits. The impact on players is also significant. It can improve resilience when they face failure. It increases intelligence, leads to success in academics. For emotional benefits, puzzles and playing games were selected. It can enhance positive feelings, and work as a source of motivation for the player. Besides, it can build social relationships, promote relaxation and also help the players to deal with anxiety and frustration. Lastly, social benefits can be achieved from cooperative based, prosocial and role playing multiplayer games according to [11]. It can have great impacts such as reducing the feelings of hostility, decreasing aggressive cognition, increasing cooperative behavior, increasing group organization etc.
Besides, this research also tried to show that gaming can be beneficial for both emotional and social purposes. However, their study is based on many decades-old papers, and video gaming has changed much in recent years. However, research has shown the importance of video games.

3.1.2 Negative impact of video games

Numerous recent studies have been done on the negative impacts of video games. For example, [12] have done research on 300 children (mean age = 6.38, SD = 0.25) about the violent video game effect on aggressive behavior. The kids were given the option of playing violent or non-violent games. After playing, they collected the necessary data and found the result. The findings showed a big impact of video games. The children showed a higher CRT score than nonviolent video game players. Here, a competitive reaction time (CRT) is universally acknowledged as a valid indicator of aggression [13]. This result shows the positive relationship between aggressive behavior and video game play. Furthermore, based on mediation analysis, this study also discovered aggressive behavior.

Another study was done by [14] for a decade on playing Grand Theft Auto, a popular violent video game. This study included 500 adolescent participants aged 10 to 13 years old. The data was collected in 11 waves. The result shows that across 10 years, VVG showed a quadratic pattern. Three categories of participants have been found: the high initial violence group (4 percent), moderate violence group (23 percent), and low increasers group (73 percent) [14].
Here is a curvilinear pattern between high initial violence and moderate groups. This gets more in low incresers. Again, the high initial group was depressed in the initial wave. There was no difference in the last wave in prosocial conduct. From the total analysis, they have discovered the moderate group is the most aggressive behavior [14]. This study has shown how video gaming negatively affects children's behavior. In another study, [15] researched 998 participants (mean age

= 36.8 years, SD = 11.2). It was self-reported data collected through questionnaires.

For participants and their friends, aggression was related to VVE. Furthermore, there was a strong correlation between the VVE of the participant and their friends, as well as the level of their reported aggression. Moreover, the participant's aggression and the friends' VVE were also strongly correlated.

3.1.3 Effects of Video Games on Cardiovascular Health

The impacts of video games on the cardiovascular system are a topic of research that is currently being researched in addition to the general link between video games and the health of the heart.

It is no secret that video games have been linked to a variety of negative health

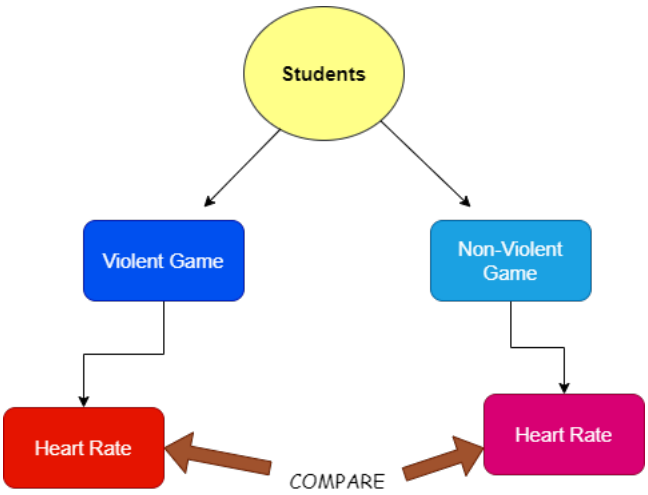


Figure 3.1: Flowchart of heart-rate comparison

effects just as positive effects. A study has shown that playing video games might affect one's heart rate and blood pressure in a negative way [16]. Midwestern university's 245 students (117 girls and 128 boys) participated in a study for academic credit. Participants averaged 19.07 years old (SD D 2.53) and first-year undergraduates were 67.8% and Caucasian (89.4%). Weekly video game play averaged 6.88 (SD D 7.89) hours (D 0–20 hours).

The study's findings show a connection between playing violent video games and recalling aggressive behavioral patterns[16]. It showed that the Violent, N64 condition had more aggressive script values than the Nonviolent, N64 condition.
While the experiment was interesting, it failed to provide any conclusive results because of concerns that the N64 versions of the games were more challenging. This is because a video game rating sheet was not used in the current study [16].
Furthermore, several other experiments and studies were conducted to test out how a video game can impact one's cardiovascular health. In another study, throughout most of the games, the usual systolic Blood Pressure seems to be much higher than it was either before as well as after. And the difference between the systolic blood pressure of novice and expert players was even more noticeable [17].

Although these were relatively slight changes, they, however, might have significant consequences for those with various heart diseases. In an experiment, they studied 23 guys who are 22–28 years old. No one knew they had cardiovascular concerns, save one with mild blood pressure increase in the past. Each volunteer participant was told of the study's design and goals. Investigators supplied machine coins. The video game is named Berserk. Arteriocorder (model 1508) determined experimentally every participant's blood pressure and heart rate. This system digitally displays systolic, diastolic, and heart rates while inflating a blood pressure cuff every 30 seconds. Each person's before-game blood pressure and pulse rate were monitored at thirty-second intervals. During video game play, the systolic blood pressure of 23 of them dramatically increased in this study. In addition, heart rate rose significantly as well [18].

3.2 Behavior change detection

3.2.1 Facial Expression Recognition

The effects of playing video games on one's behavior are far-reaching. Several methods exist for identifying the transitions in behavior. Among all these methods identifying a person's facial emotion is a useful method. Facial emotion recognition technology is a useful tool. According to the results of a study, video game playing is linked to alterations in brain activity and connection, which may affect one's emotional state and facial expressions [19]. A group of researchers led by [20] took a look at how exposure to violent media affected people's ability to understand emotions conveyed through facial expressions. Typically, The happy-face advantage [e.g., [21]] describes the tendency for people to recognize and label positive than negative facial expressions. Compared to emotional recognition research, emotional expression research is minimal.

In an experiment, a sample size of 23 ED patients (11 AN and 12 BN) and 11 HCs were taken. This program for face emotional identification consists of two major components, which are "Affect Classification" and "Facial Feature Tracking". The camera's video input is handled by the facial tracking module. This component collects and tracks "facial features" throughout the period. The facial features' spatial information is subsequently analyzed by the affect categorization component. The facial feature tracking component recognizes and records the positions of face fiducial points across time. In their method, to extract shape information from a video sequence's face, they used the Active Shape Model (ASM). ASM was presented by [22]. This is an algorithm that matches a set of shape points to a picture within the constraints of the shape's statistical model. There were therapists that supervised each participant while they played the game.

This study measured facial emotional expression and subjective emotional reactivity in eating problem patients while playing video games. ED patients had higher subjective anger while AN patients had lower facial anger. However, groups expressed delight similarly [23]. Therefore, we may come to the realization that we can employ Facial Feature Tracking and Affect Classification by making use of the Active Shape Model (ASM) in order to recognize the feelings that our volunteers are expressing and the changes in their facial behavior.

3.2.2 Speech Emotion Recognition

Our main goal for this is to detect the emotions of the players from the recordings. For doing this, we have studied some papers in order to find relevance to achieve our objectives. Emotion is particularly recognized by facial expressions, speech, physiological signals etc. In a study by [24], they used machine learning models to recognize speech emotion. To do this, they developed a Speech Emotion Recognition (SER) system with different classifiers and different methods. From the speech, modulation Spectrum (MS) and Mel Frequency Cepstrum Coefficients (MFCC) features were extracted, which were used to train classifiers. To classify seven emotions, an RNN classifier was used.
RNN is the recurrent neural network. They have used Berlin and Spanish databases for experimental datasets. For the Berlin database, it achieved 83% accuracy with speaker normalization. Again, the Spanish database achieved 94% accuracy with the RNN classifier.
In a study done by [25], they used a machine learning approach to automatically recognize human emotion. Their approach was similar to the previous study, which we have shown. First, they extracted numerical features from the sound database. Then, by using the feature selection method, they selected the most relevant features. Lastly, train the whole model to recognize seven emotions. Here, they have also used "Berlin Emotional Speech," which contained 535 audio files. For feature extraction, OpenSmile was used. OpenSmile is a widely used tool for feature extraction and signal processing. After that feature was selected. Then three classification algorithms were tested in this study which are K-Nearest Neighbors (KNN), and Support Vector Machine (SVM). SVM has the highest accuracy, at 86%, over KNN and Naive Bayes. With trained and tested with 10 fold cross-validation, SVM gave 86% accuracy. The highest and lowest precision are, respectively, sadness and happiness. Here, anger mixes with happiness, and fear mixes with other emotions.

Other researchers like [26] and [27] also followed the same pattern to develop a SER system. We have also developed our SER system in the following manner.

3.2.3 Machine learning and Deep learning models in Video game

Changes in player behavior as a result of machine learning have received surprisingly little research. [28] conducted a recent study on predicting player personalities in first-person shooter games. Both modern gaming and our study's focus on that genre, which consists largely of shooters, are related. An action game titled "The Protector" was created for the sake of this paper's data collection. The Five-Factor Model (FFM) was utilized to determine characters' attributes in-game. The Five-Factor Model (FFM) is a popular method for evaluating individual differences. More than 12,100 distinct human players (with 30,000 tuples) were included in the entire dataset. In order to quantify FFM characteristics, the dataset was split into five subsets: openness, consciousness, extraversion, agreeableness, and neuroticism. They used machine learning and deep learning algorithms to assess data sets once they had been prepared. They employed classification methods like artificial neural networks (ANNs), Support Vector Machines (SVMs), linear regression (LR), and stochastic

gradient descent (SGDs) to establish the degree to which each characteristic was present in the pool of available data. In this case, Artificial Neural Network (ANN) provided the best accurate findings (98.6%). They had used a custom structure of ANN model. In this case, the player prediction was accomplished through the use of categorization models, which drew upon the prediction capabilities of mentioned algorithms. To uncover this pattern, the authors of this paper employed a machine learning algorithm. Our paper will likewise employ machine learning models, but it will instead determine a player's level of aggression through observation of their gameplay.

3.2.4 Deep learning model in Facial Expression Recognition

Humans express themselves through change in facial expressions. That is true even when playing video games. Sudden change in situations in video games also changes the expression in face. So, not only in video games, but also in any situation, to read people's emotions, understanding facial expressions and patterns is important. And to do it faster, through machines, different recognition models and deep learning are used.
With main focus on deep learning for facial emotion detection, a framework was experimented by [29]. OpenCV computer vision library, a class library, was utilized by the said framework to finalize facial recognition. To achieve this goal, OpenCV makes use of the AdaBoost algorithm. Keras, which is an artificial neural network library, also open source (written in Python) is used to develop the Convolutional Neural Network (CNN), a neural network architecture. For training the model, Stochastic Gradient Descent algorithm was applied. Nie used the FER2013 database to train his CNN neural network.

Again the FER2013 dataset, along with a custom dataset, was applied by [30]. Minding the goal of emotional recognition, they pre-processed RGB images by transforming them into gray images. Also, they applied the Haar method to process and identify real time photographs and static images. Provided that the face is detected successfully, facial characteristics could be resized and refined. On the goal to classify emotion, [30] used CNN to train on facial characteristics, which helped them in identifying the characteristics according to five emotions. To measure community emotion, a weighted average emotion measurement was applied. It resulted in achieving the accuracy of the model at 65% for FER2013 dataset and 60% in case of custom datasets which depicts that the proposed CNN model was effective in recognizing facial expressions.

A CNN model for facial expression recognition was proposed by [31] which included three continuous emotions. Taking inspiration from Xception, the proposed model applied residual blocks and depth separable convolution in order to reduce the total parameter size to 33,000. In order to achieve emotional stability identification, a convolutional neural FER network was applied by them. This model applies convolutional operations to the input images, to automatically extract and learn characteristics. This method removed the requirement of manual extraction of features from the images. This proposed model was effective as it achieved 81% accuracy for not visible results. The model effectively detected both positive and negative

emotions, 87% accuracy in case of positive and 85% in case of negative emotion detection. However, one downside is that it was less effective in detecting neutral emotions, gaining only 51% accuracy in this case.
To improve identification of mental state and diagnosis, [32] established a new deep Convolutional Neural network (CNN) architecture. Through a new style, this method operates on images of the face and examines emotional states. It takes out deep features from AlexNet architecture and to finalize results, applies linear LDA. The machine includes three components, input videos of facial expressions, pre-processing steps for images and the predictive interpretations of facial expressions. After experimenting, the results presented that the proposed method outperforms other precise and efficient techniques.

With features learned through CNN, [33] attempted to detect emotional states by applying Electrodermal Activation (EDA) signals while merging the characteristics learned through a Convolutional Neural Network (CNN). As the DEAP database is publicly available, they acquired EDA signals from this database and assimilated them to a standard. By applying the cvxEDA technique, the acquired signals were further processed in order to separate them into tonic and phasic components. In order to acquire and evaluate changes over a short period of time, the phasic component underwent Fourier transformation. From the phasic signal, 38 time, frequency, and time-frequency characteristics were extracted. The CNN then enforced this characteristics to gain effective and enhanced features.In order to allocate the emotional states, five machine learning algorithms were applied. Linear Discriminant Analysis (LDA), Multi Layer Perceptron (MLP), Support Vector Machine (SVM), Extreme Learning Machine (ELM) are prominent. The findings of [33]'s study points that the proposed approach is effective in categorizing emotional states based on enthusiasm dimensions.

Applying the LeNet Convolutional Neural Network architecture, [34] suggested an economical and practical approach for the classification of seven unique emotions in real time in their works. The said emotions are Happiness, Sadness, Surprise, Anger, Disgust, Fear and Neutrality. To train the CNN model, a collection of facial expression images were acquired which showed very high accuracy. The Haar Cascade library was applied to enhance the model's performance. This method helped to reduce the unwanted pixels around the face in a specific image. Also, to reduce training time and number of required networks, the pixel positions in the images were optimized. Ozdemir and his team merged the data from three different datasets. These datasets are Karolinska Directed Emotional Faces (KDEF), Japanese Female Facial Expression (JFFE), while also merging their own custom dataset. Extraordinarily, the testing and the assessment results indicated that, when used for testing, the custom dataset was effective as it outperformed the other two databases, which were already established. Therefore it indicated that the real time evaluating model showed the effectiveness to analyze and process any image given in a short period of seconds.

3.2.5 Deep learning model in Speech Emotion Recognition

Speech is a natural way for people to communicate by not only sharing information but also showing how people are feeling. Emotions reflect how we think, behave, and interact with others. So, it is important for people and machines to be able to recognize and understand the emotions in speech.
[35] suggest using a deep-learning network for small databases. Initially, they create spectrograms for each recording using conventional techniques, and these spectrograms are then resized while maintaining their aspect ratio. They figured out that making the pictures bigger makes the computer program better at recognizing things, but it also makes it harder for the computer and needs more computer power. They use a network with only one hidden layer, but they say it works even better than the state of the art.
A method that uses a deep neural network with convolutional, pulling, and fully connected layers is proposed by [36] . In the paper, based on Berlin Database three particular classes of emotion was detected. Angry, Sad, and neutral. They have pre processed by removing the silence. After that it was segmented into 20 ms audio.
[37] in their work introduce a method for recognizing emotions using parallelized convolutional recurrent neural networks. They begin by extracting features from each spoken phrase. They do this by using a long short-term memory network to learn these features. Additionally, they calculate Mel spectrograms and employ a convolutional neural network to capture features from the image representations. Both sets of these advanced features are learned, adjusted to a common scale, and then injected over a softmax classifier to identify the emotion from the phrase.

In earlier methods, like the one by [38] introduce an approach that relies on a combination of CNN and RNN, by not using hand-crafted features, such as spectrogram images. A deep learning and recurrent neural network approach for emotion classification is proposed by [39]. They use MFCC and spectrogram features from the audio recordings as inputs for the networks. The deep learning network has two hidden layers that are fully connected to the input and output layers. The recurrent network uses gated recurrent units (GRU) as the model. They claim that their approach achieves higher accuracy than the state-of-art. However, Deep neural networks are networks with more than one hidden layer. So, based on our definition, deep artificial neural networks are better at solving complex problems compared to networks with just one layer. In their paper, [40] created a probability distribution by deep neural network. Additionally, they used a simple and easy single neural network called ELM (extreme learning machine) to extract the emotions from the entire speech. This network is particularly effective at extracting the emotions when there are a small number of training materials available. To find out the emotions, they have mixed their predictions from the deep neural network to make features for the whole speech, and they gave those features to ELM to find the emotions.
[41] have introduced us to a convolutional neural network. It had three emotion with an accuracy rate of 66.1%. They have compared their network that was trained without any prior knowledge with a baseline Feature-based SVM. This means that their network did not use any features or labels from different sources, but learned directly from the raw data. A baseline Feature-based SVM is a common method that uses a set of predefined features and labels to train a support vector machine, which is a type of machine learning model. They have used a collection of TED talks that were

labeled by students and crowd sourced to train and test their method. They have built their CNN using the Theano toolkit. They have also trained a linear SVM with a feature set from the INTERSPEECH 2009 emotion challenge for comparison.

3.2.6 Neural Networks

A neural network is an artificial intelligence technique that trains machines to process information in a manner similar to that of the human brain. A neural network usually contains three layers: the input layer, one or multiple hidden layers, and the output layer. The input layer receives data, hidden layers process it using mathematical computations, and the output layer produces the final result. Neurons use

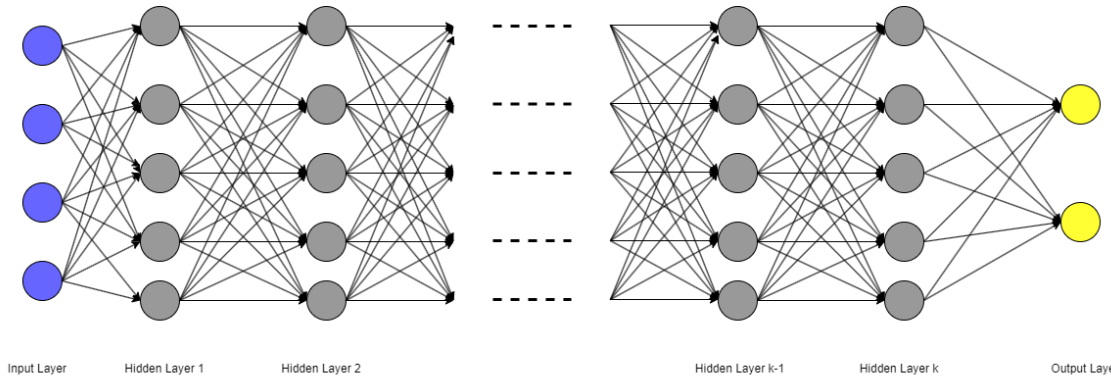


Figure 3.2: The architecture of a deep neural network

weights and biases to calculate their output and then pass it by an activation function. The number of neurons in the input and output layers depends on the data size and the problem's complexity. Training a neural network has steps like input preparation, forward propagation, backpropagation to minimize errors, and gradient descent for parameter adjustment. The process includes optimizing techniques like Stochastic Gradient Descent, Adam, and RMSprop, chosen based on the problem, dataset, and model architecture.

Activation Function

An activation function is like a decision-maker for a neuron in a neural network. It decides whether the neuron should be active based on certain criteria for handling real-world data. There are different types of activation functions, such as Sigmoid, Tanh, ReLU (Rectified Linear Unit), and Leaky ReLU. Leaky ReLU: To address the "dying ReLU" problem, where some neurons become inactive and cease learning, the Leaky ReLU introduces a small, non-zero gradient for negative inputs.
Choosing the right activation function is crucial for a neural network's performance, and different functions can be used for different layers based on the task. The figure 3.3 shows the mathematical representation formula of Sigmoid, Tanh, ReLU and Leaky ReLU as well as the graphical representation of them.

Optimizer

Algorithms or techniques known as optimizers are applied to modify a machine learning model's parameters in order to minimize a loss function. Basically, they

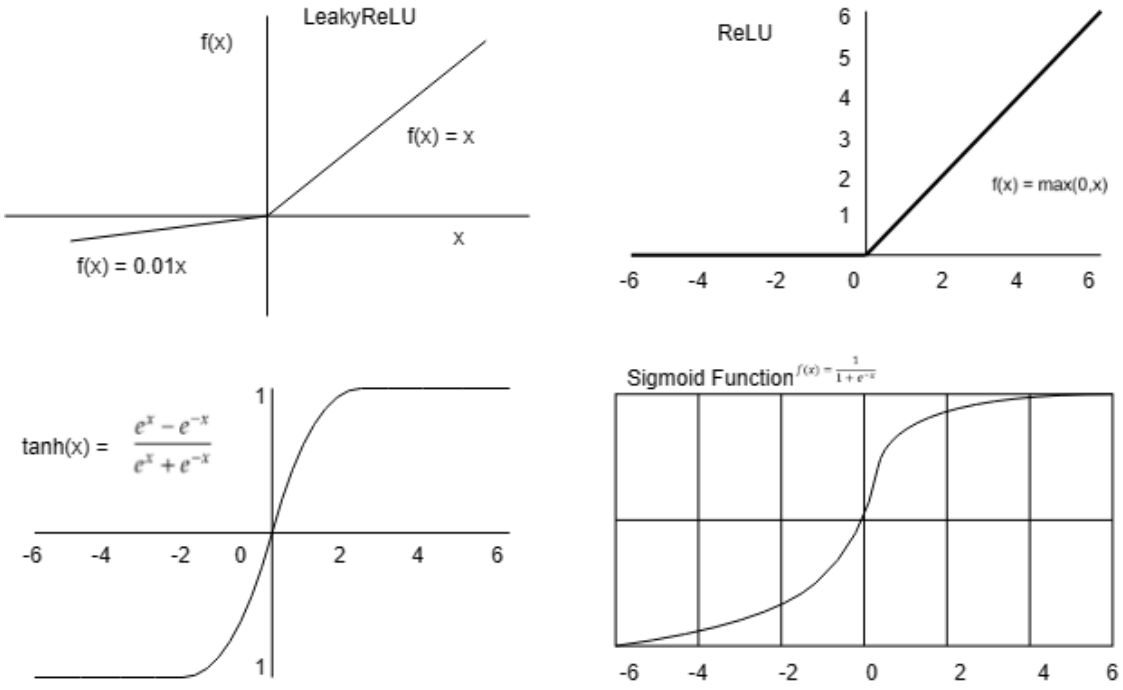


Figure 3.3: Artificial neural network activation functions

work by minimizing a "loss function," which evaluates how accurate the predictions are. Common optimizers include Stochastic Gradient Descent (SGD), Adam, AdaGrad, and, RMSprop. SGD is basic, adjusting the model based on the opposite of its mistakes while Adam is an advanced version. They efficiently adjust learning rates, making it a reliable choice in complex scenarios.

Layer Normalization

A process to adjust the input layer of a neural network is by using called layer normalization. It is like another method called batch normalization. But the difference is, instead of focusing on the whole batch of data, it works on the parts of a hidden layer. This helps to keep things stable and improves how well the network performs overall.

Convolution

Images' positions, sizes, and looks make it hard for computers to figure out about a picture. Convolutional Neural Networks (CNNs) are a type of deep learning network designed to especially help with this type of issue. These networks are good at understanding images. A CNN has different layers like convolution layers, pooling layers, and fully connected layers. All of those work together to identify different parts of the image and include textures, edges, shapes, etc. A technique called max pooling helps to focus on important parts of the image. For a 2D picture with a size of M $\times$ N, the formula for convolutional mathematics is:

$$F * I(x, y) = \sum_{j=-M}^{M} \sum_{i=-N}^{N} F(i, j) I(x - i, y - j) \tag{3.1}$$

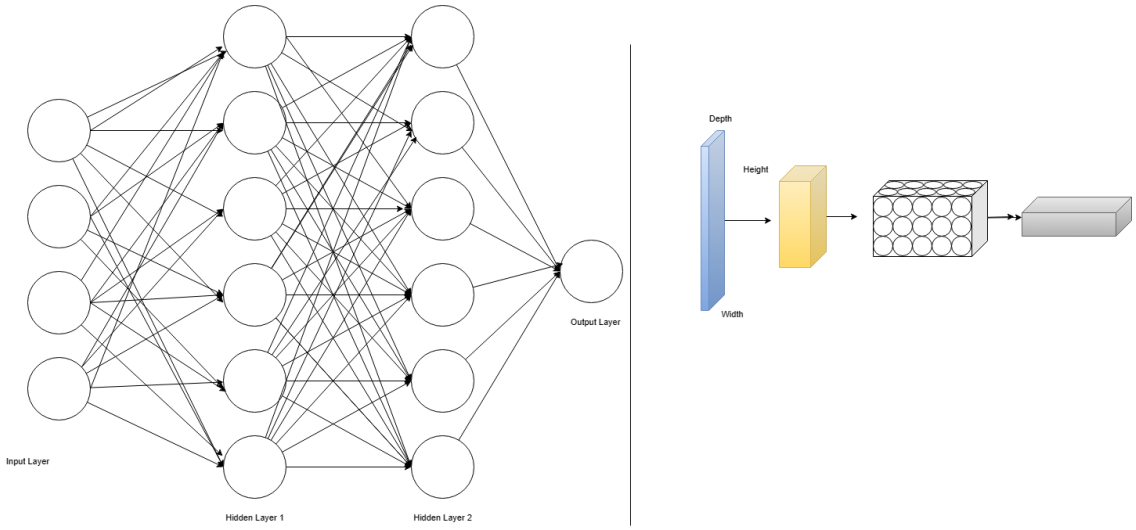


Figure 3.4: A regular artificial neural network and a convolutional neural network

Feedforward Neural Networks

Feedforward Neural Networks are a specific type of artificial neural network where information goes in one direction, from input to output, without coming back. They process input data through hidden layers, with neurons in each layer performing calculations based on weights and biases. These networks are commonly used for tasks like classification, regression, and prediction. These types of networks are very easy to use and they can also be trained using methods like backpropagation and stochastic gradient descent (SGD). But, they face problems with complicated relationships between variables.

Batch Normalization

By changing the batch's SD (Standard Deviation) and then subtracting the mean value, it helps to making it more stable in every layer. This makes the model better at figuring things out in general and makes learning smoother. It is really useful when we train deep convolutional neural networks with small batches of data.

Supervised Learning

In supervised learning, a neural network learns to give answers that we already know from the information we get from inputs. We compare its predictions with the actual answers and tell it how to adjust its weights and biases to get closer to the truth. We do this till the accuracy is increased. It is quite known that supervised learning can use past experiences to predict future outcomes. However, supervised learning also has some limitations. It cannot handle complex tasks that require more than just input-output mapping. Also, as supervised learning requires a lot of calculation work, it is very time-consuming, which can also be considered as another limitation of it.

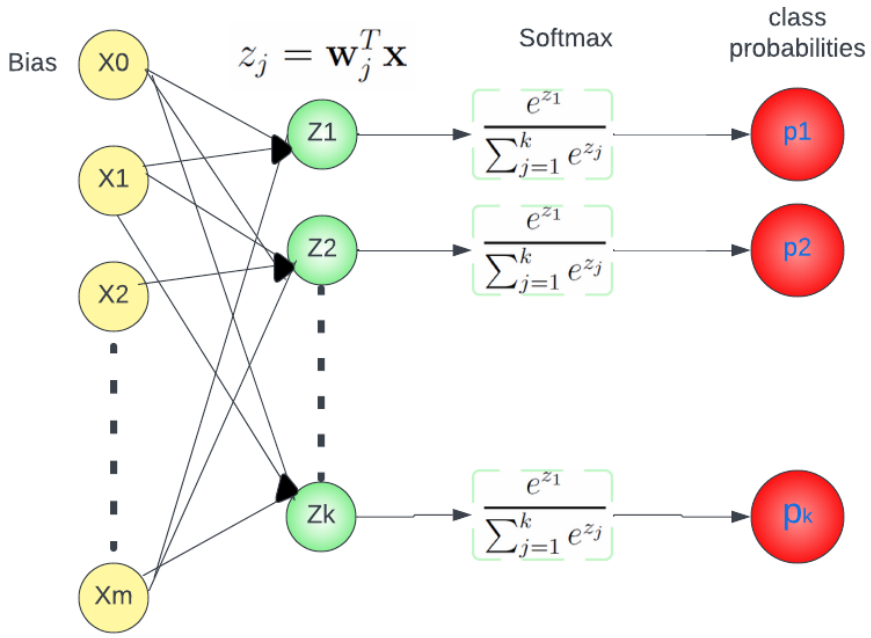


Figure 3.5: The graphical representation and formula of Softmax

Softmax

Softmax plays a vital role in neural networks by converting real numbers into a probability distribution across classes. The N-dimensional vector it produces ensures that the elements range from 0 to 1 and sum up to 1. This transformation involves exponential and normalization, as seen in the formula in figure 3.5.
In multi-class classification, Softmax works with the cross-entropy loss function to make the predicted probabilities match the actual ones.

Transfer Learning

Reusing a machine learning model that has learnt one task for a different but similar task is known as transfer learning. This can save time, data, and money, as the model can build on what it already knows instead of starting from scratch. Transfer learning works by changing and training some parts of the model with new data, while keeping other parts fixed. Transfer learning is very popular and useful for complex tasks like image recognition, where training a model from scratch is very hard and costly.

Audio features

- Mel Frequency Cepstral Coefficients (MFCC): Mel Frequency Cepstral Coefficients is a very common feature extraction technique. It is based on the human auditory system's perception of sound. In order to classify and examine sound patterns, MFCC helps in organizing the portrayal of sound data. This feature is often used in speech analysis, speech emotion recognition etc [42]

- Mel spectrogram: The Mel Spectrogram is a visual representation of an audio

signal which is the spectrum of a sound signal [43].This method is often used in speech emotion recognition. It focuses on the low-frequency part of speech [44].

- Zero Crossing Rate (ZCR):A feature in classifying matching sounds is Zero crossing rate which is used for both audio identification and musical data retrieval. It simply is the rate where a signal switches from negative > zero > positive or vice versa. ZCR is often used to measure the noisiness of a signal. For instance, when it found the noisy signal, it generates higher values [45].

- Chroma: To identify harmonic resemblance or chord similarity, Chroma is an important feature. It is a descriptor, which is regarded as the tonal content of audio signals in condensed form [46].

- Root Mean Square Value: RMS value is the feature that is the measurement of energy captured by a fixed length wave. From a general perspective, it represents an indicator of loudness. When it captures louder sound, the value of RMS becomes higher [47]

3.2.7 Used Models

Specification of Custom CNN

Convolutional Neural Network is a deep learning model that is very effective for working on images. These workings include categorization of images, object detection in images and facial recognition. CNN can detect the subtile changes due to its spatial hierarchucal nature. Our problem is also related with the detection of subtle changes. The facial features such as when someone is happy, the wrinkle around the eye, or the change in the intesity of eye, or when someone is angry, the face musle movement changes. These subtle changes is detect by CNN, which is why we chose this model. Custom CNN involves specific design of the neural network on a specific dataset. It is involved in either customizing an existing CNN architecture in accordance with their goals or building an architecture following the CNN guidelines from the scratch. The customization of the architecture involves modifying the amount and dimensions of the convolutional layers, parameters, including domain specific knowledge etc. These customizations are done so that we can optimize the architecture for a particular dataset, as existing Architecture may not give us the best fit.

Specification of Densenet121

CNN face 'Vanishing Gradient problem' as it gets more deep. However, DenseNet resolves it by modifying the CNN architechture. In this architecture, rach layer directly connected to every other one. Because of this, it became densely connceted. For 'L' layers, total direct connections are L(L+1)/2.[48] It is divided into Dense-Blocks. For each block, feature map's dimensions remains same. However, filters number changed. There is a Transition layer between blocks which make half the number of channels. In the each layers, three side by side operations occurs, first batch normalization, then a rectified linear unit (ReLU) and lastly a convolution

(Conv) [48]. The architechture of DenseNet121 is somewhat complex. It has a convolution layer with a stride of 2. It has 64 filters. Each of it has a size 7X7 and. Then a pooling layer. The max pooling size is 3x3 with same number of stride as before. After that, a Dense Block 1. It repeats for 6 times which also has 2 convolutions. Then, a Transition layer 1 which has a 1 Conv and 1 AvgPool. After that, a Dense Block 2 repeated for 12 times. Again a Transition layer 2. Then, Dense Block 3 repeated for 24 times. Then a Transition layer 3. Again a Dense Block 4 repeated 16 times. Then, a Global Average Pooling layer. It accepted feature maps. With them, it perform classification. Lastly, Output layer [48].

Specification of VGG16

The VGG16 model, a well-known convolutional neural network (CNN), has shown its relevance in the field of computer vision and deep learning. The model, first proposed by the Visual Geometry Group (VGG) , has been important in several image analysis tasks, with a special emphasis on picture categorization [49]. Within this part, we provide a comprehensive outline of the VGG16 model and explicate its significance within the context of our study regarding facial expression recognition. Our study uses the VGG16 model because of its feature extraction and picture recognition capabilities. The model's capacity to recognize sophisticated visual patterns matches our face expression recognition goals. We integrate VGG16 to improve our face expression recognition system's accuracy and resilience. However, integrating the VGG16 model into our face expression recognition research framework is crucial. The VGG16 model is essential to our face expression decoding and classification technique. We want to improve our emotion identification system by using its ability to extract discriminative characteristics from face photographs. We used the VGG16 model to meet our study goals via careful implementation and optimization.

Within the domain of facial expression recognition, the VGG16 model undertakes the task of processing an input picture via a sequence of consecutive layers. Commencing with the basic layers, the model discerns rudimentary elements such as edges, corners, and textures. As we go through the network, successive layers are capable of extracting more complex and abstract characteristics, such as the intricate details of face shapes, expressions, and the intricate interactions among facial muscles. The use of hierarchical feature extraction in VGG16 allows for the recognition of nuanced indicators associated with different emotions, including but not limited to happiness, sadness, and rage.
The numerical designation "16" in VGG16 signifies the presence of 16 levels inside the model architecture, each of which has associated weights. The VGG16 architecture consists of a total of 21 layers, which has total of thirteen convolutional layers, with five Max Pooling layers, and three Dense layers. Even though that despite the presence of 21 levels, the VGG16 model only has sixteen weight layers, which are responsible for the learnable parameters [50].
Also, we used a pre-existing VGG16 model that has been trained on a vast collection of diverse pictures, so benefiting from its weight initialization. The aforementioned methodology presented several benefits, such as expedited convergence of training and the proficient recording of universal face characteristics. The process of fine-tuning the pre-existing model for our unique purpose resulted in a smooth transfer of the model's profound comprehension of face characteristics, leading to precise

identification of facial expressions.

Specification of Resnet50

ResNet-50 is a deep learning and computer vision milestone. Its launch improved deep neural network training, notably in image categorization. Microsoft Research developed ResNet-50, short for "Residual Network with 50 layers," to overcome the challenges of very deep neural networks [51].This model's design has leftover blocks with skip connections. These connections allow the network to avoid the vanishing gradient issue and train impressively deep networks. ResNet-50's success in picture classification challenges like the ImageNet Large Scale Visual Recognition Challenge propelled it to the forefront of deep learning research.
We chose ResNet-50 for our thesis since it is relevant to our facial expression recognition study. Face expression identification is complex and requires subtle feature extraction from face photos to effectively identify emotions.ResNet-50's design enhances feature extraction. By our model, for face detection we need to detect specific changes. Here, ResNet50's deep network helps us with this. With this, we can capture facial patterns.
ResNet-50 processes face pictures across layers for facial expression recognition. Its clever "skip connections" design solves the vanishing gradient problem, which affects deep network training. Skip connections allow the model to bypass layers during training, maximizing gradient flow. Thus, ResNet-50 can decipher facial expressions' complex elements and patterns to identify emotions. The ResNet-50 architecture, consisting of 50 layers, is characterized. Here some key component is its a 7x7 kernel convolution layer with 64 additional kernels, utilizing a stride of 2 for feature extraction. It has a max pooling layer with a stride of 2 for downsampling [51].
After the architectural layers, we fine-tuned the model for face emotion identification. By fine-tuning, we use ResNet-50's pre-trained comprehension of generic traits to recognize our research's emotions. The design ends with an average pooling layer and a 1000-node fully connected layer that uses softmax activation for exact classification. We used fine-tuning to leverage ResNet-50's potential for facial emotion identification while keeping uniqueness and avoiding copying.

Specification of VGG19

A convolutional neural network with 19 layers is called VGG19. Its is trained on the ImageNet database which has over a million photos. Its architecture is similar to that of VGG16, a model first proposed by the Visual Geometry Group (VGG). However, VGG19 has more max-pooling and convolutional layers. Although it has almost surpassed VGG16 on a number of image classification benchmarks, its higher parameter count also increases processing costs.
The last three convolutional layers are utilized for classification, whereas the first 16 convolutional layers are used to extract features. [52] employed the VGG19 architecture for their research. They modified the classifier layer of the VGG19 architecture in various ways and then utilized a 0.3 dropout layer to prevent overfitting before using a single dense layer with two neurons. They pointed out that VGG19 offers one of the highest accuracy rates reported in the current literature [52]. VGG19 has

been used to be an effective classification architecture for several different datasets, and as the models' creators made them open sourced, they may be applied, either similarly to or slightly modified to other comparable jobs.

Specification of InceptionresnetV2

The convolutional neural network Inception-ResNet-v2, which has 164 layers and can categorize photos into 1000 object categories, was trained on over a million images from the ImageNet database. When it is about designing how computers learn about things, transfer learning for classification depends on specific architectures like VGG networks, ResNets, or Inception Networks. VGG networks stick to the classic setup of basic convolutional neural networks, using a sequence of convolutional, max-pooling, and activation layers before reaching fully-connected classification layers at the end.
Meanwhile, ResNet has a unique structure with shortcut links, which makes it to skip one or more levels through shortcut connections. Lastly, Inception Networks bring a twist by using pooling within a single layer along with convolution kernels of different sizes. It's like giving the computer different tools and tricks to figure out what's what [53]. The ResNet v2 is for difficult picture identification tasks, the network's depth is important in capturing abstract information.

Specification of MobileNetV2

MobileNet-v2 is a type of deep convolutional neural network with a depth of 53 layers. It is a very lightweight designed pre trained model for mobile devices. The architecture adopts an inverted residual structure, incorporating residual connections between bottleneck layers. In the intermediate expansion layer, lightweight depthwise convolutions are employed to filter features and introduce non-linearity. The overall structure of MobileNetV2 includes a fully convolutional layer with 32 filters. Also include 19 residual bottleneck layers. The design uses residual connections between bottleneck levels and utilizes an inverted remaining topology [54].

Specification of Xception

Convolutional neural network Xception has 71 layers deep. The ImageNet database contains pretrained versions of the network that have been trained on over a million photos. Thanks to this training, the pretrained network is capable of classifying images into a wide array of 1000 different item categories. These categories encompass a diverse range. On the ImageNet dataset, Xception performs slightly better than Inception v3, and much better on a bigger image classification dataset with 17,000 classes. It indicates a higher computing efficiency because it has the same amount of model parameters as Inception [55].

Specification of EfficientNet-B0

AutoML created the baseline network, EfficientNet-B0. Similar to the previous models mentioned here, EfficientNet-b0 is also a convolutional neural network trained on more than a million images. EfficientNet is a type of neural network that uses a technique called "compound scaling." This helps balance the size of the model, its

accuracy, and how fast it can do computations. It is like finding the right mix so that the network works well without being too big or slow [56].

Specification of RNN

Recurrent Neural Networks (RNN) are a type of artificial neural network which are commonly used in natural language processing, speech recognition and time series. The main difference that makes them able to correctly identify these is the internal memory. Traditional neural networks are not able to do that, which is possible through the RNNs long memory model [57]. In vanilla RNN, the present contextual idea is explored. It can easily predict the next sequence from the past information. How when it needs more past context to generate or predict a new sequence for example a word, it won't be able to. However, this problem is solved by Long Short Term Memory (LSTM) networks [58].

Specification of LSTM

LSTM was mainly designed to avoid long-term dependency problems. LSTM has a chain structure like RNN, however, inside, it has four neural network layers. These layers interact in a special way. As the main idea of this model was to solve long-term dependencies, the cell state of the previous layer always continued forward to the next layer. LSTM has the ability to add or remove information to these cell states. To add or remove information is only allowed through the gates [58].
We used the LSTM model to our speech emotion recognition model. LSTM models are well-suited for capturing long-term context from the speech data which is given. In the model, the important features were extracted from the audio speech data. These feature data were given to the model for further detection. In speech data emotions were passed through the pitch, intensity etc. The LSTM model can capture these patterns and get the prediction [57].

Specification of GRU

GRU model is similar to LSTM, however, it is the latest model which provides more benefit over the vanilla RNN and LSTM. Like LSTM, it also solves the vanishing gradient problem and long-term memory dependency. However, the model structure is much more simpler than LSTM. It has only two gates, Update gate and Reset gate. Even though the structure makes it more simple, in terms of the long-term dependency, LSTM is more powerful [57].

Specification of CNN-LSTM

In our model, we have used the CNN-LSTM combination approach. This approach was taken because CNN works well suited to extract the patterns and features from data. In our model, we have extracted many features such as MFCC, Mel-spectrum etc. Convolution layers can effectively learn from these features and get patterns. LSTM makes the model able to capture long long-term dependencies. In speech data, we often capture long data. Then, it is segmented to many small pieces of data. Which is why every speech data may have been linked with the previous data. The LSTM model helps us to solve the long term dependency in these cases [57].

Chapter 4

Methodology

4.1 Dataset

The rationale for this research stems from the recognition that video games offer a controlled and dynamic environment in which players' emotions and behaviors can be observed and analyzed. By integrating facial expression recognition and speech emotion recognition technologies, we aim to decipher the intricate interplay between in-game experiences and emotional states, ultimately shedding light on how gaming impacts players' behavior. We have created our own dataset for facial expression recognition (FER). We were able train our model with around 60,000 images for facial expression recognition. These images were extracted from online streamers and gameplay recorded videos from one individual. These all of the games were violent video games from recent time. We took Valorant, PUBG, Fornite, Call of Duty. Moreover, to ensure comprehensive understanding of gamer's behavioral changes we have trained our machine learning models for the speech emotion recognition with a combination of four distinct datasets: RAVDESS, SAVEE, TESS, and CREMA-D. With these trained models, we can efficiently predict and identify players' behavioral changes for future data.

4.1.1 Facial Expression Recognition(FER) dataset

To ensure the ease of future implementation without the need for advanced medical equipment, we adopted a straightforward approach for collecting data from online gamers and our friends.

Data Collection for FER

We took YouTube into consideration as our primary source for data collection. The widespread use and data availability are the main reasons for selecting this platform. Recorded and live videos are easily accessible in this video sharing platform. We selected diverse category of gamers for our research. Selection criteria includes: popular streamers, new comers who are struggling for more followers, different age groups, availability of videos uploaded, posting schedules, short period streamers, streamers who used to stream 10-12 hours straight in a day, good audio and video quality and finally where face is clearly visible. Besides, the variation also include the environment and other setup including, wearing glass, sun-glass, at night, different gender etc. We asked an individual to record their gameplay for two months so that

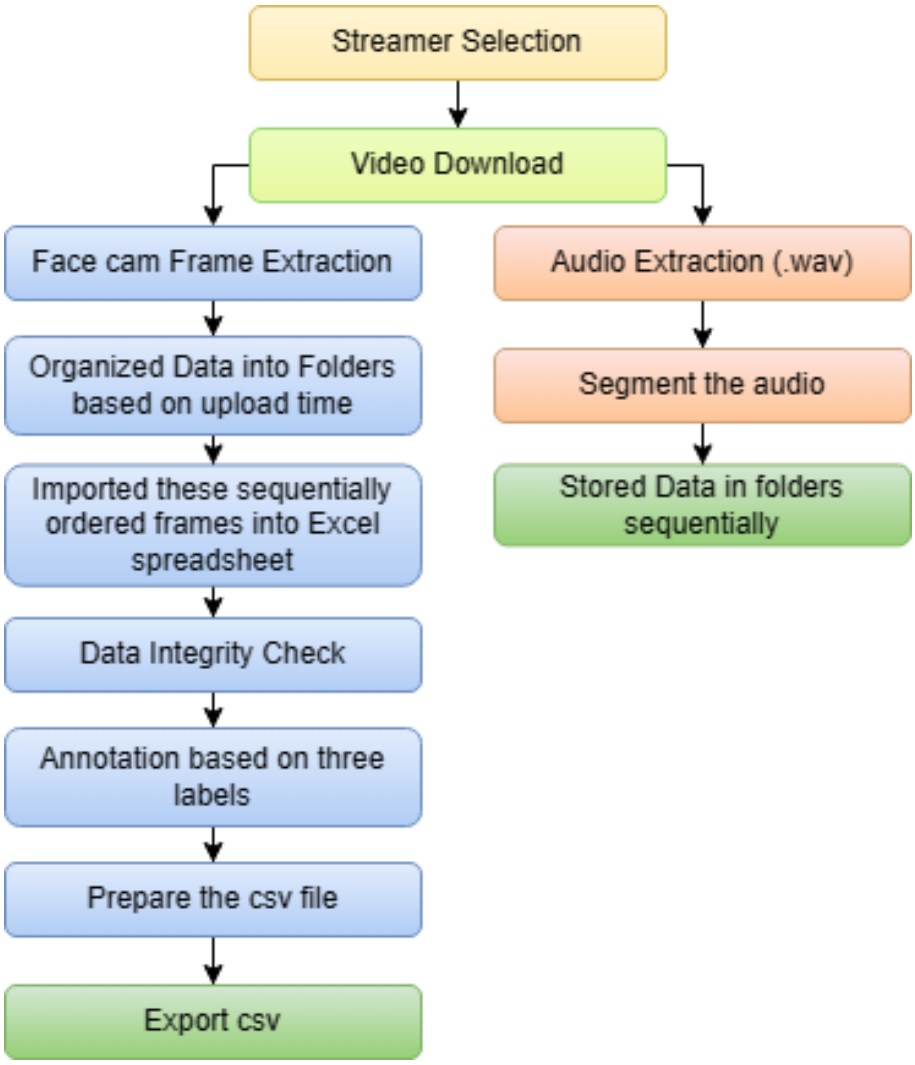


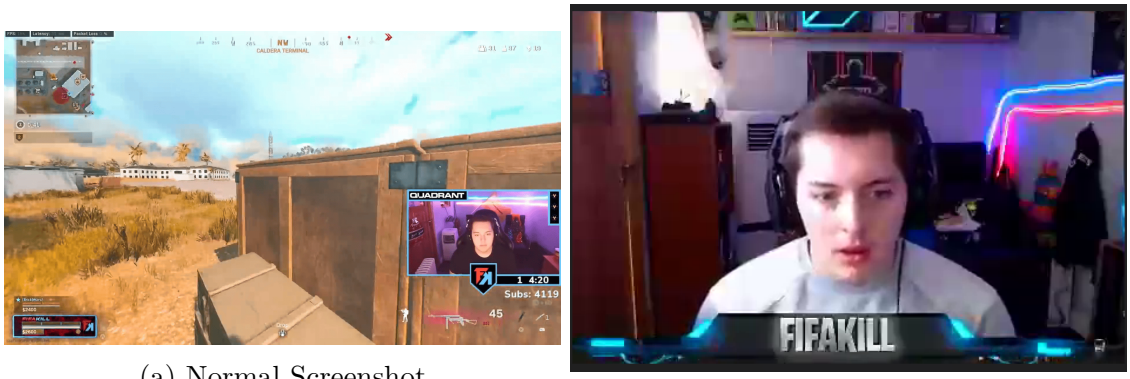
Figure 4.1: Dataset preparation

we can collect data from them also. We had to provide a high resolution webcam and microphone to our friends to get quality data.

Extraction Process

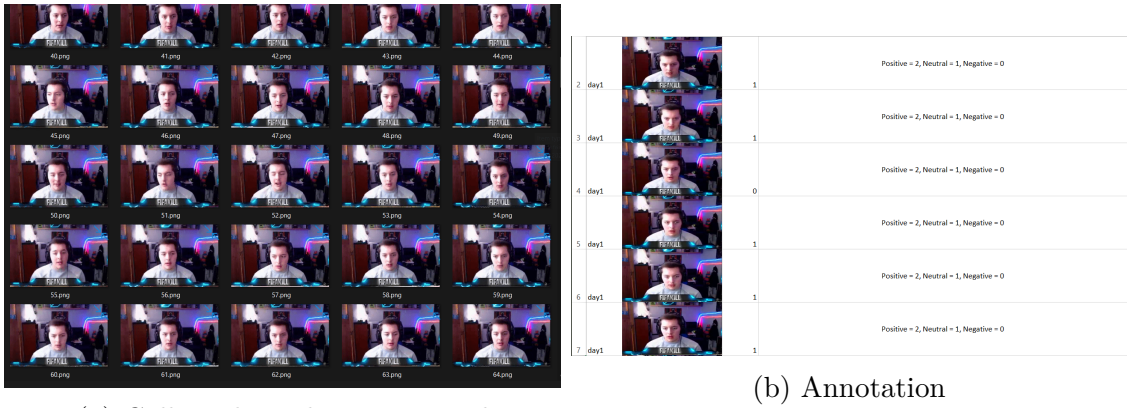
After shortlisting the right streamers and YouTube channels we started the data extraction process. To observe changes in their behavior over time, we decided to analyze last one year's worth of content. We started downloading the contents with the help of internet download manager (IDM) of a particular streamer. Given that, some YouTubers often stream for extended periods, typically around 4 to 5 hours daily, we recognized the need to focus our analysis on this type of streamers too. For the facial expression recognition aspect of our study, we only required the frames containing the streamer's webcam or live face, generally appearing as a rectangular frame within the video. To accomplish this, we developed a Python script to extract specific frames and audios from the downloaded videos. (4.2a). Which being the most hectic part during data extraction. Since, the webcam frame position is not fixed for all streamers even for a particular streamer, the webcam position doesn't not remain fixed for all the videos. So, we had to update the webcam's frame x and

y coordinates for each video. In this process, we captured still frames at regular 4-second intervals from the videos. For example, from a 30-minute daily live stream video, we extracted a total of 450 still frames, each showcasing only the streamer's face. According to Paul Ekman Group, obvious or "normal" facial expressions last between 1/2 a second to 5 seconds. [59] That's why we choose to extract frames in every 4 seconds intervals. To ensure data quality, we removed any unwanted frames. This included eliminating flashy or blurry images and manually deleting frames where the streamer temporarily left the webcam frame, such as when eating or expressing intense emotions like excitement or anger.



(a) Normal Screenshot

(b) Face proportion extract



(a) Collect through 4sec interval

(b) Annotation

Figure 4.3: Data collection procedure

To maintain the chronological sequence of frames, we organized them based on their upload time. For example: if a streamer has uploaded 28 videos in the month of May, we organized the data of each day in different folders (Day 1, Day 2…. Day 28) under folder May. We also extracted the audios in .wav format from each video in the same time and organized them in the same manner. Subsequently, we imported these sequentially ordered frames into an Excel spreadsheet for further analysis. The Excel sheet featured six essential columns:

a. player_name
b. day_number
c. frames
d. image_number
e. emotion

f. Annotation labels

	A	B	C	D
1	player_name	day_number	image_number	emotion
2	player8	day1	0	7
3	player8	day1	1	1
4	player8	day1	2	1
5	player8	day1	3	1
6	player8	day1	4	1
7	player8	day1	5	1
8	player8	day1	6	7
9	player8	day1	7	2
10	player8	day1	8	1
11	player8	day1	9	2
12	player8	day1	10	2
13	player8	day1	11	2
14	player8	day1	12	2
15	player8	day1	13	1

Figure 4.4: Final CSV file

The most challenging part of creating this dataset was manually annotating the extracted frames. Keeping in mind that we have to train our model with these data, we started labeling the data carefully in excel under emotion column. We decided to annotate the frames on four different numerical annotation labels, a. Negative expression - 0 b. Neutral expression- 1 c. Positive expression -2 d. Distorted images where facial expressions can't be detected accurately - 7 Finally, from this initial excel sheet, we created a separate csv file for model training. The structure of the .csv file is shown in Figure 4.4.

Quality Assurance

Before annotating data of a particular streamer we emphasized on observing him/her for couple of days. We used to watch videos of streamers in our free time for better understanding of a specific streamer's facial expressions in different situations. For example, how does his/her face looks like when he is angry or sad. This definitely helped us to make a perfect dataset with accurate annotation. Manual annotation for such a huge dataset was never an easy going task for us. To assure the integrity of our dataset, we finally cross checked the organization and annotation by 2 other members of our group. This eventually took a great amount of time from our research period.

Informed Consent

According to YouTube's policy, it's fair to use any published copyrighted videos without permission for educational and research purpose [60].As long as privacy is concerned, we didn't break anyone's privacy during the data collection process even we have the consent of our friends to use their gameplay records noted in legal stamp papers as proof.

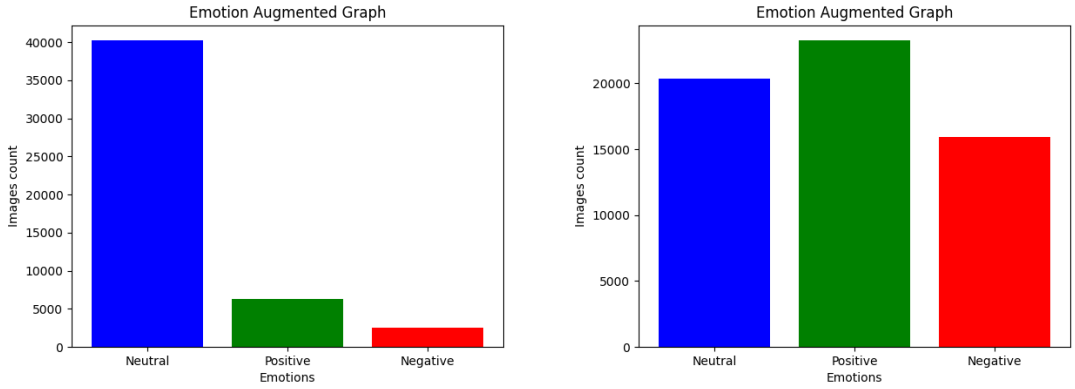
Data augmentation for FER

The collected data had annotated raw 56377 images. A lot of images from different videos were also extracted. However, after the analysis, it is found that most of our images were neutral. We have given the numbers in table 4.1.

Emotion	Positive	Neutral	Negative	Total
Raw	7541	46227	2609	56377
Augmentated	23249	20359	15945	59553
Training	19230	16283	12780	48293
Test	4019	4076	3165	11260

Table 4.1: Data Table

For training the model, such difference would make the model not much effective to detect emotion properly. That's because, if 75% of the images are neutral, then when going to test them on our validation data, there is more chance that our predicted result can be wrong. Because of that, balancing was necessary. We have augmented our data in such a way, so that we get a well balanced data to train the model. Besides, we have collected many game streamers face and annotated them to increase the variety of our dataset. We have used ImageDataGenerator from the library to augment the image. We took parameter of rescale=1./255, featurewise_center=False, featurewise_std_normalization=False, rotation_range=10, width_shift_range=0.1,height_shift_range=0.1, zoom_range=0.1, horizontal_flip=True. These parameters are generated by our augmented images. After the augmentation, we have deleted some redundant images from neutral, so that it can be balanced properly. Then the total images with augmentation were 59553 images. In figure 4.5 and table 4.1 we have given necessary information.



(a) Raw image bar chart

(b) Augmented image bar chart

Figure 4.5: Dataset of extracted annotated images

Data Pre-process for FER

Our collected data were collected to maintain sequence. From this, we needed to build the dataset which are divided to train and test. We separated them 80:20

as train:test data. In each train and test, there are three folder which are labeled as negative, neutral, positive. These represents our three emotions. As our data is collected from video, our images have a lot of redundant images. To balance our data, we have removed many redundant images from neutral as we mentioned above. Also, we had collected the images automatically and we are trying to maintain the sequence, we have collected images which don't have a player in that frame. However, this could harm our model if we don't filter it out. Thus, we have removed all the images without any face or blur images. After that, to train our model, we have converted images to 128*128 size. It saves a lots of time to train our model.

4.1.2 Speech Emotion Recognition (SER) dataset

To train our machine learning models for the task at hand, we have employed four distinct datasets: RAVDESS, SAVEE, TESS, and CREMA-D

4.1.3 Selection of Datasets

Researchers have built many simulated datasets for the purpose of Speech Emotion Recognition (SER), which have proven to be valuable tools for facilitating effective result comparisons. Prominent databases, such as EMO-DB [61], DES [62], RAVDESS [63], TESS [64], and CREMA-D [65], provide standardized compilations of emotional data. In this investigation, we will use the TESS, CREMA D, RAVDESS, and SAVEE datasets. Although these datasets include a wide range of emotional categories, it is important to note that they may have a tendency towards synthetic emotions, deviating to some extent from the emotions expressed in normal, daily discussions.
The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) is a comprehensive collection of audio and visual recordings that capture emotional expressions in both speech and song. The RAVDESS dataset is notable for its broad coverage of many emotions, including happy, sorrow, wrath, fear, surprise, disgust, tranquillity, and neutrality. The compilation has audio recordings from a total of 24 performers, with each actor delivering a set of 104 phrases [63]. The RAVDESS dataset provides a comprehensive collection of 2496 audio clips, showcasing a wide range of samples that include diverse emotional intensities and vocal modalities. These modalities encompass both regular speaking voices and singing voices. The use of North American English accents in RAVDESS serves to differentiate it, making it particularly advantageous for the assessment of instances that need this particular dialect.

The Toronto Emotional Speech Set (TESS) is a collection of emotional speech recordings consisting of six different categories.The TESS dataset was mainly intended to investigate the influence of age on the detection of emotions. The theatrical production TESS showcases the acts of two female performers who range in age, around 60 and 20 years old. The play incorporates a total of seven distinct emotions, which are conveyed via the delivery of 200 neutral lines. The emotions included in this set are anger, pleasant surprise, disgust, pleasure, sorrow, fear, and neutrality. The labels of the dataset were determined by evaluating a group of 56 undergraduate students. The sentences chosen in the Toronto Emotional Speech Set (TESS) dataset was evaluated for a threshold of 66%, which indicates high degree

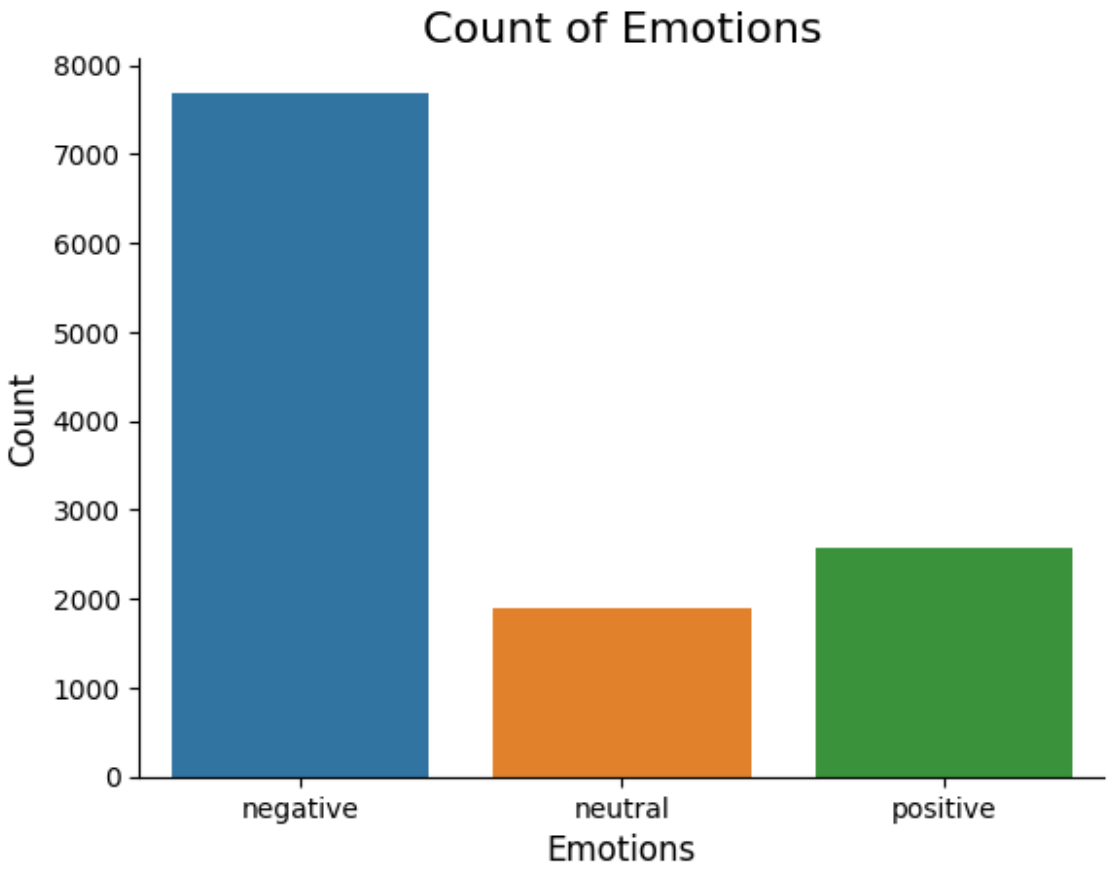


Figure 4.6: Emotion dataset distribution for SER

of confidence while labeling [64].

Crowd-Sourced Emotional Multi-modal Actors Dataset(CREMA-D) also known as CREMA-D dataset is widely used for speech emotion recognition in the field of research. As the name suggests the data in this dataset was collected from diverse group of people and mostly from online platform which is best suited for training and predicting in speech emotion recognition models. This model specially developed for classifying different emotions of an individual which is much needed in our research paper. Since, our paper is interested in finding range of emotions from a player's voice recordings which include position, negative and neutral speeches this dataset was helpful for training our models for identifying emotions of different steamers. With wide range of emotions like: neutrality, happiness, anger, sorrow, fear, and disgust, this dataset includes 7442 samples. This data was collected from 91 actors from diverse group of peoples. In this crowd-sourcing process, 2443 individuals participated in as raters for this dataset which added more diverse opinion and perspectives. While creating the dataset, two types of ratings were assigned for each sample. They are Emotion Category Rating and Intensity Level Rating. To ensure the integrity of the dataset, total of 223260 ratings were gathered from 2443 participants [65].
The Survey Audio-Visual Expressed Emotion (SAVEE) is designed with particular features for emotion recognition in audio visual contents. This assessment tool mainly emphasized on analyzing emotional expressions, which is now widely

used in researches for categorizing and recognizing emotions from audio [66]. This dataset was collected from four male English actor's audio recording, consisting of seven different emotions like : neutrality, happiness, anger, sorrow, fear, and disgust. Each actor produced 120 verbal expressions, leading to a cumulative count of 480 phrases. The incorporation of visual elements was achieved by using a total of 60 markers placed on the faces of the performers. The dataset comprises 15 phonetically-balanced statements for each emotion category, including both common and emotion-specific phrases, as well as generic utterances. The use of the SAVEE dataset has been applied by researchers to evaluate and assess different techniques for extracting features, using deep learning methodologies, utilizing gradient boosting classifiers, and integrating multisource information fusion [67].

Data Preparation for SER

We have converted all the data into a .wav file because our model works on a .wav file. Also, we have segmented all of the audio into 4 sec segments as our video's frame. We are here to identify only positive, negative and neutral emotion. Hence, we made a custom categorisation for the dataset. We have taken happy and surprise as positive emotion, neutral and calm as neutral emotion, fear, angry, disgust, sad as negative emotion. The datasets were organized in different way. Because of that, we have unpacked data in such a way, so that we can maintain the above way.
For understanding the data, we have followed ravdess data, the third from the left separated by hifen was the emotion tag. For example, 03-01-01-01-02-01-04.wav. Here, 01 from third left, (03-01-[01]-01-02-01-04.wav) represent emotion. Here, 01 means neutral. In Creama D, also third from the left separated by underscore (_) was the emotion tag. For instance, 1001_IEO_NEU _XX.wav. Here, third part is NEU, which represent neutral. The similar approach was taken in TESS data. For example, OAF_bath _happy.wav, here emotion is present in third part of the underscore separation. However, in SAVEE dataset, name was separated by underscore. In the second part of the first letter is the emotion tag in this dataset. For example, DC_a08.wav, in here, a means angry, which is taken as negative emotion.

Data augmentation for SER

In total using three techniques such as noise, pitching and stretching we have done our augmentation. Before augmentation we had a total, 12162 sounds containing emotions. We used noise() function, which added the noise into our data. Similarly pitching and stretching were added by their functions. After augmentation we got three times that which is 36486 sounds.

Data Pre-processing for SER

After the augmentation, we followed the following steps before feed the data to our models:

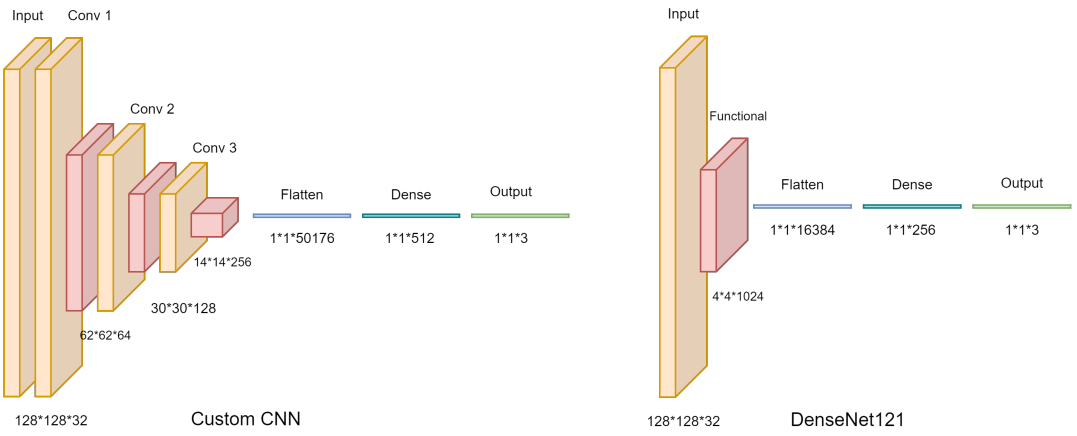
- Features extraction: Extracting the features from the raw data. Here we extracted, Mel-Frequency Cepstral Coefficients (MFCC), Mel Spectrogram, Chroma Short-Time Fourier Transform, Root Mean Square, Zero Crossing Rate.

- DataFrame Creation and CSV Export: We then created a dataframe. After the creation, we exported them to csv format. From the csv format, we manipulated and used the data.
- One-Hot Encoding: We used one hot encoding to convert the categorical form to a binary format, which will give advantages to controlling the data.
- Train-Test Split: After that, we split the data in train-test format.
- Standard Scaling: We scale the data with standard scaling. As the dataset's column differs from one and another in a great number, thus using this make them all into a similar measurement.
- Reshaping for Input: Lastly reshaping the input for giving input to models.

4.2 Model Specification

4.3 Models used in Facial Expression Recognition

4.3.1 Custom CNN for facial emotion recognition



(a) Custom CNN (b) DenseNet121

Figure 4.7: Model structure

We have trained our custom CNN with our own data. We have taken our input layer with 32 filters with $kernel_size3, 3$. The activation was used by ReLU. The input shape was 128*128*3. Total 3 hidden layers were used here. All convolutional blocks had a Con2D and a MaxPooling2D. Conv2D had same kernel size of 3*3 and activation function used ReLU. However, we have increased filter size by double in each hidden layer. Dropout was set to 0.2. Dropout is a regularization technique which prevents overfitting. It randomly sets a fraction of input units to zero during each forward and backward pass. Also used batch normalization. It normalizes the activation which allows to stabilize and speed up training.
After the hidden layer, we have a flatten layer which flattened the last layer to

1D vector. Then we used a Dense layer with 512 neurons with activation function ReLU. After that we used another Dense layer as our output which has 3 neurons as we have three emotions to detect. Here, softmax activation was used which means it predicts for each class. Thus we get the final prediction.

4.3.2 Densenet121 Model Implementation with Transfer Learning

We have taken input image shape as 128*128*3. We didn't include the top classification layer by $include_top = False$. We have taken pre-trained weights from ImageNet Dataset. Besides, except for the last 4 layers, all the layers were freezed which means it couldn't be trainable. This allows training classification layers for specific tasks. We have added a dropout layer 0.6 which helps to prevent from overfitting the model. Then a fatten layer was used which flattened the last layer to 1D vector. This converts them to fully connected layers. Then the first dense layer which has 256 neurons with ReLU. It learned from the last layers to flatten features. Then the second dense layer has 3 units for 3 emotions which has softmax, which one predicts the class.

4.3.3 VGG16 Model Implementation trained with ImageNET

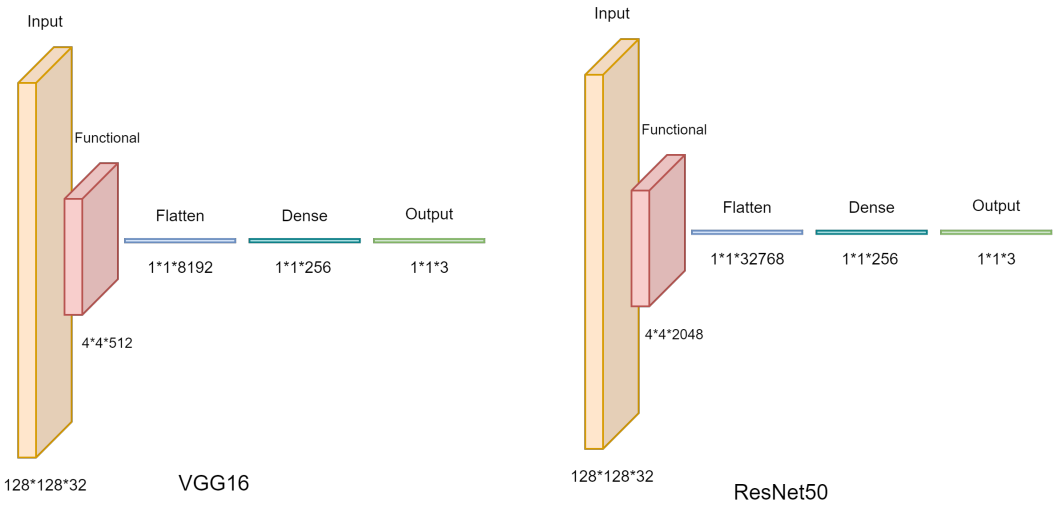
Here we have taken input as the shape of 128*128*3 as previous models. It also didn't include top classification layers to make it trainable. It takes pretrainted model weight from the ImageNet dataset. Besides that, it also freezes all layers except the last four layers. Dropout rate remained same as before, 0.2 to prevent overfitting the model. Then, it flattens with a flatten layer which converts all to a 1D vector. First dense layer has 256 neurons to take the features from the upper flatten layer. Second dense layer has 3 neutrons for 3 emotions for predicting class with softmax activation function.

4.3.4 Resnet50 Pre-trained Model Implementation

Taken image input was 128*128*3. Top classification layer is also not included. Weight was taken from the ImageNet dataset. Except the last four layer, all other layers were freezed, which will allow us to fine tune for our own classification problem. Dropout rate was 0.2. Then a flatten layer was added to make everything in a 1D vector. Dense layer had 256 neurons to take features from the flatten layer. Also, in the second dense layer, 3 neurons predict a class of three emotions.

4.3.5 VGG19 Implementation with Weights from ImageNet

In this model, we have taken image input as 128*128*3. The top classification layer is also not included. The weight was taken from the ImageNet dataset. The layer were freezed the layer except the last four, which will allow to fine tune for our own classification problem. Dropout rate was set to 0.2. Moreover, a flatten layer was added to make everything in a 1D vector. Dense layer had 256 neurons to take



(a) VGG16

(b) ResNet50

Figure 4.8: Model structure

features from the flatten layer. Also, in the second dense layer, 3 neurons predict a class of three categoty emotions.

4.3.6 InceptionresnetV2 trained with ImageNET Implementation

We have taken image input as 128*128*3. The top classification layer is also not included. The weight was taken from the ImageNet dataset. The layer were freezed the layer except the last four, which will allow to fine tune for our own classification problem. Dropout rate was set to 0.2. Moreover, a flatten layer was added to make everything in a 1D vector. Dense layer had 256 neurons to take features from the flatten layer. Also, in the second dense layer, 3 neurons predict a class of three categoty emotions.

4.3.7 MobileNetV2 Implementation for Lightweight

We have taken image input as 128*128*3. The top classification layer is also not included. The weight was taken from the ImageNet dataset. The layer were freezed the layer except the last four, which will allow to fine tune for our own classification problem. Dropout rate was set to 0.6. Moreover, a flatten layer was added to make everything in a 1D vector. Dense layer had 256 neurons to take features from the flatten layer. Also, in the second dense layer, 3 neurons predict a class of three categoty emotions.

4.3.8 Xception Model Implementation:Extream Inception

We have taken image input as 128*128*3. The top classification layer is also not included. The weight was taken from the ImageNet dataset. The layer were freezed the layer except the last four, which will allow to fine tune for our own classification

problem. Dropout rate was set to 0.2. Moreover, a flatten layer was added to make everything in a 1D vector. Dense layer had 256 neurons to take features from the flatten layer. Also, in the second dense layer, 3 neurons predict a class of three categoty emotions.

4.3.9 EfficientNet-B0 Model Efficienct Implementation

We have taken image input as 128*128*3. The top classification layer is also not included. The weight was taken from the ImageNet dataset. The layer were freezed the layer except the last four, which will allow to fine tune for our own classification problem. Dropout rate was set to 0.2. Moreover, a flatten layer was added to make everything in a 1D vector. Dense layer had 256 neurons to take features from the flatten layer. Also, in the second dense layer, 3 neurons predict a class of three categoty emotions.

4.4 Model used in Speech Emotion Recognition

4.4.1 CNN for Speech Emotion Recognition

For the SER model, we had to go through some stages. We have collected and pre processed the data. Then we have used onehotencoder to normalize the data. After That, we split all the data to train test data. Also we used a standard scaler to scale the data. Besides, we also expand the dimension of data to make it compatible with our model.
We have extracted five features, Zero Crossing Rate, Chroma_stft, MFCC, RMS(root mean square) value, Mel Spectrogram to train our model. We can't directly train audio to our model to predict. However, this helps to convert our data to some numbers which our model can learn and successfully predict emotion from voice.
A custom CNN model was used to train the datasets. Total 4 convolutional blocks were used. As the audio data is not 2D, thus we used Conv1D and MaxPooling1D. Except for the output layer, all the layers had activation ReLU. Input layer and second layer had filter size 256. For every conv layer, strides used 1, and for max-pooling layer, stride was 2. Kernel size and pool size was always 5. Third and fourth layer had 128 and 64 filter sizes. Dropout layer was added 0.5. After that a flatten layer was added. First dense layer had a unit of 32. Last dense layer had 3 neurons as our output emotion was 3.

4.4.2 LSTM Model Implementation for Sequential

The input size was the x_train.shape(). In the first LSTM layer, total hidden cells were 256. In next LSTM layer, the same 256 hidden cells were taken. In third LSTM layer, hidden cells were taken 128 units. For this three LSTM layer, the return sequence were always true. Then, we have added a dropout layer of 0.5, which means 50% will be set to zero. After that, added LSTM layer which have hidden cells unit of 64, and the return sequence was false. Which means, only final output will be returned. Then, we added dense layer 32 which refers to it became a fully connected layer with 32 neuron with ReLU activation function. We then,

added another dropout layer to prevent overfitting. Lastly a dense layer of 3 neuron to classify output class.

4.4.3 GRU Model for Efficiency

To train our model, GRU model was used to find out whether it is more optimized than LSTM or not. We followed the exact same approach so that we can figure out which model is performing better in our classification.

4.4.4 CNN-LSTM Hybrid Model for Speech Emotion Recognition

In this model, Conv1D layer added with the x_train.shape as input. Conv 1 dimension works on sequential data. Total 128 filters were applied with the kernel size of 5. Stide taken was 1. Padding remain same and activation function was ReLU. Then, a maxpooling layer was added. with the pool size of 5 and stride of 2, padding same. Then added another Conv1D and MaxPooling1D layer sequentially with the same configuration as before. Then, another Conv1D and MaxPooling1D added which were just the filter number changed in Conv1D. Then, a dropout layer of added with 0.2. Then, a LSTM layer was added of 32 units with the return sequence False. Then a dropout layer with 0.5.Lastly a dense layer to classify 3 emotion.

Chapter 5

Result analysis

5.1 Models result

5.1.1 FER model results

DenseNet121

DenseNet121 was compiled with 'adam' optimizer. We have run our model for 100 epochs. After 100 epochs we got a loss of 0.3116, train accuracy 0.8746, precision 0.8816, recall 0.8678, f1 score 0.8745, validation loss 0.5172, validation accuracy 0.8090, validation precision 0.8152, validation recall 0.7990, validation f1 score: 0.8069

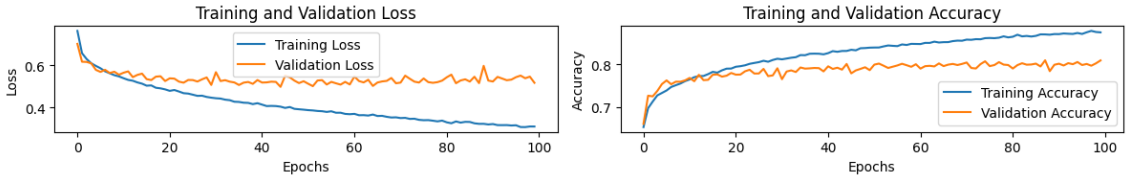


Figure 5.1: Model with DenseNet121 evaluation

Custom CNN

Our custom CNN was compiled with 'adam' optimizer. We have run our model for 100 times. We have 0.8966 accuracy on training, and loss was 0.2788. The precision 0.9020, recall 0.8913, f1 score 0.8965. Validation accuracy was 0.7748 and loss was 0.9434, validation precision was 0.7828, validation recall 0.7626, validation f1 score 0.7724.

VGG16

VGG16 was compiled with 'adam' optimizer. We have run the model for 100 epochs. After 100 epochs we got a training loss of 0.1020, train accuracy 0.9724, precision 0.9752, recall 0.9701, f1 score 0.9726 and validation loss 2.1304, validation accuracy 0.7661, validation precision: 0.7693, validation recall 0.7627, val f1 score 0.7659.

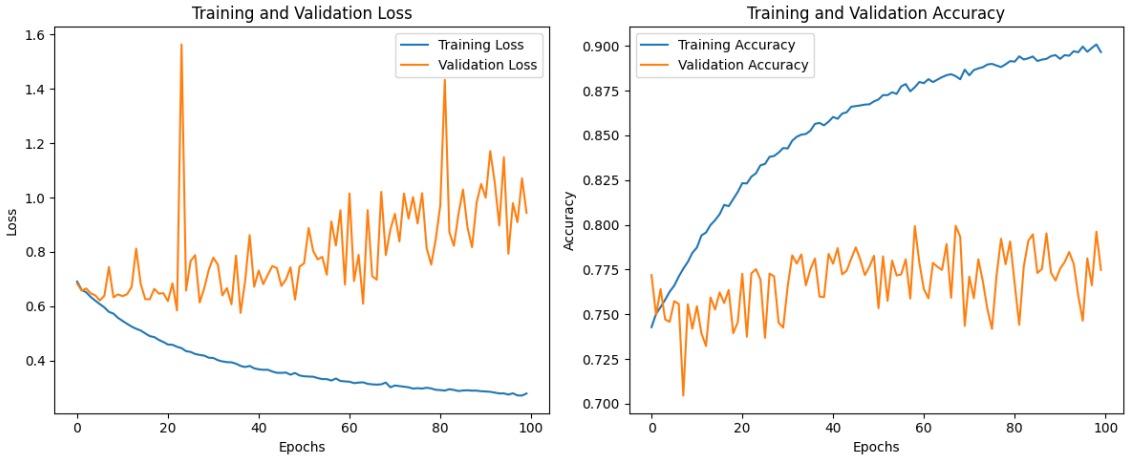


Figure 5.2: CNN model for FER evaluation

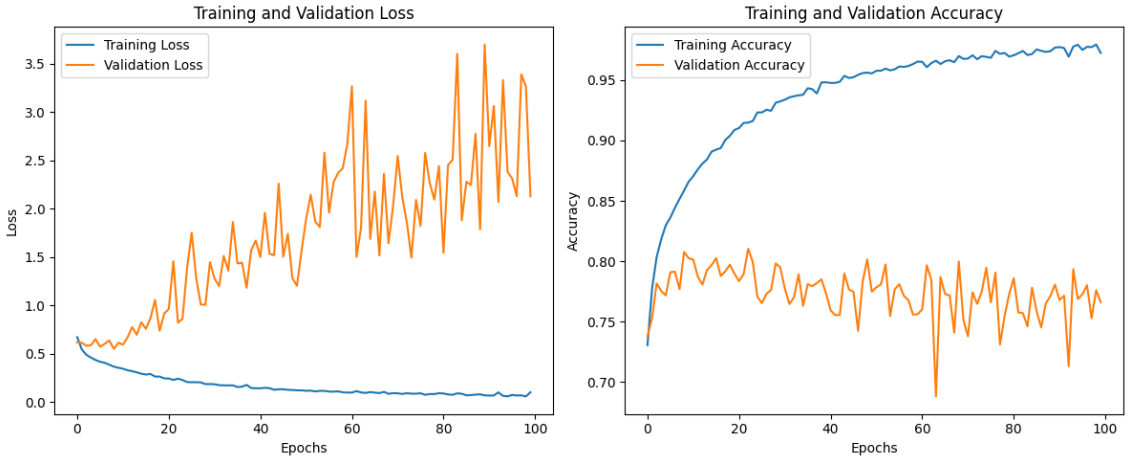


Figure 5.3: Model with VGG16 evaluation

ResNet50

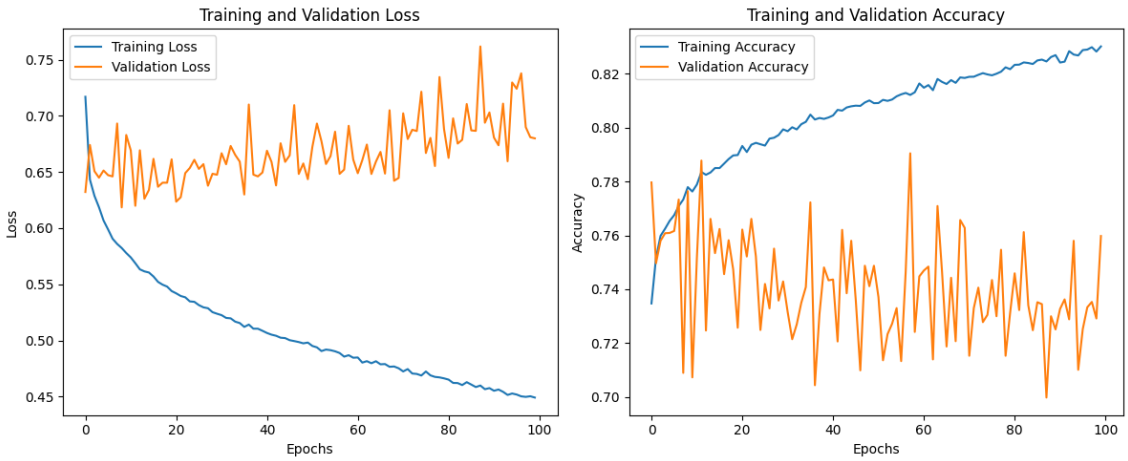


Figure 5.4: Model with ResNet50 evaluation

ResNet50 was compiled with the 'adam' optimizer. We have run the model for 100 epochs. After 100 epochs we got a training loss of 0.4491, train accuracy 0.8301,

precision 0.8425, recall 0.8150, f1 score 0.8283, validation loss 0.6800, validation accuracy 0.7598, validation precision 0.7718, validation recall 0.7452, validation f1 score 0.7581

VGG19

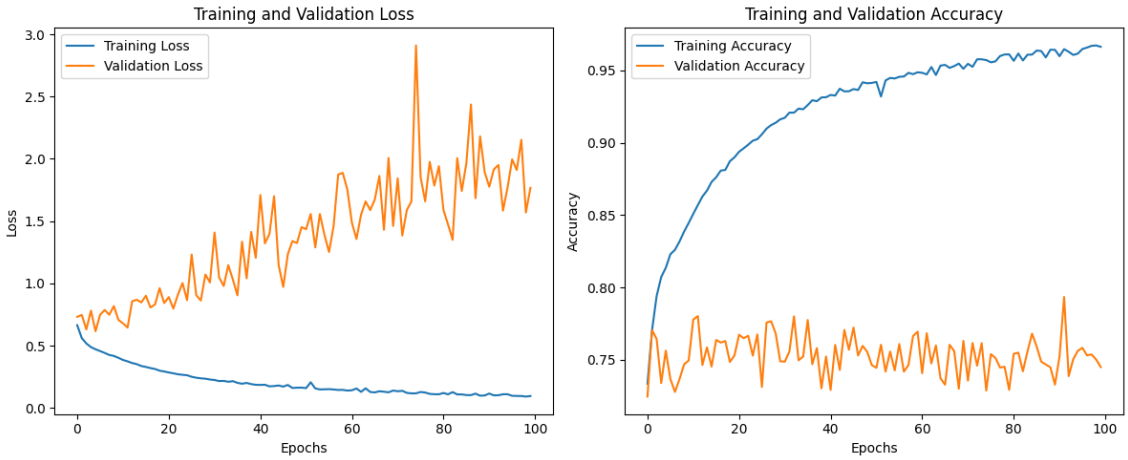


Figure 5.5: Model with VGG19 evaluation

VGG19 was compiled with the 'adam' optimizer. We have run the model for 100 epochs. After 100 epochs we got a training loss of 0.0970, train accuracy 0.9664, precision 0.9688, recall 0.9647, f1 score 0.9667, validation loss 1.7678, validation accuracy 0.7449, validation precision 0.7490, validation recall 0.7406, validation f1 score 0.7447

InceptionResnetV2

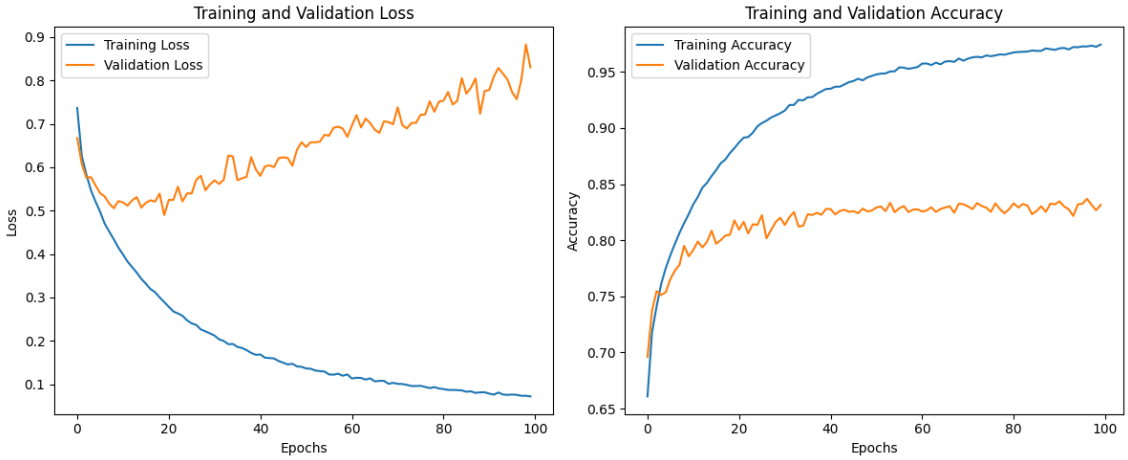


Figure 5.6: Model with InceptionResnetV2 evaluation

InceptionResnetV2 was compiled with the 'adam' optimizer. We have run the model for 100 epochs. After 100 epochs we got a training loss of 0.0718, train accuracy

0.9743, precision 0.9753, recall 0.9733, f1 score 0.9743, validation loss 0.8304, validation accuracy 0.8315, validation precision 0.8333, validation recall 0.8305, validation f1 score 0.8318

MobileNetV2

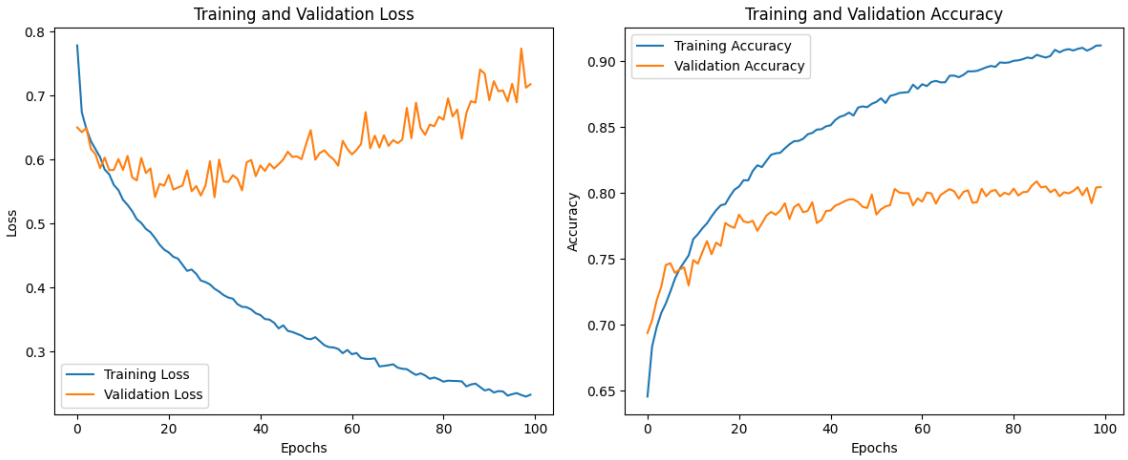


Figure 5.7: Model with MobileNetV2 evaluation

MobileNetV2 was compiled with the 'adam' optimizer. We have run the model for 100 epochs. After 100 epochs we got a training loss of 0.2323, train accuracy 0.9121, precision 0.9187, recall 0.9055, f1 score 0.9119, validation loss 0.7179, validation accuracy 0.8044, validation precision 0.8103, validation recall 0.7995, validation f1 score 0.8048

Xception

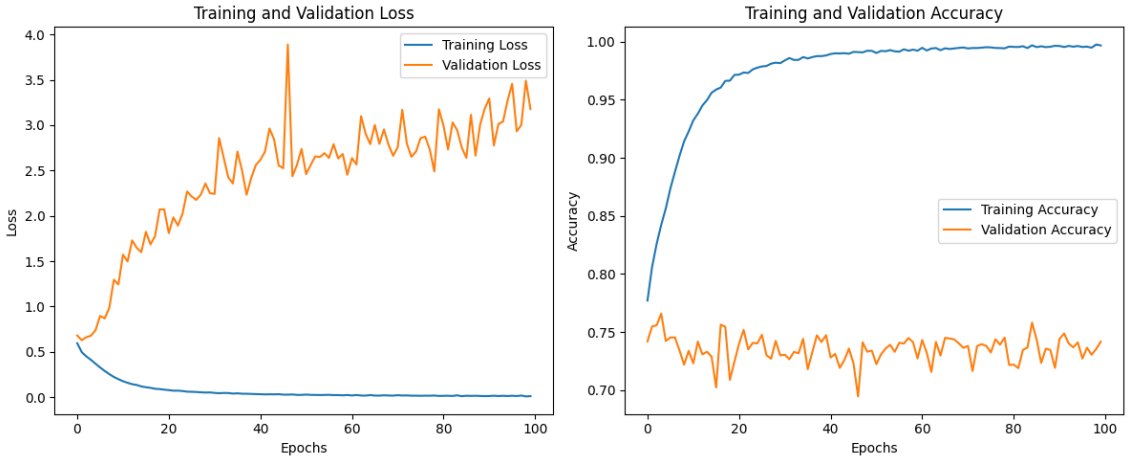


Figure 5.8: Model with Xception evaluation

Xception was compiled with the 'adam' optimizer. We have run the model for 100 epochs. After 100 epochs we got a training loss of 0.0126, train accuracy 0.9967, precision 0.9968, recall 0.9966, f1 score 0.9967, validation loss 3.1769, validation

accuracy 0.7417, validation precision 0.7427, validation recall 0.7413, validation f1 score 0.7420

EfficientNetB0

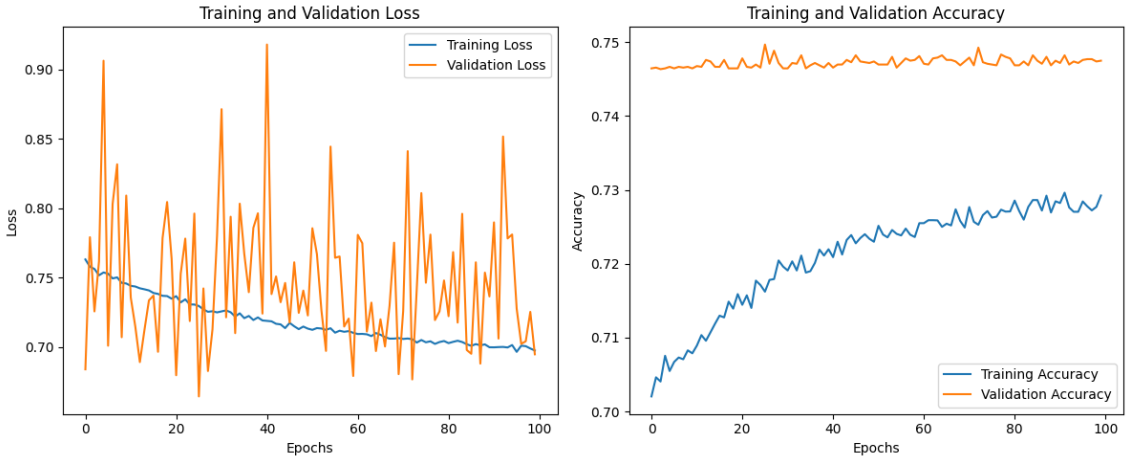


Figure 5.9: Model with EfficientNetB0 evaluation

EfficientNetB0 was compiled with the 'adam' optimizer. We have run the model for 100 epochs. After 100 epochs we got a training loss of 0.6976, train accuracy 0.7292, precision 0.7959, recall 0.6455, f1 score 0.7116, validation loss 0.6949, validation accuracy 0.7475, validation precision 0.7756, validation recall 0.7314, validation f1 score 0.7525

5.1.2 FER Model Result Analysis

Model	Epochs	Train Loss	Val Loss	Train Acc.	Val Acc
DenseNet121	100	0.3116	0.5172	0.8745	0.8090
Custom CNN	100	0.2788	0.9434	0.8966	0.7748
VGG16	100	0.1020	2.1304	0.9724	0.7661
ResNet50	100	0.4491	0.6800	0.8301	0.7598
VGG19	100	0.0970	1.7678	0.9664	0.7449
InceptionResnetV2	100	0.0718	0.8304	0.9743	0.8315
MobileNetV2	100	0.2323	0.7179	0.9121	0.8047
Xception	100	0.0126	3.1769	0.9967	0.7417
EfficientNetB0	100	0.6976	0.6949	0.7292	0.7475

Table 5.1: Model Performance Metrics

Based on the FER models metrics, we can understand that Densenet121 performed overall best. It has the highest validation accuracy of 0.8090. The training accuracy is 0.8745 as well. When training accuracy is much higher than validation accuracy, the model becomes an overfit model. Which means when a new data will be shown to them, the model would like to fail to detect correctly. In this case, many of our experimental models become overfitted. We can see InceptionResnetV2 performed

0.8315 validation accuracy. However, the training acuracy was 0.9743. We didn't choose this model for performing experiment because of most of our experiments are based on unseen data. So, whenever, the model will see the unseen data, it might give wrong outcome. Thus we chose DenseNet121 which has given validation accuracy of 0.8090 and training accuracy of 0.8745. Besides, compared to other models, it has less validation loss too. The reason behind this, because of the transfer learning. DenseNet121 trained on ImageNet dataset. From their, we got many features weight. This directly influenced the performance of model. Moreover, because of its densely connected layer, and the connection between the dense block, its efficiency increased and accuracy improved. Also, it reduces the vanishing gradient problem as well. Thus we have chosen the best model Densenet121 for FER.

5.1.3 SER model results

CNN

Our CNN model was compiled with 'adam' optimizer. We have run our model for 100 times. The accuracy was achieved upto 0.7592 accuracy on training, and loss was 0.5271. Validation accuracy was 0.7552 and loss was 0.5470 .

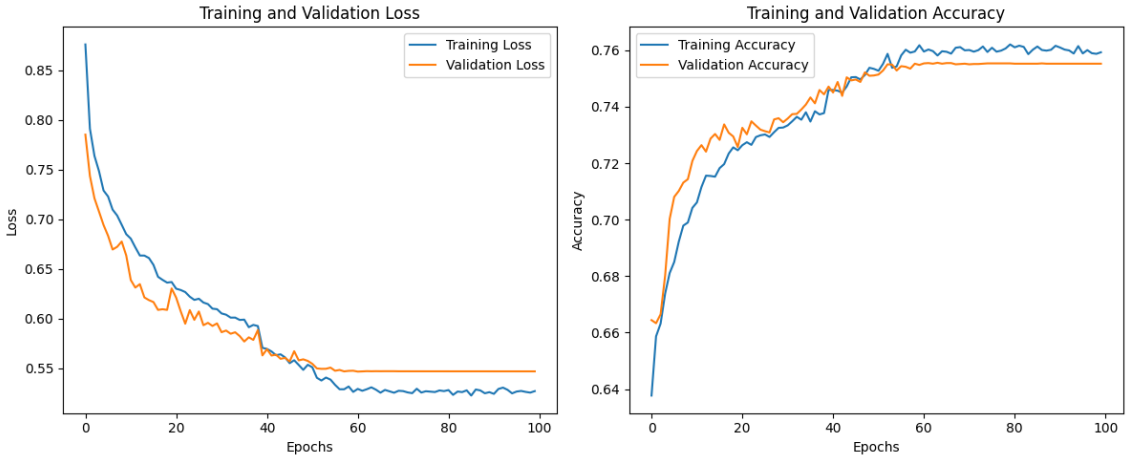


Figure 5.10: CNN model for SER evaluation

CNN-LSTM

Our CNN-LSTM model was compiled with 'adam' optimizer. We have run our model for 100 times. The accuracy was achieved upto 0.7464 accuracy on training, and loss was 0.5670. Validation accuracy was 0.7505 and loss was 0.5682.

LSTM

Our LSTM model was compiled with 'adam' optimizer. We have run our model for 100 times. The accuracy was achieved upto 0.7050 accuracy on training, and loss was 0.6767. Validation accuracy was 0.7032 and loss was 0.6757.

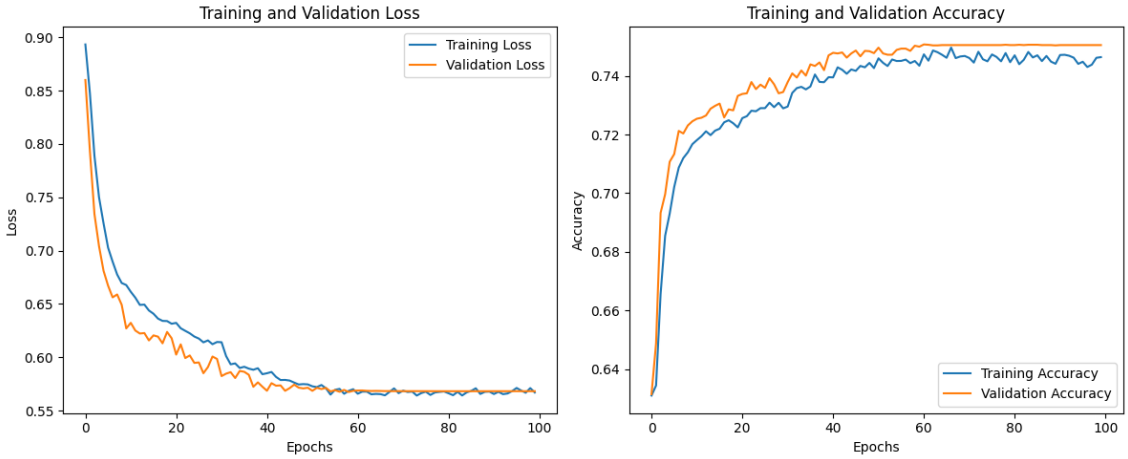


Figure 5.11: CNN-LSTM model for SER evaluation

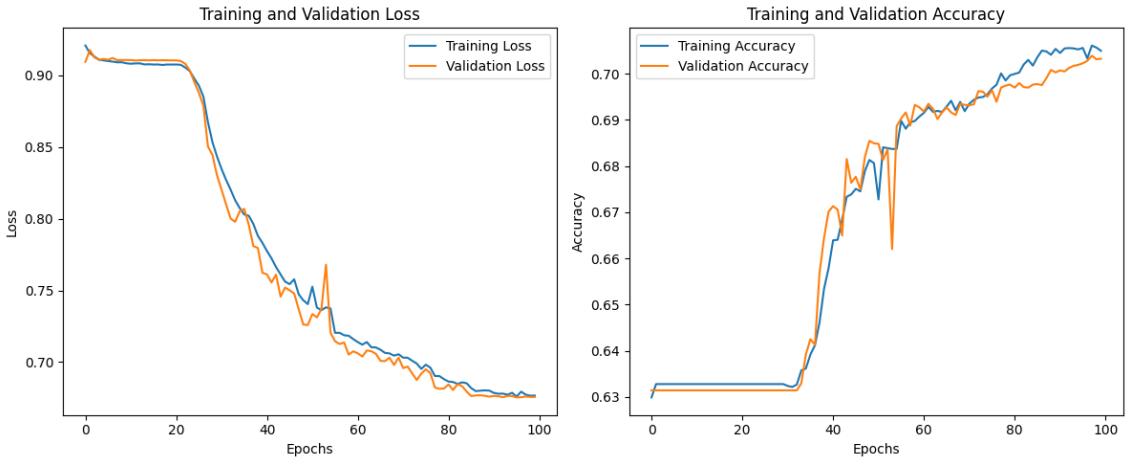


Figure 5.12: LSTM model for SER evaluation

GRU

Our GRU model was compiled with 'adam' optimizer. We have run our model for 100 times. The accuracy was achieved upto 0.9974 accuracy on training, and loss was 0.0072. Validation accuracy was 0.7370 and loss was 3.5459.

Model	Epochs	Train Loss	Val Loss	Train Acc	Val Acc
Custom CNN	100	0.5271	0.5470	0.7592	0.7552
CNN-LSTM	100	0.5670	0.5682	0.7464	0.7505
LSTM	100	0.6767	0.6757	0.7050	0.7032
GRU	100	0.0072	3.5459	0.9974	0.7370

Table 5.2: Model Performance Metrics for SER

Again, based on the SER models metrics, our Custom CNN model perform best. It has the highest validation accuracy with the lowest validation and training loss. CNN-LSTM combined model also perform well. Both of these two models gave similar results. However, the other two model performed comparatively less. It is

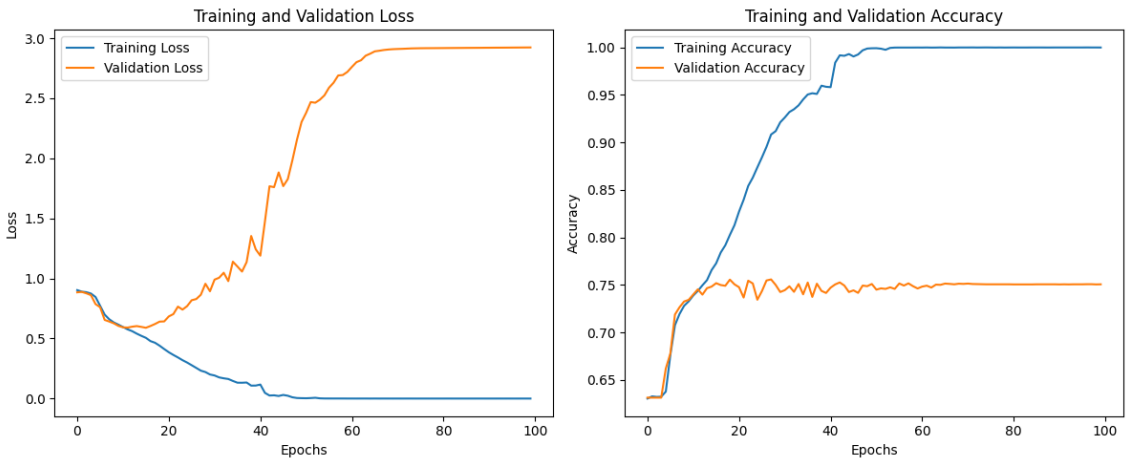
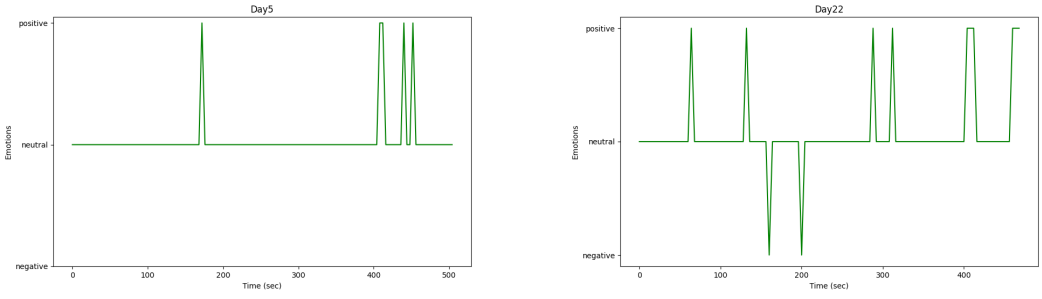


Figure 5.13: GRU model for SER evaluation

because of the dataset nature. Our dataset is not in sequential manner. We know, LSTM model perform well on sequential dataset. Hence, the models related to lstm couldn't perform well. Thus we have chosen the best model and CNN for SER.

5.2 Emotion recognition

From the training model, we have chosen the best model Densenet121 for Facial Emotion Recognition and CNN for Speech Emotion Recognition.
For Facial Emotion Recognition, we have taken images by interval from the gameplay videos (which included the face of the players). The images by frames were then predicted for emotion recognition. As the videos were taken by different days, it allowed us to detect the pattern of the player's emotion. The patterns varied from player to player. Also, we found that some players showed positive emotion during their gaming, some showed negative emotion while gaming. Although the neutral pattern was seen vastly across all players. First let us look at the graphs of the pattern of a single player whose reaction may vary from day to day.

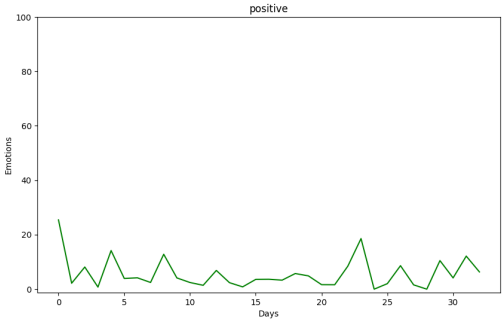


(a) Player 6, Day 5 emotions

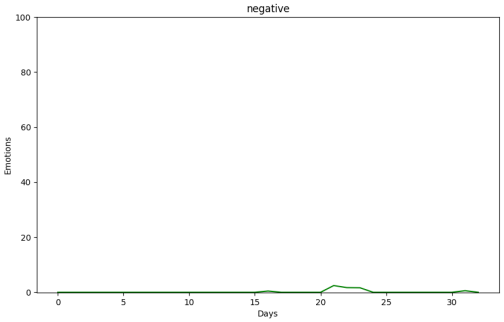
(b) Player 6, Day 22 emotions

Figure 5.14: Emotional Condition while Gaming

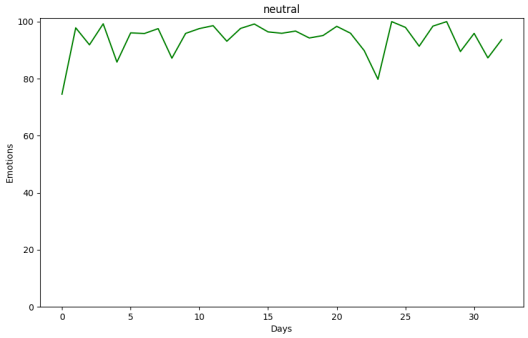
Here, in we can see that out of the three emotions, the percentage of showing neutral emotion is much higher. It can be explained as we have taken First person shooter



(a) Positive



(b) Negative



(c) Neutral

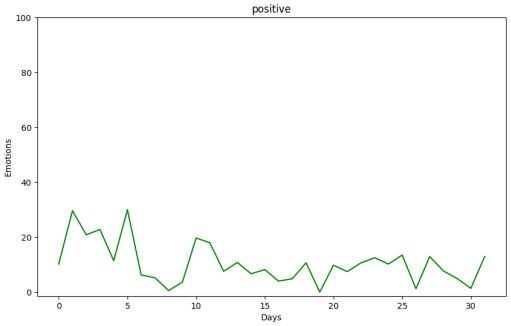
Figure 5.15: Player 6 emotions over course of days

games like Valorant, out of total match time, a player may see action less. Or, even if the player is seeing action (by action, we mean either the player or his/her teammate eliminating the opponent player / getting eliminated by the opponent player), the player may show no emotion during those times unless the match or the round ends. Even after ending the match, the player may show no facial expression, but expressing his/her emotion through speech. That is why the percentage of showing neutral emotion is higher according to our analysis. We will allocate threshold values to judge a player's overall emotional status during the player's playtime for neutral/ positive/ negative. But first, let us look at the case of positive and negative emotions.

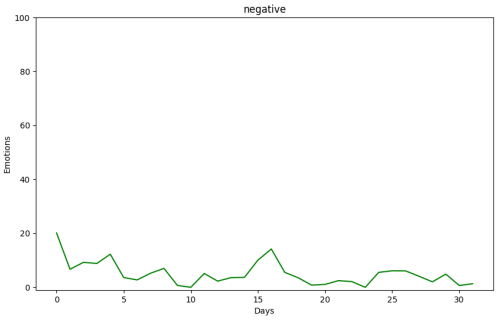
Now, our judgment about a player showing positive or negative emotions during the gaming session, we can tell by looking at the next highest percentage of emotion shown. For player 6, on day 5, it is seen that for the whole time (aside neutral), the player was showing positive emotions. For day 22, the percentage of showing positive emotion was higher than the negative.

Now if we were to judge for only these two days, we can say that video gaming is making positive impacts on player 6. Just like this, we have found out the patterns over a course of multiple days of different players.

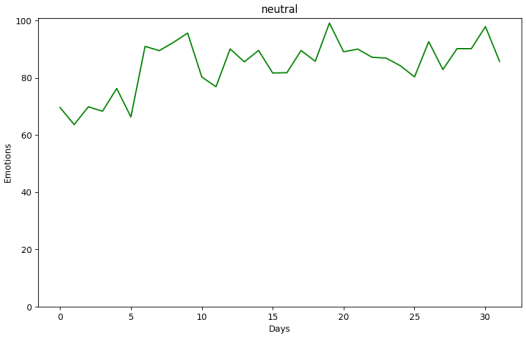
Not all players showed the same type of patterns. One case being a player may show more positive emotions than another player who is also showing positive emotions throughout the course of the day. Another case might be a player showing a mixture of both, that is, showing both positive and negative emotions whose percentages are



(a) Positive



(b) Negative



(c) Neutral

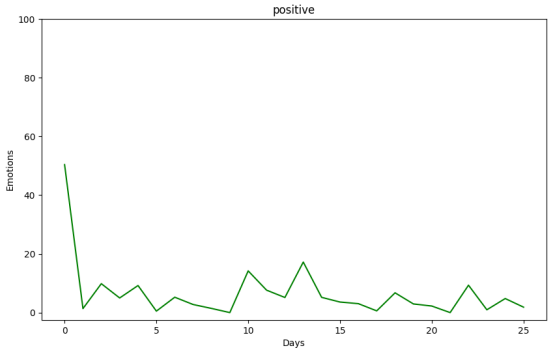
Figure 5.16: Player 9 emotions over course of days

the same, or close to the same. It will be clear if we look at the figure 5.15.
Let us look at another player, Player 9 in figure 5.16:
Before judging, we will take some threshold values. It should be noted that this threshold values are taken just to differentiate between the emotions. This values may change for a huge number of streamer for which we will have huge number of variations in our data.

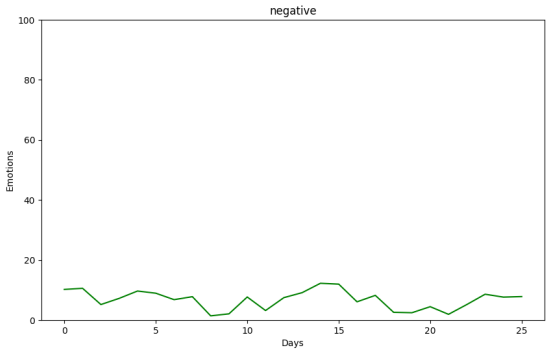
- If Average Neutral emotion for a player is greater than 98, then we will say that the player is showing neutral emotions most. Other emotions are very very low.

- If Average Neutral emotion is less than 98, and the difference between average Positive and Average Negative reaction is less than 5.5, then the player is not showing neutral behavior but showing mixed emotions, containing both positive and negative emotions.

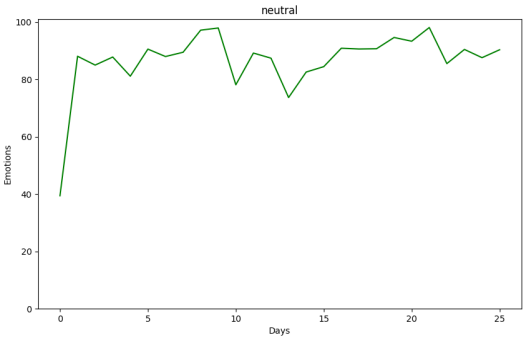
- If Average Neutral emotion is less than 98, and the difference between average Positive and Average Negative reaction is greater than 5.5, then the player is not showing neutral behavior but showing either one of the reactions, either Positive or Negative more.



(a) Positive



(b) Negative



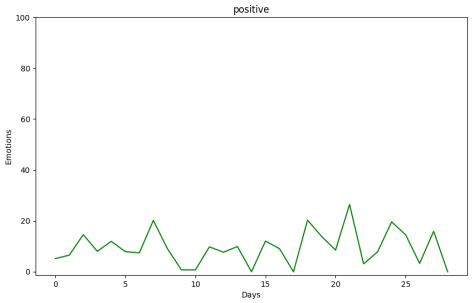
(c) Neutral

Figure 5.17: Player 4 emotions over course of days

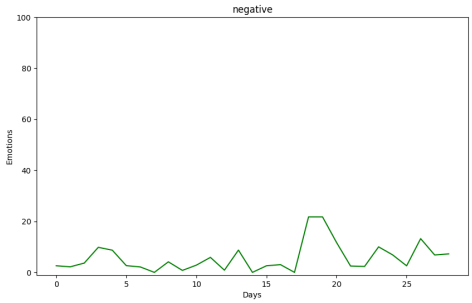
If we look at the case of Player 6, he is now showing more positive emotion (aside neutral; which has been explained earlier) than negative emotion. His percentage of showing negative emotion is very very low. Average positive emotions for the player is 6.6. Average negative emotion for the player is 0.61. Also average neutral emotion for the player is 93.7, which is lower than 98. The difference of negative and positive reaction is 6, which is greater than 5.5. Therefore we can come to the conclusion that video gaming is making a positive impact on player 6.
If we look at the case of Player 9, he is now showing mixed positive and negative reactions.. Average positive emotions for the player is 9.89. Average negative emotion for the player is 4.77. Also average neutral emotion for the player is 81.184, which is less than 98. The difference of negative and positive reaction is 5.13, which is less than 5.5.Therefore we can come to the conclusion that player 9 is having mixed emotions during his gaming time.
Let us look at another player, Player 4 5.17.
Average Neutral for player 4 is 85.30. Average Positive is 9.36, and average Negative is 7.21. Therefore the difference between negative and positive is 2.15, which is less than 5.5. Therefore, this player is also showing mixed emotions.
Let us look at the table of categorization of player Table 5.3:
Therefore, our final conclusion about the judgment would be, we can not tell that video gaming will affect positively or negatively generally among all gamer population. Rather, this pattern varies from player to player.

Name	Avg. Neutral	Avg. Positive	Avg. Negative	Neutral > 98?	(Pos Neg)	Difference < 5.5?	Type
Player 6	93.7	6.6	0.61	No	6	No	Type 2
Player 9	81.184	9.89	4.77	No	5.13	Yes	Type 0
Player 4	85.3	9.36	7.21	No	2.15	Yes	Type 0

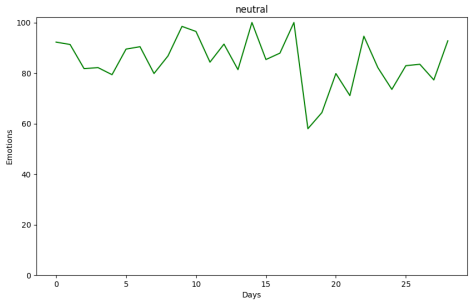
Table 5.3: Categorization of Player Type



(a) Positive



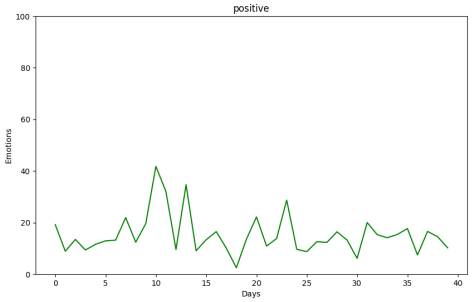
(b) Negative



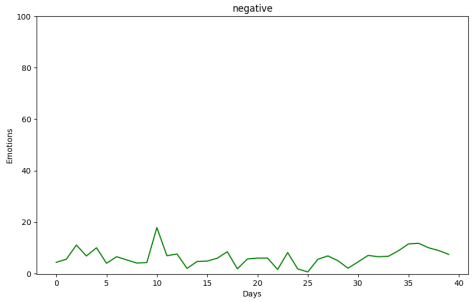
(c) Neutral

Figure 5.18: Player 5 emotions over course of days

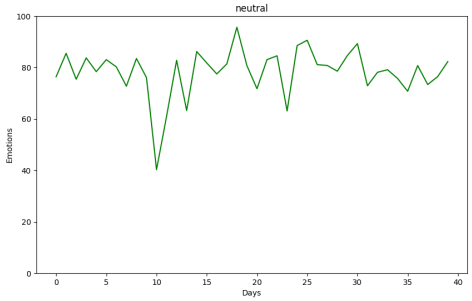
Let us look at some other player's graphs that we have analyzed.
We can see that there are again similar patterns in these graphs. For example, Positive emotion is higher for player 10 if we compare between negative and positive. Same goes for player 8, player 7. While Player 5 is having mixed emotions. Among these streamers, none showed higher negative emotion during their gameplay time. But all the streamers we have taken are not enough if we are to analyze it on a large scale. Because from all the streamers we have watched, they often laughed or joked when they faced negative action (by negative action, we mean getting defeated in-game, objective failure etc.) But if it were possible to take videos from about a 100 streamers and record their data for at least a year, then we could have found higher negative emotion among some players.
Further, we can categorize the streamers by type. For example, here Players 4 and Player 9 both show mixed emotions. So, we can assign the players to type 1 (Mixed/Neutral) streamer. Player 6 can be assigned to type 2 (Positive) streamer category. Another type can be assigned to type 3 (Negative) streamer though we did not find any streamers belonging to this category out of all the streamers we



(a) Positive

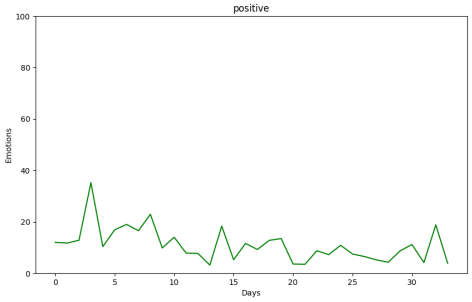


(b) Negative

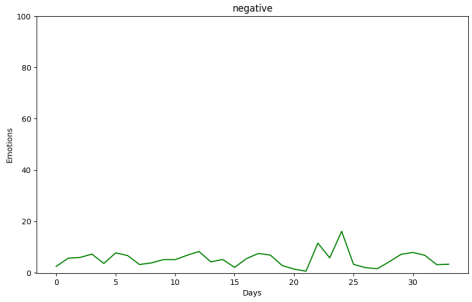


(c) Neutral

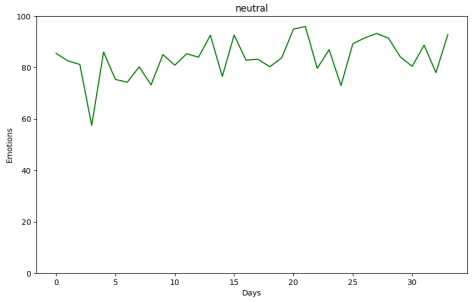
Figure 5.19: Player 7 emotions over course of days



(a) Positive

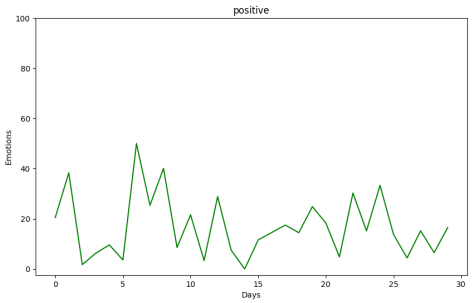


(b) Negative

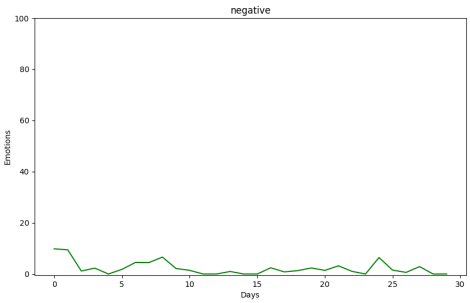


(c) Neutral

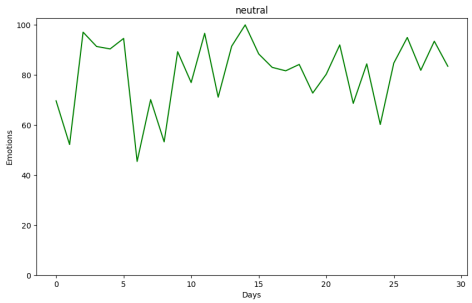
Figure 5.20: Player 8 emotions over course of days



(a) Positive



(b) Negative



(c) Neutral

Figure 5.21: Player 10 emotions over course of days

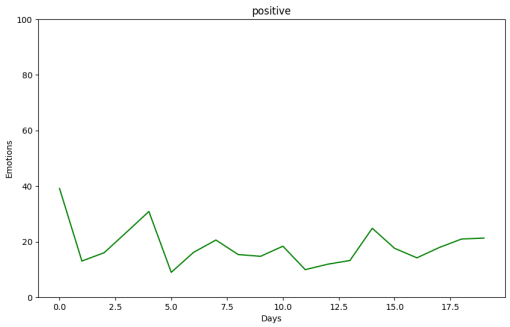
have taken.

Many of the people who love video games, also watch these streamers and then the reactions and emotions of those streamers may affect the viewers when they play the games themselves. Though we can not exactly tell that, for example, a video gamer who is not a streamer, Gamer 1, his emotional changes may be similar to the streamers he watches, or it may not. In order to identify that, we have taken a friend's emotion during his gameplay over a course of time. We are going to name him Gamer 1. The Gamer 1 also follows Type 2 (Positive) streamers.

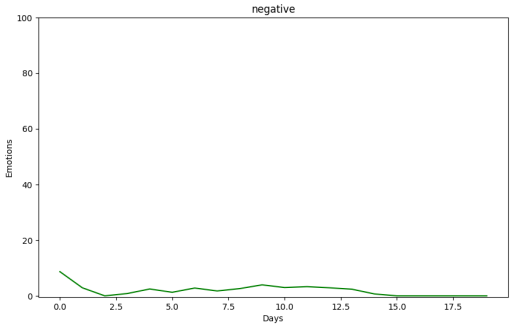
Average Neutral for gamer 1 is 75.23. Average Positive is 16.65, and average Negative is 1.8. Therefore the difference between negative and positive is 14.85, which is greater than 4. Therefore,the gamer is showing positive emotions. It was also mentioned that the gamer follows type 2 (Positive) streamers. Therefore, it is possible that the emotions of those streamers during their streaming time had affected the Gamer 1.

We can bolster our claim by adding the graphs we got from speech emotion recognition. For Gamer 1.

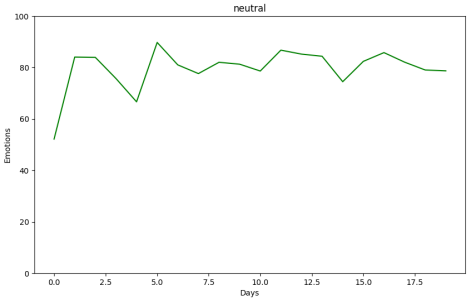
Now, we can again see out of these two days, the number of negative emotions shown through speech is very low. And the number of positive emotions shown is higher. Its the same case for multiple days. Therefore, we are close to similar results to that of the pattern we got from FER results. But we got the best accuracy for FER's Densenet50. Therefore, our judgment was based on that model. Also, the reason our choosing FER for judgment was, while detecting speech emotion recognition,



(a) Positive

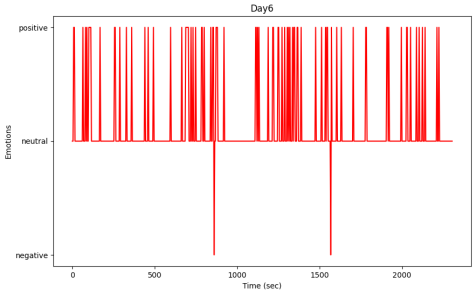


(b) Negative

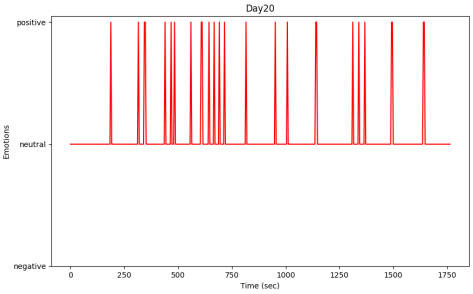


(c) Neutral

Figure 5.22: Gamer 1 emotions over course of days



(a) Gamer, Day 6 emotions



(b) Gamer, Day 20 emotions

Figure 5.23: Emotional Condition while Gaming

not only the player was talking, but also his teammates, and ingame character voices.Therefore, it interfered with actual emotions of the player. As a result, the graph we have given, it is possible that other player's emotions, aside from gamer 1, are also included.

Chapter 6

Future works and Limitation

6.1 Limitations

Facial expression and speech tone recognition algorithms must be understood before being used in gaming. These mistakes or limits should be noted:

- Sarcasm and purposeful Misdirection: Algorithms may not always identify sarcasm or purposeful misdirection in speech tone or facial expressions. Gamers utilize these strategies to deceive opponents or entertain teammates, and algorithms may misread their feelings.

- Context Sensitivity: Facial expression and speech tone recognition algorithms may ignore context. Emotions may vary quickly in gaming, and algorithms may not always account for context, leading to misinterpretations.
- Different people display emotions differently, so what one person sees as sad or furious may be normal for another. Individual variations may be difficult for algorithms to account for.

- Algorithms may create false positives and negatives. They may misinterpret or misidentify feelings. This might misjudge a gamer's mood.

- Ambiguous emotions and sentiments can be interpreted. Algorithms may misclassify small emotional subtleties.
- Environmental variables: Background noise, bad lighting, and other factors might impact speech tone and facial expression recognition. These variables may cause emotional misunderstanding.

- Training Data Bias: Algorithms trained on a dataset that does not appropriately represent gamers' variety and emotional expressions may perform badly for specific demographics or cultures.

- Facial expression recognition captures and analyzes people's faces, raising privacy issues. Such technologies in games may worry gamers.

- Facial expression and speech tone recognition algorithms can reveal players' emotions, but their limits and interpretation mistakes must be considered. Instead than replacing human judgment, these algorithms should be employed to enhance the game experience. To solve privacy and ethical problems, gamers should be informed and consented to using such technologies.

6.2 Future Works

We have seen that we have successfully categorized the multiple streamers in types based on their emotions during video gaming. In our case, type 0 (Negative), type 1 (Mixed/Neutral) and type 2 (Positive). Also were able to establish that a normal video gamer's emotion may be affected by the streams of the types of streamers he watches. This framework can be applied in the following way.

- In future, it will be possible to categorize a vast amount of streamers based on the framework.

- One case can be, a gamer may play any way he likes. We can then detect his emotion over a course of time. He may show either positive, negative, neutral emotion. Let's say, the gamer is showing negative emotion. Then, one can suggest this gamer to watch type 2 (Positive) streamers in the hope to improve his behavior.

- Another case might be, a player might be always showing negative behavior. The player itself or the guardians of the players can then look at the streamers the player follows if he does so. The type of streamers can be found by either from the previous categorized list of streamers, or can be newly categorized using our framework. If it is found that the streamers fall under type 0 (Negative), the guardian or the player themselves can change their view to Type 2 streamers.

- The framework could have been bolstered if more outside factors for emotion aside from gaming could have been taken into account for the streamers which was not possible in this stage. If taken, decisions would be more precise if it was only gaming that was affecting the emotions or both gaming and outside factors.

- Also, we have seen the case with Speech Emotion recognition, where not only the player was talking but his whole team which interfered with getting only the player's emotion. If it were possible to get only the microphone record of the players, the results we would get would only be for those players.

6.3 A Complete Model

Here, we can provide a video or many videos, then our structure can extract the images from the videos. Besides, it also extract audio. After that, the structure can crop the audios. Then, the structure can also load the best model which is suitable for each FER and SER. After that, our structure can predict emotion of the images and the cropped audios. Based on them, it can generate graphs for each day showing that the frequent change of emotion per four second. It also generate a long term changes based on provided multiple videos. Based on them, it can suggest the player about their behavioral changes over the time.

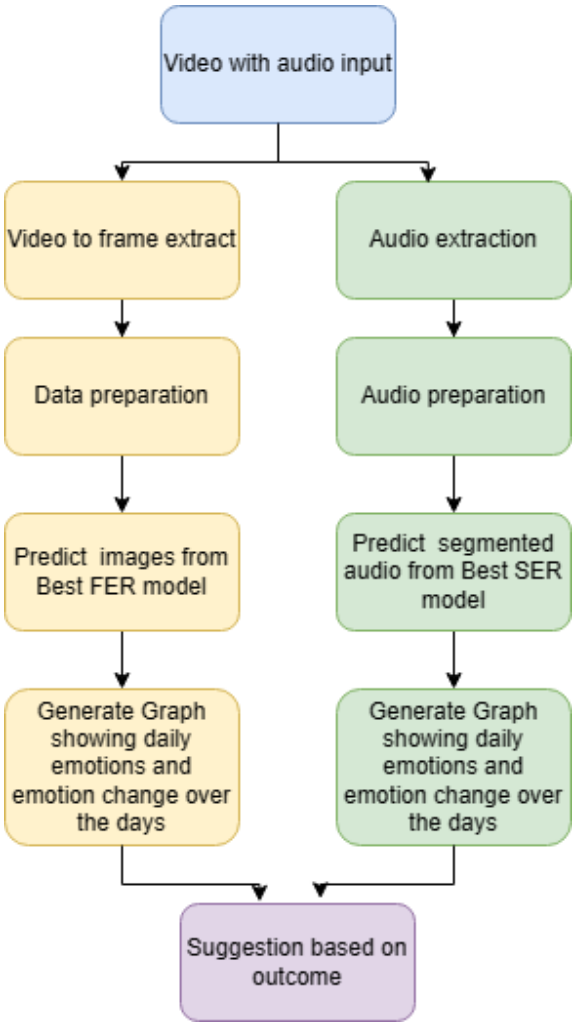


Figure 6.1: The complete structure

Chapter 7

Conclusion

As our world is becoming virtual rapidly, people from all age groups also started to experience reality in a virtual world. This is where video games come in. Just like how we react and express our emotions in different real life scenarios, we do the same while playing video games. But while playing video games, an individual's behavior pattern might change over a period of time. But as it was vague if given a year, the gamer becomes more aggressive or happy while playing video games, our paper sought to find out if long term video gaming can change a person's overall behavior or not. Using the different models that have been described in the paper, we can analyze the behavior patterns of a particular gamer, as data for various gamers were taken. Our research paper primarily focuses on online multiplayer games such as Valorant, Counter-Strike Global Offensive, Fortnite, etc. to detect the behavioral pattern of a particular individual based upon their speech and facial expressions which are later trained in deep learning models such as CNN, and its pre trained models (DenseNet121, VGG16, ResNet50) in order to deduce the outcome; whether positive or negative. Based on these findings, we were able to resolve that most of the time the gamers showed Neutral behavior, which is normal in the case of competitive games. The second highest proportion of reactions was shared by Happiness, whether the gamer was defeated in the game or not. In case of practical implementation, the paper will reinforce the suggestions in the social field where a person is told to play video games to cure depression and engage in social activity through the virtual world. However, our outcome could have been more accurate if we could have tracked heart rates and in-game chats, as it was not possible to contact the video gamers to contribute to this research paper. Additionally, wireless brain sensors could also be used to detect the brain signals and run in appropriate machine-learning models to support the results and data of our research topic. As the medical equipment needed to track the mentioned dates are expensive, we believe that our research will be still helpful for an economical approach in determining the behavior patterns while playing video games.

Bibliography

[1] [Online]. Available: https://www.uswitch.com/broadband/studies/online-gaming-statistics/.

[2] J. Clement, *Online gaming*. [Online]. Available: https://www.statista.com/topics/1551/online-gaming/#topicOverview.

[3] J. Howarth, *How many gamers are there? (new 2023 statistics)*, Aug. 2023. [Online]. Available: https://explodingtopics.com/blog/number-of-gamers.

[4] J. M. Jiménez and Y. C. Araya, "El efecto de los videojuegos en variables sociales, psicológicas y fisiológicas en niños y adolescentes," *Retos. Nuevas tendencias en Educación Física, deporte y recreación*, no. 21, pp. 43–49, 2012.

[5] M. R. Rodríguez and F. M. G. Padilla, "El uso de videojuegos en adolescentes. un problema de salud pública," *Enfermería Global*, vol. 20, no. 2, pp. 557–591, 2021.

[6] P. J. Carrillo López, M. García Perujo, *et al.*, "Consumo habitual de videojuegos y rendimiento académico en escolares de primaria," *Education in the knowledge society: EKS*, 2022.

[7] F. Gómez Gonzalvo, J. Devís Devís, J. P. Molina Alventosa, *et al.*, "El tiempo de uso de los videojuegos en el rendimiento académico de los adolescentes," *Comunicar: revista científica iberoamericana de comunicación y educación*, 2020.

[8] D. Nash, H.-R. Lee, C. Janson, C. Richardson-Olivier, and M. J. Shah, "Video game ventricular tachycardia: The "fortnite" phenomenon," *HeartRhythm Case Reports*, vol. 6, no. 6, pp. 313–317, 2020.

[9] I. Granic and A. Lobel, "E, cm, & engels, rcme (2014). the benefits of playing video games," *American Psychologist*, vol. 69, no. 1,

[10] D. Uttal, "Meadow ng. tipton e. hand ll. alden ar. warren c. newcombe ns," *The malleability of spatial skills: a meta-analysis of training studies. Psychol Bull*, vol. 139, pp. 352–402, 2013.

[11] M. Quwaider, A. Alabed, and R. Duwairi, "The impact of video games on the players behaviors: A survey," *Procedia Computer Science*, vol. 151, pp. 575–582, 2019.

[12] Q. Zhang, Y. Cao, and J. Tian, "Effects of violent video games on aggressive cognition and aggressive behavior," *Cyberpsychology, Behavior, and Social Networking*, vol. 24, no. 1, pp. 5–10, 2021.

[13] B. J. Bushman and C. A. Anderson, "Comfortably numb: Desensitizing effects of violent media on helping others," *Psychological science*, vol. 20, no. 3, pp. 273–277, 2009.

[14] S. M. Coyne and L. Stockdale, "Growing up with grand theft auto: A 10-year study of longitudinal growth of violent video game play in adolescents," *Cyberpsychology, Behavior, and Social Networking*, vol. 24, no. 1, pp. 11–16, 2021.

[15] T. Greitemeyer, "The spreading impact of playing violent video games on aggression," *Computers in human behavior*, vol. 80, pp. 216–219, 2018.

[16] C. Barlett, C. D. Rodeheffer, R. Baldassaro, M. P. Hinkin, and R. J. Harris, "The effect of advances in video game technology and content on aggressive cognitions, hostility, and heart rate," *Media psychology*, vol. 11, no. 4, pp. 540–565, 2008.

[17] *Games That People Play - Time Jan 1982 - VideoGame Pavilion*, Oct. 2015. [Online]. Available: https://vgpavilion.com/mags/1982/01/time/games-that-people-play/.

[18] G. Gwinup, T. Haw, and A. Elias, "Cardiovascular changes in video-game players: Cause for concern?" *Postgraduate medicine*, vol. 74, no. 6, pp. 245–248, 1983.

[19] W. Li, Y. Li, W. Yang, Q. Zhang, D. Wei, W. Li, G. Hitchman, and J. Qiu, "Brain structures and functional connectivity associated with individual differences in internet tendency in healthy young adults," *Neuropsychologia*, vol. 70, pp. 134–144, 2015.

[20] S. J. Kirsh, J. R. Mounts, and P. V. Olczak, "Violent media consumption and the recognition of dynamic facial expressions," *Journal of interpersonal violence*, vol. 21, no. 5, pp. 571–584, 2006.

[21] L. Billings, D. Harrison, and J. Alden, "Age differences among women in the functional asymmetry for bias in facial affect perception," *Bulletin of the Psychonomic Society*, vol. 31, no. 4, pp. 317–320, 1993.

[22] T. F. Cootes, C. J. Taylor, *et al.*, *Statistical models of appearance for computer vision*, 2004.

[23] L. Claes, S. Jiménez-Murcia, J. J. Santamaría, M. B. Moussa, I. Sánchez, L. Forcano, N. Magnenat-Thalmann, D. Konstantas, M. L. Overby, J. Nielsen, *et al.*, "The facial and subjective emotional reaction in response to a video game designed to train emotional regulation (playmancer)," *European Eating Disorders Review*, vol. 20, no. 6, pp. 484–489, 2012.

[24] L. Kerkeni, Y. Serrestou, M. Mbarki, K. Raoof, M. A. Mahjoub, and C. Cleder, "Automatic speech emotion recognition using machine learning," in *Social media and machine learning*, IntechOpen, 2019.

[25] M. Gjoreski, H. Gjoreski, and A. Kulakov, "Machine learning approach for emotion recognition in speech," *Informatica*, vol. 38, no. 4, 2014.

[26] R. A. Khalil, E. Jones, M. I. Babar, T. Jan, M. H. Zafar, and T. Alhussain, "Speech emotion recognition using deep learning techniques: A review," *IEEE Access*, vol. 7, pp. 117 327–117 345, 2019.

[27] M. Ghai, S. Lal, S. Duggal, and S. Manik, "Emotion recognition on speech signals using machine learning," in *2017 international conference on big data analytics and computational intelligence (ICBDAC)*, IEEE, 2017, pp. 34–39.

[28] M. Quwaider, A. Alabed, and R. Duwairi, "Shooter video games for personality prediction using five factor model traits and machine learning," *Simulation Modelling Practice and Theory*, vol. 122, p. 102 665, 2023.

[29] Z. Nie, "Research on facial expression recognition of robot based on cnn convolution neural network," in *2020 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, 2020, pp. 1067–1070. DOI: 10.1109/ICPICS50287.2020.9202139.

[30] P. Kaviya and T. Arumugaprakash, "Group facial emotion analysis system using convolutional neural network," in *2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184)*, 2020, pp. 643–647. DOI: 10.1109/ICOEI48184.2020.9143037.

[31] E. Hussein, U. Qidwai, and M. Al-Meer, "Emotional stability detection using convolutional neural networks," Feb. 2020, pp. 136–140. DOI: 10.1109/ICIoT48696.2020.9089440.

[32] Z. Fei, E. Yang, D. D.-U. Li, S. Butler, W. Ijomah, X. Li, and H. Zhou, "Deep convolution network based emotion analysis towards mental health care," *Neurocomputing*, vol. 388, pp. 212–227, 2020.

[33] N. Ganapathy, Y. R. Veeranki, and R. Swaminathan, "Convolutional neural network based emotion classification using electrodermal activity signals and time-frequency features," *Expert Systems with Applications*, vol. 159, p. 113 571, 2020.

[34] M. A. Ozdemir, B. Elagoz, A. Alaybeyoglu, R. Sadighzadeh, and A. Akan, "Real time emotion recognition from facial expressions using cnn architecture," in *2019 Medical Technologies Congress (TIPTEKNO)*, 2019, pp. 1–4. DOI: 10.1109/TIPTEKNO.2019.8895215.

[35] A. Slimi, M. Hamroun, M. Zrigui, and H. Nicolas, "Emotion recognition from speech using spectrograms and shallow neural networks," in *Proceedings of the 18th International Conference on Advances in Mobile Computing & Multimedia*, 2020, pp. 35–39.

[36] P. Harár, R. Burget, and M. Kishore Dutta, "Speech emotion recognition with studies," in *Proceedings of the 4th International Conference on Signal Processing and Integrated Networks (SPIN)*, Noida, India, 2017, pp. 137–140.

[37] P. Jiang, H. Fu, H. Tao, P. Lei, and L. Zhao, "Parallelized convolutional recurrent neural network with spectral features for speech emotion recognition," *IEEE Access*, vol. 7, pp. 90 368–90 377, 2019. DOI: 10.1109/ACCESS.2019.2927384.

[38] W. Lim, D. Jang, and T. Lee, "Speech emotion recognition using convolutional and recurrent neural networks," in *2016 Asia-Pacific signal and information processing association annual summit and conference (APSIPA)*, IEEE, 2016, pp. 1–4.

[39] V. Praseetha and S. Vadivel, "Deep learning models for speech emotion recognition," *Journal of Computer Science*, vol. 14, no. 11, pp. 1577–1587, 2018.

[40] K. Han, D. Yu, and I. Tashev, "Speech emotion recognition using deep neural network and extreme learning machine," in *Interspeech 2014*, 2014.

[41] D. Bertero and P. Fung, "A first look into a convolutional neural network for speech emotion detection," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, IEEE, 2017, pp. 5115–5119.

[42] Z. K. Abdul and A. K. Al-Talabani, "Mel frequency cepstral coefficient and its applications: A review," *IEEE Access*, vol. 10, pp. 122 136–122 158, 2022. DOI: 10.1109/ACCESS.2022.3223444.

[43] "Mel spectrogram-based advanced deep temporal clustering model with unsupervised data for fault diagnosis," *Expert systems with applications*, 2023.

[44] "Speech emotion recognition using a dual-channel complementary spectrogram and the cnn-ssae neutral network," *Applied Sciences*, 2022. DOI: 10.3390/app12199518.

[45] T. Giannakopoulos and A. Pikrakis, *Introduction to Audio Analysis: A MATLAB® Approach*. Elsevier Science, 2014, ISBN: 9780080993898. [Online]. Available: https://books.google.com.bd/books?id=zbHVAQAAQBAJ.

[46] A. Shah, M. Kattel, and A. Nepal, "Chroma feature extraction," Jan. 2019.

[47] D. Dwivedi, A. Ganguly, and V. Haragopal, "6 - contrast between simple and complex classification algorithms," in *Statistical Modeling in Machine Learning*, T. Goswami and G. Sinha, Eds., Academic Press, 2023, pp. 93–110, ISBN: 978-0-323-91776-6. DOI: https://doi.org/10.1016/B978-0-323-91776-6.00016-6. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780323917766000166.

[48] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[49] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[50] X. Zhang, J. Zou, K. He, and J. Sun, "Accelerating very deep convolutional networks for classification and detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 10, pp. 1943–1955, 2015.

[51] K. He, X. Zhang, S. Ren, and J. Sun, *Deep residual learning for image recognition. corr abs/1512.03385 (2015)*, 2015.

[52] G. Meena, K. K. Mohbey, A. Indian, and S. Kumar, "Sentiment analysis from images using vgg19 based transfer learning approach," in *Procedia Computer Science*, M. K. Mishra, J. K. Patro, and R. K. Behera, Eds., vol. 204, 2022, pp. 411–418. DOI: 10.1016/j.procs.2022.08.050.

[53] A. Campilho, F. Karray, and B. ter Haar Romeny, "Classification of breast cancer histology images through transfer learning using a pre-trained inception resnet v2," in *Image Analysis and Recognition*, A. Campilho, F. Karray, and B. ter Haar Romeny, Eds., vol. 10882, 2018, pp. 763–770. DOI: 10.1007/978-3-319-93000-8_86.

[54] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, *Mobilenetv2: Inverted residuals and linear bottlenecks*, 2019. arXiv: 1801.04381 [cs.CV].

[55] F. Chollet, *Xception: Deep learning with depthwise separable convolutions*, 2017. arXiv: 1610.02357 [cs.CV].

[56] M. Tan and Q. V. Le, *Efficientnet: Rethinking model scaling for convolutional neural networks*, 2020. arXiv: 1905.11946 [cs.LG].

[57] A. Sherstinsky, "Fundamentals of recurrent neural network (rnn) and long short-term memory (lstm) network," *Physica D: Nonlinear Phenomena*, vol. 404, p. 132 306, 2020, ISSN: 0167-2789. DOI: https://doi.org/10.1016/j.physd.2019.132306. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167278919305974.

[58] A. Iosifidis and A. Tefas, *Deep Learning for Robot Perception and Cognition*. Elsevier Science, 2022, ISBN: 9780323885720. [Online]. Available: https://books.google.com.bd/books?id=4EU6EAAAQBAJ.

[59] P. Ekman, *Universal facial expressions of emotion*, Paul Ekman Group, Retrieved May 9, 2023, from https://www.paulekman.com/universal-facial-expressions-of-emotion/, n.d.

[60] *Fair use on YouTube*, YouTube Help, Retrieved May 10, 2023, from https://support.google.com/youtube/answer/9783148?hl=en, n.d.

[61] F. Burkhardt, A. Paeschke, M. Rolfes, W. F. Sendlmeier, B. Weiss, *et al.*, "A database of german emotional speech.," in *Interspeech*, vol. 5, 2005, pp. 1517–1520.

[62] I. S. Engberg, A. V. Hansen, O. Andersen, and P. Dalsgaard, "Design, recording and verification of a danish emotional speech database," in *Fifth European conference on speech communication and technology*, 1997.

[63] S. R. Livingstone and F. A. Russo, "The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english," *PloS one*, vol. 13, no. 5, e0196391, 2018.

[64] K. Dupuis and M. K. Pichora-Fuller, "Recognition of emotional speech for younger and older talkers: Behavioural findings from the toronto emotional speech set," *Canadian Acoustics*, vol. 39, no. 3, pp. 182–183, 2011.

[65] H. Cao, D. G. Cooper, M. K. Keutmann, R. C. Gur, A. Nenkova, and R. Verma, "Crema-d: Crowd-sourced emotional multimodal actors dataset," *IEEE transactions on affective computing*, vol. 5, no. 4, pp. 377–390, 2014.

[66] S. Haq and P. Jackson, "Machine audition: Principles, algorithms and systems, chapter multimodal emotion recognition," *IGI Global, Hershey PA*, pp. 398–423, 2010.

[67] K. K. Kishore and P. K. Satish, "Emotion recognition in speech using mfcc and wavelet features," in *2013 3rd IEEE International Advance Computing Conference (IACC)*, IEEE, 2013, pp. 842–847.