ENHANCED SMART MUSIC CONTROLLER BY APPLYING CNN IN IoT

A PROJECT REPORT

Submitted by

Piyush Raj [Reg No: RA1611008010339]

Nitish Kumar [Reg No: RA1611008010319]

P J Sahrudh [Reg No: RA1611008010363]

Shivaditya Singh [Reg No: RA1611008010195]

Under the Guidance of

Ms. M. Safa

(Asst. Professor, Department of Information Technology)

In partial fulfillment of the Requirements for the Degree of

BACHELOR OF TECHNOLOGY

DEPARTMENT OF INFORMATION TECHNOLOGY

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

APRIL 2020

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

KATTANKULATHUR-603203

BONAFIDE CERTIFICATE

Certified that this project report titled "**ENHANCED SMART MUSIC CONTROLLER BY APPLYING CNN IN IoT"** is the Bonafede work of **"Piyush Raj [RegNo:RA1611008010339], Nitish Kumar [Reg No: RA1611008010319], P J Sahrudh [Reg No: RA1611008010363], Shivaditya Singh [Reg No: RA1611008010195],** who carried out the project work under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or award was conferred on an earlier occasion for this or any other candidate.

Ms.M. Safa

GUIDE

Assistant Professor

Dept. of Information Technology

Dr. G.VADIVU

HEAD OF THEDEPARTMENT

Dept. of Information Technology

Signature of Internal Examiner

Signature of External Examiner

ABSTRACT

It is important to implement an alternative approach to the smart home system and understand the human using the emotional states along with environmental parameters, just using the facial recognition increases the error chances, instead it also relies on the environmental conditions to deduct the subjects' emotional states more accurately. Smart Systems are evolving in every sense both in how they are made and on what technologies they are made through. Internet of Things is a major technology that provides an important part in creating a complicated convoluted interconnected system which do not require the human intervention to work and send data to other interconnected devices. The technologies including Facial Recognition, Cloud Computing along with the use of external sensors can be interconnected to produce a smart system which would have failed if not the processing power of the Cloud had not been used. Facial recognition is achieved using a class of deep neural network through Convolutional NeuralNetwork.

ACKNOWLEDGEMENT

This project would not have been completed successfully without the guiding hand of Ms.M.Safa, our guide and our panel member as well, whose valuable recommendations helped us garner a lot of traction in our mission to help people in their daily lives in making their emotional state better. The counsel of Mr. P. Savaridassan, our panel also helped us gather a lot of constructive feedback aiding our project completion. We would further like to acknowledge our parents, and the IT department for enabling the availability of resources as criticaljunctures.

Date:09-05-2020

Author

Nitish Kumar [RA1611008010319]

Piyush Raj [RA1611008010339]

P J Sahrudh [RA1611008010363]

Shivaditya Singh[RA1611008010195]

TABLE OF CONTENTS

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

FER	Facial Emotion Recognition
LDR	Light Dependent Resistor
DHT-11	Digital Humidity and Temperature sensor
ReLU	Rectified Linear Unit
PMM	Power Management Modules
IoT	Internet Of Things

CHAPTER 1

INTRODUCTION

1.1 GENERAL:

1.1.1 IMPORTANCE OF EMOTIONAL STATE

It is an important aspect of every human self which isn't meant to be constant and shouldn't be for a long time, which also creates an impact on the person also could relieve stressfulness or even presence of many states at once could lead to a problem such as feeling congestion which makes life unhealthy. The states are disturbed with the involvement of technology in people's lives, pleading for attention all the time. Many studies show that indeed causes problems in people's lives such as getting disconnected from the society, in the worst cases leading to depression which is fatal.

There are number of emotions in a human being and understanding them is almost like understanding colors, both of these have tons of properties in them the more they are researched not as diverse as colors but there are many, but the project focuses on almost eight of these emotions to get a streamlined result to understand human emotional states.

Emotional states are the ones that can predict a human's condition and hence is really necessary to focus upon, they are the pillar of the project and can't be ignored at all.

1.1.2 UNDERSTANDING EMOTIONAL STATES

Nowadays emotional states are important and effect people more than anything and its continuous fluctuation is a causing a disadvantage and could be understood through SADL (Social Activities of Daily Living), or when simplified means through the various activities and surroundings in these conditions. Understanding the state of surroundings tells a story how an emotion is present.

Understanding Human Emotions is practically impossible so technology of facial recognition and Cloud along with IoT are used so to create a model that can predict the emotional states that a person

could be present in by detecting their facial emotions. Emotional states are as important as a person as that defines
what the persons current emotional state is, and the goal relies heavily on this prediction and hence the accuracy is necessary to be as good as possible.

Overtime it has become easy to predict human emotional by training a model based on older data states but even better technologies will be invented that would focus on other body parameters that would increase the accuracy even much more which at the present time is quite hard to receive , but open source technology is what is allowing everyone to have this power to experiment.

1.1.3 USING STUDIES AND TECHNOLOGIES TO SOLVE THE PROBLEM

There have been several studies to understand human emotions and their impact through parameters not surroundings as the focus to the result. The project focuses more on using the environmental conditions of where a person is present and try to change it to more acceptable set using a medium that is understood and accepted easily using music. The music here is a catalyst to balance the emotions which could be unnecessary and depending on the parameter the music tends to adjust automatically to serve its purpose to sooth the emotional state before it becomes fatal.

1.2 PURPOSE:

1.2.1 MOTIVATION:

Day by Day the Security is becoming complex and stressful. Technology helps us push the current generation technology to create even more secure and dependent solutions and thus will be indeed the next generation cutting edgetechnology.

Home Automation is seeing leaps and bounds in its usage and its daily life uses as it does provide all the benefits to a person, it does from connecting to their favorite contacts or to

recommend them different life styles along with the power and technology of IoT and Cloud both of them which aren't slowing down in the near future.

There are untapped ideas and implementation in this field of technology, but this project does what most other systems fail to do which is increase the accuracy through facial detection this might not be a huge jump in detection but it is necessary in the near future to improve these systems which could be the possibility that our team is out looking.

There are a number of emotions in a human being but only focusing on a few is important so to streamlinefor better results among users. Two of the major emotions are Sadness and Happiness, out of these two emotions a person has one of them always but there are a number of them as well , direct emotion detection is not possible as machine cannot read human minds but can predict human emotional states by understanding human emotions for better result.

1.2.2 PROBLEMSTATEMENT:

Yes, there are solutions to automate and change the appliances, but as moving move forward in time the need will increase to automate more even without the human interference and also provide better solution using better options and hence the system detects the mood using various conditions such as weather, human activity etc, to analyze the mood and adapt the interconnected Systems. For example, Amazon's Alexa detects emotion, but it doesn't consider external factor while making decision. Our model increases the emotion detection accuracy by including the external factors thus making our model moresuccessful.

The aim is to maximize accuracy in detecting human emotional states using facial recognition and environmental conditions as alone one or the other wouldn't have been better so it is better to use both instead.

1.2.3 APPROACH:

The solution is solely based on IoT first and then the various systems will be attached accordingly .There will be a limited reasonable parameters to the system and the system when activated tries to understand the person and categorize his mood and adjust the system to that like the lights are dimmed ,fans gets faster, music genre changes so that it fits perfectly as needed.

The whole procedure is divided into three major components so that none of them loose focus: -
A) Collecting environmental data B) Facial Recognition Data C) Training a Data Set

1.3 IOT

1.3.1 Introduction to IoT

Iot abbreviated as Internet of Things is one of the important technologies in the present day as it helps to create an interconnected system that does not require human intervention at all time to function properly. It contains multiple sensors that allows it to communicate with the cloud if necessary, which broadens its usage even more, it also addons like a bluetooth module or WIFI module that will allow it to be connected and controlled by devices which support these technologies.

There are microcontrollers created and distributed by different industries so that they can be utilized with those add ons for better use, but as the size of the device increase the number of pins present on it increases increasing its capability to connect even more devices for better functionality.

The project utilizes ESP-32 as the base microcontroller to connect to other devices so that it works properly and connect to other devices when necessary to fully function and fulfill the works of the user.

1.3.2 Applications of IoT

1.It is used to create automated systems which usually do not require human intervention to function.

2. It has multiple pins, both ranging from digital to analog pins and the number of external parts that could be attached are till they are produced.

3.Easily gather and monitor data without much human effort as they have this powerful system that have developed far.

4.Easily attach new devices to it as they have a number of analog to digital and digital to analog pins throughout the board and also has a usb to connect on a powerful device to upload code and modify the device as necessary.

5. IoT is fairly used in many sectors of technology through mobiles to windmills the possible number of outcomes are extremely unpredictable.

1.3.3 Advantages of IoT

1. It allows to modify the system as much as possible through giving full access to components also allowing to upload own source code to modifying the attaches systems which makes it as easy to use as an open sourced system.

2. The system supports devices that have the capability of analog or digital connections as the system supports specific data to specific module types.

3.The thing that sets apart from other systems is that it also allows to be connected to cloud which enhances the performance, reliability and usages vastly.

1.3.4 Challenges of IoT

1. As the number of interconnected systems can be huge as it is capable of doing, it is very likely to have some kind of security issue if not build with care and if in a corporate system it can be hard to rectify.

2. Government still does not back the technology fully but as soon as it is supported more applications will occur and more features can be produced.

3.Scaling systems can be hard in these systems as the number of interconnected devices are huge and in most use cases, they depend on one another to function properly which can cause scalability issues.

4.Monitoring the workflow can be hard if there is any issue in one of the systems in largely scaled systems, as the track of the number of devices can be lost to the sheer number of the connected devices.

1.4 CNN

1.4.1 Introduction to CNN

It is a type of deep learning technology that is capable of identifying things upon training and creating a model using facial recognition technology. Then the trained model can be used to identify patterns in human faces which are stitched to the type of model trained for which it might be used, but is mainly categorized to identify faces and segmentation of facial patterns. It has evolved from the past and does a lot of works ranging from easier identification of people to tracing their identification.

1.4.2 Steps of CNN

These are the steps (or Plan of Attack) to follow for Convolutional Neural Networks:

1. Convolution Operation:

   In simply scientific terms, convolution is a capacity gotten from two given capacities by combination which communicates how the state of one is adjusted by the other. That can sound confounding for what it's worth, yet to exacerbate the situation, we can investigate the convolution equation:

$$(f * g)(t) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau)\, g(t-\tau)\, d\tau$$

Fig 1: Convolution formula

These are the three elements that are included in convolution operation:

- Input image

- Feature detector: It is also called as a kernel or a filter. Instinctively, the matrix portrayal of the input picture is duplicated element wise with the element indicator to deliver a feature map.
- Feature map: convolution is applied on the input image using a feature detector (*convolution filter)* to produce a *feature map*.

2. Max Pooling:

After a convolution activity we perform pooling to diminish the dimensionality. This empowers us to diminish the quantity of parameters, which both abbreviates the preparation time and battles overfitting. Pooling layers downsample each component map freely, diminishing the stature and width, keeping the profundity(depth) flawless.

The most well-known sort of pooling is max pooling which just takes the maximum incentive in the pooling window. In opposition to the convolution activity, pooling has no parameters. It slides a window over its info, and just takes the maximum incentive in the window. Like a convolution, we determine the window size and step.

Types of pooling that can be used:

- Mean pooling
- Max pooling
- Sum pooling

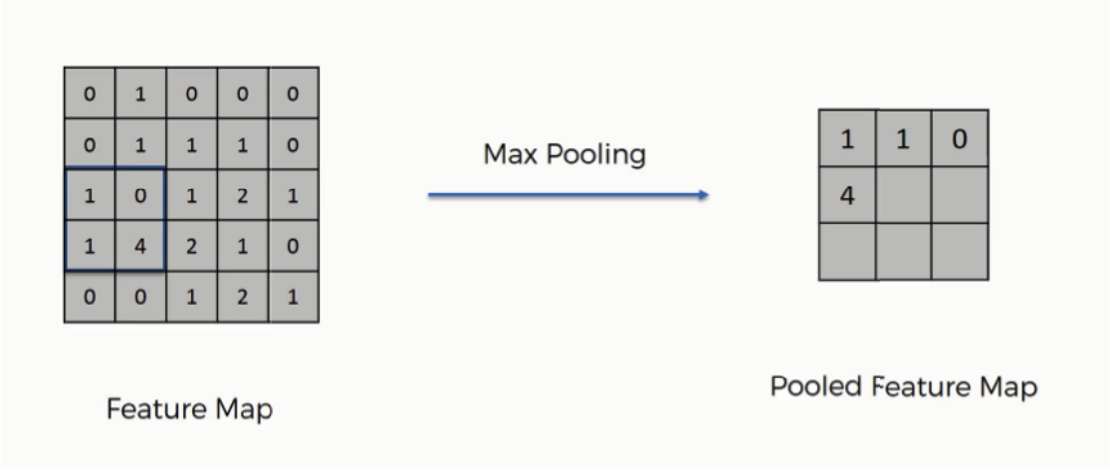

Fig 2: Max Pooling

3. Flattening:

Subsequent to completing the past two stages, a pooled highlight map must be at this point. As the name of the progression progresses, it is going to straighten the pooled include map into a section like in the picture underneath i.e. it is converted into 1D array for inputting or it can be said as it is converted into a single long vector for ANN.

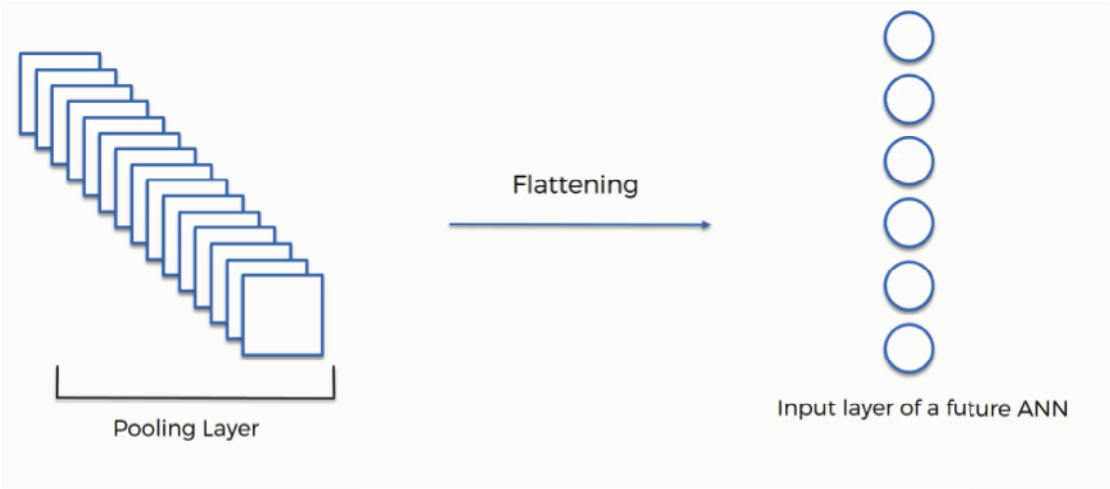

Fig 3: Flattening

4. Full Connection:

Then ANN and convolutional neural systems combine as we add the previous to our last mentioned.

It's here that the way toward making a convolutional neural system starts to take a progressively sophisticated and complex turn.

The job of the ANN is to take this information and join the highlighted features into a more

extensive assortment of traits that make the convolutional organize progressively fit for grouping pictures, which is the entire reason from making a convolutional neural system.

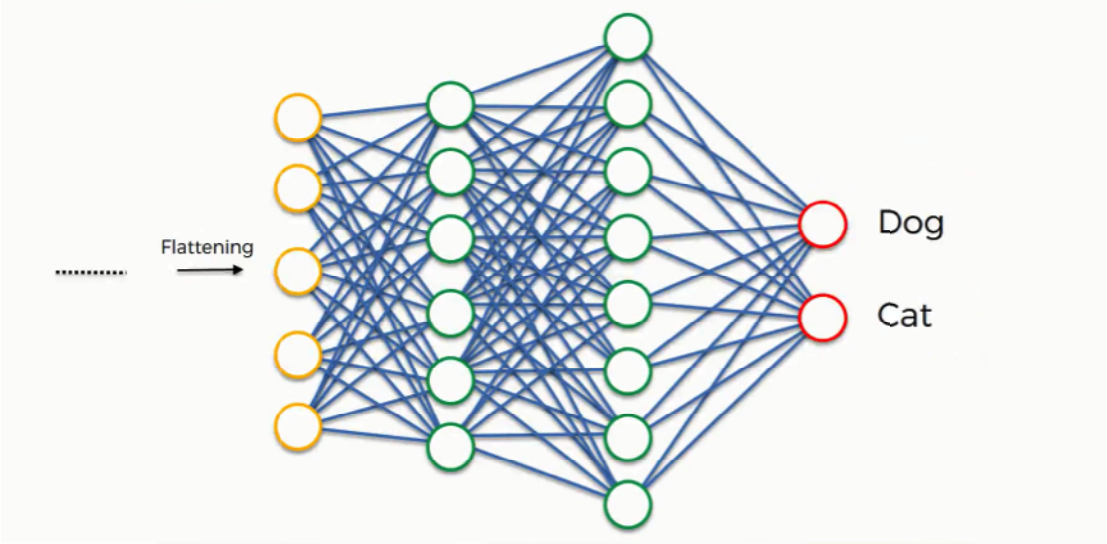

Fig 4: Model Connection

1.4.3 Applications of CNN -

1. Can be used to identify faces upon training it with a number of faces from a data set which will create a model that will do the work.

2.It is also used for identification, identification and recognition work in different ways as they require much more details to differentiate between two data and hence more training and better model for better identification.

3.Many details in a picture are unseen to the human eye or they are missed due to the minor detail but this technique helps to identify even of the minute detail which are missed in most cases.

4. It is used to classify content from sceneries, pictures in sequence or groups so that they can used easily later on.

5.It is also used to break down an image from its full form to classify items in an image so that they can be studied and identified easily.

6. It can be also used in searching images and image verification that are a common feature in most search engines that improves search engines capabilities.

7. They can be used in identifying drugs which is a major problem in pharmaceutical industries which could visually differentiate drugs but cannot in terms of contents but this could also of big help.

1.4.4 Advantages of CNN

1. It reduces the number of input values that are needed to train the model for recognition.

2.It also accepts input size from small to large sizes without any restrictions.

3. Many other methods are as fast as this method and is really fast using techniques such as mathematical formulas and more.

4. This technique is also very effective as it has multiple layers of work, which bring out the accuracy but indeed increase the complexity and due to the presence of multiple features it has improved performance.

5.Even though the inputs ranges to large sizes as well, the need for large data sets is not necessary for training the data model in this technique.

1.4.5 Challenges of CNN –

1. This technique is so fast when put in use with real life scenarios it is pretty hard to make it function as it requires a faster graphical processing unit to work properly.

2. Even through the recognition is pretty accurate in identifying images most of the time it is quite challenging when the images are not in their normal orientation and get twisted.

3. Overfitting is one of the major problems on almost all kinds of models and this is to be seen when

the inputs size gets larger and larger , and even through scientific research it is not quite possible to make good model through good data set but only if it is trained well enough.

4. There is no flexibility in the models as they can't or they fail to recognize even when their orientation is messed even a little bit visually, it gets confused just as like a human mind in many cases.

1.5 Cloud Computing

1.5.1 Introduction to Cloud Computing

To understand Cloud Computing we necessarily need to understand that storage capacities have come to a halt locally or they just have become more expensive over time that huge storages is practically extremely expensive and in affordable if not used on a daily basis , along with that to perform such huge scale processes the need of graphical performance is also required and if everyone wants to have this setup locally it would be quite impossible , hence to solve this issue companies provide cloud services through cloud providers which enable any user or person on a low fee to access storage and necessary graphical performance for a low affordable cost and when the work is done they can be closed.

There are many types of cloud providers ranging from open source cloud, private cloud, public cloud in which some of the cloud providers charge money to use their services and some of them are running due to contributions from public and in both of these cases security certainly varies and gets more vulnerable if exposed to public.

1.5.2 Applications of Cloud Computing

Cloud Computing is possible through the internet and is only due to presence of various cloud providers such as Azure, Aws etc.

1.Online file storage is one of the features that most of the cloud services provide and most of the service even offer a limited space to try out their service. If the necessary storage is not enough more storage can be bought commonly through subscriptions.

2.Many cloud provides are now boosting this idea of cloud security where the security service works

from the cloud but does learn from data samples from previous iterations and hence keeps getting better rather than older security models which tend to be hardcoded before usage only on local devices.

3. There is also another new application that is getting manipulated for good by the users that is running and shifting their businesses to the cloud commonly nowadays knows as E-Commerce which relies much on the cloud such as for the hosting, storage, processing power etc.

4. Cloud services can be used for data and recovery in the case of disaster as the local storage might be indeed get destroyed permanently adversely affecting the processes.
5.Cloud services are now also focusing for enterprise to provide infrastructure as a service through a monthly or yearly subscription which gets renewed from time to time to continue the service.

6. The cloud services have a number of capabilities; they also provide graphical processing power for the user for short time to use when needed charging for only what is used and for what time it is used.

1.5.3 Advantages of Cloud Computing

1. There is huge savings on money spent by organizations or users to scale their equipment in time for an upgrade and the shift to cloud services can be cheaper than to continually using upscaling and downscaling if necessary, locally.

2. Security is better in cloud services as the cloud providers provide a robust inbuilt security system that grow with each generation and also can smartly detect any dangers coming to data.

3.There is huge flexibility when compared to local upgrades as upgrading over the cloud is far simpler than upgrading locally which would take much more time and even when upgrading over the cloud it also provides extra advantages for the same upgrade which would be an added benefit.

4. The cloud computing services provide mobility to organizations and users to access and compute their data from anywhere in the world as to access the cloud services they only need an internet connection.

5. Collaboration is upscaled or can be seen progressively increased between users or inside organizations as they have the capability of working on the same system without having a separate

system to get the work done and so the features become more useful.

6. There is one main concern with every user is the loss of data in case of disaster as the data used to be normally stored inside physical hard drives without any proper protection and once it is broken it is gone forever , but cloud providers provide chance for disaster recovery as when a disaster occurs the data is necessarily stores in another place in a server so it becomes easier to rely on this system.

1.5.4 Challenges of Cloud Computing –

1. The first and biggest challenge of this technology is lack of understanding and low resources to understand and use it, even though understanding the use in real world scenarios is easy and resourceful many find the skillset very difficult.

2. There are numerous amounts of cloud providers, new ones developing and some even in the belly of bigger corporations.

3. Another problem or simply the challenge of choosing a cloud platform for enterprises or single users is the understanding of pricing model for usage and some have good pricing but some have a terrible one with no consistency in pricing for all the platforms which is a major problem.

4. Security is now better than ever on platforms but it also depends on the amount of trust there is in that platform to respect and protect user's privacy, it comes down to looking at the platform's history and personal beliefs on which platform to choose.

CHAPTER 2

LITERATURE REVIEW

[1] The headway made in HSH based on IoT innovations with the end goal of portraying (challenges) and clarifying the fundamental difficulties or challenges of this research study, reviewing the rising idea of Health -Smart Home and picture potential uses of these frameworks, for example, remote monitoring and social alarmsystems also discussing about the use of the technology like Raspberry Pi, mobile phones and embedded systems can be used to solve the upcoming challenges caused. The implementation of image processing and embedded system has provided the efficient and cheap equipment's which helped to deploy the system and enhance the home medical treatment.

[2] The key point is to confirm that emotions can be recognized from various things such as facial expressions, voice, sentences, body temperature and so on. In patented technology by Panasonic (Panasonic), we recognize feelings form talking content and sounds of voice. In addition, the Empath API (Smart medical) recognized feelings using the physical sound of speech. Microsoft's Emotion API (Microsoft) can recognize emotions from facial expressions.

[3] The aim is to develop and implement a novel, automatic emotion detection system and facial recognition system based on AI (Artificial Intelligence) and IoT (Internet of Things).From the study we can design and actualize a non-invasive and wearable emotion recognition technologies in IOT industries (smart Industry) .

[4] The proposed benefits from the interconnectivity of Internet of Things (IoT) is to perceive furthermore, adjust to complex negative passionate conditions (emotions) of workers (e.g., stress, frustration, etc.). Adopting a deep learning approach for emotion ordering (classification) through an iterative process by add and remove of large amount of sensor signals from different modalities.

[5] A new approach for emotion recognition system using a deep learning approach using big data is used, the Big Data accommodates the speech and video. In the proposed framework, a speech signal is first processed in the frequency domain to obtain a Mel-spectrogram, which can be treated as an image. At that point this Mel-spectrogram is taken care of to a convolutional neural system (CNN).

[6] CNN algorithm is used to distinguish the expression of the person who pressed the emergency signal button on the device to reduce false alarm. A very low complex Convolutional Neural Network (CNN) architecture has been developed, to perceive the articulation on the end gadget itself, describing about a system where citizen security can be maintained. It Uses the LoRa Technology to ensure the long-range service. The system comprises of nodes which keeps logging its state to the server based upon the emotion facilitated the node end. This is used to reduce the false alarm which could be raised. Main focus is kept on the cost-efficient system to provide the logging system which could keep track of the citizens.

[7] The Key Point is to propose an IoT-based emotion recognition system for affective state mining. Human psycho physiological observations are collected through EMG, EDA and ECG medical sensors and analyzed through a deep convolutional neural network (CNN) to decide the covert affective state, describing about the wearable IoT subsystem which could help in maintaining Human psycho physiological observations.

[8] This presents an IoT-based quick and precise face detection smart system with spatial correlation of associated face part regions. The face detection algorithms are actualized in edge devices, raspberry Pi3,helping in human face detection based upon an algorithm which calculates the spatial correlation of various associated face parts with respect to the environment states and using these data they define a correlation among the part which results precise face detection using this spatial correlation.

[9] Image processing technology is used to validate the individual to enter in home. For image processing, we will use pi camera module. Pi camera module is attached to Raspberry pi, and it helps to store different faces in the databases.

[10]A bi-modal system can be used to combine the data obtained from facial expressions and emotions obtained from the speech signal. Emotion recognition can be characterized mainly using facial expressions and verbal expressions, aiming to establish a relationship between the vocal and facial expressions because of their higher involvement in impact caused on children with high functioning autism.

CHAPTER 3

PROPOSED METHODOLOGY

3.1 FUNCTIONALREQUIREMENTS:

- Data collection over the environment using sensors
- Data collection from fer2013 Dataset used for training the machine learning model for distinguishing the motion
- Preprocessing the data to convertitintothegreyscale48*48-pixel image format
- Feeding the grayscale image to CNN based emotion Detector to detect the emotion
- Storing the collected data from both the sensors the trained ML model over the Adafruit Cloud.
- Transferring the data over the cloud using ESP32microcontroller.
- Controlling the physical device based on the instruction received by the processed result from the application deployed on cloud in JSON format.

3.2 NON- FUNCTIONAL REQUIREMENTS:

- Simple User Interface
- Scalability.
- Cost-Effectiveness.
- Improved integration of Standardized Computer System.
- Enhanced integration and interoperability.

3.3 HARDWAREREQUIREMENTS:

3.3.1 ESP32:

ESP is a low cost on system chip which provides capabilities through wireless communication and Bluetooth connectivity. The ESP32 arrangement utilizes a Tensilica Xtensa LX6 chip in both double center and single-center varieties includes built-in for radio wire switches, filters, power enhancer, RF balun, low noise receive amplifier and PMM (Power Management Modules). ESP32 is made and created by Espressif Systems, a Shanghai-based Chinese organization, and is produced by TSMC utilizing their 40 nmprocess.

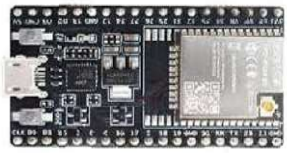

Fig 5- ESP-32 controller

3.3.1.1 PIN DESCRIPTION FOR ESP-32:

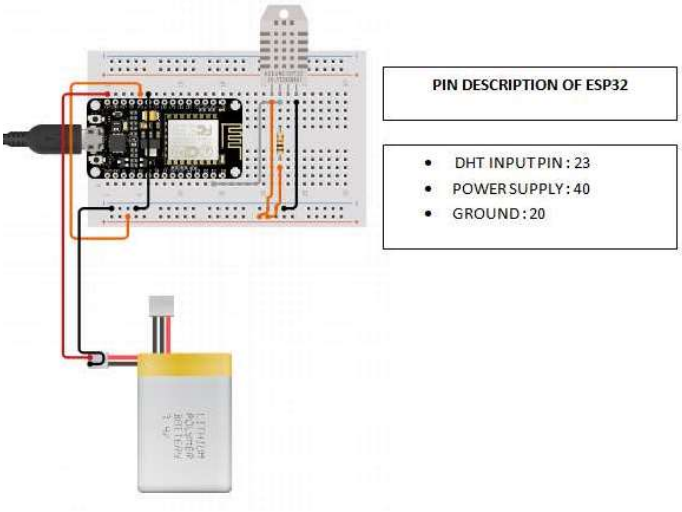

Fig 5- PIN DESCRIPTION FOR ESP-32 CONTROLLER

Esp-32 microcontroller has in total of 48 pins with multiple functions, and some pins cannot be used for development. **Pin 32** is taken as input pin to take input from the sensors **DTH-11**and**LDR.**Pin40isusedforthepowersupply;Pin40isusedtogroundtheentireboard.

3.3.2 DHT11:

THE DHT -11 is an affordable measurement tool for temperature and humidity which gives higher reliability and a higher amount of stability for the measurements of a situation. It utilizes a capacity humidity sensor and a thermistor to quantify the encompassing air, and lets out a digital signal on the digital pin. It genuinely is easy to utilize however timing must be taken care of before taking inputs.

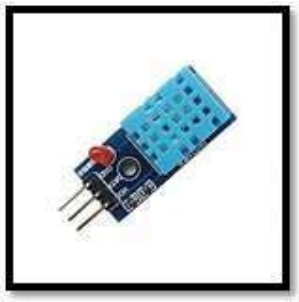

Fig 6- DHT-11 sensor

3.3.3 LDR:

LDR is an independent device which has a variable resistance and whose value changes either when the brightness goes low or high, the resistance value goes up or down in their respective cases.

They are comprised of semiconductor materials that have high obstruction. There are a wide range of images used to demonstrate a photoresistor or LDR.

Fig 7- LDR (Light Independent Resistor)

3.4 SOFTWAREREQUIREMENTS:

3.4.1 PYTHON:

Python is an interpreted, high-level, general-purpose programming language created byGuido van Rossum and first released in 1991.Pythons de sign philosophy emphasizes code reliability with the notable use of significant whitespace. Its language constructs object-oriented programming approach to help developers write clear, logical code for small and large-scale projects.

3.4.2 Adafruit IO Cloud:

Adafruit is a cloud service provider like many other providers similar toAmazon Webservicesorotherwhichdoesa lotofworkthatnormallocalstoragescannotdo

and provides with a lot of tools to use along with cloud storage to connect to other services.

3.4.3 SYDERIDE:

Spyder is a unique software tool that provides a combination of advanced features ranging from editing and analysis to debugging, more than this the features can be extendedusingitspluginsystem.

3.4.4. Arduino IDE:

Arduino is a cross platform IDE supporting major desktop environments which helps to write and run code on supported Arduino and on 3rd party boards with integration. ArduinocodeiswrittenintweakedversionsofCandC++languages.

3.4.5 Existing System:

In existing system, the emotion was detected using the image/video or using combination of video and sensor as wearable device to measure the heartbeat. This system did not consider the external factor which directly affected the human emotion and body such as temperature, humidity etc. Without the consideration of external environment, they missed out an important variable for emotion detection. For example, a person may be angry or may be irritated due to heat or hot temperature outside, which would have been considered same in previous system. Also, in some research it was found that some people react differently to some emotion compared to other people, so using only video/ image to detect emotion was not right. This system needed to be trained extensively to get accurate result and needed a huge amount of dataset to make that possible.

3.4.6 Issues in Existing Systems:

1. Expressions can never be identified with how an individual is feeling precisely in these non-attacking strategies.

2.Sensors have limitation to the area they cover in detecting environmental parameters.

3.Comprehension and changing in accordance with human behavior requires significant investment (time).

4.Model needed to be trained extensively to get an accurate result. This may cause problem of overfitting.

5. Experts are consistently improving and developing new strategieswhich do not invade or exploits human’s privacy.

6. There is a human involved the intricacy of the framework increases to provide the best results.

3.5 ArchitectureDefinition:

The following sub-sections explore the modules of the proposed architecture. The proposed system uses ESP32 micro-controller and takes input from two sources. One is the environmental parameters Humidity, Temperature and Light Intensity in the room using DHT-11 and LDR sensors respectively,and the second takes an image of the user in 48x48 format and then using the proposed CNN model the emotion of the user is detected and the data from these are stored on the cloud. Using these data, the application decides the playlist and intensity of light in theroom.

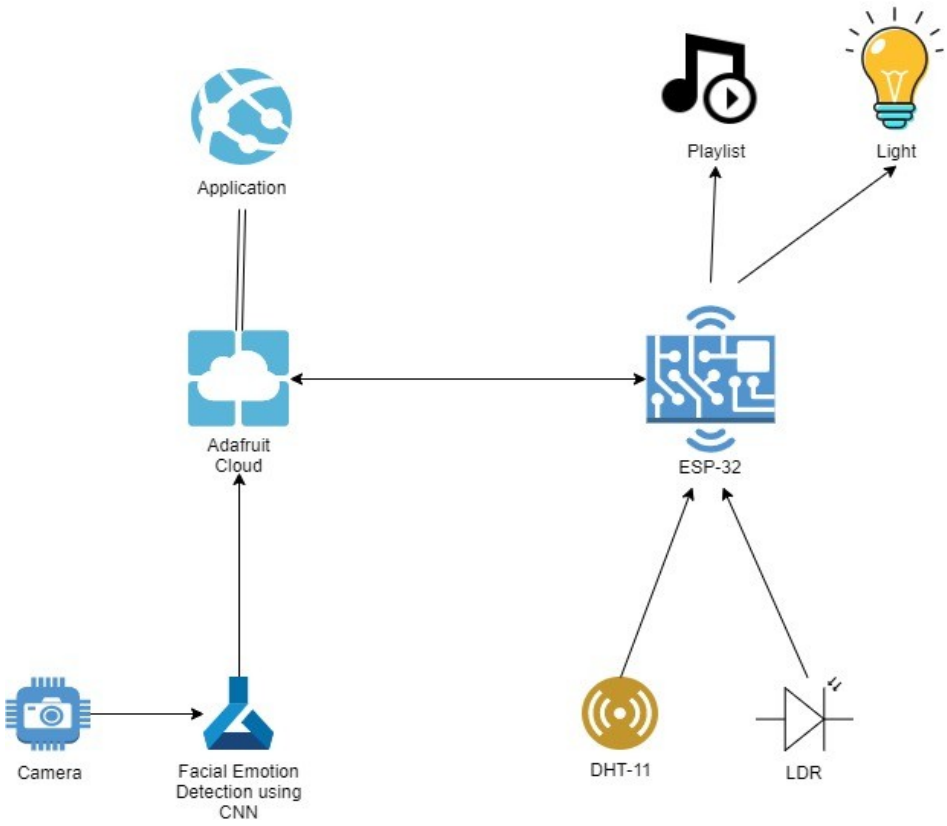

Fig 8– *System Architecture*

3.6 Input and Output:

I/O module controls the environmental parameters input and the output as the songs playlist and controls the intensity of the light in the room. The input parameters are the Humidity, Temperature and Light Intensity. To measure humidity and temperature DHT-11 sensor is used, and LDR sensor to measure intensity of the light. The input is sent to the cloud storage and is further used by theapplication.

The output of the system is the playlist and intensity of light in the room, that data is calculated by the application and is forwarded to micro-controller, which controls the music and light as per the instructions received.

3.7 Model for EmotionDetection:

For Facial emotion detection CNN (Convolutional Neural Networks) is used. OpenCV uses the facial features defined to detect the emotion and work on hardcore level. But CNN uses the pattern in the images of categories of emotions to guess the emotionin the image.

OpenCV is a very hardcore method to detect the emotion and thus accuracy method is low. To increase the accuracy, Convolutional Neural Networks have been implemented. As it uses the pattern in the images to detect the emotion thus as more the model is trained its accuracy increases. Also, CNN is easy to use as it detects the features at theneurons level.

CNN is a class of deep neural networks which uses supervised learning to detect visual imagery. Implementing CNN to detect facial expressions and conclude to emotional states of the user so that the system could adapt properly to the user.

Convolution is a form of mathematical operation that comprises of a combination of multiple functions in order to produce another function. Convolution is generally performed on input data by adding a convolution layer by using ra filer to reproduce a feature map. Pooling layer also gets added after the convolution layer, performing sequential dimensional reduction which reduces the number of parameters and its computations that results in minimizing the training time and controlling the problem of overfitting. One of the most widely used technique is max pooling, which takes the maximum value in each of its window and then decreasesthe featuremap size whilst keeping the importantinformation.

3.8Application:

Data from the input sources and the emotion detected from the image is forwarded to the cloud storage to use it as the constraints for adjusting the music and light in the room.

The Application is built over the data collected as the input constraints and then decides the music and its nature to be played and the light in the room. The collected data is forwarded to the microcontroller and it adjusts the music and light as per the instructions received from the application.

The solution is solely based on IoT first and then the various systems will be attached accordingly .There will be a limited reasonable parameters to the system and the system when activated tries to understand the person and categorize his mood and adjust the system to that like the lights are dimmed ,fans gets faster, music genre changes so that it fits perfectly as needed. The whole procedure is divided into three major components so that none of them loose focus: -

A) Collecting environmental data B) Facial Recognition Data C) Training a Data Set

CHAPTER 4

DESIGN ANALYSIS

In this chapter,the design analysis of the system has been discussed. The following diagrams below are UML diagrams, these diagrams are a standard way to visualize how the system has been created /designed.The UML diagrams are used to portray the behavior and structure of the framework.

4.1 COMPONENT DIAGRAM

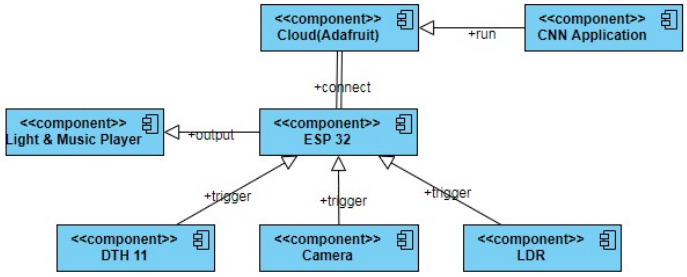

Fig 9: Component Diagram

The component diagram shows the orientation and organization of the physical components. The diagram is a perfectly suitable show of their inter connectedness, and how module is dependent on others.

4.2 CLASS DIAGRAM

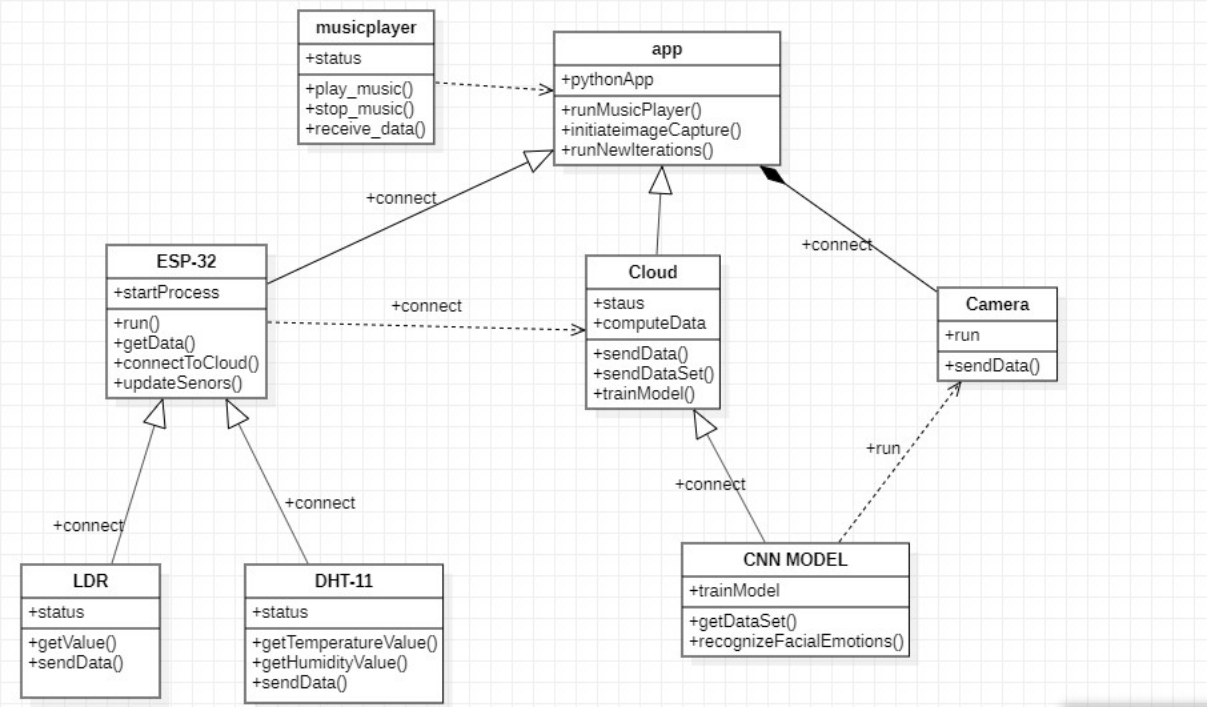

Figure 10: Class Diagram

The class diagram focuses on showing the relationships among the ESP-32 controller, the Application, sensors and the cloud. It defines the system by showing the classes, attributes and operations.

4.3 USE CASE DIAGRAM

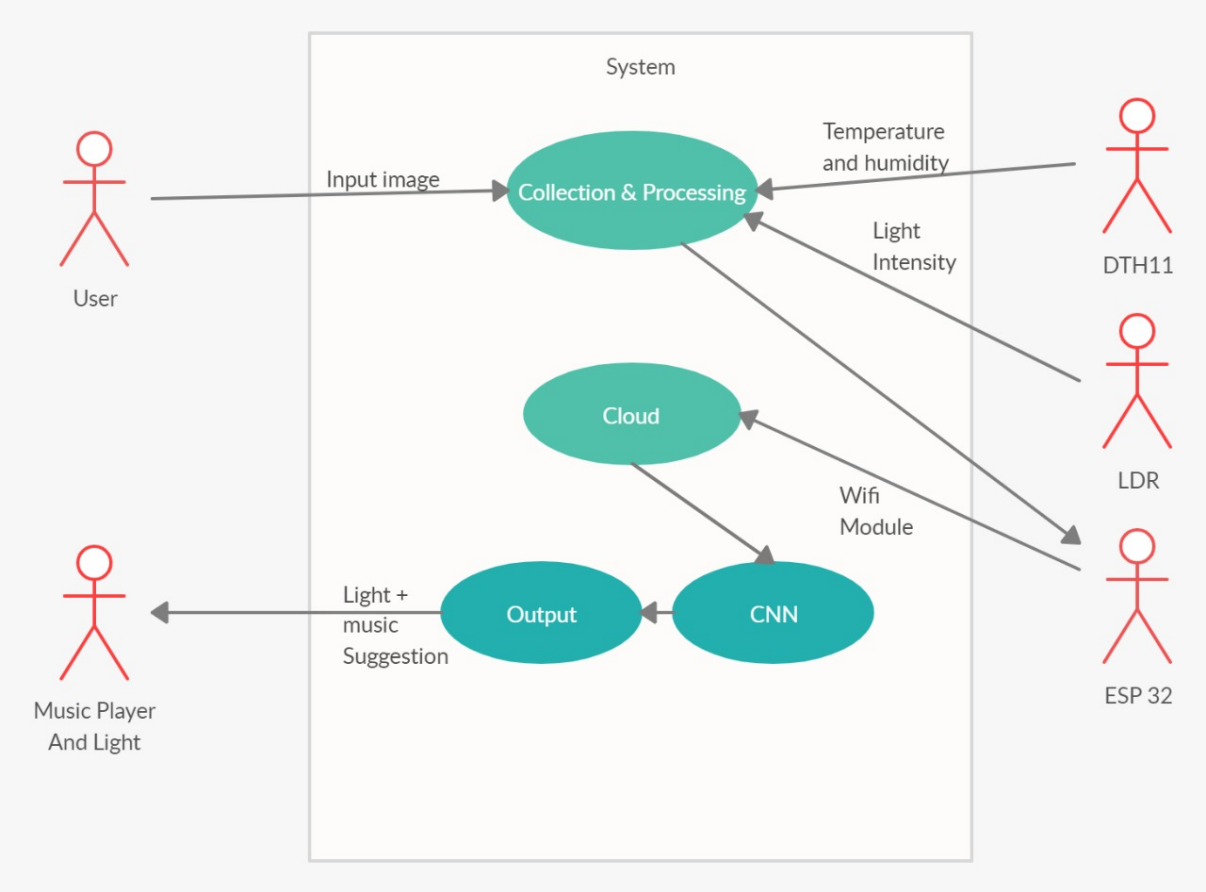

Figure 11: Use Case Diagram

The diagram shows how the system would interact with human presence.The System always tries to adapt to the User by sensing the emotional state and according to that if the user is sad it will play music to elevate emotional states and so on.

4.4 SEQUENCE DIAGRAM

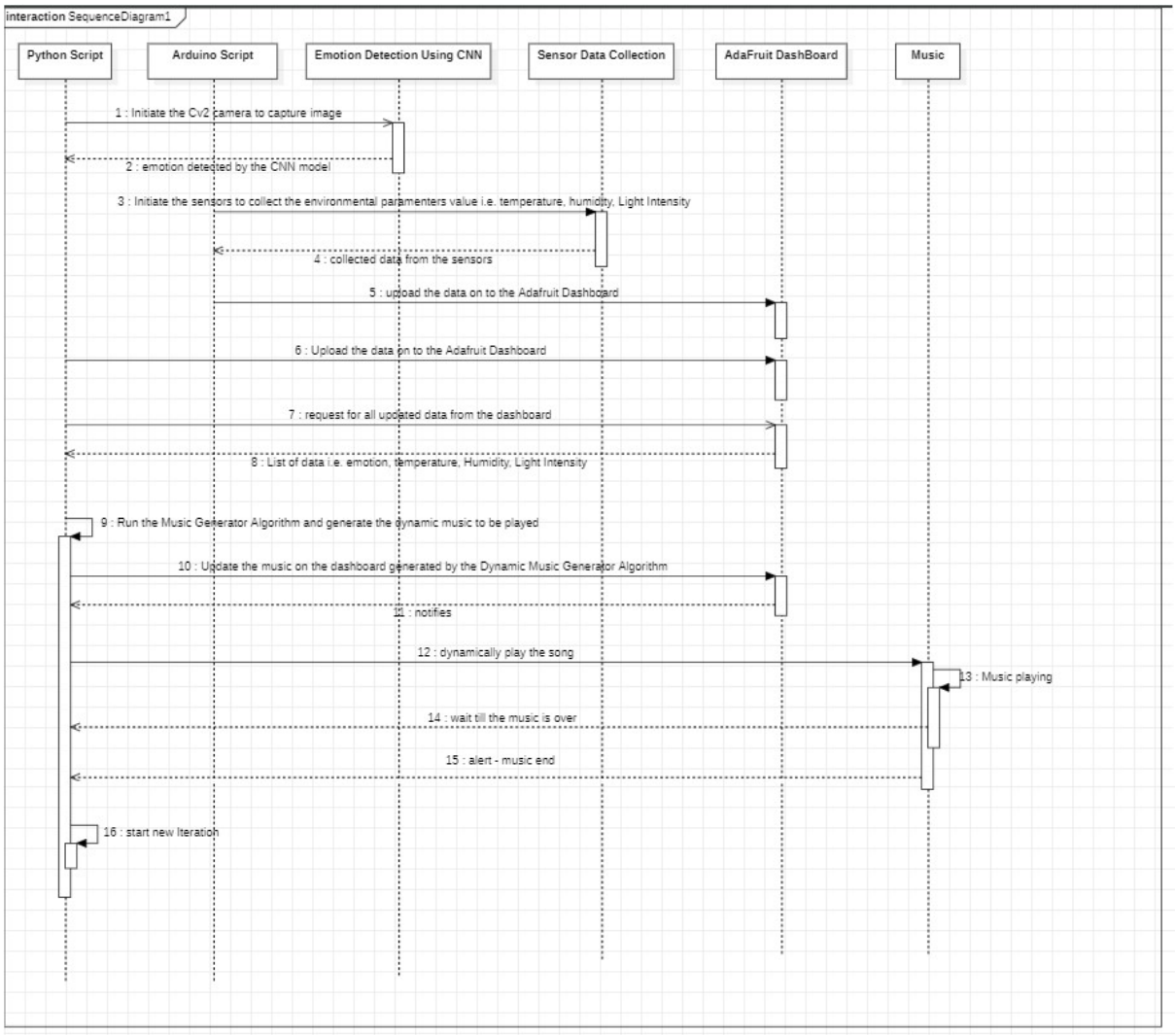

Fig 12: Sequence Diagram

The Diagram explains the interaction between the different modules more precisely, it is clear that one the major core of this system is the python app which runs and triggers majority of the other elements and almost every other element is dependent on some other module to run.

4.5 Activity Diagram

The diagram shows the execution and flow of the music controller system.

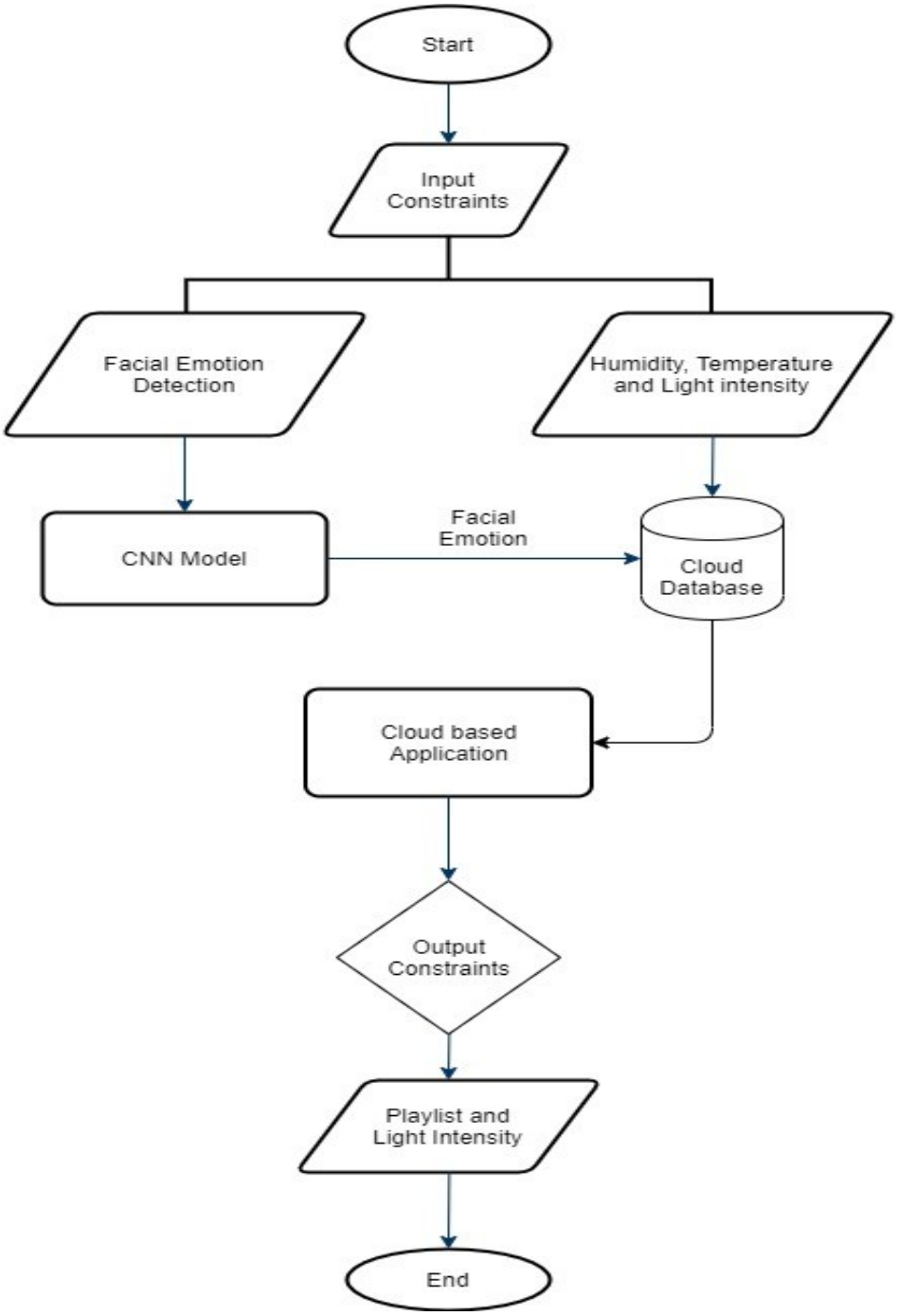

Fig 13 : Activity Diagram

CHAPTER 5

MODULE STUDY

The system has the following major modules-

5.1 ESP-32(input and output constraints):

The microcontroller ESP-32 administers the input and the output constraints. It accepts the temperature, humidity and the intensity of light in the room as the input constraints from the DHT11 and LDR sensors. Arduino code is launched on the microcontroller to process the input and send it to another module and to control the output devices i.e. the music on the speaker and theLED.

5.2 Facial Emotion Detection (CNN model):

It is a type of deep learning technology that is capable of identifying things upon training and creating a model using facial recognition technology. Then the trained model can be used to identify patterns in human faces which are stitched to the type of model trained for which it might be used, but is mainly categorized to identify faces and segmentation of facial patterns. It has evolved from the past and does a lot of works ranging from easier identification of people to tracing their identification.

Convolutional Neural Network is used to detect facial emotion off the user. The data setimplemented is Fer2013. It has 48x48 pixels images that are mapped to an emotion from oneof the seven categories of emotions i.e. 0: [Angry], 1: [Disgust], 2: [Fear], 3: [Happy], 4: [Sad], 5: [Surprise], and 6: [Neutral]. The overall process is done through three stages-

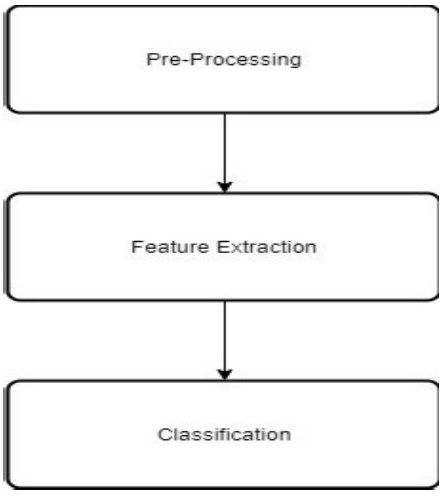

Fig 14– Facial Emotion Detection Workflow

Facial Detection Formula:

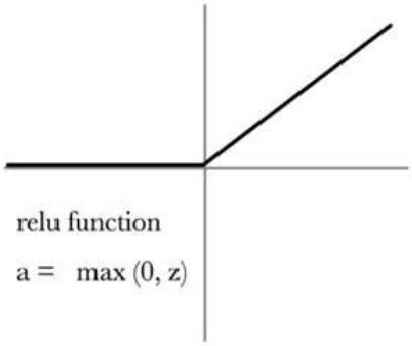

Fig 15- Relu Function Graph

$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$$

Fig 16- The Relu Function

$$G[m,n] = (f * h)[m,n] = \sum_j \sum_k h[j,k] f[m-j, n-k]$$

Fig 17- Kernel/Filter Formula

5.2.1 Pre-processing:

The image is reshaped, and the training and test datasets are classified that is to be used for modeltrainingandtesting.Sinceitisagrey-scaleimagethevalueofeachpixelisinrange0-255. So, changing the training and test dataset to float and then divide them by 255 so that values lie in the range0-1.

5.2.2 Feature Extraction:

Feature Extraction helps us to extract the most important features from the image. The convolutional layers are used and in addition to it using batch normalization, max-pooling, dropout and dense functions to extract features from the image.

After each set the features reduces thus using the important features for classification in the final layer. In Max-pooling, max from pool is considered and others are ignored which is done using the kernel size. Dropout is used to define the probability whether the neuron will be dropped or not.

5.2.3 Classification:

The last layer of the network is SoftMax which gets used for classification, due to the extracted facial features getting simply stored for training images. An image must fall in either of the seven categories of emotions. So, each image is classified to an emotion that the person in the image.

5.3 Music Classification Algorithm:

In this module the output response is the songs to be played. This algorithm takes the sensor value, and emotion state as the input and based on these input parameters it returns a songs to be played.

The return songs are dynamic in nature and depend upon the availability of the songs in each domain and also the song selection is dynamic in nature. I.e. for the same input parameter, the output will changes each time this algorithm is called. But the result will be distinguishable based upon the emotion. This algorithm provides the emotional state and environmental state seniority value. For this algorithm we have defined emotional state to have higher

```
emotion = (aio.receive(('major-project.facial-emotion')).value)
if temp >=15 and temp <= 25 and humid >=85:
    return "rainy"

elif(emotion == "angry")
    if (temp <= 50 and humid < 70):
        return "sufi"
    else:
        return "happy"

elif(emotion == "disgust")
    if(humid <= 50 and temp <= 30 ):
        return "happy";
    else:
        return "party"

elif(emotion == "fear"):
    if (humid <= 50 and temp <= 45):
        return "motivational";
    else:
        return "happy"

elif(emotion == "happy"):
    return "party";

elif(emotion == "surprise"):
```

Fig 18: Music Classification Algorithm

seniority value. Although, environmental State will impact on the final result. In some of the cases we defined the environmental state to have higher seniority than the emotional state. Based upon this seniority value the classification is made.

5.4 Adafruit Dashboard:

Fig 19: Adafruit Dashboard

Adafruit is a cloud service provider like many other providers similar to Amazon Web services or other which does a lot of work that normal local storages cannot do and provides with a lot of tools to use along with cloud storage to connect to other services.

This is the tools that are used to provide a dashboard to actually see what is going on internally, i.e. how the music is being played and the sensor results. This is a dashboard which keeps the track of every single data flowing through the systems and subsystems. Which helps us analyze the functionality of the system? This provides a open view to the sensor data and the music being played in the system which changes dynamically.

CHAPTER 6

SOFTWARE TESTING

6.1 UNIT TESTING:

It is the method toward testing each and every module made by the designer. The entire program is separated into various bundles which consist of little units of code. It improves the general structure of the module and refractors the code any place basic. These modules are attempted self-ruling free of various modules. They are attempted in a progressive solicitation likewise; it checks for redundancy. On the off chance that there ought to emerge an event of repetition it deletes the duplicate records. It also checks for run time error or problem and checks if the association/connection gave take them to the person/individual page. Favored point of view of performing unit testing is its ability to check each module solely which is strong in finding the humblest of smallest slip-ups. Since unit testing is finished at an in all regards beginning period the cost of testing is immaterial when appeared differently in relation to other testing. Modules which are too huge for unit testing can be surveyed using integration testing.

6.2 INTEGRATION TESTING

This is ensuing(next) stage after unit testing is done or performed. Once, every module attempted independently is away from any type of error or problem, these individual modules are merged together and attempted by and large. The crucial clarification behind playing out this test is to check for issues when all of the units are combined or joined. These can be done in different way, which are as following:

1. Top Down Integration: - Top-down blend or combine joins and tests all of the modules start to finish. Be that as it may, one burden of this testing is that it needs more stubs.

2. Base Up Integration: - The base up strategy is the opposite way around of top-down approach. Noteworthy modules are attempted last which can make issues in the midst of mix.

3. Big Bang Integration: - In this kind of testing all of the functionalities are fused and attempted simultaneously. This strategy is dependent upon the number of modules present. Lesser The modules logically fruitful it is.

4. Hybrid Integration – It is a blend or combination of all the above systems.

6.3 SYSTEM TESTING

System Testing is next stage after integrating testing. The entire(whole) item is checked for error or any type of bug. System testing is divided into further two categories:

1. White box testing
2. Black box testing

1. White Box Testing:

It is a testing procedure which is done by s/w engineers. In this programming testing strategy, the inside structure/plan/usage of the item being tried is known to the analyzer. The analyzer picks contributions to practice ways through the code and decides the fitting yields.

2. Black Box Testing:

Also called as Behavioral Testing, is a product testing strategy in which the inside structure/plan/usage of the thing being tried isn't known to the analyzer. These tests can be useful or non-useful, however typically practical.

6.4 REGRESSION TESTING

REGRESSION Testing is characterized as a sort of programming testing to affirm that an ongoing system or code change has not unfavorably influenced existing highlights i.e. features.

It is only a full or incomplete selection of effectively executed parts of code which are re-executed to guarantee existing functionalities work fine.

The testing is done to ensure that new code changes ought not effect-sly affect the current functionalities. It guarantees that the old code despite everything works once the most recent code changes are finished.

6.5 SMOKE TESTING

S/w testing that contains a non-comprehensive arrangement of tests that target guaranteeing that the most significant capacities work. The consequence of this testing is utilized to choose if a form is sufficiently steady to continue with further testing.

6.6 ACCEPTANCE TESTING

This is the last time of testing which is performed by or before clients. This testing is on a very basic level done to check whether the made thing satisfies the client's need. They are 4 particular habits by which affirmation testing can be performed. They are as following:

1. Client acceptance testing
2. Business acceptance testing
3. Alpha testing
4. Beta testing

TEST CASE ID	TEST CASE DESCRIPTION	STEP DETAILS	EXPECTED RESULT	ACTUAL RESULT	STATUS
001	Sensor Integration testing	Sending environment parameter (temperature, humidity, light intensity) value collected by sensors to dashboard	Display the all sensor data on the dashboard	Displayed the data collected by sensor in the Adafruit dashboard in Gauge Meter	**PASS**
002	Detecting Human Emotion	Python application detects the facial emotion and sends the data to the Adafruit cloud	Displayed the emotion detected by the CNN model.	Display the emotion detected by CNN model on the Adafruit Dashboard	**PASS**
003	Detecting Emotion from multiple faces detected in the camera	Multiple face is captured in the camera then most promising face's emotion is taken into account.	Display the emotion of the most promising face detected by the camera	Publish the emotion of most promising face detected from the camera	**PASS**
004	Play the music based on the emotion	Play the music based on the various environmental parameters and	Display the music to be played on the	Played the calculated music and name of song is displayed	**PASS**

		emotions detected	Adafruit Dashboard and play the music	on the Dashboard	
005	synchronicity of the application	Wait to Play the next music based on the emotion and sensor data until current song is finished playing or no song is being played	Do not play song or send song name to dashboard to the cloud until any song is being played.	System Waited till the moment when no more song is playing. Thereafter any song is getting played based on the emotion and sensor data.	**PASS**

Fig 20: Different Test cases with their obtained outcome

Test Cases, as above mentioned are performed to check and validate if the operations and functions involved in performing the analysis are being in correct manner or not.

CHAPTER 7

RESULT AND OUTPUT

7.1 MODEL ACCURACY:

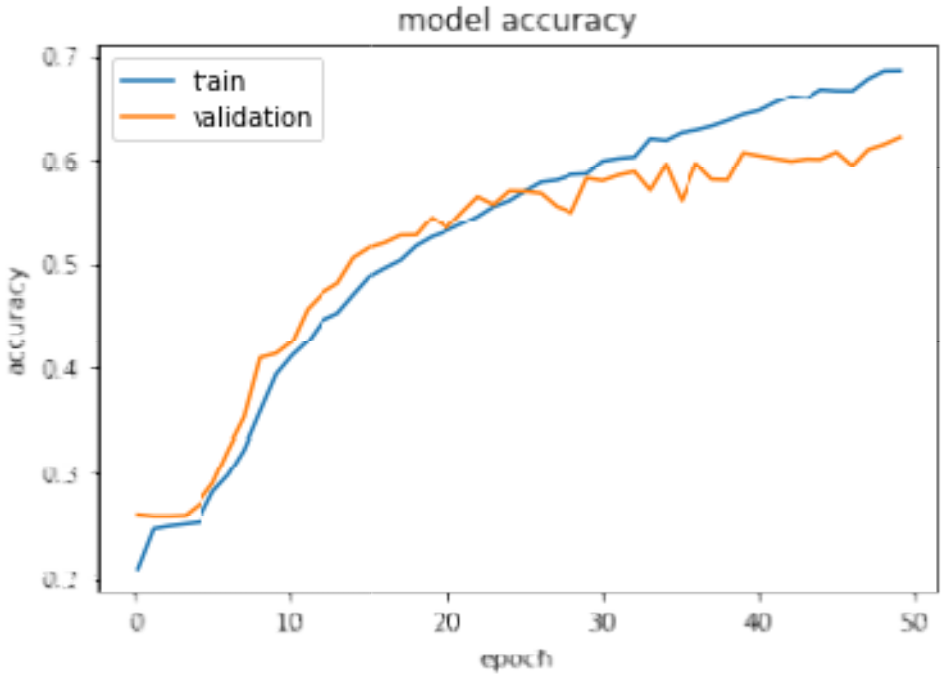

Fig 21 – Model accuracy plot

The Accuracy of training data and the validation data is plotted over the increasing numbers of epochs. From the plot it can be understood that accuracy for training keeps on increasing up to 50 epochs but for validation accuracy does not increases much after 35-40 epochs.

Thus, it can be concluded from the plot that after 35-40 epochs the model as issue of over-fitting that is a drawback that must be dealt in further research. Further research would focus on minimizing the absolute difference between the training accuracy and validation accuracy to be in range within 1% which would result in much efficient system making better predications of the facial emotions.

7.2MODELLOSS:

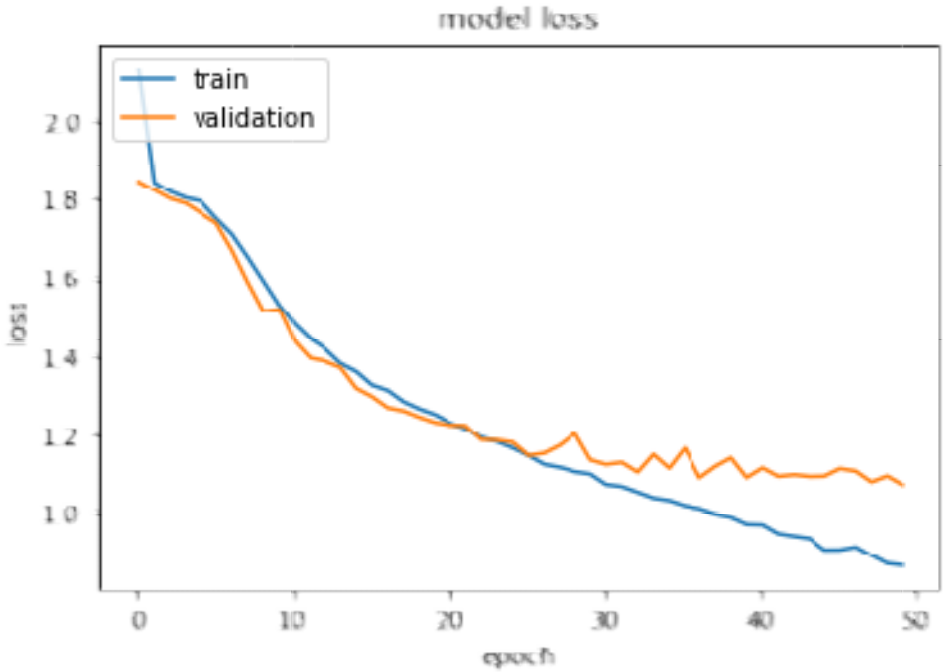

Fig 22 – Model lossplot

Lossoftrainingdatasetandvalidationdatasetisplottedoverincreasingnumbersofepochs.X-axis on the graph represents the epoch and Y-axis represents the data loss caused due to the model. This plot shows that loss is reduced to below 1 for the training dataset at 50 epochs, but lossof validation is not much reduced after 40 epochs. The loss in the data model for validation can be caused because of the over-fitting nature of the model. The future research will be helpful in dealing with these drawbacks. The aims for the future research in this direction would be to minimize the loss in the data and reduce it to the range of 0.2 – 0.3.

7.3 DATASET COUNT:

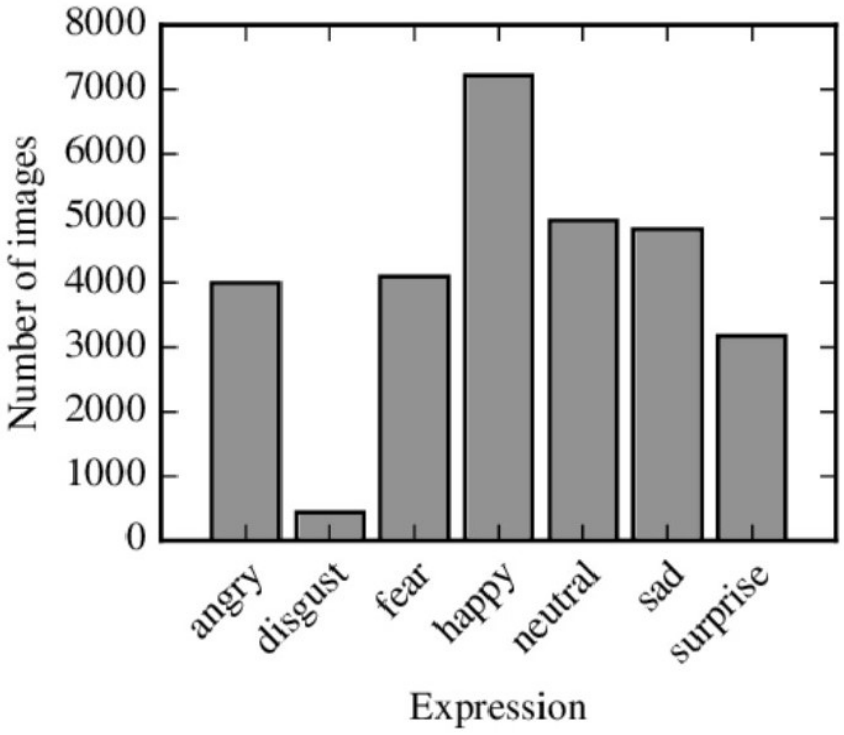

Fig 23 – Dataset Bar Plot for each expression

The above graph represents the dataset used for training our model. This graph represents the dataset available for each class of emotion i.e. (angry, disgust, fear, happy, neutral, sad, surprise) for training each emotion set into the model. For e.g.: we can see that we have 4000 datasets available for training angry emotion to the model as well we have around 5200 datasets available for neutral emotion and around 4800 datasets available for training sad emotion.

7.4 CONFUSION MATRIX:

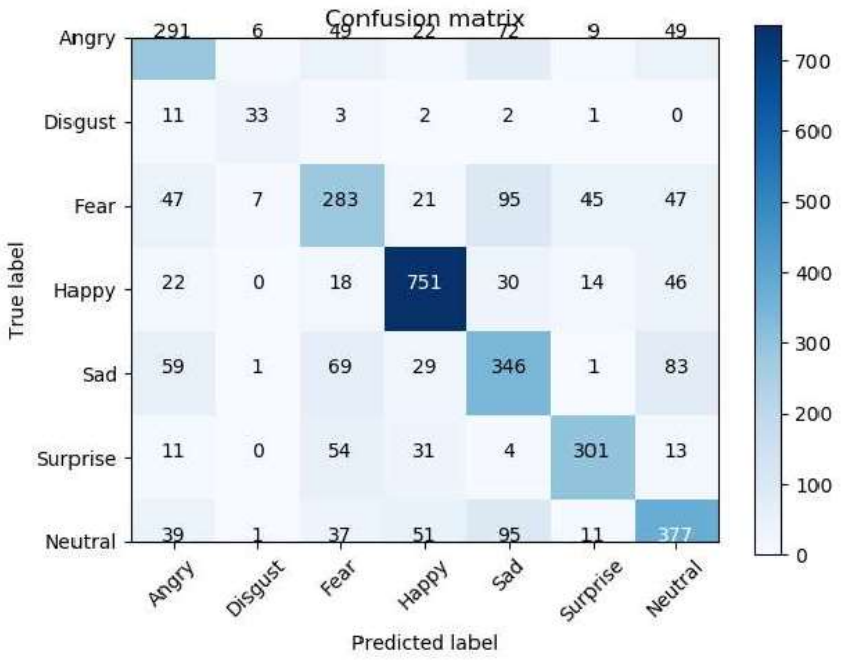

Fig 24 – Confusion Matrix

The Confusion matrix helps us to define the accuracy, performance and the effectiveness of the model. This calculates the probability of the predicted value against the actual value and hence letting us know the performance of the model designed. The confusion matrix comprises of 4 basic terms TP, TN, FP, FN. i.e. (True Positive, True Negative, False Positive, False Negative). In this diagram we are having 7 classes hence the 7*7 matrix. Here in the diagram we can see that actual angry label is having performance accuracy of 291 when compared to (predicted angry) and other (predicted label) has less performance accuracy against true angry label. Hence, we can see that model has a decent performance but there is still a lot of scope for further improvement.

7.5 CNN MODEL OUTPUT:

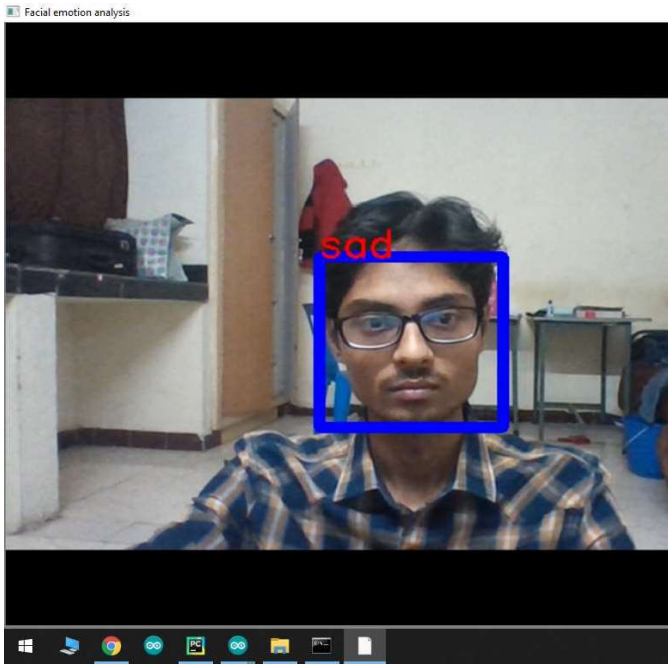

Fig 25– Facial Emotion Detection output

Above figure depicts the output of the emotion predicted by the CNN model. This predicted emotion is then sent over to the cloud (Adafruit) where all the output result suchas temperature, humidity value etcis displayed over the Adafruit Dashboard.

7.6 : ADAFRUIT DASHBOARD:

Fig 26 – Music to be played as per conditions

The above figure is a snapshot of the Adafruit Dashboard which provides the overview of the data we are receiving from the sensor and CNN model and after that the Music Selector algorithm use these data to predict the music to be played and then the music is played. The above Dashboard is dynamic and keeps on receiving the value every 2 second so that throttling limit is not reached. The Dashboard also displays the music being played based on the criteria provided.

7.7 EXPERIMENTAL RESULTS (TABULATED RESULTS)

Epochs	Train_Result(% value)	Test_Result(% value)
25	56.65	55.23
40	65.16	60.02
50 (Best)	71.21	66.23
60	72.23	62.23

Fig 27- Accuracy Chart

In the above table we can see that for the for different epochs we get varied accuracy but 50 epochs make the training data more accurate at almost 71%.

This depicts that the data is trained in different scenarios with different epoch sizes and the epochs are the reason for training the data set to predict the test results but as seen the trained data set varies to be accurate while increasing the number of epochs but it tends to be more accurate and stable at 50 epochs in analyzing the dataset.

As we recognize that while there are 50 epochs to train the data set it is more accurate in identifying at almost 66% which means that approximately 33% cannot be identified due to several factors.

CHAPTER 8

CONCLUSION

Thus, in conclusion human emotional states can jump vastly at times due to stress or any personal reasons causing unstable behavior leading to unpredictable nature which could be harmful for oneself. This in again affects their daily lives which it could ruin the momentum. The controller can study and predict human emotional states using facial detection by comparing to various emotions from a data set and combining the data with various environmental conditions thus understanding the setting in which the user or person is existing currently , then checks if the user is in an extreme condition and if so tries to ease the situation by using a music controller along with setting the perfect lightning conditions if needed. When thought in real life scenarios this could be an alternative solution to major problems like loneliness, stressful work lifeordepression. It instead of implementing on a full scale in a smart home-based system if compressed onto a smart handheld device along with haptic feedback the solution could get far more superior than ever imagined. Using these techniques, we are getting not the best of results. The main reason is as all pixels vertically concatenated to feed to the auto-encoder which might result in the loss of structural integrity of the image. In the future convolution filters as the encoder part and the encoder part and deconvolution units in place of normal hidden units can be used for better usage. The model has overfitting as one of its majordrawbacks.

CHAPTER 9
FUTUREENHANCEMENT

Over time new technologies emerge, and the market takes time to stabilize and adapt to these new technologies but with the current state of technology which was unexpected and is in a better than ever state advancements are only to be imagined not in a major way but more through more advanced methods These techniques, are getting the almost the best of results but the main reason is due to the pixels vertically concatenated to feed to the auto-encoder which might result in the loss of structural integrity of the image. In the future convolution filters as the encoder part and the encoder part and deconvolution units in place of normal hidden units can be used for better usage. The model has overfitting as one of its major drawbacks which can be reduced in future iterations through better implementation of technology. AI assistant support to the whole module would be a huge leap and would change the whole way data is read on the user and prediction of the user's emotional state would vastlyimprove.

Along with these improvements and advancements this module can be built as full fledge system not just a prototype and can be integrated to other home automation system which could proper their use to do a better job in real world scenarios without breaking users privacy concerns.

CHAPTER 10

REFERENCES

[1] IOT Based Monitoring and Control System for Home Automation. International Journal ofResearch.

[2] The Internet of Things: A Survey. Computer Networks. 2787-2805.10.1016/j.comnet.2010.05.010.

[3] Discovering facial expressions for states of amused, persuaded, informed, sentimental and inspired.

[4] Reliable Emotion Recognition System Based on Dynamic Adaptive Fusion of Forehead Biopotentials and Physiological Signals. Computer methods and programs in biomedicine. 122.10.1016/j.cmpb.2015.07.006.

[5] I. Michael Revina& W.R. Sam Emmanuel (2018). A Survey on Human Face Emotion Recognition Techniques.

[6] Multi-features descriptors for Human Activity Tracking and Recognition in Indoor-Outdoor Environments

[7] Simple convolutional neural network on image classification

[8] Understanding of a convolutional neural network

[9] Face expression recognition: A brief overview of the last decade. SACI 2013 - 8th IEEE International Symposium on Applied Computational Intelligence and Informatics, Proceedings

[10] Integration of Cloud computing and Internet of Things: A survey. Future Generation Computer Systems

[11] Machine- to-Machine Communications: Architectures, Technology, Standards, andApplications.

[12] Integration of Cloud Computing with Internet of Things: Challenges and Open Issues

[13] Prototype-based modeling for facial expression analysis IEEE Trans.

[14] Human Centric Facial Expression Recognition. Proc. Br. HCI pp.1–12.

[15] Facial expression n recognition using deep neural networks IEEE Int. Conf. Imaging Syst. Tech.

[16] Rashid, Convolutional neural networks-based method for improving facial expression recognition Intel.

[17] Facial Expression Recognition Based on a Deep Convolutional-Neural-Network Structure. IEEE 15th Int. Conf. Soft. Eng. Res. Manag. Appl.123–128.

[18] Zhang et al., 2017 C. Zhang, P. Wang, K. Chen, Identity-aware convolutional neural networksforfacialexpressionrecognitionJ.Syst.Eng.Electron.

[19] Door Security System for Home Monitoring Based on ESP32 Procedia Computer Science.

[20] A Low-Cost Home Automation System using Wi-Fi Based Wireless Sensor Network Incorporating Internet ofThings(IoT).

APPENDIX

FACIAL EMOTION DETECTION (CNN Code):

```
# -*- coding: utf-8 -*-

"""FacialEmotionDetection.ipynb


from google.colab import drive

drive.mount('/content/drive')


cd 'drive/My Drive/FinalYearProject'


from keras.callbacks import TensorBoard

tensorboard = TensorBoard(log_dir='/content/logs',histogram_freq=0,write_images=False)

keras_callbacks = [tensorboard]



import pandas as pd

import numpy as np
```

```
import warnings

import tensorflowastf

warnings.filterwarnings("ignore")


data = pd.read_csv('./fer2013.csv')


width, height = 48, 48


datapoints = data['pixels'].tolist()


#getting features for training

X = []

for xseq in datapoints:

    xx = [int(xp) for xp in xseq.split(' ')]

    xx = np.asarray(xx).reshape(width, height)

    X.append(xx.astype('float32'))


X = np.asarray(X)

X = np.expand_dims(X, -1)


#getting labels for training

y = pd.get_dummies(data['emotion']).as_matrix()


#storing them using numpy
```

```
np.save('fdataX', X)

np.save('flabels', y)


print("Preprocessing Done")

print("Number of Features: "+str(len(X[0])))

print("Number of Labels: "+ str(len(y[0])))

print("Number of examples in dataset:"+str(len(X)))

print("X,y stored in fdataX.npy and flabels.npy respectively")

!nvidia-smi

import sys, os

import pandas as pd

import numpy as np

from sklearn.model_selection import train_test_split

from keras.models import Sequential

from keras.layers import Dense, Dropout, Activation, Flatten

from keras.layers import Conv2D, MaxPooling2D, BatchNormalization

from keras.losses import categorical_crossentropy

from keras.optimizers import Adam

from keras.regularizers import l2


num_features = 64

num_labels = 7
```

```
batch_size = 64

epochs = 50

width, height = 48, 48

x = np.load('./fdataX.npy')

y = np.load('./flabels.npy')


x -= np.mean(x, axis=0)

x /= np.std(x, axis=0)

#for xx in range(10):

#    plt.figure(xx)

#    plt.imshow(x[xx].reshape((48, 48)), interpolation='none', cmap='gray')

#plt.show()

#splitting into training, validation and testing data

X_train, X_test, y_train, y_test = train_test_split(x, y, test_size=0.1, random_state=42)

X_train, X_valid, y_train, y_valid = train_test_split(X_train, y_train, test_size=0.1,
random_state=41)
#saving the test samples to be used later

np.save('modXtest', X_test)

np.save('modytest', y_test)

#desinging the CNN model = Sequential()

model.add(Conv2D(num_features, kernel_size=(3, 3), activation='relu', input_shape=(width,
height, 1), data_format='channels_last', kernel_regularizer=l2(0.01)))

model.add(Conv2D(num_features, kernel_size=(3, 3), activation='relu', padding='same'))

model.add(BatchNormalization())

model.add(MaxPooling2D(pool_size=(2, 2), strides=(2, 2)))
```

```
model.add(Dropout(0.5))

model.add(Conv2D(2*num_features, kernel_size=(3, 3), activation='relu', padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(2*num_features, kernel_size=(3, 3), activation='relu', padding='same'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2), strides=(2, 2)))
model.add(Dropout(0.5))

model.add(Conv2D(2*2*num_features, kernel_size=(3, 3), activation='relu', padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(2*2*num_features, kernel_size=(3, 3), activation='relu', padding='same'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2), strides=(2, 2)))
```

```
model.add(Dropout(0.5))

model.add(Conv2D(2*2*2*num_features, kernel_size=(3, 3), activation='relu', padding='same'))
model.add(BatchNormalization())
model.add(Conv2D(2*2*2*num_features, kernel_size=(3, 3), activation='relu', padding='same'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2), strides=(2, 2)))
model.add(Dropout(0.5))

model.add(Flatten())

model.add(Dense(2*2*2*num_features, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(2*2*num_features, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(2*num_features, activation='relu'))
model.add(Dropout(0.5))

model.add(Dense(num_labels, activation='softmax'))

model.summary()

tf.test.is_gpu_available()
```

```
#Compliling the model with adamoptimixer and categorical crossentropy loss
model.compile(loss=categorical_crossentropy,
            optimizer=Adam(lr=0.001, beta_1=0.995, beta_2=0.995, epsilon=1e-7),
            metrics=['accuracy'])

#training the model
history = model.fit(np.array(X_train), np.array(y_train),
          batch_size=batch_size,
          epochs=epochs,
          verbose=1,
          validation_data=(np.array(X_valid), np.array(y_valid)),
          shuffle=True , callbacks = keras_callbacks)

#saving the model to be used later
fer_json =model.to_json()
with open("fer.json", "w") as json_file:
   json_file.write(fer_json)
model.save_weights("fer.h5")
print("Saved model to disk")

from keras.models import model_from_json
model = model_from_json(open("fer.json", "r").read())
#load weights
model.load_weights('fer.h5')
```

```
# Commented out IPython magic to ensure Python compatibility.
import matplotlib.pyplot as plt
# %matplotlib inline

# accuracy plot
plt.plot(history.history['acc'])
plt.plot(history.history['val_acc'])
plt.title('model accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'validation'], loc='upper left')
plt.show()

#loss plot
plt.plot(history.history['loss'])
plt.plot(history.history['val_loss'])
plt.title('model loss')
plt.ylabel('loss')
plt.xlabel('epoch')
plt.legend(['train', 'validation'], loc='upper left')
plt.show()

dir(model)
```

```
from keras.utils import

plot_modelplot_model(model,

to_file='model.png')


classes = ['Angry','Disgust','Fear','Happy','Sad','Surprise','Neutral']


import seaborn as sns

y_pred = model.predict_classes()


con_mat = tf.math.confusion_matrix(labels=y_true, predictions=y_pred).numpy()

con_mat_norm = np.around(con_mat.astype('float') / con_mat.sum(axis=1)[:, np.newaxis],
decimals=2)


con_mat_df = pd.DataFrame(con_mat_norm,

                index = classes,

                columns = classes)

figure = plt.figure(figsize=(8, 8))

sns.heatmap(con_mat_df, annot=True,cmap=plt.cm.Blues)

plt.tight_layout()

plt.ylabel('True label')

plt.xlabel('Predicted label')

plt.show()


import matplotlib.pyplot as plt
```

```
def plot_confusion_matrix(y_true, y_pred, classes,
                          normalize=False,
                          title=None,
                          cmap=plt.cm.Blues):
    """
    This function prints and plots the confusion matrix.
    Normalization can be applied by setting `normalize=True`.
    """
    if not title:
        if normalize:
            title = 'Normalized confusion matrix'
        else:
            title = 'Confusion matrix, without normalization'


    # Compute confusion matrix
    cm = confusion_matrix(y_true, y_pred)
    # Only use the labels that appear in the data
    classes = classes
    if normalize:
        cm = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
        #print("Normalized confusion matrix")
    #else:
        #print('Confusion matrix, without normalization')
```

```
#print(cm)


fig, ax = plt.subplots(figsize=(12,6))
im = ax.imshow(cm, interpolation='nearest', cmap=cmap)
ax.figure.colorbar(im, ax=ax)
# We want to show all ticks...
ax.set(xticks=np.arange(cm.shape[1]),
       yticks=np.arange(cm.shape[0]),
       # ... and label them with the respective list entries
       xticklabels=classes,  yticklabels=classes,
       title=title,
       ylabel='True label',
       xlabel='Predicted label')


# Rotate the tick labels and set their alignment.
plt.setp(ax.get_xticklabels(), rotation=45, ha="right",
         rotation_mode="anchor")


# Loop over data dimensions and create text annotations.
fmt = '.2f' if normalize else 'd'
thresh = cm.max() / 2.
for i in range(cm.shape[0]):
    for j inrange(cm.shape[1]):
```

```
        ax.text(j, i, format(cm[i, j], fmt),
                ha="center", va="center",
                color="white" if cm[i, j] > thresh else "black")
    fig.tight_layout()
    return ax


plot_confusion_matrix(y_true, y_pred, classes,normalize=False,title=None,cmap=plt.cm.Blues)
```

Code for Data transmission to Cloud Storage:

```
import os
import cv2
import numpy as np
from keras.models import model_from_json
from keras.preprocessing import image
import msvcrt
import time
current_time = time.time()
#sendind data to cloud
from Adafruit_IO import Client
ADAFRUIT_IO_USERNAME  ="p_2897"
ADAFRUIT_IO_KEY = "edc2fc2fa04642c1a9d9d337bbcd8ba3"
from Adafruit_IO import Client,Feed,Data
aio = Client(ADAFRUIT_IO_USERNAME, ADAFRUIT_IO_KEY)

#load model
model = model_from_json(open("fer.json", "r").read())
#load weights
model.load_weights('fer.h5')
```

```
face_haar_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')

cap=cv2.VideoCapture(0)

whileTrue:
    ret,test_img=cap.read()# captures frame and returns boolean value and captured image
    if notret:
        continue
    gray_img= cv2.cvtColor(test_img, cv2.COLOR_BGR2GRAY)

    faces_detected = face_haar_cascade.detectMultiScale(gray_img, 1.32, 5)

    for (x,y,w,h) in faces_detected:
        cv2.rectangle(test_img,(x,y),(x+w,y+h),(255,0,0),thickness=7)
        roi_gray=gray_img[y:y+w,x:x+h]#cropping region of interest i.e. face area from image
        roi_gray=cv2.resize(roi_gray,(48,48))

        import matplotlib.pyplot as
        pltplt.imshow(roi_gray)

        img_pixels = image.img_to_array(roi_gray)
        img_pixels = np.expand_dims(img_pixels, axis = 0)
        img_pixels /= 255

        predictions = model.predict(img_pixels)

        #find max indexed array
        max_index = np.argmax(predictions[0])

        emotions = ('angry', 'disgust', 'fear', 'happy', 'sad', 'surprise', 'neutral')
        predicted_emotion = emotions[max_index]

        if time.time()- current_time>= 2:
```

```
            aio.send('major-project.facial-emotion', predicted_emotion)
            current_time = time.time()

        cv2.putText(test_img, predicted_emotion, (int(x), int(y)),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0,0,255), 2)

    resized_img = cv2.resize(test_img, (1000, 700))
    cv2.imshow('Facial emotion analysis ',resized_img)

    if cv2.waitKey(10) == ord('q'):#wait until 'q' key is pressed
        break

    if msvcrt.kbhit() and msvcrt.getch() == chr(27):
        break

cap.release()
cv2.destroyAllWindows
```

Arduino Code for ESP32:

```
#include<WiFi.h>

#include <Adafruit_Sensor.h>

#include<DHT.h>

#include  <DHT_U.h>

#include"AdafruitIO_WiFi.h"


#defineDHTTYPE      DHT11     // DHT11

#define DHTPIN23

#define WIFI_SSID "PiyushRaj"

#define WIFI_PASS "passit123"

#define IO_USERNAME "p_2897"
```

```
#defineIO_KEY       "edc2fc2fa04642c1a9d9d337bbcd8ba3"

AdafruitIO_WiFiio(IO_USERNAME, IO_KEY, WIFI_SSID, WIFI_PASS);
DHT_Unifieddht(DHTPIN, DHTTYPE);

uint32_t delayMS;


AdafruitIO_Feed *_humid = io.feed("major-project.humidity");
AdafruitIO_Feed *_temp = io.feed("major-project.temperature");
AdafruitIO_Feed *_musicType = io.feed("major-project.typeofmusic");
AdafruitIO_Feed *emotion = io.feed("major-project.facial-emotion");

void setup() {
  Serial.begin(115200);
  Serial.println("***************Intializing Devices***************\n");
  // Initialize device.
  dht.begin();
  Serial.println(F("DHTxx Unified Sensor Example"));
  // Print temperature sensor details.
  sensor_t sensor;
  dht.temperature().getSensor(&sensor);
  Serial.println(F("------------------------------------"));
  // Print humidity sensor details.
```

```
  dht.humidity().getSensor(&sensor);

  WiFi.begin(WIFI_SSID, WIFI_PASS);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.println("***************Connecting to WiFi..***************");
  }
  Serial.println("***************Connected to the WiFi network***************");

  Serial.print("***************Connecting to Adafruit IO***************");
  // connect to io.adafruit.com
  io.connect();
  // wait for a connection
  while(io.status() <AIO_CONNECTED) {
    Serial.print(".");
    delay(500);
  }
  // we are connected
  Serial.println();
  Serial.println(io.statusText());
  delayMS = sensor.min_delay / 1000;
}
```

```
void loop() {
  // Delay between measurements.
  delay(delayMS);
  // Get temperature event and print its value.
  sensors_event_t event;
  dht.temperature().getEvent(&event);
  if (isnan(event.temperature)) {
    Serial.println(F("Error reading temperature!"));
  }
  else {
    Serial.print(F("Temperature: "));
    Serial.print(event.temperature);
    Serial.println(F("°C"));
    _temp->save(event.temperature);
  }
  // Get humidity event and print its value.
  dht.humidity().getEvent(&event);
  if (isnan(event.relative_humidity)) {
    Serial.println(F("Error reading humidity!"));
  }
  else {
    Serial.print(F("Humidity:"));
```

```
    Serial.print(event.relative_humidity);
    Serial.println(F("%"));
    _humid->save(event.relative_humidity);
  }
// String _emotion = (String)emotion->get();
// _musicType->save(WhatToDo(event.temperature,event.relative_humidity,_emotion));
  delay(3000);
}

String WhatToDo(float temp,floathumid,String emotion){
  if(temp >=15 && temp <= 25 &&humid >=85)
    return "Rainy";
  else if(emotion == "Angry")
    return "Sufi";
  else if(emotion == "Disgust")
    return "Happy";
  else if(emotion == "Fear")
    return "Motivational";
  else if(emotion == "Happy")
    return"Party";
  else if(emotion == "Surprise")
    return"Retro";
  else if(emotion == "Neutral")
    return "Random";
```

```
  else if(emotion == "Sad")
    return "Breakup";
}
```

PAPER PUBLICATION STATUS

"Paper Accepted"

International Journal of Advanced Science and Technology
Vol. 29, No. 6, (2020), pp. 2739 - 2749

Enhanced Smart Music Controller by Applying CNN in IoT

Piyush Raj[1], M. Safa[2]*, Nitish Kumar[3], P J Sahrudh[4], Shivaditya Singh[5]

[1,3,4,5] *Final Year UG Students,* [2]*Assistant Professor*
Department of Information Technology
SRM Institute of Science and Technology, Chennai-603203
Email : piyushrajkhg@gmail.com , safam@srmist.edu.in

Abstract

It is Important to implement an alternative approach to the smart home system and understand the human using the emotional states along with environmental parameters ,just using the facial recognition increases the error chances, instead it also relies on the environmental conditions to deduct the subjects' emotional states more accurately. Smart Systems are evolving in every sense both in how they are made and on what technologies they are made through. Internet of Things is one of the major technologies that play a part in creating a complicated convoluted interconnected system which do not require the human intervention to work and send data to other inter connected devices. The technologies including Facial Recognition, Cloud Computing along with the use of external sensors can be interconnected to produce a smart system which would have failed if not the processing power of the Cloud had not been used. Facial recognition is achieved through the use of a class of deep neural network known as Convolutional Neural Network (CNN).

Keywords: *CNN (Convolutional Neural Network), FER (Facial Expression Recognition), IoT (internet Of Things), LDR (Light Dependent Resistor), DHT*

PLAGIARISM REPORT

Report_TeamByte_Optimizer.pdf

ORIGINALITY REPORT

7%	5%	4%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Priya Gupta, Nidhi Saxena, Meetika Sharma, Jagriti Tripathi. "Deep Neural Network for Human Face Recognition", International Journal of Engineering and Manufacturing, 2018 Publication	1%
2	thinkmind.org Internet Source	1%
3	Leandro Y. Mano, Bruno S. Faiçal, Luis H.V. Nakamura, Pedro H. Gomes et al. "Exploiting IoT technologies for enhancing Health Smart Homes through patient identification and emotion recognition", Computer Communications, 2016 Publication	1%
4	www.ijrte.org Internet Source	1%
5	medium.com Internet Source	1%
6	Submitted to Colorado Technical University Online	1%