DESIGN AND IMPLEMENTATION OF FACIAL EMOTIVE RECOGNITION SYSTEM USING MACHIINE LEARNING ALGORITHMS

A Thesis

Submitted in partial fulfillment of the requirements for the award of the degree of

DOCTOR OF PHILOSOPHY

in

Electronics and Electrical Engineering

By

Navjot Rathour

(41400728)

Supervised By

Dr. Anita Gehlot
Associate Professor

Co-Supervised by

Dr. Rajesh Singh
Professor

LOVELY PROFESSIONAL UNIVERSITY
PUNJAB
2020

DECLARATION

This thesis is an account of research undertaken between August 2015 and May 2020 at The Department of Electronics and Communication Engineering, Lovely Professional University, Phagwara, India.

Except where acknowledgement is the customary manner, the material presented in this thesis is, to the best of my knowledge, original and has not been submitted in whole or part of degree in any university.

Navjot Rathour
Registration no.41400728
Department of Electronics and Communication Engineering
Lovely Professional University, Phagwara, India

CERTIFICATE

This is to certify that the declaration statement made by the student is correct to best of my knowledge and belief. She has submitted the Ph.D. thesis “Design and Implementation of Facial Emotive Recognition System Using Machine Learning Algorithms” under my guidance and supervision. The present work is the result of her original investigation, effort and study. No part of the work has ever been submitted for any other degree at any University. The Ph.D. thesis is fit for the submission and fulfillment of the condition for the award of Ph.D. degree in Electronics and Communication Engineering from Lovely Professional University, Phagwara.

Dr. Anita Gehlot
Associate Professor
School of Electronics and Electrical Engineering
Lovely Professional University
Phagwara, India

Dr. Rajesh Singh
Professor
School of Electronics and Electrical Engineering
Lovely Professional University
Phagwara, India

ABSTRACT

The face is the index of mind, and facial expressions play an important role in understanding human emotions. A machine with Emotional-intelligence is beneficial from the Human-Computer interaction point of view and equally crucial in a massive set of applications, few of them are like anomalous event detection, interactive entertainment, ailment detection, customer service and satisfaction, personal trainer, health scrutinizing system, mood detector and many more. As it was found in 1872 by Darwin's research that facial expressions are the most common and straightforward way via which Humans can interact with each other and communicate verbally. So, the research has also proved that facial expressions are most important for understanding emotions.

The right amount of research work has already been carried out by the research community that provides a huge list of methods for understanding the facial emotions of Human beings in a better way. Implementation of a facial emotion recognition system on a laptop, desktop, or any other firm and powerful system is smooth but not practical. As such heavy and bulky systems can't be carried along all the time as they are not portable and handy. Moreover, one has to spend a lot as it needs a lot of wired connection for camera installation and supporting circuitry like DVR, Modem, and Monitor. The in-depth literature review concluded that much research has yet unexplored, and embedded devices play an essential role in that. With the advancement of technology, the embedded systems have become compact and powerful enough with quite a significant amount of processing speed and capability and that too at a very affordable and cheap cost.

Moreover, buzz words like IoT and edge computing have already attracted and revolutionized the research community. So, this mainly acted as the base of the research work. The main objective behind this work is to develop a system at a low cost that can recognize facial expressions of Human beings at any place and at any

point in time. The system needs to be capable enough to give the solution in real-time by understanding the Human facial expression instantly.

Working with Embedded devices that are new in the market for which very little research work is available in the market is a challenging job. On the other hand, it is essential when it comes to portable, low cost, and robust systems. Moreover, all these features need to implement in real-time so that a quick and instant solution of the problem should be available, which is directly related to human facial emotion. So the devices like Raspberry-pi, which costs only a mere $35, is a single-board computer. This board is available in the updated versions with quite attractive features in the market now—Raspberry-Pi 4, which comes with a 64-bit 1.5GHz processor and 1-4GB of RAM. Moreover, IoT and Edge computing have really accelerated this research work towards the achievement of the main objective.

The research community already knows that when it comes to computer vision algorithms, they are computationally quite expensive. But with advancements, we can get compiled packages of the libraries like OpenCV, Scikit-learn, and many more. So it is easy to work with Deep Learning, Machine learning, and computer vision now with tall these packages. Another important thing that has identified in the latest research is the latest resurgence in deep learning has created additional interest in an embedded device, such as the Raspberry Pi. Deep learning algorithms are super powerful, demonstrating unprecedented performance in tasks such as image classification, object detection, and instance segmentation. The main issue that lies behind the Deep learning algorithms is its vast and incredible computational expense. But as already discussed that the way computer vision libraries are making the things possible, similarly it applies to the Deep learning. Libraries such as Tensor Flow Lite enable to train the customized data set, and then after optimization, the same can be deployed on a resource-constrained device like Raspberry-Pi for faster inference. To run these super-powerful algorithms on a resource-constrained device like Raspberry-pi, some additional hardware is also required, and the same has been launched in the market in 2019 in the form of co-processors both by Google and Intel. Intel launched Movidius NCS, and Google

launched Coral USB Accelerator. These are the co-processors that can be connected with the Raspberry-pi in the form of a neural compute stick and have the capability to augment the CPU. So by combining it with the optimized libraries, a faster speed can be achieved as compared to a standalone CPU of Raspberry-Pi.

This thesis describes the design and implementation of facial emotion detection and recognition system that has been implemented in real-time using a small, powerful and resource-constrained device known as Raspberry-pi along with Intel Movidius Neural Compute Stick-II with the help of deep convolution neural networks. It starts with the selection of hardware and selection of co-processor devices, and after doing the literature review on various pre-processing, segmentation, and classification techniques. A literature review has also presented on deep convolution neural networks. So By taking into consideration all the above-discussed points, the researcher summarizes the following objectives.

- **RO1**-To develops a pre-processing algorithm to enhance the quality of emotive facial descriptions using a variety of image processing techniques.
- **RO2**-To develops an emotive facial feature extraction method using a machine learning algorithm.
- **RO3-**To implement computational intelligence techniques for classification and recognition of the extracted emotive facial features.
- **RO4-**To authenticates the performance of the projected method by comparing the performance with already existing techniques.

To select the database for facial emotion detection in real-time, comprehensive statistics of databases accessible in the field of facial expression recognition were generated and are summarized in this thesis. It then proposes a model for detection of emotions in real-time on a resource-constrained device, i.e., raspberry-pi along with a co-processor, i.e., Intel Movidius NCS2. The training of the deep network has done on the Google-CoLab with 12GB NVIDIA Tesla K80 GPU benefiting from a speedup of up to factor 10 over the computations on a CPU. The facial emotion detection test accuracy ranged from 56% to 73% using various models and the accuracy that has become 73% performed very well with the FER 2013 dataset in comparison to the state of art results mentioned as 64% maximum. This thesis presents a system that can be benefited by society in various applications, as mentioned in the very beginning. Moreover, the results achieved via this thesis are validated with the help of a physiological sensor, and proper setup has been done to carry the validation of the results under experimental conditions after calibration of the sensors. The experimental results discussed in this thesis are justified based on the experiment performed.

DEDICATION

“All action is of mind, and the mirror of the mind is the face, its index the eyes.”
MARCUSTULLIUS CICERO

This thesis is dedicated to my family, especially to my husband, Mr. Vineet Thakur, and my daughter Baby Aradhya.

ACKNOWLEDGMENTS

Firstly, I would like to thank God, who enabled me to use my wisdom and knowledge for this research work. Moreover, the patience that God has given me in all the hard times during this period is unforgettable.

I would like to express my sincere thanks to my supervisors, Dr. Rajesh Singh and Dr. Anita Gehlot, for their endless support, guidance, and supervision for each stage of this research. I am truly fortunate and blessed enough to have the opportunity to work with such a person.

I am grateful to my Mother-in-law, Ms. Kusum Bala, and my Father-in-law Mr. Joginder Singh for their endless motivation and support. Their endless motivation and support helped me to complete my research work on time. Besides, I would also like to thank all my fellow friends and colleagues who always motivated me with their valuable suggestions.

I would like to take this opportunity to record my thanks to the entire family of Lovely Professional University, specifically the Division of Innovation and Entrepreneurship, for all the support and encouragement. I am also very thankful to my father, Mr. M.G. Rathour, my mother, Ms. Sushma Rathour, and my brother Mr. Arun Kumar Rathour for their encouragement, moral support, and keeping me always active with their love, blessings and endless trust on me.

Date: June 29st,2020 Navjot Rathour

CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

3D	3 Dimensional
AAM	Active Appearance Model
AI	Artificial Intelligence
ANN	Artificial Neural Network
ASM	Active Shape Model
AU	Action unit
BBN	Bayesian Belief Network.
BEs	Basic Emotions
BP	Blood Pressure
CAE	Convolutionary Auto Encoder
CBF	Cloud Based Function
CEs	Compound Emotions
CK	Cohn Kenedy
CMU	Carnegie Melon University
CNN	Convolutional Neural Network
COPE	Classification of Pain Expressions database
CPU	Central Processing Unit
CSRC	Contributing Source
CV	Computer Vision
DAE	Deep Auto Encoder
DBN	Deep Belief Network
DCT	Discrete Cosine Transformation
DL	Deep Learning
DoG	Gaussian difference
DRML	Deep Region and Muli Label Learning
FACS	Facial Action Coding System
FAM	Fuzzy Art Map
FAS	Facial Action System

FEP	Facial Expression Parameters
FER	Facial Emotion Recognition
FLs	Facial Landmarks
FN	False Negative
FP	False Positive
GA	Geometric Algebra
GAN	Generative Opposing Network
GB	Giga Byte
GCN	Global Comparison Standardization
GPU	Graphics Processing Unit
HMM	Hidden Markov Model
HOG	Histogram of Oriented Gradients
IoT	Internet of Things
JAFFE	Japanese Female Facial Expression Database
KDE	The Karolinska Directed Emotional Faces
KNN	k-Nearest Neighbour
KPCA	Kernel of Principal Component Analysis
LBP	Local Binary Pattern
LBP-TOP	Local Binary Patterns on three orthogonal planes
LCD	Liquid Crystal Display
LDP	Local Directional Pattern
LFDA	Local Fisher Discriminatory Facial Discrimination
LFW	Labeled Faces in the Wild
LIPO	Lithium Ion Polymer
LSTM	Long-Short-Term Memory
LTP	Local Ternary Pattern
M	Mean
MAOP-DL	Multi-Angle Pattern-Based Deep Learning
MEs	Micro Expressions
ML	Maximum Likelihood
MLP	Multilayer Perceptron
MTCNN	Multi-Task CNN

NCS	Neural Compute Stick
NIR	Near-infrared
NN	Nearest Neighbour
NS	Nearest Subspace
Open VINO	Open Visual Inference and Neural Network Optimization
PCA	Principal Component Analysis
PDF	Probability Density Function
PDM	Point Distribution Model
P-MLBP	Polytypical Local Multi Block Binary Pattern
RAM	Random Access Memory
RBF	Radial Base Function
RBFNN	Radial Basis Function Neural Network
RBM	Restricted Boltzmann Machines
RELU	Rectified Linear Unit
RF	Radio Frequency
RNN	Recurrent Neural Network
SD	Standard Deviation
SDK	Software Development Kit
SGD	Stochastic Gradient Descent
SIFT	Scale-invariant Transform function
SPD	Symmetrical Positive Definite Multiplier
SRC	Sparse Representation-Based Classification
SVM	Support Vector Machine
SVM	Support Vector Machine
TAN	Tree-improved Naive Bayesian
TN	True Negative
TP	True Positive
TPU	Tensor Processing unit
USB	Universal Serial Bus
VIS	Visible Light
XOR	Exclusive OR

CHAPTER 1

INTRODUCTION

The chapter discusses the historical background of the Facial Emotion recognition System. To begin with, basic expressions of the human face along with the flow chart, discussed. Next section discusses the challenges that are generally faced during Facial Emotion Recognition. The motivation behind the research, along with specific research issues and objectives, are also discussed later in this chapter. Finally the research objectives, Methodology followed for research in the form of stages is represented diagrammatically. In the end, the thesis contributions with publications and thesis organization illustrated with the help of a flow chart.

1.1 Background

Computer animated agents and robotics bring new dimensions to the experience of human beings, which makes it essential for how machines influence our daily activities in society. Face-to-face contact is a process that operates in the order of milliseconds at a time scale. The level of insecurity at this stage is significant and makes it necessary for humans and the machines not to slow conceptual inference processes, but to rely on sensory-rich primitives in their perception.

Over recent years, omnipresent computers and digital processing came out to be an extremely vital part of everyday life. It also pushes advances in the creation of agent-controlled interfaces. As an essential element in human life, our feelings and influences also made us communicate and understand our intentions, emotions, and feelings affecting unconsciously our day to day activities like thinking, decision making, and interpersonal relations. Therefore, in the current technological era where life is so complicated, the detection of emotions automatically has become a current center of the AI field, since the importance of impacts in human life and daily functioning are well known. Emotion-intelligent machines not only benefit from Human-Computer Interact but also depend upon a variety of applications such as personal training, health scrutinizing systems, customer services, anomalous event detection, smart robots, and interactive computer entertainment. Darwin's research (1872) found that the most common way of Communicating between human beings is through facial expressions and verbal communication(Darwin, 1872). Most facial muscle activity and voice sound perform the most crucial role in the contact.

Mehrabian (1968) nevertheless indicated that the speaker's face look adds 55% to the spoken word's power, which is more than the verbal (7%) and auditory (38%) (Wiener & Mehrabian, 1968). Therefore, visual, emotional contact tends to be the most tangible form. That is the main reason behind the most commonly employed scheme for the calculation of human emotional condition. The first automated analyses of facial appearance recognition were performed by Suwa et al. (1978) in 1978 with the help of computers. That gradually improved since the 1990s (Sown, 1978).

The facial facades grouped mostly into several indispensable sentiments, as depicted in Figure 1.1 (e.g., optimistic, joy, rage, disgust, disappointment, anxiety, surprise), which Darwin (Darwin, 1872) described. Consequently, the appropriate teaching and evaluation resources on facial expressions can be made available in most existing systems where attempts have taken to understand such simple words, these specific feelings based on classifying them. Researchers also put much effort into calculating facial anatomy, defining the facial appearance and its activity, and even classifying facial looks. According to Tian et al. (Y.-l. Tian, Kanade, & Cohn, 2002, 2005; Y. Tian, Kanade, & Cohn, 2011), the general structure for facial glance perception is well established to be identical to that and provided in Figure 1.2. The method of general emotional recognition for the face consisted of facial development, facial identification, definition, and facial expression interpretation. The first step is, in particular, to focus on different attempts to get face area from pictures or videos and include all facets in the model of a reference with face markings, extract characteristics for a defined body language and categorize the terms. The second step is the use of facial appearance extraction and representation techniques. Affect recognition is essential to understand one's facial structure. It can be seen as generating excellent characteristics to define the presence, form, and motion of facial expressions well. More specifically, facial look applications designed to describe the structure of the face more accurately. Generally, they are fundamental properties, e.g., texture, form, and color.

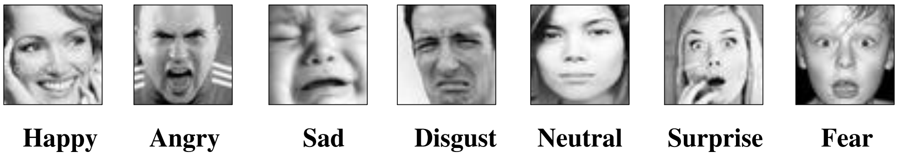


Figure 1.1: Seven specific feelings extensively used for interpretation of Expression of the face (M. Lyons et al., 1998)

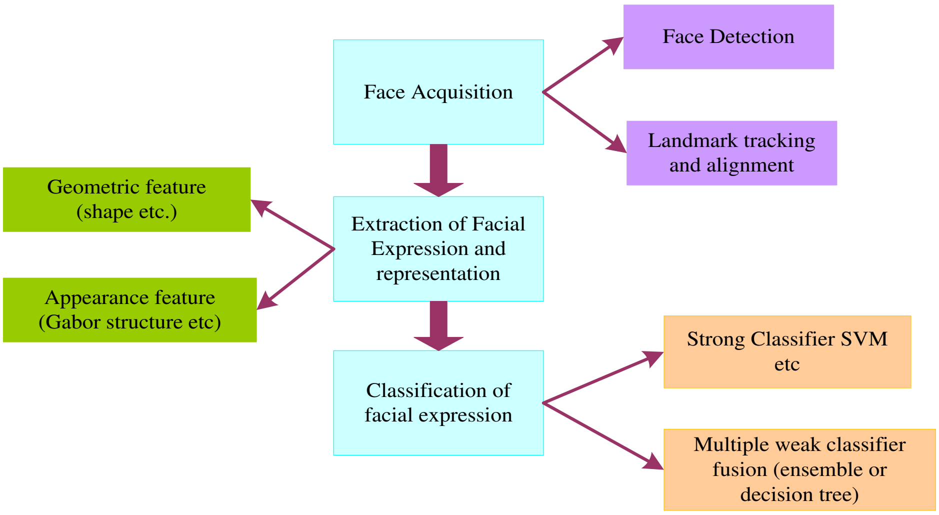

Face Detection
Face Acquisition
Landmark tracking and alignment
Geometric feature (shape etc.)
Extraction of Facial Expression and representation
Appearance feature (Gabor structure etc)
Strong Classifier SVM etc
Classification of facial expression
Multiple weak classifier fusion (ensemble or decision tree)

Figure 1.2: Flowchart of facial expression recognition.

They can be used for the reduction of variations in facial appearance within the class, while optimizing differences within groups, to the result of Fisher's rule. Recent advances in facial descriptors have shown that facial anatomy and presentation methods are two common strategies used to capture features of facial expressions of interest. They are used for the identification of facial expressions in the study of still photographs and complex image series. Nevertheless, forming applications with efficient computing expenses are not resilient to low resolution, whereas, for facial images, Gabor and LBP provide broad dimensions of function. Another downside to current facial appearance recognition techniques is that recent research fails to discern facial terminology when the images are captured in the environment, which is highly controlled. In reality, providing facial photos of good quality in the real-world environment isn't always straightforward.

Conversely, bad-quality face photos, probably at low resolution or weak optical security lighting, raise further problems for identification of Facial Expression. The sentiment acknowledgment in the wild challenge and laboratory investigated first the efficiency of methods of emotional recognition in the wild. This also revealed that a facial look recognition system would be able to see the facial Expression in less detail instantly and take into consideration the complete range of head movements. Some effort has been made over the years to develop architectural and design features for the detection of Facial Expression. The problem is far from being resolved, even though it has established several interesting applications by placing restrictions on the ecosystems.

1.2 Problem Identification

While different methods proposed for understanding facial expressions have produced excellent results, the research community still has different problems that need to be addressed.

1. In a single person, the most significant one is facial variation. Many variables can affect so that two photos of the same individual look completely different, such as light, face Expression, or occlusion.

2. Another thing which needs to be considered is the climate. Except in managed situations, face pictures have very different backgrounds, which can make it more challenging to identify the face. Many of the most popular solutions concentrate on treating the face alone, discard all the surroundings, to address this issue.

3. Smart conferencing, Video Conference and Visual Monitoring are real-world applications that include face-lift recognition that works well with videos with low resolution. There are many facial look recognition techniques, but a handful of these approaches work correctly on low-resolution images.

4. Implementation of such a program in real-time on lightweight and convenient computers such as Raspberry-Pi can be a real game turn. With the advent of edge computing devices, these complicated structures on Raspberry-Pi can be implemented and deployed on resource-constrained devices.

1.3 Motivation

The motivation behind the proposed work came from an incident that happened in my surrounding. A student committed suicide because of depression. At that moment, I felt that some system should be there that can perceive the emotions in real-time and can identify such problems. Moreover, in today’s scenario, we all are working in a competitive environment. Everyone is stepping forward with high aspirations, and expectations and failure at any stage generally lead to depression. Stress and anxiety are common problems nowadays. I, therefore, decided to start working on the system of recognition of human facial emotion and what research has already been done in the same field. A system that can identify expressions in real-time can be beneficial for this purpose. Moreover, such devices should be portable and handy enough to implement the system without the requirement of hefty wiring, bulky laptops, and time-consuming processing.

1.4 Research Issues and Objectives

The foremost ideas of this research are to address the issue related to emotion detection in real-time, efficient preprocessing algorithms, efficient method to extract features to identify a face and related emotions in real-time, and finally, provide a cost-effective solution to society by comparing it with existing systems and techniques.

In the present study researcher targets the following major components:

- **Pre-processing Algorithm:** It is responsible for the accurate extraction of human faces from a live video frame.
- **Feature Extraction:** It is responsible for various exact features of the face that can act as a base for exact classification.
- **Classification and Recognition:** It is responsible for the exact identification and related emotion of the person.

  To work on the above mentioned significant components, first, there is a need to identify the platform that is capable enough to perform the challenging task. This helps to implement the system in real-time with greater accuracy and lesser time. Moreover, it is easily deployable and cost-effective also. To select the platform, the following issues need to take into consideration.

  - How to utilize a resource-constrained device like Raspberry-Pi for such a task.
  - How to train the model to extract features.
  - How to deploy the pre-trained model to perform the task in real-time.
  - How to validate the performance of the system.

  After the selection of the platform, Datasets selection needs to identify carefully.

  - How to select an efficient dataset.

- How to train the selected dataset.
- Which deep-network should be used to increase the efficiency and deploy the pre-trained Network?
- Which edge device is compatible with a selected platform?

By taking into consideration all the above-discussed points, the researcher summarizes the following objectives.

- **RO1**-To develop a preprocessing algorithm to enhance the quality of emotive facial descriptions using a variety of image processing techniques.
- **RO2**-To develop an emotive facial feature extraction method using machine learning algorithm.
- **RO3-**To implement computational intelligence techniques for classification and recognition of the extracted emotive facial features.
- **RO4-**To authenticate the performance of projected method by comparing the performance with already existing techniques.

1.5 Research Methodology

The research is expected to develop an emotion recognition system and improve it in real-time. The proposed approach works efficiently on resource-constrained devices, i.e., Raspberry-pi, along with an edge computing device to increase the processing capability and implement the efficiency in real-time.

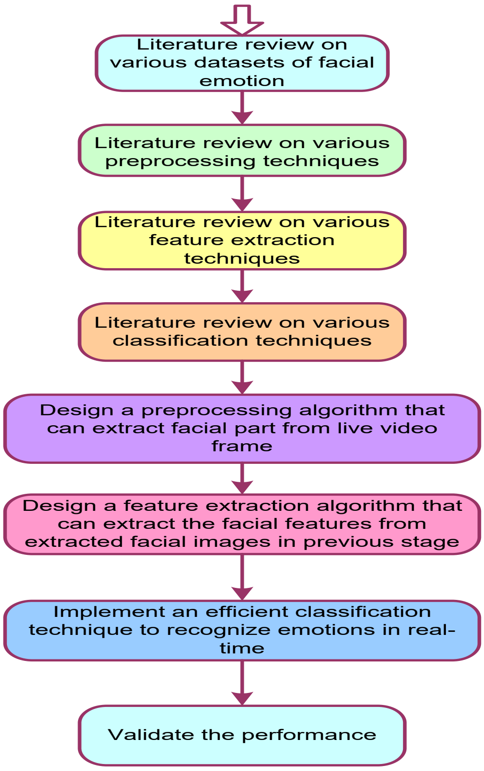

Literature review on various datasets of facial emotion
Literature review on various preprocessing techniques
Literature review on various feature extraction techniques
Literature review on various classification techniques
Design a preprocessing algorithm that can extract facial part from live video frame
Design a feature extraction algorithm that can extract the facial features from extracted facial images in previous stage
Implement an efficient classification technique to recognize emotions in real-time
Validate the performance

Figure 1.3: Flowchart of Research Methodology

1. To achieve the first Objective, a preprocessing algorithm has been designed that locates the facial part in the frame and detects it to crop the face and remove the background, which is of no use and unwanted, from the image. A HOG feature dependent approach has been used to detect the facial part and locate the facial part in real-time. Selection of descriptors done based on computational complexity and cost along with the performance analysis in real-time.

2. To achieve the second Objective, a facial feature extraction technique designed with an ensemble of regression to locate the landmarks on the cropped image. This algorithm is also merged with an affine transform to apply various image correction techniques on the facial image, i.e., extracted from a video frame and further fed to the input layer of the training model.

3. To achieve the third Objective, a deep network by open face used to extract embeddings and then classified using the SVM classifier.

4. To achieve the fourth Objective, comparative analyses have been done with recording the physiological sensors along-with the facial images and perform a comparative analysis along with the correlation of both as the complete system is novel in itself, so comparative analysis has been done with the existing techniques only.

1.6 Thesis Contribution

The present search addresses the defined research questions. The thesis contribution mentioned below:

- A literature survey of all the preprocessing, feature extraction and classification techniques is done in detail.
    1. A detailed investigation has done to study various preprocessing, feature extraction, and classification techniques.
    2. The arrangement of the mentioned techniques is made as per the common characteristics.
    3. The merger of preprocessing algorithms has to remove the background and implement the algorithm in real-time.
- A detailed literature survey has done to select the facial emotion datasets
    1. A detailed investigation has done to understand various facial emotion datasets available for research work.
    2. Datasets for training have been selected based on the classes, resolution, and no of samples available in each class.
- Selection of platform has done to develop a system which is capable of detecting human facial emotions in real-time with unique features as mentioned next
    1. The device needs to IoT enabled
    2. The device should be capable of identifying facial emotions in real-time
    3. The device needs to be cost-effective and low power

4. The device should be able to compute a sophisticated algorithm in real-time

1.7 Thesis Organization

The entire organization of the thesis and its chapter-wise arrangement depicted in the flow chart in Figure 1.4. Mainly Chapter 2 is related to the literature review of all the existing techniques of Preprocessing, Feature Extraction, and Classification. Chapter 3 is about Deep learning concepts and various architectures. Chapter 4 explains the details of Raspberry-Pi and its co-processors. Chapter 5 explains the detailed methodology and developed algorithms to implement a facial emotion recognition system in real-time. Chapters 6 and 7 show the detailed Software and Hardware Development of the system. Chapter 8 Shows the Results and related discussions, and finally, Chapter 9 explains the Conclusion and future scope.

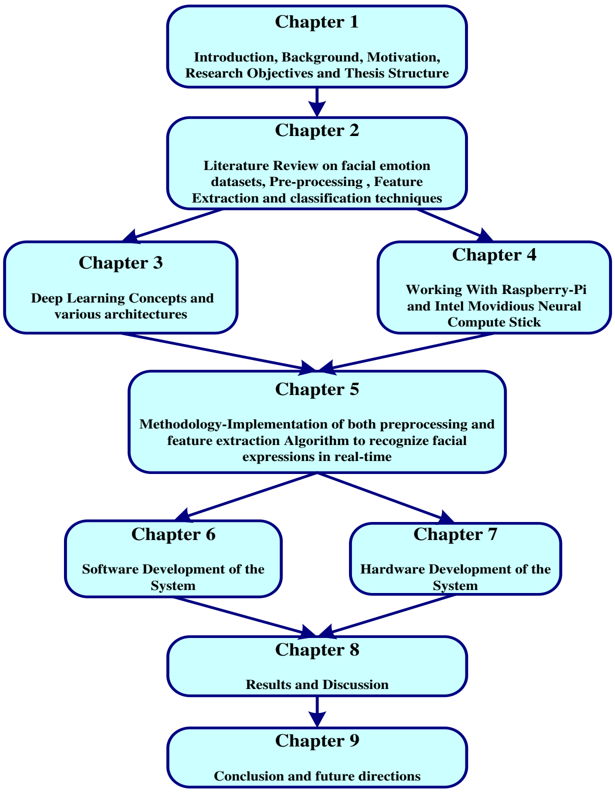

Chapter 1
Introduction, Background, Motivation, Research Objectives and Thesis Structure
Chapter 2
Literature Review on facial emotion datasets, Pre-processing , Feature Extraction and classification techniques
Chapter 3
Deep Learning Concepts and various architectures
Chapter 4
Working With Raspberry-Pi and Intel Movidious Neural Compute Stick
Chapter 5
Methodology-Implementation of both preprocessing and feature extraction Algorithm to recognize facial expressions in real-time
Chapter 6
Software Development of the System
Chapter 7
Hardware Development of the System
Chapter 8
Results and Discussion
Chapter 9
Conclusion and future directions

Figure 1.4: Chapter-wise thesis organization

Chapter 2 represents the detailed methodological survey on preprocessing, feature extraction, and classification techniques. The outcome of the findings satisfies Research objectives as follows:

First Objective Achieved

- Rathour, N., Singh, R., & Gehlot, A. (2020). Image and Video Capturing for Proper Hand Sanitation Surveillance in Hospitals Using Euphony—A Raspberry Pi and Arduino-Based Device. In *International Conference on Intelligent Computing and Smart Communication 2019* (pp. 1475-1486). Springer, Singapore. **(Scopus)**

The second Objective Achieved

- Navjot Rathour, Anita Gehlot, Rajesh Singh "*A Standalone Vision Device to Recognize Facial Landmarks and Smile in Real-Time Using Raspberry Pi and Sensor*." International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-2S8, August 2019 **(Scopus)**
- Navjot Rathour, Anita Gehlot, Rajesh Singh "*SPRUCE-An Intelligent Surveillance device for monitoring of dustbins using Image processing and Raspberry Pi*." International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-8 Issue-6, August 2019**(Scopus)**

Chapter 3 Explains the Deep learning basics and various Deep networks, and Chapter 4 Explains the resource-constrained device called Raspberry-Pi, Its key features and capability. It also explains how to speed the processing capability of this credit card-sized system with a combination of Co-processor. The outcomes of these chapters helped to select the Raspberry-Pi as the central processor to implement this system in real-time and to combine the system with co-processor to boost the processing capability.

The third Objective achieved

- Research Paper on the topic "*A Vision-based Architecture for Security System with Raspberry-Pi*." International Journal of

Signal and Imaging Systems Engineering.IJSISE-272692-Under Review**(Scopus)**

- Research Paper on the topic "*Real-Time Facial Emotion Recognition Framework for Employees in Private Organizations using Raspberry-Pi.*" Indonesian Journal of Electrical Engineering and Informatics-Under Review **(Scopus)**

Chapter 5 Explains the detailed methodology and also explains that the algorithm has designed to implement real-time facial emotion recognition.

- Research Paper on the topic "*Fast Face Recognition Framework using Raspberry-Pi and Intel Movidius Neural Compute Stick*." Accepted in the 3rd International Conference on Intelligent Circuits and Systems. **(Scopus)**

Chapter 6 and Chapter 7 explains the Software and Hardware Development of the system. These chapters extracted from the following:

The fourth Objective achieved

- Rathour, Navjot, et al. "Cloud technology-based vision mote with a scathing wearable device to monitor human emotions." Indian Patent No. 20191100778.28 Feb.2019

A detailed description of the thesis presented in this chapter. This chapter helps to understand the historical background and the detailed strategy that is used for deciding the research field, framing the objectives, achieving the objectives, and organize the thesis is discussed in this chapter.

CHAPTER 2

REVIEW OF LITERATURE

This chapter includes a detailed literature review for the FER system. All the preprocessing, Feature extraction and classification techniques mentioned in the literature are discussed in this chapter. This chapter gives a detailed review of the datasets available for the Facial Emotion recognition System as well.

2.1 Facial appearance recognition databases

One of the main aspects of human life is communications capacity. They use incredibly complex cues not only from a verb but also from a nonverbal point of view. The most crucial information sources include facial expressions. Throughout our daily lives, work on facial expressions has concentrated mostly on the emotional aspect until today and given the wide variety of facial expressions. However, most facial expressions repositories accessible for the research population often only include psychological terminology and missing the much-uncharted dimension. Getting sufficiently marked facial appearance data is a requirement for automated detection of facial appearance. The majority of the available study on perception of facial look focused on data sets of intentionally articulated feelings triggered by requesting contributors to execute a chain of effect related terms by facing camera (M. Lyons et al., 1998).

2.1.1 The Cohn-Kanade database

This study was composed of 100 students from universities aged between 18 and 30. 65% were black, 15% were African Americans, and 3% of them were Asian or Latino people. An experimenter instructed the participants to perform a sequence of Expression by showing faces in a count of 23, including single units of action and AU combinations, six of which concentrated on prototypic representations of anger, abominable behavior, fear, excitement, sorrow and shock (Takeo Kanade, Cohn, & Tian, 2000).

2.1.2 The extended Cohn-Kanade database

This variant (CK+) is the Cohn-Kanade edition subset. This section has also been extended to 593 sequences of 7 phrases, consisting of 123 subjects (an additional 107 series, 26 artifacts, and expressions of contempt(Lucey, 2010).

2.1.3 NIR&VIS database

The collection comprised of 80 topics aged between 23 and 58. 73.8% of the samples were men, recorded using two imaging technologies, The first one is Near-infrared (NIR) technology, and the other one is Visible Light (VIS) technology. It comprises six basic emotions (i.e., rage, disgust, terror, happiness, sorrow, and surprise).All the images considered under the conditions of natural, low, and dark illumination (S. Li, Yi, Lei, & Liao, 2013).

2.1.4 Infant Classification of Pain Expressions database

The baby pain signals database (COPE) list features 204 face photographs of 26 neonates and includes five of those neonatal gestures, 67 being rest, 18 weeping, 23 air pressures, 36 discomforts, and 60 pains. At baseline rest, images of the children took when enduring many noxious stimuli: physiological disruption, an air sensation on the ears, rubbing on the heel's exterior lateral surface, and pain from a heel handle (Brahnam, Chuang, Sexton, & Shih, 2007).

2.1.5 BU-3DFE database

The study of BU-3DFE comprises 100 subjects (56 percent female and 44 percent male) between the ages of 18 and 70. Seven movements have done before the 3D face scanner. In every six prototype expressions (happiness, sadness, fear, anger, surprise, and pain), Except neutral Expression, there are four degrees of sensitivity. There are, therefore, 25 3D instant models in terms of voice for each subject, which means in total a count of 2,500 3D facial voice models are available in the corresponding database image of the facial texture, captured in two perspectives, are connected to each face shape pattern. As a result, this set includes 2500 images with two texture views and 2,500 geometrically formed versions (L. Yin, Wei, Sun, Wang, & Rosato, 2006)(L. Yin, Wei, Sun, Wang, & Rosato, 2006).

2.1.6 Multi-PIE database

The facial database CMU Multi-PIE includes photographs from 337 respondents. Respondents were mostly male (70 percent). Sixty percent of subjects were White Americans, 35 percent African Americans, and 3 percent Asians. The participants were 28 years of age. Information was recorded over six months during four sessions. The subjects were taught in each course to show different facial expressions (neutral, grin, anger, squint, resentment, and scream). Although showing a variety of facial expressions, participants photographed in fifteen perspectives and nineteen light circumstances(Gross, Matthews, Cohn, Kanade, & Baker, 2010).

2.1.7 Nova Emotions

Nova Emotions seeks to reflect facial and emotional state as observed in an uncontrolled nature. The data collected in a way where participants were sitting in front of a device, i.e., a gaming device that recorded their reaction in the game itself to scenes and challenges. At times, the game also responded to the player's response, which permitted random expressions gathered from a large pool of variations. The collection of Nova Emotions contains over 42K images taken, and 40 different people assume that. College students of age ages of 18 and 25 were the bulk of participants. The information offers several positions and enlightenments (Tavares, Mourão, & Magalhães, 2016).

2.1.8 FER 2013

The FER 2013 task was generated with 184 emotion-related keywords, including blissful, angry, using the Google image search APIs. To access up to 600 different search queries, the keywords coupled with phrases for class, sex, and ethnicity—the first 1000 photos for each question obtained with image data. Acquired images transferred into post-processing, which included cropping of the facial area and orientation of pictures. Frames then divided into the associated fine-grained emotion groups, unlabeled structures declined, and cropped regions modified. The resulting data comprises almost 36K photographs, split into eight classes (where7 positive expressions and a neutral one), with each type of emotions containing a few thousand images (the exception being disgust with only 547 frames) (I. J. Goodfellow et al., 2015).

2.1.9 The Yale Face Database

It consists of 15 men images with a total of 165 GIF-scale grayscale images. There are 11 photographs per subject: center light, w / glass, cheerful light, left lights, w / no glasses, regular, right-light, sad, sleepy, surprising and wink (Belhumeur, Hespanha, & Kriegman, 1997)

2.1.10 The Yale Face Database B

Contains images of 10 objects with 5760 single-source light under 576 lighting conditions (9 poses and 64 illumination conditions). A picture of ambient (background) lighting took for every topic in a specific position (Georghiades, Belhumeur, & Kriegman, 2001)

2.1.11 PIE Database, CMU

A compilation of 41,368 68 men's images, each in 13 different locations, 43 different light conditions and four different expressions (Sim, Baker, & Bsat, 2001)

2.1.12 Project - Face in Action (FIA) Face Video Database, AMP, CMU

This example is imitating the circumstance in the real world, for instance, when a person passes an airport check-in stage. Six cameras record three different angles of human faces. The focus range for four of the six cameras is less, whereas the four other cams are more concentrated. Expect to catch 200 topics in different periods during three sessions. In-door, as well as outdoor scenes, filmed within one day. The video sequences assume user-dependent posture and variance in voice (Goh, Liu, Liu, & Chen, 2005)

2.1.13 Cohn-Kanade AU Coded Facial Expression Database

Although 100 students from the ages of 18 to 30 years old 65% girls, 15% African, and 3% Asian and Latino, AU codes are reported as a database of facial Expression for the Cohn- Kanade section. The subjects asked to perform 23 facial demonstrations by an experimental experimenter with a single action unit. The grayscaling values digitized into 640 neutral-to-target image sequences employing 480, or 490-pixel arrays with 8-bit precision. Sequence files included in the image files; these are short text files explaining how to read images (T Kanade, 2000)

2.1.14 Japanese Female Facial Expression (JAFFE) Database

The collection contains 213 photographs depicted by ten women models of Japan, representing seven facial expressions (6 traditional facial expressions + 1 neutral). Each image was categorized into six emotional adjectives by 60 Japanese subjects (Kamachi, Lyons, & Gyoba, 1998)

2.1.15 IIIT - Cartoon Faces in the Wild

The IIIT-CFW is the cartoon archive for the wild, comes from Google's image search. Terms like Obama + cartoon, Modi + cartoon, and used to pick up cartoon pictures of 100 people. Eight thousand nine hundred twenty-eight cartoon facets of world-famous celebrities with various occupations included in the dataset. However, for cross-modal rehabilitation, such as photo2cartoon recruitment, we deliver 1000 real faces for the public. The IIIT-CFW is available to research problems such as facial synthesis, heterogeneous identification of the face, retrieval of cross-medium systems (Mishra, Rai, Mishra, & Jawahar, 2016)

2.1.16 MPI Facial Expression database

The MPI face expression index includes a significant number of natural feeling and interaction representations to fill this void. The study consists of 55 diverse facial expressions of 19 German participants. The use of the technical procedure to guarantee both defined and natural facial expressions used to express words. The protocol for the methodology focused on regular scenarios used to identify background information required for each word. There are three repetitions, two intensities, and three different angles of camera needed for all facial expressions. A comprehensive frame definition generates the dynamic and static versions of the database. The result shown in two separate conditions. To examine the significance and nature of this video series, in addition to a detailed analysis of the array. Findings show clearly that visual information alone is a significant contribution to the comprehension of conversational phrases. The MPI Facial Expression Database can enable people of the research community in various fields to investigate the care of a broader range of human facial expressions in addition to perceptual and cognitive science (Kaulard, Cunningham, Bülthoff, & Wallraven, 2012)

2.1.17 The Karolinska Directed Emotional Faces (KDEF)

A collection of 4,900 images representing human emotional expressions is Karolinska Directed Emotional Gesicht. The research performed at the Karolinska Institutet, Department of Clinical Neuroscience, Stockholm, Sweden by Daniel Lundsqvišt, Anders Flykt, and Professor Arne Öhman, in 1998. Initially, the content used in psychiatry and scientific research. In the field of vision, concentration, motivation, memory, reverse masking, the most explicit material developed. Particular attention, therefore, paid, for example, to align participants looking during the shooting with smooth, even bright, multi-angled shortage faces, to the usage of standard T-Shirt colors and to place the eyes and mouths in defined image coordination during scanning. The collection includes 70 men, each with seven different emotions from 5 perspectives, and each taken (twice) (Jeong & Ko, 2018).

2.1.18 KMU-FED database

A new test dataset introduced, known as the KMU-FED, to check the feasibility of the suggested method in the context of a real driver for FER in a real-life driving environment, including real-life problems. In a specific driving field, authors captured the test data set sequences using a NIR camera to generate the dataset. The KMU-FED database comprises of FEs obtained by drivers on the dashboard or wheel of the NIR system which contains 55 picture portions of 12 individuals that include different lighting changes and partial hair or sunglass occlusions (facing, left, correct and back) (Lundqvist, Flykt, & Öhman, 1998)

2.2 Ekman's six basic emotions

Commonly used methods via which the facial expressions detected are with the help of basic six emotions, i.e., fear, disgust, anger, happiness, sadness, and surprise. All these six expressions are known as universal expressions. Figure 1 shows the basic six emotions (Z. Zeng et al., 2008)Right to Left from top row: sad, happy, disgust and surprise, anger, fear. They are known as universal expressions because they found universally among human beings in all human beings, cultures, and ethnicities. Facial Expression is a crucial topic in the area of Artificial Intelligence and CV. Although FER can use various sensors, this study emphasis studies that explicitly use facial expression and facial expression images because facial expressions are the primary source of intercommunication. First, the overview of popular groups of FER systems and their significant algorithms illustrates the standard approaches of FER. They are deep learning approaches implemented via deep networks analyzed from end to end and interpret the findings of an updated plan of in-depth study, which is hybrid using a CNN that covers the temporal dimensions of consecutive frames for every frame and long-term memory. This research involves the spatial features of every single frame. This research is a summary of the current state of the art studies as well as a search for suitable paths to future practice.

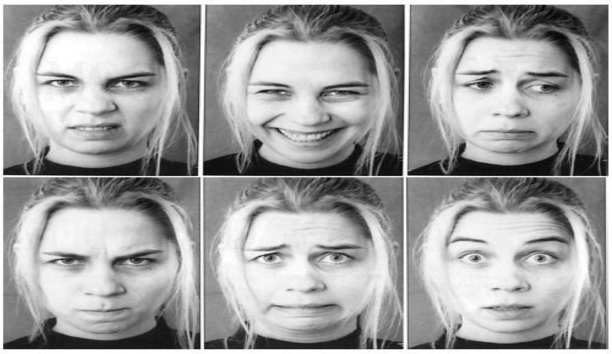
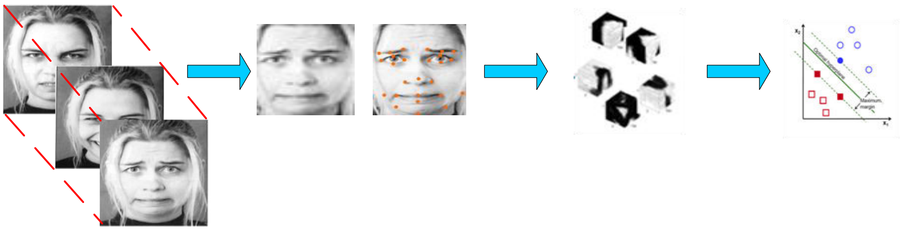


Figure 2.1: Example of six basic expressions (Z. Zeng et al., 2008)

Figure 2.2 shows the basic FER system. Usually, a basic FER program consists of two main steps: facial abstraction and facial expression recognition. But the principal aim of this work is to include recent developments on each of these measures, i.e., abstracting facial expressions and classifying facial emotion recognition on FER behaviors. At the same time, previous work has already examined the

FER method (in the last decade) (Sandbach, Zafeiriou, Pantic, & Yin, 2012),(Y.-L. Tian et al., 2005), (Căleanu, 2013). This review, therefore, aims primarily to present a new development on FER from 2013 to 2019 in particular.

2.3 Introduction to FER



a) Input images b) Face detection & Landmark detection c) Feature extraction d) Classification

Figure 2.2: Basic FER System

We have also performed several FER experiments to analyze the various classification strategies quantitatively. Although study analysis needs certain specific terminology and these specific terminologies have a crucial role to play, let's look at the specialized vocabularies mentioned below before evaluating FER work. The real-time FER system architecture depicted in Figure 2.3.

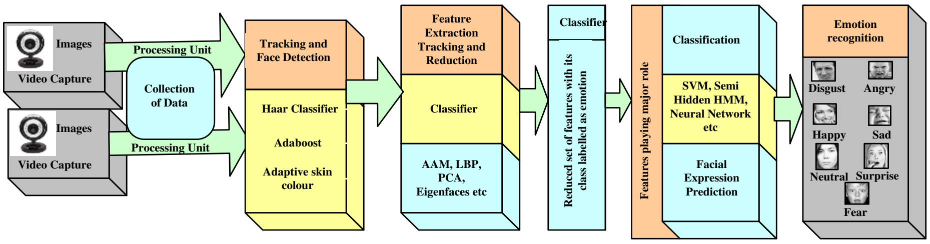

Images
Video Capture
Images
Video Capture
Processing Unit
Collection of Data
Processing Unit
Tracking and Face Detection
Haar Classifier
Adaboost
Adaptive skin colour
Feature Extraction Tracking and Reduction
Classifier
AAM, LBP, PCA, Eigenfaces etc
Classifier
Reduced set of features with its class labelled as emotion
Features playing major role
Classification
SVM, Semi Hidden HMM, Neural Network etc
Facial Expression Prediction
Emotion recognition
Disgust
Angry
Happy
Sad
Neutral
Surprise
Fear

Figure 2.3: The system architecture of a real-time FER system

2.3.1 Facial Action Coding System (FACS): This works on changes in facial muscles and can identify facial behavior that conveys human emotions identified by Ekman and Friesen in 1978 (Hamm, Kohler, Gur, & Verma, 2011). It involves motions of different facial muscles, known as Action Units, which are significant provisional improvements in the face (Jeong, Ko, Kwak, & Nam, 2017)

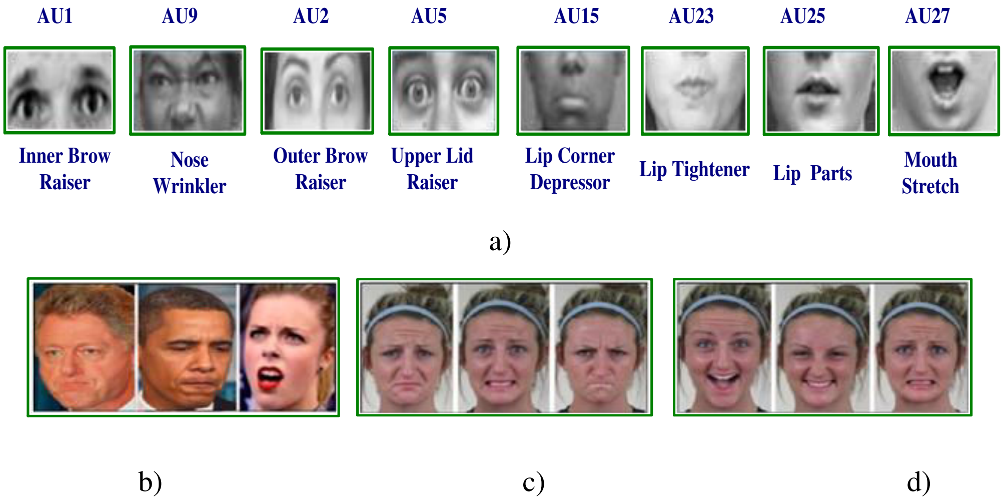
2.3.2 Facial Landmarks (FLs): Visually unique conditions defined in Figure 2.4 in face regions, such as the nose, eyebrow ends, and mouth. A FER function vector used for the pair coordinates of either one of the two landmarks or a local texture of the landmark. In three ways, FL detection techniques classified by modeling, such as ALM, a regression-based model integrating local with global modeling, and CNN approaches. FL detection strategies categorized into a regression model. FL models learned from rough shape and configuration initialization. Next, the original form moved step by step to a better pre-convergence location (Du et al., 2014)

2.3.3. Basic Emotions (BEs): Seven fundamental human emotions exist are happy, shocked, rage, sorrow, fear, repulsive, and neutral.

2.3.4. Compound Emotions (CEs): These two primary emotions mixed (Fabian Benitez-Quiroz, Srinivasan, & Martinez, 2016) listed 22 feelings: 7 basic emotions, 12 most frequently expressed composite emotions and three other emotions (appeal, dislike, and admiration). Many CE examples depicted in Figure 2.4. where a). AUs lower and Upper face (CK+ dataset images) (Lucey, 2010,) b).Spontaneous expressions (Facial images gathered from youtube) c). Basic emotions (sad, fearful, and angry) (CE dataset images)(Du et al., 2014,) d). Compound emotions (sadly fearful, happily disgusted and happily surprised) (CE dataset images) (Du et al., 2014)

2.3.5. Micro Expressions (MEs): Such motions take place involuntarily and display more natural, involuntary facial movements (Du et al., 2014). They travel within a shorter period to reveal the actual and intrinsic sensations of a person. Illustration 2.c describes different ME examples.

2.3.6. Facial action units (FAUs): There are codes of essential unilateral behavior (46 AUs) usually used by persons or muscle groups while producing facial gestures of a specific emotion, as seen in Figure 2.4 d. Solitary AU found to recognize facial feelings, and the system recognizes facial groups as per the mixture of AUs. For example, if a picture is annotated with an algorithm such as 1, 2, 25, and 26 AUs, the software label it as a "surprised" emotion type, as seen in Table 2.1.



a)

b) c) d)

Figure 2.4: Sample Examples of various Action Units and Facial Emotions

Table 2.1. Action Units involved in the compound and essential emotion category (Du et al., 2014)

Category	Aus	Category	Aus
Happy	25,12	Sad with fear	15,25, 1,4
Sad	4,15	Sad with disgust	4,10
Angry	4,7,24	Fear with surprise	1,2,5,20,25
Disgust	17,9,10	Fear with disgust	20,25,1,4,10
Fearful	1,4,20,25	Angrily surprised	26, 4,25
Surprised	1,2,25,26	Disgust with surprise	1,2,5,10
Happily Sad	4,6,12,25	Happy with fearl	25,26,1,2,12
Happy with Surprise	1,2,12,25	Anger with disgust	4,10,17
Happy with Disgust	10,12,25	Awed	25, 1,2,5
Sadly Surprised	25,26,1,4	Appalled	4,9,10
Sad with anger	15,4,7	Hatred	4,7,10

2.4 Face detection and tracking

The primary and crucial starting phase for FER is face detection. The picture divided into two parts at this point: one which includes faces, and the other, which depicts non-face regions. Table 2.1 gives a summary of the latest face recognition algorithms, their precision about correct face detection, and real-time environment results. The key characteristics used to describe the face in the video frame include facial structure, form, skin tone, and Expression. One algorithm used for face recognition is haar-classifier (Deshmukh, Patwardhan, & Mahajan, 2016), (Bailenson et al., 2008), (Sung, Lee, & Kim, 2006), (Adeshina, Lau, & Loo, 2009)]. Haar-like properties for recognition of artifacts, as shown in Figure. 2.5 were working (Wilson & Fernandez, 2006). One can conveniently scale Haar features by growing the scale of the group of pixels being analyzed or decreasing it. The usage of the software expects to recognize items of different sizes. Here features are particularly successful in identification because they are ideal for face detection throughout the training phase. In the training cycle itself, the Haar classifier used face detection to identify a group of features that relate most to the issue of face detection. In the development phase, the technical difficulty is much smaller, resulting in a very fast degree of facial speech. Scientists typically use adaptive skin-color models to classify the face (Mayer et al., 2008), (Happy, George, & Routray, 2012), (Sandbach et al., 2012) as defines the initial goal by the skin-color pattern dependent facial detection process. At YIQ, I say, as shown in Figure 2.6 of face-colored skin improves, varying from 30 to 100. Simultaneously, as illustrated in Figure 2.7, the YUV space. 4, they remember the skin-color spectrum of the face is between 105 ° and 150 °. By synthesizing the YIQ and YUV color schemes, we set up a skin-colour interface for the initial picture. Thanks to the usage of skin colour for segmentation, accuracy is high and can be easily separated from the non-face part. The trouble with this method is they aren't dealing with varying light levels. Because of great computational difficulty, the adaptive gamma correction approach is used to get rid of lighting and pose, which is not ideal in a real-time setting. Another face detection solution is Adaboost (ensemble method) due to low computational complexity and high precision, which is ideal for setting in real-time (Peng, Wen, & Zhou, 2009), (Sung et al., 2006), (Hassan, Maqbool, Ahsan, & Qayyum, 2010)].

Table 2.2: Face recognition algorithms

Algorithm	Performance in real-time	Accuracy
Haar classifier (Ghimire & Lee, 2013), (Wilson & Fernandez, 2006)]	While performing face detection while training the computational complexity is reduced to a greater extent because of a group of features which contribute the maxim urn	Due to suitable Haar features the accuracy is quite high
Adaptive skin color (Punitha & Geetha, 2013), (Shbib & Zhou, 2015)	The computational complexity is very high because the approach like adaptive gamma correction used to solve the problem of illumination, so it is not suitable for real-time applications.	It fails in various illumination levels, but accuracy is good as it can detect skin color
Adaboost (Sung et al., 2006), (Lu, Huang, Chen, & Yang, 2007) contour points (Geetha, Ramalingam, Palanivel, & Palaniappan, 2009)	Because quite a less number of features the computational complexity is quite low so computation cost is minimal	Accuracy is very high as the classifier is robust and single face need to be detected that too via contour points

A combo is a simultaneous method, where the relation to the next type is one output for classification. The second classifier acts on the first classifier's performance, which leads to higher precision. Many classifiers can be cascaded in this method. Images of the face are trained, and a robust classifier is built that helps to distinguish high precision. No computational expense is needed for affect recognition as the new look is connected to the model the classifier has placed in place.

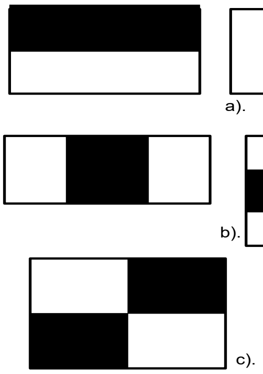


Figure 2.5: Common HAAR Feature
(Wilson & Fernandez, 2006)
a) Edge features
b) Line features
c) Four rectangle features

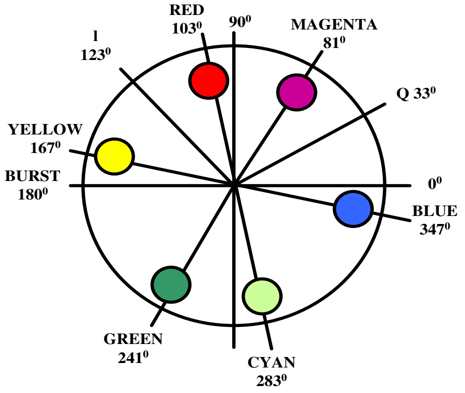


Figure 2.6: YIQ color model
(Ibraheem et al., 2012)

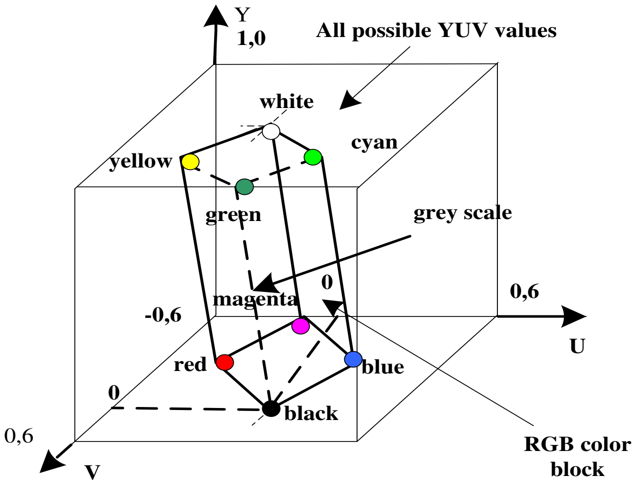


Figure 2.7: YUV color model (Ibraheem et al., 2012)

The method for identification and identification of face motion is not only practical but also useful when it comes to productivity since the characteristics extracted are less numerous and make it ideal for a reliable setting. For the identification of a face in a series, face- tracking is used. To integrate face changes over time, the spatial relationship between the frames needs to be utilized in terms of changes in size, position, and location of the search for the face. Tracking accepts the variations in the image's temporality. Two groups can be used to separate face monitoring: i) head tracking and ii) face tracking

(Verma, Schmid, & Mikolajczyk, 2003). Core monitoring methods focus on the overall head operation while attribute monitoring methods concentrate on the management of the attributes extracted from the forehead. Indirect supervision of the camera helps to keep students under the track. The author established head tracking methods in this section, and in the following section, the tracking methods of attributes have viewed.

Color-oriented technology, including adaptive skin color ((Zhao-Yi, Yan-Hui, & Yu, 2010),(Peng et al., 2009)) or forms-related solutions such as active-shape models (Punitha & Geetha, 2013), uses the intakes from the whole head that can be area-based (Geetha et al., 2009). Approaches that utilize full-head perception cannot withstand occlusion. Color-based systems aren't sufficiently flexible to accommodate shifts in lighting. The solutions for monitoring the face of user from videos as it is in the algorithm of mean shift, i.e., STAAM, and motion details are discussed in Table 2.2. The mean algorithm change method offers substantial accuracy as algorithms by adaptive skin color identify target window features and consider the correlations between candidate window and target window. Majorly it is an iterative technique that causes time problems and is therefore less suitable for the situation in real-time (Zhao-Yi et al., 2010) STAAM is a visual device that fitts with two cameras that help build facials on a 3D scale.

STAAM enhances precision because it does not rely on the stability of the camera, but due to the use of the 3D form and presentation model, technical complexity is also enhanced (Sung et al., 2006). The approach to business learning is acceptable in real-time because it is incredibly accurate and computer-efficient. Image recognition is a method of facial stimulation that first locates the face and then reduces the costs for computing. The experimental results indicate that approximately 95% of the frames in the video are correctly monitored by the system, as these are not prone to lighting and face size conditions (Geetha et al., 2009). The downside to movement awareness is that it seems like only one face moves in the picture with facial tracking technology. These methods track face-to-face by documenting a variety of features, including the ears, lips, nose, and eyes. These algorithms monitor contour dots

(Geetha et al., 2009) or obey each tracker's eyes and mouths (Verma et al., 2003). It adds that it must suffice to observe these components to use only frontal facial views.

2.5 Extraction methods of the facial features for static images

For fixed or static photos, two types of methods of facial extraction are available: 1) geometric methods based on characteristics 2) methods based in aspect.

2.5.1. Geometric feature-based methods

It recognized that it is nice to have a face with lips, ears, brows, nose, collar, and chin. The scale, shape, orientation, and location of these organs influence the development of facial Expression. By imagery, facial elements like the mouth, nose, eyes, and brows can be organized and coordinated. Spatial methods use spatial connections between facial characteristics to eliminate facial symptoms. Nonetheless, usually capturing geometric aspects requires a sophisticated technique for the recognition of shapes in stages. That is difficult to adapt in natural or real settings to changing circumstances. Furthermore, graphical solutions, such as wrinkles and furrows, are often ignored for skin textural changes required for facial Expression to be modeling. Geometric features based on three distinct extraction methods: active forms (ASMs), dynamic appearances models (AAMs), and SIFTs.

2.5.1.1 ASM: (Timothy F Cootes, Taylor, Cooper, & Graham, 1995) propose a feature-matching solution that focuses on an active shape model (ASM) mathematical algorithm. ASM is a (PDM) point distribution model that measures several shape variations and a range of adaptive models that aggregate the gray values over a set of main feature points. Figure 8 offers an overview of the ASM role extraction method (Chang, Hu, Feris, & Turk, 2006), defined by 58 different facial symbol aspects. The ASM method comprises of different two stages. Models constructed from the training samples in the first form with several annotated landmark function points. Instead, regional texture models developed for each specified end of service. Secondly, an iterative search method can be used to transform the structure of the example according to the two construction models. (Shbib & Zhou, 2015) used as the FER nose, describe the geometric displacement between the location of the predicted ASM feature point and mean ASM shape. (Cament et al., 2015) have developed an enhanced version of ASM facial recognition in recent years, called active type and mathematical models (ASSM), which has computational applications for FER.

2.5.1.2 AAM: Coos et al. created the Active Appearance Model (AAM) in 2001 (Timothy F. Cootes, Edwards, & Taylor, 2001) AAM extend ASM effectively by acquiring widespread knowledge of its type and shape. AAM develops a simulation model based on statistical data training in detail, followed by a useful data test evaluation using this mathematical model. Under ASM, AAM not only uses global knowledge of shape and texture but also performs a comparative study of the local culture of the surface to determine the relationship between form and texture.

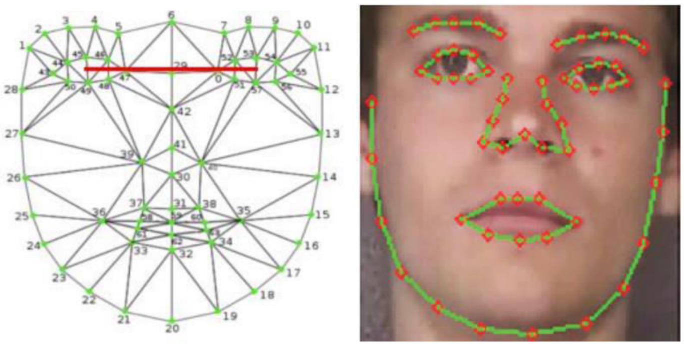
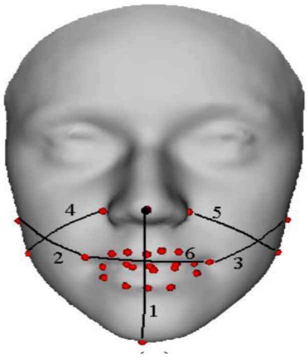


Figure 2.8: ASM based Facial landmark detection using 58 facial landmarks (Cament et al., 2015)

To achieve a FER method, The differential AAM and variable learning implemented (Cheon & Kim, 2009). The difference between AAM reference images and input images(such as images with neutral voice) measured for assessment of the differential-AAM functions. Some advanced AAM models have also been developed, such as AAM-based Directed Gradient (HOG) histograms (Antonakos, Alabort-i-Medina, Tzimiropoulos, & Zafeiriou, 2014) dense-based AAM and AAM-based regression (Y. Chen, Hua, & Bai, 2014). The efficacy of these newly developed AAM variants on FER is a fascinating piece of work to investigate.



Components	#Points
Nose	2
Face contour	5
Mouth	20
Line 1	12
Line 2	16
Line 3	16
Line 4	16
Line 5	16
Line 6	9

a) b)

Figure 2.9 .SIFT based feature extraction method (Berretti et al., 2010) b) The list component wise landmark detail.

2.5.1.3 SIFT: Scale-invariant Transform function, i.e. (SIFT), has been suggested by David Lowe (Lowe, 1999, 2004) is called a matching descriptor based on a local image. The features of SIFT are invariant for size, rotation, and translation of the pictures and not completely varied by a change in the lighting condition. A SIFT based feature extraction method used in Berretti et al., Figure 2.9 shows (Berretti et al., 2010) to obtain SIFT-descriptors at these central locations as main marks for the facial marking in the major morphological areas of the face a). A sample image that located with 85 facial landmarks and lines. In this case, the SIFT-functional Extractor used (Soyel & Demirel, 2010) and has recently discriminated against the scale-invariant feature transformation method, that can make useful judgments about the final appearance. Li et al. (Y. Li, Liu, Li, Huang, & Li, 2014) used geometric algebra (GA) to add a new transformation of the scale-invariant function called GA-SIFT to the multispectral image. A novel concept of multispectral objects based on the GA theory was initially developed with spectral and spatial definitions. Thirdly, a multispectral image had been given scale space. Second, GA-dependent volatility was observed in Gaussian close SIFT representations. The feature points are defined and listed on a permanent basis, based on the GA theory.

2.5.2. Appearance-based methods

Appearance-based approaches advocate using entire or different regions on a face picture to represent the hidden characteristics of a face picture, particularly those of subtle changes, such as wrinkles and cracks. Two descriptive appearance-based methods are mainly available for knowledge extraction, by (Ojala, Pietikäinen, & Harwood, 1996) such as local binary patterns (LBP and the wavelet representation by Gabor.

2.5.2.1 LBP: Local Binary Pattern (LBP) (Ojala et al., 1996) is a trained textural description operator, capable of quantifying the image and extracting information from the adjacent texture. The primary benefit of LBP operators is its strength, innovative and gray invariance that can come out of the displacement, distortion, and illumination problems of the image. Consequently, the LBP operator very well established. The LBP unit FER Extraction, seen in (Shan et al., 2009), is illustrated in Figure 2.10.

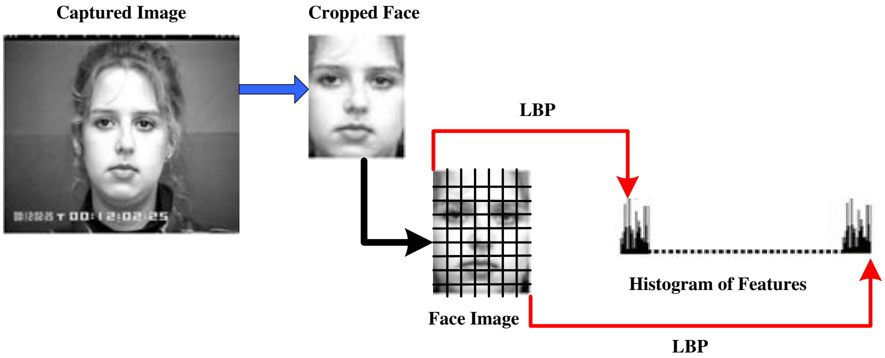


Figure 2.10. Feature extraction method using LBP (Shan et al., 2009)

Three main phases monitor the extraction method used for the processing of LBP (Shan et al., 2009). A picture of the face first divided into several parts, and they didn't overlap. Second, for every computer, the LBP histograms had been created. In previous research (S. Zhang, Zhao, & Lei, 2012a), (Xiaoming Zhao

& Zhang, 2012), they analyzed the effectiveness of dimensionality reduction strategies on FER practices by the LBP operator, such as biased analyzes of local fisheries. Some types of LBP operators found in the literature throughout the last years (D. Huang, Shan, Ardabilian, Wang, & Chen, 2011). Typical LBP implementations currently include Local binary templates LBP(G. Zhao & Pietikainen, 2007), Three orthogonal planes LBP known as LBP-TOP (G. Zhao & Pietikainen, 2007) geographic directional pattern (LDP) (Owusu, Zhan, & Mao, 2014), spatial mapping (LTP) (Ahsan, Jabid, & Chong, 2013). Li et al. (X. Li, Ruan, Jin, An, & Zhao, 2015) implemented a polytypic local multi-block binary pattern known as P-MLBP for a three-dimensional FER. P-MLBP contains all of the essential facial expression-dependent parts that accurately reflect characteristics and integrates 3D image profile and texture information to enhance facial appearance.

2.5.2.2 Gabor: The Gabor wavelet representation (Zhengyou Zhang et al., 1998) is a conventional way of expressing the features of facial discourse. A filter series filters a picture in detail, and the effects filtered show the connection between the gradient of the local pixel, texture similarity. The technique for the Gabor wavelet representation used to extract the facial expression attribute. It can detect multi-scale, multi-directional touch movements, and had a small effect on lighting changes. Figure 2.11 shows the representation of the Gabor wavelet used in (Zhengyou Zhang et al., 1998), where 18 Gabor kernels used in three sizes and six directions in total. Liu et al. (S.-s. Liu & Tian, 2010) proposed a FER approach based on the study of the Gabor wavelet functions and the central kernel of components (KPCA).

As a local Gabor filter used for bypassing the standard Gabor filter, which contributed to the assumption that the processing speed increased, Gu et al. (Gu, Xiang, Venkatesh, Huang, & Lin, 2012) carried out FER with the synthesis of radial coding and classification characteristics from the local Gabor. The input images for this study were initially exposed to multi-scalar local Gabor filters and were then employed by radial grids to encode Gabor decomposition. Using Gabor techniques of extraction to eliminate large amounts of the facial traits representing various facial types of deformation, Owusu et al. (Owusu et al., 2014)have recently established a neural-Adaboost FER system.

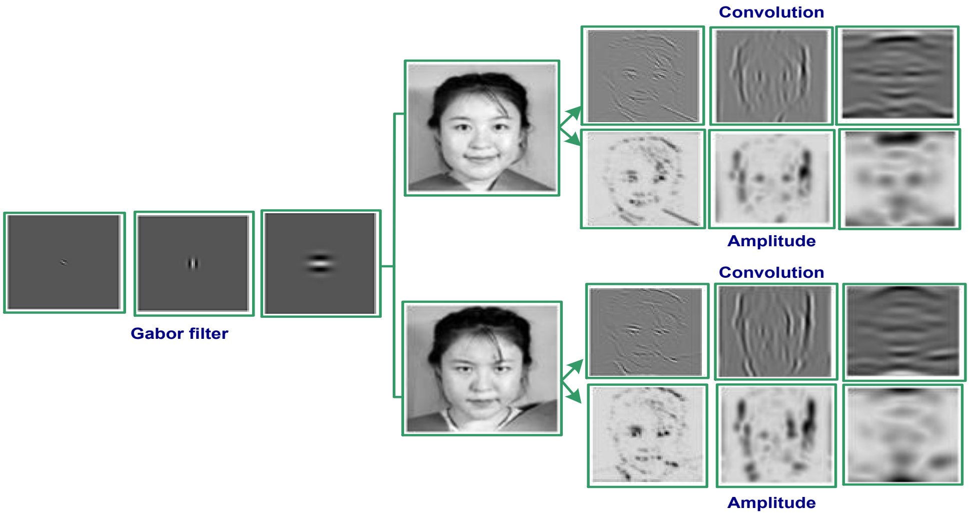


Figure 2.11: Gabor wavelet representation: Examples of three kernels (Zhengyou Zhang et al., 1998)

2.6 Methods of extracting the facial features for dynamic image sequences

Complex imaging loops reflect the facial face's constant agitation cycle. For a complicated set of images, facet speech aspects are conveyed mainly by movements of facial muscles and deformation. There are currently two conventional methods for extracting features for complicated image sequences: optical rotation, and feature point detection. Various methods compiled in Table 2.3

Table 2.3: Feature Extraction methods

Algorithm	Performance	Accuracy	Model Type
LBP(Happy et al., 2012)	Challenging to implement in real-time as it has got high complexity in time	Accuracy is quite high as it has got both local and global features	Appearance-based
Gabor feature (Happy et al., 2012)	High memory requirement and less computational speed for real-time	High discriminative power	Appearance-based
PPBTF (Lu et al., 2007)	Because of pattern mapping, it has got very high speed	Accuracy is high due to template patterns	Appearance-based

Canny edge detection and AAM (Zhao-Yi et al., 2010)	Computational complexity reduced as the non-profiled part is discarded during computation	Accuracy improved because of the identification of a profile part of the facial image	Geometric based
MRASM and LK-flow method (D. Huang et al., 2011)	LOW computational cost	Jittering movement leads to high accuracy	Geometric based
MICV (Lowe, 2004)	Less complexity is low because of fewer features	Only the mouth region is focused, so accuracy is high	Appearance-based
Pyramid LBP (Khan, Meyer, Konik, & Bouakaz, 2013)	Time and memory-efficient as it extracts features in a pyramid	High accuracy because of discriminative ability	Appearance-based

2.6.1. Optical flow

Negahdaripour (Negahdaripour, 1998) redefines the visual flow system as dynamic forms change shape and radiation. For each pixel in an image, the fundamental definition of an optical flow system is a vector with a distance—these vectors of speed form the field of motion of an image. At a time of operation, the point of reference is the actual position of the target. The optical flow approach has a significant advantage, i.e., the optical flow not only holds the specifics of the movement but also holds a detailed understanding of an object's 3D structure. The FER region's optical flow technique is majorly utilized to derive affect characteristics from a sequence of complex images since it identifies facial faults and describes the image sequence movement patterns. Figure 2.12 shows a method used to remove optical flow (Cohn, Kanade, & Li, 1998) that takes place in two sequences of Facial Expression. (Cohn et al., 1998) used a detailed multi-resolution optical flow wavelet to research comprehensive face movement, and then specified PCA-based Eigen flows in both horizontal and vertical directions to represent the related short fluid fields. (Yacoob & Davis, 1996) used optical displacement theory and gradient theory to explain the temporal and spatial variations of the images across successive frames. Alternatively, movements of the facial muscles calculated to classify different expressions based on variations in the facial movement vectors. (Sánchez et al., 2011) compared two optical approaches that rely on the wave to FER. Growing was distinctive and powered by a minute number of exact facial lines that need to have opted. The other approach was comprehensive, and even more, cases were employed, distributed evenly around the central facial region.

2.6.2. Feature point tracking

Tools for the interpretation of functional points frequently select functional points with significant eye and mouth adjustments. And after that, you can obtain knowledge about the displacement or deformation of the facial feature. The feature point surveillance method used in (Pantic & Patras, 2006), which selected fifteen feature-points based on Facial Action System (FACS), was used to detect trait-point movement in image sequence with the particle filter. (Tie & Guan, 2012) proposed a method that would automatically delete 26 benchmarks from video samples in one face-model and track the benchmarks through multiple particle filters. To eliminate outstanding data from video sequences, (Fang et al., 2014) used the traditional facial point analysis system, but did not use any individual preprocessing or external user-furnished details to select peak speech frames.

2.7 Feature Reduction

A complicated issue in the field of pattern recognition, like a study of facial character emotions, involves an enormous dimension of vectors in the input function. The next logical step is for the reduction of difficulty of real-time calculations by incorporating techniques of feature reduction to the set of discriminatory services. Most of the existing strategies outlined in Table 2.5 restrict the ultimate selection of characteristics to any subset of emotional classes such as sorrow, frustration, terror, anticipation, annoyance, outrage, and rage. It allows to take only those function subsets that most add to the spectrum of emotions offered, allowing us to reduce the size and thus increase efficiency. The selection of devices using Adaboost is most useful to detect, as it tries to pick the right characteristics (Alazrai & Lee, 2012). Table 2.4 outlined the feature tracking models along with its accuracy and performance.

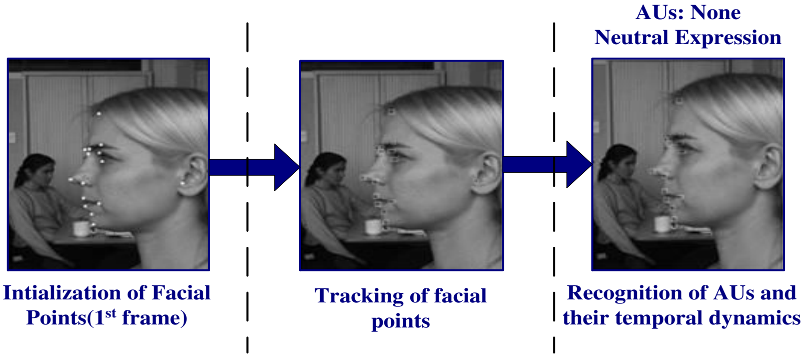

AUs: None
Neutral Expression
Intialization of Facial Points(1st frame)
Tracking of facial points
Recognition of AUs and their temporal dynamics

Figure 2.12: The feature point tracking method (Pantic & Patras, 2006)

Table 2.4 Feature tracking

Algorithm	Accuracy	Performance	Model
Guided particle optimization (GPSO) (Ghandi, Nagarajan, & Desa, 2009)	Accuracy is good (85-95%)	Less efficient in comparison to others	Geometric based
Optical flow calculation(X. Li et al., 2015; Zhengyou Zhang et al., 1998)	Accuracy is high as the tracking of the feature point is based on the brightness of pixels	As a lesser number of feature tracking is required, so it is suitable in real-time	Geometric based

Most methods used the projection of Eigenspace (also known as PCA) (Samnani & Jain, 2017), (S.-s. Liu & Tian, 2010), (Shan et al., 2009) to minimize dimensionality, that is the space transformation. The PCA then reduces flexibility. By projecting the set of the specified features on the main components (D. Huang et al., 2011), The LBP features extracted from the initial images in the face substituted by a Local Fisher Discriminatory Facial Discrimination (LFDA) analysis (Gu et al., 2012) LFDA is used to produce tiny embedded images of high-dimensional LBP data, with substantial improvements in efficacy for FER tasks.

2.8 Facial Expression classification

Detection of facial Expression contributes to the development of an efficient facial expression detection system. Simplified methods of FER classification include the Markov implicit model (HMM), the artificial neural Network (ANN), a Bayesian (BN) network, the neighbor k-nearest (KNN), and the vector support (SVM).

Table 2.5 Methods of feature reduction

Algorithm	Real-time performance	Accuracy
PCA (Cament et al., 2015; D. Huang et al., 2011), (S.-s. Liu & Tian, 2010), (Shan et al., 2009)	As the features are decidedly less, so the computational speed has improved	Provides good recognition accuracy as only those features got selected that are prominent
LFDA (Verma et al., 2003)	The features got reduced so in this case, the computational complexity is very less	High recognition rate
Adaboost (Alazrai & Lee, 2012)	Less complicated while computation as the features have reduced	High accuracy for emotion recognition because of powerful features.

.

2.8.1 Hidden Markov model-HMM

HMM is a process known as Markov and which mainly concealed unknown parameters used for characterization of a mathematical model's spontaneous signaling information. HMM is made up of two methods that interweave, which is Markov's corresponding, unobservable lines, with many properties. The other is a probability density set distribution, which applies to any condition. The following triplet described an HMM as:

$$\lambda = (A, B, \pi) \tag{2.1}$$

In that A is the probability matrix for a state transformation, B is a distribution for the observer, and p is the original state distribution. B is a probability matrix with a discrete HMM density, and HMM B with constant density is implied with probability distribution function measurement parameters, for example, a Gaussian or Gaussian distribution. The commonly used density function (PDF) representation of the pattern called the finite mixing form:

$$b_i(O) = \sum_{k=1}^{M} c_{ik} N(O, \mu_{ik,} U_{ik}), 1 \leq i \leq N \tag{2.2}$$

Where M is for different observations, N shows the total states in the HMM model, Cik is the parameter which shows the mixture for the k_{th} mixture for the state I, $N(O, \mu_{ik}, U_{ik})$ is the Gaussian pdf with the average vector µik and the covariance matrix Ui. The adaptive multi-strength HMM FER scheme was proposed by (Aleksic & Katsaggelos, 2006), using the multi-stream HMM method to implement reliability stream of the group dependent weights for facial and facial expression (FEPs) parameters(Y Sun & Akansu, 2014) proposed in their video sequences an integrated Markov regional secrecy model (RHMM) framework.

2.8.2. Artificial Neural Network –ANN

ANN proves to be a robust statistical method that can differentiate data from nonlinear dynamic interactions between input and output. Two ANN models, i.e., a multilayer (MLP) network and neural network design (RBFNN), have mainly used as a radial basis for FER in recent years. In all ANN type, that is to say. MLP and RBFNN are quite similar, just below is the fundamental RBFNN definition. A 3 layer network is feed-forward and contains the first single layer as the input layer, second layer as a hidden layer, and the final layer as output and classified as RBFNN. The example of the RBFNN structure displayed in Figure 2.13. The input results labeled with a part of an input vector x for each input neuron. The hidden layer designed to combine input and extract data. There are a single biased neuron and n Neurons in the hidden layer. Radial base function (RBF) that is of the hidden layer is usually referred to as

$$y_i = \begin{cases} exp\left(\frac{\|x-p_i\|}{2\sigma_i^2}\right), & i = 1,2, \dots \dots n, \\ 1 \ , & i = 0 \end{cases} \tag{2.3}$$

In which pi and si show middle and the neuron width, symbol kk shows the Euclidean distance, Central pi refers to the neuron and hidden layer weight vector. The closer x is to pi, the higher the Gaussian function means the value. The output layer made up of m neurons that fit potential problem categories. The output layer is directly connected to the hidden layer, with the following formula calculated by a weighted sum of hidden neurons, which is linear.

$$Z_j = \sum_{i=0}^{n} y_i \; w_{ij}, \quad j = 1,2, \dots \dots .., m \tag{2.4}$$

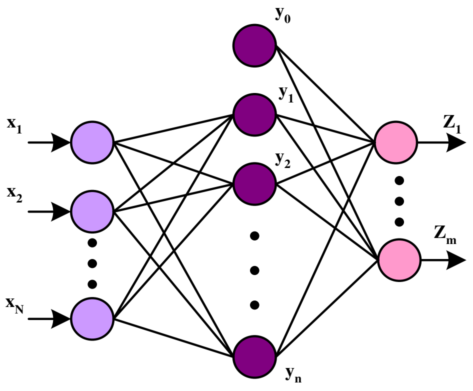


Figure 2.13 The RBFNN Framework

Where w_{ij} shows the weight of i^{th} and the j^{th} layer of a hidden neuron and output neuron (Ma & Khorasani, 2004) proposed a new FER technique using the 2D (DCT) Discrete Cosine Transformation that acts as a feature for complete facial image and a feed-forward single layer constructive neural network as a speech classifier for a face. For the teaching and generalizing views, the highest recognition rates reached were up to 100 percent and 93.75 percent (without rejection), respectively. For understanding the basic facial emotions and static images in-depth, (De Silva, Ranganath, & De Silva, 2008) proposed an updated version of RBF's known as Cloud-Based functions. The highest recognition precision was 96.1 percent for the CBF neural network solution. (Kaburlasos, Papadakis, & Papakostas, 2013)has recently been described as a fuzzy FER neural network, FAM, as a radically different extension of the fuzzy ARTMAP neural classifier (FAM).

2.8.3 Bayesian network-BN

BN is the Network that is transparent and probabilistic. The so-called probabilistic reasoning involves the use of certain scientific evidence to deduce the other specifics of likelihood. BN is built, based on probabilistic logic, to solve the problem of ambiguity and incompleteness. A BN classifier uses a spatial acyclic graph to describe the relationship between the feature data and sample labels. This graph reflects base BN settings. In general, a defined type used to teach BN classifiers; the standard example is the classifier Naive-Bayes. Given a set θ with a BN classifier, to classify an observed feature vector $x \in R^n$ an ML(maximum likelihood) based optimizing classification rule with the n dimensions, to one of $|C|$ class labels, $c \in \{1,2, \ldots \ldots |C|\}$,is denoted by:

$$\hat{c} = argmax\, P(x|c{:}\,\theta), \qquad (2.5)$$

When building the BN classification, there are two forms of guidelines for choices. The first is to pick the network configuration, i.e., used to evaluate the dependencies of the majority of variables in the list. The other is to understand how the computer conveys results. Cohen et al. employed multiple BN classifiers to identify, based on improved delivery requirements, visual facial Expression, and attribute dependence configurations. They also utilized Naive BN classification for updating the Gaussian distribution for Cauchy, using the Gaussian tree-improved Naive Bayesian (TAN) classification method for the relation of different face behaviors. (Xi Zhao, DellandréA, Zou, & Chen, 2013)suggest a transparent probabilistic approach to the definition of a 3D Facial Expression and Action Group, which relies on an advanced Bayesian Belief Network (BBN). The proposed BBN puts together Bayesian feature model-based inference and the Gibbs Boltzmann distribution, offering an approach the is hybrid and helped in combining visual, morphologically characteristic with imagery attributes.

2.8.4. K-nearest neighbour-KNN

KNN is like a classification algorithm based on case studies. A sample with k closest examples in the field of function is the KNN system principle; utilizing a majority voting of neighbors, its marking is allowed to the most common class among its kNNs. The KNN algorithm of classification often used for measuring the Euclidian distance without previous information. Given two vector x= (x_1; x_2; _ _ _; xm) and y= (y_1; y_2; _ _ _; y_m), their Euclidean distance is represented as

$$d(x,y) = \sqrt{\sum_{i=1}^{m}(x_i - y_i)^2} \tag{2.6}$$

(Sebe et al., 2007) employed geometrical features on the Cohn-Kanade map using the KNN method to achieve a maximum rating accuracy of 93 percent. (Gu et al., 2012) Gabor features and classifier synthesis for the KNN-based FER has been used locally with k=1 radial encoding.

2.8.5 Support vector machine-SVM

SVM is a computational risk minimization technique that is superior to the traditional approach used by the classical neural networks to numerical risk minimization. The SVM theory is that input vectors converted into a nonlinear conversion into an upper space of dimensionality, and a correct hyper-plane found to differentiate the results. The above process completed with the following question of optimization:

Given the training data set (x_1, y_1),..., (x_l, y_l), $y_i \in \{-1,1\}$ for an optimal hyperplane, a transform, i.e., noon linear, Z =Φ(x), is used for dividing training data linearly. The offset and the weight, i.e., b and w are decided as follows:

$$\begin{cases} w^T Z_i + b \geq 1, & y_i = 1 \\ w^T Z_i + b \leq -1, & y_i = -1 \end{cases} \tag{2.7}$$

$$min_{w,b}\ \Phi(w) = \frac{1}{2}(w^T w) \tag{2.8}$$

$$s.t. \quad y_i(w^T Z_i + b) \geq 1, i = 1,2, \dots \dots, n \tag{2.9}$$

The decision function can be written according to the language method

$$K(u, v) = \sum_i \alpha\varphi_i(u)\varphi_i\,(v) \tag{2.10}$$

The decision function can be written according to the language method.The kernel idea is uniquely defined as Hilbert space H by the non-negative symmetric K(u;v) function:

$$Z_i{}^T Z = \ \Phi(x_i{}^T \quad \Phi(x) = \ K(x_i, x) \tag{2.11}$$

Where kernel function is K in H space. In the Hilbert region H this shown as an internal product:

$$f = sgn\left[\sum_{i=1}^{l} \lambda_i y_i\, K(x_i, x) + \ b\right] \tag{2.12}$$

The decision role can then be modified as:

$$K(x_i, x_j) = x_i{}^T x_j \tag{2.13}$$

The polynomial kernel function denoted as follows:

$$K(x_i, x_j) = (\gamma x_i{}^T x_j \ + \ coefficient)^{degree} \tag{2.14}$$

The RBF kernel function denoted as follows:

$$K(x_i, x_j) = \exp(-\gamma|x_i - x_j|^2 \tag{2.15}$$

The sigmoid kernel function is given as:

$$K(x_i, x_j) = \tanh(\gamma x_i{}^T x_j + \ coefficient) \tag{2.16}$$

The used SVM model has four essential features, such as polynomial feature, linear feature, sigmoid and RBF feature, and has a set of functions. The following are described in the SVM model. For the improved FER feature selection method with a 3D geometric facial point position, the SVM classification was used by (Yurtkan & Demirel, 2014). (Ghimire & Lee, 2013) used FER graphics in the face picture series, achieving 97% accuracy with the help of SVM graduation program in CK

2.8.6 Sparse representation-based classification-SRC

SRC(Wright, Yang, Ganesh, Sastry, & Ma, 2008) formulated as a lightweight sensing (CS) basis (Donoho, 2006). The theory of the SRC approach is based on the premise that a dictionary is created with the entire collection of training samples. The classification question is therefore posed as one of seeking a sparse definition as an extended mix of training samples via a smoothing resolution of the problem of l1-standard optimization. This case formally is established for single-class samples.

$$y_{k,test} = \alpha_{k,1}y_{k,1} + \alpha_{k,2}y_{k,2} + \cdots + \alpha_{k,n_k}y_{k,n_k} + \varepsilon_k \qquad (2.17)$$

$$= \sum_{i=1}^{n_k} \alpha_{k,i}y_{k,i} + \varepsilon_k \qquad (2.18)$$

Where a sample of the kth class is $y_{k,test}$ and the ith training sample of the kth class is $y_{k,i}$ shows the corresponding weight is given as $\alpha_{k,j}$ and ε_k denotes error of approximation.

For training purpose, the above mentioned can be re-written fo c object classes and modified as

$$y_{k,test} = \alpha_{1,1}y_{1,1} + \cdots + \alpha_{k,1}y_{k,1} + \cdots + \alpha_{k,n_k}y_{k,n_k} + \cdots$$
$$+\alpha_{c,n_c}y_{c,n_c} + \varepsilon \qquad (2.19)$$

In a matrix form i.e $y_{k,test} = A\alpha + \varepsilon$

$$\text{where}\begin{cases} A = \left(y_{1,1}\middle| \dots \middle|y_{1,n_1}\middle| \dots \middle|y_{k,1}\middle| \dots \middle|y_{k,n_k}\middle| \dots \middle|y_{c,1}\middle| \dots \middle|y_{c,n_c}\right) \\ \alpha = [\alpha_{1,1} \dots \alpha_{1,n_1} \dots \alpha_{k,1} \dots \alpha_{k,n_k} \dots \alpha_{c,1} \dots \alpha_{c,n_c}]^{`} \end{cases} \qquad (2.20)$$

to get a α weight vector, the norm minimization problem of l_1 need to solve as follows:

$$\min_{\alpha}\|\alpha\|_{1,} \quad subject\ to\ \|y_{k,test} - A\alpha\|_2 \leq \varepsilon \qquad (2.21)$$

The quadratic programming method can be used to solve this convex optimization problem. The sparse solution of α is shown as follows:

Step 1: Solve the problem of norm minimization equation $l_{1.}$

Step 2: For each class, I, find the reconstructed sample $y_{recons}(i) = \sum_{j=1}^{n_i} \alpha_{i,j}\, y_{i,j}$ as the given sample for a test by $r(y_{test,i}) = \|y_{k,test} - y_{recons}(i)\|_2$

Step 3: The class label decided by using the rule: identify$(y_{test}) = $ arg $min_i r\ (y_{test}, i)$for the given test sample.

In earlier works (S. Zhang, Zhao, & Lei, 2012c), (S. Zhang, Zhao, & Lei, 2012b), SRC's output was evaluated when classifying images with a noticeable or occluded facial feature. They observed that relative to the SVM, SRC, nearest neighbor (NN), and nearest subspace (NS) had better performance and higher robustness. (Mohammadi, Fatemizadeh, & Mahoor, 2014) suggested the development of a dictionary focused on PCA for the simple meaning and classification of Facial Expression. The descriptive facial Expression for every subject is taken initially from an image, i.e., the neural face of the same problem. For each class of facial Expression, the PCA approach employed to model discrepancies in these discrepancies. As dictionary electrons were the key components studied. Thus a particular test picture for six standard facial expressions was separately presented as critical elements with a linear blend. (Ouyang, Sang, & Huang, 2015)have recently built a reasonable and steady FER by fusing a variety of sparse and representative classifiers, e.g., HOGCSRC and LBPCSRC fusing. Table 2.6 (Du et al., 2014), (Fabian Benitez-Quiroz et al., 2016),(Ghimire & Lee, 2013; Suk & Prabhakaran, 2014),(Happy et al., 2012),(G. Zhao, Huang, Taini, Li, & PietikäInen, 2011),(Szwoch & Pieniążek, 2015),(Gunawan, 2015) defines the latest model FER strategies and critical benefits.

Table 2.6 Conventional FER approaches and their benefits.

Database	Emotions	Visual Features	Decision Method	Reference
CK+ (Lucey, 2010)	Seven-emotions,17 AUs detected	Histogram of gradients (HoG)	A linear SVM	InfraFace (Yacoob & Davis, 1996)

CK+ (Lucey, 2010)	Seven emotions	Between the continuous frames, there is the displacement of the landmarks	Multi-class AdaBoost, SVM	Ghimiree and Lee (Ghimire & Lee, 2013)
CK+ (Lucey, 2010),JAFFE (M. J. Lyons, Akamatsu, Kamachi, Gyoba, & Budynek, 1998)	Six prototypical emotions	To select the localized feature from (SWLDA) the Stepwise linear discriminant analysis has been used from the expression	Six prototypical emotions	Stepwise approach (Owusu et al., 2014)
CE (Du et al., 2014)	Seven emotions and 22 compound emotions	Gabor filter has been used to define each fiducials point	Subclass discriminant analysis with Nearest mean classifier Kernel	Compound emotion (Du et al., 2014)
CK+ (Lucey, 2010)	Seven emotions	The landmarks displacement has been set by ASM that took place between landmarks	SVM	Real time mobile (Suk & Prabhakaran, 2014)
CK+ (Lucey, 2010), CE (Du et al., 2014), DISFA (Mavadati, Mahoor, Bartlett, Trinh, & Cohn, 2013)	23 essential and compound emotions	Euclidian distances of landmarks that have been normalized, Landmarks angels and landmarks points centered by Gabour filter	Discriminant analysis with kernel sub-class	Emotinet (Fabian Benitez-Quiroz et al., 2016)
Self-generated	Six emotions	Histogram of a face image and Local Binary Pattern (LBP)	PCA	Global Feature (G. Zhao & Pietikainen, 2007)
BU-3DFE (L. Yin et al., 2006)	Six prototypical emotions	3D patch shape and curve shape by analyzing the shape of the curve to the shape of the patch	Mutiboosting and SVM	3D Facial Expressions (Aleksic & Katsaggelos, 2006)

2.9 Deep learning-based FER

Much development has lately been made in the field of deep-learning systems. Deep learning is used in the world of information sciences, such as CNN and RNN. These algorithms have a wide variety of applications, including operations for attribute extraction, identification, and surveillance. Such algorithms have a wide variety of uses, including operations to pick, arrange, and identify attributes. As shown in Figure 2.14, CNN includes three different layers: a convolution sheet, a maximum pooling sheet, and layers that eventually are connected. In combination with a series of filter bench, images or feature maps are taken as their source and slide-windows these inputs to create maps with a facial image spatial structure. In a table of operation, the weights of coevolutionary filters are swapped, and the feature map inputs are locally connected. Secondly, sub-sampling layers reduce the representation of spatial resolution by combining or maxing out the information provided by the function map to reduce their dimensions to the minimum and thus ignore minor changes and geometric distortions. On the overall initial picture, the last connected layers of the Convolutional Neural Network system determine the class scores. Most deep learning methods have modified the CNN for AU recognition directly.

The CNN modeling program was used by (Breuer & Kimmel, 2017) to describe a model that was taught using different FER datasets and to show the capacity to classify networks that provide emotional training in both data sets and actual FER activities. The first derives spatial existence from picture collections, while the second derives spatial structure from landmarks that are temporal facial landmarks. These two versions are fused using a modern convergence method to improve facial speech recognition efficiency.

Wide Field and Multi-label Learning (DRML) suggested by (K. Zhao, Chu, & Zhang, 2016), a single approach for an extensive network. DRML uses feed functions to activate critical facial parts and captures qualified facial structural information. The whole network is trainable from end to end, which immediately reveals correct representations of fundamental differences within a local region. Many techniques have changed CNN for FER use specifically, as we stated in our post. But because the time variations in the facial components of a CNN are not reflection methods, the latest, hybrid technology is created that combines the spatial characteristics of a single CNN frame with the temporal dimensions of a consecutive frame and the long-lasting memory of the facial components. LSTM is a specific type of RNN able to recognize LSTMs that are long-term dependence to overcome the short-term memory problem. LSTM has a chain-like structure, while various models arranged differently, as seen in Figure 2.14. In most recurring neural networks, four neural network replication modules built (Olah, 2017)

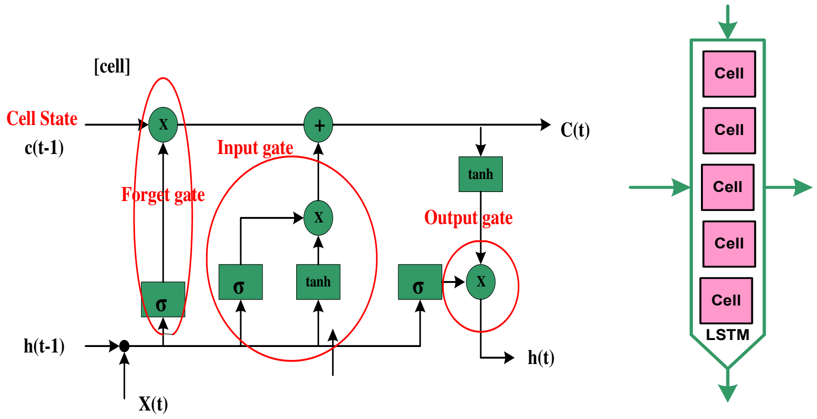
a. As shown in Figure 2.14, the cell status is a line across the top of the diagram, which is horizontal. The cell state can be omitted or inserted by LSTM.

b. The new data you want to save in the cell state is placed in a forgotten Gate Layer.

c. A gate layer input is used to define the cell and its values, that is changed.

d. An output gate layer gives the cell state-based output.

LSTM is a particular RNN category that can research long-term dependence. The LSTM or RNN approach for sequential picture processing has two benefits over different approaches. Second, when paired with other systems, such as CNN, LSTM models are straightforward concerning delicate end-to-end balance. Secondly, an LSTM accepts all inputs and outputs of defined duration (Donahue et al., 2015),(the Chu, De la Torre, & Cohn, 2017) proposed an AU algorithm for multi-level facial recognition that combines spatial as well as temporal characteristics.



a) b)

Figure 2.14: LSTM basic structure (Olah, 2017)

a).Single LSTM cell b). LSTM cascaded cells

Second, the spatial representations built using a CNN that is capable of minimizing inequalities created by handmade descriptors (e.g., HoG, Gabor). In addition to these representations, LSTMs stacked to model temporal dependencies irrespective of the video series data length. To create a 12 AU per plate, CNNs, and LSTM outputs are also connected to the fusion network.

(Hasani & Mahoor, 2017a) have presented a video sequence of specific objects which collects spatial connection and temporal connections within facial images for 3D Inception-ResNet architecture, accompanied by the LSTM unit. The input of this network is only facial icons that emphasize that face elements are essential and not facial areas that are not influenced by facial expressions.The recurrent network used by (Graves, Mayer, Wimmer, Schmidhuber, & Radig, 2008) clarified the temporal interactions during classification on image sequences. Such research studies have shown that in laboratory studies, a two-way network offers substantially better performance than a one-way LSTM (Bidirectional LSTM and Unidirectional LSTM).

Table 2.7 CNN and LSTM based FER approaches

Reference	Recognition Algorithm	Emotions Analyzed	Database
Hybrid CNN-RNN (Ebrahimi Kahou et al., 2015)	Temporal averaging for aggregation and Hybrid RNN-CNN for the propagation of information	Seven emotions	EmotiW (Ng, Nguyen, Vonikakis, & Winkler, 2015)
(D. H. Kim, Baddar, Jang, & Ro, 2017)	CNN has used for understanding the spatial image characteristics of the frames that show the expression state, and then LSTM understands the temporal characteristics	Six emotions	CASME II (Yan et al., 2014)
(Breuer & Kimmel, 2017)	CNN-based feature extraction	Eight emotions, 50 AUs detection	CK+ (Lucey, 2010)Nova Emotions (S. Zhang et al., 2012b)

Join Fine-Tunning (Jung et al., 2015)	CNN has used for appearance-based features that are temporal in nature as well as for facial landmark points.	Seven emotions	CK+ (Lucey, 2010)
DRML (K. Zhao et al., 2016)	Feedforward network and learning of weights	eight AUs for DISFA and 12 AUs for BP4D	DISFA (Mavadati et al., 2013) BP4D (Yan et al., 2014)
Multi-level AU (Chu et al., 2017)	Extraction of spatial representations via CNN and LSTMs for temporal dependencies	12 AU detection	BP4D (Yan et al., 2014)

(Jain, Zhang, & Huang, 2017) suggested (MAOP-DL) a deep learning model based on multi angel pattern learning approach for solving the question of abrupt lighting shifts and determining the right orientation for the set of features utilizing appropriate multi-angle configurations. This method first subtracts the context and isolates the focus from the pictures, then excludes the main elements related to the trends and facial points of the picture. The related features then omitted at random, and an LSTM-CNN is used to determine the right facial expression file. Unlike traditional methods, in-depth learning-based strategies commonly validated by experts in the deep neural network who evaluate functionality and classifiers. For the time the profound learning algorithm is assumed, it is, therefore, necessary to predict machine uncertainty. Table 2.7 shows quick research on various methods based on one CNN alone or combination of LSTM and CNN combination. Although FER approaches to deep learning have achieved tremendous success in experimental assessment, there is still a range of conditions that merit further consideration:

- Knowledge needs considerable dataset and enormous processing capacity because the system becomes deeper and deeper.
- A massive number of datasets that are collected manually are required

- There is a great deal of memory required, so it takes time for both to practice.
- Memory and technological complexity converted deep learning as inappropriate for deployment on resource-constrained web platforms and required significant ability and practice to obtain specific high-end parameters, such as the speed of learning, coevolutionary kernel filters, and other layers. These are dependencies based hyperparameters, which makes it exceptionally costly to turn around. While they work very well with various uses, there is still a shortage of strong CNN philosophy, which implies that consumers don't know why or how they function.

2.10 Deep Facial Expression Recognition

In this section, we discuss the three main phases of the automated deep FER, i.e., pre-processing, depth, and depth ranking. At each point, we review the widely used algorithms and recommend the current best practices.

2.10.1 Pre-processing

Some patterns for unconstrained situations, such as various environments, illuminations, and head locations, are unrelated to facial movements. Pre-processing is, therefore, necessary if the visual semantic information conveyed by the face is arranged and transformed before the deep neural network is prepared to acquire functional properties.

2.10.2 Face alignment

Face alignment in the identification activities is a typical pre-processing stage, shared with several photos. We list several well-known methods and strategies that are commonly available and used in deep FER. The first step is to identify the target, then eliminate all context and non-face regions, providing a sequence of training outcomes. Viola-Jones (V&J) facial sensor (Viola & Jones, 2001) is a robust and computational facial sensing tool that is commonly used and used. Although face recognition is the only vital technique to allow practical learning, the FER performance can be significantly improved through coordination with the more facial orientation of recognized landmarks. This move is essential since there may be significant variance in the in-plane and face scale rotation. Table 2.8 looks at innovative algorithms for facial recognition that are commonly utilized in deep FER and apply accuracy and performance to them. AAM is a traditional, optimization-based generative model that expected holistic appearance-based parameters of the face and global forms. The MOT and DRMF hierarchical model uses partial methods of unequal frameworks describing the identity of representations of the space world around the location. Some discriminative models, on the other hand, simply utilize a regression function cascaded to assign the topic position to famous places and got strengthened. Small networks have mainly been used lately for face synchronization. Cascaded CNN is the previous study which concededly forecasts

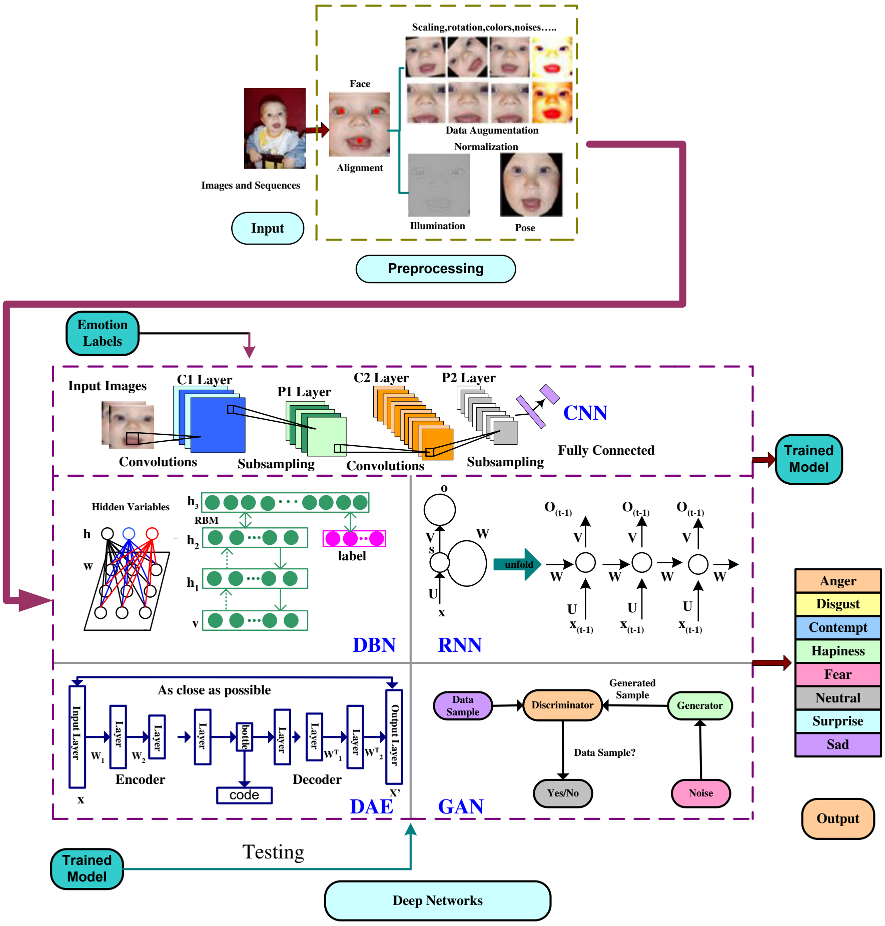


Figure 2.15.The primary pipeline of the deep facial recognition system

Landmarks. On this basis, the Tasks-Constrained Deep Convolution Network (TCDCN) (Zhanpeng Zhang, Luo, Loy, & Tang, 2014) and Multi-Task CNN (MTCNN) are continuing to exploit multi-task learning for a performance boost. The most successful and state-of-the-art approaches to face alignment were cascaded regression due to its quick pace and precision. By using only one detector for facial coordination, several approaches suggested integrating multiple markers to help approximate the mark by detecting faces in complex, unconstrained conditions. To supplement one another, Yu et al. (Zhiding Yu & Zhang, 2015) concatenated three different indexes of facial expressions. Kim et al. (B.-K. Kim, Lee, Roh, & Lee, 2015) considered several references (the initial image and the histogram).

Table 2.8: Summary of different types of face alignment detector that are used in DEEP FER model

Type		**# Point**	**Real-time**	**Speed**	**Performance**	**Used in**
Holistic	AAM (T. Cootes, Edwards, & Taylor, 1998)	68	X	fair	Poor generalization	(N. Zeng et al., 2018)
Part-based	MoT (Zhu & Ramanan, 2012)	39/68	X	Slow/Fast	Good	(Kahou et al., 2013),(Devries, Biswaranjan, & Taylor, 2014),(Shin, Kim, & Kwon, 2016)
	DRMF (Asthana, Zafeiriou, Cheng, & Pantic, 2013)	66	X			(Shin et al., 2016),(Meng, Liu, Cai, Han, & Tong, 2017)
Cascaded regression	SDM (Xiong & De la Torre, 2013)	49	✓	Fast/Very Fast	Good/ Very Good	(Ng et al., 2015)
	3000fps (Ren, Cao,	68	✓			(Hasani &

	Wei, & Sun, 2014)					Mahoor, 2017b)
	Incremental (Asthana, Zafeiriou, Cheng, & Pantic, 2014)	49	✓			(D. H. Kim et al., 2017)
Deep learning	Cascaded CNN (Yi Sun, Wang, & Tang, 2013)	5	✓	Fast	Good/ Very Good	(Kaihao Zhang, Huang, Du, & Wang, 2017)
	MTCNN (Kaipeng Zhang, Zhang, Li, & Qiao, 2016)	5	✓			(Zhenbo Yu, Liu, Liu, & Deng, 2018; Zhenbo Yu, Liu, & Liu, 2018)

2.10.3 Data augmentation

Deep neural networks need adequate training for the generalization of a particular recognition function. Nonetheless, the most commonly available FER databases do not have adequate volume of the image equalized) and various facial recognition trends (V&J(Viola & Jones, 2001), and MoT and maximum trust in the interface has been selected for picture training. Thus, the increased data is a crucial step for the creation of deeper FER. Throughout the test process, the input images are randomly taken from 4 corners and the middle of the file and then rotated horizontally so that 10X more enormous datasets as compared to the original one can be obtained. The analysis follows two conventional prediction methods: either for predicting the middle part of the face (e.g.(X. Liu, Vijaya Kumar, You, & Jia, 2017) but also calculate the predictive value of each of the 10 crops (e.g.(Wright et al., 2008),(Levi & Hassner, 2015)).

In addition to the regular increase in the volume data, multiple offline data collection operations were conceived to extend data in terms of both size and variety. Because of the development of elementary on - the-fly technologies, related offline data increasing practices were also planned to expand data on both scale and complexity. Spontaneous movements and transitions, such as rotating, spinning, skewing, moving, movement, comparison, and light jittering, are the most commonly used techniques. For, for example, regular noise models, salt & pepper, and speckle noise (Pitaloka, Wulandari, Basaruddin, & Liliana, 2017), as well as Gaussian noise(Lopes, de Aguiar, De Souza, & Oliveira-Santos, 2017),(Zavarez, Berriel, & Oliveira-Santos, 2017) used to improve the sample size. Subsequently, the through pixel saturation value (HSV color space components S and V) is adjusted for the contrast transition data lift. Numerous activity variants may provide more unsightly examples of testing and allow the network more resilient to deviated and distorted pictures. In the (W. Li, Li, Su, & Zhu, 2015) article, the authors introduced five picture appearance filters and 6 matrices for refined transformation formalized in minor geometrical transformations of the identity matrix. (Disk, average, Gaussian, unsharp and motion filters). To generate randomly varied images in the form of rotation, skew, and height, a more complex affine transform matrix was proposed in (Zhiding Yu & Zhang, 2015).

Deep learning techniques can also be used for improving efficiency. For example, in (Abbasnejad et al., 2017), a synthetic data generation unit was designed to secretively generate faces with varying expressive saturations using a 3D convolutionary neural network (CNN). It may also be feasible to expand the opposing generative network (GAN)(I. Goodfellow et al., 2014) to increase data by producing various appearance-distinctive postures and motions.

2.10.4 Face normalization

Differences in lighting and head position cause significant changes to the photos and thereby reduce the FER efficiency. So we incorporate two different strategies of normalizing the face to improve these variations: normalizing the light and standardizing (formalizing) behaviors.

2.10.4.1 Illumination normalization: Even in different photographs from the same individual with the same word, lighting, and contrast can vary, especially when the conditions are limited, resulting in wide variances intra-class. For normalization of illumination, other commonly used light-standardization algorithms, including IS-driven normalization, DCT-based standardization (W. Chen, Er, & Wu, 2006), and Gaussian difference (DoG) have evaluated. To eliminate lighting normalization (Junnan Li & Lam, 2015), homomorphic standardization-based filters stating that they produce the most successful outcomes of all the other approaches. Similar studies have shown that equalization of histograms and light normalization leads to a better output of the face recognition, as opposed to the standardization of light. Therefore, many studies of profound FER (e.g. (Zhiding Yu & Zhang, 2015),(Pitaloka et al., 2017), (Ebrahimi Kahou et al., 2015) use histogram equalization to improve pre-processing of global contrasts. This method is successful when the backdrop and the foreground are close to luminosity. The direct application of histogram equalization can still overemphasize the contrast among spaces. (Kuo, Lai, & Sarkis, 2018) proposed to combine histogram equalization with linear mapping, a weighted summation method to resolve a problem. In (Pitaloka et al., 2017), three different methods: global comparison standardization (GCN), position equalization, and histogram equalization. For the training and test phases GCN and histogram equalization to achieve maximum efficiency have been registered.

2.10.4.2 Pose normalization: Another problem that is intractable in unconstrained settings is the substantial variation in posture. Several studies utilized pose standardization techniques for giving FER frontal facial views (e.g., (Yao et al., 2016),(Yao et al., 2016). The most popular of which was proposed by Hassner et an l(Hassner, Harel, Paz, & Enbar, 2015). In particular, a reference model, i.e., 3D textured and is familiar to the faces produced after facial characteristics identified, so that identity characters approximated reliably. Then, each picture with the reference coordinate scheme synthesized with the initially formalized look. Furthermore, Sagonas et al. [developed an efficient mathematical method for simultaneous recognition of landmarks and transition of facial positions utilizing only the frontal pictures. Many deep GAN-based models for the synthesis of frontal view (e.g., FF-GAN(X. Yin, Yu, Sohn, Liu, & Chandraker, 2017), TP-GAN(R. Huang, Zhang, Li, & He, 2017) and DR-GAN(Tran, Yin, & Liu, 2017) have recently been suggested and enabling tracking of the tests.

2.11 Deep networks for feature learning

Deep learning has been a burning topic in recent years for researchers, and in some implementations, it has generated state-of-the-art success (Deng & Yu, 2014) Deep learning is built with hierarchical structures, numerous nonlinear changes, and representations to explore high-level abstractions. In this segment, we are currently discussing more profound FER education techniques.

2.11.1 Convolution neural network (CNN)

In many computer science systems like FER, CNN was commonly used. In the starting of the XXI century numerous research in the FER field (Fasel, 2002b), (Fasel, 2002a) revealed that in cases of previously unknown face-setting variations (Fasel, 2002b) the CNN is robust to change its location and settings and works more strongly than the multilayer perceptron (MLP) to overcome the subject independence issues and the location, rotation, and size-I invariance by using the CNN. The coevolutionary layer has a collection of learning filters that converge across the whole image and create various particular types of activation maps. Three key advantages lie in the convolution method: spatial communications that identify similarities between the neighboring pixels; weight sharing in a single characteristic chart which increasingly decreased the count of parameters to study; and the invariance of shift into the position of an object. The grouping layer enters the aggregation layer, which decreased the feature maps in terms of its spatial size and the network measurement costs. Average bundling and peak bundling are the two most popular nonlinear methods for the position invariance. Usually, the layer which is connected mainly placed at the end of the network so as the entire layer's neurons are entirely exposed to prior layer activation and so 2D function maps can be transformed to 1D functional map to improve viewing and gradients.

Table 2.9 lists the specifications and features of various rising CNN FER versions. There is also some well-known influenced software, in addition to these networks. In (B. Sun et al., 2015),(B. Sun, Li, Zhou, & He, 2016), area dependent CNN (R-CNN used for the learning of FER functions. Faster R-CNN (Ren et al., 2014)has been used in (Jiaxing Li et al., 2017)to define facial words with high-quality ideas being created. Ji et al. proposed that 3D CNN collect movement information stored in some of the neighboring frames to detect activity for 3D-convolution. Tran et al. (Fan, Lu, Li, & Liu, 2016) proposed the C3D, which uses large-scale, directed 3D convolution training datasets to know the features of space/time. This network for FER, involving image sequences, was used by many related experiments (for example, (Fan et al., 2016; Tran et al., 2017)

Table 2.9 CNN Models comparison

(DA=Data Augmentation, BN-Batch Normalization)

	Alex Net	**VGGNet**	**GoogleNet**	**ResNet**
Year	2012	2014	2014	2015
# of layers	5+3	13/16 + 3	21+1	151+1
Kernel size*	11, 5, 3	3	7, 1, 3, 5	7, 1, 3, 5
DA	✓	✓	✓	✓
Dropout	✓	✓	✓	✓
Inception	X	X	✓	X
BN	X	X	X	✓

2.11.2 Deep belief network (DBN)

Hinton et al. (Hinton & Sejnowski, 1986) suggested DBN as a graphic model that correctly reflects training data in the hierarchy. A stack of small Boltzmann (RBM) machines (Hinton, 2012) stochastic genital models of two-layer, consisting of a hidden and visible layer unit, are the basic DBN. These two strata form a two-part axis in an RBM without side intercourse. Higher layer units are trained in DBN, except for the top two layers that have undirected connections with the concentrations in the lower layers of the neighboring layers. Pre-training and fine-tuning(Hinton, 2012) are two steps in the DBN planning. First, a plan that is a smart plan for layer by layer and is efficient too (Bengio, Lamblin, Popovici, & Larochelle, 2007) unattended configures the deeper networks, which can somehow prevent bad, optimal local results without the need for many labeling details. For this procedure, RBMs are used to estimate the approximate gradient for DBN log-like structure by contrasting differences (Hinton, 2002). The network parameters and target output are then changed to a particular power downhill route.

2.11.3 Deep autoencoder (DAE)

DAE was first created in (Hinton & Salakhutdinov, 2006) for awareness of valid coding of dimensionality reductions. The DAE is intended to restore its inputs by increasing a reconstruction error, which is designed to estimate the target value, relative to the networks as described above. DAE variations occur, such as denoising self-encoder (DAE(Vincent, Larochelle, Lajoie, Bengio, & Manzagol, 2010) The sparse auto-encoder network (Le, 2013) restores the original information untouched from partially corrupted data, which increases the sparsity of the representation of the learned element. The Convolutional Auto Encoder (CAE1) (Masci, Meier, Cireşan, & Schmidhuber, 2011), that uses convolutionary along with potentially pooling lays mainly for hidden network layers and the Auto Encoder Variance(VAE) (Kingma & Welling, 2013) which is graphic design for certain types of latent variables, for the purpose of building co-defector. The CAE2 (Rifai, Vincent, Muller, Glorot, & Bengio, 2011) implements activity-dependent normalization, which contributes to locally invariant feature characteristics.

2.11.4 Recurrent neural network (RNN)

RNN, a relationship-based model that collects time data, is best suited for the arbitrary data series. The RNNs have multiple edges that cover the next steps in time and provide the same parameters each point, compared to an in-depth neural network single feed preparation. The RNN is packed with classic transmission BPTT (Paul J Werbos, 1990) Long-Short-Term Memory (LSTM) is an expert replica of regular RNN, which is used to overcome conventional

Hochreiter & Schmidhuber-built RNN gradient loss and bursting problems (Hochreiter & Schmidhuber, 1997). Three gates track and regulate the LSTM cell status, an input door to activate or block the input cell state signal, and an output gate that does or does not affect other neurons in the cell condition. Through the integration of these three frames, the LSTM modeling of longer-term dependents in a sequence is widely used for video expression recognition.

2.12 Facial expression classification

After assessing the deep characteristics, FER is to assign a specific face in one of the fundamental emotional categories. In comparison to traditional methods, deep networks can run end-to-end FER with particular abstraction procedures and functional classification. To track back propagation errors, the edge of the network is further fed by the loss layer; otherwise, the predictable likelihood for any sample is explicitly indicated. Softmax loss is the most common feature used in CNN, which decreases the relation between the projected class probability and the diffusion of the basic facts. Alternatively, (Tang 2013) showed the end-to-end job advantage by utilizing a linear vector support mechanism (SVM) that minimizes margin-based loss rather than cross-entropy. Similarly, (Dapogny & Bailly, 2018) deep-neural forest adaptation (NFs) was tested, NFs substituted the loss-layer Softmax, and high FER efficiency was achieved. Apart from the whole learning process, the deep neural network (mainly CNN) may also be used as a feature elimination method, and other separate classifiers, such as supporting vector machines or random forests, can be applied to derived representations showed that DCNN and Symmetrical Positive Definite multiplier (SPD) kernel determined Gaussian Covariance descriptors are more effective than the standard layer classification of Softmax. Facial expressions are used in various applications like suspicious activity detection for group and individual activities (Rastogi 2020).

This chapter discussed the specific methods and techniques used for facial emotion recognition, along with the pros and cons of various methods. This chapter helps to find the research gaps, and concluded as:

- Facial Recognition is likely to be affected by facial variation.
- Milieu plays a vital role in facial emotion recognition and may exert influence on the scalability and reliability of the FER System.
- Handful of systems are available that works well on low resolution images and videos in real-time.
- Fusion approach needs Deep Neural Networks as the parameters are enormous in number.
- HOG with SVM proved to best combination in the FER system when the image resolution is 48X48 or above.
- Computational cost is essential for real-time systems.

CHAPTER 3

DEEP LEARNING

The chapter discusses some key concepts that help further explain the fundamental nature of learning function; secondly, the artificial neurons, and the basic structure formed by neural networks. To begin with, deep neural networks are developed, and a summary of its key concepts is given with various architectures in use. Lastly, convolutionary neural networks are defined in more detail.

3.1 Introduction to Deep Learning

“Deep learning approaches are multi-tiered representational learning techniques, obtained by writing nonlinear but straightforward modules, which turn each of them into a more extensive, more complicated presentation. The central aspect of the deep study is that these are not human-made layers, through a comprehensive learning process; they are guided by data (LeCun, Bengio, & Hinton, 2015) They are human-made layers. Deep learning is artificial intelligence (AI) subfield, a mechanical learning subfield. For an overview of this relationship, see Figure 3.1. AI's main aim is to propose a range of modes and algorithms that can be overcome problems automatically, intuitively and virtually, which are otherwise extremely difficult for computers.

An excellent example of this kind of AI issue is perception and understanding of the content of an image. This function can do nothing - it's hard to do, but computers have proved extremely complicated. While AI involves a large number of automated research (inferences, planning, heuristics), there seems to be a particular interest in the identification of patterns and data learning. Artificial neural networks are algorithms for machine learning which rely on the structure and function of the brain that can learn from data. As we conclude, deep learning forms part of the ANN algorithm family, and, in most cases, both definitions used interchangeably. You may indeed wonder that there have been many different names and incarnations in the world of deep learning for more than sixty years, based on research patterns, accessible hardware and information systems, and standard solutions at the moment from an influential researcher. During the remainder of this chapter, we explore a short, more profound learning history that forms a robust neural network and how deep learning has been one of the great success stories in contemporary machinery and computer vision learning.

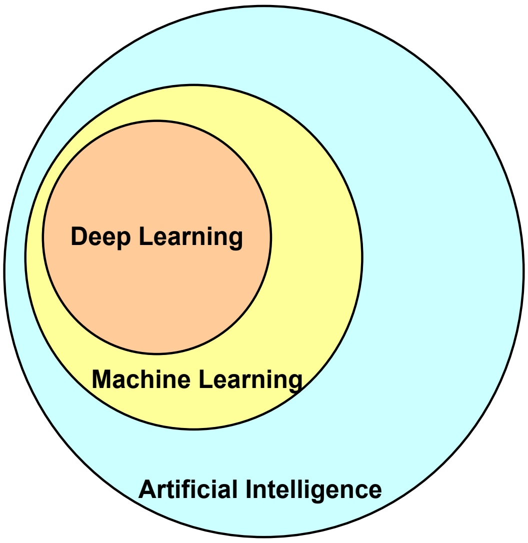


Figure 3.1: A Venn diagram depicting deep learning

A descriptive overview of neural networks and more in-depth learning is quite cumbersome. It might surprise you to know that there have been various shifts since the 1940s, such as cybernetics, connectivity, and the most common Artificial Neural Networks (ANNs). Even if the human brain is affected by the way its neurons interact, ANNs should not be interactive representations of the brain. Also, they are an example that illustrates how this behavior is replicated via artificial neural networks in contrast to a fundamental model of the brain. McCulloch and Pitts (L. Zhang & Zhang, 1999) published the first description of the neural network in 1943. It was a binary classification network that was able to identify two different groups based on specific data. The concern was to evaluate a particular input that must be calibrated manually to a human – obviously, this method is not ideal if a human operator is to interfere. Rosenblatt then produced this mini algorithm of the Perceptron (Rosenblatt, 1957) in the 1950s – this model automatically learned the necessary weights to identify a source of information (no human interaction needed). An example of a Perceptron design is given in Figure 3.2. The Stochastic Gradient Descent (SGD), which is still used in the formation of very deep neural networks today, was based on this automated training method. Perceptron-based techniques during this time are gaining importance.

Nevertheless, the neural science network was mostly stagnating for almost a decade in the 1969 publication of Minsky and Papert. Their research showed that a linear activation function (without depth) perceptron was unable to solve non-linear problems. The XOR dataset in Figure 3.3 shows a non-linear problem canonically. Take a second to persuade yourself that a single line cannot differentiate between the blue stars and the red circles.

The authors have claimed that (at that point) we did not have the necessary computational resources (these were correct in retrospect) to create large, deep neural networks. This single paper alone almost destroyed neural network studies. Fortunately, Werbos (1974) has an early downfall of the backpropagation and research algorithm (Paul John Werbos, 1994). Their analysis of the backpropagation algorithm allowed neural networks to train multi-layer feed (Minsky & Papert, 1969). For nonlinear supporting functions, researchers are now able to study the nonlinear features and try to solve the XOR issue completely. More analyses have demonstrated the ability of neural networks to estimate a continuous structure (Hornik, Stinchcombe, & White, 1989)(but not to ensure that the network learns the required parameters to describe a feature).

The back-propagation method is the cornerstone that allows us to learn from their mistakes and effectively train neural networks. The new iteration of neural networks, as we knew it, now is called deep learning. Because we have faster and more sophisticated equipment with more training data available, which removes profound knowledge from the previous incarnation, networks of many more hidden layers can be generated where basic concepts in the lower layers and advanced models learned in the higher network layers. The CNN (LeCun et al., 1989) is a quintessence example of a deeper applied learning to object learning, which applied to the recognition of handwriting character that automatically learns to sort patterns. Nearer network filters represent borders and corners, while higher layers used to distinguish between image groups. The Learning Machine hierarchy has (usually) collapsed into three unattended, unmonitored, and semi-attended groups of learning. In the managed case, both a list of inserts and target outputs provided with a machine learning algorithm. The algorithm then learns patterns to map data points automatically to the appropriate destination output. The directions regulated are like an instructor who makes you take a test. You work hard to provide the right answer to your exam because of your experience, but if you are incorrect, you'll be guided next year by your instructor for a more reliable and more experienced guess.

In a chaotic situation, algorithms attempt to discover discriminatory features randomly. In this situation, our student wants to answer and ask the same questions, even if the student does not know the right answer. Unauthorized learning is more a challenge than directed learning as image classification used for these image sets to evaluate models so that distinguishing patterns can be quickly recognized and connects input data to the real purpose. Previously we used hand-made features to quantify image quality – as is usual during departure study, we never use pixel intensity to feed our models.

3.2 Basics of Neural Network

Neural biological systems are primarily a collection of neurons, the basic building blocks connected via axon terminals, which enable neurons activation in a continuous signal path. A neural bio-network primarily consists of a neuron collection; the basic building blocks interconnected through axon terminals, which enables neurons to build a continuous signal path. Figure 3.4 shows a shallow ANN with no hidden layer.

3.2.1 The Artificial Neuron

As for Figure 3.2, the output of such an artificial neuron represented as:

$$y = f\ (\textstyle\sum_{n=0}^{N} W_n x_n \ + \theta) \tag{3.1}$$

Where xn refers to the nth neuron input, Wn refers to the weight analogous to that input, θ refers to the neuron bias/threshold parameter, and N refers to the number of neuron inputs. Besides, f (•) refers to the neuron's activation or transfer role, as set out in the section below. Each neuron in a network is connected to every subsequent neuron to which it is connected. This neuron can receive this feedback, and then the signal is being amplified by the related weight. The resulting output is paired with other adjacent neurons from the previous layer by moving them through the process of induction that causes this neuron to be activated. Activation feature: Activation functions are utilized to boost or strengthen the neural network according to its properties. Since the sum of the linear function still represents a linear function, it would still be comparable for the use of a single layer network of linear activation functions without weight sharing. Therefore, the use of a nonlinear activation mechanism is needed to benefit from multiple layers of the network. Rectifier functions offer an excellent solution to traditional non-linear control mechanisms such as the Hyperbolic Tangent Sigmoid. The main benefits of using this activation function are: (Glorot, Bordes, & Bengio, 2011):

- The gradient flow is nice via the active network paths.
- Sparsity is eventually implemented on the network (for example, random stimulation allows half the neurons activated by positive inputs). Some are the dissociation of information (strongly intertwined depictions, because a shift in feedback influences many images); stronger spatial divisibility (due to the fragmented depiction in a large area).
- The rate of computation is lower because there is no linear estimation of functions.

3.3 Deep Neural Networks

A deep neural network classification is a network between input and output layers that consist of higher than a single hidden layer. The multi-layered network scale of a neural network used before its combination with deep neural networks is not clearly defined. Figure 3.5 shows the Deep Neural network, which consists of two hidden layers. The incorporation of hidden neuron layers between the network would render it a universal approximator (Leshno, Lin, Pinkus, & Schocken, 1993), which ensures that because of the necessary parameters multi-layered neural networks are willing, due to the multi-layered input architecture (regardless of their choices for activation, (Hornik, 1991)to estimate of a continuous function of the more simplified subset. The more extensive the network, the higher the capability to be trained further, and more are the complex functions. Each layer tests the non-linear alteration of the previous layer, to enhance the network representational power.

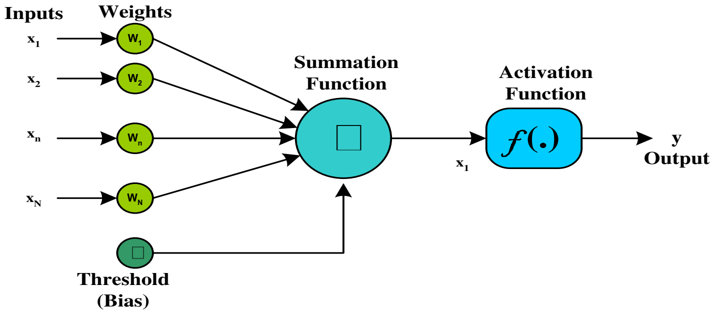


Figure 3.2: The Artificial Neuron

As has already been stated, an exponentially higher number of O(2n) neurons be characterized by a deep network with O(log $_n$) layer count that can express an n-bit parity feature with a flawed network with no hidden laying (Bengio & LeCun, 2007). Figure 3.3 shows a nonlinear, separable problem that Perceptron cannot solve is the data set XOR Exclusive OR). Allow one second to assume that it is challenging to draw a single line that separates the red stars from the green circles.

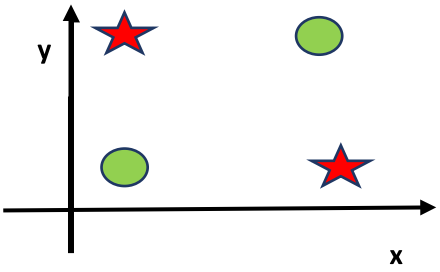


Figure 3.3: An example of a non-linear, separable problem.’

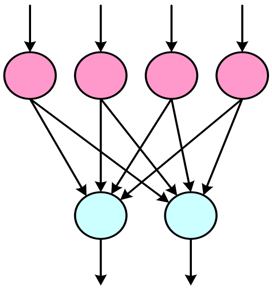


Figure 3.4: Shallow ANN with no hidden layer

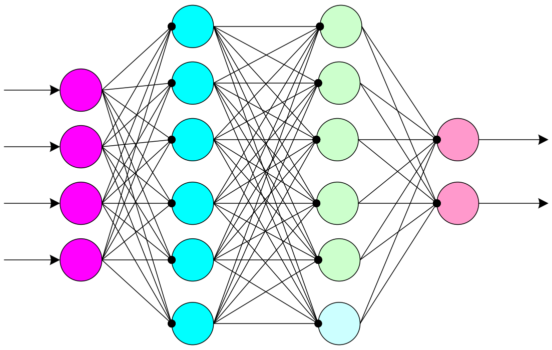


Figure 3.5: A Deep Neural Network with two hidden layers

But a fundamental problem in the formation of the deeper neural network was the need for extensive computer tools that were not available until recently. Although work is underway in this field, the recent availability of powerful GPUs was a significant step forward For GPUs training times for large networks were lowered by many orders of magnitude, as opposed to 16-core 12 GB CPUs for a given network (our tests showed a quicker 5000-core 6 GB GPU test of as many as 82 seconds).

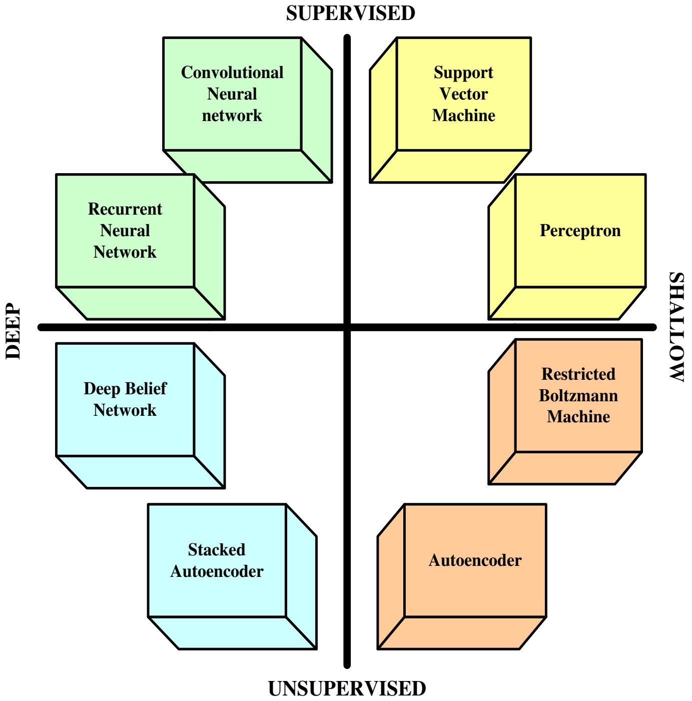


Figure 3.6: Types of Machine Learning Models, Neural Networks and their properties

3.4 Deep Network Architectures

A number of the Deep Neural Network architectures used in this section have their main features. Figure 3.6 shows a range, based on training and distance properties, of network classes (and others).

3.4.1 Stacked Autoencoders: Autoencoders are artificial neural networks that enable the application of ' blank ' data. Auto-encoder attempts to learn a conversion into a compact and distributed illustration for a particular set of inputs. Therefore, Autoencoders may be seen as dimensional (or failure in compression). One automated coder is a layer of information that corresponds to the raw data display, along with a hidden layer of encoding. Figure 3.7 depicts a stacked autoencoder that readily connects the outputs of the individual layer to the following layer of input. The encoding is usually conducted in an autoencoder, which minimizes the dissimilarity factor (reconstruction error) between the data and that of the hidden layer (using inverse weights).

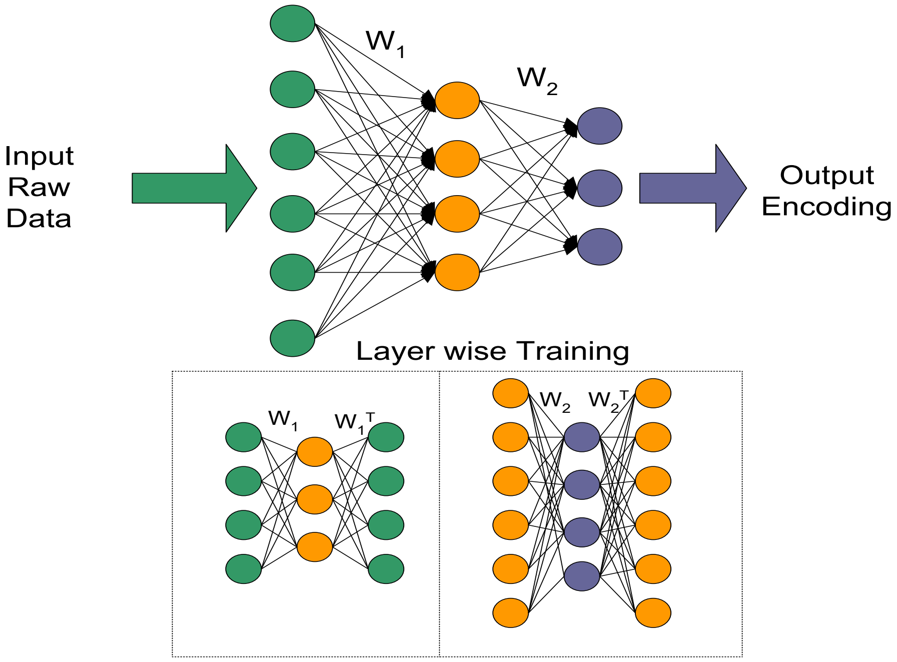


Figure 3.7: Stacked Autoencoder

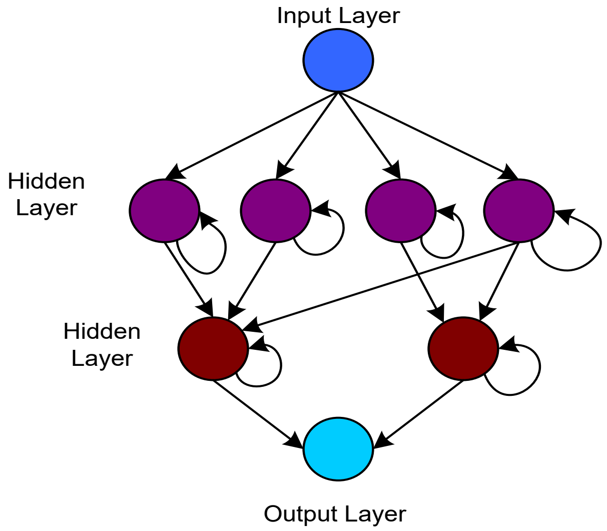


Figure 3.8: Recurrent Neural Network

One of the most valuable benefits of stacked automatic encoders: ease of use layer by layer (Bengio et al., 2007), which enables them to learn from unchecked input encoding.

3.4.2 Deep Belief Networks: The networks are based on the pure software modules known as the Restricted Boltzmann Machines (RBM) (Fischer & Igel, 2012) consists of two layers. Deep Belief Networks (and RBMs) are a core feature/benefit of being an energy-based probabilistic and generative (Hinton, 2009)model and, therefore, can be used to embed the neuronal network in other probabilistic models and to produce etiquette-based data using the neural nets. Such networks can also train unattended data in a layered fashion. Nonetheless, a direct descent of the gradient is insoluble, and therefore it is essential to estimate algorithms such as contrast differences (Carreira-Perpignan, 2005).

3.4.3 Recurrent Neural Networks: Current neural networks are networks built to include time-based encoding information, such as long-term and short-term storage (Hochreiter & Schmidhuber, 1997)in Figure 3.8. Neurons ' activation characteristics are based on recent input history. Weights of the relation among neurons are interpreted as some type of long-term memory, and their values depend on network inputs. This is done by how data processed over time in the network. These weights change gradually, particularly when compared to a short time activity memory. Figure 3.9 shows the feature map through a convolution layer, with a standard weight associated with the same color, and the basis of activation are blue color-neurons by creating a feature map.

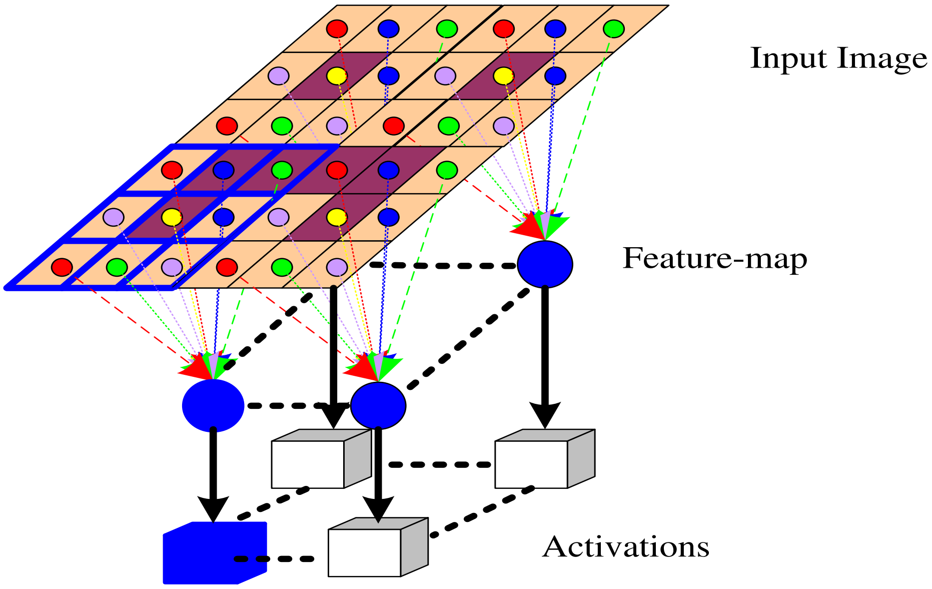


Figure 3.9: Feature map through a convolution layer

3.5 The depth of Deep Networks

From his 2016 speech, to quote Jeff Dean: ' If you hear deep learning, you just consider a vast, deep neural net. Deeply describes the average figure of layers, thus the popular term adopted in the press. It helps us to conceive of profound learning as broad neural networks with layers that slowly grow in size on each other. There is still no concrete response. The short reply is that there is no consensus among network depth experts to take a close look at. Now we have to examine the problem of the network type. CNN is a kind of deep learning algorithm by its very definition. But if we had just one CNN, is it a network that is not very shallow and is considered authoritative within the deep learning community?

1. Our lack of big, labeled training data sets

2. Too sluggish to train major neural networks is our computers

3. As a result of these problems, during the 1980s and 1990s, it was quite challenging to train a network that has got more than two hidden layers.

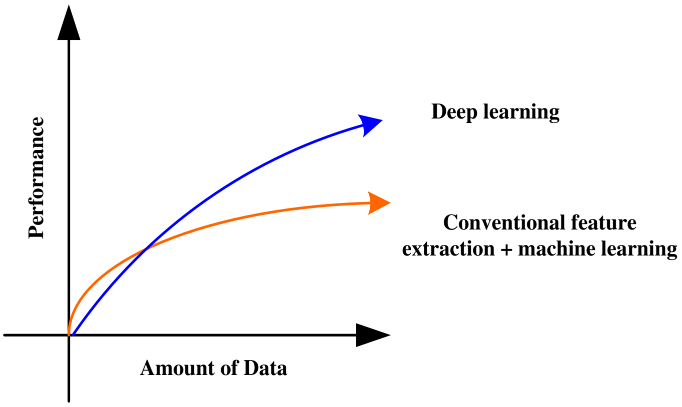
In reality, in his 2016 lecture, Deep Learning, Geoff Hinton shares the argument in which he addressed why the previous deep learning incarnations.

1. Our labeled data sets were smaller.
2. Our computers have been too sluggish for millions of times.
3. We have stupidly configured the weights of the network.
4. We have used the false nonlinearity function.

We now have:

1. Quicker computers
2. Hugely optimized equipment (GPUs, for example)
3. Big, labeled data sets with millions of pictures
4. A better understanding of the weight initialization and the function
5. Superior commencement and accepting of why earlier nonlinear functions have languished the research.

The classification reliability augments as the size of the network enlarges. It varies from conventional algorithms of machine learning in which we have a definite plateau, even as the training data available increases. Figure 3.10 provides an example of what researchers have to offer for deep learning, based on a picture based on Andrew Ng's lecture in 2015. Increasing the training data gives a better neural network algorithm, thus making it more accurate. Due to the combination of increased precision and added knowledge, we are liable to correlate deep learning with large data sets. While using an in-depth learning program, you can find a solution to the supplied neural network by using the following thumb rule.



.

Figure 3.10: Data Vs. Performance in machine learning and deep learning.

3.6 Convolutional Neural Networks

1. Use different network designs as Convolutional Neural Networks or (LSTM) Long Short-Term Memory Networks.

2. You are doing deep learning if the network gaining depth is less than 2.

3. Is your network having a depth of > 10? If so, you think deeply. In the last sixty years, profound training has been the subject of various incarnations based on various educational institutions – yet each think tank focuses on the function and structure of the artificial neural network of the brain. Machine learning involves the use of artificial neural networks regardless of network size, width, or advanced network architecture. While the network, as mentioned above, architectures have a consistent approach, some types of problems, such as visually displayed details, are not good at addressing them. Because every neuron present in the layer below is linked to individual neurons, the weight of links for the input dimension proliferates. Often this leads to weak growth. The major downside of the network architectures is the absence of the spatial input structure (such as an image). The project requires the use of the network structural design to reduce the number of training constraints to meet these requirements by using the spatial properties of the data. A neural network is usually made up of a combination of filters that apply to all the spatially organized inputs of the map of the activated output (with a similar spatial structure). Figure 3.11 depicts the normalization of local contrast with a kernel size 2x2 over a 4x4 serial

input: The output matrix boxes represent outputs generated by the LCN kernels over inputs displayed in the same colored boxes.

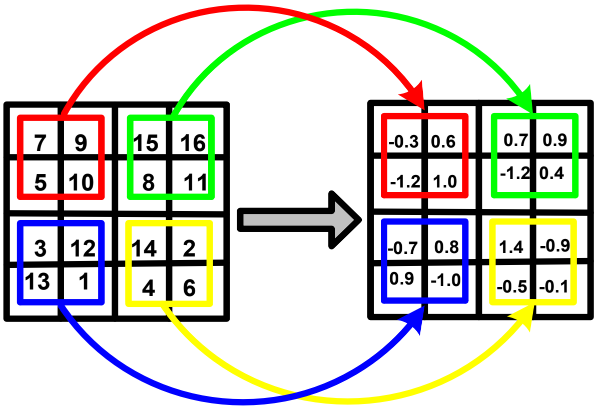
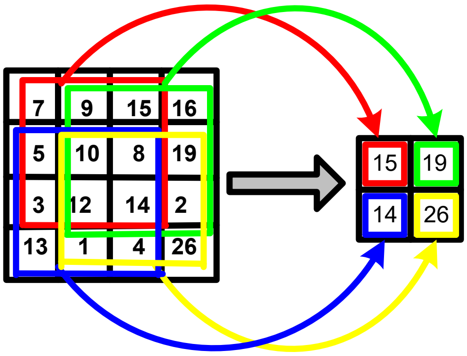


Figure 3.11: The operation of Local Contrast normalization



.

Figure 3.12: The max-pooling operation

Figure 3.12 depicts the max pooling operation with a kernel of size 3 × 3 and stride of 1 in along all dimensions of input of size 4×4: The boxes on the left side (of each color) display max-pooling kernels and the boxes on the right side show the maximum pooling outputs—the process characterized by using a filter at several points of the input matrix. When network inputs are shown in a double-dimensional matrix structure (for instance, images), the weight layer between a 2D input patch and a single neuron output is represented as a filter (also kernel). In multiple-input positions (it can overlap), these filters repeatedly used to generate outputs with a 2D activation matrix. The network, therefore, now understands input data spatially ordered.

Assume that a matrix which is 2-D in size I x J and the size of Î x Ĵ for convolutional layer translate into a two-dimensional input room, with a stride Î & J̃. This layer brings into being the size of the output $\frac{\mathrm{I}-\hat{\mathrm{I}}}{\tilde{\mathrm{I}}}$ x $\frac{\mathrm{J}-\hat{\mathrm{J}}}{\tilde{\mathrm{J}}}$. The neurons layers calculate their activations based on equation 3.1

$$yi,j = f\left(\sum_{\hat{\imath}=0}^{\hat{I}-1}\sum_{\hat{\jmath}=0}^{\hat{J}-1} K_{\hat{\imath},\hat{\jmath}} x_{i+\hat{\imath},j+\hat{j}}\right) \forall i \in [1,\frac{I-\hat{I}}{\tilde{I}}), j \in 1,\frac{J-\hat{J}}{\hat{J}}) \qquad (3.2)$$

Where the i^th^ row and j^th^ column in the input and output space are denoted by $x_{i,\ j,}$ and $y_{i,\ j}$, the primary feature of convolutionary layers is the combination of the filter weights at different points throughout the input.

3.6.1 Max-Pooling

The manufacture of a convolutionary layer conclusively indicates the continuation of functions defined in the input space in an explicit place. If the place of this function in the entry is translated, the layer activation rate will then always convert proportionally. The goal is to apply force to small input irregularities and give a degree of invariance to the filters. The bundling process usually involves the reading of a 2-D input patch and the creation of a single output calculated using the input patch feature. This spreads across other fields of information, preserving the entire data region. It is important to note that for a pooling process, no weights are necessary; this simply acts as a sub-sample step. Max pooling is a typical kind of pooling in which output compares with its input number. This would be the function calculated by using a given 2-D input patch X to define the output of a max-pooling procedure:

$$y = \max(X) \qquad (3.3)$$

It cannot be regarded as a place on the input room but rather as a function in the field of the input area. The overlapping kernel activity is displayed in Figure 3.12.

3.6.2 Local Contrast Normalization

Standardization of the local contrast is a means of enforcing rivalry between neighboring neuron activations. A nearby neuronal patch minimizes stimulation. Subtractive and divisive criteria combine this standardization: divide the outcomes by standard values deviations and remove the mean of the local patch from each value.

The following words can be described:

Contrast normalization can be carried out by a local input image of Size I x J

$$y_{i,j} = \frac{x_{ij} - \frac{1}{IJ}\sum_{\hat{\imath}\in I,\hat{\jmath}\in J} x_{\hat{\imath},\hat{\jmath}}}{\sqrt{\sum_{\tilde{\imath}\in I,\tilde{\jmath}\in J}\left(x_{\tilde{\imath},\tilde{\jmath}} - \frac{1}{IJ}\sum_{\bar{\imath}\in I,\bar{\jmath}\in J} x_{\bar{\imath},\bar{\jmath}}\right)^2}} \ \forall i \in I, j \in J \tag{3.4}$$

Where $x_{i,\, j,}$ and $y_{i,\, j}$ corresponds to the input and output situated in the input and output area at the ith row and jth column. The relevance of the non –overlapping kernels of such local contrast normalization is well depicted in Figure 3.11.

3.7 Edge Computing

Automatic expression recognition is a process of understanding human expressions from facial emotions. The real success behind the automatic facial emotion recognition is the Internet of Things (IoT) devices, edge computing, cloud computing, and next-generation networks. There are various applications in which IoT and edge computing have brought a substantial change in terms of high speed, low-cost transmission, mobility, and pervasiveness. It has seen that there is an exponential increase in the data that has been collected, and Big data has played an essential role in refining that data. Although the accuracy of the system, that can recognize this data is always a challenge. However, deep learning has proved to bring revolution in many systems that are capable of detecting and recognizing automatically.

3.7.1 Understanding Transfer Learning

Transfer learning is a concept that is dedicated to deep learning. As compared to traditional learning in which the system is designed and dedicated to a particular task, transfer learning is the process in which anyone can leverage weights, features, etc.

from a model that is pre-trained and can be used to train a new model which is even having lesser number of datasets.

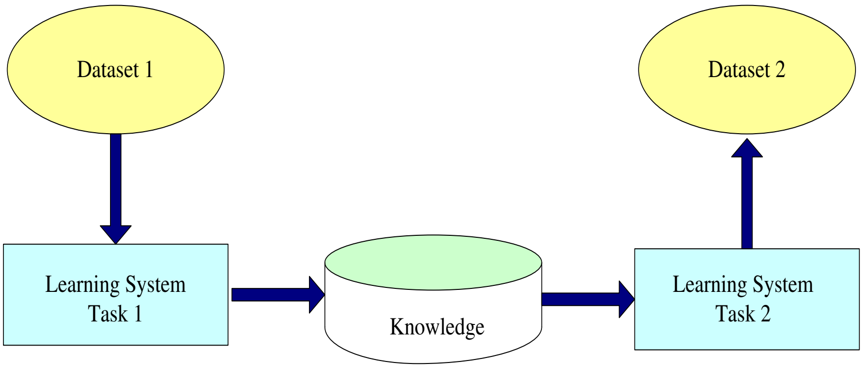


Figure 3.13. Transfer Learning

3.7.2 Transfer Learning for deep learning

Many misconceptions are popular in context with deep learning. The most popular one is that deep learning can't be done until and unless anyone is not having millions of labeled examples for any problem. The truth is that the unlabeled data can also be trained, and the training can be done on the nearby surrogate objective, and for those, the labels can be easily generated. The learned representation can also be transferred from any related task, as shown in Figure 3.13.

3.8 Deep Transfer Learning Strategies

In recent years deep learning has gained considerable attention and progress. Because of this, complex problems can be tackled with ease and yield amazing results. There are various deep learning networks that have performed well and became state of the art. These pre-trained networks became the basis of transfer learning, particularly in the context of deep transfer learning.

3.8.1 Off-the-shelf Pre-trained Models as Feature Extractors

Deep learning is the process in which different layers present in a network used for learning different features. These layers finally connected to the last layer from which the final output is achieved. So basically, we can utilize this layered architecture to extract the features without utilizing its final fully connected layer. The main question that arises here is that if these networks perform well in actual practice. The literature has proved that such networks are strong enough to perform well in different tasks.

3.8.2 Fine-Tuning Off-the-shelf Pre-Trained Models

This type of training consists of not only changing or replacing the final layers; preferably, it also includes the fine-tuning of previous layers and retraining them. Deep networks are those networks that are highly configurable, and that can be done by using various hyperparameters. The initial layers of the deep network are generally used to capture generic features while the later ones focus more on the specific task in hand. Following figure 3.14 shows the face recognition problem in which the initial layers are to learn very generic features, and the subsequent layers are learning task-specific features. Using the insight, individual weights can be fixed and used for retraining, and remaining can be fine-tuned as per our requirements. In such models, the knowledge has been utilized, and the state of the network is utilized as a starting point for retraining purposes. So, better performance can be achieved with such models in less training time.

3.9 Pre-Trained Models

One of the standard requirements of transfer learning is the models that can handle the task in hand. Fortunately, the world of deep learning believes in sharing. There are plenty of deep learning architectures that are available for people and research community by the developers and team of those networks, and those networks are state of the art networks. In deep learning, the most common and popular fields are computer vision and Natural language Processing. The models that pre-trained and available are huge in count in terms of weight and parameters that model achieves while training. All these pre-trained models are available in different forms, and the most popular deep learning Python library is Keras, via which the most popular models can be downloaded. Not only that, but one can also download various web sources, and the majority of them are open-sourced.

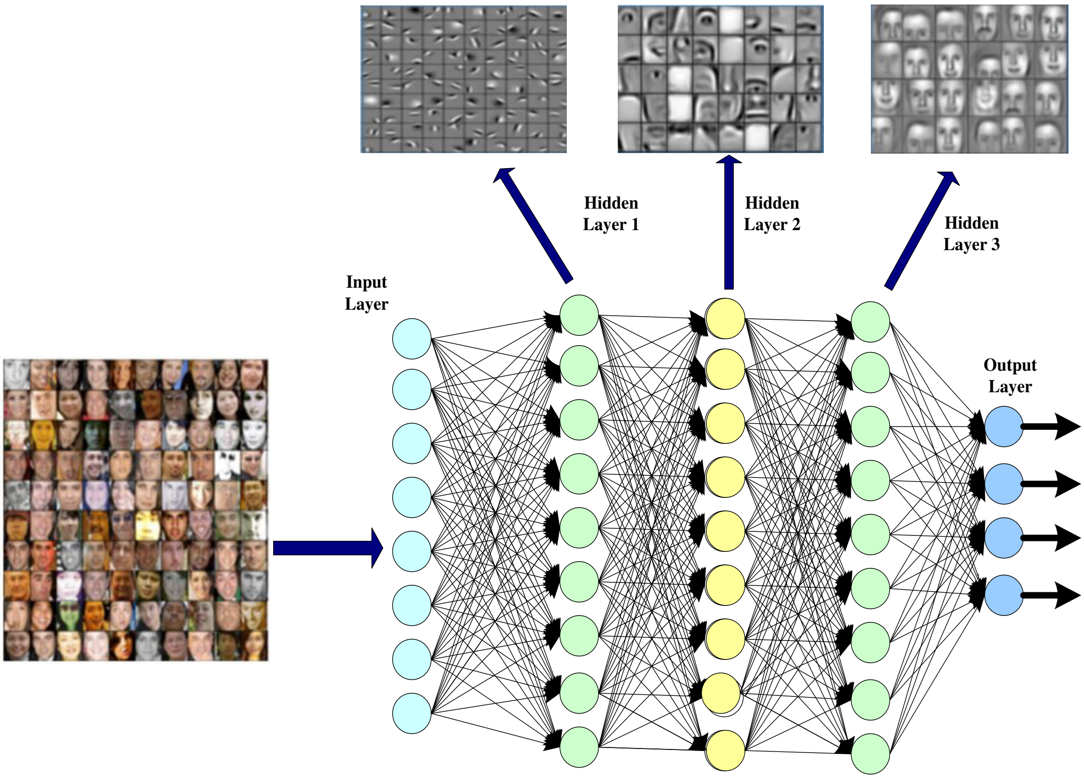


Figure 3.14. Representation of Deep neural networks learn hierarchal feature representations

Some popular models for computer vision are:

- VGG-16
- VGG-19
- Inception V3
- Xception
- RestNet-50

3.10 Types of Deep Transfer Learning

There are so many terms that are used interchangeably in the context of Deep Transfer Learning. At times it becomes challenging to identify the difference between transfer learning, multi-task learning, and domain adaptation. All these are used to resolve similar issues. So mainly, transfer learning is the general concept in which the target task is being resolved using the domain knowledge of source task.

3.10.1 Domain Adoption

Such learning is mainly related to the conditions where the target and the source domains are different $P(X_t)! =P(X_s)$.There is adrift in the segregation of data in the domains it can be the origin or final destination that needs tweaks for the implementation of transfer learning.

3.10.2 Domain Confusion

In particular, the re-iteration of multiple layers in deep learning of a various set of features. Instead of letting the model learn any representation, the emphasis is on the representation of both the domains to be as close as possible. The main idea is to increase and encourage the similarity.

3.10.3 Multitask Learning

This kind of model is a little different from the essence of transfer learning. In these, several tasks are learned all together without and differentiation between the targets and the source. In such a model's learners is having all information about multiple tasks, unlike transfer learning. The following Figure 3.15 depicts that.

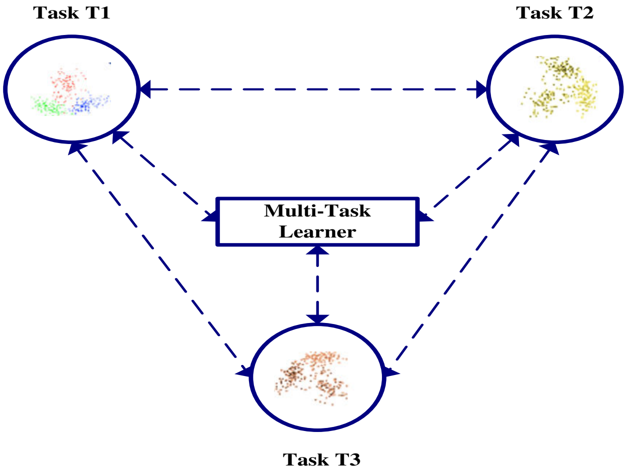


Figure 3.15: Depiction of Multitask learning where information received from all the tasks simultaneously

3.10.4 One-shot Learning

Whenever Deep learning comes into the picture, the first thing that strikes is massive data because of the essential nature of deep learning. The one-shot is the method in which is the copy of transfer learning, where it inferred to achieve the desired output based on only a few examples. Such a model is useful in the real-world scenario if it is not possible to label data for all the possible classes.

3.10.5 Zero-shot Learning

This type of model is another extreme variant of transfer learning. Such a model does not require any labeled examples to learn any task. The zero-shot learning model automatically makes intelligent adjustments during the training stage itself that will automatically and cleverly adjust itself to understand the unseen data as well.

This chapter explained the Deep Learning concepts in detail and the benefits of using Deep Learning in the FER system. Different layers of Deep network with their functionality Illustrated and the benefits of transfer learning mentioned in the end. This chapter concludes the benefits of deep learning over conventional machine learning algorithms when it comes to implementation with a humongous dataset of facial emotion images.

CHAPTER 4

RASPBERRY-PI

This chapter explains how the Raspberry Pi can be used for computer vision, including how deep learning, IoT, and edge computing are pushing innovation in embedded devices, both at the hardware and software level. Further, this chapter expands the use of RPi within the CV and DL fields, paving the way to build computer vision applications on the Raspberry Pi. The Raspberry Pi credit card-sized device, which is of low cost and connected with a standard mouse and a keyboard to a computer monitor or Television. It is a small, capable device that can explore the computers in languages like Scratch and Python at all ages and teaching how to program them."— The foundation of Raspberry pi of all the computer devices which have not only made innovation easier in recent ten years, only a minimal number, if any device is over the Raspberry Pi and only a mere $35, this handy board is a mini-computer which is similar to hardware on the desktop ten years ago.

People used this for fun, innovation, and research projects since the Raspberry Pi (RPi) was first released in 2012, including:

i. Creating a wireless print server

ii. Utilizing the Rpi as a media center

iii. Adding a controller to the Rpi and playing retro Atari, NES, SNES, etc. games via a software emulator

iv. Building a stop motion camera

...and the list goes on. At the core, the Raspberry Pi and its associated community have its roots in both:

- Practicality — anything designed with the Rpi in some capacity should be of use.

- Learning — The core ethos of the hacker is learning and learning. When using an Rpi, you should be pushing the limits of your understanding and enabling yourself to learn a new technique, method, or algorithm.

4.1 Use the Rpi for CV and DL

We will quickly explore the past of Raspberry Pi in the first part of this segment. I'll then discuss how computer vision can be applied to the Rpi, followed by how the current trends in the Internet of Things (IoT) and Edge Computing applications are helping drive innovation in embedded devices (both at the software and hardware level). We'll then wrap up by looking at coprocessor devices, such as Intel's Movidius NCS and Google's Coral USB Accelerator, as shown in Figure 4.2, and how they are facilitating state-of-the-art deep learning on the Rpi. Figure 4.1 shows . all in a device approximately the size of a credit card and under $35. Credit: Seed Studio

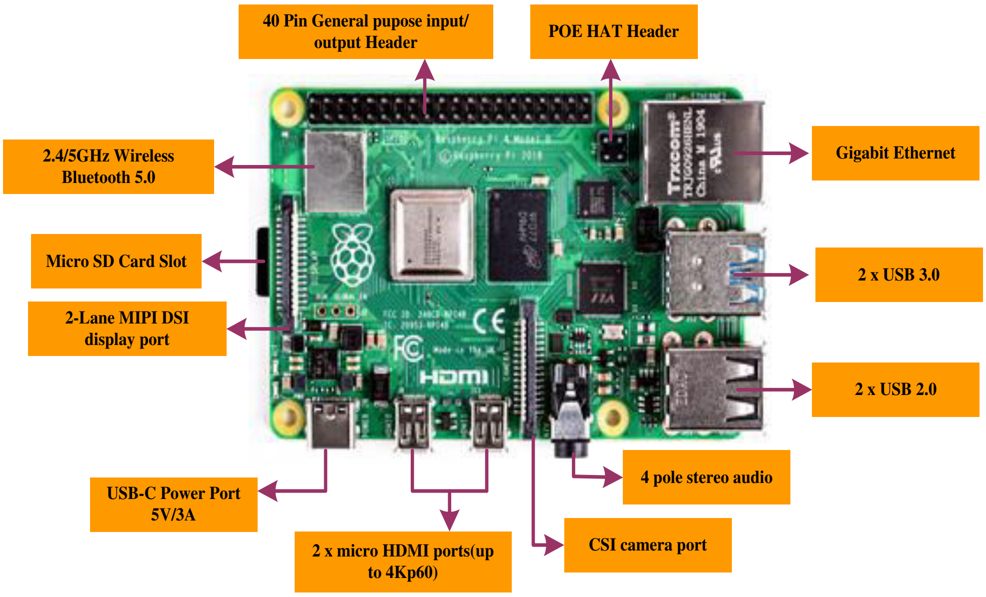

40 Pin General pupose input/ output Header
POE HAT Header
2.4/5GHz Wireless Bluetooth 5.0
Gigabit Ethernet
Micro SD Card Slot
2 x USB 3.0
2-Lane MIPI DSI display port
2 x USB 2.0
4 pole stereo audio
USB-C Power Port 5V/3A
CSI camera port
2 x micro HDMI ports(up to 4Kp60)

Figure 4.1: The Raspberry Pi 4

4.1.1 The Rpi for CV and DL Applications

The very first Raspberry Pi with a 700 MHz processor and 512 GB RAM released in 2012. The current iteration, Rpi 4B (Figure 4.1), has a quad-core processor of 1.5GHz

64-bit and RAM 1-4 GB (depending on the model) over the years. The Raspberry Pi has similar specs to desktop computers from a decade ago, meaning it’s still incredibly underpowered, especially compared to our current laptop/desktop devices— but why should we be interested? In the beginning, the field of computer vision can now effectively be performed on the Rpi from the research perspective of understanding algorithms, which were extraordinarily costly from the computer and could run on high-end machines only ten years ago. Also, from a practitioner’s viewpoint, we should look at this – the computer-view libraries are now ready to install, simple to use (as soon as you understand it), and optimized for algorithms. Effectively, the Raspberry Pi has brought computer vision to embedded devices, whether you are a hobbyist or an experienced practitioner in the field, and we are not limited to computer vision algorithms. Using coprocessor devices, such as the Movidius NCS or Google Coral Accelerator, we are now capable of deploying state-of-the-art deep neural networks to the Rpi as well! It is an incredibly exciting time to be involved in computer vision on embedded devices — the possibility for innovation is nearly endless.

4.1.2 Cheap, Affordable Hardware

Part of what makes the Raspberry Pi so attractive is the relatively cheap, affordable hardware. At the time of this writing, a Raspberry Pi 4 costs $35, making it a minimal investment for both:

Figure 4.. Hobbyists who wish to teach themselves new algorithms and build fun projects.

ii. Professionals who are creating products using the Rpi hardware. At only $35, the Rpi is well-positioned, enabling hobbyists to afford the hardware while providing enough value and computational horsepower for industry professionals to build production-level applications with it.

4.1.3 Computer Vision, Compiled Routines, and Python

It is no secret that computer vision algorithms can be computationally expensive. Typically, when a programmer needs to extract every last bit of performance out of a routine, they’ll ensure every variable, loop, and construct optimized, typically by implementing the method in C or C++ (or even dropping down to assembler). While compiled binaries are undoubtedly fast, the associated code takes significantly longer to write and is often harder to maintain. But on the other hand, languages such as Python, which tends to be easier to write and maintain, often suffer from slower code execution. Luckily deep learning, computer vision and machine learning, and libraries are now providing compiled packages. Libraries such as OpenCV, sci kit-learn, and others:

- Are implemented directly in C/C++or provide compiled Cython optimized functions (Python-like functions with C-like performance).
- Provide Python bindings to interact with the compiled functions.

Effectively, this combination of compiled routines and Python bindings gives us the best of both worlds. We can leverage the speed of a compiled function while at the same time maintaining the ease of coding with Python.

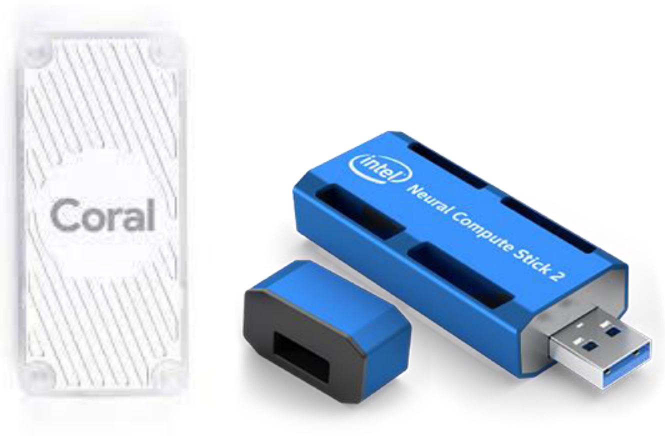


Figure 4.. b)

Figure 4.2. Coprocessors for Raspberry-Pi a). Google coral TPU USB Accelerator b). Neural compute stick 2

4.1.4 The Resurgence of Deep Learning

As the latest resurgence in deep learning has created additional interest in an embedded device, such as the Raspberry Pi. Deep learning algorithms are super powerful, demonstrating unprecedented performance in jobs such as image recognition, object detection, and image segmentation. The problem is that deep learning algorithms are incredibly computationally expensive, making them challenging to run on embedded devices. But just as computer vision libraries are making it easier for CV applied to the Rpi, the same is right with deep learning. Libraries such as Tensor Flow Lite enable deep learning practitioners for training a model on a custom dataset, optimize it, and then deploy it to resource-constrained devices as the Rpi, obtaining faster inference.

4.1.5 IoT and Edge Computing

The Raspberry Pi is often used for Internet of Things applications. IoT and edge computing is the most popular and cutting-edge technology nowadays. The majority of the applications that were not able to implement in real-time are now made with the help of edge computing.

4.1.6 The Rise of Coprocessor Devices

As mentioned earlier, deep learning algorithms are computationally expensive, which is a big problem on the resource-constrained Raspberry Pi. To run these computationally intense algorithms on the Rpi, we need extra hardware. Both Google and Intel have released as Intel (Movidius NCS) and Google (Coral USB Accelerator) devices that can be plugged in as USB in R-Pi (Figure 4.2). We call such devices “coprocessors” as they are designed to augment the capabilities of the primary onboard R-Pi CPU. Combined with the optimized libraries from both Google and Intel, we can obtain faster inference on the Rpi than using the CPU alone.

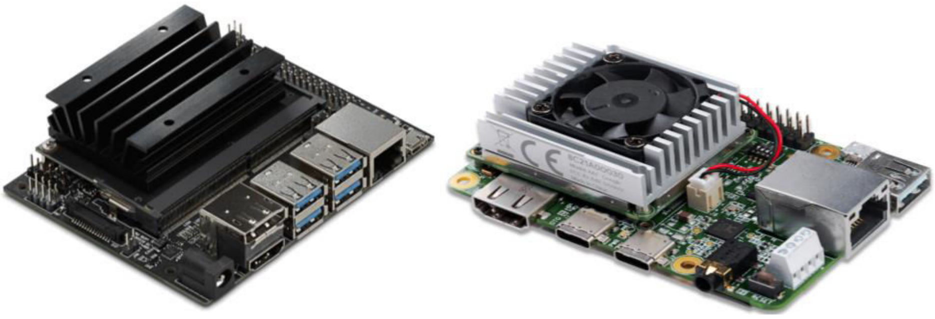


Figure 4.3. Embedded boards a). NVIDIA’s Jetson Nano b). Google Coral’s Dev Board

4.1.7 Embedded Boards and Devices

Of course, there are situations where the Raspberry Pi itself is not sufficient, and additional computational resources are required beyond what coprocessors can achieve. In those cases, you would want to look at Google Coral's Dev Board and NVIDIA's Jetson Nano (Figure 4.3) — these single board computers are similar in size to the Rpi but are much faster (albeit more expensive).

4.2 Deep Learning and Raspberry-Pi

Deep learning has probably played a significant role in computer vision. The level of accuracy that has been achieved with various deep learning algorithms is unbeatable in terms of accuracy in the area of computer vision. Such accuracy demands some cost to provide that unbeatable accuracy. That price is well known in the Embedded Community as computational resources because embedded devices are mainly lacking in these resources. When it comes to working with devices like Raspberry-Pi, which is resource-constrained, a GPU is always required as the smallest model can take multiple orders when being trained on CPU only. When the devices like Raspberry-Pi considered, nobody can even think of trained a deep learning model because of its low computational capability and being so underpowered. As shown in Table 4.1, the estimated values given by the researcher of the prestigious University of Oxford shows a huge computational requirement for common and popular Convolution Neural Networks. Implementation of such networks on the device like Raspberry-Pi can provide approximately 0.41 GFLOPs when Raspberry-Pi 3 with 1.2 GHz of ARM Cortex-A53 processor. So, in comparison to this performance and minimum requirement by famous Convolution Neural Networks, as shown in Table 4.1, quite a massive amount of optimization is required.

Table 4.1. Estimates of FLOP counts and memory consumption for seminal CNN

Model	FLOPs	Memory
Alex Net	727 MFLOPs	233 MB
Squeeze Net	837 MFLOPs	30 MB
VGG16	16 GFLOPs	528 MB
Google Net	2 GFLOPs	51 MB

ResNet-18	2 GFLOPs	45 MB
ResNet-50	4 GFLOPs	83 MB
ResNet-50	4 GFLOPs	83 MB
ResNet-101	8 GFLOPs	170 MB
Inception V3	6 GFLOPs	91 MB
DenseNet-121	3 GFLOPs	126 MB

The next essential parameters that need to take into consideration are Memory and Computational capability. Talking about RAM, Raspberry-Pi can have 1-4 GB of RAM. This RAM needs to take care of the deep learning model but also the internal operations of Raspberry-Pi. In addition to that computation capability of the embedded device also plays an important role. Most of the embedded devices don't consume much of the power. Drawing less power also points towards a less powerful machine.

To work with an embedded device like Raspberry-Pi, thinking of training deep learning is entirely incorrect. Instead of that, one need to follow the steps shown in Figure 4.4.

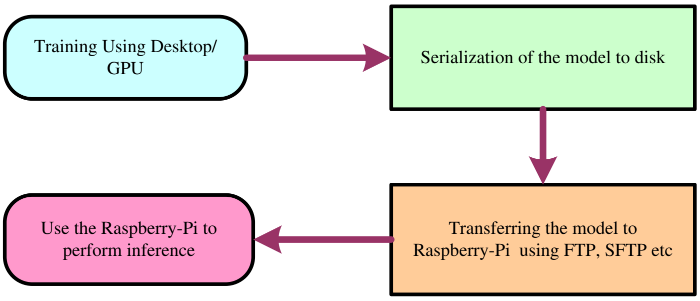

Training Using Desktop/ GPU
Serialization of the model to disk
Transferring the model to Raspberry-Pi using FTP, SFTP etc
Use the Raspberry-Pi to perform inference

Figure 4.4. Steps to implement deep learning model on Raspberry-Pi

4.3 Face recognition using Raspberry-Pi

Face recognition is not confined to one discipline; instead, it is a multi-disciplinary. There are various face detection and facial expression detection techniques, but this first introduced in 1978 (Sown, 1978). In surveillance software, the identification and authentication of users utilizing video and image technologies play an essential role (Nguyen, Yosinski, & Clune, 2015). As observed from the last couple of years that face recognition technology has replaced various biometric security systems (Mandal et al., 2014),(Wang, Zhao, Prakash, Shi, & Gnawali, 2013),(S. Chen, Pande, & Mohapatra, 2014). The primary reason is the capability of recording and even performs interaction. Compared to other biometric techniques like iris and fingerprint, the human face is more reliable (Houmb, Georg, Jurjens, & France, 2008). On the other hand, there are specific techniques like fingerprint scanning, which is more accurate in comparison with face recognition. Multiple forms of face detection are accessible in the literature (Yang, Zhang, Frangi, & Yang, 2004). Deep networks have demonstrated adequate performance to identify the face. A nine-layer network, trained on ~ 4 Million images with a primary classifier, has achieved an accuracy of 97.35% with deep learning (Taigman, Yang, Ranzato, & Wolf, 2014). Various algorithms are available to detect for face detection in real-time, out of which Viola-Jones algorithm is the most popular one (Viola & Jones, 2001). Hough transform is also a new technique that used for face recognition (Varun, Kini, Manikantan, & Ramachandran, 2015). Raspberry-Pi has achieved quite the right amount of popularity over the last couple of years, and various applications proposed like home automation that helps the house owner to get the alerts, and the control can be done via the Internet .

4.3.1 Use of Co-Processor

There are situations when deep learning needs to be performed on Raspberry-Pi. In such a situation's Co-processors need to be utilized. The most popular co-processors used with the Raspberry-Pi are Google coral TPU USB Accelerator, and the other one is Intel's Neural Compute stick. The Neural compute stick can be used by simply plugging in and accessing it via OpenVINO Toolkit or NCS2 SDK. This powerful device can run between 80-150 GFLOPs and, too, with a minimal power requirement of 1 W . The computational capability can make this tiny device, named as Raspberry-Pi capable of performing deep learning using all state-of-the-art deep networks mentioned in Table 4.1. The second Co-processor can be Google coral TPU USB Accelerator.

Similarly, Co-processor also works as discussed for Intel's NCS, but Google has already reported that their coral series products are 10X faster than of Intel's NCS. But the practical implementation and comparative analysis say that to achieve a speed of 10X faster than NCS; one needs to have USB 3. Working with Raspberry-Pi 3/3 B+ gives a similar speed. To get the 10X speed, one needs to use Raspberry-Pi 4, which is having USB 3. The iteration rate is quite comparable between Coral and NCS when it comes to USB 3 with Coral. Figure 4.5 shows the pipeline process with coprocessors using Raspberry-Pi.

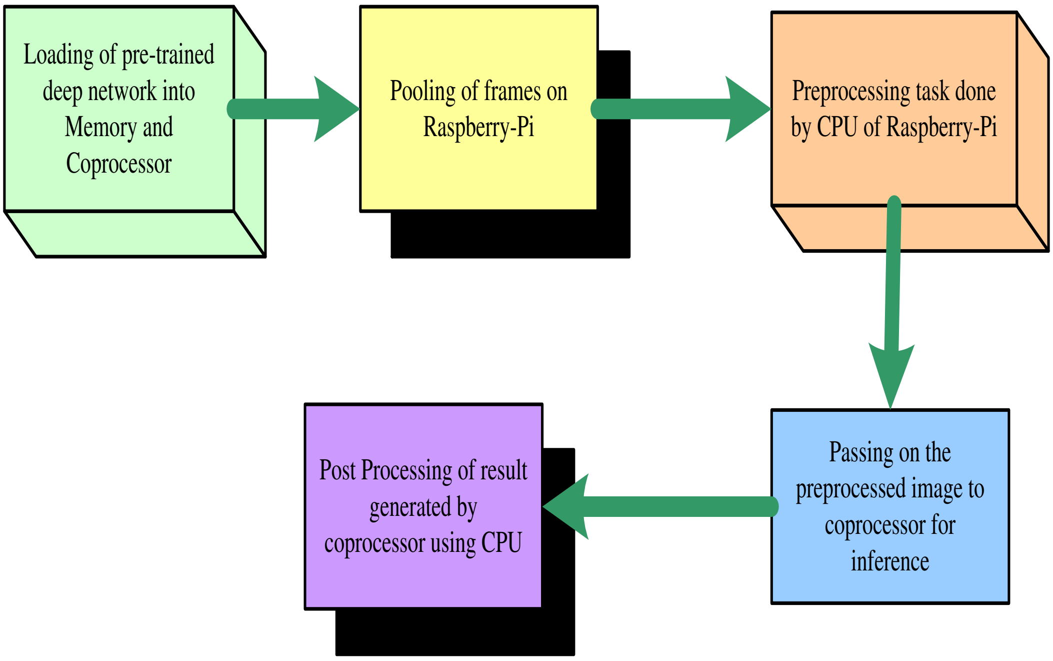

Loading of pre-trained deep network into Memory and Coprocessor
Pooling of frames on Raspberry-Pi
Preprocessing task done by CPU of Raspberry-Pi
Passing on the preprocessed image to coprocessor for inference
Post Processing of result generated by coprocessor using CPU

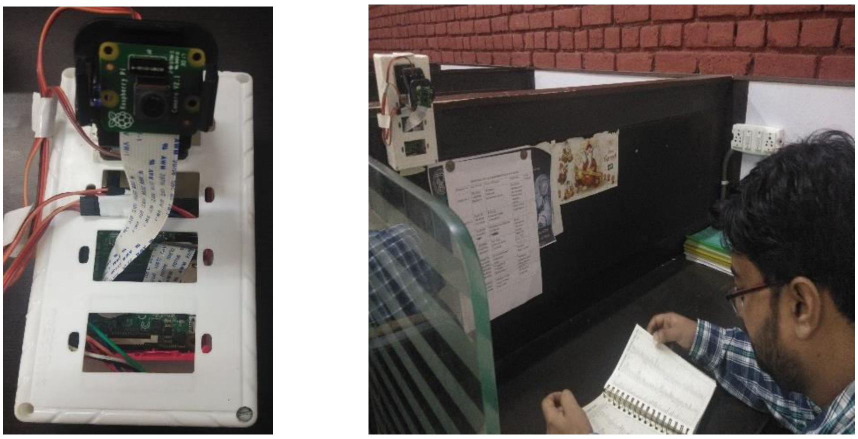
Figure 4.5. Pipeline with Coprocessors

4.3.2 Methodology

This section explains the complete methodology used for fast recognition of face using Intel’s Neural Compute Stick 2 and Raspberry-Pi 4B Model. The entire process starts with the creation of a facial dataset.

4.3.2.1 Dataset Creation

The dataset has been created automatically by using OpenCV and webcam. A data set of 20 people working in a private organization has been created under various lighting conditions and with different emotional states and facial expressions at 6 different time slots of the day. In Compilation, a total of 6 images of each person added in the facial image dataset of 120 images shown in figure 4.6 a).



a). **b).**

Figure 4.6 a). Hardware Setup using Raspberry-Pi and Pi Cam b). Dataset Creation

Hardware setup having Raspberry-Pi and Pi-Camera used, and dataset of facial images captured as shown in figure 4.6 b). These images captured using OpenCV and a few images of the dataset shown in figure 4.7.

a) b) c)

Figure 4.7. Dataset Images a). Himani b). Swanil c). Ravi

4.3.2.2 Face detection and Facial Embeddings Extraction using Movidius NCS

The primary step after the creation of the dataset is to quantify it and then to measure the faces in the training dataset. The deep network has not trained here using Raspberry-Pi; instead has been used to extract the facial embeddings from the dataset using a pre-trained deep network. The network is pre-trained with ~500k images at Carnegie Melon University as a part of the Open Face project (Amos, Ludwiczuk, & Satyanarayanan, 2016). The network architecture used is based on the ResNet-34, as mentioned in a paper for Face Recognition using Deep Residual Learning (He, Zhang, Ren, & Sun, 2016) with some reductions in the number of layers and filters. This network is trained by Davis King (King, 2009) on public data set Labeled Faces in the Wild (LFW). This network achieves around 91.6% accuracy when compared with other modern methods. The famous Adam Geitgey and Davis King, who are the author of the face recognition module and creator of dlib has explained the details about how the network trained to produce 128-d to quantify the faces, as shown in figure 4.8. These 128 dimensions are a unique measurement of each face and act as a base for the classifier to recognize the face. To achieve this step with Movidius NCS a shell script has been used to set up the environment for NCS using Open VINO. The detector used to detect the face using NCS is a deep-learning face detector based on caffe which helps to locate the faces in the images, and the model used for embedding is the OpenCV deep-learning torch embedding model.

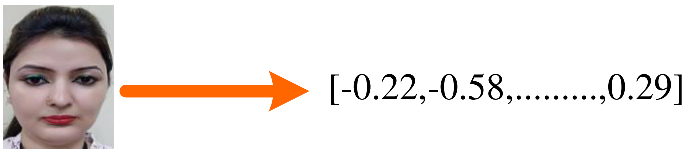
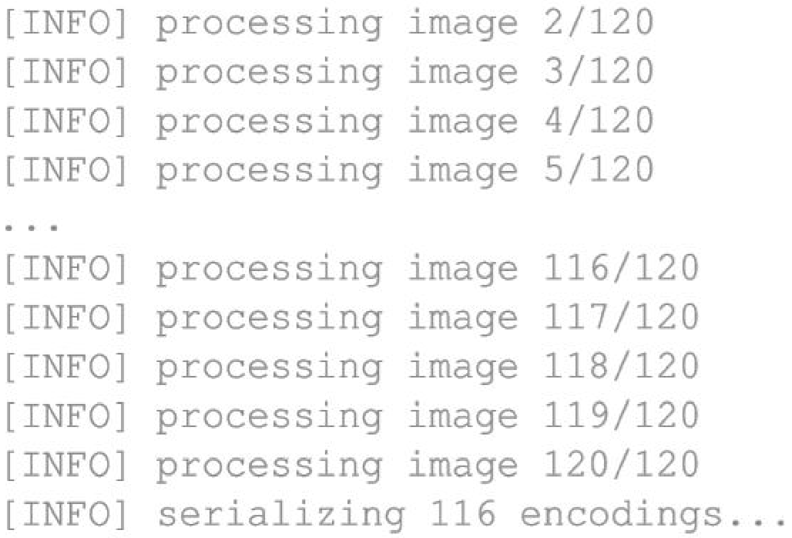


Figure 4.8. Generation of the 128-d real-valued number feature vector of each training image in the dataset



```
[INFO] processing image 2/120
[INFO] processing image 3/120
[INFO] processing image 4/120
[INFO] processing image 5/120
...
[INFO] processing image 116/120
[INFO] processing image 117/120
[INFO] processing image 118/120
[INFO] processing image 119/120
[INFO] processing image 120/120
[INFO] serializing 116 encodings...
```

Figure 4.9. Computation of facial embeddings with OpenCV and Movidius

The time taken to process 120 images of the dataset was 57 seconds with Intel's NCS and USB 3 of the Raspberry-Pi 4B model. Figure 4.9 shows the completion of embeddings computations.

4.3.2.3 Training

The next important step is training a standard machine learning model so that the trained model could be able to identify an actual person via those extracted 128-d embeddings that are unique for each face. Plenty of options are available, like Random Forest, SVM, and k-NN. When it comes to training a smaller dataset, k-Nearest Neighbor is useful via face_recognition library and dlib (King, 2009). In this work, a more robust classifier known as Support Vector Machine (SVM). It is achieved with the help of Scikit-learn. The kernel used in this process is the Radial Basis kernel. The kernel is quite tricky to tune when compared with another linear kernel. So, to use this kernel, a process is known as 'grid searching' has been used. This process is helping to find the optimal parameters during machine learning for a particular model.

4.3.2.4 Real-Time Face Recognition

Once the network trained, the next step is to recognize the face in real-time. The detailed process of Face recognition in video streams using Movidius NCS shown in figure 4.10. This part of the methodology is a pre-trained model or Deep Learning models like the previous two stages of detector and embedder. Instead, this is an SVM based machine learning model for face recognition. In this stage, the CPU of the Raspberry-Pi utilized to recognize the faces in real-time. Figure 4.10 shows the complete method of face recognition with the help of Movidius NCS. The entire process starts with the warming up of the camera sensor and the beginning of the video stream. A counter for counting frames per second initialized to benchmark. After capturing the frame from video, resize and detect the face in that frame. After the formation of blobs, the face detected, but before that, weak detection needs to filter out and then extract the ROI. The pointed ROI then used to detect spatial dimensions to ensure correct recognition. After ensuring that the spatial dimensions are large enough and are more than minimum probability, then finally, the face blobs are constructed and passed through the embedder to generate a 128-d vector. SVM embedder is then finally predicted with the name and probability index.

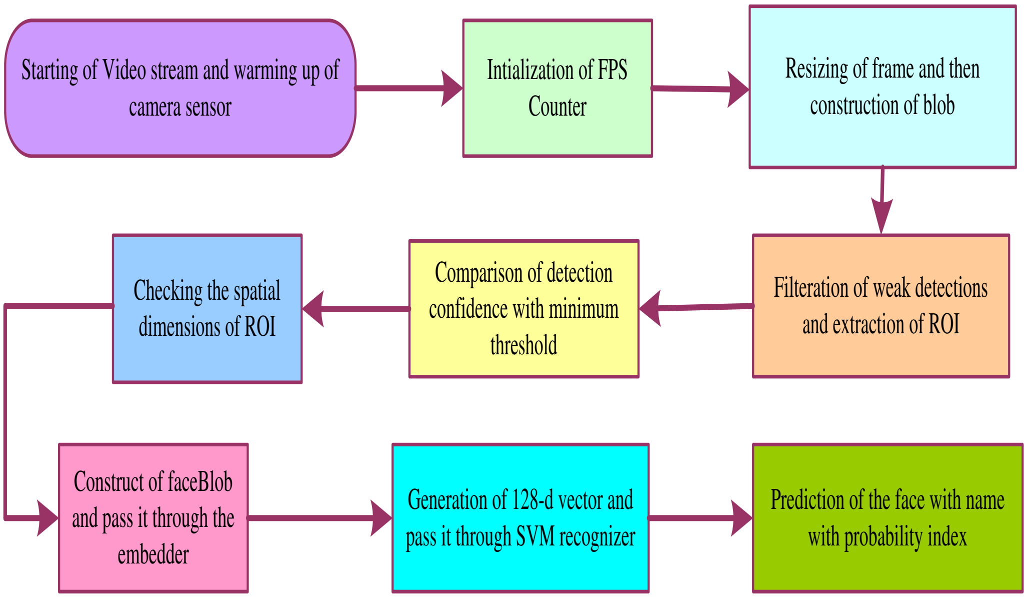
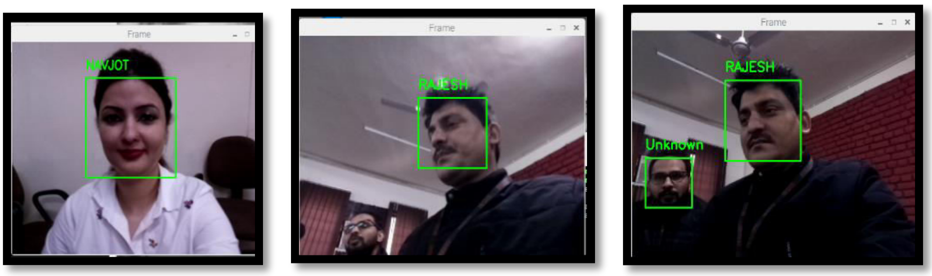

Starting of Video stream and warming up of camera sensor
Intialization of FPS Counter
Resizing of frame and then construction of blob
Filteration of weak detections and extraction of ROI
Comparison of detection confidence with minimum threshold
Checking the spatial dimensions of ROI
Construct of faceBlob and pass it through the embedder
Generation of 128-d vector and pass it through SVM recognizer
Prediction of the face with name with probability index

Figure 4.10. Process flow of face recognition in real-time using Movidius NCS

The face recognition results achieved from video, implemented on Raspberry-pi in real-time, are shown in figure 4.11. At this stage, the confidence has applied; only the exact match displayed by displaying the name of the person after the correct classification.



a) b) c)

Figure 4.11. Face Recognition results a). Face Recognized as Navjot b). Face recognized as Rajesh c). Unknown face detected

This chapter provided an alternative of a resource-constrained device over those conventional CPUs, i.e., Raspberry-Pi. Low power and credit card-sized device can be used for CV and DL with the help of co-processors and perform very well in real-time. This chapter gave a detailed insight into the Methodology to work with such powerful yet compact and low power embedded devices.

CHAPTER 5

METHODOLOGY

This Chapter explains the Methodology and approaches used in the research work. For Methodology, two different approaches used. In the first approach, facial emotion detection has been done on the cloud while the face detection part has been implemented using Raspberry-pi itself and then uploaded on the cloud. In the second method, a pretrained deep network has been imported on Raspberry-Pi itself using Co-Processor Intel Movidius neural compute stick. The first method was not using any co-processor and implemented with the help of CPU of Raspberry-Pi only.

5.1 First Method

Face detection is altogether different from face recognition. Face recognition is a combination of multiple related problems.

1. To begin with, find the faces inside the picture

2. Secondly, look at the face profoundly and try the identify person even if the lighting conditions are different or face is tilted. Extract the unique features of the face to tell it apart from other faces. Those features can be measurements of eyes, nose, ears, or distance between them, shape.

4. Lastly, compare all the above-extracted features with already known faces. Confirm the identified person by giving the exact name.

The elucidation of each step elaborated in the next sections. The entire process divided into three tightly coupled tasks. The first task is to train the pertained deep network after segregating the dataset into testing, training, and validation. The entire dataset of emotive facial images segregates into 8:1:1 ratio. A dataset with N images divided into $ժ_{TR}$ for training, $ժ_{VD}$ for cross-validation, and ꝺ$_T$ for testing purposes. It means a training set of N number of images I consist of $ꝺ_{TR} = \{I_1^{TR}, I_2^{TR}, I_3^{TR} \dots I_{4N/5}^{TR}\}$as training images $ꝺ_{VD} = \{I_{(4N/5\,)+1}^{VD} \dots I_{9N/10}^{VD}\}$ as validation images and $ꝺ_T = \{I_{(9N/10)+1}^{T} \dots I_N^T\}$As testing images. A deep convolutional neural network is known as Mini-Xception used for testing, training, and validation of emotive facial images. Training, validation, and testing is done on Google-CoLab with 12GB NVIDIA Tesla K80 GPU using FER 2013 dataset. The whole system divided into two tightly coupled functions, i.e., face classification and real-time face emotion classification. A pretrained deep network known as Open Face used for face recognition. To begin with, a real-time image from the video captured as $\overset{r}{I}_1$.Total number of images captured in real-time is δ_R=$\{\overset{r}{I}_1, \overset{r}{I}_2 \dots \overset{r}{I}_n\}$.In order to train the deep network 6 images of each subject used, and the network is trained with a single triplet method of training for 20 different people with $\mathbb{N}$ images and denoted as δ_{TR}. Once the training is complete the real-time captured image as $\overset{r}{I}_1$ The first step is to find the face within the caught picture and delete the unnecessary details. The faces were detected using a well-known method known as HOG (Oriented Gradient Histogram). After the detection of face the facial image $\overset{r}{I}_1$ has been cropped, which further preprocessed to remove the effects of bad lighting, tilted face, and skewness. in the cropped image $\overset{r}{I}_{cr}$ .The cropped image preprocessed with the face landmark estimation algorithm. This algorithm locates 68 landmarks on the cropped image $\overset{r}{I}_{cr}$ and with the help of simple affine transformations, the image is preprocessed $\overset{r}{I}_{pr}$ using rotation, shear, and scale to center the eyes and mouth of the cropped image $\overset{r}{I}_{cr}$ at best. The preprocessed image $\overset{r}{I}_{pr}$ is fed to the pre-trained network to extract the features from $\overset{r}{I}_{pr}$ image and generate 128 embedding's that are measurements of face. The feature of the preprocessed image $\overset{r}{I}_{pr}$ is generated with the help of a neural network generates a feature vector of 128 embeddings as Φ_n^{embd} .The final step is to recognize the image by measuring the closest match of 128 embedding's Φ_n^{embd} by comparing it with the database images. The feature vector Φ_n^{embd} is passed through the simple SVM classifier €, to recognize the face. The second task is to transmit the output of the preprocessing stage $\acute{I}_{pr}^{r}$ to the cloud where already a pre-trained network of the emotive facial recognition system is available. This image $\acute{I}_{pr}^{r}$ is again passed through all the layers of mini-Xception deep network that is a fast and depth-wise separable convolutional neural network for the recognition of emotion captured in $\acute{I}_{1}^{r}$ image. The deep-network on the cloud is trained on seven primary emotions and labeled as $\{\varepsilon_c = \varepsilon_1\, \varepsilon_2\, _{\dots}\, \varepsilon_7\ \}$ Where the basic classes of seven emotions represented as c, the entire architecture with its detailed framework represented in Figure 5.1. The description of the parameter used in the proposed architecture given in Table 5.1.

Table 5.1. Description of Parameters of the proposed architecture

ժ→ Emotive Facial Dataset	**Φ_n^{embd}→128 Feature vectors as face embedding's**
ժ$_{TR}$→Training Images	$\acute{I}$→Real-time image representation
ժ$_{VD}$→Validation Images	$\acute{I}_1^r$→First real-time facial image
ժ$_{T}$→ Testing Images	$\acute{I}_{pr}^r$→Preprocessed real-time facial image
δ_R→Real-Time Captured Images	$\acute{I}_{cr}^r$→Cropped real-time facial image
$I\,_1^{TR}$→First Training Image	$\acute{I}_n^r$→ n real-time facial images
$I_{4N/5}^{TR}$→80% Training Images	€→SVM Classifier
$I_{9N/10}^{VD}$→10% Validation Images	ε_c→Classifier with labels having c=0 to 6 classes
I_N^T→ N number of testing images	I→ Image representation
δ_{TR}→ Training Dataset of facial images	N→Face detection database images

Step 1. The very first step of the system is to locate the face on the real-time video frames the generalized block diagram for the detection of the face is shown in figure 5.2. As everyone witnessed that face detection is a common feature in cameras nowadays. In this step, Open CV's Haar cascade used for face detection. A bounding box located on the facial part of the image $\acute{I}^{r}$, which is being converted to grayscale to extract the facial part in the input image $\acute{I}^{r}$. The face detector that has been used in

this stage is Haar cascade introduced by Viola and Jones in the year 2001(Viola & Jones, 2001). Haar cascade is almost 20 years old descriptor, but still, it has been used because of its speed and in comparison, to another descriptor including HOG + Linear SVM (Dalal & Triggs, 2005). Moreover, this work carried on the resource-constrained device known as Raspberry-Pi, which needs a descriptor with less computational complexity.

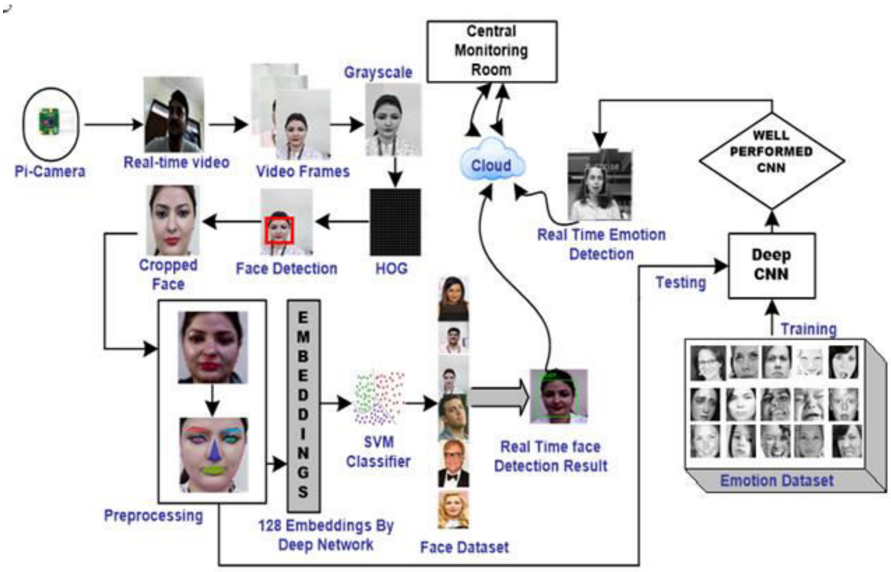


Figure 5.1Architecture for face recognition and facial emotion recognition in real-time using Raspberry-Pi

Figure 5.2 depicts the flow chart of a real-time face detection system. The block diagram of a process flow for training an object detector using HOG shown in figure 5.3. Active Shape Model and HOG used for Face alignment and feature extraction. ASM programmed fiducials point area calculation is connected first to an outward appearance picture, and afterward, Euclidean separations between focus gravity arrange and the clarified fiducials focus directions of the face picture are determined. To extricate the separate deformable geometric data, the framework removes the geometric twisting distinction includes between an individual's unbiased articulation and the other fundamental articulations. In ASM info, face shape iteratively distorted to get the shape model. After examination with the shape model element purpose of the information, the facial picture separated.

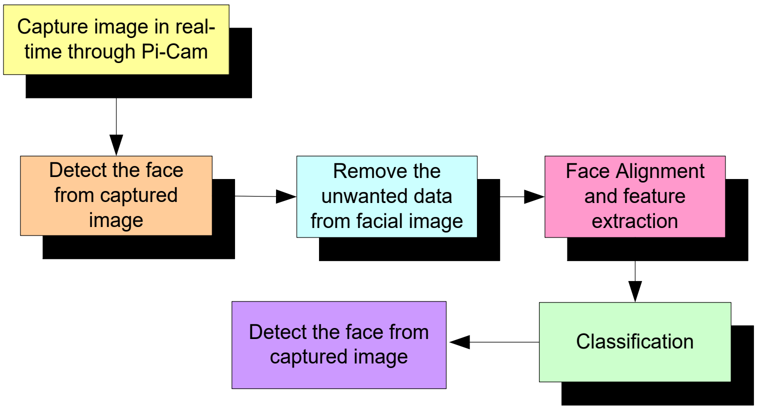

Capture image in real-time through Pi-Cam
Detect the face from captured image
Remove the unwanted data from facial image
Face Alignment and feature extraction
Classification
Detect the face from captured image

Figure 5.2 Block diagram of a real-time face detection system

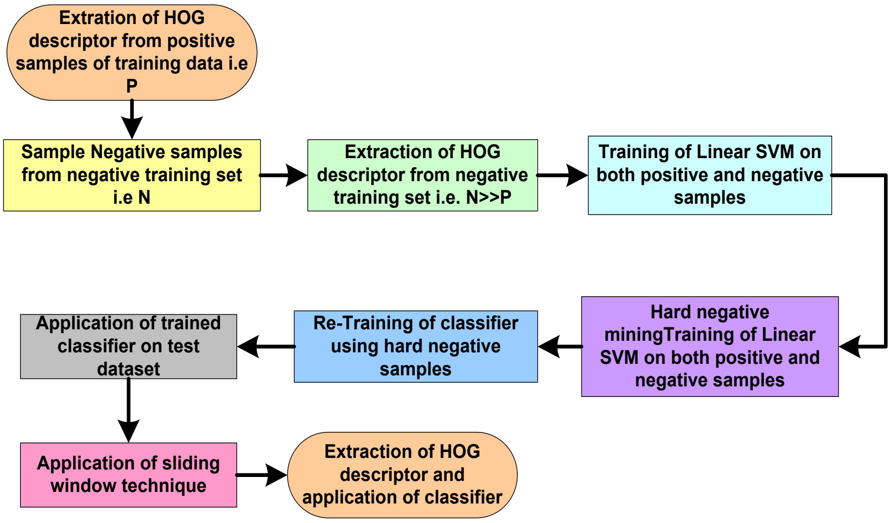

Extration of HOG descriptor from positive samples of training data i.e P
Sample Negative samples from negative training set i.e N
Extraction of HOG descriptor from negative training set i.e. N>>P
Training of Linear SVM on both positive and negative samples
Hard negative miningTraining of Linear SVM on both positive and negative samples
Re-Training of classifier using hard negative samples
Application of trained classifier on test dataset
Application of sliding window technique
Extraction of HOG descriptor and application of classifier

Figure 5.3. Process flow of training an object detector using HOG

Step 2. Function d*etectMultiScale* called. This function of the detector is responsible for detecting the faces in the image. This function returns *rects, i*.e., a list of locations for bounding boxes. The bounding boxes represent x, y, w, h where x: the rectangle x-cord, y: the rectangle's y-coordinate, w: the rectangle width, and h: the rectangle height. When all these values considered together, they will make a bounding box and locate the faces in the image I^{r} and the image is cropped with a facial part known as I^{r}_{cr}. As already mentioned, it is easy to implement Haar cascade for real-time face detection using Raspberry-Pi, but other advanced descriptors, i.e., HOG + Linear SVM and deep learning-based face detectors can also use. Figure 5.4 shows the detection of face using HOG and finally cropped face that can be classified further using the descriptor.

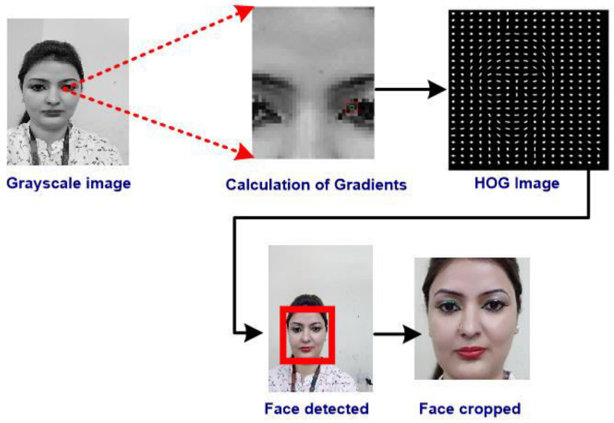


Figure5.4 Face detection using Histogram of Oriented Gradients (HOG)

Step 3. For pictures with frontal face, the descriptor can easily position the face in the image, but posing and projecting the faces is the major problem. As the faces turn and those faces look entirely different from the system. To resolve this issue, an algorithm is known as facial landmark estimation (Kazemi & Sullivan, 2014) used. This algorithm locates 68 specific points on the face known as facial landmarks as the face is captured in real-time, so the image captured can be having faces turned in a different direction. To deal with such situations, we wrapped each picture so that our system can locate the eyes and lips in a sample place, after locating these landmarks that locate the eyes, nose, chin, lips, and eyebrows on any face. $S = \left(x_1^T, x_2^T, \ldots.., x_p^T\right)^T \in \mathbb{R}^{2p}$ S is the vector that is representing the p number of facial landmarks in image *I.* The main aim is to perform the estimation of S to the best possible estimate, which is nearest to the exact shape and is denoted by $S'^{(t)}$. It is done with the help of a cascade of the regressor. Each regressor keeps on predicting and continuously updating the vector so that the estimation comes out to be accurate. $S'^{(t+1)} = S'^{(t)} + r_t S'^{(t)}$ Is the method in which the regressor $r_t(.,.)$ is being used in cascade for prediction and updating the vector $S'^{(t+1)}$.The face alignment result using an ensemble of regression trees is shown in Figure 5.5. One can visualize that face is exactly centered using this, and this pre-processing technique is beneficial in increasing the classification accuracy.

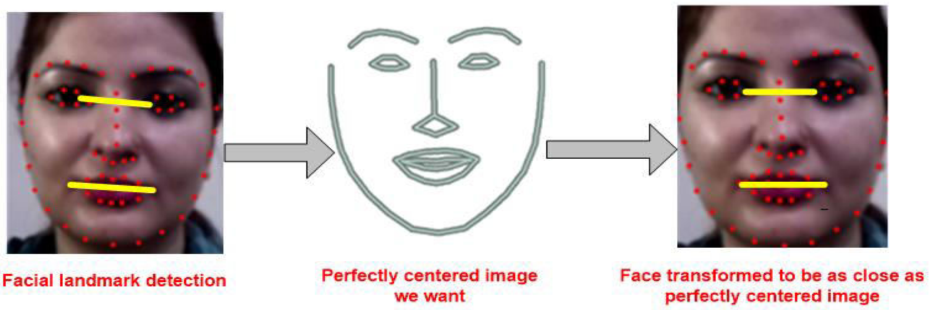

Facial landmark detection
Perfectly centered image we want
Face transformed to be as close as perfectly centered image

Figure 5.5 Face is aligned in real-time using an ensemble of regression trees and affine transform

Step 4. The next major step is to extract the features from the image now centered precisely. To achieve that, the best way is to measure and take different dimensions of the face because the facial dimensions of every person are unique. The main point of the challenge is which measurement plays the most crucial role in identifying a person apart from others. It becomes challenging to identify by using handcrafted or conventional feature extraction methods. The deep neural networks play an essential role here, as there can be n number of features that are important to identify for telling a face apart from other faces.

Moreover, to speed up and automate the process, a deep neural network is the best option. For this purpose, a pre-trained network provided by Open Face has been used (Amos et al., 2016). This network just needs inputs in the form of images, and the network automatically extracts the features in the form of 128-embeddings, as shown in Figure 5.6. These embeddings are unique for each face and act as the reference for correct recognition of a face. Instead of training the facial images individually, the network trained on 3 facial images at a single point. Of these three pictures, 2 are one individual, and the third is a different person's picture. The same two photos alluded as (anchor picture) and (positive picture).

The third picture is labeled "bad image" by another human. The picture of the anchor is the image to be trained. To understand triplet loss, consider $f(y) = \Im^{s}$which is representing an image y into s-dimensional Euclidean space. We oblige this implanting to live on the s-dimensional hypersphere.$\|f(y)\|_2 = 1$.As shown in Figure 5.7, the main aim is to achieve a minimum distance between y_j^m (Anchor) of a specific person with all the other images y_j^p(Positive) of the same person as compared to the image of any other person y_i^n(Negative) .Figure 5.8 more detailed view along with the images of the subjects to be recognized. A triplet of two similar images of subject 1 and one different image of the second subject trained to generate 128-d data for triplet calculation and train the model.

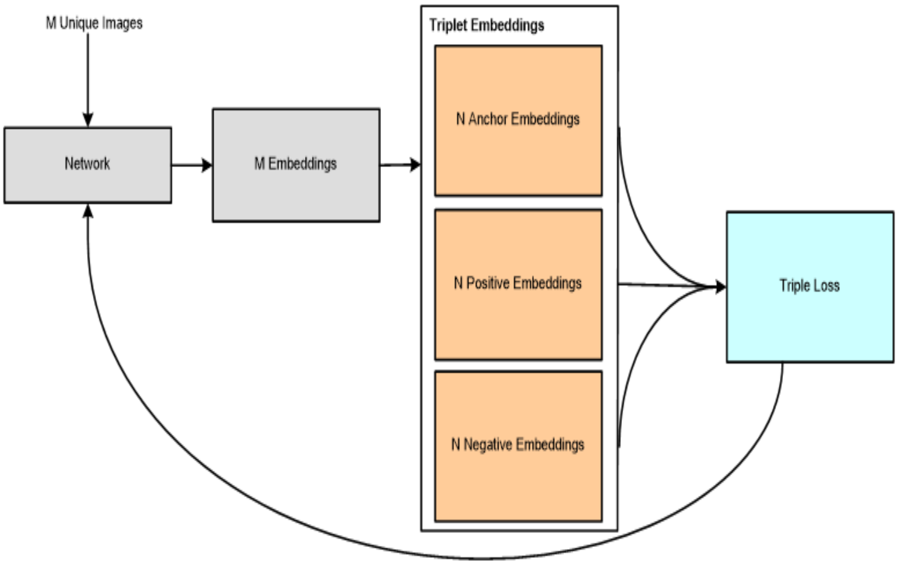


Figure 5.6 Network training flow for M unique images

So, we want to have:

$$\left\|y_j^m - y_j^p\right\|_2^2 + \beta\left\|y_j^m - y_j^n\right\|_2^2 \forall \left(y_j^m, y_j^n, y_j^p\right) \in \zeta \tag{5.1}$$

Where β is the enforced margin between negative and positive pairs of images, and ζ is the set of all the possible triplets and has numbers equal to number P.

$$\Sigma_j^P \left[\left\|f(y)_j^m - f(y)_j^p\right\|_2^2 - \left\|f(y_j^m) - f(y_j^n)\right\|_2^2 + \beta\right]_+ \tag{5.2}$$

The generation of multiple triplets helps to overcome the issue faced in Eq. (1) and selection of suitable and sophisticated triplets results in the improvement of the deep learning model.

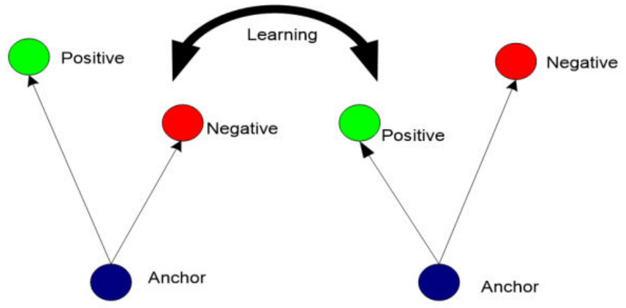


Figure 5.7 Triplet loss

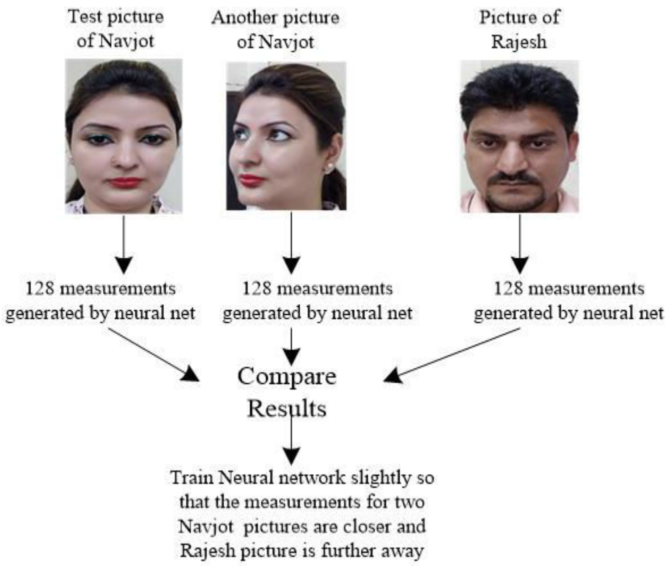

Test picture of Navjot
Another picture of Navjot
Picture of Rajesh
128 measurements generated by neural net
128 measurements generated by neural net
128 measurements generated by neural net
Compare Results
Train Neural network slightly so that the measurements for two Navjot pictures are closer and Rajesh picture is further away

Figure 5.8 Generation of 128-dimensional data from triplet

Step 5. The next step is to recognize the face correctly using various classifiers. There are n-numbers of classifiers available, but here the process is kept simple because of the limited resources available on Raspberry-Pi. To make it a quick and easy simple SVM classifier used as running this classifier on hardware done in milliseconds. Moreover, real-time implementation always needs fast and accurate classifiers. The logic flow of face recognition with a neural network is shown in Figure 5.9 consist of basic blocks that are required to detect the human face in real-time.

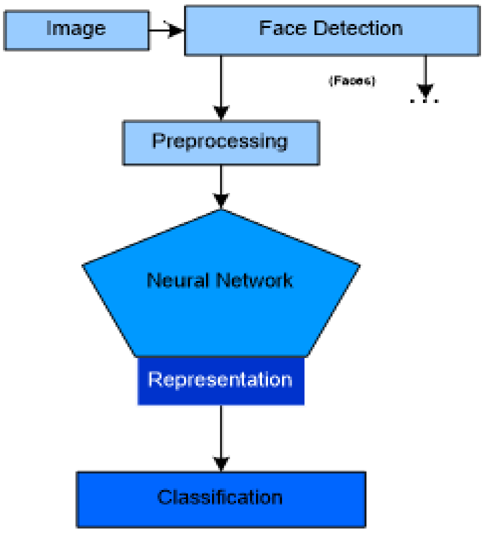


Figure 5.9 Logic flow for face recognition with neural network

Step 6. Once the face identified, the image cropped and uploaded on the cloud for further processing to understand the facial emotion of the classified image. It is done with the help of the GPU. The entire process is divided into three tightly coupled tasks. The first task is to train the pertained deep network after segregating the dataset into training, testing, and validation. The entire dataset of emotive facial images is divided into 8:1:1 ratio. A dataset with N images divided into Ɗ$_{\mathrm{TR}}$ for training, Ɗ$_{\mathrm{VD}}$ for cross-validation, and Ɗ$_{\mathrm{T}}$ for testing purposes. This means a training set of N number of images I consist of $Ɗ_{TR} = \{I_1^{TR}, I_2^{TR}, I_3^{TR} \dots I_{4N/5}^{TR}\}$as training images $Ɗ_{VD} = \{I_{(4N/5\,)+1}^{VD} \dots I_{9N/10}^{VD}\}$ as validation images and $Ɗ_T = \{I_{(9N/10)+1}^{T} \dots\ I_N^T\}$As testing images, a deep convolutional neural network known as Mini-Xception used for testing, validation, and training of emotive facial images. All these steps are done on Google-CoLab with 12GB NVIDIA Tesla K80 GPU using FER 2013 dataset.

5.1.1 Convolutional Neural Networks

The capability of image classification is well-known CNNs. A first neural network consists of multiple layers of linked neurons. Such a layer's store and filter the input images into various layers. An underlying neural network Convolutionary (CNN) consists of three-layer types: a convolutionary layer, and a fully connected layer connected with a max-pooling layer. The first two layers are responsible for removing information, adding non-linearity, and decreased features to reduce duplication. The last layer, known as the fully connected layer helps to classify the features extracted in earlier layers. The utterly connected layer contains most of the parameters. The number of parameters has also been reduced presented in architectures like Inception V3, in which the last layer added, i.e., of Global Average Pooling operation. By taking the average and converting the feature map into a scalar form, this layer reduces the feature map. The use of residual modules and depth-separable convolutions w also introduced further to reduce Modern Convolutional Networks architectures. The depth-separated CNN's separate the function extraction task and combine it into a convolutionary layer, thus further reducing the parameters. We have thus used an Octavio Arriaga & al. (CNN) mini-Xception neural network, which reduced the parameters by using profoundly separable convolution layers rather than simple convolution layers and eliminates fully connected layers. The logic flow for the process of face recognition with a neural network shown in Figure 5.10.

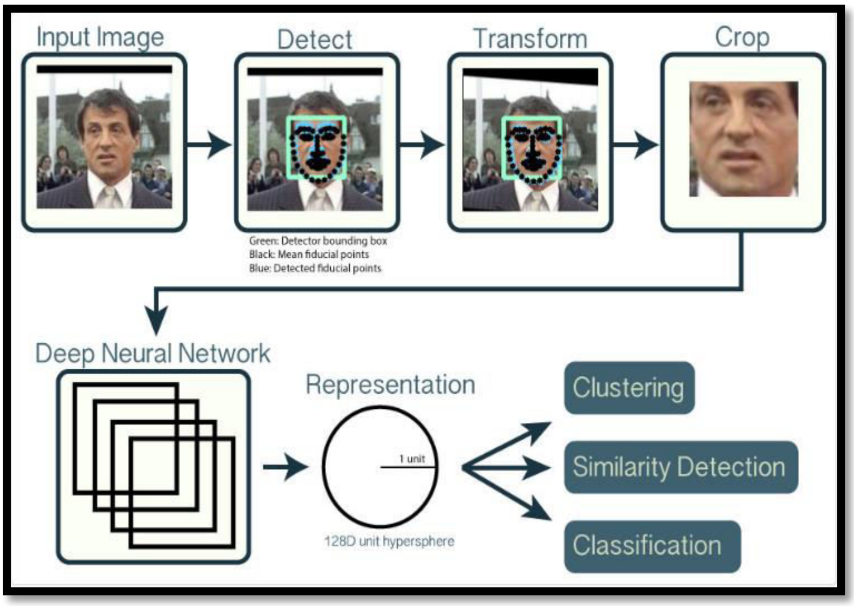


Figure 5.10 Logic flow for face recognition with neural network

5.1.2 Dataset

Our dataset comprises of 35,888 photographs in seven types of facial emotions (0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral) as shown in Table 5.2. FER 2013 comprises of gray-scale photographs of 48X48 pixels. The dataset that we have used is in csv format consist of only two columns, i.e., "emotions" and "pixels" and is kept in Google drive. The entire data is divided into 8:1:1 ratio for training $\eth_{TR}$, validation $\eth_{VD}$ and testing $\eth_{T}$.Large number of datasets is available to detect facial emotions. We have validated our system with the FER2013 dataset only because of its non-uniform distribution of images, as one can quickly analyze from Table 5.2. The efficiency of the system can further increase by using a uniformly distributed dataset.

5.1.3 Training CNN model: Mini Xception

The dataset kept on Google drive, and the trained on Google-CoLab with 12GB NVIDIA Tesla K80 GPU. CNN trained with 80% of training data from the FER dataset, and the remaining 10% of the dataset kept for validation. Testing has been done on the remaining 10% of data on the input given by Raspberry-pi after detecting the face and converting the cropped and pre-processed images into 48X48 sizes. The architecture of Mini-Xception proposed by Octavio Arriaga et. al.

Table 5.2 Distribution of classes in FER2013 Dataset

	Emotion	**Number**
0	Angry	4953
1	Disgust	547
2	Fear	5121
3	Happy	8989
4	Sad	6077
5	Surprise	4002
6	Neutral	6198

This architecture is trained on FER 2013 dataset because we want the response to be quick and the proposed architecture of Mini-Xception has been proved to be quick and light because of its unique architecture and replacement of convolutional layers with depthwise convolutional layers decrease the number of limitations and make it reliable to implement it on real-time emotion recognition as our system is based on Raspberry-Pi, which is having certain constraints in terms of memory and processing capability. Hence, a smaller number of parameters are helpful in the future advancement of this system. To achieve that, depthwise convolution and pointwise convolutional layers help to reduce the parameters by 80X when compared with original CNN.

The next section explains the Algorithms designed to perform pre-processing for improving the accuracy of face detection and algorithm to detect the face in real-time. Two different algorithms are explained in Table 5.3 and Table 5.4.

Table 5.3 Algorithm 1: Face detection in real-time

```
Input: Real-time video of subjects Í
I. Capture the real-time image Í from the real-time video frames
II. Facial Dataset creation
For k=1 to the size of $(\boldsymbol{\delta_{TR}})$ = $\mathbb{N}(\boldsymbol{\delta_R} = \{\acute{I}^r_1, \acute{I}^r_2 \ldots \acute{I}^r_n\})$, N is the count of each subject sample, and n is the total number of subjects captured
        a. Select an image $\acute{I}^r_1$ from $\boldsymbol{\delta_R}$
        b. Convert the image $\acute{I}^r_1$ to grayscale image:
$\acute{I}^r_G \leftarrow$ Gray Scale $(\acute{I}^r_1)$
        c. Detect the face region using Histogram of Oriented Gradients (HOG):
$\acute{I}^r_H \leftarrow$ HOG$\acute{I}^r_G$
        d. Crop the facial region $\acute{I}^r_{CR} \leftarrow$ Cropped $\acute{I}^r_H$
        e. Preprocess the cropped image $\acute{I}^r_{CR}$ by applying facial landmark and affine transform:
$\acute{I}^r_{pr} \leftarrow$ Pre-processing $\acute{I}^r_{CR}$
        f. Repeat the sub-steps a to e of II of database creation for n x $\mathbb{N}$ times
        End
III. Label all $\boldsymbol{n}\mathbb{N}(\acute{I}^r_{pr})$ images of a dataset with the name of the subjects:
$(\boldsymbol{\delta_{TR}})^l \leftarrow$ Labeled $\boldsymbol{n}\mathbb{N}(\acute{I}^r_{pr})$
IV. Training and feature extraction:
For k=1 to the size of $(\boldsymbol{\delta_{TR}})^l$ ,where l represents labelled data
        a. Select the anchor image $\boldsymbol{y^m_j} = (1.\acute{I}^r_1)$ of first subject
        b. Select the positive image $\boldsymbol{y_j^p} = (2.\acute{\mathrm{I}}_{\mathbf{1}}^{\mathbf{r}})$ Of the same subject
        c. Select the negative image.$\boldsymbol{y_i^n} = (1.\acute{\mathrm{I}}_{\mathbf{2}}^{\mathbf{r}})$ Of the second subject
        d. Feed the images $\boldsymbol{y_j^m}, \boldsymbol{y_j^p}$ ,$\boldsymbol{y_i^n}$ to the pre-trained deep network

e. Repeat the training to achieve $\left\|\boldsymbol{y_j^m} - \boldsymbol{y_j^p}\right\|_2^2 + \boldsymbol{\beta}\left\|\boldsymbol{y_j^m} - \boldsymbol{y_j^n}\right\|_2^2 \forall \left(\boldsymbol{y_j^m}, \boldsymbol{y_j^n}, \boldsymbol{y_j^p}\right) \in \boldsymbol{\zeta}$,
where $\boldsymbol{\beta}$ is the enforced margin between negative and positive pair of images and
$\boldsymbol{\zeta}$ is the set of all the possible triplets and has numbers equal to number $\boldsymbol{M}$
 f. Generate multiple triplets to improve deep learning by $\sum_j^M \left[\left\|\boldsymbol{f}(\boldsymbol{y})_j^m - \boldsymbol{f}(\boldsymbol{y})_j^p\right\|_2^2 - \left\|\boldsymbol{f}(\boldsymbol{y_j^m}) - \boldsymbol{f}(\boldsymbol{y_j^n})\right\|_2^2 + \boldsymbol{\beta}\right]_+$
 g. Generate the feature vector$\Phi_n^{embd}$, where n is a deep network that generates the number of embedding
End
V. Cross validate by capturing an image $\acute{\mathrm{I}}^{\mathbf{r}}$ in real-time
        i. Repeat the substeps a to e of II Facial Dataset creation for $\acute{\mathrm{I}}^{\mathbf{r}}$
        ii. Repeat the substeps a to f of IV Training step and feature extraction
VI. Feed the image to a classifier to classify the image $\acute{\mathrm{I}}^{\mathbf{r}}$ in real-time by passing all the feature vectors $\Phi_n^{embd}$ called embedding's that were generated in the substeps a to g of IV Training step and the
   Output: Prediction of the face of the subject with name in real-time
```

Table 5.4 Algorithm 2 Emotion detection in real-time

```
Input Emotive facial dataset $\boldsymbol{Ꝺ}$
I. Divide the dataset into training $\boldsymbol{Ꝺ_{TR}}$,validation $\boldsymbol{Ꝺ_{VD}}$ and testing $Ꝺ_T$
II. Training and feature extraction:
For j=1 to the size of ($\boldsymbol{Ꝺ_{TR}}$) = $\boldsymbol{I_{4N/5}^{TR}}$, where N is the total count of images in $Ꝺ$
        a. Selection of the image $\boldsymbol{I_1^{TR}}$from$\boldsymbol{Ꝺ_{TR}}$
        b. Feed the input $I_1^{TR}$ to the deep network
        c. Training the network by feeding the $\mathbf{images}$ ($\eth_{TR}$) along with their labels and let the
            network extract all the parameter
        End
III. Cross Validate:
For j=1 to the size of ( $\eth_{VD} = \{I_{(4N/5\,)+1}^{VD} \cdots I_{9N/10}^{VD}\}$
        a. Select the image $I_{(4N/5\,)+1}^{VD}$ from $\eth_{VD}$
b. Repeat the sub-steps b and c from II Training and feature extraction for images $\eth_{VD}$
c. Use validation images $\eth_{VD} = \{I_{(4N/5\,)+1}^{VD} \cdots I_{9N/10}^{VD}\}$ the reduction of overfitting
        End
IV. Testing:
 For j=1 to the size of $\eth_T = \{I_{(9N/10)+1}^{T} \cdots I_N^T\}$
        i. Select the image $I_{(9N/10)+1}^{T}$ from $\eth_T$
        ii. Repeat the sub-steps b and c from II Training and feature extraction for images $\eth_T$
        iii. Use testing images $\eth_T = \{I_{(9N/10)+1}^{T} \cdots I_N^T\}$ to test the trained network for efficiency
End
V. Real-time testing:
Take the input from subset e of II from Algorithm 1
For k=1 to the size of $(\boldsymbol{\delta}_{TR}\ ) = \mathbb{N}(\ \boldsymbol{\delta}_R = \{\overset{r}{I}_1, \overset{r}{I}_2 \ldots \overset{r}{I}_n\})$, $\mathbb{N}$ is the count of each subject sample, and n is the total number of subjects captured
i. Resize the image $\overset{r}{I}_{pr} \leftarrow$ Pre-processing $\overset{r}{I}_{CR}$
$\overset{r}{I}_{RS} \leftarrow$ Resize $\overset{r}{I}_{pr}$
        ii. Repeat the sub-steps b and c from II Training and feature extraction for images $\overset{r}{I}_{RS}$
End
```

VI. *Predict the facial expression*$\{\boldsymbol{\varepsilon_c} = \boldsymbol{\varepsilon_1}\, \boldsymbol{\varepsilon_2}\, _{\dots}\, \boldsymbol{\varepsilon_7}\ \}$ *Where the basic classes of seven emotions are represented as* $\boldsymbol{c}$ ***= [Angry, disgust, fear, Happy, Sad, Surprise, and Neutral].***

Output: Prediction of facial emotion of the subject in real-time

5.2 Second Method

In this method, facial emotion recognition implemented on Raspberry-Pi itself instead of the cloud. The preprocessing step is similar in this method also. The deep network has been trained and imported on Raspberry-Pi, and the process has been speedup via co-processor, i.e., Intel Movidius Neural Compute Stick 2. The detailed architecture using this method shown in Figure 5.11.

Step 1. Selection of Deep Convolutional Neural Network: Working with a resource-constrained device like Raspberry-Pi also needs architectures that do not require less power and occupy lesser space and perform fast processing. So, working with VGG and ResNet is not suitable as they require 200-500 MB that is huge for the resource-constrained device due to their sheer size and capability to perform computations. So, here working architectures like Mobile Nets will work, which are different from conventional CNNs as they use depthwise separable convolution. So, Mini_Xception model has been used to train FER 2013 dataset as this network splits the convolution into two stages. The first stage is a 3X3 depthwise convolution, and the second stage performs 1X1 pointwise convolution. That is the primary point that helps in reducing the number of parameters in the network. The only thing that needs to be compromised is accuracy because these networks are not as accurate as actual CNNs.

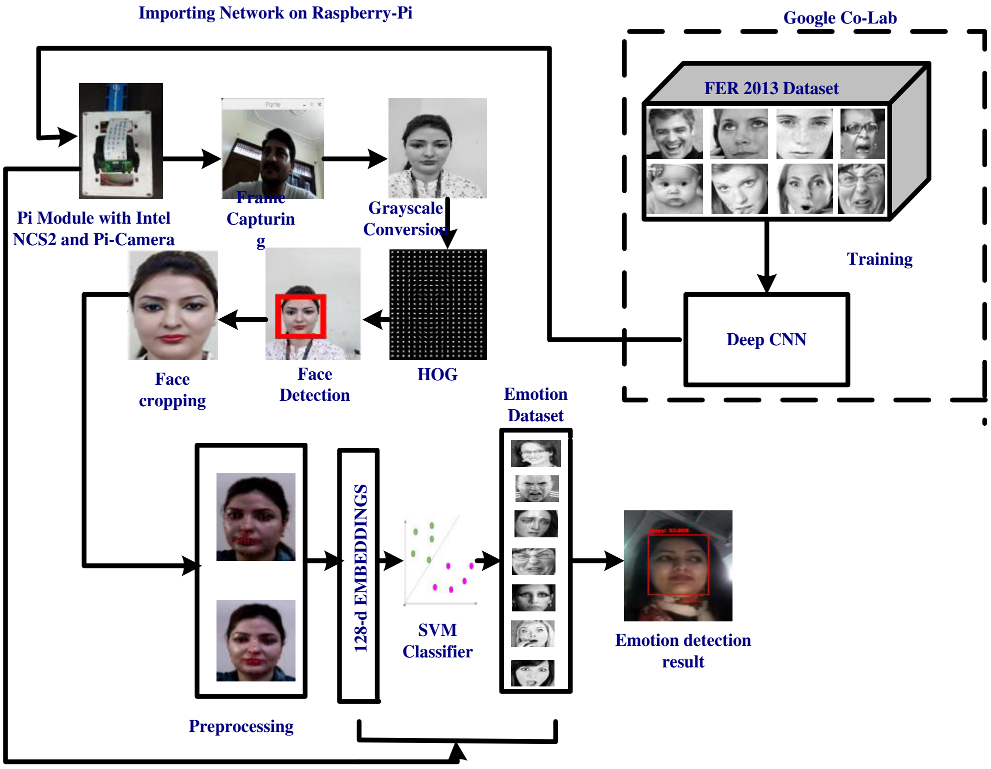


Figure 5.11 Architecture for face recognition and facial emotion recognition in real-time using Raspberry-Pi and Intel Movidius NCS2

Step 2.Training of pre-trained network and importing: As OpenCV 3.3 launched back in 2017 was a highly improved deep neural network module. This module supports quite a several frameworks, which include Caffe, Tensor Flow, and PyTorch/Torch as well. The caffe module has been used in this to import on Raspberry-Pi. The network has been trained via the FER2013 dataset using Google co-lab with K80 GPU. Once the network has been trained --prototxt files which define the model itself having all layers in its and—caffe model file that contain all the weights of actual layers have been imported and parsing of command-line arguments. First, the model is loaded via --prototxt and --model file paths and then stored the model as a net.

Step3.Feeding the pre-processed image: The next step is to feed the preprocessed image into the network. The preprocessing stages are already explained in the first method. The preprocessing includes setting the blob dimensions and normalization.

Step 4. Detecting expression from live video frame: To detect the faces, the blob has been passed through the net, and from that, the detections have been loop over, and the bounding boxes have been drawn around the detected faces Alone with its expression and confidence. The face embeddings are extracted with a pre-trained Mini_Xception model included the face embedding model/directory. At the time of execution, two files were generated named embeddings. Pickle and le. The pickle that is stored inside the output directory. All these embeddings consist of 128-d vector for each expression in the dataset. Finally, for correlation, a Support Vector Machines (SVM) machine learning model on the top of the embeddings has been trained. The result of training our SVM will be serialized to the recognizer. Pickle in the output/ directory.

Detailed Methodologies used to achieve the objectives illustrated diagrammatically, along with a stepwise illustration. Stepwise discussion for two different Methodologies along with Pseudocode helps to understand the workflow in a better way. This chapter concluded HOG as the best descriptor and SVM as the best classifier when implemented in real-time.

CHAPTER 6

SOFTWARE DEVELOPMENT

This chapter explains the critical flowchart of deployment of a pretrained network on Raspberry-Pi. After completion of training using CPU or GPU, it is essential to deploy the network on Raspberry-Pi. Hence the same is explained in this chapter.

The real-time implementation of a facial emotion recognition system has been done using a vision mote. This mote is designed to capture the face and its emotions in real-time. The vision mote is designed with Raspberry-Pi and Pi camera. Raspberry-Pi, along with Neural compute stick, made this system powerful and cost-effective over existing complex and expensive systems. The system has been trained with the FER-2013 dataset, i.e., the facial emotion dataset. The Deep network has been trained and imported on a neural compute stick. A deep learning detector (Base on caffe) has been used to detect the face, and the embedding model (Based on Torch) has been used to take out the facial embeddings for facial emotion detection in real-time. Threshold has been used for filtering week facial emotion detection. Figure 6.1 shows the flow chart of software development for training the dataset using a deep convolution neural network and importing the data on Raspberry-Pi using a neural compute stick.

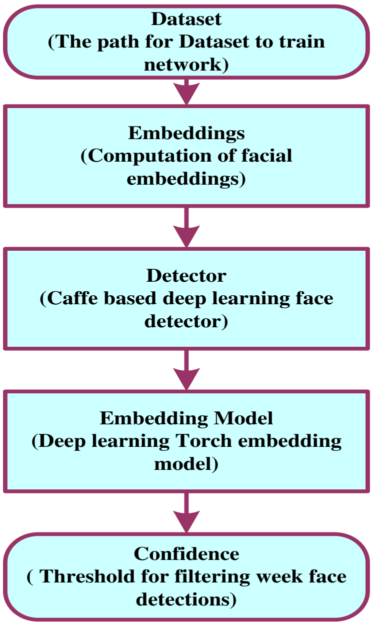

Dataset
(The path for Dataset to train network)
Embeddings
(Computation of facial embeddings)
Detector
(Caffe based deep learning face detector)
Embedding Model
(Deep learning Torch embedding model)
Confidence
(Threshold for filtering week face detections)

Figure 6.1: -Flow chart for software development

With the advancement of technology, it is easy to optimize the pre-trained networks and deploy them on resource-constrained devices like Raspberry-pi. With the development of CV libraries like Keras, Torch, and Caffe, it becomes possible to work with ease using deep networks as well.

CHAPTER 7

HARDWARE DEVELOPMENT

This chapter explains the detailed steps of the hardware development of the system, which is strong enough to detect the facial emotions of human beings in real-time. The complete hardware development is divided into two parts. First part is the vision node which has got camera, Raspberry-pi, Servo motors to provide pan and tilt and co-processor implement deep network, the other part of the system is a wrist band which has got two sensors, i.e., heart rate and BP sensor to record the physiological values of the person and correlated with facial emotions. The Architecture of the vision node is shown in Figure 7.1.

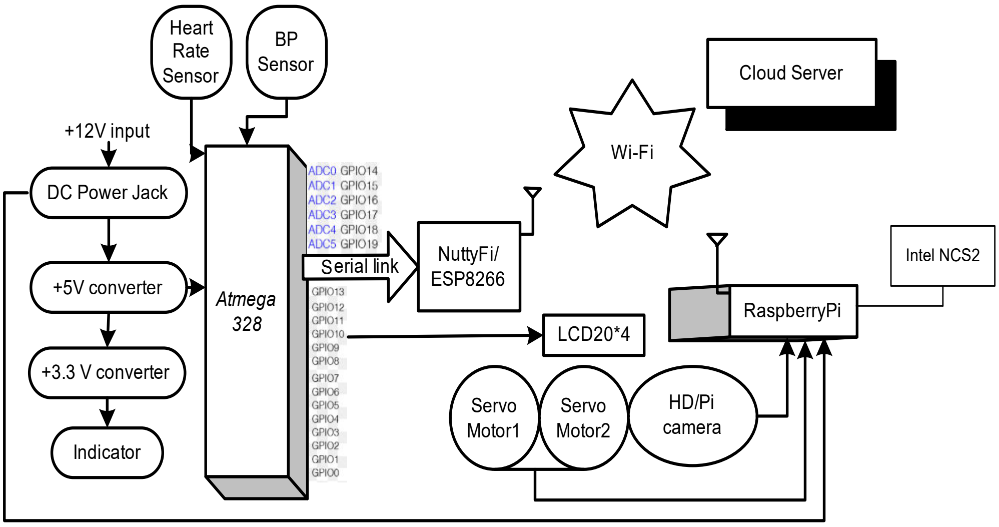


Figure 7.1: Block diagram of vision mote

The central part of the vision node is Raspberry-Pi and Pi camera, as shown in the figure. In the vision node, servo motors were used to pan and turned the device to monitor the face in real-time, and, besides, this node contains both RF Modem and Wi-Fi to store the data in the cloud. The RF modem is used to collect the wearable band data and to transfer the information to the server. The LCD is used to display the captured values from the wrist band. The wrist band has Physiological sensors on it, i.e., BP Sensor and Heart Rate Sensor on it. The sensor values are received by Raspberry-Pi using Pyfirmata and combined with the facial emotion images that are captured via Pi-camera in real-time. The data gathered from both the camera and sensors are then correlated, and conclusive expression and confidence have been extracted to understand the facial emotion of the person in real-time.

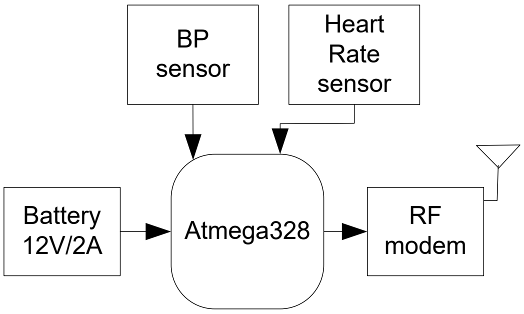


Figure7.2: Block diagram of a Wrist band

Figure 7.2 depicts the block diagram of the wrist band that can collect the physiological values of the person via two different sensors, i.e., BP sensor and Heart Rate sensor. The BP sensor can detect the systolic and diastolic values of heart rate and display the same on the LCD as well as on the cloud. The Heart sensor senses the values of heart rate and displays the same on LCD along with sending all the values on the cloud also. The physiological sensor values collected along with the real-time emotion detected images and conclusion had been taken by looking into the threshold and recorded values on raspberry-pi itself. Figure 7.3 shows the front view and top view of the vision mote.

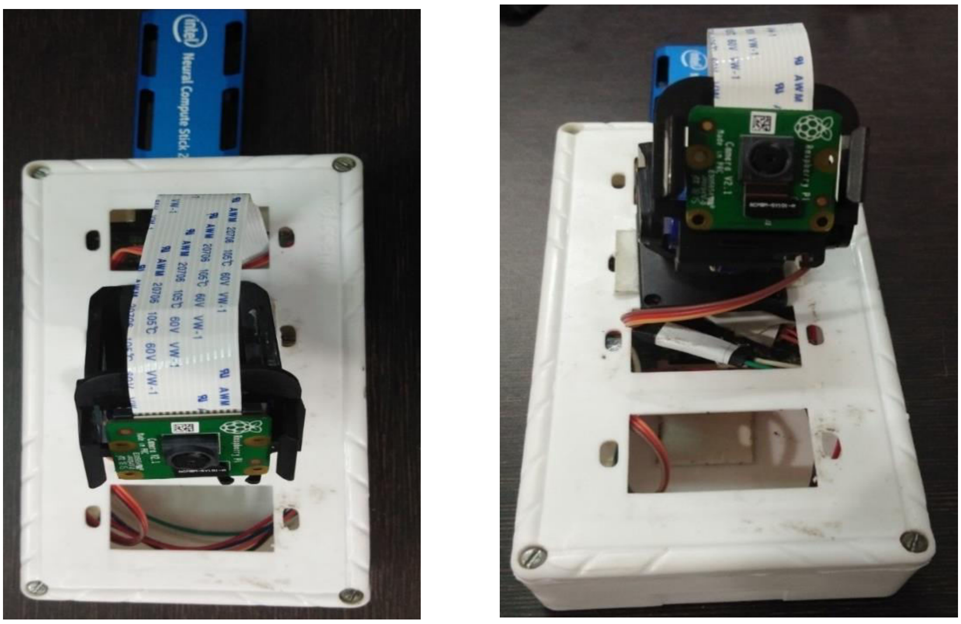


Figure 7.3: a) Top view of vision mote b). Front view of vision mote

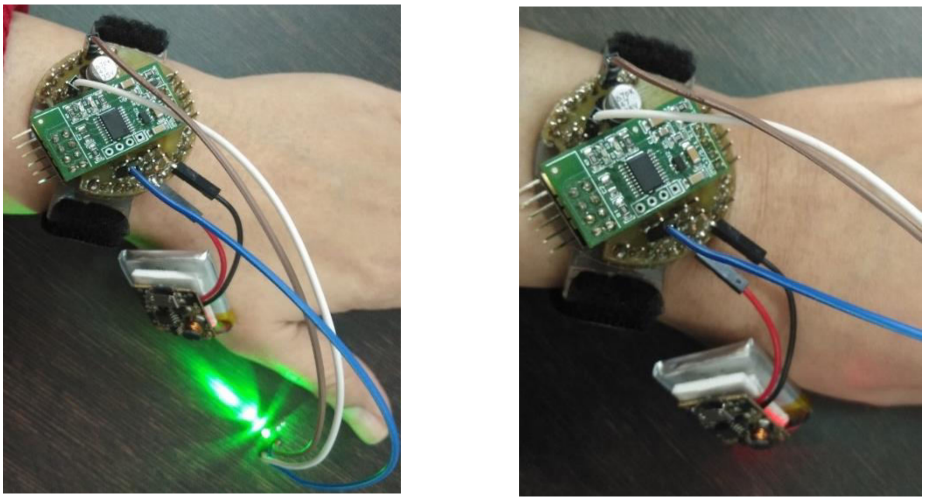


Figure 7.4. a).Wrist band with BP Sensor b). Wrist band with LIPO battery

Figure 7.4 shows the images of the wrist band having both physiological sensors, i.e., heart rate and BP, both attached to it. This is having an RF Modem that is having the capability to send to RF modem in another part of the room. Moreover, the other node has a Wi-Fi module also that is uploading the recorded values on the cloud. The band is working with LIPO battery, so it is having quite a long backup and can work for long. The battery is rechargeable, and deficient power is required to perform that. The vision node is shown in the figure, which is displaying the recorded values and uploading the same on the cloud also via the wi-fi module. The figure shows the real-time value of BP and heart rate.

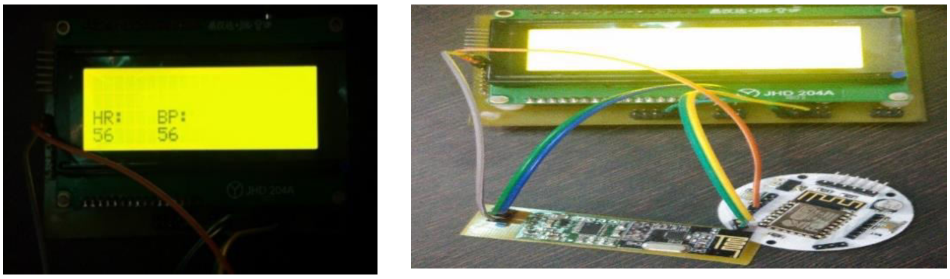


Figure 7.5. a). The display unit of vision node b). Wi-Fi and RF Modem along with vision node

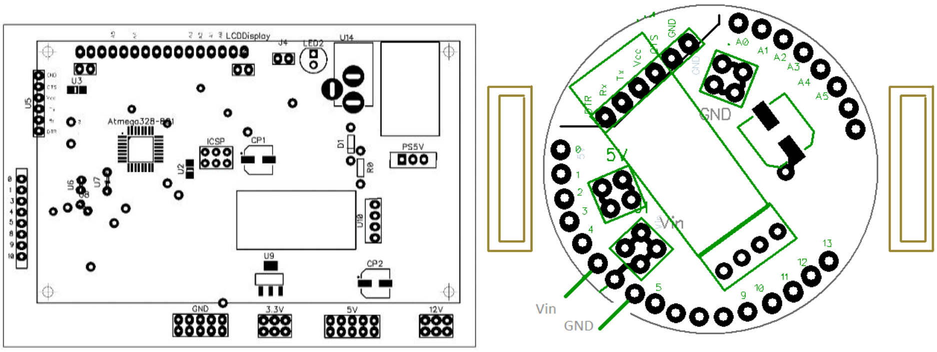


Figure7.6 a).Bit map of RF Modem b).Bit Map of the Wrist band

The complete system has been designed by customization of the boards, and the bit map of the RF modem, i.e., part of the vision mote, is shown in Figure 7.6. The bit map of the wrist band is also shown in the figure, which is having an RF modem in itself also. The threshold value that is used to detect the criticality of the situation is shown in the table. Table 7.1 explains the threshold values for systolic and diastolic values in mm Hg.

Table7.1: Blood pressure categories for ages (18 years and older)

	Systolic (mm Hg)	Diastolic (mm Hg)
Hypotension	<90	<60
Desired	90-119	60-79
Prehypertension	120-139	80-89
Stage 1 Hypertension	140-159	90-99
Stage 2 Hypertension	160-179	100-109
Hypertensive crisis	>=180	>=110

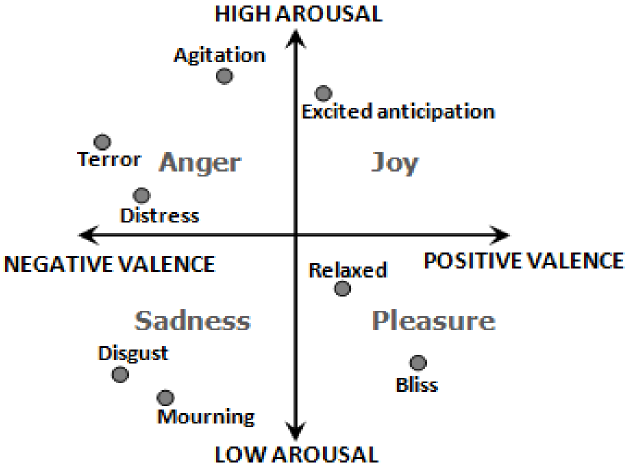


Figure7.7: Illustration of the 2D model based on valance and arousal (Rusell,1976)

Figure 7.7 shows the two-dimensional model based on valance and arousal—the model explaining the essential 4 emotional state and corresponding primary and tertiary emotions.

Figure 7.8 shows the experimental setup that is established to capture the real-time facial emotion of the subjects along with the physiological values that include heart rate and blood pressure. This experimental setup includes a wrist band with physiological sensors like heart rate and BP and vision node as well. Vision node is capturing the facial emotions in real-time as the subjects are made to watch the videos that can take them to various emotional states, as mentioned in table 8.3 to 8.5, and proper time has been given to all the subjects, carry out this work efficiently. It generally takes time to switch from one emotional state to another state. So proper care has been taking in that direction while carrying this experiment

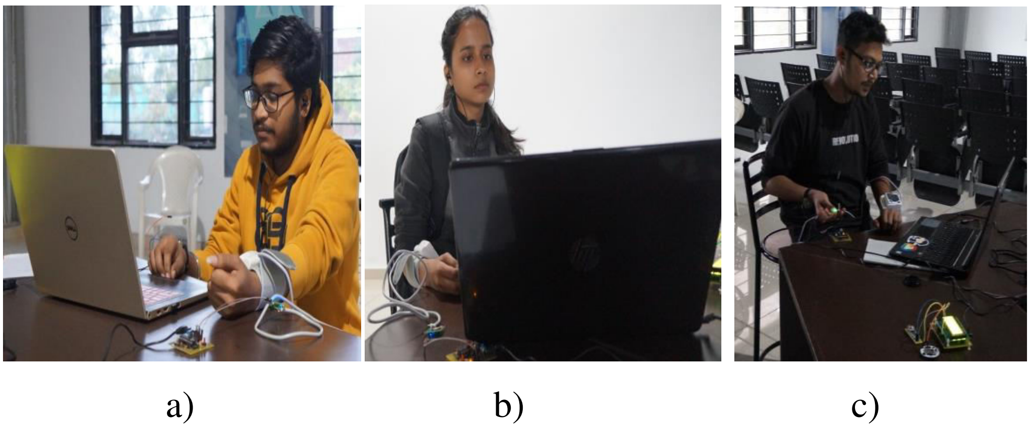


Figure7.8: a). b) and c) Experimental setup done to capture the facial emotions and physiological sensor values in real-time on different subjects.

This chapter gives the insight of a prototype that is designed using customized boards and Raspberry-Pi. For the validation of facial emotions captured by this system, two physiological sensors used. This chapter depicts the complete hardware development process.

.

CHAPTER 8

RESULT AND DISCUSSION

In this chapter, a detailed description of all the experiments is given. The performance of various models is explained in this section. The experiments that have been displayed in this chapter are done on Google-CoLab with 12GB NVIDIA Tesla K80 GPU.

8.1 Experiments on the FER-2013 Dataset

This section explains the experiments conducted on the FER-2013 dataset using different models. The dataset has been tested on various models, and the best performing network has been selected.

8.1.1 Best Performing Deep Networks

Mobile Nets are low-latency, low power, efficient and small models that are designed to work with resource constraints applications. These networks are beneficial in terms of feature extraction, object detection, and segmentation. MobileNetV2 is having two types of blocks. The first one is the residual bock, which is having a single stride. Another block is having a stride of 2, and the purpose is for downsampling. Both of these blocks are having three layers for both types of blocks, as shown in Figure 8.2. MobileNet_V2 is the second version of Mobile Net and has got two features, i.e., a Linear bottleneck between layers, and the other one is shortcut connections between those bottlenecks, as shown in Figure 8.1. The generalized description of layers is shown in Table 8.1. The first layer is a Convolutional layer of 1x1 with ReLU, and the second layer is a depth-wise convolutional layer of 3x3 with ReLU, and the third layer is again a 1x1 convolutional layer but without any ReLU, i.e., without any non-linearity.

Table 8.1 Layers description of MobileNet_V2

Input	Operator	Output
$h \times w \times \mathrm{k}$	$1 \times 1\ conv2d, ReLU6$	$h \times w \times (tk)$
$h \times w \times tk$	$3 \times 3\ dwise\ S = s, ReLU6$	$\frac{h}{s} \times \frac{w}{s} \times (tk)$
$\frac{h}{s} \times \frac{w}{s} \times tk$	Linear $1 \times 1\ conv2d$	$\frac{h}{s} \times \frac{w}{s} \times k'$

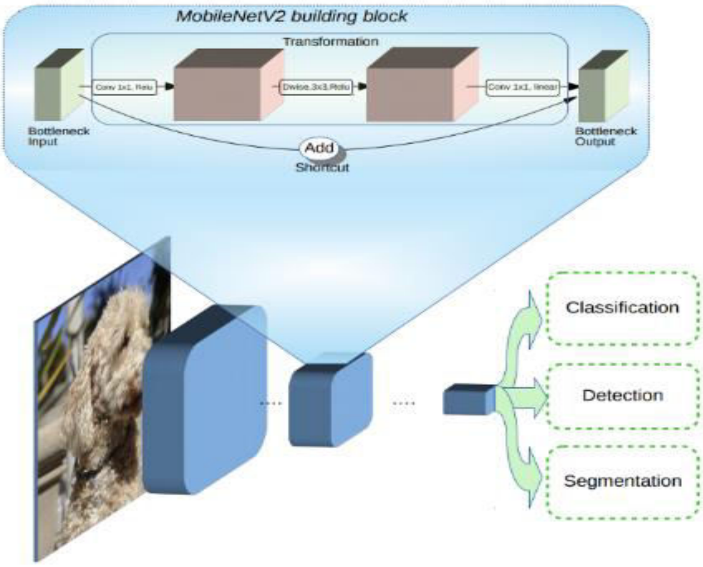


Figure 8.1 Layers of MobileNetV2

The input image, which is *48* × *48* in size, is fed to the first hidden layer of the network, i.e., a convolutional layer with a kernel of *1* × *1* with a stride of 1 both dimensions. The total number of feature maps in this layer is 64, and the output would get $64 \times 4 = 384\ channels$. The output is then fed to a depth-wise separable layer with a kernel of *3* × *3* with a stride of 1 and finally, it is fed to convolutional layer with a kernel of *1* × *1* with a stride of 1. The network is trained via Google Co-Lab in batch mode, using AdaGrad and Adam optimizer and achieves the accuracy of 72.5% with 20 epochs. The performance of this network is shown in Figure 8.3. The dataset has been divided into 80:10:10 for training, validation, and testing. The confusion matrix for the same has also been shown in figure 8.4.

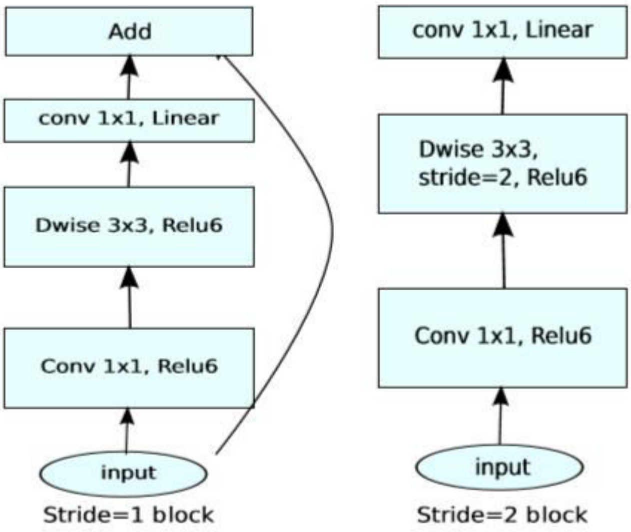


Figure 8.2 Mobile Net V2

The network has been trained with 80% of the dataset using the MobileNet_V2 model. Figure 8.3 shows the accuracy of the model while training the data. The network has reached an accuracy of 72.5 % with 20 epochs, which is quite higher than the accuracy, as mentioned in the state of the art. MobileNet_v2 results are the results that have been achieved on FER 2013 dataset. The training has been done using 20 epochs. Epoch here signifies the iterations or the number the times this network will see the entire data set. The result for the same is shown in Table 8.2 as follows.

Table 8.2 MobileNet_V2 Model results

Epoch	Data Train		Data Validate	
	Acc	Loss	Val ACC	Val Loss
1	0.719426	0.759785	0.638897	1.031655
2	0.71953	0.748763	0.631931	1.025391
3	0.722979	0.742163	0.646141	1.027024
4	0.724268	0.740209	0.636946	1.026796
5	0.724546	0.736157	0.642797	1.020436
:	:	:	:	:

18	0.727507	0.731395	0.639454	1.03757
19	0.725208	0.734367	0.640011	1.035657
20	0.725974	0.731205	0.63611	1.048003

Table 8.2 shows the result of data train training and data test using 20 epochs the values are not changing significantly for each epoch and results are satisfactory as the accuracy has reached a maximum of 72.5% while training, on the other hand, it has reached a maximum of 63.6% during validation. Figure 8.3 shows two graphs that show the Model accuracy and model loss while training and testing on the FER dataset. The first graph shows the accuracy while the second one shows the loss.

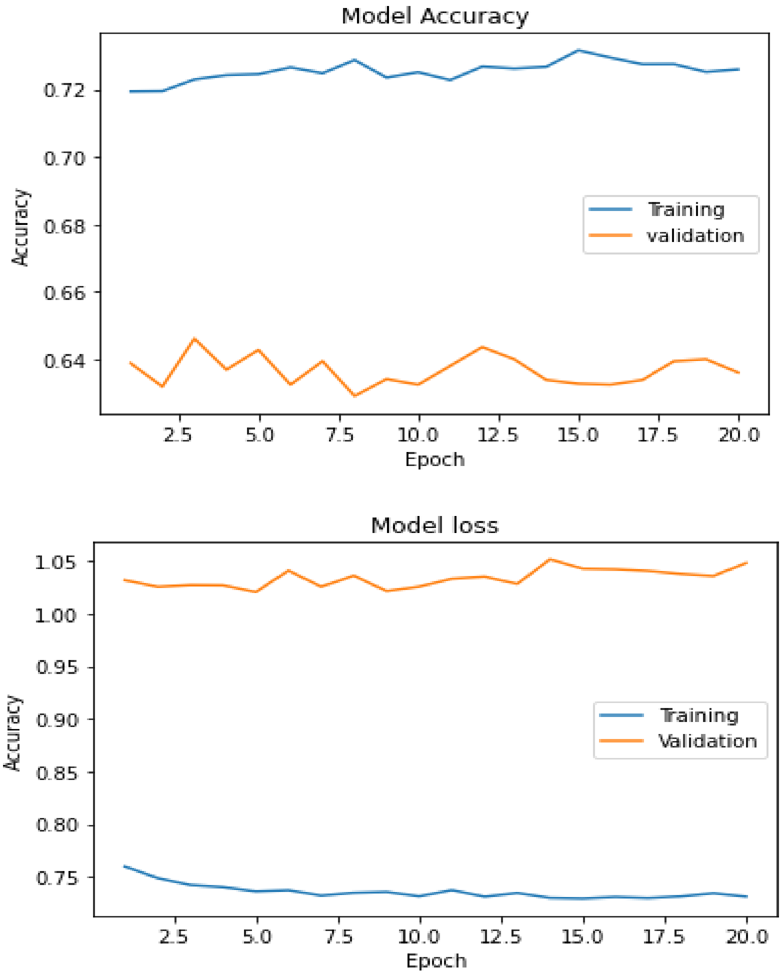


Figure 8.3 Model Accuracy and Model Loss of MobileNetV2 using FER 2013 dataset

The performance the network can be viewed from figure 8.4 as 94% of the sad faces are classified correctly, and 78% of the happy faces are classified correctly,59% of the surprised faces are classified correctly, but the classification of angry, disgust, fear and neutral faces is not correct. As the significant contribution of this thesis is to detect the stress that is detected via the detection of sad faces. So, this model is not accurate for detecting basic 7 expressions, so another lite model has also been identified.

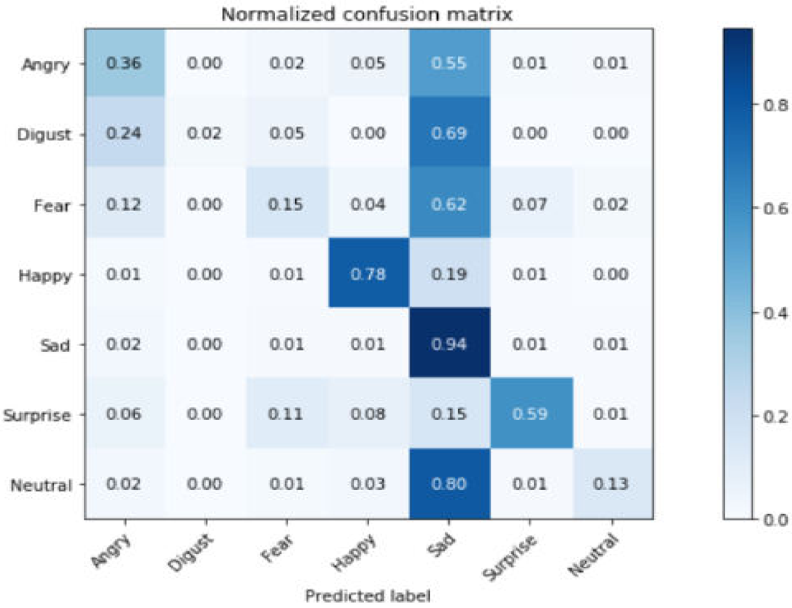


Figure 8.4 Confusion matrix with Mobilenet_V2 Model

Another model that has been used for training is Mini_Xception model. This is a modified depthwise separable convolutional neural network. In comparison with the conventional convolution neural network, this particular model does not require to perform convolution across all the channels. This particular thing is making this model lighter and also reduces the connections that are very less in comparison to conventional models. The model architecture of Mini_Xception model is shown in figure 8.5. The primary point of benefit in this architecture is that it does not contain any fully connected layers, and the inclusion of depthwise separable convolutions helps to minimize the count of parameters. The introduction of residual models also enables the gradients to perform better in backpropagation to lower layers. The network is trained via Google Co-Lab in batch mode, using SGD and Adam optimizer and achieves the accuracy of 69% with 35 epochs. The performance of this network is shown in figure. The confusion matrix for the same has also been shown in Figure 8.5.

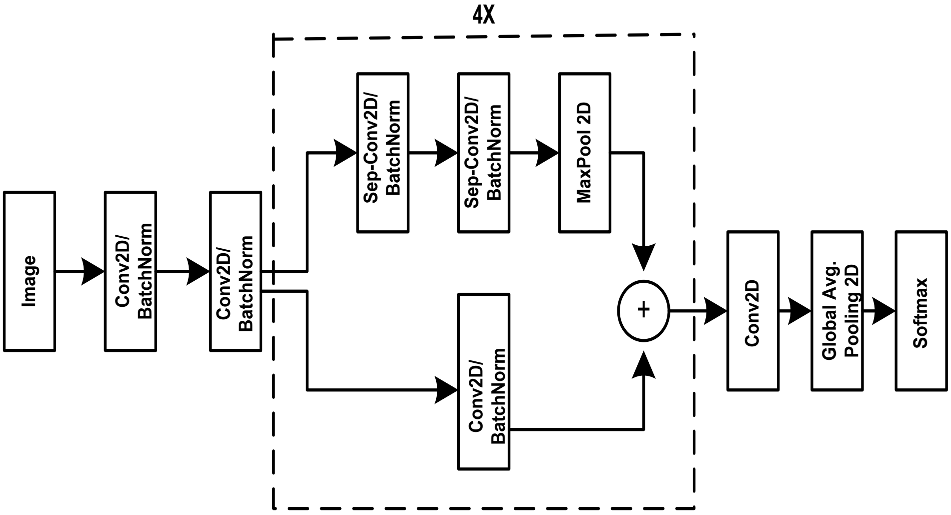


Figure 8.5 Mini_Xception model Architecture

The Mini_Xception model contains 4 extra depth-wise separable convolution layers, and a batch normalization operation follows each convolutional layer, and the ReLU activation function and the final layer is global average pooling with a soft-max activation function. The performance of the network can be viewed in figure 8.6. The data has been divided into 80:10:10 for training, validation, and testing. The model accuracy is shown in figure 8.6 which has reached the training accuracy of

73 %.The accuracy that has been achieved with the model using 35 epochs is quite high and can be considered for the deployment on the system for real-time facial emotion detection. Figure 8.7 shows the training loss, and from the graph, it is visible that loss is decreasing exponentially, and till the 35th epoch, the loss has reduced to a minimum. The Confusion Matrix of the model is shown in figure 8.8. From the confusion matrix, it has been seen that the disgusted faces are misclassified as angry faces, and the reason behind that is the count of disgusted faces in the dataset is least. The primary reason behind the misclassification is the nonuniform dataset. That’s what the FER 2013 dataset is distributed.

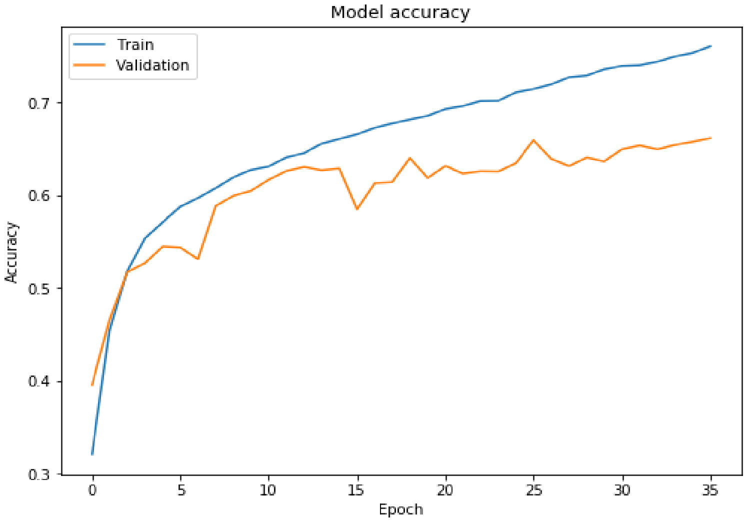


Figure 8.6 Training Accuracy of Mini_Xception model Architecture

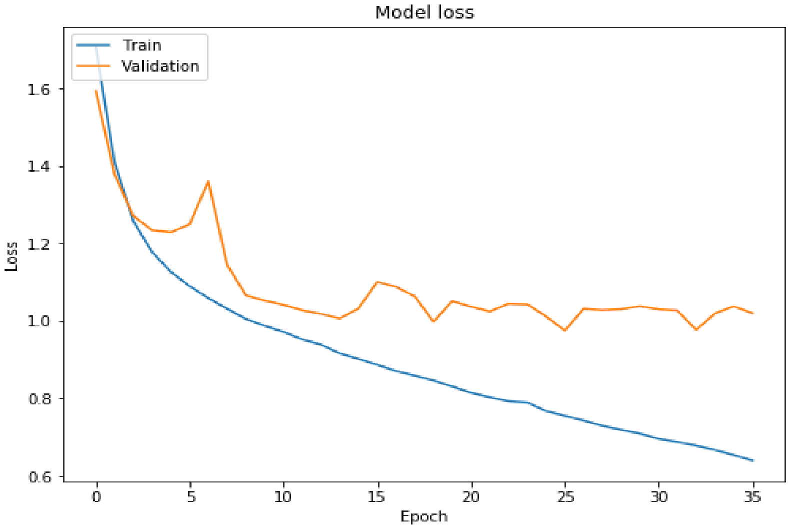


Figure 8.7 Training loss of Mini_Xception model Architecture

Table 8.3 shows the result of data train training and data test using 35 epochs the values are changing significantly for each epoch and results are satisfactory as the accuracy has reached a maximum of 73% while training, on the other hand, it has reached a maximum of 65.6% during validation. Figure 8.6 and 8.7 shows graphs which show the accuracy and loss of model during training and testing on FER dataset. The top graph shows the accuracy while the bottom one shows the loss.

Table 8.3 Mini_Xception Model results

Epoch	Data Train		Data Validate	
	Acc	Loss	Val ACC	Val Loss
1	0.4408	1.4444	0.4252	1.4744
2	0.4410	1.4450	0.4255	1.4742
3	0.5089	1.2818	0.4260	1.4740
4	0.5409	1.2011	0.5043	1.2904
5	0.5666	1.1405	0.5205	1.2344
:	:	:	:	:
33	0.7275	0.7417	0.6394	1.0075
34	0.7252	0.7341	0.64001	1.1055
35	0.7319	0.7112	0.65611	1.1156

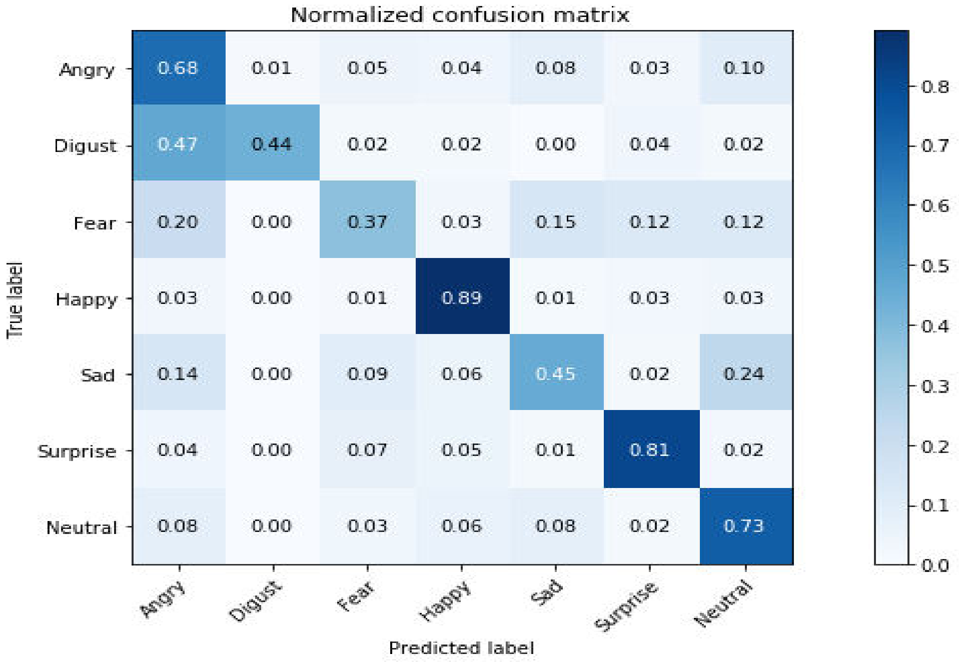


Figure 8.8 Confusion matrix with Mini_Xception Model

8.1.2. Classification and Accuracy

Classification is the stage where the model that has trained using dataset classifies the validation or testing data .this is done automatically and gave the result as a confusion matrix, as shown in Figure 8.4 and Figure 8.8 for two different models, while training has been done. The result of this stage is the level accuracy in terms of validation, where accuracy is a percentage of the test data classified into the correct class and can be calculated as follows.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (8.1)$$

Where FP is False Positive, FN is False Negative TP is True Positive, and TN is True Negative

The sample images from the FER2013 dataset has been shown in figure 8.9. The entire dataset consists of 35,887 images of 48X48 resolution. The entire dataset is classified into 7 different classes, and those classes are ‘Angry,’ ‘Disgust,’ ‘Fear,’ ‘Happy,’ ‘Sad,’ ‘Surprise,’ and ‘Neutral.’ The FER dataset is a nonuniform dataset with 7 basic emotions, and that is the main reason that the accuracy of the system reported in state of the art is quite less as compa1red to the remaining datasets. The total count of images in every class is shown in figure 8.10. From the graph, it is clear that the minimum count of images in the dataset is disgusting images. The reason behind the less confidence for disgusted faces is clear from the count of disgusted faces in the dataset.

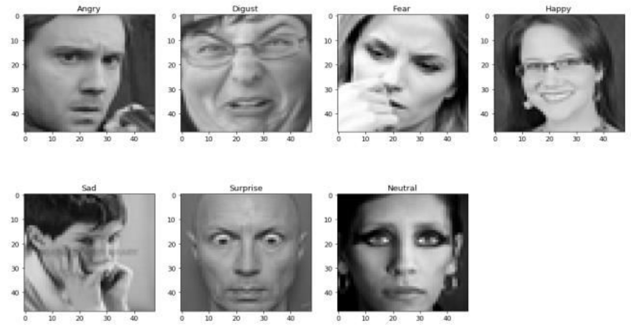
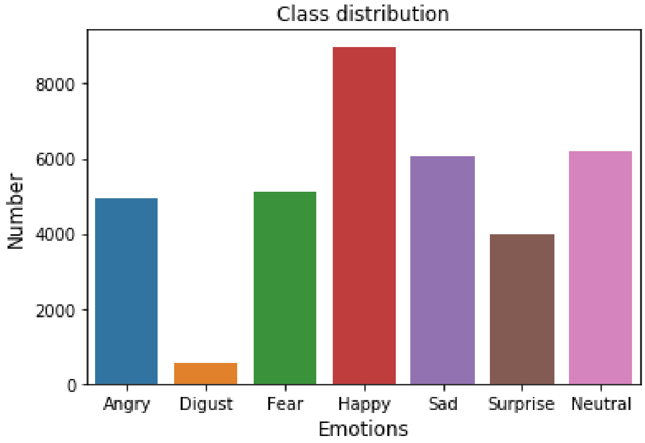
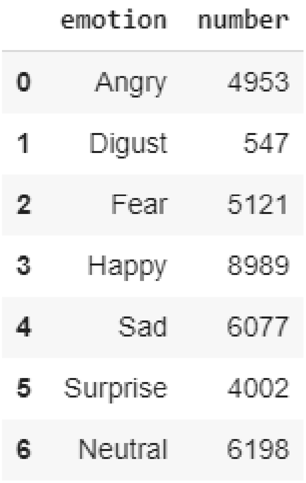


Figure 8.9 Sample images from FER2013 dataset





```
   emotion  number
0    Angry    4953
1   Digust     547
2     Fear    5121
3    Happy    8989
4      Sad    6077
5 Surprise    4002
6  Neutral    6198
```

Figure 8.10 a).Class distribution of FER-2013 dataset as basic 7 classes b).The exact distribution of images under 7 basic emotions

Instead of Mini_Xception and Mobilenet_V2 model, the dataset has been trained on Densenet161 and ResNet Model also. The results for these models are quite less as compared to the previous two models. So, these two models are not considered for further consideration as the accuracy is quite less in comparison to the accuracy with the other two models. The brief description of the model is shown in Table 8.2, which explains the name of the Model, Accuracy, Learning Rate, Test accuracy, and the optimizer used for the model.

Table 8.4 Various Models tested on FER2013 Dataset

Model	Accuracy	Learning rate	Test accuracy	optimizer
Mini_Xception	73%	0.005	69%	Adam, SGD
Densenet161	59%	0.001,0.001,0.005	43%	Adam, SGD
Resnet38	68%	0.0001	60%	SGD, AdaGrad
Mobilenet_V2	72.5%	0.0001,0.001	64%	AdaGrad, Adam

As the entire dataset has been divided into training, validation, and testing set, the graphical representation of the data segregated in the ratio of 80:10:10 is shown in Figure 8.11. The accuracy achieved from the models discussed above is based on the same segregation of datasets.

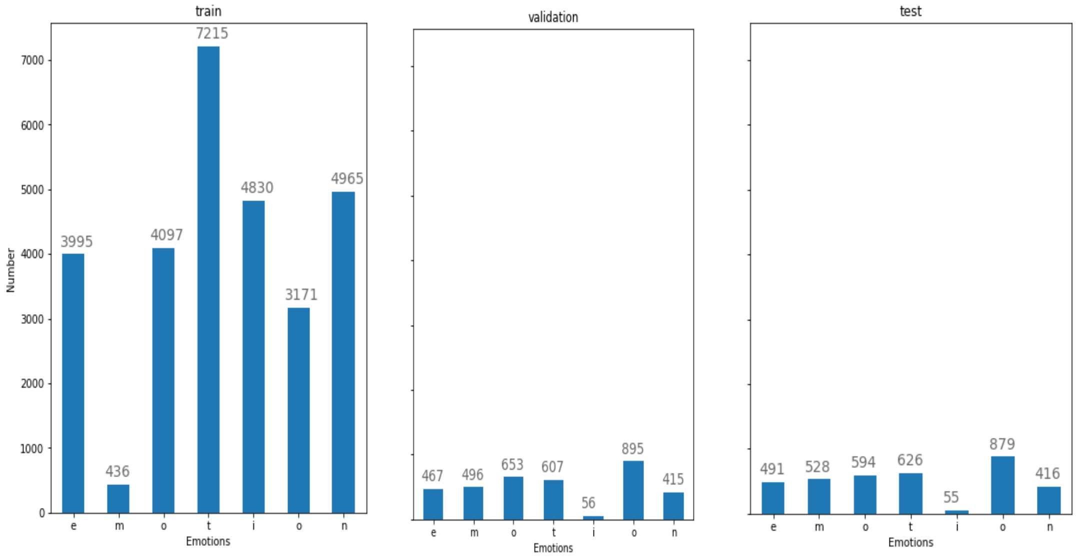


Figure 8.11. Graphical representation of data segregated for training, validation, and testing

The model that has been imported on raspberry-pi for understanding emotions in real-time is Mini_Xception model. The data that has been fed and captured in real-time has been passed through a preprocessing algorithm. The preprocessing algorithm has already been discussed in the Methodology chapter. The results are discussed further in the subsequent sections. Figure 8.12 shows the face transformation results that help to align the face and correctly center the facial image. The facial part has been located even if the face is tilted or rotated. Because the landmarks have been estimated using a cascade of the regressor. The final result of landmark estimation has been shown in figure 8.13, which shows that this will able to locate the faces even if the face is tilted. So, the face and the facial parts can be easily located via this, and this method is beneficial in real-time applications because this will take around one millisecond to locate the faces on the video or images. Such an algorithm is constructive with a resource-constrained device like Raspberry-Pi.

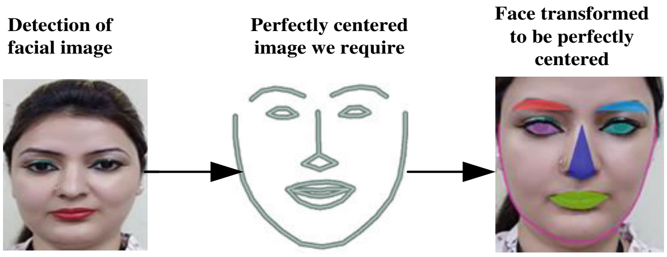
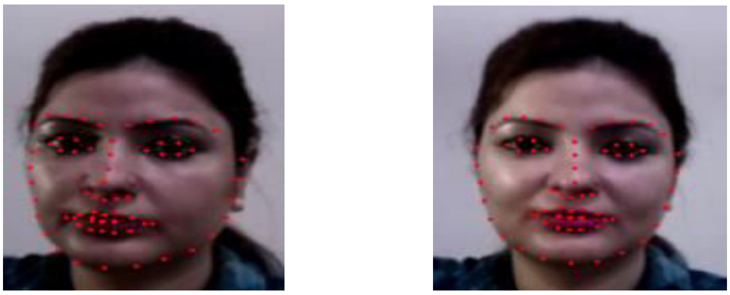

Detection of facial image
Perfectly centered image we require
Face transformed to be perfectly centered

Figure 8.12 Face transformation results

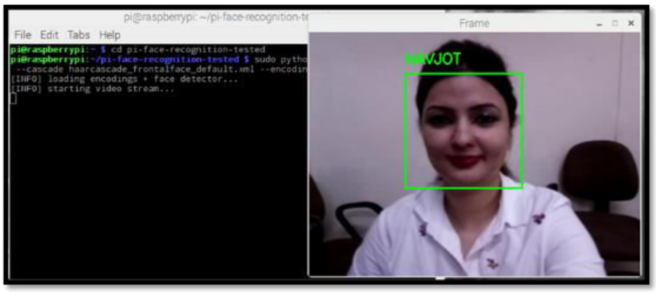


a)

b)

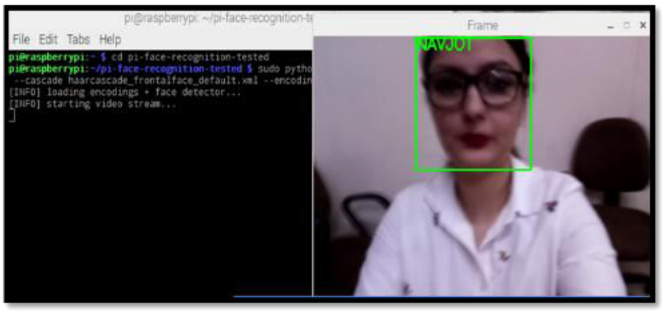
Figure 8.13 Landmark detection results using ensemble of Regression in real-time). Tilted face b) Straight Face

The system has been tested to identify the faces in real-time using Raspberry-Pi. The results of face detection, as shown in figure 8.14, are real-time detection results. The system is capable of detecting the facial images in real-time, even with the presence of objects like spectacles on the face. The system is efficient, and the accuracy of the system is dependent on the dataset that has been used for training purposes. For only face detection in real-time, the dataset should be maximum in number. The maximum is the dataset higher will be the classification accuracy.



```
pi@raspberrypi:~ $ cd pi-face-recognition-tested
pi@raspberrypi:~/pi-face-recognition-tested $ sudo pytho
 --cascade haarcascade_frontalface_default.xml --encodin
[INFO] loading encodings + face detector...
[INFO] starting video stream...
```

a)

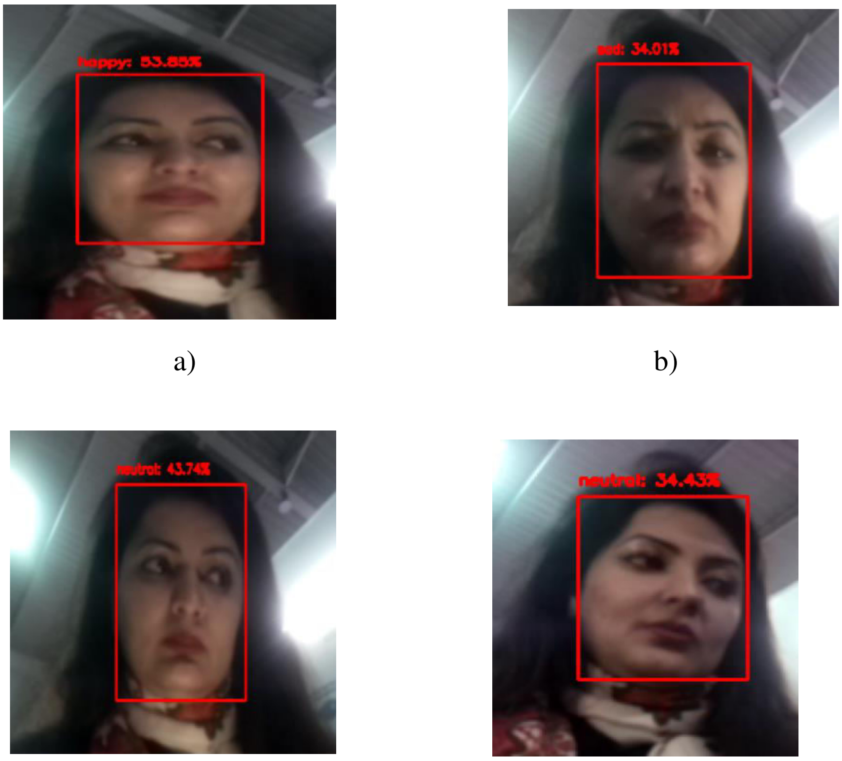


```
pi@raspberrypi:~ $ cd pi-face-recognition-tested
pi@raspberrypi:~/pi-face-recognition-tested $ sudo pytho
 --cascade haarcascade_frontalface_default.xml --encodin
[INFO] loading encodings + face detector...
[INFO] starting video stream...
```

b)

Figure 8.14 Face detection in real-time a) Without spectacles b). With Spectacles

The next important task is to detect facial emotion in real-time. To get this, a pre-trained deep network, discussed in the previous section, has been imported using Neural compute stick on Raspberry-Pi. The results of emotions detected in real-time are shown in figure 8.15.



a)

b)

c)

d)

Figure 8.15 Emotion detection results in real-time using Raspberry-Pi a). Happy b). Sad c) and d) Neutral

The results shown in Figure 8.15 are the results captured in real-time video on Raspberry-Pi with Intel Movidius Neural Compute Stick. The emotions detected in real-time are showing the confidence of the emotions detected. As shown in figure 8.15 a) the face seems to happy and detected as happy face with a confidence of 53.85 % and in b) the face seems to be sad and detected as sad with a confidence of 34.01% in figure c) and d) the face seems to be neutral and detected as neutral with variation in confidence, i.e., 43.74 % and 34.43% and detected correctly as neutral faces. Now, the next section discusses the experimental setup and the emotion detection in real-time using physiological sensors like Blood pressure and heart rate sensor. The setup has been used to detect the facial emotions in real-time and validate the same via the physiological sensors. A wrist band which is designed to record the physiological values like heart rate and Blood pressure of the subject under various situations.

The subjects are set to see the videos that help them to enter in various emotional states, and then the heart rate and Blood pressure of those subjects along with facial expression have been captured at the same time. The recorded values with time stamp and facial images with timestamps are then used further to validate the expression recorded via the system. The system has been designed particularly to detect the emotions with two tiers of validation. Firstly, via facial images only and then for further validation of the extracted emotions, physiological sensors have been used. The experimental values of 20 different subjects under different emotional states are recorded, and Table 8.6 to Table 8.9 shows the recorded values of 3 different subjects for basic emotions like anger, neutral, happy, and sad. Table 8.5 shows the description of videos that have been used to carry out the experiment using 20 different subjects. Three different videos that can bring any person in a happy state is mentioned in this table. The duration of each video also mentioned in all the tables.

Table 8.5 Details of Happy Videos for Experimental Analysis

S.No	Name of Video	Duration of Video	Video Type
1	Tom and Jerry	25:59:00	Happy
2	Mr.Bean at the Dentist	27:20:00	Happy

3	Contagious Laughter Compilation	16:00:00	Happy

Table 8.6 shows the description of videos that have been used to carry out the experiment using 20 different subjects. Three different videos that can bring any person in a sad state are mentioned in this table. The duration of each video also mentioned in the table.

Table 8.6 Details of Sad Videos for Experimental Analysis

S. No	Name of Video	Duration of Video	Video Type
1	Sridevi Funeral video	14:54	Sad
2	Mumbai Terror Attack videos	16:00	Sad
3	She tells her story of why she fled away from North Korea	9:38	Sad

Table 8.7 shows the description of videos that have been used to carry out the experiment using 20 different subjects. Three different videos that can bring any person in an angry state mentioned in this table. The duration of each video also mentioned in each table.

Table 8.7 Details of Angry Videos for Experimental Analysis

S.No	Name of Video	Duration of Video	Video Type
1	Best of Angry people	14:29	Angry
2	Nirbhaya’s Mothers Interview	17:25	Angry
3	Pit Bull Terrier Dog Attacks	9:38	Angry

Table 8.8 Experimental results of three subjects under happy conditions

S. No	Subject Number	Age	Blood Pressure (mm Hg)		Heart Rate (BPM)	Expression	Timestamp
			Systolic	Diastolic			
1	Subject 1	20	130	101	91	HAPPY	4:27:30
2	Subject 1	20	130	100	91	HAPPY	4:30:20
3	Subject 1	20	132	102	94	HAPPY	4:33:30
4	Subject 1	20	135	114	91	HAPPY	4:35:20
5	Subject 1	20	139	112	102	HAPPY	4:38:35
6	Subject 2	22	122	108	98	HAPPY	4:41:18
7	Subject 2	22	126	96	80	HAPPY	4:44:35
8	Subject 2	22	128	93	83	HAPPY	4:47:30
9	Subject 2	22	126	96	80	HAPPY	4:44:30
10	Subject 2	22	128	93	90	HAPPY	4:47:24
11	Subject 3	21	145	97	79	HAPPY	4:50:24
12	Subject 3	21	143	99	83	HAPPY	4:53:12
13	Subject 3	21	128	93	70	HAPPY	4:56:34
14	Subject 3	21	145	97	75	HAPPY	4:59:16
15	Subject 3	21	143	90	79	HAPPY	5:02:34

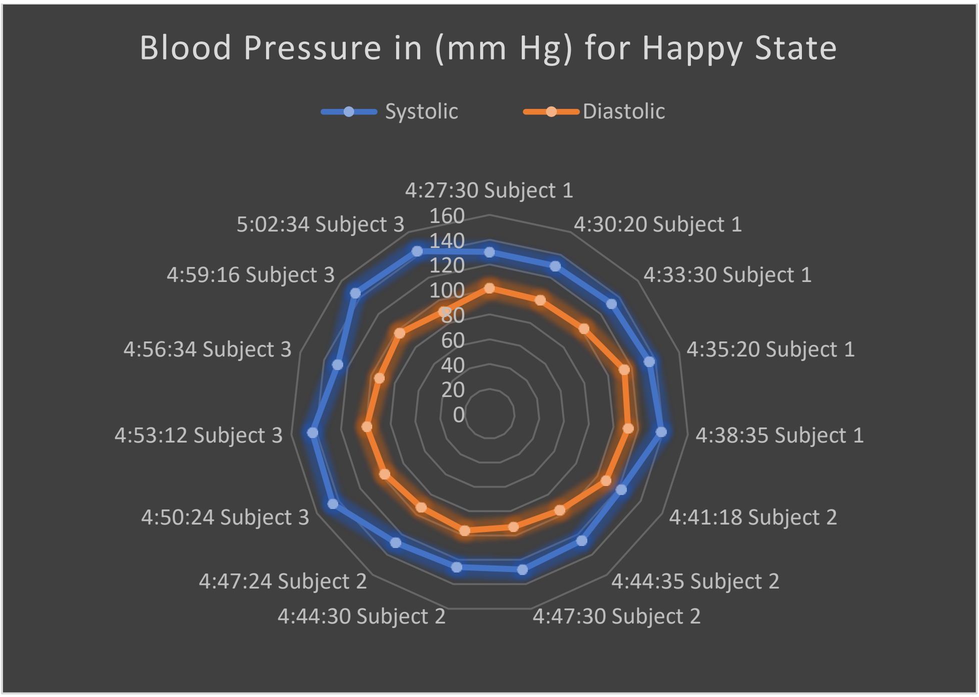


Figure 8.16 Radar plot for variation in Blood Pressure under Happy state

Table 8.9 Experimental results of three subjects under Neutral conditions

S. No	Subject Number	Age	Blood Pressure (mm Hg)		Heart Rate (BPM)	Expression	Timestamp
			Systolic	Diastolic			
1	Subject 1	20	118	79	75	NEUTRAL	5:30:10
2	Subject 1	20	117	79	76	NEUTRAL	5:33:40
3	Subject 1	20	118	77	76	NEUTRAL	5:36:46
4	Subject 1	20	119	77	75	NEUTRAL	5:39:52
5	Subject 1	20	117	79	76	NEUTRAL	5:42:23
6	Subject 2	22	121	80	78	NEUTRAL	5:45:12
7	Subject 2	22	122	80	77	NEUTRAL	5:48:43
8	Subject 2	22	122	79	77	NEUTRAL	5:51:16
9	Subject 2	22	120	80	76	NEUTRAL	5:54:24
10	Subject 2	22	119	79	76	NEUTRAL	5:57:20
11	Subject 3	21	103	63	61	NEUTRAL	6:00:25
12	Subject 3	21	103	62	61	NEUTRAL	6:03:10
13	Subject 3	21	105	66	70	NEUTRAL	6:06:30
14	Subject 3	21	105	66	71	NEUTRAL	6:09:50
15	Subject 3	21	107	63	74	NEUTRAL	6:12:10

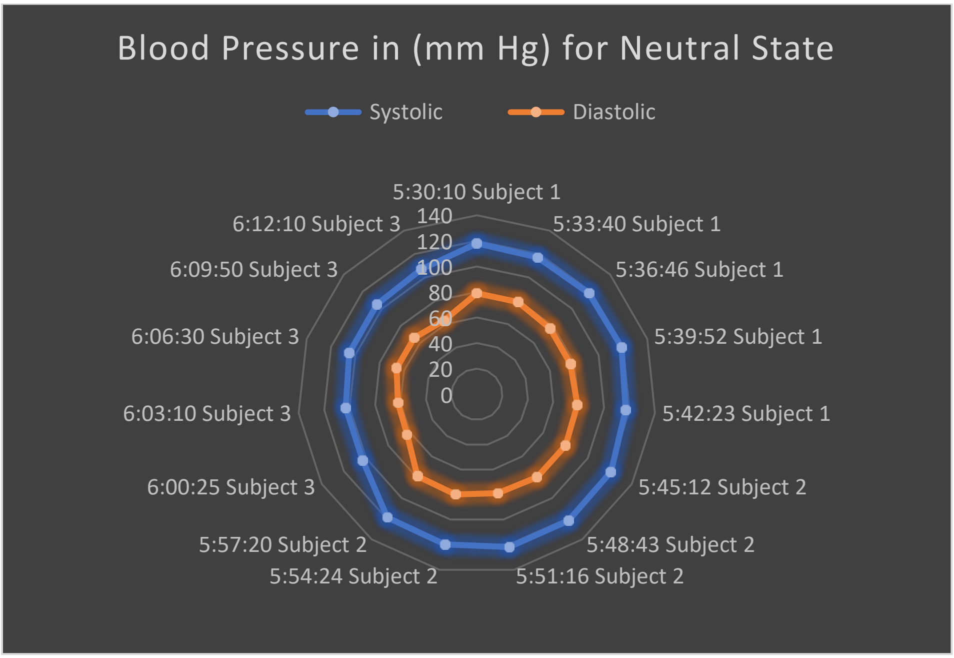


Figure 8.17 Radar plot for variation in Blood Pressure under Neutral state

Table 8.10 Experimental results of three subjects under Angry conditions

S. No	Subject Number	Age	Blood Pressure (mm Hg)		Heart Rate (BPM)	Expression	Timestamp
			Systolic	Diastolic			
1	Subject 1	20	136	109	85	ANGRY	6:45:00
2	Subject 1	20	135	108	84	ANGRY	6:48:20
3	Subject 1	20	137	109	86	ANGRY	6:51:43
4	Subject 1	20	137	109	87	ANGRY	6:54:20
5	Subject 1	20	136	109	87	ANGRY	6:57:32
6	Subject 2	22	140	113	102	ANGRY	7:00:10
7	Subject 2	22	129	110	98	ANGRY	7:03:23
8	Subject 2	22	130	110	98	ANGRY	7:06:32
9	Subject 2	22	128	109	100	ANGRY	7:09:43
10	Subject 2	22	142	112	105	ANGRY	7:12:32
11	Subject 3	21	138	110	108	ANGRY	7:15:10
12	Subject 3	21	140	112	109	ANGRY	7:18:34
13	Subject 3	21	142	111	107	ANGRY	7:21:42
14	Subject 3	21	138	110	107	ANGRY	7:24:10
15	Subject 3	21	140	112	108	ANGRY	7:27:30

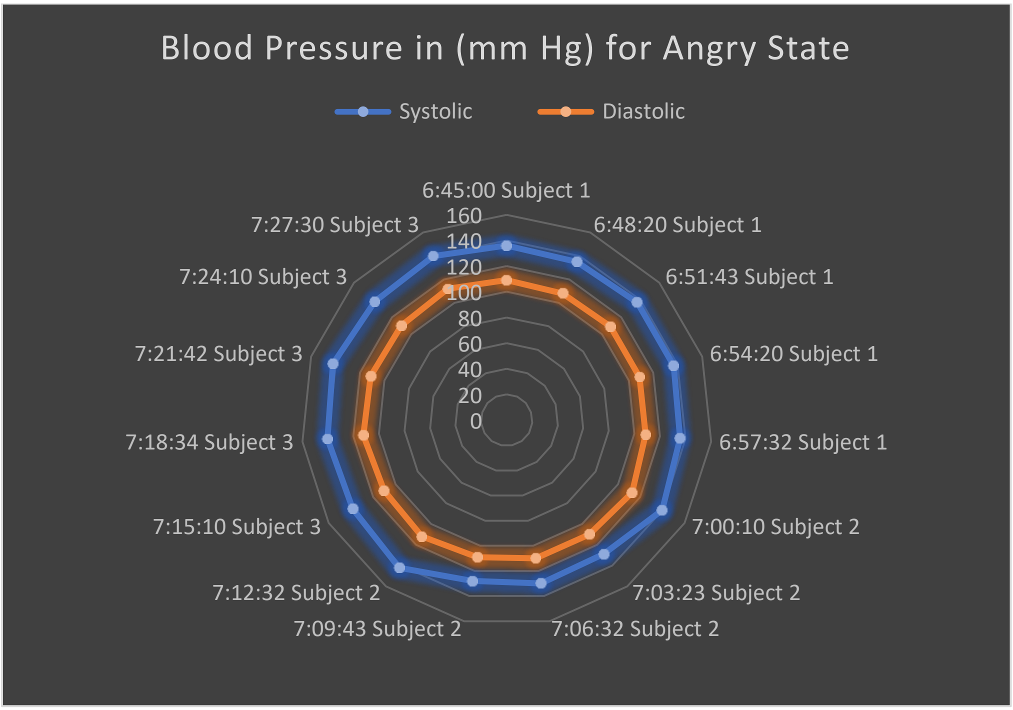


Figure 8.18 Radar plot for variation in Blood Pressure under Angry state

Table 8.11 Experimental results of three subjects under Sad conditions

S. No	Subject Number	Age	Blood Pressure (mm Hg)		Heart Rate (BPM)	Expression	Timestamp
			Systolic	Diastolic			
1	Subject 1	20	90	70	100	SAD	5:00:00
2	Subject 1	20	92	75	110	SAD	5:03:30
3	Subject 1	20	94	81	100	SAD	5:06:20
4	Subject 1	20	94	80	90	SAD	5:09:43
5	Subject 1	20	92	76	92	SAD	5:12:20
6	Subject 2	22	92	79	100	SAD	5:15:34
7	Subject 2	22	90	80	93	SAD	5:17:43
8	Subject 2	22	92	80	97	SAD	5:20:20
9	Subject 2	22	90	79	99	SAD	5:23:42
10	Subject 2	22	92	80	100	SAD	5:26:10
11	Subject 3	21	85	72	99	SAD	5:29:30
12	Subject 3	21	89	75	92	SAD	5:32:40
13	Subject 3	21	88	74	90	SAD	5:35:20
14	Subject 3	21	87	75	90	SAD	5:38:10
15	Subject 3	21	86	72	89	SAD	5:41:30

The values shown in table 8.8 to table 8.11 are recorded under the experimental environment where the subject has made to sit and wear a band that consists of a heart rate sensor and a blood pressure sensor on it. Once the subject has made to wear this sensor, the setup of raspberry-pi with a pi camera has also started to capture the expression of the person in real-time along with the physiological values. Figure 8.20 shows the captured facial expressions with a time stamp.

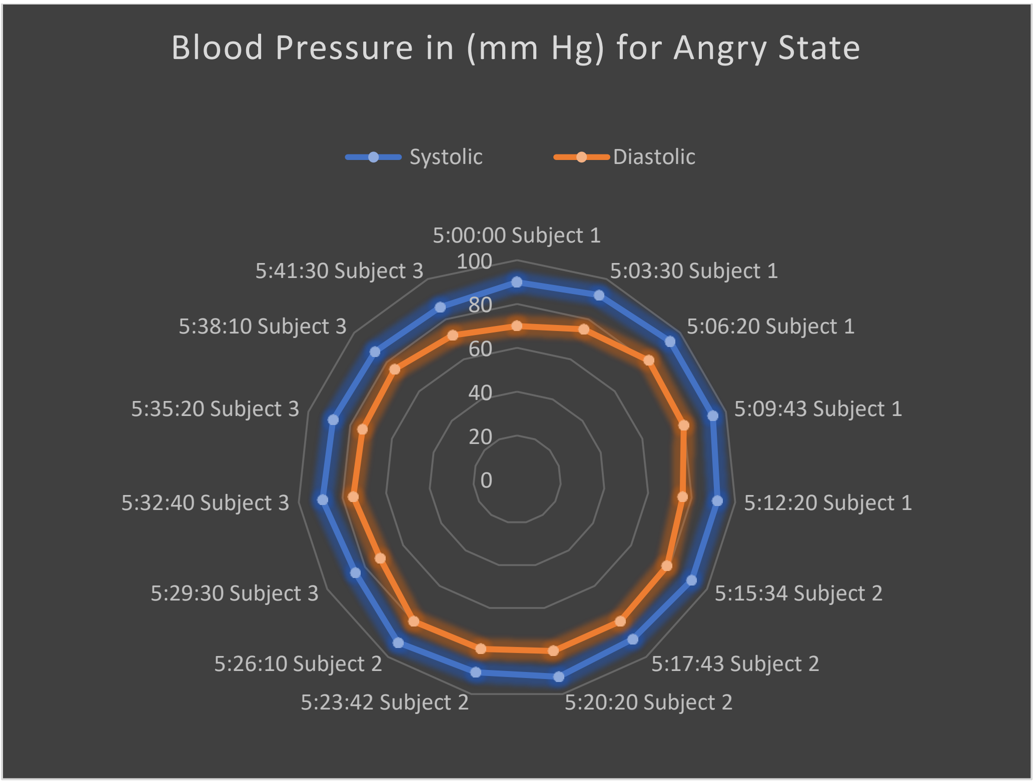
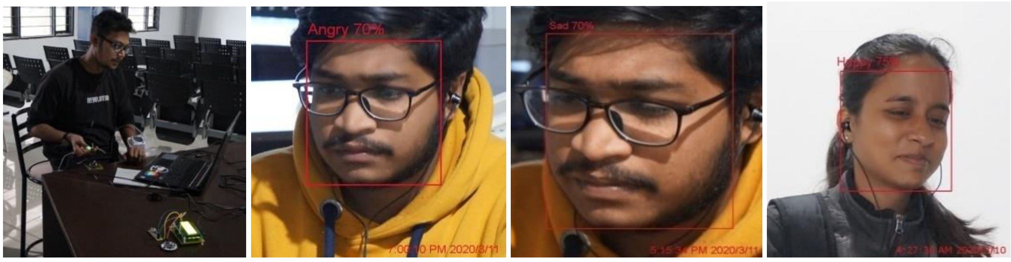


Figure 8.19 Radar plot for variation in Blood Pressure under Sad state



a) b) c) d)

Figure 8.20 a).Experimental setup and b), c), d) expressions captured via Experimental setup with the timestamp

A time-synchronized algorithm used to capture the facial expression and physiological values of the participants, i.e., heart rate and blood pressure. To validate the results, various analyses done. It is found in the literature that emotional arousal increases the systolic and diastolic blood pressure. Moreover, it has also been found in the literature that happiness, anger, and anxiety increases the blood pressure, and the level of variation is dependent upon the individuals. To visualize the physiological values, box plots plotted.

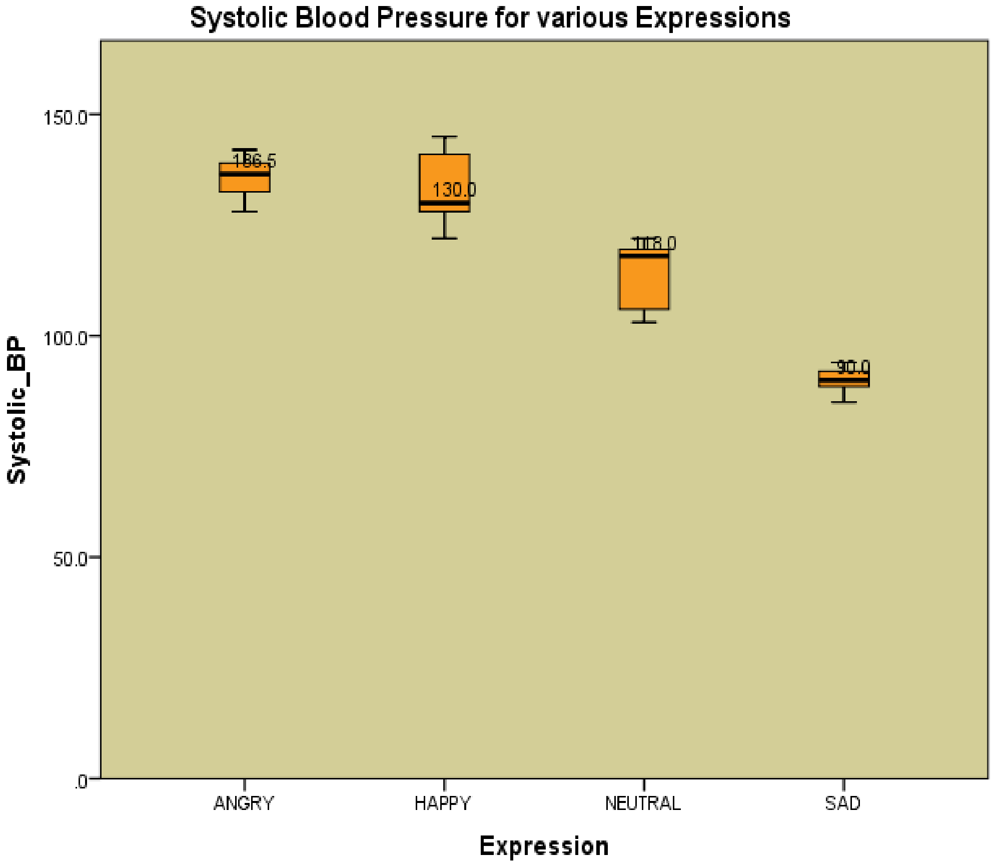


Figure 8.21 Box plot for Systolic Blood Pressure for four basic expressions

Figure 8.21 shows the box plot of the systolic blood pressure of the participants for all the four expressions. From the box plot, it clear that the sadness tends to decrease the systolic blood pressure of the participants to the lowest; on the other hand, anger and happiness tend to increase the systolic blood pressure of the participants. The first quartile and third quartile for each expression also shown that shows the 25% and 75% of the values are lying under these quartiles. The medians for all the recorded values also labeled on the box plot of each expression, which depicts the distribution of the systolic values for that particular expression.

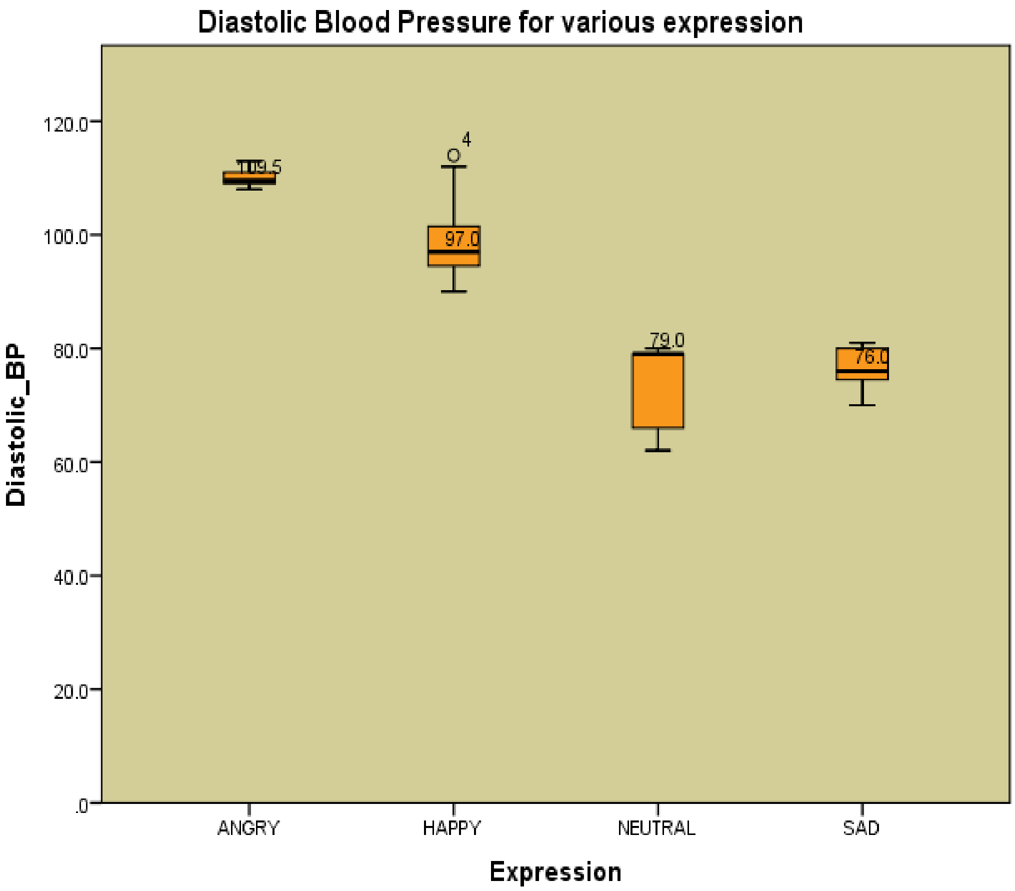


Figure 8.22 Box plot for Diastolic Blood Pressure for four basic expressions

Figure 8.22 shows the box plot of the diastolic blood pressure of the participants for all the four expressions. From the box plot, it clear that the sadness tends to decrease the diastolic blood pressure of the participants to the lowest. On the other hand, anger and happiness tend to increase the diastolic blood pressure of the participants. The first quartile and third quartile for each expression also show that shows the 25% and 75% of the values are lying under these quartiles. The medians for all the recorded values also labeled on the box plot of each expression, which depicts the distribution of the systolic values for that particular expression—one outlier, i.e., fourth recorded value of subject 1 from table 8.6. The value is comparatively high as compared to other recorded value, i.e., 114, has been depicted as an outlier.

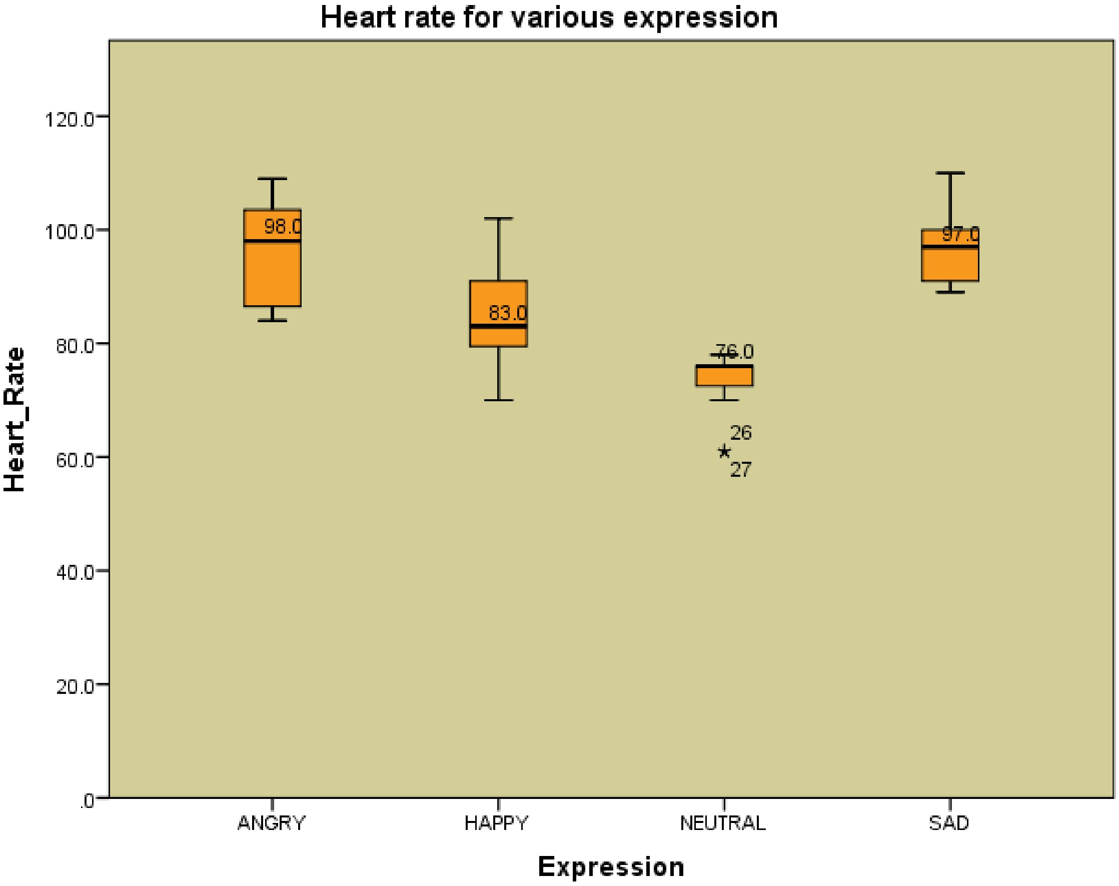


Figure 8.23 Box plot for Heart Rate for four basic emotions

Figure 8.23 shows the box plot of the Heart rate variation of the participants for all the four expressions. From the box plot, it clear that the anger raised the blood pressure of participants to the maximum level while the neutral state has shown the minimum. The first quartile and third quartile for each expression show that shows the 25% and 75% of the values are lying under these quartiles. The medians for all the recorded values also labeled on the box plot of each expression, which depicts the distribution of the Heart rate values for that particular state. Two outliers for the neutral state, i.e., 26th and 27th value recorded and located as 11th and 12th value from table 8.7. Both the values are the same and comparatively low when compared to other recorded values, i.e., 61, hence depicted as an outlier. To validate the variation of physiological recorded values for various mental states, a paired sample t-sample t-test applied. As paired sample t-test is the analytical solution and mainly used when we want to see if the mean difference between the two sets of observations found or not. So to validate our variation on the experimentally recorded values, this test has been utilized.

Paired sample t-test analysis between Happy and Neutral state.

$H_1 =$ There is a significant decrease in the Systolic blood pressure of participants when their emotional state is changing from Happy to Neutral

$$H_1 : \mu_1 - \mu_2 < \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypotheses means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.12, to compare the systolic blood pressure of participants while watching different videos using the experimental setup for 20 participants for Happy and Neutral videos. “There was a significant difference in systolic blood pressure while watching happy videos (M=133.333, SD=7.7429) and systolic blood pressure while watching neutral videos (M=114.400, SD=7.3853) conditions; t (14) =5.157, p= .000”.There was a significant decrease in the systolic blood pressure when participants watched neutral videos after watching happy videos. Hence enough evidence has been found that shows the mean difference between the systolic blood pressure of participants is statistically significant when their emotional state is changing from Happy to Neutral. Hence, the hypothesis accepted which says that there is a significant decrease in the Systolic blood pressure of participants when their emotional state is changing from Happy to Neutral

Table 8.12 Paired Samples Statistics for Systolic BP (Happy to Neutral State)

	Mean	N	Std. Deviation	Std. Error Mean
Systolic_BP_Happy	133.333	15	7.7429	1.9992
Systolic_BP_Neutral	114.400	15	7.3853	1.9069

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Systolic_BP_Happy - Systolic_BP_Neutral	18.933	14.2200	3.6716	11.0585	26.8081

t	Df	Sig. (2-tailed)
5.157	14	.000

H_1 (Alternate Hypothesis)= There is a significant decrease in the Diastolic blood pressure of participants when their emotional state is changing from Happy to Neutral

$$H_1 : \mu_1 - \mu_2 < \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypotheses means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.13, to compare the diastolic blood pressure of participants while watching different videos using the experimental setup for 20 participants for Happy and Neutral videos. “There was a significant difference in diastolic blood pressure while watching happy videos (M= 99.400, SD=7.0791) and diastolic blood pressure while watching neutral videos (M=73.933, SD=7.3918) conditions; t (14) =12.222, p= .000”.There was a significant decrease in diastolic blood pressure when participants watched neutral videos after watching happy videos.

Hence enough evidence has been found that shows the mean difference between the diastolic blood pressure of participants is statistically significant when their emotional state is changing from Happy to Neutral. Hence hypothesis accepted which says that there is a significant decrease in the diastolic blood pressure of participants when their emotional state is changing from Happy to Neutral

Table 8.13 Paired Samples Statistics for Diastolic BP (Happy to Neutral State)

	Mean	N	Std. Deviation	Std. Error Mean
Diastolic_BP_Happy	99.400	15	7.0791	1.8278
Diastolic_BP_Neutral	73.933	15	7.3918	1.9085

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Diastolic_BP_Happy - Diastolic_BP_Neutral	25.466	8.069	2.0836	20.9977	29.9356

t	Df	Sig. (2-tailed)
12.22	14	.000

H_1 (Alternate Hypothesis)= There is a significant decrease in the Heart rate of participants when their emotional state is changing from Happy to Neutral

$$H_1 : \mu_1 - \mu_2 < \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypotheses means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.14, to compare the Heart rate of participants while watching different videos using the experimental setup for 20 participants for Happy and Neutral videos. “There was a significant difference in Heart rate while watching happy videos (M= 85.733, SD=8.9400) and Heart rate while watching neutral videos (M=73.267, SD=5.4178) conditions; t (14) = 5.983, p= .000”.There was a significant decrease in diastolic blood pressure when participants watched neutral videos after watching happy videos. Hence enough evidence has been found that shows the mean difference between the Heart rate of participants is statistically significant when their emotional state is changing from Happy to Neutral. Hence hypothesis accepted which says that there is a significant decrease in the Heart rate of participants when their emotional state is changing from Happy to Neutral

Table 8.14 Paired sample statistics for Heart Rate (Happy to Neutral State)

	Mean	N	Std. Deviation	Std. Error Mean
Heart_Rate_Happy	85.733	15	8.9400	2.3083
Heart_Rate_Neutral	73.267	15	5.4178	1.3989

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Heart_Rate_Happy -					

Heart_Rate_Neutral	12.4667	8.0699	2.0836	7.9977	16.9356

t	Df	Sig. (2-tailed)
5.983	14	.000

Paired sample t-test analysis on Neutral and Angry state.

H_1 (Alternate Hypothesis)= There is a significant increase in the Systolic blood pressure of participants when their emotional state is changing from Neutral to Angry

$$H_1 : \mu_1 - \mu_2 > \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypotheses means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.15, to compare the Systolic blood pressure of participants while watching different videos using the experimental setup for 20 participants for Neutral and Angry videos. "There was a significant difference in Systolic blood pressure while watching Neutral videos (M= 114.400, SD=7.3853) and Systolic blood pressure while watching Angry videos (M=136.533, SD=4.4379) conditions; t(14) =-8.137, p= .000".There was a significant increase in the Systolic blood pressure when participants watched neutral videos after watching happy videos.

Hence enough evidence has been found that shows the mean difference between the Systolic blood pressure of participants is statistically significant when their emotional state is changing from Neutral to Angry. Hence hypothesis accepted, which says that there is a significant increase in the Systolic blood pressure of participants when their emotional state is changing from Neutral to Angry.

Table 8.15 Paired sample statistics for Systolic BP (Neutral to Angry State)

	Mean	N	Std. Deviation	Std. Error Mean
Systolic_BP_Neutral	114.400	15	7.3853	1.9069
Systolic_BP_Angry	136.533	15	4.4379	1.1459

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Systolic_BP_Neutral - Systolic_BP_Angry	-2.1333	10.5347	2.7201	-27.9673	-16.2994

t	Df	Sig. (2-tailed)
-8.137	14	.000

H_1 (Hypothesis)= There is a significant increase in the Diastolic BP of participants when their emotional state is changing from Neutral to Angry

$$H_1 : \mu_1 - \mu_2 > \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypothesis means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.16, to compare the Diastolic blood pressure of participants while watching different videos using the experimental setup for 20 participants for Neutral and Angry videos. “There was a significant difference in Diastolic blood pressure while watching Neutral videos (M=73.933, SD=7.3918) and diastolic blood pressure while watching Angry videos (M=110.200, SD=1.4736) conditions; t(14) =-17.413, p= .000”. There was a significant increase in BP when participants watched Angry videos after watching Neutral videos. Hence enough evidence has been found that shows the mean difference between the Diastolic BP of participants is statistically significant when their emotional state is changing from Neutral to Angry.

Hence hypothesis accepted, which says that there is a significant increase in the Diastolic BP of participants when their emotional state is changing from Neutral to Angry.

Table 8.16 Paired sample statistics for Diastolic BP (Neutral to Angry State)

	Mean	N	Std. Deviation	Std. Error Mean
Diastolic_BP_Neutral	73.933	15	7.3918	1.9085
Diastolic_BP_Angry	110.200	15	1.4736	.3805

`

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Diastolic_BP_Neutral - Diastolic_BP_Angry	-36.2667	8.0664	2.0827	-40.7337	-31.7996

t	Df	Sig. (2-tailed)
-17.413	14	.000

H_1 (Hypothesis)= There is a significant increase in the Heart rate of participants when their emotional state is changing from Neutral to Angry

$$H_1 : \mu_1 - \mu_2 > \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypothesis means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.17, to compare the Heart Rate of participants while watching different videos using the experimental setup for 20 participants for Neutral and Angry videos. "There was a significant difference in Heart rate while watching Neutral videos (M=73.267, SD=5.4178) and Heart rate while watching Angry videos (M=98.067, SD=9.6471) conditions; t(14) = -7.170, p= .000".There was a significant increase in the Heart rate when participants watched Angry videos after watching Neutral videos. Hence enough evidence has been found that shows the mean difference between the Heart rate of participants is statistically significant when their emotional state is changing from Neutral to Angry.

Hence hypothesis accepted, which says that there is a significant increase in the Heart rate of participants when their emotional state is changing from Neutral to Angry.

Table 8.17 Paired sample statistics for Heart Rate (Neutral to Angry State)

	Mean	N	Std. Deviation	Std. Error Mean
Heart_Rate_Neutral	73.267	15	5.4178	1.3989
Heart_Rate_Angry	98.067	15	9.6471	2.4909

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Heart_Rate_Neutral - Heart_Rate_Angry	-24.8000	13.3962	3.4589	-32.2185	-17.3815

t	Df	Sig. (2-tailed)
-7.170	14	.000

Paired sample t-test analysis on the angry and sad state

H_1 (Alternate Hypothesis)= There is a significant decrease in the Systolic blood pressure of participants when their emotional state is changing from Angry to Sad.

$$H_1 : \mu_1 - \mu_2 < \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypotheses means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.18, to compare the Systolic blood pressure of participants while watching different videos using the experimental setup for 20 participants for Angry and Sad videos. “There was a significant difference in Systolic blood pressure while watching Angry videos (M=136.533, SD=4.4379) and Systolic blood pressure while watching Sad videos (M=90.200, SD=2.7568) conditions; t (14) =31.535, p= .000”.There was a significant decrease in diastolic blood pressure when participants watched neutral videos after watching happy videos.

Hence enough evidence has been found that shows the mean difference between the diastolic blood pressure of participants is statistically significant when their emotional state is changing from Happy to Neutral. Hence hypothesis accepted, which says that there is a significant decrease in the Systolic blood pressure of participants when their emotional state is changing from Angry to Sad.

Table 8.18 Paired sample statistics for Systolic BP (Angry to Sad State)

	Mean	N	Std. Deviation	Std. Error Mean
Systolic_BP_Angry	136.533	15	4.4379	1.1459
Systolic_BP_Sad	90.200	15	2.7568	.7118

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Systolic_BP_Angry - Systolic_BP_Sad	46.3333	5.6904	1.4693	43.1821	49.4846

t	Df	Sig. (2-tailed)
31.535	14	.000

H_1 (Alternate Hypothesis)= There is a significant decrease in the Diastolic Blood pressure of participants when their emotional state is changing from Angry to Sad

$$H_1 : \mu_1 - \mu_2 < \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypotheses means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.19, to compare the diastolic blood pressure of participants while watching different videos using the experimental setup for 20 participants for Angry and Sad videos. "There was a significant difference in diastolic blood pressure while watching Angry videos (M=110.200, SD=1.4736) and in diastolic blood pressure while watching Sad videos (M=76.53, SD=3.563) conditions; t(14) =33.722, p= .000".There was a significant decrease in diastolic blood pressure when participants watched neutral videos after watching happy videos.

Hence enough evidence has been found that shows the mean difference between the diastolic blood pressure of participants is statistically significant when their emotional state is changing from Angry to Sad. Hence hypothesis accepted which says that there is a significant decrease in the diastolic blood pressure of participants when their emotional state is changing from Angry to Sad

Table 8.19 Paired sample statistics for Diastolic BP (Angry to Sad State)

	Mean	N	Std. Deviation	Std. Error Mean
Diastolic_BP_Angry	110.200	15	1.4736	.3805
Diastolic_BP_Sad	76.53	15	3.563	.920

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Diastolic_BP_Angry - Diastolic_BP_Sad	33.6667	3.8668	.9984	31.5253	35.8080

t	Df	Sig. (2-tailed)
33.720	14	.000

H_0 (Null Hypothesis)= There is no significant difference in the Heart Rate of the participants when their emotional state is changing from Angry to Sad.

H_1 (Alternate Hypothesis)=There is a significant difference in the Heart Rate of the participants when their emotional state is changing from Angry to Sad.

$$H_1 : \mu_1 - \mu_2 < \partial_0$$

Where $\mu_1 - \mu_2$the difference between the hypotheses means and ∂_0 is the hypothesized difference

A paired-sample t-test conducted, as shown in Table 8.20, to compare the Heart Rate of participants while watching different videos using the experimental setup for 20 participants for Angry to Sad videos. "There was no significant difference in the Heart Rate while watching Angry videos (M= 98.067, SD=9.6471) and while watching Sad videos (M=96.067, SD=5.8367) conditions; t (14) =.587 p= .566".

Hence enough evidence has not been found that shows the mean difference between Heart Rate of the participants is statistically significant when their emotional state is changing from Angry to Sad. Hence the null hypothesis is accepted, which says that there is no significant difference in the Heart Rate of the participants when their emotional state is changing from Angry to Sad.

Table 8.20 Paired sample statistics for Heart Rate (Angry to Sad State)

	Mean	N	Std. Deviation	Std. Error Mean
Heart_Rate_Angry	98.067	15	9.6471	2.4909
Heart_Rate_Sad	96.067	15	5.8367	1.5070

	Paired Differences				
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference	
				Lower	Upper
Heart_Rate_Angry - Heart_Rate_Sad	2.0000	13.1909	3.4059	-5.3049	9.3049

t	Df	Sig. (2-tailed)
.587	14	.566

Table 8.21 Confusion Matrix of the developed system for FER

Expected Emotion	Observed Emotion				Correct %
	Happy	Neutral	Angry	Sad	
Happy	14	1	0	0	93.0%
Neutral	0	13	0	2	86.6%
Angry	0	0	13	2	86.6%
Sad	0	0	2	12	80.0%

Table 8.21 shows the confusion matrix for the 3 samples, which shows the accuracy of 86.55% in real-time. For the calculation of accuracy, the formula mention in

equation 8.1 used. The developed system shows promising results with the proposed methodology. The accuracy calculation carried using real-time captured expressions. During the validation of the developed system, the expressions of the participants captured. Then finally used for calculation of the accuracy in real-time using our developed system. Moreover, for validation of proposed preprocessing and feature extraction techniques, a detailed comparison is also made as shown in Table 8.22

Table 8.22 Comparison of the proposed model with existing ones

Author, Year	**Dataset Used**	**Technique Used**	**Count of Facial points in pre-processing**	**Accuracy**	**Total Time Taken and** **CPU Details**
Zhao, Yan, and Yu,2010]	JAFFE	AAM, Canny Filter, and Least Square Method	24	85%	--
[Abdat,Maaoui and Pruski,2011]	FEEDTUM, Cohn-Kanade	RBF,Shi and Thomasi method,SVM	38	95%	721 ms, Intel Pentium 3.4 GHz Processor
[Verma and Dabbagh,2013]	JAFFE	LBP, Ada Boost and SVM	Texture based	86.67%	227ms for SVM,1052 ms for Adaboost,Intel i3 2.2 Ghz Processor
[Suk & Prabhakaran,2014]	Cohn-Kanade	SVM and ASM	77	72%	421.6 ms
[Mistry and Zhang,2014]	Cohn-Kanade	LBP and AAM with Neural network classifier	68	88%	--
Our Method	OpenFace and FER 2013	HOG, Ensemble, SVM and Deep Convolutional Neural Network	68	91.6% for face detection and 86.55% for Emotion Detection in real-time	HOG -7ms Ensemble of Regressor-1ms OpenFace with SVM-8 ms Mini Xception Raspberry-Pi 3B+ Model ARM Cortex-A53 1.4GHz and Intel Movidius Myriad X Vision Processing Unit 4GB,700 MHz

As shown in Table 8.22, our proposed system is capable of detecting faces and facial emotions in real-time using the Raspberry-pi 3B+ model. Moreover, the system is comparatively high-speed and accurate, as well. Our system has achieved accuracy with speed in real-time—even a resource-constrained device for achieving this much speed and accuracy still with a minimal power requirement. Intel NCS needs only 1 Watt of power, and Raspberry pi 3 B+ needs 1.7 to 2 Watt. So the total power requirement is 3 Watts merely. This system is not only energy-efficient as well as performing 100 GFLOPS with this power consumption.

This chapter validates all the objectives in real-time. As per experiments, an efficiency of `91.6% for detecting faces in real-time and 88.6% for detecting emotions in real-time are illustrated in this chapter. The system performed fast when compared with already existing systems. This chapter concluded efficient results on a low power device with minimum power consumption and high computational speed.

CHAPTER 9

CONCLUSION AND FUTURE DIRECTIONS

This chapter is all about the significant contributions, conclusion, and future directions of the current research work in hand. The current research work focused on four objectives. All the four objectives are discussed in detail in Chapter 1 under thesis contribution and achieved successfully. The research publications achieved for the validity and novelty of the work also mentioned in chapter 1.

9.1 Major Outcomes

The main aim of this work is to develop a system that is capable of understanding human emotions at any point in time, irrespective of age, gender, and race. Moreover, successful efforts have been made to make the system compact and cost-effective over existing oppressive, costly, hefty, and complex facial emotion detection systems. This system comprises of Heart rate and BP sensor, camera module, latest processor, co-processor, controller, and customized boards.

- This system is IOT enabled and having the capability to work as an independent portable Node.
- This system is not only limited to camera or IoT. It is a combination of all those essential things that are required to collect the data and give a real-time result of differential expression of human beings.
- The sample size is also significant, and the age structure is also not limited to one age group.
- This is a cost-effective solution in comparison with existing costly systems.
- This system is portable and compact.
- This system is capable of running pre-trained deep learning models; hence can be trained with a vast dataset and can provide greater accuracy.

- This system is not limited to only one kind of application, person, or age group. However, this is a solution to various problems where facial emotion and identification of person plays an important role.
- An efficiency of 73% with FER 2013 dataset after optimization and deployment of deep networks on Raspberry-Pi achieved. This system performed well in comparison to 66%, as mentioned in state of the art.

9.2 Benefits of current work

The research is going to benefit humanity in various areas:

- Health----This device can understand human emotions via facial sensors also. This device can detect conditions like stress, depression, anxiety, sadness, excitement, and many more expression.

- Security---This device can detect the human face and be very compact and handy; this device is going to be a great help where authorized entry is allowed. So, this device can replace the complex, heavy, and bulky face detection system.

- Cost-effective-----This device is a cost-effective solution over the existing problems, where camera deployment, connecting it to the cloud, and then working with the software to detect faces and their emotions is a very costly system. So, this compact system is the solution to all existing problems, and it is a shallow power-consuming device.

9.3 Conclusion and Discussion

The primary goal of this work is to develop a system that can replace the existing bulky, wired, and system dependent system that almost makes the work of face and facial emotion detection impossible while walking on roads, airports, hospitals, public places. One has to spend a lot to get the benefit of such a system. Moreover, it has also observed from the literature review that, however, studies have carried out various techniques that are required to achieve facial emotion recognition. However, no literature has found in the direction of designing and implementation of devices (portable, cheap, and efficient) in real-time.

This thesis suggests the development of an intelligent device that is capable of detecting human faces and their emotions in real-time. This thesis presents a smart system and an IOT based vision Mote device, which is designed for detecting the real-time behavior of a person. It is a small contribution to the social cause as the device which is designed for detecting real-time behavior of people under different situations. This device automatically detects human presence and capture the human face along with its facial emotions. Hence, it can collect real-time data and upload the captured emotions on the cloud that can be accessed remotely.

An optimal solution for increasing the speed of the deep network achieved, with the help of a co-processor called Intel Movidius Neural Compute Stick-II. The system is easy to handle and can be used anywhere at any point in time. An efficiency of 73% achieved that is greater than the state of the art.

9.4 Novelty

- A real-time, standalone facial emotion recognition device using Raspberry-Pi and edge device proposed.
- To make the system handy and cost useful customization of the boards has been done.

9.5 Future Directions

In the future, the system can be implemented with the help of more powerful embedded boards that are available in the market like NVIDIA's Jetson Nano and Google Coral's Dev Board. These boards may increase the cost a little bit can make the existing system more efficient and capable of handling more complex Deep Neural Networks. Moreover, to make system maintenance, free solar batteries are also suggested. In this work, only those deep networks are optimized, that can be easily deployable on Raspberry-Pi being a resource-constrained device, and the efficiency of 86.55% to recognize facial emotions, achieved in real-time, but in the future, with the help of embedded boards various deep learning models can be used with better efficiency.

This chapter concluded all the essential contributions to achieve the research and outcomes of the research. This chapter gave the future direction to the research community for the continuation of research work in the same field.

References

Abdat, F., Maaoui, C., & Pruski, A. (2011, November). Human-computer interaction using emotion recognition from facial expression. In *2011 UKSim 5th European Symposium on Computer Modeling and Simulation* (pp. 196-201). IEEE.

Abbasnejad, I., Sridharan, S., Nguyen, D., Denman, S., Fookes, C., & Lucey, S. (2017). *Using synthetic data to improve facial expression analysis with 3d convolutional networks.* Paper presented at the Proceedings of the IEEE International Conference on Computer Vision Workshops.

Adeshina, A. M., Lau, S.-H., & Loo, C.-K. (2009). *Real-time facial expression recognitions: A review.* Paper presented at the 2009 Innovative Technologies in Intelligent Systems and Industrial Applications.

Ahsan, T., Jabid, T., & Chong, U.-P. (2013). Facial expression recognition using local transitional pattern on Gabor filtered facial images. *IETE Technical Review, 30*(1), 47-52.

Alazrai, R., & Lee, C. G. (2012). *Real-time emotion identification for socially intelligent robots.* Paper presented at the 2012 IEEE International Conference on Robotics and Automation.

Aleksic, P. S., & Katsaggelos, A. K. (2006). Automatic facial expression recognition using facial animation parameters and multistream HMMs. *IEEE transactions on information forensics and security, 1*(1), 3-11.

Amos, B., Ludwiczuk, B., & Satyanarayanan, M. (2016). Openface: A general-purpose face recognition library with mobile applications. *CMU School of Computer Science, 6*, 2.

Antonakos, E., Alabort-i-Medina, J., Tzimiropoulos, G., & Zafeiriou, S. (2014). *Hog active appearance models.* Paper presented at the 2014 IEEE International Conference on Image Processing (ICIP).

Asthana, A., Zafeiriou, S., Cheng, S., & Pantic, M. (2013). *Robust discriminative response map fitting with constrained local models.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Asthana, A., Zafeiriou, S., Cheng, S., & Pantic, M. (2014). *Incremental face alignment in the wild.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Bailenson, J. N., Pontikakis, E. D., Mauss, I. B., Gross, J. J., Jabon, M. E., Hutcherson, C. A., . . . John, O. (2008). Real-time classification of evoked emotions using facial feature tracking and physiological responses. *International journal of human-computer studies, 66*(5), 303-317.

Belhumeur, P. N., Hespanha, J. P., & Kriegman, D. J. (1997). Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on pattern analysis and machine intelligence, 19*(7), 711-720.

Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2007). *Greedy layer-wise training of deep networks.* Paper presented at the Advances in neural information processing systems.

Bengio, Y., & LeCun, Y. (2007). Scaling learning algorithms towards AI. *Large-scale kernel machines, 34*(5), 1-41.

Berretti, S., Del Bimbo, A., Pala, P., Amor, B. B., & Daoudi, M. (2010). *A set of selected SIFT features for 3D facial expression recognition.* Paper presented at the 2010 20th International Conference on Pattern Recognition.

Brahnam, S., Chuang, C.-F., Sexton, R. S., & Shih, F. Y. (2007). Machine assessment of neonatal facial expressions of acute pain. *Decision Support Systems, 43*(4), 1242-1254.

Breuer, R., & Kimmel, R. (2017). An in-depth learning perspective on the origin of facial expressions. *arXiv preprint arXiv:1705.01842*.

Căleanu, C.-D. (2013). *Face expression recognition: A brief overview of the last decade.* Paper presented at the 2013 IEEE 8th International Symposium on Applied Computational Intelligence and Informatics (SACI).

Cament, L. A., Galdames, F. J., Bowyer, K. W., & Perez, C. A. (2015). Face recognition under pose variation with local Gabor features enhanced by active shape and statistical models. *Pattern Recognition, 48*(11), 3371-3384.

Carreira-Perpignan, M. (2005). *HGE On contrastive divergence learning.* Paper presented at the Proceedings of the International Conference on Artificial Intelligence and Statistics.

Chang, Y., Hu, C., Feris, R., & Turk, M. (2006). Manifold based analysis of facial expression. *Image and Vision Computing, 24*(6), 605-614.

Chen, S., Pande, A., & Mohapatra, P. (2014). *Sensor-assisted facial recognition: an enhanced biometric authentication system for smartphones.* Paper presented at the Proceedings of the 12th annual international conference on Mobile systems, applications, and services.

Chen, W., Er, M. J., & Wu, S. (2006). Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 36*(2), 458-466.

Chen, Y., Hua, C., & Bai, R. (2014). Regression-based active appearance model initialization for facial feature tracking with missing frames. *Pattern Recognition Letters, 38*, 113-119.

Cheon, Y., & Kim, D. (2009). Natural facial expression recognition using differential-AAM and manifold learning. *Pattern Recognition, 42*(7), 1340-1350.

Chu, W.-S., De la Torre, F., & Cohn, J. F. (2017). *Learning spatial and temporal cues for multi-label facial action unit detection.* Paper presented at the 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017).

Cohn, J. F., Kanade, T., & Li, C.-C. (1998). *Subtly different facial expression recognition and expression intensity estimation.* Paper presented at the Proc. IEEE Conf. Computer Vision and Pattern Recognition.

Cootes, T., Edwards, G., & Taylor, C. (1998). Active appearance models. IEEE Transactions on Pattern Analysis and Machine Intelligence. *IEEE Transactions on pattern analysis and machine intelligence, 23*(6), 681685.

Cootes, T. F., Edwards, G. J., & Taylor, C. J. (2001). Active appearance models. *IEEE Transactions on pattern analysis and machine intelligence, 23*(6), 681-685.

Cootes, T. F., Taylor, C. J., Cooper, D. H., & Graham, J. (1995). Active shape models-their training and application. *Computer Vision and Image Understanding, 61*(1), 38-59.

Dalal, N., & Triggs, B. (2005). *Histograms of oriented gradients for human detection.* Paper presented at the 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05).

Dapogny, A., & Bailly, K. (2018). *Investigating deep neural forests for facial expression recognition.* Paper presented at the 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018).

Darwin, C. (1872). *The expression of the emotions in man and animals by Charles Darwin*: Murray.

De Silva, C. R., Ranganath, S., & De Silva, L. C. (2008). Cloud basis function neural network: a modified RBF network architecture for holistic facial expression recognition. *Pattern Recognition, 41*(4), 1241-1253.

Deng, L., & Yu, D. (2014). Deep learning: methods and applications. *Foundations and trends in signal processing, 7*(3–4), 197-387.

Deshmukh, S., Patwardhan, M., & Mahajan, A. (2016). Survey on real-time facial expression recognition techniques. *Iet Biometrics, 5*(3), 155-163.

Devries, T., Biswaranjan, K., & Taylor, G. W. (2014). *Multi-task learning of facial landmarks and expression.* Paper presented at the 2014 Canadian Conference on Computer and Robot Vision.

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., & Darrell, T. (2015). *Long-term recurrent convolutional networks for visual recognition and description.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Donoho, D. L. (2006). Compressed sensing. *IEEE Transactions on information theory, 52*(4), 1289-1306.

Du, S., Tao, Y., & Martinez, A. M. (2014). Compound facial expressions of emotion. *Proceedings of the National Academy of Sciences, 111*(15), E1454-E1462.

Ebrahimi Kahou, S., Michalski, V., Konda, K., Memisevic, R., & Pal, C. (2015). *Recurrent neural networks for emotion recognition in video.* Paper presented at the Proceedings of the 2015 ACM on International Conference on Multimodal Interaction.

Fabian Benitez-Quiroz, C., Srinivasan, R., & Martinez, A. M. (2016). *Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Fan, Y., Lu, X., Li, D., & Liu, Y. (2016). *Video-based emotion recognition using CNN-RNN and C3D hybrid networks.* Paper presented at the Proceedings of the 18th ACM International Conference on Multimodal Interaction.

Fang, H., Mac Parthaláin, N., Aubrey, A. J., Tam, G. K., Borgo, R., Rosin, P. L., . . . Chen, M. (2014). Facial expression recognition in dynamic sequences: An integrated approach. *Pattern Recognition, 47*(3), 1271-1281.

Fasel, B. (2002a). *Head-pose invariant facial expression recognition using convolutional neural networks.* Paper presented at the Proceedings. Fourth IEEE International Conference on Multimodal Interfaces.

Fasel, B. (2002b). *Robust face analysis using convolutional neural networks.* Paper presented at the Object recognition supported by user interaction for service robots.

Fischer, A., & Igel, C. (2012). *An introduction to restricted Boltzmann machines.* Paper presented at the Iberoamerican congress on pattern recognition.

Geetha, A., Ramalingam, V., Palanivel, S., & Palaniappan, B. (2009). Facial expression recognition–A real time approach. *Expert Systems with Applications, 36*(1), 303-308.

Georghiades, A. S., Belhumeur, P. N., & Kriegman, D. J. (2001). From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on pattern analysis and machine intelligence, 23*(6), 643-660.

Ghandi, B. M., Nagarajan, R., & Desa, H. (2009). *Particle swarm optimization algorithm for facial emotion detection.* Paper presented at the 2009 IEEE Symposium on Industrial Electronics & Applications.

Ghimire, D., & Lee, J. (2013). Geometric feature-based facial expression recognition in image sequences using multi-class adaboost and support vector machines. *Sensors, 13*(6), 7714-7734.

Glorot, X., Bordes, A., & Bengio, Y. (2011). *Deep sparse rectifier neural networks.* Paper presented at the Proceedings of the fourteenth international conference on artificial intelligence and statistics.

Goh, R., Liu, L., Liu, X., & Chen, T. (2005). *The CMU face in action (FIA) database.* Paper presented at the International Workshop on Analysis and Modeling of Faces and Gestures.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., . . . Bengio, Y. (2014). *Generative adversarial nets.* Paper presented at the Advances in neural information processing systems.

Goodfellow, I. J., Erhan, D., Carrier, P. L., Courville, A., Mirza, M., Hamner, B., . . . Lee, D.-H. (2015). Challenges in representation learning: A report on three machine learning contests. *Neural Networks, 64*, 59-63.

Graves, A., Mayer, C., Wimmer, M., Schmidhuber, J., & Radig, B. (2008). *Facial expression recognition with recurrent neural networks.* Paper presented at the Proceedings of the International Workshop on Cognition for Technical Systems.

Gross, R., Matthews, I., Cohn, J., Kanade, T., & Baker, S. (2010). Multi-pie. *Image and Vision Computing, 28*(5), 807-813.

Gu, W., Xiang, C., Venkatesh, Y., Huang, D., & Lin, H. (2012). Facial expression recognition using the radial encoding of local Gabor features and classifier synthesis. *Pattern Recognition, 45*(1), 80-91.

Gunawan, A. A. (2015). Face expression detection on Kinect using active appearance model and fuzzy logic. *Procedia Computer Science, 59*, 268-274.

Hamm, J., Kohler, C. G., Gur, R. C., & Verma, R. (2011). Automated facial action coding system for dynamic analysis of facial expressions in neuropsychiatric disorders. *Journal of neuroscience methods, 200*(2), 237-256.

Happy, S., George, A., & Routray, A. (2012). *A real time facial expression classification system using local binary patterns.* Paper presented at the 2012 4th International conference on intelligent human computer interaction (IHCI).

Hasani, B., & Mahoor, M. H. (2017a). *Facial expression recognition using enhanced deep 3D convolutional neural networks.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.

Hasani, B., & Mahoor, M. H. (2017b). *Spatio-temporal facial expression recognition using convolutional neural networks and conditional random fields.* Paper presented at the 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017).

Hassan, I., Maqbool, O., Ahsan, Q., & Qayyum, U. (2010). *Cascading neural network with AdaBoost for face detection.* Paper presented at the Proc. Int. Conf. Applied Sciences & Technology, Islamabad, Pakistan.

Hassner, T., Harel, S., Paz, E., & Enbar, R. (2015). *Effective face frontalization in unconstrained images.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). *Deep residual learning for image recognition.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural computation, 14*(8), 1771-1800.

Hinton, G. E. (2009). Deep belief networks. *Scholarpedia, 4*(5), 5947.

Hinton, G. E. (2012). A practical guide to training restricted Boltzmann machines. In *Neural networks: Tricks of the trade* (pp. 599-619): Springer.

Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science, 313*(5786), 504-507.

Hinton, G. E., & Sejnowski, T. J. (1986). Learning and relearning in Boltzmann machines. *Parallel distributed processing: Explorations in the microstructure of cognition, 1*(282-317), 2.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation, 9*(8), 1735-1780.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks, 4*(2), 251-257.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*(5), 359-366.

Houmb, S. H., Georg, G., Jurjens, J., & France, R. (2008). An integrated security verification and security solution design trade-off analysis approach. In *Information Security and Ethics: Concepts, Methodologies, Tools, and Applications* (pp. 2234-2258): IGI Global.

Huang, D., Shan, C., Ardabilian, M., Wang, Y., & Chen, L. (2011). Local binary patterns and its application to facial image analysis: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 41*(6), 765-781.

Huang, R., Zhang, S., Li, T., & He, R. (2017). *Beyond face rotation: Global and local perception gan for photorealistic and identity preserving frontal view synthesis.* Paper presented at the Proceedings of the IEEE International Conference on Computer Vision.

Ibraheem, N. A., Hasan, M. M., Khan, R. Z., & Mishra, P. K. (2012). Understanding color models: a review. *ARPN Journal of science and technology, 2*(3), 265-275.

Jain, D. K., Zhang, Z., & Huang, K. (2017). Multi angle optimal pattern-based deep learning for automatic facial expression recognition. *Pattern Recognition Letters*.

Jeong, M., & Ko, B. C. (2018). Driver's facial expression recognition in real-time for safe driving. *Sensors, 18*(12), 4270.

Jeong, M., Ko, B. C., Kwak, S., & Nam, J.-Y. (2017). Driver facial landmark detection in real driving situations. *IEEE Transactions on Circuits and Systems for Video Technology, 28*(10), 2753-2767.

Jung, H., Lee, S., Yim, J., Park, S., & Kim, J. (2015). *Joint fine-tuning in deep neural networks for facial expression recognition.* Paper presented at the Proceedings of the IEEE international conference on computer vision.

Kaburlasos, V. G., Papadakis, S. E., & Papakostas, G. A. (2013). Lattice computing extension of the FAM neural classifier for human facial expression recognition. *IEEE Transactions on Neural Networks and Learning Systems, 24*(10), 1526-1538.

Keauhou, S. E., Pal, C., Bouthillier, X., Froumenty, P., Gülçehre, Ç., Memisevic, R., . . . Ferrari, R. C. (2013). *Combining modality specific deep neural networks for emotion recognition in video.* Paper presented at the Proceedings of the 15th ACM on International conference on multimodal interaction.

Kamachi, M., Lyons, M., & Gyoba, J. (1998). The japanese female facial expression (jaffe) database. *URL http://www. kasrl. org/jaffe. html, 21*, 32.

Kanade, T. (2000). Cohn-Kanade au-coded facial expression database. *Robotics Institute, Carnegie Mellon University*.

Kanade, T., Cohn, J. F., & Tian, Y. (2000). *Comprehensive database for facial expression analysis.* Paper presented at the Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580).

Kaulard, K., Cunningham, D. W., Bülthoff, H. H., & Wallraven, C. (2012). The MPI facial expression database—a validated database of emotional and conversational facial expressions. *PloS one, 7*(3).

Kazemi, V., & Sullivan, J. (2014). *One millisecond face alignment with an ensemble of regression trees.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Khan, R. A., Meyer, A., Konik, H., & Bouakaz, S. (2013). Framework for reliable, real-time facial expression recognition for low resolution images. *Pattern Recognition Letters, 34*(10), 1159-1168.

Kim, B.-K., Lee, H., Roh, J., & Lee, S.-Y. (2015). *Hierarchical committee of deep cnns with exponentially-weighted decision fusion for static facial expression recognition.* Paper presented at the Proceedings of the 2015 ACM on International Conference on Multimodal Interaction.

Kim, D. H., Baddar, W. J., Jang, J., & Ro, Y. M. (2017). Multi-objective based spatio-temporal feature representation learning robust to expression intensity variations for facial expression recognition. *IEEE Transactions on Affective Computing, 10*(2), 223-236.

King, D. E. (2009). Dlib-ml: A machine learning toolkit. *Journal of machine learning research, 10*(Jul), 1755-1758.

Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kuo, C.-M., Lai, S.-H., & Sarkis, M. (2018). *A compact deep learning model for robust facial expression recognition.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.

Le, Q. V. (2013). *Building high-level features using large scale unsupervised learning.* Paper presented at the 2013 IEEE international conference on acoustics, speech and signal processing.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature (2015). *T, 521*, 436.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. *Neural computation, 1*(4), 541-551.

Leshno, M., Lin, V. Y., Pinkus, A., & Schocken, S. (1993). Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks, 6*(6), 861-867.

Levi, G., & Hassner, T. (2015). *Emotion recognition in the wild via convolutional neural networks and mapped binary patterns.* Paper presented at the Proceedings of the 2015 ACM on international conference on multimodal interaction.

Li, J., & Lam, E. Y. (2015). *Facial expression recognition using deep neural networks.* Paper presented at the 2015 IEEE International Conference on Imaging Systems and Techniques (IST).

Li, J., Zhang, D., Zhang, J., Zhang, J., Li, T., Xia, Y., . . . Xun, L. (2017). Facial expression recognition with faster R-CNN. *Procedia Computer Science, 107*, 135-140.

Li, S., Yi, D., Lei, Z., & Liao, S. (2013). *The casia nir-vis 2.0 face database.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition workshops.

Li, W., Li, M., Su, Z., & Zhu, Z. (2015). *A deep-learning approach to facial expression recognition with candid images.* Paper presented at the 2015 14th IAPR International Conference on Machine Vision Applications (MVA).

Li, X., Ruan, Q., Jin, Y., An, G., & Zhao, R. (2015). Fully automatic 3D facial expression recognition using polytypic multi-block local binary patterns. *Signal Processing, 108*, 297-308.

Li, Y., Liu, W., Li, X., Huang, Q., & Li, X. (2014). GA-SIFT: A new scale invariant feature transform for multispectral image using geometric algebra. *Information Sciences, 281*, 559-572.

Liu, S.-s., & Tian, Y.-t. (2010). *Facial expression recognition method based on gabor wavelet features and fractional power polynomial kernel PCA.* Paper presented at the International Symposium on Neural Networks.

Liu, X., Vijaya Kumar, B., You, J., & Jia, P. (2017). *Adaptive deep metric learning for identity-aware facial expression recognition.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.

Lopes, A. T., de Aguiar, E., De Souza, A. F., & 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.

Lowe, D. G. (1999). *Object recognition from local scale-invariant features.* Paper presented at the Proceedings of the seventh IEEE international conference on computer vision.

Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. *International journal of computer vision, 60*(2), 91-110.

Lu, H.-C., Huang, Y.-J., Chen, Y.-W., & Yang, D.-I. (2007). Real-time facial expression recognition based on pixel-pattern-based texture feature. *Electronics letters, 43*(17), 916-918.

Lucey, P. (2010). "The extended Cohn-Kanade dataset (CK+): a complete dataset for action unit and emotion-specified expression," in Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW.

Lundqvist, D., Flykt, A., & Öhman, A. (1998). The Karolinska directed emotional faces (KDEF). *CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet, 91*(630), 2.2.

Lyons, M., Akamatsu, S., Kamachi, M., & Gyoba, J. (1998). *April. Coding facial expressions with Gabor wavelets. Automatic Face and Gesture Recognition, 1998.* Paper presented at the Proceedings of the 3rd IEEE International Conference.

Lyons, M. J., Akamatsu, S., Kamachi, M., Gyoba, J., & Budynek, J. (1998). *The Japanese female facial expression (JAFFE) database.* Paper presented at the Proceedings of third international conference on automatic face and gesture recognition.

Ma, L., & Khorasani, K. (2004). Facial expression recognition using constructive feedforward neural networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 34*(3), 1588-1595.

Mandal, B., Chia, S.-C., Li, L., Chandrasekhar, V., Tan, C., & Lim, J.-H. (2014). *A wearable face recognition system on google glass for assisting social interactions.* Paper presented at the Asian Conference on Computer Vision.

Masci, J., Meier, U., Cireşan, D., & Schmidhuber, J. (2011). *Stacked convolutional auto-encoders for hierarchical feature extraction.* Paper presented at the International conference on artificial neural networks.

Mavadati, S. M., Mahoor, M. H., Bartlett, K., Trinh, P., & Cohn, J. F. (2013). Disfa: A spontaneous facial action intensity database. *IEEE Transactions on Affective Computing, 4*(2), 151-160.

Mayer, C., Wimmer, M., Stulp, F., Riaz, Z., Roth, A., Eggers, M., & Radig, B. (2008). *A real time system for model-based interpretation of the dynamics of facial expressions.* Paper presented at the 2008 8th IEEE International Conference on Automatic Face & Gesture Recognition.

Meng, Z., Liu, P., Cai, J., Han, S., & Tong, Y. (2017). *Identity-aware convolutional neural network for facial expression recognition.* Paper presented at the 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017).

Minsky, M., & Papert, S. (1969). An introduction to computational geometry. *Cambridge tiass., HIT.*

Mistry, K., Zhang, L., Neoh, S. C., Jiang, M., Hossain, A., & Lafon, B. (2014, December). Intelligent Appearance and shape based facial emotion recognition for a humanoid robot. In *The 8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA 2014)* (pp. 1-8). IEEE.

Mishra, A., Rai, S. N., Mishra, A., & Jawahar, C. (2016). *IIIT-CFW: a benchmark database of cartoon faces in the wild.* Paper presented at the European Conference on Computer Vision.

Mohammadi, M. R., Fatemizadeh, E., & Mahoor, M. H. (2014). PCA-based dictionary building for accurate facial expression recognition via sparse representation. *Journal of Visual Communication and Image Representation, 25*(5), 1082-1092.

Negahdaripour, S. (1998). Revised definition of optical flow: Integration of radiometric and geometric cues for dynamic scene analysis. *IEEE Transactions on pattern analysis and machine intelligence, 20*(9), 961-979.

Ng, H.-W., Nguyen, V. D., Vonikakis, V., & Winkler, S. (2015). *Deep learning for emotion recognition on small datasets using transfer learning.* Paper presented at the Proceedings of the 2015 ACM on international conference on multimodal interaction.

Nguyen, A., Yosinski, J., & Clune, J. (2015). *Deep neural networks are easily fooled: High confidence predictions for unrecognizable images.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Ojala, T., Pietikäinen, M., & Harwood, D. (1996). A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition, 29*(1), 51-59.

Olah, C. (2017). Understanding LSTM Networks. Aug. 2015. *URL https://colah. github. io/posts/2015-08-Understanding-LSTMs.*

Ouyang, Y., Sang, N., & Huang, R. (2015). Accurate and robust facial expressions recognition by fusing multiple sparse representation based classifiers. *Neurocomputing, 149*, 71-78.

Owusu, E., Zhan, Y., & Mao, Q. R. (2014). A neural-AdaBoost based facial expression recognition system. *Expert Systems with Applications, 41*(7), 3383-3390.

Pantic, M., & Patras, I. (2006). Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 36*(2), 433-449.

Peng, Z.-y., Wen, Z.-q., & Zhou, Y. (2009). *Application of mean shift algorithm in real-time facial expression recognition.* Paper presented at the 2009 International Symposium on Computer Network and Multimedia Technology.

Pitaloka, D. A., Wulandari, A., Basaruddin, T., & Liliana, D. Y. (2017). Enhancing CNN with preprocessing stage in automatic emotion recognition. *Procedia Computer Science, 116*, 523-529.

Punitha, A., & Geetha, M. K. (2013). HMM based real time facial expression recognition. *International Journal of Emerging Technology and Advanced Engineering, 3*(1), 180-185.

Rastogi, R., Jain, R., Jain, P., Singhal, P., Garg, P., & Rastogi, M. (2020). Inference-Based Statistical Analysis for Suspicious Activity Detection Using Facial Analysis. In Computational Intelligence in Pattern Recognition (pp. 29-51). Springer, Singapore.

Ren, S., Cao, X., Wei, Y., & Sun, J. (2014). *Face alignment at 3000 fps via regressing local binary features.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Rifai, S., Vincent, P., Muller, X., Glorot, X., & Bengio, Y. (2011). Contractive auto-encoders: Explicit invariance during feature extraction.

Rosenblatt, F. (1957). The Perceptron-a Perceiving and Recognizing Automaton (85-460-1). *Ithica: Cornell Aeronautical Laboratory*.

Samnani, P., & Jain, R. (2017). *Facial Expression Recognition & Face Detection using D-CNN–A Deep Vision.* Lovely Professional University,

Sánchez, A., Ruiz, J. V., Moreno, A. B., Montemayor, A. S., Hernández, J., & Pantrigo, J. J. (2011). Differential optical flow applied to automatic facial expression recognition. *Neurocomputing, 74*(8), 1272-1282.

Sandbach, G., Zafeiriou, S., Pantic, M., & Yin, L. (2012). Static and dynamic 3D facial expression recognition: A comprehensive survey. *Image and Vision Computing, 30*(10), 683-697.

Sebe, N., Lew, M. S., Sun, Y., Cohen, I., Gevers, T., & Huang, T. S. (2007). Authentic facial expression analysis. *Image and Vision Computing, 25*(12), 1856-1863.

Shan, C., Gong, S., & McOwan, P. W. (2009). Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing, 27*(6), 803-816.

Shbib, R., & Zhou, S. (2015). Facial expression analysis using active shape model. *International Journal of Signal Processing, Image Processing and Pattern Recognition, 8*(1), 9-22.

Shin, M., Kim, M., & Kwon, D.-S. (2016). *Baseline CNN structure analysis for facial expression recognition.* Paper presented at the 2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN).

Sim, T., Baker, S., & Bsat, M. (2001). *The CMU pose, illumination, and expression (PIE) database of human faces*: Citeseer.

Sown, M. (1978). *A preliminary note on pattern recognition of facial emotional expression.* Paper presented at the The 4th International Joint Conferences on Pattern Recognition, 1978.

Soyel, H., & Demirel, H. (2010). Facial expression recognition based on discriminative scale invariant feature transform. *Electronics letters, 46*(5), 343-345.

Suk, M., & Prabhakaran, B. (2014). *Real-time mobile facial expression recognition system-a case study.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.

Sun, B., Li, L., Zhou, G., & He, J. (2016). Facial expression recognition in the wild based on multimodal texture features. *Journal of Electronic Imaging, 25*(6), 061407.

Sun, B., Li, L., Zhou, G., Wu, X., He, J., Yu, L., . . . Wei, Q. (2015). *Combining multimodal features within a fusion network for emotion recognition in the wild.* Paper presented at the Proceedings of the 2015 ACM on International Conference on Multimodal Interaction.

Sun, Y., & Akansu, A. (2014). Facial expression recognition with regional hidden Markov models. *Electronics letters, 50*(9), 671-673.

Sun, Y., Wang, X., & Tang, X. (2013). *Deep convolutional network cascade for facial point detection.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Sung, J., Lee, S., & Kim, D. (2006). *A real-time facial expression recognition using the STAAM.* Paper presented at the 18th International Conference on Pattern Recognition (ICPR'06).

Szwoch, M., & Pieniążek, P. (2015). *Facial emotion recognition using depth data.* Paper presented at the 2015 8th International Conference on Human System Interaction (HSI).

Taigman, Y., Yang, M., Ranzato, M. A., & Wolf, L. (2014). *Deepface: Closing the gap to human-level performance in face verification.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Tang, Y. (2013). Deep learning using linear support vector machines. *arXiv preprint arXiv:1306.0239*.

Tavares, G., Mourão, A., & Magalhães, J. (2016). Crowdsourcing facial expressions for affective-interaction. *Computer Vision and Image Understanding, 147*, 102-113.

Tian, Y.-l., Kanade, T., & Cohn, J. F. (2002). *Evaluation of Gabor-wavelet-based facial action unit recognition in image sequences of increasing complexity.* Paper presented at the Proceedings of Fifth IEEE International Conference on Automatic Face Gesture Recognition.

Tian, Y.-L., Kanade, T., & Cohn, J. F. (2005). Facial expression analysis. In *Handbook of face recognition* (pp. 247-275): Springer.

Tian, Y., Kanade, T., & Cohn, J. F. (2011). Facial expression recognition. In *Handbook of face recognition* (pp. 487-519): Springer.

Tie, Y., & Guan, L. (2012). A deformable 3-D facial expression model for dynamic human emotional state recognition. *IEEE Transactions on Circuits and Systems for Video Technology, 23*(1), 142-157.

Tran, L., Yin, X., & Liu, X. (2017). *Disentangled representation learning gan for pose-invariant face recognition.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Varun, R., Kini, Y. V., Manikantan, K., & Ramachandran, S. (2015). Face recognition using hough transform based feature extraction. *Procedia Computer Science, 46*, 1491-1500.

Verma, R. C., Schmid, C., & Mikolajczyk, K. (2003). Face detection and tracking in a video by propagating detection probabilities. *IEEE Transactions on pattern analysis and machine intelligence, 25*(10), 1215-1228.

Verma, R., & Dabbagh, M. Y. (2013, May). Fast facial expression recognition based on local binary patterns. In *2013 26th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)* (pp. 1-4). IEEE.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., & Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of machine learning research, 11*(Dec), 3371-3408.

Viola, P., & Jones, M. (2001). *Rapid object detection using a boosted cascade of simple features.* Paper presented at the Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001.

Walecki, R., Rudovic, O., Pavlovic, V., Schuller, B., & Pantic, M. (2017). Deep structured learning for facial expression intensity estimation. *Image Vis. Comput, 259*, 143-154.

Wang, X., Zhao, X., Prakash, V., Shi, W., & Gnawali, O. (2013). *Computerized-eyewear based face recognition system for improving social lives of prosopagnosics.* Paper presented at the 2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops.

Werbos, P. J. (1990). Backpropagation through time: what it does and how to do it. *Proceedings of the IEEE, 78*(10), 1550-1560.

Werbos, P. J. (1994). *The roots of backpropagation: from ordered derivatives to neural networks and political forecasting* (Vol. 1): John Wiley & Sons.

Wiener, M., & Mehrabian, A. (1968). *Language within language: Immediacy, a channel in verbal communication*: Ardent Media.

Wilson, P. I., & Fernandez, J. (2006). Facial feature detection using Haar classifiers. *Journal of Computing Sciences in Colleges, 21*(4), 127-133.

Wright, J., Yang, A. Y., Ganesh, A., Sastry, S. S., & Ma, Y. (2008). Robust face recognition via sparse representation. *IEEE Transactions on pattern analysis and machine intelligence, 31*(2), 210-227.

Xiong, X., & De la Torre, F. (2013). *Supervised descent method and its applications to face alignment.* Paper presented at the Proceedings of the IEEE conference on computer vision and pattern recognition.

Yacoob, Y., & Davis, L. S. (1996). Recognizing human facial expressions from long image sequences using optical flow. *IEEE Transactions on pattern analysis and machine intelligence, 18*(6), 636-642.

Yan, W.-J., Li, X., Wang, S.-J., Zhao, G., Liu, Y.-J., Chen, Y.-H., & Fu, X. (2014). CASME II: An improved spontaneous micro-expression database and the baseline evaluation. *PloS one, 9*(1).

Yang, J., Zhang, D., Frangi, A. F., & Yang, J.-y. (2004). Two-dimensional PCA: a new approach to appearance-based face representation and recognition. *IEEE Transactions on pattern analysis and machine intelligence, 26*(1), 131-137.

Yao, A., Cai, D., Hu, P., Wang, S., Sha, L., & Chen, Y. (2016). *HoloNet: towards robust emotion recognition in the wild.* Paper presented at the Proceedings of the 18th ACM International Conference on Multimodal Interaction.

Yin, L., Wei, X., Sun, Y., Wang, J., & Rosato, M. J. (2006). *A 3D facial expression database for facial behavior research.* Paper presented at the 7th international conference on automatic face and gesture recognition (FGR06).

Yin, X., Yu, X., Sohn, K., Liu, X., & Chandraker, M. (2017). *Towards large-pose face frontalization in the wild.* Paper presented at the Proceedings of the IEEE International Conference on Computer Vision.

Yu, Z., Liu, G., Liu, Q., & Deng, J. (2018). Spatio-temporal convolutional features with nested LSTM for facial expression recognition. *Neurocomputing, 317*, 50-57.

Yu, Z., Liu, Q., & Liu, G. (2018). Deeper cascaded peak-piloted network for weak expression recognition. *The Visual Computer, 34*(12), 1691-1699.

Yu, Z., & Zhang, C. (2015). *Image based static facial expression recognition with multiple deep network learning.* Paper presented at the Proceedings of the 2015 ACM on international conference on multimodal interaction.

Yurtkan, K., & Demirel, H. (2014). Feature selection for improved 3D facial expression recognition. *Pattern Recognition Letters, 38*, 26-33.

Zavarez, M. V., Berriel, R. F., & Oliveira-Santos, T. (2017). *Cross-database facial expression recognition based on fine-tuned deep convolutional network.* Paper presented at the 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI).

Zeng, N., Zhang, H., Song, B., Liu, W., Li, Y., & Dobaie, A. M. (2018). Facial expression recognition via learning deep sparse autoencoders. *Neurocomputing, 273*, 643-649.

Zeng, Z., Pantic, M., Roisman, G. I., & Huang, T. S. (2008). A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *IEEE Transactions on pattern analysis and machine intelligence, 31*(1), 39-58.

Zhang, K., Huang, Y., Du, Y., & Wang, L. (2017). Facial expression recognition based on deep evolutional spatial-temporal networks. *IEEE Transactions on Image Processing, 26*(9), 4193-4203.

Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters, 23*(10), 1499-1503.

Zhang, L., & Zhang, B. (1999). A geometrical representation of McCulloch-Pitts neural model and its applications. *IEEE Transactions on Neural Networks, 10*(4), 925-929.

Zhang, S., Zhao, X., & Lei, B. (2012a). Facial expression recognition based on local binary patterns and local fisher discriminant analysis. *WSEAS transactions on signal processing, 8*(1), 21-31.

Zhang, S., Zhao, X., & Lei, B. (2012b). Facial expression recognition using sparse representation. *WSEAS transactions on systems, 11*(8), 440-452.

Zhang, S., Zhao, X., & Lei, B. (2012c). Robust facial expression recognition via compressive sensing. *Sensors, 12*(3), 3747-3761.

Zhang, Z., Luo, P., Loy, C. C., & Tang, X. (2014). *Facial landmark detection by deep multi-task learning.* Paper presented at the European conference on computer vision.

Zhang, Z., Lyons, M., Schuster, M., & Akamatsu, S. (1998). *Comparison between geometry-based and gabor-wavelets-based facial expression recognition using multi-layer perceptron.* Paper presented at the Proceedings Third IEEE International Conference on Automatic face and gesture recognition.

Zhao-Yi, P., Yan-Hui, Z., & Yu, Z. (2010). *Real-time facial expression recognition based on adaptive canny operator edge detection.* Paper presented at the 2010 Second International Conference on MultiMedia and Information Technology.

Zhao, G., Huang, X., Taini, M., Li, S. Z., & PietikäInen, M. (2011). Facial expression recognition from near-infrared videos. *Image and Vision Computing, 29*(9), 607-619.

Zhao, G., & Pietikainen, M. (2007). Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Transactions on pattern analysis and machine intelligence, 29*(6), 915-928.

Zhao, K., Chu, W.-S., & Zhang, H. (2016). *Deep region and multi-label learning for facial action unit detection.* Paper presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Zhao-Yi, P., Yan-Hui, Z., & Yu, Z. (2010, April). Real-time facial expression recognition based on adaptive canny operator edge detection. In *2010 Second International Conference on MultiMedia and Information Technology* (Vol. 2, pp. 154-157). IEEE.

Zhao, X., DellandréA, E., Zou, J., & Chen, L. (2013). A unified probabilistic framework for automatic 3D facial expression analysis based on a Bayesian belief inference and statistical feature models. *Image and Vision Computing, 31*(3), 231-245.

Zhao, X., & Zhang, S. (2012). Facial expression recognition using local binary patterns and discriminant kernel locally linear embedding. *EURASIP journal on Advances in signal processing, 2012*(1), 20.

Zhu, X., & Ramanan, D. (2012). *Face detection, pose estimation, and landmark localization in the wild.* Paper presented at the 2012 IEEE conference on computer vision and pattern recognition.