National College of Ireland

Improving Emotion Detection and Music Recommendation Through Advanced Facial Recognition and Optimized Hyper-parameters Tuning

MSc Research Project
Data Analytics

Vipin Sharma
Student ID: 22207406

School of Computing
National College of Ireland

Supervisor: Ahmed Makki

National College of Ireland
Project Submission Sheet
School of Computing

National College of Ireland

Student Name	Vipin Sharma
Student ID:	22207406
Programme:	Data Analytics
Year:	2024
Module:	MSc Research Project
Supervisor:	Ahmed Makki
Submission Due Date:	12/08/2024
Project Title:	Improving Emotion Detection and Music Recommendation Through Advanced Facial Recognition and Optimized Hyper-parameters Tuning
Word Count:	8623
Page Count:	20

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL Internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use another author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Vipin Sharma
Date:	12th September 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Improving Emotion Detection and Music Recommendation Through Advanced Facial Recognition and Optimized Hyper-parameters Tuning

Vipin Sharma
22207406

Abstract

Facial expression recognition is an important component of human-computer interaction which enables systems to understand and respond to human emotions. This study investigates how hyper-parameters optimize the accuracy and flexibility of facial expression recognition models and their integration with a music recommendation system. The primary goal is to develop the personalized music recommendation model on seven types of different facial expressions and determine the impact of hyper-parameters such as learning rate, optimizers, batch size, number of epochs, and activation functions on model performance. In the research, Convolutional neural networks (CNN) with hyper-parameter tuning and other two pre-trained deep learning models ResNet50 and Xception with hyper-parameter tuning were applied, and analyzed the validation accuracy and validation loss. A complete design specification is shown that combines advanced facial recognition techniques with user-define function (UDF) music recommendation features. The accuracy of the model is 65% with the proposed CNN architecture, 51.75% with ResNet50, and 70% with Xception when we evaluated different models under various hyper-parameters. All the findings show that hyper-parameter tuning increases the performance of facial expression-based music recommendation systems and provides a valuable contribution to the development of more interactive and responsive user experiences.

1 Introduction

1.1 Background

Emotion detection and music recommendation have become a very important application that uses deep learning and machine learning techniques to influence the user experience in various domains such as entertainment, mental health, and human-computer interaction. These days every industry uses a wide range of applications which is based on automated facial expression recognition such as data-driven animation, neuromarketing, interactive games, sociable robotics, and numerous other human-computer interaction systems Lopes et al. (2017). Facial expression is one common way that people use to communicate. It is a type of non-verbal communication and it uses body language to deliver a wide range of different emotions. There are different types of emotions the human according to the Chaudhry et al. (2023) there are mainly seven types of different emotions i.e.,

'angry', 'sad', 'happy', 'surprise', 'neutral', 'fear', and 'disgust'. In this study, the author also uses these types of different emotions to suggest the music. In contrast, emotion detection systems have made significant advancements with the advent of deep learning and particularly advanced facial recognition technologies. These can detect and interpret many emotions with high accuracy. Talking about Music, It has a significant effect on users. It's safe to say that listening to music is one of the most popular activities. As a result, there has been continued interest from the music industry. Singh and Dembla (2023) Given how simple and affordable it is to listen to music, it is not surprising that over the past 15 years, digital and streaming music has gradually beat physical recordings in terms of revenue for the recorded music industry (IFPI, 2018). There are some famous methods such as Collaborative filtering, content-based filtering, or hybrid approach used for music recommendation systems. However, Though these techniques have achieved many successes, they usually fail to modify recommendations based on the user's current emotional state. Nowadays, there are a lot of facial data is available on different database websites which helps deep learning models to recognize facial expressions better and gives good accuracy. While working with facial expressions there are some challenges faced that limit the performance of the model in the real world each facial image has different conditions like different shapes and facial orientations. The different range of users' skin color and the background color of the image also affect the learning rate. According to Raja Sekaran et al. (2021) There are different ways to express each emotion on the face some people may express their emotions fully in a given scenario, while others may not. Despite these advancements, the challenge remains in seamlessly integrating emotion detection and music recommendation systems to create a unified framework that leverages the strengths of both domains.

1.2 Motivation

The study is mainly focused on finding out how the integration of facial feature recognition and pre-trained deep learning-based models such as ResNet50, Xception, and convolutional neural networks can be able to improve the performance of music recommendation systems based on emotions by optimizing the hyperparameters of Convolutional Neural Networks (CNNs) and the pre-trained models as well.

1.3 Research Question

How can the accuracy of deep learning-based facial expression recognition systems with optimized hyper-parameters be enhanced by combining facial recognition with song recommendation?

By exploring this question the study also contributes to the development of a music recommendation system based on facial expressions and checks what are the impact of using different hyper-parameters such as learning rate, optimizers, batch size, epochs value, and activation function on the model and these all the hyper-parameter tunning also increase the performance of the model.

1.4 Research Objectives

- Apply and Analyze the two advanced pre-trained deep learning models' ResNet50 and Xception. Also, check accuracy and error rates in facial emotion recognition.
- To develop a simple user-defined function (UDF) for a music recommendation framework that combines advanced facial recognition.
- Using the different hyper-parameters such as learning rate, optimizers, batch size, epochs value, and activation function of the proposed system to enhance its accuracy and flexibility.
- Run the model on different epoch values and different hyperparameter values to evaluate the performance of the model.

1.5 Report Structure

The below report is organized as follows section 2 reviews the related work which gives the general knowledge of existing research and identifies gaps. Section 3 details the methodology followed in the research, including data collection and analysis techniques. Section 4 outlines the design specifications, covering system workflow and complete architecture. Section 5 describes the implementation process, highlighting the different models used in the study and key components. In Section 6, the evaluation and discussion of the results are presented, followed by conclusions and suggestions for future work in Section 7. Finally, all the references are listed at the end of the report.

2 Related Work

2.1 Integration of Facial Expression Recognition with Music Recommendation

In the last few years combining facial Expression recognition with music recommendations gained more attention in the field of deep learning, artificial intelligence, and human-computer interaction. There is a study by Gobinath et al. (2024) on Emotional Harmony through Deep Learning: A Facial Expression-Based Music Therapy uses convolutional neural networks (CNNs) to classify the different emotions from the facial expression and create a Mood-Music App which suggests the songs based on the emotion of the user which helps to reduce the stress level and balance the user mood. The Mood-Music application is good and gives accurate performance to identify different emotions like happiness, sadness, neutrality, and, anger also gives an accurate music playlist which enhances the user's mood. However, while reading the paper found that the study requires a large dataset to identify the emotion perform the classification more accurately, and give personalized song recommendations accurately. Similarly, another music player application called Emotion-Based Music Player developed by Chankuptarat et al. (2019) uses the heart rate analysis and facial recognition of the users to detect emotions and recommend music accordingly. This study also helps to reduce the stress and enhance the mood of users. The model gives good accuracy on the different emotions such as Neutral, Angry, Sad, and Happy but the model gives the best accuracy on the Happy

emotion, giving 98% accuracy the problem is with the Sad emotion it gives 40% accuracy so it requires more accurate algorithm to improve the performance and detect the human emotion. Another important contribution by Asha et al. (2024) which uses the pre-trained deep learning model VGG16 and Convolutional neural network (CNN) to give the suggested playlist based on the user's emotion which is detected from the facial expression. The Pre-trained model VGG16 CNN gives an accuracy of 92% and the CNN model gives less accuracy as compared to the VGG16 CNN model. This work contributes to and enhances the emotion detection, facial expression, and music recommendation system. Some suggestions are required and shown in the paper it requires more datasets to further improve and also develop different methods, pre-trained deep learning models, and continuous learning algorithms to adapt to dynamic user emotions. Another research by Álvarez, Zarazaga and Baldassarri (2020) based on the location and emotion to make a Mobile music recommendation system called The DJ-Running application which uses an emotional wearable device and geospatial data to provide dynamic, personalized music recommendations for runners who perform exercise regularly. Everybody knows that while doing an exercise everyone listens to the music by applying machine learning on the Spotify dataset which fetches the songs from the web API and the take Emotion dataset applying the Regression models and performing statistical analysis on top of that this type of application not give the high level of running experience but also use the different type of multiple data sources to create a high level interactive and responsive application. However, there is one thing that is very difficult to deal with in real-time data processing and requires the real-time dataset to perform the testing of the model currently the model is trained at the Sports Medicine Center of the government of Spain and the local athletics clubs. Hanafi et al. (2023) develops web applications that recommend song systems based on human facial expressions and emotions. Using the two famous technologies to detect human emotion OpenCV and Deep Face to detect the face of the user and when the face is detected author connects that emotion to the Spotify API to retrieve a playlist of songs based on the emotion. This study is mainly talking about positive and negative emotions and talking about the accuracy of the model so the paper mentions that the accuracy of the Deepface model which is used to detect the emotion gives 97% accuracy. However paper was not focused on the different types of emotions so further studies use the different types of emotions and put more focus on them to improve the emotion of the web application. Another research by Kumar et al. (2023) created a system to detect the user stress level and recommend not only songs but movies also. By using the Convolutional neural networks (CNNs) to recognize facial expressions to detect stress levels and categorize the different types of emotions such as happy, angry, sad, neutral, and fear. The main part of this system uses HAARCASCADE to detect the Face and OpenCV library to capture the picture from the video recording and for classification of the emotion used the ferjj.h5 pre-trained deep learning model and for the music classification using the KNN algorithm. The application works well and suggests Movie and music but there is a lack of a dataset in the future same method will be used to recommend mobile applications and gifts for the user. Similarly, another intelligent music recommendation system by Gupta et al. (2023) based on facial emotion takes the user feedback and improves the model performance. Applying the CNN model to analyze the different facial emotions and recommends music that is related to the user's emotional state and improves the user experience reducing the stress level. There are five types of different facial expressions mentioned such as sad, neutral, surprised, angry, and happy. For the music recommendation using the PCA and LDA algorithms which help to get

better accuracy recommends the accurate songs based on the different facial expressions. The accuracy of the model on the CNN network is 91.68% accuracy and on the support vector machine (SVM) is 69.45%. There are two authors Singh and Dembla (2023) and Chaudhry et al. (2023) detect emotion by using deep learning and transfer learning. Both papers apply CNN, VGG16, and EfficientNetB0 on the image dataset to classify several emotions. The accuracy of the models comes between 60% to 75%. In the second paper author uses a pre-trained deep-learning model which helps to give good accuracy. Also, focus on transfer learning in refining emotion-based recommendation systems these types of learning increase user satisfaction by giving emotionally relevant music. Some of the future work mentioned in the papers is to integrate this system into the automatic music player and focus on the different multimodal deep learning which helps to integrate the different data from many different sources including the text, audio, and video which help to increase the accuracy and precision of the model. Also, explore real-time applications and implement large datasets for better accuracy.

2.2 Facial Expression Recognition Techniques

In the last few years, facial expression recognition (FER) got more attention because of its different applications which are used in various fields such as Health, security, human-computer interaction, and many more. There are many different kinds of facial recognition developed by using different technologies to increase the recognition performance but there is a drastic change in the model performance when deep learning and Convolutional neural networks (CNNs) are used to build the FER model. Deep learning and CNNs are used in every field because of the availability of data nowadays to train the models also there are advanced GPU systems are available in the market.

Kim et al. (2019) describe that facial expression recognition is most important to understand human emotion. In the era of Artificial intelligence, it is very important to create a strong model that can extract the expression of the face. In the paper, the author uses hierarchical deep learning and extracts the feature from the image by using the feature-based network fused with the geometric feature in the hierarchical structure. Also, using the autoencoder technique to generate facial images with neutral emotions of the human and then techniques help the author to recognize the dynamic facial expressions between different expressions. There are two different datasets used for this study CK+ and JAFE datasets and on both the dataset author gets good accuracy around 92-93% but the main problem of this paper is that the author takes very less images dataset for the experiment and there are not more neutral face images. The techniques used in the paper are good like action units(AU) and autoencoder techniques. Using the Geometric feature-based network for feature detection reduces the surroundings and captures only the important features of the images. Another paper Lopes et al. (2017) describes a method to integrate the Convolutional neural network (CNN) with particular image pre-processing steps to increase the accuracy of the FER using different datasets like CK+, JAFE, and BU-3DFE. In the paper, it is make sure that the training and testing data do not mix otherwise it impacts the accuracy of the model and for detecting facial expression recognition in this paper author combines the Convolutional Neural Network with the pre-define pre-processing steps. By applying the pre-processing techniques author only extracts the important features of the face and uses them for training purposes by implementing this the accuracy of the model is 96.76% in the CK+ dataset and the training of the model is also fast. The limitation of the paper is that the variation

between the classes is less with the less number of images data and another limitation is that the images with the frontal faces have a controlled environment. To handle these limitations we can work on the large dataset and use the deeper CNNs network with the large number of layers to handle these types of constraints. Ravi et al. (2020) conducted a comparison study on the two different effectiveness techniques Local binary patterns (LBP) and Convolutional Neural networks (CNNs) checking which one is better and giving a good performance. For this author used the three different datasets CK+, JAFE, and Yale Face datasets For training purposes 70% of the data and 30% of the data were used for testing. In the paper, the static image was only used for the experiment which is one disadvantage also did not mention more about implementation in detail and critical analysis of all three datasets. The accuracy of the CNN model is better than the LBP model, The CNN model uses the softmax method to classify the images into different classes and the LBP uses the Support vector machine (SVM) classifier. The accuracy of LBP on the CK+ dataset is 90% whereas the CNN gives 97.32% accuracy on the same data but the CNN gives the lowest accuracy on the YALEFACE dataset which is 31.82% which is very low. Additionally, Raja Sekaran et al. (2021) proposed a transfer learning approach using the pre-trained AlexNet model. Also, talking about the existing approach that is used on FER like handcrafted feature-based methods (HCF) and deep learning models. Basically, according to the paper, the HCF depends on how the manual features are extracted and the deep learning uses the CNN networks to extract the features and do classification. This paper used the pre-trained AlexNet model and fine-tuned the pre-trained model on emotion-specific datasets to achieve the best results. This pre-trained model AlexNet reduces the training time, works fast, captures all the important features in the image, and also reduces the risk of overfitting, achieving good accuracy on both CK+ and FER datasets.

2.3 Music Recommendation System

In Recent years the music recommendation system has been very famous. It is very interesting to develop a recommendation song engine based on human emotional expression. Advanced technology is used to create a music recommendation like machine learning, deep learning, and many user-defined functions (UDF). By using real-time video capture can detect the different emotions of humans and recommend music by implementing the Convolutional neural network (CNN) for emotion detection and k-means clustering for music recommendationSingh et al. (2023). This not only personalizes the current emotional state into the result but also keeps creating and evolving with changing user choices contingently. Additionally, reinforcement learning (RL) is also used to create a playlist recommendation to improve the user listening experience by involving feedback from the user Hu et al. (2017). By using the Markov Decision Process and reinforcement learning framework RLWRec to design a system that generates the music playlist. RL is used to take the feedback from the user and enable them to continuously learn from the feedback or user interaction behavior. This resolves the one limitation of the classical recommendation system which depends on static user profiles of previous behaviors of the users like the YouTube and search engine works. Another author Kumar and Rakesh (2022) developed a Music recommendation system by using a machine learning algorithm. To increase the user experience and satisfaction by recommending the music according to the user demands author uses the different types of machine learning techniques such as collaborative filtering, Hybrid Recommender system, and content-based filtering. This

paper deeply discusses how the methods and techniques are used providing a comparison between which techniques are working best but there is only one method discussed in the paper which is content-based filtering. To create a model using the Spotify dataset which includes information on the songs like artist, Genre, and Year. In the future, this study will benefit when we work on evaluating user satisfaction and real-world performance. Streamlit Webapp is an open source platform for creating web applications author Joseph et al. (2023) uses this platform to create a web portal that takes some inputs like language and singer name and captures the face photo of the user to detect emotions and then recommends the song url based on the user's emotion which is directly connected to YouTube. In the paper, the author uses the convolutional neural network to detect the user's emotions and the OpenCV library used for image processing to capture the object, faces, and design in images and videos. Some advanced changes could made in the system affecting the model accuracy and pre-processing also fast. So the suggestion for future work is to use the pre-trained deep learning model to detect the facial expression.

2.4 Conclusion of Literature Review

These all are the previous studies show the importance of combining music with different emotional states to enhance mental health, physical health, and also show the current challenges such as the requirement of a larger dataset, and improved performance of the model such as accuracy, and precision by using the artificial intelligence, deep learning, machine learning. Also, found that there is no paper that mentions about the hyper-parameter tunning how it works and what are impact of the hyper-parameter is on the model design and system.

3 Methodology

Figure 1 shows the complete workflow of model design.

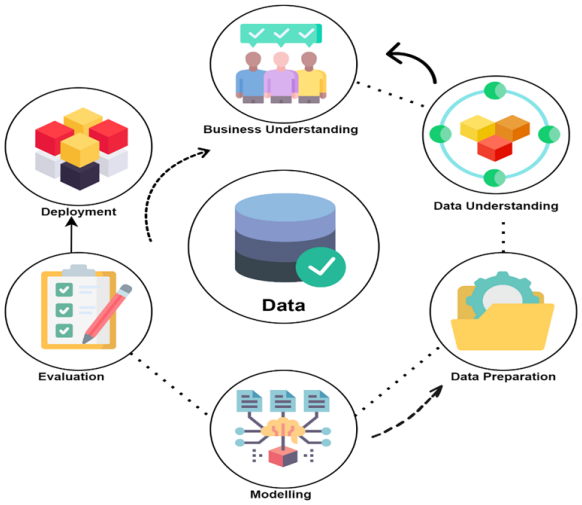

Figure 1: Methodology (Original Illustration)

3.1 Business Understanding

The main understanding of the research is to find how much impact the hyper-parameter tuning can have on the model's performance and accuracy and the second is to enhance the user experience by using the pre-deep learning models and Convolutional Neural Network to detect the facial expression and recommend the song name, artist name, and popularity of the song. It also helps to provide the behavior of the user, enabling targeted marketing and also helps to generate revenue through premium services. Moreover, this research is also used in the health sector, and hospitals to increase the user's mental health and physical health.

3.2 Data Understanding

This research work uses two different types of datasets one is the facial expression dataset named FER-2013 and another one is the Spotify dataset. Both datasets get from the open-source library [1]Kaggle which is a platform for a variety of datasets used by many data scientists and researchers. The FER-2013 dataset has 32198 images consisting of 48*48 pixels greyscale size face images and there are seven different categories Angry, Disgust, Neutral, Fear, Happy, Surprise, and Sad. The second dataset Spotify dataset consists the 687 rows of data and 19 different columns which include name, album, artist, popularity, and many more things.

3.3 Data Preparation

This is the third phase of methodology in this the data is pre-processed and converted in the proper format before applying the hyper-parameter tuning with CNN and a pre-trained deep learning model. With the first data FER-2013 which is used for face expression detection, the author fixes the batch size 64 and the image shape 48*48. To pre-process the images using many things which include rescaling the image, flipping the image, batch sizing, batch normalization, MaxPooling, dropouts, and paddings. Also, uses the cv2 library which includes all the necessary libraries for image resizing and performing the data augmentation for better image capturing which includes the rotation ranges, zoom range, width range, width shift range, horizontal flip, and fill mode. For the second dataset, no need to do pre-processing author just checked that there are no missing values and null values in the dataset.

3.4 Modelling

The main goal of this research is to work with the hyper-parameter tuning and how much impact it can have on the model so the author first deploys a Convolutional neural network with the hyper-parameter tuning parameters, ResNet50 (Residual Network), and Xception deep learning models to recognize the facial expression. For the music dataset create a function called recommend song which merges the different emotions with the different songs by using the if else condition. The important things while deploying the CNN with hyper-parameter and pre-trained deep learning models after that the author use the Keras callback such as ModelCheckpoint, EarlyStopping, and ReduceLROnPlateau which helps prevent the overfitting of the model, saving the model when optimal

[1]`https://www.kaggle.com/`

results are obtained, and the training process become more controlled and increases the model performance.

3.5 Evaluation

For the evaluation of the model performance, there are two key factors i.e., accuracy and loss percentages matrices. These two matrices show how the model was performed during evaluation and the OpenCV Haar Cascade is used when the author processes the new data for testing and validation.

3.6 Deployment

The Deployment model is the last phase of the methodology, this phase involves deploying the trained model into the real world where it can use the new different images and make the prediction. Here also the Haar Cascade is used when the model gets the new images with the background this will help to remove the background and focus on the facial expression only to make the prediction. Once we get the prediction of the facial expression it will combine with the songs dataset and predict the new songs according to those emotions.

4 Design Specification

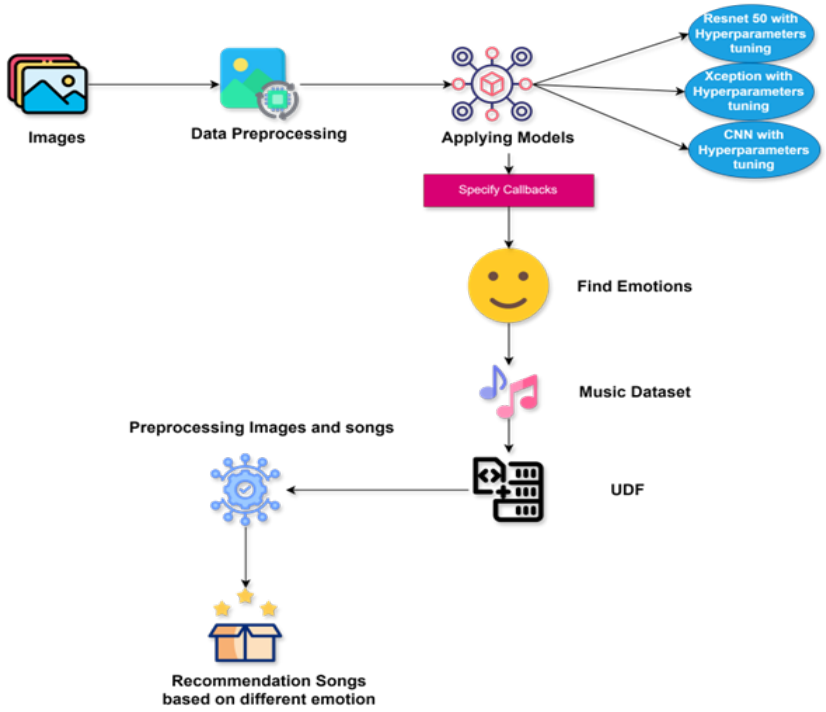

Figure 2: Design Specification (Original Illustration)

The facial expression detection and music recommendation system author designed to show the workflow of the system and how the different interconnected parts contribute

to the overall functionality and performance in Figure 2. To start with the first step, the images are collected and go for the preprocessing steps where the author can do the resizing, normalizing, and augmentation which ensures that the model is trained on a diverse and robust dataset. At this stage, also uses OpenCV's Haar Cascade classifier which is used to detect faces in the images so that they can easily extract features from the backgrounded images for further analysis.

Once these faces have been detected and cropped then they are passed through it go for the applying model's steps where the author applies pre-trained deep learning models which include ResNet-50, Xception, and the custom Convolutional Neural Network (CNN) whose hyperparameters have been tuned. These choices have been made due to their high accuracies and efficiencies when it comes to image classifications because while learning about deep learning author knew that the Resnet50 and Xception are the best models for the image classification and CNN with hyper-parameters used to check how much impact the hyper-parameter have in the model. Additionally, during training, the author also specifies some callbacks such as EarlyStopping, ModelCheckpoint, or ReduceLROnPlateau because they optimize the learning process, prevent overfitting and save only the best model possible.

The next step is to map the emotions onto music preferences based on the emotional state predicted by trained models from face images. For this, the author creates a user-defined function (UDF) that associates various moods with certain genres or styles of music ensuring that the recommendations made to the individual are not only personalized but also meaningful.
After this, the system preprocesses the image with songs and performs the evaluation component used for evaluating its performance using metrics such as accuracy, Loss, and user satisfaction to ensure that recommendations are accurate. In the final step get the songs based on the different human emotions.

5 Implementation

The FER-2013 which is used for facial expression consists of 32198 images consisting of 48*48 pixels greyscale size face images and there are seven different categories Angry, Disgust, Neutral, Fear, Happy, Surprise, and Sad. Songs dataset consists the 687 rows of data and 19 different columns which include name, album, artist, popularity, and many more things. The Asus VivoBook 15 system is used which has Operating System Microsoft Windows 11 64-bit, RAM: 8.00 GB, 12th Gen Intel(R) Core(TM) i5-1235U CPU @1300Mhz, Intel(R) Iris Graphics GPU, and Version: 10.0.22631 Build 22631. The Python programming language and their packages are used for the coding work.

5.1 Data Pre-processing and Augmentation

When the data is loaded successfully the initial part is to perform the pre-processing, The author implemented many techniques for pre-processing so that the model enhances the performance. When dealing with the image dataset all the images are preprocessed using the Keras library and the ImageDatagenerator class is used for the image pre-processing.

The dataset is divided into train and test data. For training data, the author defined the train preprocessor with many different techniques such as rescaling the pixel values to range 0, 1, rotating images up to 10 degrees, applying zoom up to 20%, shifting images horizontally and vertically by 10%, and flipping images horizontally. This augmentation

step helps to understand the same image better by performing various transformed versions of the same images. For the test data, a very simple pre-processing was applied, which involves rescaling the image pixels by 1/255 so that all the test data maintain their original pixels data and there is no effect while evaluating the model performance.

After this, the augmented and pre-processed data are loaded from trained_data variables from the specified directories using the flow_from_directory method. The training data was loaded from the train data with some tuned parameters such as the class mode is categorical, the target size is 48*48 pixels which is defined in the starting, with RGB color mode, shuffling is also enabled and the batch size is 64. Similarly, the test data was loaded with the train data path for this the shuffling is not enabled because while training the train model, the author ensured that the test data is preserved for the testing, the class mode is categorical, and the batch size is the same. These are all the data preprocessing and augmentation steps to achieve or to improve the model accuracy and robustness which is the main goal of this research.

5.2 CNN Architecture with hyper-parameter:

Training Phase

While creating the CNN architecture model with the hyper-parameter tuning for the image classification task by using the Tensorflow and Keras libraries. To build the model architecture of the Convolutional Neural network (CNN) the author created a function named Convoltional_Neural_network() which includes the sequential of one-by-one convolutional, batch normalization, max pooling, dropout, and fully connected layers. The model starts with defining the function and ends by printing the summary of the model.

Batch Normalization

In the CNN architecture model, every convolutional and fully connected layer is followed by batch normalization layers. Batch normalization helps to stabilize and speed up the training process by normalizing the inputs into each layer of the CNN. Also helps to internal covariate shift allowing the network to use higher learning rates so that the training process is more efficient and gives a good result.

Padding

The same padding approach is used in the convolutional neural network layers so that all the output feature maps have the same dimensions as the input of the network. This way the method maintains the same resolution of input images throughout all parts of this CNN network which is essential for tasks requiring high localization precision and boundary detection of the image.

Flattening

A flattening layer comes after a series of pooling and convolutional layers in our network. This is very important and the basic work is to convert the 2D images into the 1D. Lastly, the author makes sure that flattening allows the model to take care of the last classification task using those high-level features extracted via its convolutional layers. The pooling section consists of a flattening layer. In other words, this flattens the 2-D feature maps into vectors appropriate for input into a fully connected softmax unit. Therefore, at test time author can resize any image using any arbitrary scale factor greater than zero since scaling does not permit cropping or resizing in general.

Dense Layer

After the output from the convolutional layers has been flattened successfully, the

network changes to a series of dense layers that are fully connected internally. These dense layers have decreasing units from 1024 to 32 and are combined with batch normalization and dropout layers. to improve training stability and reduce overfitting. **Activation Function**

After all but this the one last layer in both convolutional and fully connected neural networks has used the rectified linear unit (ReLU) activation function and the non-linearity is introduced into this model by the ReLU activation function to learn better depth patterns. The Softmax activation function is used in the final layer of the CNN model because this activation function is used to classify the multi-class classification. This activation function produces a probability distribution over seven different target classes that can be useful in multi-class classification problems.

Compilation

The loss function defined for this model is categorical cross-entropy while the optimizer used Adam. Accuracy is also used during evaluation since it's an evaluation metric. So, the author compiles the model with categorical_cross_entropy as a loss function, adam as an optimizer, and accuracy as another metric. This step of training specifies how weights will be updated and performance will be measured.

Model Training and Hyperparameter Tuning

To train the Convolutional Neural Network (CNN) model, the author uses a comprehensive approach involving hyperparameter tuning[2] and extensive training on the augmented dataset. As defined earlier the model was compiled by using the categorical cross-entropy loss function and the Adam optimizer, with accuracy as the evaluation metric.

The Training is completed by using the fit method which is used to train the model on the trained data and the author uses the tested data for validation purposes there are some major parameters that help to increase the model training author is defined such as used 10 epochs in total as well as callbacks to be used for monitoring and improving of possible training via learning rate adjustment or early stopping. Additionally, the author also defined steps per epoch and validation steps to ensure per epoch and validation phase that the right number of batches were processed to make the learning process more robust.

5.3 Resnet50 Architecture:

Resnet is defined as the Residual Network which was proposed in 2015 by researchers at Microsoft [3]. The purpose of creating a Residual Network is to reduce the vanishing and exploding problem. The main technique used by this network is skip connection in this technique all the activation layers are connected to the further layers by skipping some layers in between. This complete system becomes like a residual block.

Using this Architecture author creates a model that can detect the different emotions of the users. Firstly, the author defines the key parameters for training of Resnet50 model on the dataset, for this define a variable called image shape which is set to 224 which means the dimension of the image is 224*224 pixels. The 224*224 is the standard size for the ResNet Architecture. The batch size is also defined which shows how many images are processed in each training epoch and the batch size is 64 authors take for the model.

[2] https://www.geeksforgeeks.org/hyperparameter-tuning-fixing-overfitting-in-neural-networks/

[3] https://www.geeksforgeeks.org/residual-networks-resnet-deep-learning/

In the data preprocessing steps, define the variable train preprocessor which uses the ImageDataGenerator class to apply the many data augmentation techniques that help to train the image data which includes rescaling the pixel values, rotating the images by up to 10 degrees, applying random zooms, shifting the images by horizontally and vertically, flip the image horizontally, and fill mode is set to the nearest which help to manage the pixels during the transformation of the image. After this rescale the pixel values of the image 1/255 by using the same pre-processing ImageDataGenerator class. When all the basic pre-processing is complete the load the training data is to the particular directory by using the train preprocessor and the image is resized to 224*224 pixels which is processed in RGB color mode, shuffle is true by doing this make sure that the order of evaluation is maintained.

There are three parameters passed during fine-tuning the model the first is the image size which is set to 224*224 and By setting include top= False, the ResNet50 model to not take the fully connected to layers top layers and allows to create of custom layers which are best for the specific task and the third parameter is weights =imagenet which loads the pre-trained weights from the image net dataset this helps to model use the pre-existing knowledge of the imagenet to improve the model accuracy, performance, and reduce the training time. The author also applies the concept of freezing and unfreezing layers by using this the model learns the more generic features of the dataset and stores the learned features for better understanding.

The author also created a def function based on the neural network architecture using the Keras Sequential API. The model first includes the pre-initialized ResNet50 convolutional base, which is used for feature extracting, for preventing overfitting in the model the dropout is set to be 0.25. To normalize the output of the previous layer use BatchNormalization, Flatten is used to flatten the output layer into a one-dimensional vector. A 64 units of Dense layer is added with the activation function set to relu to learn high-level features from the flattened input. Again passed another BatchNormalization layer which is used for the further regularization with a dropout value of 0.50 which helps to prevent the overfitting of the model lastly, the model defines the Dense layer includes 7 units that show the different classes of emotion with the softmax activation function which give the output of each class and softmax is used for the multi-class classification problems. This complete function returns the model which is ready to be compiled and trained on the facial expression dataset. After this create a ResNet50 model architecture and to generate the model summary using the .summary() which gives the complete summary of the model such as how many layers, the number of parameters, the flow of all the layers, and so on.

5.4 Xception Architecture:

Xception stands for Extreme Inception. A linear stack of depth-wise separable convolution layers with residual connection is the short and simple definition of the Xception Architecture Chollet (2017). It is very easy to use with the help of the Tensorflow keras library, It consists of 36 convolutional layers creating the feature extraction base of the network.

Using this Architecture author creates a model that can detect the different emotions of the users. Firstly, the author defines the key parameters for training of Xception model on the dataset, for this define a variable called image shape which is set to 224 which means the dimension of the image is 224*224 pixels. The batch size is the same as the

author defines in the ResNet50 model which is 64 for the Xception model.

Similarly, the data preprocessing steps, define the variable train preprocessor which uses the ImageDataGenerator class to apply the many data augmentation techniques that help to train the image data which includes rescaling the pixel values, rotating the images by up to 10 degrees, applying random zooms, shifting the images by horizontally and vertically, flip the image horizontally, and fill mode is set to the nearest which help to manage the pixels during the transformation of the image. After this rescale the pixel values of the image 1/255 by using the same pre-processing ImageDataGenerator class. When all the basic pre-processing is complete the load the training data is to the particular directory by using the train preprocessor and the image is resized to 224*224 pixels which is processed in RGB color mode, shuffle is true by doing this make sure that the order of evaluation is maintained.

For the Xception same approach was applied, image size set to 224x224, include top=False, which to allow for custom layers and the weights= imagenet which uses the pre-trained ImageNet weights. The Freezing and unfreezing were also performed to increase the feature learning.

For the Xception model, the author also creates a def function based on the neural network architecture using the Keras Sequential API. The model first includes the pre-initialized Xception convolutional base, which is used for feature extracting, for preventing overfitting in the model the dropout is set to 0.25. To normalize the output of the previous layer use BatchNormalization, Flatten is used to flatten the output layer into a one-dimensional vector. A 64 units of Dense layer is added with the activation function set to relu to learn high-level features from the flattened input. Again passed another BatchNormalization layer which is used for the further regularization with a dropout value of 0.50 which helps to prevent the overfitting of the model lastly, the model defines the Dense layer includes 7 units that show the different classes of emotion with the softmax activation function which give the output of each class and softmax is used for the multi-class classification problems. This complete function returns the model which is ready to be compiled and trained on the facial expression dataset. After this create an Xception model architecture and generate the model summary by calling the .summary() which gives the complete summary of the model.

5.5 Songs Dataset and Implementation

The Spotify Song dataset the author got from the Kaggle. The author uses this dataset for the Music player recommendation dataset have 687 rows and 19 different columns which include many things regarding the music such as music name, album name, artist name, popularity, and many more things. Initially, the dataset is in the form of CSV so the first step is to read the CSV into the visual Code studio To do this the author passed the location of the dataset and stored it in the data frame named Music Player(MP) with the help of pandas library. To check whether the dataset is loaded properly or not author uses the MP.head() to focus on the top four rows of data from the entire Dataframe and call only 3 columns: name, artist, and mood. By using the .head() command the author gets confirmation that the dataset loaded successfully for further analysis.

There are different types of emotions in the FER-2013 dataset to deal with this it is important to check how many unique values are in the Music Player(MP) DataFrame mood column. Using the MP.value_counts() it give a series of unique mood values, also the values counts of the different moods. For more important things it also helps to

understand the different types of emotion which is provided in the data frame and their distribution.

There are a basic understanding of the songs dataset is completed now creating the new function which helps to predict the emotional class and recommend the songs based on that emotional class. The New function is created called Recommend Songs (RG) and a parameter is passed into its name as a pred class which is used to represent the predicted Facial expression of the user. This pred class value is very important because the pred class removes all the different emotional states and just predicts the top songs according to users' emotional states with the most popular songs. This function works with Python conditional statements and checks the values of the pred class. The author created a total of four conditional statements for different types of emotions if the pred class is 'Disgust' then the function only gives the songs with the Sad emotion and removes all other songs with different moods and sorts all the top 5 songs with their popularity in the descending order. If the pred class is either 'Happy' or 'Sad' then the Happy mood songs are showing. The author also used the reset_index(drop=True) to always reindex the top five songs of the different emotional states All the songs start from the index zero and the display(Play) function is important it showing the resulting DataFrame in the last of every conditional statement.

This function RS uses the mood attributes which is given in the songs dataset and gives the recommendation based on the predicted emotional state of the users. This function is created very dynamically so the user receives the correct recommendation that aligns with their emotional state and current mood such as the user being sad, happy, calm, or energetic.

5.6 Specifying Callbacks:

To maximize the models' performance, the callbacks are of great importance as they also help to prevent model overfitting and optimize its performance. In this study, three models were employed; a Convolutional Neural Network (CNN) with hyperparameter tuning, ResNet-50, and Xception each applied specific callbacks to optimize their performances and avoid overfitting. The author uses a model checkpoint for the CNN which is used to save it when the best validation accuracy is reached. For early stopping specify patience for checking validation loss 10 epochs then Use ReduceLROnPlateau to decrease the learning rates so as minimize overfitting of the model by reducing its rate of learning that eventually suppresses its generalization capabilities plateaus where the factor is 0.2 which helps to reduce the learning rate while validation loss plateaued defined as 5 epochs having a minimum threshold of 0.000005. For ResNet-50 and Xception model, the author similarly used the same argument "modelcheckpoint" for saving the best model however I set 7 epochs for EarlyStopping on monitoring validation accuracy and put ReduceLROnPlateau is 0.5 which aims at reducing Lr and we define minimum Lr=0.00005. By using these different callback parameters model gives the optimal performance and there is no risk for overfitting.

6 Evaluation

This section contains all the important conclusions and analyses part of the implementation phase. There are two sections in the entire study. The first part works on the facial dataset where the author applies the CNN with hyper-parameter tuning and two

other pre-trained deep learning models with hyper-parameters ResNet50 and Xception. The second is to work with the songs dataset to create a built-in function to classify the different songs.

6.1 Facial Expression Recognition Models

The Performance of different models CNN with hyper-parameters, ResNet50, and Xception has been discussed in this section. Every model is assessed with the accuracy score and loss score. The main purpose of the evaluation is to find out how well the models are working with the facial dataset and when doing the hyper-parameter which models give the best accuracy and detect the features of the different images accurately.

6.1.1 Experiment 1: Convolutional Neural Network (CNN)

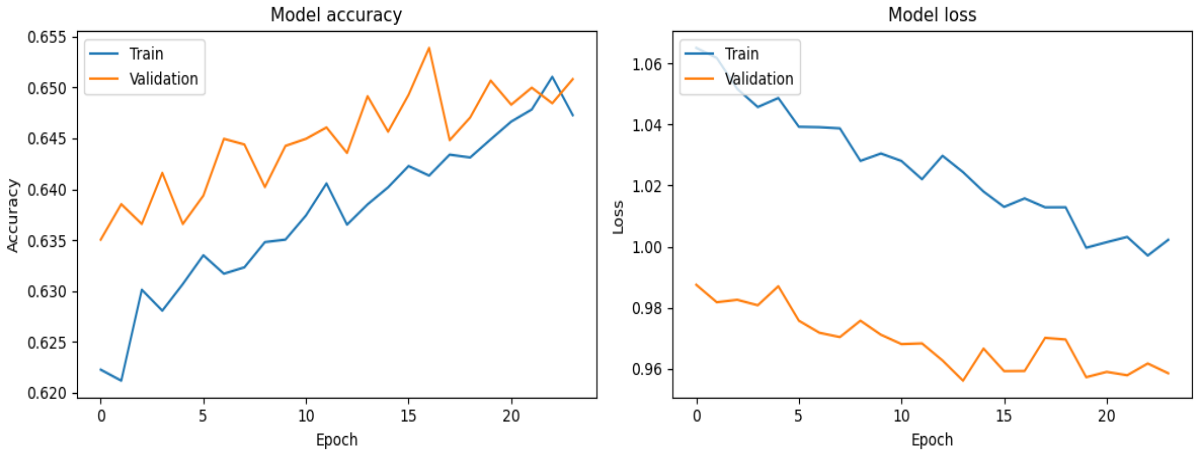

Figure 3: CNN Architecture Validation accuracy and Training loss

The CNN model is evaluated on the one hundred epochs but while running the hundred epochs the early stopping occurs at epoch sixty epoch and achieves the validation accuracy 64.97% and loss of 0.95.

In starting the author ran the model on fourty epochs, the model validation accuracy came to 63.46% and a loss of 0.98, indicating that the CNN model gives better accuracy while increasing the epoch.

At epoch sixty-four, the ReduceLROnPlateau callback reduced the learning rate to 8.000 to counteract a performance plateau. The model performance was successfully optimized by doing the hyper-parameter tuning, also by applying the callback functions. In Figure 3, The validation accuracy is increases and the loss decreases when the number of epochs is increased but at epochs 62-64 the validation accuracy is shown as constant, and early stopping occurs.

6.1.2 Experiment 2: Resnet 50 Architechture

The Pre-trained Deep learning ResNet50 Model was initially trained on five epochs where the model achieved a validation accuracy of 27.22% with a loss of 2.05%. However, significant improvement was observed when the author increased the epoch value by fifty,

then the validation accuracy increased by 51.75% and the loss decreased by 1.25%. This improvement in the accuracy of the ResNet50 model shows how well the model learns the image dataset and the hyperparameter tuning also helps the model to raise the accuracy and decrease the loss over time. The training loss and validation accuracy of the model is illustrated below in Figure 4.

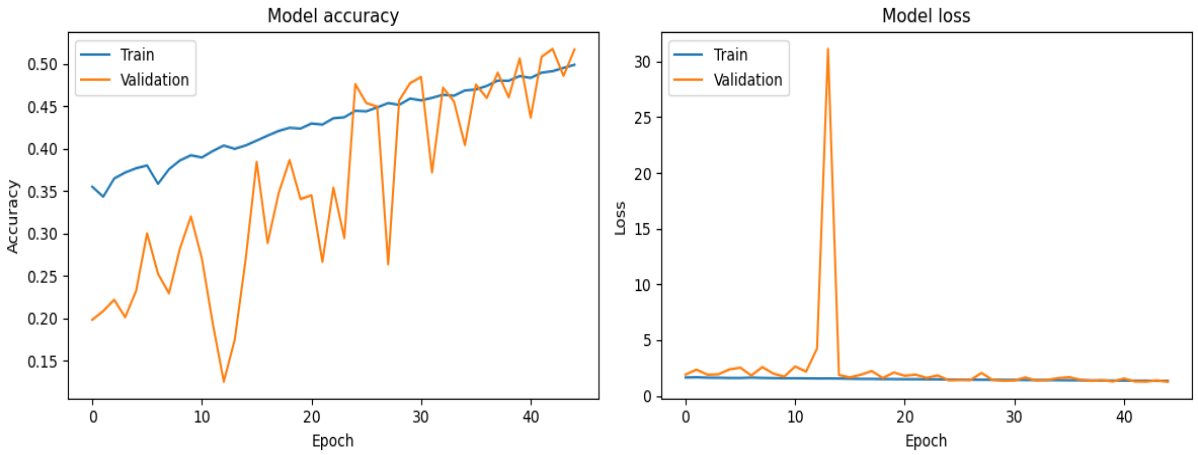

Figure 4: ResNet50 Architecture Validation accuracy and Training loss

6.1.3 Experiment 3: Xception Architecture

Same as the ResNet50 model, running the model on five epochs the validation accuracy is approximately 63% and the loss is 0.98 of the model. while when the model runs on the 25 epoch values the early stopping occurs with a validation accuracy is 70% and loss of 1.15%. It shows that while increasing the epoch values and performing the hyper-parameter tuning on the pre-trained model the accuracy increases. Figure 5 shows the training loss and validation accuracy of the model.

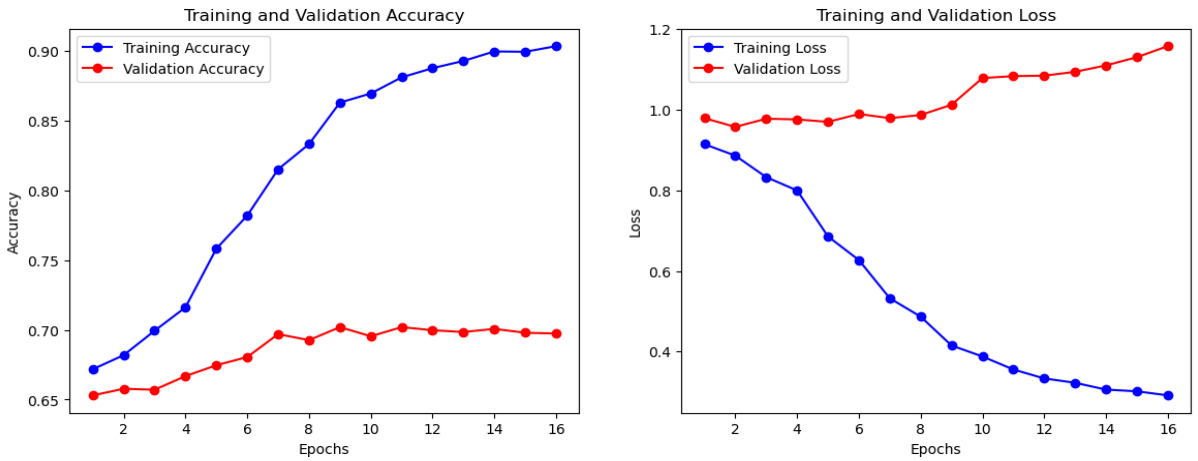

Figure 5: Xception Architecture Validation accuracy and Training loss

6.1.4 Song Dataset Pre-define Function

For the Song data, the author creates a pre-build function that is used to categorize songs from the Music Player dataset into different moods and rank them by their popularity to ensure the most favored tracks are recommended. For Disgust mood, it recommends the top five Sad songs, for Happy or Sad, it suggests the most popular Happy songs. For Fear or Angry Predictions the Calm song recommendations, while Surprise or Neutral gives Energetic song suggestions.

6.2 Discussion

While applying the CNN with hyper-parameter and two-pre-trained models on the Facial dataset. The Xception model gives a better result as compared then CNN and Resnet50. The author hyper-tuned all the models CNN, Resnet50, and Xception but majorly worked with CNN architecture which gives the validation accuracy of 64.97. In the CNN architecture applying the multiple convolutional and fully connected layers by using a Sequential architecture. The CNN model has three main convolutional blocks, each with layers of Conv2D, BatchNormalization, MaxPooling2D, and Dropout to enhance the feature extraction and prevent overfitting in the model during training time. A 32-filter Conv2D layer and a 64-filter Conv2D layer, both using the ReLU activation function, are the first convolutional blocks. Applying 64 and 128 filters in the second block and 128 and 256 filters in the third, the number of filters increases with each next block. All the blocks use the same padding and the dropout value is 0.25 for all the convolutional blocks. After this, all the convolutional layers are flattened and the output goes through a sequence of fully connected layers whose size is also decreasing slowly from 1024 to 32 units. To secure the model for better performance author applies the BatchNormalization and Dropout after each of these layers and the final output of the CNN model gives the seven classes using a softmax activation function. For the two pre-trained deep learning models Resnet50 and Xception models author directly used the image dataset. The Xception model performs better than the ResNet50 model and gives good result and accuracy. The author performed the fine-tuning on both models by leveraging their weights trained on the ImageNet dataset and excluding their top layers to adapt them for our specific image classification task. The ResNet50 model was initialized with input shape (224, 224, 3) and removed the top classification layer where the include_top = False and this is the same fine-tuning used with the Xception model. Performing all these hyper-parameter tuning which includes the number of layers in CNN, epochs values, activation function, batch size, and Optimizer helps to build a good model for the image classification task. Previous authors Kumar et al. (2023) Joseph et al. (2023) use the CNN model to detect facial expressions and also they mention in the papers CNN is a good architecture for the classification task and good when working with the image dataset. While conducting this study the author agrees with the point that the CNN works well with the image classification task but the one more important thing is that also hyper-tuning of the model helps to increase the accuracy and other parameters. Also, the author hyper-tuned the pre-trained deep-learning models ResNet50 and Xception so both models also give good results and accuracy.

7 Conclusion and Future Work

The author discovered the integration of deep learning-based facial expression recognition models with personalized song recommendations in user experience. The main work of this study is to learn about how the model is optimized by using hyper-parameters such as learning rate, optimizers, batch size epochs, etc., for better accuracy and versatility of these systems. Using the Convolutional neural network CNN with hyper-parameter tuning and advanced pre-trained deep learning models such as Xception and ResNet50 in our research, the author observed significant improvements in facial emotion recognition accuracy. The design and implementation of a music recommendation user-defined function (UDF) provided an example of how to create the easiest, time-appropriate function to classify the different songs with emotion. The CNN model with hyper-parameter which the author created gives an accuracy of approximately 65%, The two Pre-trained models ResNet50 give an accuracy of 51.75% and the second model Xception gives an accuracy of 70%. By performing all this hyperparameter tuning the author concludes that the hyperparameter tuning plays an important role in increasing the model performance and accuracy.

In the case of facial emotion recognition, the author applies the two advanced pre-trained deep learning models to use other pre-trained models and their combinations to make an even better accuracy and better prediction. Build real-time applications for taking the attendance of students and colleagues in college, and offices both at government levels and also on a large scale companies even real-time applications also used while performing therapy by doctors. There is no need to perform hyper-tuning, all the models are hyper-tuned in this research. The dataset used in this study is good but try out different datasets that are available for free on several platforms like Kaggle but make sure and check the ethical consequences and secure user privacy & security.

References

Asha, M. L. A., Rafi, M. A., Ahamed, M. S. and Imran, M. H. (2024). Suggesting playlist and playing preferred music based on emotion from facial expression, *2024 3rd International Conference for Innovation in Technology (INOCON)*, IEEE, pp. 1–5.

Chankuptarat, K., Sriwatanaworachai, R. and Chotipant, S. (2019). Emotion-based music player, *2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST)*, IEEE, pp. 1–4.

Chaudhry, M., Kumar, S. and Ganie, S. Q. (2023). Music recommendation system through hand gestures and facial emotions, *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*, IEEE, pp. 1–7.

Chollet, F. (2017). Xception: Deep learning with depthwise separable convolutions, *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1251–1258.

Gobinath, A., SNS, A. K., Athithyan, M., Anandan, M., Rajeswari, P. et al. (2024). Emotional harmony through deep learning: A facial expression-based music therapy, *2024 Third International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, IEEE, pp. 1–5.

Gupta, N., Agarwal, S., Joshi, K., Gupta, V. K., Shukla, S. K. and Singh, G. (2023). Intelligent music recommendation system based on face emotion recognition, *2023 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)*, IEEE, pp. 110–115.

Hanafi, Q. N., Sulaiman, S. and Mahamad, S. (2023). A web application to recommend songs based on human facial expressions and emotions, *International Visual Informatics Conference*, Springer, pp. 76–86.

Hu, B., Shi, C. and Liu, J. (2017). Playlist recommendation based on reinforcement learning, pp. 172–182.

Joseph, M. M., Varghese, D. T., Sadath, L. and Mishra, V. P. (2023). Emotion based music recommendation system, *2023 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, IEEE, pp. 505–510.

Kim, J.-H., Kim, B.-G., Roy, P. P. and Jeong, D.-M. (2019). Efficient facial expression recognition algorithm based on hierarchical deep neural network structure, *IEEE Access* **7**: 41273–41285.

Kumar, R. and Rakesh (2022). Music recommendation system using machine learning, *2022 4th International Conference on Advances in Computing, Communication Control and Networking (ICAC3N)*, pp. 572–576.

Kumar, T. V., Rajasekaran, P., Prabhu, S., Pratheeks, V., Mageshpoopathi, S. and Prasath, R. V. (2023). A deep learning based system for detecting stress level and recommending movie or music, *2023 International Conference on Sustainable Computing and Smart Systems (ICSCSS)*, IEEE, pp. 20–24.

Lopes, A. T., De Aguiar, E., De Souza, A. F. and Oliveira-Santos, T. (2017). Facial expression recognition with convolutional neural networks: coping with few data and the training sample order, *Pattern recognition* **61**: 610–628.

Raja Sekaran, S. A.-P., Poo Lee, C. and Lim, K. M. (2021). Facial emotion recognition using transfer learning of alexnet, *2021 9th International Conference on Information and Communication Technology (ICoICT)*, pp. 170–174.

Ravi, R., Yadhukrishna, S. and prithviraj, R. (2020). A face expression recognition using cnn lbp, *2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 684–689.

Singh, A., Sharma, S., Singh, B., Jha, S. K. and Mishra, H. O. S. (2023). Smart song recommendation system using machine learning, *2023 9th International Conference on Signal Processing and Communication (ICSC)*, pp. 609–614.

Singh, K. K. and Dembla, P. (2023). A study on emotion analysis and music recommendation using transfer learning, *Journal of Computer Science* **19**: 707–726.
ΩÁlvarez et al.

Álvarez, P., Zarazaga, F. and Baldassarri, S. (2020). Mobile music recommendations for runners based on location and emotions: The dj-running system, *Pervasive and Mobile Computing* **67**: 101242.