Emotion detection in videos using non sequential deep convolutional neural network

Author

Haider Riaz

Regn Number

2016-NUST-Ms-CE-00000171968

Supervisor

Dr. M. Usman Akram

DEPARTMENT OF COMPUTER ENGINEERING

SCHOOL OF MECHANICAL & MANUFACTURING ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

Feburary 2018

Emotion detection in videos using non sequential deep convolutional neural network

Author

Haider Riaz

Regn Number

2016-NUST-Ms-CE-00000171968

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Computer Engineering

Thesis Supervisor:

Dr. M. Usman Akram

Thesis Supervisor's Signature: _____

DEPARTMENT OF COMPUTER

SCHOOL OF MECHANICAL & MANUFACTURING ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY, ISLAMABAD

JANUARY, 2018

Declaration

I certify that this research work titled "*Emotion detection in videos using non sequential deep convolutional neural network"* is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources it has been properly acknowledged / referred.

Signature of Student

Haider Riaz

2016-NUST-Ms-CE-00000171968

Plagiarism Certificate (Turnitin Report)

This thesis has been checked for Plagiarism. Turnitin report endorsed by Supervisor is attached.

Signature of Student

Haider Riaz

Registration Number

2016-NUST-Ms-CE-00000171968

Signature of Supervisor

Dr. M. Usman Akram

Copyright Statement

- Copyright in text of this thesis rests with the student author. Copies (by any process) either in full, or of extracts, may be made only in accordance with instructions given by the author and lodged in the Library of NUST School of Mechanical & Manufacturing Engineering (SMME). Details may be obtained by the Librarian. This page must form part of any such copies made. Further copies (by any process) may not be made without the permission (in writing) of the author.
- The ownership of any intellectual property rights which may be described in this thesis is vested in NUST School of Mechanical & Manufacturing Engineering, subject to any prior agreement to the contrary, and may not be made available for use by third parties without the written permission of the SMME, which will prescribe the terms and conditions of any such agreement.
- Further information on the conditions under which disclosures and exploitation may take place is available from the Library of NUST School of Mechanical & Manufacturing Engineering, Islamabad.

Acknowledgements

I am thankful to my Creator Allah Subhana-Watala to have guided me throughout this work at every step and for every new thought which You setup in my mind to improve it. Indeed I could have done nothing without Your priceless help and guidance. Whosoever helped me throughout the course of my thesis, whether my parents or any other individual was Your will, so indeed none be worthy of praise but You.

I am profusely thankful to my beloved parents who raised me when I was not capable of walking and continued to support me throughout in every department of my life.

I would also like to express special thanks to my supervisor Dr. M. Usman Akram for his help throughout my thesis and also for DIP and Computer Vision courses which he has taught me. I can safely say that I haven't learned any other engineering subject in such depth than the ones which he has taught.

Along with my advisor, I would like to acknowledge my entire thesis committee: Dr. Arslan Shokat, Dr. Sajid Gul Khawaja and Dr. Farhan Hussain for their cooperation and prudent suggestions.

Finally, I would like to express my gratitude to all the individuals who have rendered valuable assistance to my study.

Dedicated to my exceptional parents: ***Riaz Ahmed & Rukhsana Riaz***
and adored siblings whose tremendous support and cooperation led me to this wonderful accomplishment.

Abstract

Emotions are fundamental for humans. They affect perception and everyday activities such as communication, learning and decision-making. Facial expression and body language are the main sources of this information. The goal is to classify these emotions to improve human-computer interaction. In proposed method, a non-sequential deep convolutional neural network is presented. It consists of multiple networks which run in parallel. These parallel networks are then merged together followed by relu, max-pool, drop-out, dense and soft-max layers. In proposed model, we have used multiple networks to cover local and global feature. After feature extraction from CNN, they are fed to Recurrent Neural Network (RNN) using Long Short- Term Memory (LSTM) layer in which time dependency is included. Every current output is dependent on previous all outputs. This way a sequence is learned in complete video. After that score based voting system is used to finally assign emotion to video. The evaluation of proposed method is done by using Surrey Audio-Visual Expressed Emotion (SAVEE) dataset containing four actors and Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS) containing 24 actors, covering seven emotions in their videos. K fold testing is used for evaluation of our proposed model. Results obtained from each dataset were extremely positive and the recognition rates 99.64% on SAVEE and 87.49 on RAVDESS were among the highest ever achieved.

Key Words: *Emotion Detection, Non Sequential Neural Network, Deep Convolutional Neural Network, Deep Learning, CNN-LSTM*

Table of Contents

List of Figures

List of Tables

Chapter 1: INTRODUCTION

1.1 Description and Motivation

Humans need expressions to effectively communicate a message. We as emotional beings cannot properly interact or convey our message to one another without emotions. We need these emotions to convey our feelings and intentions. Similarly, our interactions with machines can improve significantly if they are able to decode our message completely, which is possible only if they are able to distinguish between the basic emotional states of the person who is delivering the message. Emotion recognition using audio-visual information has a much bigger role to play in the field of Human Computer Interaction (HCI). This area of research has numerous applications in robotics, health and mobile computing, just to name a few and has an ever-growing demand in industry as we are becoming more and more reliant on machines to carry out our day to day tasks. For that purpose, we need intelligent machines that can learn adapt, control and more importantly understand human behavior.

1.2 Problem Statement:

For us, emotion recognition may seem simple because we are naturally designed to discern and understand them with ease but same cannot be said for the computers. Special learning techniques and training algorithms are required for this process. A lot of work is being conducted in this area of research. Accurate emotion recognition using a combination of computer vision and machine learning techniques is an immense task and requires attention on several issues that if neglected, can have a highly negative impact on system's performance. Most common factors among these include noise, voice quality, illumination changes, presence of occlusions and more importantly, diversity in facial textures and non-identical emotional expressions among different people. So we explore Convolutional Neural networks and its variants for recognition of human emotions from videos.

1.3 Application:

The classification of emotions is a growing area of research with emerging applications in computer vision industry. Few applications are Security systems, Interactive Computer Simulations, Psychology and Driver Fatigue Monitoring.

1.3.1 Security System:

These days street crimes happens a lot. If CCTV cameras are deployed with emotion detection algorithm they can also detect for any suspicious behavior from any person using emotions and can alarm any suspicious activity or crime before it happens.

1.3.2 Smart Cars:-

Car companies are trying to make their cars more personal and safe to drive. They have introduced AI in their cars to understand human emotion and act accordingly. Through emotions computer can understand when driver is fatigue and alarm him/her.

1.3.3 Emotion Detection in Interviews:

To understand candidate personality for the job, companies are deploying AI to detect candidate emotions to understand their moods and assess their personality. It is also helpful in detecting lies during interviews through sudden changes in emotions.

1.3.4 Market Research:

Market research companies have deployed CCTV cameras in shopping malls which can monitor consumer behavior using emotion detection and can understand recent trends where consumers are showing more interest which can be helpful in capturing market.

1.4 Aims and Objectives:

Major objectives of this research are as follows:

- Review and comparison of recent and most impactful advancements in automatic emotion recognition systems.
- Designing of a unique Neural network architecture which can identify unique features from videos and classify accordingly.

1.5 Structure of Thesis

The work on this thesis is structured as follows:

Chapter 2 covers the importance of emotions and how they are categorized and discusses a basic emotion recognition model and neural networks.

Chapter 3 includes review of the literature and discusses the significant work done in the past.

Chapter 4 presents the proposed methodology.

Chapter 5 details the experiments performed and the databases employed in this research.

Chapter 6 draws conclusions from this study and discusses the future scope of this research.

Chapter 2: Theoretical Background

Relevant concepts will be reviewed in this chapter. Here, the background of important studies (reviewed in this report) will be discussed. Chronological revision of machine learning, affective computing, and other relevant fields are considered to complete this theoretical background. The top down method has been utilized to explain concepts. This chapter also has some discussion about the related subjects.

2.1 Affective Computing

It arises from, relates to, or influences effective phenomenon such as emotions, according to Rosalind Picard [1]. Decision making, communication, and learning are key human experiences that are based on emotions. So the main aim of affective computing is to incorporate emotions into computing.

Affective computing is of the view that the concept of humans as rational decision-making beings is not complete without emotions. A school of though is of the view that so-called "pure reason" does not exist. It true that emotions play an important role in routine decision-making because making decisions would become much time-consuming in the absence of emotions. On the other hand, some researchers elaborate how emotions make our decision biased [2].

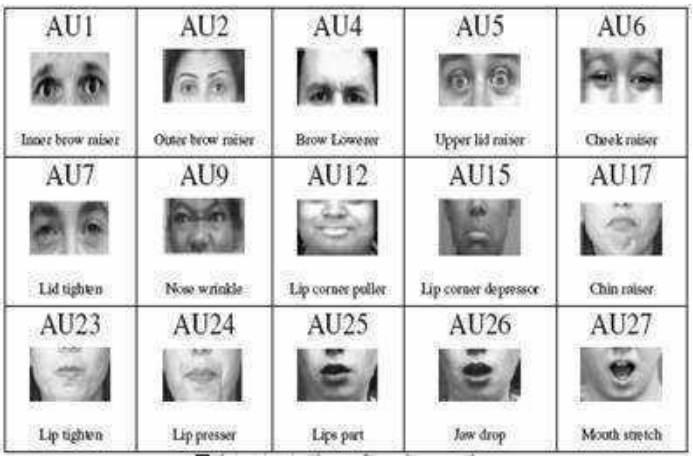

Figure 2-1 Facial action units [72]

Emotions are a class of qualities that associate with motor system intrinsically. The mind mostly has a set response for a particular state of emotion [3]. It is strange that the importance of emotions has been studied without taking human interaction into consideration. The human beings have a rare quality known as empathy that helps them feel or understand what other person, animal, or thing is going through [4]. Furthermore, it is empathy, that allows humans to establish close relationships with communities. The quality of empathy plays a crucial role in pro-social behavior (social perception and interaction) and is fundamental to developing pro-social systems [5]. Due to these reasons, effective computing should develop a proper mechanism to measure modulations thereby understanding the emotional state of a subject. Detecting vocal and facial emotions are two ways to do so. This project only explores facial emotions.

2.1.1 Facial Emotion Recognition

The basic research in this domain was first conducted by psychologist Paul Ekman. Ekman's Facial Action Coding System has supported most of contemporary studies on this subject [6]. This system provides a mapping an emotion space and facial muscles. In this comprehensive system, the movements of human face depend on the appearance of the face. This classification was first developed by a Swedish anatomist Carl-Herman Hjortst. Some facial gestures that correspond to a set of facial emotions units have been displayed in Figure 2.1.

There are some challenges facing this mapping. One of the challenge is that this mapping cannot identify when a person is acting and when a person is serious. It is true that human beings often make wrong gestures. Let take example of a person who is paralyzed and someone asks that person to smile. So paralyzed person is only able to smile with half face. On the other hand, both sides of the face raise when a joke is told. It shows different emotions originate from a different path.

Though expressions and emotions can be difficult to recognize, there are some opportunities that can help with computers make the right choice. It is possible these days to imitate the facial units of Ekman. This will provide computer with graphical faces that provide a more natural interaction [7]. It is possible for computers to recognize some emotions such as anger, happiness, disgust, and surprise [8]. The section 2.6 focuses on facial emotion recognition.

2.2 Machine Learning

Machine Learning (ML) stems from Artificial Intelligence Arthur Samuel first presented a concept similar to machine learning in 1959 as. He stated that Machine Learning allows computer systems to learn and decide without explicit programming. 60 years later, we have realized this concept to some extent. At that time, this concept was a kind of superficial but many people got excited about it and explored it one way or another. This domain of Artificial Intelligence differs from other fields because a feature needs to be added in other fields and self learning does not exist in there. Let take example of traditional software development. A programmer has to change the code if new feature needs to be added. Machine learning does not work this way because the software enables the machine to learn using existing set of instructions. based on input data, models are created by machine learning. In turn, output is generated by these models in the form of decisions or predictions. That is why system is able to take decision when a new requirement appears.

There are three main categories of machine learning. These categories consider a learning system executes learning process. These three categories include: reinforcement learning, supervised learning, and unsupervised learning.

When labeled inputs enter a model, it is known as supervised learning. This type of machine learning contains related belonging class. The model used here adapts itself in such a way that inputs correspond to the outputs. On the other hand, unlabeled inputs are received in unsupervised learning. Due to this, model explores patterns to learn from the data. In reinforcement learning, the third type of machine learning, an agent is punished or rewarded based on the decision it took in for achieving a goal.

Supervised learning will be discussed in this study because images being studied here are labeled. Here, label means emotion that the image represents.

2.3 Artificial Neural Networks

There are tools in supervised learning that aim at solving problems within its domain. Artificial Neural Networks (ANN) is one of those tools. ANN represent a set of instructions and it performs label predictions. Analyzed as a black box, ANN's input contains labeled examples and output contains a vector with predictions. Here, a probability distribution for all labels represent predictions [9]. Simple processing units make ANN and it represents huge parallel distributed processor. ANN has the ability to store practical knowledge for use future [10].

It (ANN) is not always massive. New ideas can be tried by using small implementations. This definition from Engelbrecht focuses more on topology. Artificial neurons form this layered network. Hidden layers, an input layer, and an output layer form an ANN. Biological neurons form artificial neural model.

The following three steps can explain ANN:

1. Data input into the network.
2. Input gets transformed with the help of a weighted sum
3. Previous transformation's turns into an intermediate state with the help of

It is clear from these above three steps that all of then form a layer. A layer shows a block with the highest-level on a network. We can represent a transformation as a neuron or a unit. In the end, the intermediate state becomes a input for the next state of layer. The topology of an ANN is exhibited in figure 2.2:

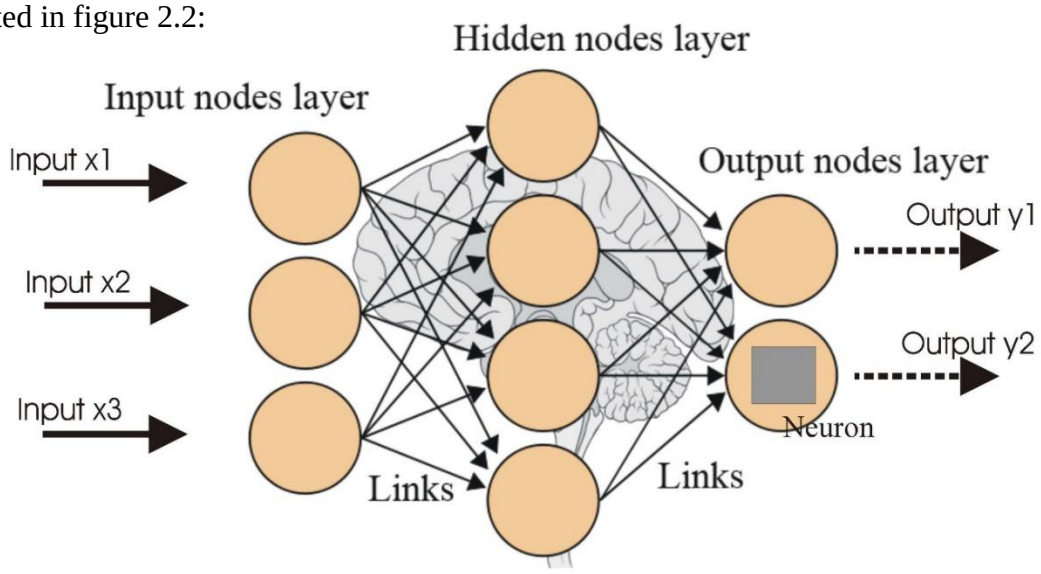

Figure 2-2 Artificial neural network topology [52]

When we consider the definition of Engelbrecht, an interesting question comes to mind. If human brain inspires this model, when did the computational model come from? The historical background needs to be studied to find answer to this important question.

2.3.1 Rise and Fall of ANN

We can trace back ANN's history to the 1940s. Back then, different scientists tried to explore the brain structure but computational model was not applied at that time. However, two researchers named Warren McCulloch and Walter Pitts tried to formulate an ANN as a model suitable in 1943 to perform computations [11]. Later, in 1949, a theory for formulated by Donald Hebb; he discussed that learning process involves adapting on the brain [12]. Then it took ten years for researchers to implement perceptron (an ANN implementation). A researcher Frank Rosenblatt first studied ANN implementation known as perceptron [13] and it is considered as the simplest ANN architecture. Furthermore, it was the first time that an ANN was able to learn by means of supervised learning.

The topology of a perceptron is introduced by in the figure 2.3 below. It is good that simple architectures can explain most of ANN models. One can find inputs X_1 and X_n in the figure 2.3. This layer represents the input layer. W_n is the corresponding weight for each input.

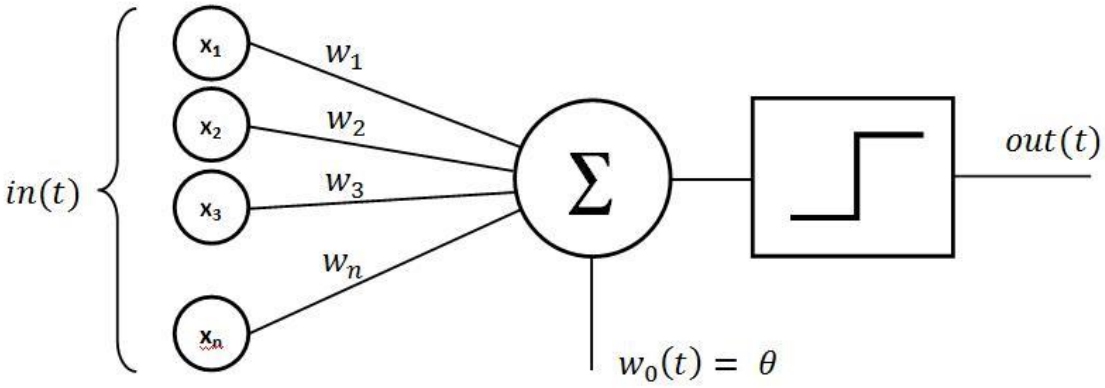

Figure 2-3 Perceptron topology [73]

A weighted sum is performed on the neuron (unit). Moreover, in order to make neurons to implement a linear function, a bias is added. The curver will move on the X-axis due to the independence of the bias.

$$y = f(t) = \sum_{i=1}^{n} X_i * W_i + \Theta$$

Furthermore, the result of f(t) is the input of an activation function. The output of the node is defined by the input. The binary step function suits this topology as the perceptron is a binary classifier. 0 and 1 (a couple of input classes) will be the output.

$$output = \begin{cases} 0, & y < 0 \\ 1, & y \geq 0 \end{cases}$$

In the end, the real value is used to measure against the prediction. The weights on the first layer are updated by using this error signal which helps in predicting the results. This process works through trough backpropagation learning [14].

ANN has been a hot research topic back in the 1960s. Minsky and Papert, however, published a paper in 1969 that put an end to this golden era [15]. They, actually, discussed some limitations of perceptron that discouraged other researchers to continue working on it. Minsky and Papert discussed how general abstractions would not be suitable in it. Furthermore, they stated that it would not be possible to overcome these limitations by deriving complex architectures. Though these limitations were major obstacles for progress of ANN back then, it has now become possible to overcome such challenges successfully. The start of the 1970s, was not a prosperous period for artificial intelligence discipline. That is why the early years of the 1970s are termed as the "AI Winter" due to various unsuccessful projects.

2.3.2 ANN Revival

From the mid-1970s to mid-1980s, the shift on Artificial Intelligence completely moved to symbolic processing. By the mid-1980s, ANN started flourishing again due to a set of factors belonging to different areas. As the symbolic processing showed slow progress, one of the factors that helped ANN was that symbolic processing was narrowed to small simulations. One of the

other major reasons was the advanced hardware and better computer availability which was non-existent in the previous decades.

By the late 1980s and early 1990s, computer technologies became advanced and connectionism progressed. Some important implementations and simulations were not possible in the second half of the 20th Century. Even in 1988, Charles R. Rosenberg and Terrence J. Sejnowski conducted an important research study about NETtalk [16]. As a notable ANN, NETtalk was trained or designed to pronounce English words. As a result, connectionism became popular.

After one year, inspiring results on handwritten recognition of zip code by using a multilayer ANN were exhibited by Yann LeCun. Being the first work on consolidation of machine learning and image recognition, this paper came up with promising results. Moreover, the concepts about weight sharing and feature maps, which are convolutional neural networks, were also introduced in this paper [17].

Many researchers made use of support vector machines (SVM) during the 1990s. They used such machines because they were simpler than that of ANN. Moreover, SVMs were able to deliver better results. At that time, ANNs were famous again but there were other famous topics in the machine learning domain. Though slow and steady progress kept on, the real growth started in the early 2010s. The world has been able to generate a great quantity of data over the past decades. Furthermore, some special computational devices are touching the height of innovation. The best thing is; the cutting-edge technology is not expensive anymore and notable individuals and organizations have access to them. It is not the era of Big Data which is has taken Machine Learning to a new level. Today, it is possible to perform huge computations. Thus, Deep Learning has taken the place of ANN.

2.4 Deep Learning

We can say that the Deep Learning domain came into existence due to evolution of ANN. Two or three non-linear hidden layers are used in Deep Learning to train a system. The fields of natural language processing, computer vision, and automatic speech recognition have benefited due to DL (Deep Learning). Feature engineering is not required in DL which is a great headway. Here in DL,

algorithms have the ability to learn features by themselves. Let's take the example of image recognition. pixel representations of images fuel an ANN. After that, the DL algorithm will be able to decide what pixel combination relates to a particular feature. This process will be repeated for the whole image. Data is processed through layers here and meaningful representations of objects is obtained through abstract forms.

DL has started to become famous due to constant achievements. A 2012 paper on its application into automatic speech recognition (a major industrial application) conducted many benchmarking tests that explained how ANN worked better than the Gaussian mixture models [18]. This important research study was supported by four research groups: Microsoft Research, Google Research, University of Toronto, and IBM Research. Another breakout publication appeared in 2014 that advanced natural language processing [19]. In this study, researchers elaborated how Long-Short Term Memory is better than statistical machine translation. The default tool for translation at that time was statistical machine translation. Whereas, LSTM or Long-Short Term Memory is a special ANN architecture also known as recurrent neural network. English to French translations have been performed through this network.

Finally, Convolutional Neural Networks (CNN), a deep learning technique is explained because it is relevant for this project. A team of researchers from Toronto University [20] published a paper in 2012 that came up with better results on the ImageNet classification competition. This important study laid the foundations for modern DL. On the 2012 edition, its solution using deep CNN achieved an error rate of 15.3% on top-5 classification while the second best achieved 26.2%. The section 2.5 of this paper will share more details about CNN. This research study is designed around the use of GPU for training and the rectified linear unit's use as the activation function [21] which are two important Machine Learning concepts.

2.4.1 Rectified Linear Unit

An ANN architecture necessarily has the activation function of a unit (neuron). Since the early days of ANN, researchers have been using different functions. But a good error approximation is not possible due to step function's binary nature.

The sigmoid functions were utilized to overcome this challenge. They were used to provide promising results for small networks. However, it was not appropriate to scale sigmoid function on large networks [20]. As it could lead to huge numbers, the cost for computations was too high [21]. The gradient vanishing problem was among other significant issues with sigmoid function. The prevention of learning occurs due to high value of gradient value [22][23].

Compared to previous common activation functions, the rectified linear unit function (ReLU) provided benefits in this situation. It did not suffer from the gradient vanishing problem and provided a good error approximation but it was used to cost less. The figure 2.4 displays ReLU and it is also mentioned below:

$$\mathbf{f(x) = max(0, x)}$$

The research conducted by Krizhevsky et al. [20] elaborated that the use of ReLU lowered the epochs's number required to converge when using Stochastic Gradient Descent by a factor of 6. There is a major drawback of ReLU's use which is the weakness when input distribution is below zero. It is due to the reason that neuron will not activate by any data point. Use of GPU. There are practical reasons to train deep networks with the help of a GPU.

To reduce training time compared to CPU training is the main reason [24]. Though the speed depends on the network topology, the use of GPU provides 10 times faster speed [25].

The way of processing different tasks is what differentiates GPU from CPU. A few cores are used in CPUs to execute sequential serial processing. However, GPU represents a mighty parallel architecture. To manage several tasks at the same time, thousands of tiny cores work in harmony in this architecture.

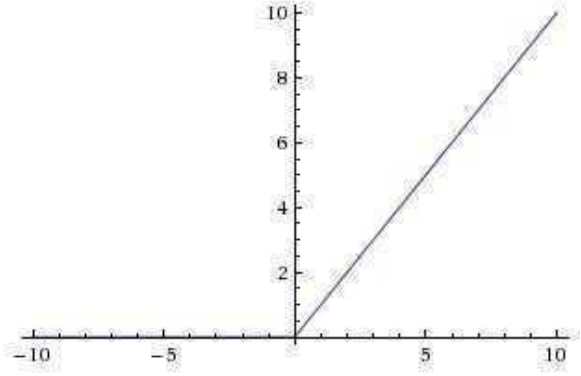

Figure 2-4 Rectified Linear Unit (ReLU) [74]

From the above discussion, it is clear that the Deep Learning works better with GPU rather than CPU. However, most of projects still make use of CPU due to different reasons.

2.5 Convolutional Neural Networks

The development of the convolutional neural networks is related to the study of visual cortex. Hubel and Wiesel conducted a study in 1968. The monkeys visual cortex's receptive fields were under consideration here [26].

Two types of cells, complex and simple, are also presented in it. The complex cells are locally invariant and they focus on a broader spectrum of objects while the simple cells cover edge-like shapes. Therefore, shapes and objects' correlation is exploited in local visual areas. Resultantly, the whole visual field is mapped by sets of cell arrangements.

Neocognitron was one of the early implementations based on the ideas of Hubel and Wiesel. Kunihiko Fukushima developed a neural network model known as Neocognitron [27] in 1980. The first model layer is formulated by simple cells represented by units. Whereas, the complex cells are represented by the second layer units. One of the greatest achievements of Neocognitron includes the implementation of the local invariant property. Moreover, there is one to one output mapping in it. One and only one specific pattern is mapped by a complex cell.

The learning process of Neocognitron is the main drawback. In the past, there was no method to measure errors. In 1985, the backpropagation modern form [28] derived by Finish Mathematician Seppo Linnainmaa in 1970 [29] [30] also got applied in ANN. But its use was limited at that time and few applications were developed using backpropagation [28]. The use of backpropagation was introduced into ANN in 1985 by Hinton, Rumelhart, and Williams [31].

The gradient of the error is measures by backpropagation with respect to the weights on the units. Gradient changes with weight value increase or decrease. After that, the gradient will be used to find weights thereby minimizing the network error. When using backpropagation, the network is able to autotune its parameters with some optimizer such as Gradient Descent (GD) which is a first-order optimization algorithm. GD finds a function's local minimum [32]. The typical

backpropagation steps have been highlighted in figure 2.5. For more details, literature [33][34] can be consulted.

As mentioned before, the groundbreaking research on CNN has been credited to Yann LeCun. The classifier on handwritten digits (MNIST) LeCun's was the first accurate backpropagation practical application [17]. As this system of LeCun was able to read a huge amount of handwritten checks, it is regarded as one of the highly successful and reliable uses of CNN. It was the research of LeCun that resulted in the topologies of CNN. The most popular CNN topology is LeNet-5 [35].A document recognition experiment used the implementation of LeNet-5 topology.

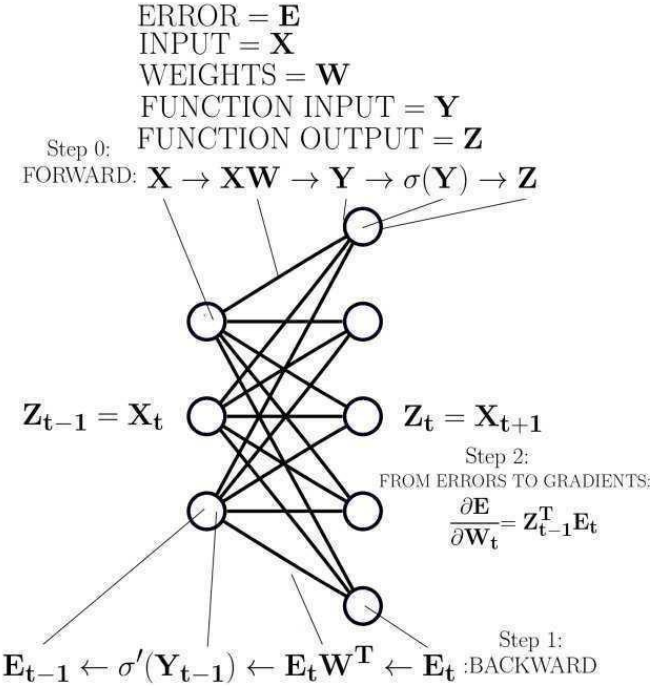

Figure 2-5 Backpropagation algorithm [70]

As per this research, hand designed solutions are not good to solve pattern recognition problems. However, the use of automatic learning solutions is better in this regard. The machine learning is better for this because it is quite a complex task to address all possible events that can naturally occur. There are two main modules in this system: a feature extractor and a classifier.

CNN's key components are: dropout, local receptive field, convolution operation, weight sharing, spatial sub-sampling, and stochastic gradient descent. The components of CNN will be discussed next.

2.5.1 Convolution Operation

Convolution, a Mathematical concept, is a mixing of two functions. It is a Mathematical operation that works as a filter. To stop unnecessary thing, a kernel filters all inputs allowing only specific information.

The following two elements are needed to execute this operation:

- The input data
- Kernel or the convolution filter

This operation results in a feature map. A graphical explanation of how a convolutional operation works has been shown in figure 2.6. Feature maps' (output channels) number helps neural networks learn new features. Here, all channels are independent because they try to learn new features.

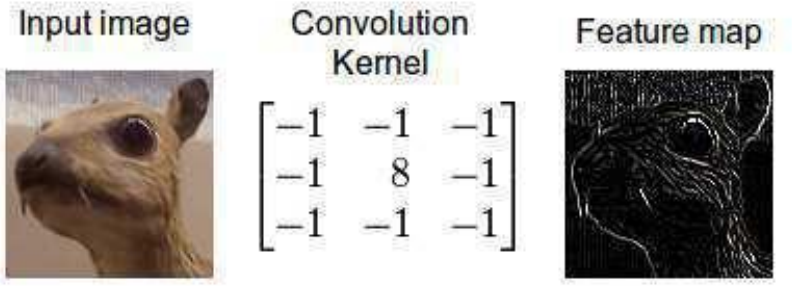

Figure 2-6: Convolution Operation [74]

In the end, we can say that when performing the convolution, the algorithm to be used can be defined by the type of padding. There edges of input have a special case. The border of input will be discarded by one type of padding because there is no more input next to it that can be scanned. On the contrary, the input value of 0 will be completed by the other padding. The parameters are reduced during convolution operation. Refer to the related research [36] for more details about this

2.5.2 Weight Sharing

Regardless of its position, intuition says that a detected feature is always meaningful. high-dimensional input's translationally-invariant structure is exploited by weight sharing. Mean to say, the image of a cat is not recognizable due to cat's position. Another example is; the position of a noun should not change the meaning of a sentence.

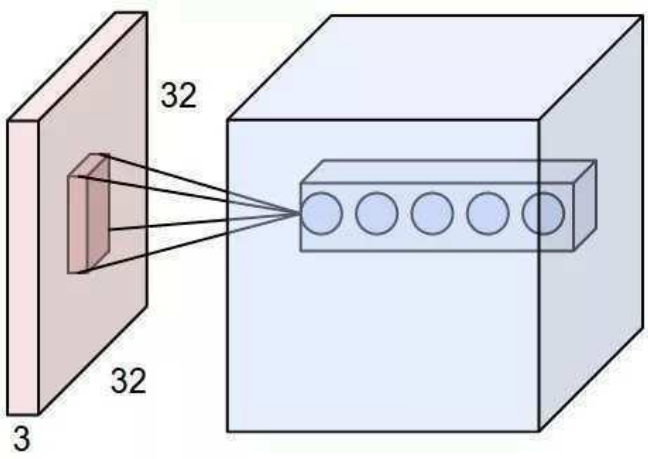

Figure 2-7: Local receptive field of size 5x5x3 for a typical CIFAR-10 image, 32x32x3 [74]

A plane is generated after the convolution operation. This place is normally composed of the results of applying the same filter through the entire input. This plane is named feature map. A convolutional operation with a kernel results in a feature map. Different weights initialize kernels to perceive different features. This way, the feature stays for the whole feature map having irrelevant position with respect to the network.

There is a set of feature maps in a convolutional layer that extract different features at each input location. Convolution operation means the process where input is scanned and unit state is stored on the feature map.

2.5.3 Local Receptive Field

A kernel size, local receptive field and filter size are similar terms. Actually, it refers to an area where a neuron gets connected in case of high-dimensional input. Hyperparameter of the network is the local receptive field. Mean to say, the shape has been defined in advance.

The locally-sensitive neurons on the visual cortex have a strong influence on this concept. Here, neurons are not connected to the whole input space. Only locally-connected areas are focussed on. Such connections happen only on height and width dimensions. A local-connection is not used for input depth dimension; it is fully-connected through all channels. For example, an image can be a high-dimensional input. Three dimensions, height, width, and depth represent an image. The image's kernel acts locally on height and width when a local receptive field is applied over an image. Mean to say, nothing happens to the depth. We can number of channels correspond to the depth dimension. Mean to say, red, blue, and green channels exist in an RGB image. The resultant image has three images in three colors: blue, green, and red.

The input can be allocated using spatial sub-sampling. Here, elemental features (edges, end-points or corners) can be extracted by the neurons. The higher-order features can be extracted by the network by implementing this idea to other layers. Overfitting can be avoided through the reduction of connections.

It can be seen in figure 2.7 how one neuron is linked to a feature map of 5x5x3. The process works for an entire image. After this process, the height and width of the input will decrease. This does not happen in case of input depth dimension.

When the convolution operation ends, the filters applied to the input shows the depth dimension. Different features in the image can be captured using a set of filters. Different weights initialize a filter. For a given filter, weights remain the same during convolution and this is known as weight sharing.

Feature learning is the main focus of such operations instead of classification. Both capabilities can be obtained by using fully connected layers. The convolutional layers' weights and back propagation style stochastic gradient descent can be used to optimize these layers.

2.5.4 Spatial Sub-Sampling

Pooling is also known as spatial sub-sampling. It refers to an operation where values of a given area are reduced to a single value. The feature position's influence on feature map can be reduced by reducing its spatial resolution. After a convolution operation, this operation can be performed by selecting a pixel with high response rate.

Maximum and average are two polling types. The mean on the defined area is computed by the average polling. The highest value on the area can be selected by the maximum polling. In case of a large value, the predicted performance can reduce. Its working mechanism is similar to a convolution operation. The pixel having more value is picked by this operation for a given filter region [37].

The feature map's dimension reduces this way. To ability of the system to learn feature by position is prevented by this reduction. Afterwards, the feature can be generalized for new examples. As features often have different positions, this is important.

2.5.5 Dropout

By using dropout function, the impact of units with a strong activation can be reduced to a minimum value. To enable other units to learn features automatically, dropout method shutdowns units during training. All units need specific level of independence to reduce strong unit bias. This leads to better generalization and strong regularization.

2.5.6 Stochastic Gradient Descent

Just one respect differentiates Stochastic Gradient Descent (SGD) and Gradient Descent (GD) and. The different lies in total examples that are looked at to enumerate parameters' gradients. All training set examples are used by the original version that performs this operation.

Chapter 3: LITERATURE REVIEW

Emotion recognition and analysis based on the video or audio data is a research hotspot in computer vision and affective computing domain. Humans are highly social species and mostly they interact and exchange information by their face gestures. Using these gestures one can understand emotions and support all facts of life. With the dawn of new digital technologies, growth of globalization, it is needed to understand which facial expressions are related to social communication and those which causes misunderstanding should be avoided. It has been shown that understanding face expressions can alter the interpretation of what is actually spoken and these expressions control the way in which conversation should take place.

Now let us dig a little deeper in this area of research by studying and analyzing the past works, conducted by various researchers

3.1 Classical techniques of Emotion Detection:

Wang *et al.* [38] investigated on kernel based methods to identify emotions from the audio-visual information. Hamming window was used, having a size of 512 points while keeping 50% overlap ratio between successive window frames. From these frames, audio features such as MFCC coefficients, pitch and power were evaluated. Face area in frames was detected in HSV color space using the Planer envelop approximation method and its parameters were tuned in order avoid false face detections. After detecting face areas, facial feature were extracted by using Gabor filters having filter bank of 5 scales and 8 orientations. To reduce the dimensionality of the extracted features, Gabor coefficients were down sampled and principal component analysis (PCA) was applied to further reduce the dimensions of the feature set and to decrease the computational complexity of the problem. Kernel Matrix Fusion (KMF) is employed to map multi-modal features into a single subspace. Kernel matrixes are developed separately for each modality, followed by the application of unsupervised Kernel principal component analysis (KPCA) or supervise Kernel Discriminant Analysis (KDA) and the transformed KPCA features are passed on to the HMM classifier. RML dataset was employed for this experiment and the proposed method outperformed other strategies including CCA, KCFA, feature level and score level fusion.

Haq *et al.* [39] evaluated emotional states on SAVEE dataset in his audio-visual emotion recognition experiment. For that purpose, features were extracted from both modalities separately and fusion process was tested at both feature and score level stage. Features such as MFCC, pitch, energy and duration were extracted from the audio input whereas visual features were extracted on the basis of positions of 2D marker coordinates on face areas. After extraction, distinctive features were selected using Plus *l*-Take Away *r* algorithm which is based on the Mahalanobis and Bhattacharyya distance having selection on basis of KL-divergence. After the useless features were discarded, the remaining features were dimensionally reduced by the application of PCA and LDA and were passed on to the Gaussian classifier. Feature and score level fusion strategies were put to test and the results displayed high classification accuracies, with visual recognition approach and decision level fusion delivering better accuracies than audio recognition and feature level fusion strategies respectively.

Rashid *et al.* [40] used spatio-temporal features obtained from the visual streams which were dimensionally reduced using Principal Component Analysis (PCA). MFCC and some prosodic features are identified as audio feature representatives. Codebook was formulated for both audio and visual features in the Euclidean space after the application of PCA and SVM was used for classification of emotional states and final class prediction was derived on basis of the prediction values coming from each classifier using Bayes Sum Rule (BSR). Visual features outperformed compared to audio but combination of both modalities generated even better results.

Moreover, Action Units (AUs), valenceand arousal space (i.e., V-A space) are proposed by

Chang *et al.* [41] to model facial behavior.

3.2 Machine learning Based Emotion Detection:

KalaiSelvi *et al.* [42]**,** presented the system for recognizing emotions through facial expression displayed in the video. Face emotions are recognized by using Dynamic Bayesian Network and achieved a promising success rate. Researchers in the field of computer vision have tried to develop Micro-Expressions (MEs) detection and recognition algorithms but the biggest obstacle has been the lack of a suitable database.

However, Grobova *et al.* [43] made a hidden sadness database, which includes 13 video clips. They proposed a new approach for automatic hidden sadness detection algorithm, and applied Support Vector Machine and Random Forest classifiers are applied for state of the art accuracy.

Zhalehpour *et al*. [44], presented a new audio-visual dataset in his paper called BAUM-1 database. This dataset comprises of six different emotional stares enacted by 31 subjects, 17 of which were female. The author used this dataset along with eNTERFACE in this audio-visual emotion recognition experiment and compared their results. Peak frame was selected in videos using maximum dissimilarity (MAXDIST) based peak frame selection which selects frames on the criteria of "maximum dissimilarity". After that, Linear Phase Quantization (LPQ) features were extracted along with Patterns Oriented Edge Magnitudes (POEM) features as the visual set of features from these databases. Linear Phase Quantization (LPQ) is similar to Local Binary Patterns (LBP) in a way that both produce feature vectors based on local histograms. LPQ were preferred because they offer better results compared to the well-known LBP. Furthermore, POEM features were calculated because of their robustness towards illumination changes as they neglect the pixel intensities and only consider gradient magnitudes. They also provide both local and global information compared to LBP which only provide local information. In addition to the visual features, audio features such as Mel-frequency Cepstral Coefficients (MFCC) and Relative Spectral Feature based on Perceptual Linear Prediction (RASTA-PLP) were formulated using 12 and 20 order filters respectively, with an overlap ratio of 50% and a window size of 25 msec. First and second order derivatives were calculated for the already obtained coefficients and several statistical functions were applied such that the final feature vector obtained had 675 distinct features, which were then used for classification. State of the art SVM was used as a classifier for both modalities. For video features, linear kernel was employed to neutralize the curse of dimensionality while a radial basis kernel was selected to classify audio features. The outputs from both classifiers were fused on the basis of weighted product rule where the confidence values obtained from each classifier for a video sequence are multiplied and the label of the one with the maximum product is selected.

Seng *et al.* [45] used a combined machine learning and rule-based approach to solve multi-modal emotion recognition problem in his research. After preprocessing and extraction of the face area region, feature extraction techniques were applied. Bi-Directional Principal Component Analysis (BDPCA) and Least-square Linear Discriminant Analysis (LSLDA) were employed for visual

feature extraction and were used to discriminate between 6 emotional classes. A new fusion method known as Optimized Kernel-Laplacian Radial Basis Function (OKL-RBF) was adopted as neural classifier for the extracted visual features. Using audio information, both prosodic (pitch, log energy, zero crossing rate, teager energy operator) and spectral features (MFCC coefficients) were evaluated. Both sets of features were fused at the decision level of the audio path and outputs were calculated using combined rule-based mechanism along with two RBF classifiers. Finally, score level fusion strategy in form of a proportional weighing mechanism was employed to fuse the resultant audio and visual output. Numerous datasets were used to test this strategy including eNTERFACE and RML and results verified an increase in performance on comparison with other systems.

Cid *et al.* [46] proposed an audio-visual approach in which he employed Dynamic Bayesian Network classifier to predict six universal emotional states. For audio features, the author evaluated pitch, energy and speech rate and passed them on as input the DBN classifier. From visual content, a set of edge-based features were formulated for all the videos and neutral state features were used to normalize these features, extracted from remainder of classes, in order to make sure that the extracted features were invariant to the scale or distance between robot and the user. Just like audio modality, visual recognition also involved DBN as the prediction model. A third DBN classifier is used in a decision level fusion scheme, to fuse the confidence values of audio and visual classifiers and formulate results of its own as the final predicted output. Experiment on SAVEE database proved the robustness of this approach as high classification accuracies were achieved.

Shan *et al.* [47] estimated facial expressions using Local Binary Patterns. Various machine learning models were employed in his person independent emotion recognition system. Several dataset were used in this research including JAFEE and MMI both including seven different emotion states. The experiments showed that the LBP features are very effective for facial emotion recognition. Furthermore, Adaboost technique was used to evaluate most useful and discriminative LBP features. These boosted-LBP features were passed on to the Support Vector Machine (SVM) classifier and the results were compared with simple LBP. The best results were obtained on using SVM with RBF kernel and boosted LBP features outperformed simple LBP features on general comparison.

Dhall *et al.* [48] evaluated visual features using pyramid histogram of gradients (PHOG) along with local phase quantization (LPQ) features which is another variant of LBP. This system was tested on SSPNET GEMEP-FERA dataset and classification was carried out using SVM and largest margin nearest neighbor classifiers.

Shaukat *et al*. [49] used soft-competition based ensemble method in order to simultaneously use different feature measures in his speech emotion recognition experiment. These measures included energy, voice, frequency, duration and Mel-frequency cepstral coefficients (MFCC). Each separate feature measure was passed on to a separate support vector machine (SVM) classifier and their results were linearly combined to discriminate different emotional states. Different speech datasets like Berlin, Baby Ears, DES and GEES were employed in this study and the results showed improvement in recognition rates compared to other composite feature representation strategies.

In 2015, Shaukat *et al*. [50] proposed a facial emotion recognition system (FER) which evaluated features by using a combination of Scale Invariant Features Transform (SIFT), Gabor wavelets and Discrete Cosine Transform (DCT). In the preprocessing stage, face area was detected and set to a uniform size for every single image in the dataset. Extracted features were concatenated and passed on to a SVM classifier with a radial basis kernel function. The experiment was performed on JAFEE dataset and the results obtained proved in favor of this approach.

In 2013, Khan *et al.* [51] developed pyramids of Local Binary Patterns and used them for facial feature analysis. The new variant (PLBP) used normal local binary patterns in a pyramidal fashion such that each specific face region is divided into finer sub-regions iteratively by doubling the division at each iteration level. The author conducted a psyco-visual experiment which determined the most salient face regions for a particular expression. Conclusions drawn from this experiment showed that certain face regions were salient for specific emotion classes. Experiments were performed using SVM, Random forest, 2 nearest neighbor, C4.5 Decision tree and Naïve Bayes Classifier out of which 2NN outperformed every other classifier, obtaining highest classification results

Ping *et al.* [52] Proposed subject independent low dimensional manifold features for dynamic emotion recognition. These features helped in dynamic emotion recognition with continuous labels of emotion representation along with emotion intensity of each frame.

Emotion recognition performed by Haq *et al.* [53] uses prosodic and spectral as audio and marker locations on face as video features. After that PCA and LDA are used for feature reduction which are then fed to Gaussian classifier for emotion classification.

Zhang et al. [54] proposed simple model, single task hierarchal model and multi task hierarchal model. RVDESS dataset contain speech and song videos separately. Single task model deal both speech and song task separately whereas multi task model share both task. Although both task are apparently different but results from multi task model shows that shared model performs better than single task model.

3.3 Convolutional Neural Network Based Emotion Detection

Yan *et al.* [55] presented a bimodal emotion recognition approach using the convolutional neural networks and feature fusion method. They had cut out the facial images and get the audio emotion data from videos and then CNN and openSMILE tool are used to extract features. They have shown that the accuracy of Principal Component Analysis (PCA) and sparse kernel reduced-rank regression fusions methods are quite better as compared to EmotiW 2016.

Chao *et al.* [56] proposed a sequence encoding technique for categorical emotions in videos using Long short term memory recurrent neural networks (LSTM-RNN). Recently, researchers tackled dynamic emotion recognition based on manifold features.

Paleari *et al*. [57] developed a multi-modal model in his research. Neural Nets were employed to test the classification accuracy. Audio features were extracted using PRAAT toolkit and included MFCC, LPCC, Pitch, Energy, Harmonicity, and formant frequencies. Before facial features extraction, face area was identified and selected by applying Viola Jones algorithm. Coordinates based features were extracted from this face area and were further used in generating distance set features. eNTERFACE database was employed in this research and three different neural networks were employed for each emotion using data from audio, coordinates and the distance feature sets respectively.

Haung *et al*. [58] used neural networks to classify emotions using eNTERFACE‘ dataset. A combination of prosodic features (pitch, energy and speed) and MFCC coefficients as feature representatives from the audio modality. From the visual content, geometric features (triangular

facial features) and appearance based features (Local Principal Texture Pattern) are both used in conjunction as visual features. Back propagation algorithm was employed to train uni-modal networks for each feature vector and a collaborative decision making model was introduced which adopted the principal of genetic learning. The results were compared with other fusion models including concatenated feature fusion, equal weighted fusion and BPN weighted decision level fusion schemes and showed an improvement in comparison to all aforementioned fusion strategies and other uni-modal schemes.

Fadil *et al* [59], proposed a multi-modal approach that used audio-visual information to extract features and deep belief networks to discriminate different emotion states. With the help of voice activity detector, segments of voices were detected. Prosodic features, Spectral characteristics and Cepstral features were extracted from the audio data. Prosodic features such as standard deviation and the mean value of fundamental frequency and energy were obtained. Spectral characteristics such as mean logarithmic spectrum were calculated. Furthermore, well-known and highly used cepstral features known as Mel-Frequency Cepstral Coefficients were calculated and employed in this research. Furthermore, discrete Fourier coefficients along with PCA projections were used to find the visual feature set. These discriminative audio and visual features withdrawn from emotional video clips were fed to auto encoder. Then the mean number was also found for those frames and then that was used to all the videos uniformly. The videos were then classified into six different emotions. This classification was done with the help of deep networks. Deep networks are such an architecture that comprises of different layers. These layers are meant for capturing the high order correlations between features. In order to evaluate the proposed technique, an experiment was performed using RML Emotion Database. The results obtained from this experiment depicted that this approach was quite suitable for determining emotions using multi-modal analysis.

Gharavian *et al*. [60] proposed a fuzzy ARTMAP Neural Network model (FAMNN) in his recognition experiment. Audio features such as MFCC, pitch, energy, formants and visual features including marker locations on face, were extracted and then dimensionally reduced using PCA feature reduction algorithm. Features were further reduced by using a selection procedure known as Fast Correlation Base Filter (FCBF) method. FAMNN was regularly used at various steps of the experiment and finally Particle Swarm Optimization technique is used to estimate the best values for the choice parameter (α), vigilance parameter (ρ) and the learning rate (β) of the

FAMNN. Experiments were conducted on SAVEE audio-visual data set and the results showed that the fusions at feature and score level enhance the outcome of the single modality approach. Also, PSO improved the recognition capability of the system such that PSO optimized FAMNN outperformed the audio system by 57% and the visual approach by 4.5%.

Zhang *et al.* [61] proposed deep belief networks (DBN) model for emotion classification. Two-stage learning strategy was employed in which one stage deals with training the audio-visual network and the other one is the fusion network. The first stage involves fine tuning of the audio and visual networks through pre-trained AlexNet and C3D-Sports-1M respectively. The second stage involves training of DBN fusion network using target emotional database. 1-D audio signal is transformed into three channels of Mel-spectrogram of 64 x 64 x 3 dimensions and is passed on as input to the CNN. Size of this Mel-spectrogram can be changed conveniently in accordance with the input of the existing CNN models pre-trained on image datasets. Visual features are extracted from video segments using 3D-CNN. This process involves division of videos into segments having 16 frames, followed by detection of face in each frame of the segment by using a real-time face detector provided by Viola and Jones. After detecting face, eye distance is calculated for each frame and is normalized to a fixed value of 55 pixels. RGB image of 150 x 110 x 3 is separated from each frame base on the set value of eye distance. But the image is then resized to 227 x 227 x 3 in order to achieve fine tuning when passed on as an input to the pre-trained 3D-CNN model. The DBN model used for network fusion is constructed by two RBM's stacked over each other and is trained in two steps. First step involves using greedy layer-wise training algorithm in the bottom up manner for unsupervised pre-training. After pre-training, RBMs are initialized and fine-tuned to optimize the network parameters. Audio and Video network parameters are fixed for second stage training while fusion network parameters are updated to provide accurate prediction values. After the fusion network is trained, feature representations are obtained for each audio-visual segment. Since each segment varies in length, average pooling is applied to all segment features to obtain uniform global feature representations. Different fusion strategies were studied which included feature level, score level and decision level fusion and their impact on classification rate was compared to the proposed DBN network based fusion. RML, eNTERFACE and BAUM-I emotional datasets were selected for this experiment and linear SVM was used as the classifier. The results obtained proved highly in favor of DBN fusion network compared to the other fusion based strategies.

Noroozi *et al.* [62] developed a multi-modal emotion recognition approach in which audio data was split from the video clips and was processed to provide features such as MFCC, their deltas, pitch, intensity, percentile, formants and their bandwidth, Filter Bank Energies and other statistics. Moreover, key frames from a video clip were identified and selected using k-means clustering. Facial landmarks and their geometric relations were used to calculate these key frames and the geometric features obtained from visual content were passed onto a multi-class SVM classifier. Similar key frames were also used to train a convolutional neural network (CNN). And the resulting confidence values from all three classifiers (audio, video and CNN) were passed as input to another stage of SVM or RF classifiers to get the final emotion recognition rates. Experiments were performed on eNTERFACE, RML and SAVEE database and this recognition model generated one of the best accuracies ever achieved on all three employed datasets.

A novel Bayesian nonparametric multimodal data modeling framework has been proposed by

Xue *et al.* [63] to learn the emotions from video and audio, where the adopted image data are deep features extracted from key frames of video via convolutional neural networks (CNNs), and the adopted audio data are extracted by Mel-frequency cepstral coefficient (MFCC) features

A hybrid CNN-RNN architecture for emotion transaction analysis has been proposed by Ronghe *et al.* [64] that can recognize the emotion in a frame in video and predict its appropriate reaction.

Cheng *et al.* [65]presented emotion detection from facial recognition for wireless applications where a robust emotion recognition achieved from low bit rate video. While video frames are down sampled at the encoder side, the decoder is embedded with a deep network model for joint super-resolution (SR) and recognition.

Another model which is presented by Barros *et al.* [66] in 2016 used RBM (Reward Based Mechanism) to associate the different states of expressions to the already perceived emotions by training two parallel network braches using the MLP (Multi-Layer Perceptron). A combination of CNN and SOM (Self Organizing Map) is used in perceptron engine for identification and reorganization of emotion.

Liu et al. [67] proposed a Deep Complete Canonical correlation analysis (DCCCA) in which they not only linear correlation is used but also complex non linear correlation is also used. According to Liu Complete CCA uses only linear projections where as in real case mostly non linear

correlation exist. They have employed two Deep Neural networks to map data into higher dimension and then used CCA to maximize their correlation.

As it can be inferred from the literature, facial emotion detection is a complex task. Its complexity has led to several approaches that has something in common: the need for feature extraction, and then applying a classifier on top. All these techniques have gradually tried to improve the performance measures for emotion recognition by either focusing on a single modality or using multi-modal approaches but there is still need for improvement as these systems are either computationally complex or do not achieve the desired recognition accuracy. The proposed methodology in this project uses convolutional neural networks to avoid the feature extraction stage; since the network would be able to detect features by itself. Furthermore, learning the sequence over each time step from previous frames is also implemented using Long Short Term Memory (LSTM) units.

Table 3-1 Summary of recent emotion recognition systems

Author	Dataset	Modality	Extracted Features	Classification Strategy
Paleari *et.al*. [57]	eNTERFACE	Bi-modal	A: MFCC, LPCC Formants, Pitch, Energy V: Distance features	AV: Neural Nets
Zhalehpour *et al.* [44]	eNTERFACE BAUM-1	Bi-modal	A= MFCC, RASTA-PLP V=LPQ, POEM	A: SVM(rbf) V: SVM(linear)
Huang *et al*. [58]	eNTERFACE	Bi-modal	A: Pitch, energy, speed, MFCC V: TFP, LPTP	AV: Neural Net (Genetic learning)
Shan *et al*. [47]	JAFEE MMI	Uni-modal	V: Boosted LBP	V: SVM (RBF)
Fadil *et al*. [59]	RML	Bi-modal	A: MFCC, log spectrum, std, mean F0 V: FFT,PCA	AV: Deep Networks Multi-layer Perceptron
Shaukat *et al*. [49]	Berlin, BabyEars GEES DES	Uni-modal	A: Energy, frequency, duration, MFCC	A: Linear Combination of SVMs
Wang *et al*. [38]	RML	Bi-modal	A: MFCC, Pitch and power V: Gabor Filters with PCA	AV: HMM(KPCA,KLDA input)
Dhall *et al.* [48]	SSPNET GEMEP-FERA	Uni-modal	V: PHOG with LPQ	V: SVM, Largest Margin
Seng *et al*. [45]	RML	Bi-modal	A: Pitch, energy, ZCR, TEO, MFCC V:BDPCA, LSLDA	AV: RBF classifiers
Shaukat *et al*. [50]	JAFEE	Uni-modal	V: Scale Invariant Feature Transform (SIFT), Gabor Wavelets, DCT	V: SVM (RBF)
Cid *et al*. [46]	SAVEE	Bi-modal	A: Pitch, energy, speech rate	A: DBN V: DBN

			V: Edge based features	AV: DBN
Haq *et al.* [39]	SAVEE	Bi-modal	A: MFCC, pitch, energy, duration V: 2D marker Coordinates	AV: Gaussian Classifier (PCA input)
Gharavian *et al.* [60]	SAVEE	Bi-modal	A: MFCC, pitch, energy, formants V: Facial marker locations	A: FAMNN V: FAMNN AV: PSO optimized FAMNN model
Khan *et al.* [51]	Cohn-Kanade MMI	Uni-modal	V: PLBP (Pyramids of LBP)	V: SVM, 2NN, RF, DT, NB
Zhang *et al.* [61]	eNTERFACE RML BAUM-1	Bi-modal	A: Anet (AlexNet) V: 3D-CNN (Vnet)	A: Anet V: Vnet AV: DBN
Rashid *et al.* [40]	eNTERFACE	Bi-modal	A: MFCC, prosodic V: Spatio-temporal	A: SVM V: SVM AV: BSR
Noroozi *et al.* [62]	eNTERFACE RML SAVEE	Bimodal	A: MFCC, deltas, pitch, intensity, percentile, formants, bandwidth, FBE V: Geometric features, 3D-CNN	A: SVM, RF V: SVM, RF AV: SVM, RF (with or without PCA)
Barros *et al.* [66]	FABO SAVEE EmotiW	Bimodal	A: MFCC, power spectrograms V: Geometric features, 3D-CNN	AV: CCCNN,SOM
Liu *et al.* [67]	RAVDESS MNIST	Bimodal	A: MFCC, LPCC V:eigen values, CNN	AV: Deep Complete-CCA
Zhang *et al.* [54]	RAVDESS UMSSED	Bimodal	A: MFCC, Energy, Spectral, Voicing, Rasta V:Upper face, lower face	AV: Simple model, St-heir model, Mt-heir model

Chapter 4: METHODOLOGY

This chapter presents proposed methodology of this research in detail. It comprises of neural network models which extract unique features and does a very good classification among seven emotions. First we will discuss every layer of our proposed model one by one followed by classification technique used in our method. Summary of our proposed emotion detection system is given in figure 1.1

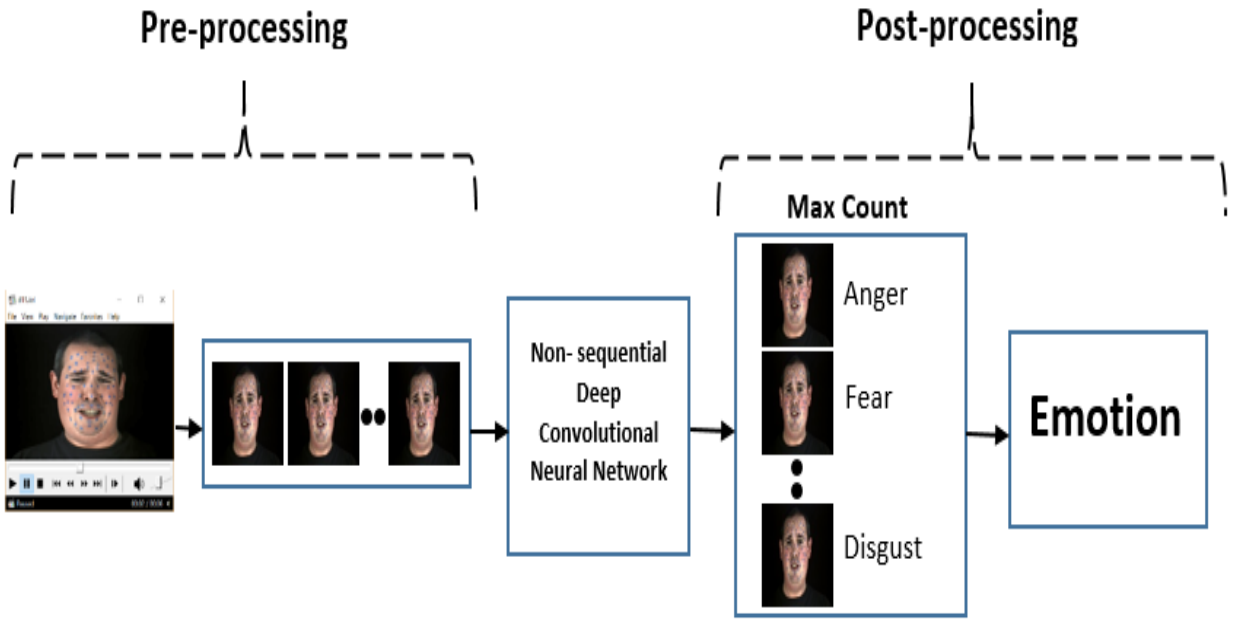

Figure 4-1 Proposed Emotion Detection System

The suggested approach describes the degree to which non-sequential convolutional neural networks is successful in achieving desired results in emotion detection in videos. The proposed system consists of three levels video segmentation, CNN and post processing. Videos that have been processed before are then moved to convolutional network to foretell or estimate the desired outcome. After that emotions of each frame are individually counted to assign overall emotion to the video.

1) **Pre- Processing**
2) **Non Sequential Deep Convolutional Neural Network.**
3) **Post- Processing**

4.1 Pre Processing

Directing towards its target, basically, video segmentation focuses on splitting the particular order in which image is designed into scenes and shots. A set is actually a consecutive framework settlement without any intervening period whereas scene can be described as fundamental scenario based or recital portion of the video. In video segmentation, each video in our dataset consist of 60 frames per second.

4.2 Non-sequential deep convolutional neural network

In proposed Non-sequential deep neural network there are 4 parallel layers that work separately possessing uncommon set of filters. Comprising of 32 9x9 filter, first parallel layer outputs 32x128x128 image straight away and hence covering more neighbor pixel data. Second parallel layer comprise of 32 7x7 filters and 32 3x3 filter and third layer of filter 32x5x5 followed by another 32x5x5 filter as show in fig 4.1 below.

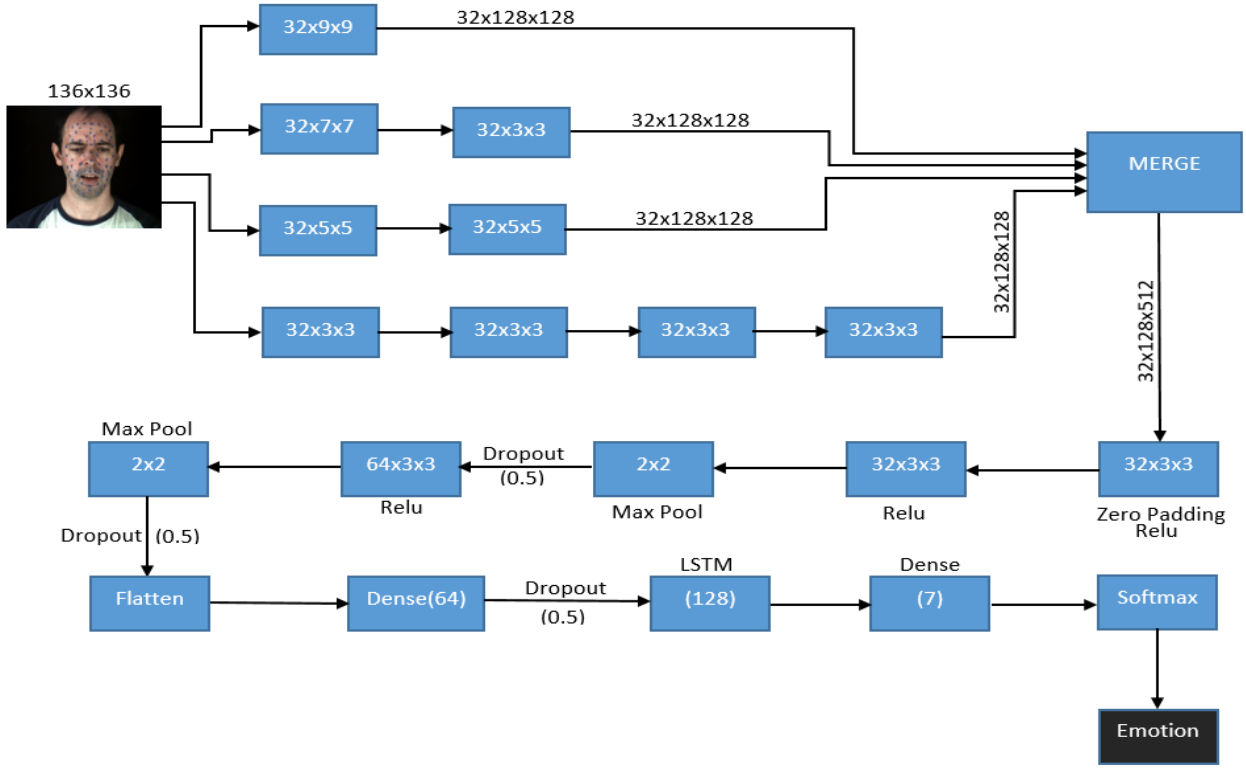

Figure 4-2 Proposed Non-Sequential Deep Convolutional Neural Network

Whereas, fourth layer consists of four nodes and each node consisting of size 32x3x3 producing same output image of size 32x128x128 for each parallel layer. The input image is same for all parallel layers of size 136x136. Now the output from all the four parallel layers is combined in to one and then fed into another Convolutional Neural Network. Because of increase in the number of parameters, dropout layer is used more than one time to reduce the effect of overfitting. First fully connected layers consists of 251968 features. Then its output is fed to LSTM layer which outputs feature vector of 128 in size. This feature vector is fed onto next fully connected layer. The feature maps of intermediate layers are given below.

Figure 4-3 Feature Map after merging four channels.

Figure 4-4 Feature Map after first Max pool Layer

Figure 4-5 Feature Map after Second Max pool Layer

Table 4-1 Complete Specification of each layer of Proposed Model

LAYER (TYPE)	**FILTER SHAPE**	**OUTPUT SHAPE**	**PARAM #**	**CONNECTED TO**
Input1 (input layer)		1x136x136	0	
Input2 (input layer)		1x136x136	0	
Input3 (input layer)		1x136x136	0	
Input4 (input layer)		1x136x136	0	
conv2d_1(Convolution)	32x9x9	32x128x128	2624	Input1
conv2d_2(Convolution)	32x7x7	32x130x130	1600	Input2
conv2d_3(Convolution)	32x3x3	32x128x128	9248	conv2d_2
conv2d_4(Convolution)	32x5x5	32x132x132	832	Input3
conv2d_5(Convolution)	32x5x5	32x128x128	25632	conv2d_4
conv2d_6(Convolution)	32x3x3	32x 134x134	320	Input4
conv2d_7(Convolution)	32x3x3	32x132x132	9248	conv2d_6
conv2d_8(Convolution)	32x3x3	32x130x130	9248	conv2d_7
conv2d_9(Convolution)	32x3x3	32x128x128	9248	conv2d_8
merge_1 (Merge)		32x128x512	0	conv2d_1, conv2d_3, conv2d_5, conv2d_9
zero_padding_1		32x132x516	0	Merge_1
conv2d_10(Convolution)	32x3x3	32x130x514	9248	Zero_padding_1
conv2d_11(Convolution)	32x3x3	32x128x512	9248	conv2d_10
Max_pooling_1		32x64x256	0	conv2d_11
Dropout_1		32x64x256	0	Max_pooling_1
conv2d_12(Convolution)	64x3x3	64x62x254	18496	Dropout_1
Max_poolong_2		64x31x127	0	conv2d_12
Dropout_2		64x31x127	0	Max_pooiling_2
Flatten_1		251968	0	Dropout_2
Dense_1		64	16126016	Flatten_1

cu_dnnlstm_1		1x128	99328	Dense_1
flatten_2		128		cu_dnnlstm_1
Dense_2		7	455	flatten_2
SoftMax		7	0	Dense_2

Several number of features are chosen for good quality of emotion detection. The Total Parameters in proposed network are 16,231,463. Specification of complete architecture is shown in Table 4.1 given below.

Total parameters: 16,331,239

Trainable parameters: 16,331,239

Non-trainable parameters: 0

4.2.1 Long Short-Term Memory (LSTM):

Long Short Term Memory networks – usually just called "LSTMs" are a special kind of RNN, capable of learning long-term dependencies. LSTMs are explicitly designed to avoid the long-term dependency problem. Remembering information for long periods of time is practically their default behavior, not something they struggle to learn!

The first step in our LSTM is to decide what information we're going to throw away from the cell state. This decision is made by a sigmoid layer called the "forget gate layer." It looks at h_{t-1} and x_t, and outputs a number between 0 and 1 for each number in the cell state C_{t-1}. A 1 represents "completely keep this" while a 0 represents "completely get rid of this."

The next step is to decide what new information we're going to store in the cell state. This has two parts. First, a sigmoid layer called the "input gate layer" decides which values we'll update. Next, a tanh layer creates a vector of new candidate values, $C_{\sim t}$, that could be added to the state. In the next step, we'll combine these two to create an update to the state.

It's now time to update the old cell state, C_{t-1}, into the new cell state C_t. The previous steps already decided what to do, we just need to actually do it.

We multiply the old state by f_t, forgetting the things we decided to forget earlier. Then we add $i_t * C_{\sim t}$. This is the new candidate values, scaled by how much we decided to update each state value.

Finally, we need to decide what we're going to output. This output will be based on our cell state, but will be a filtered version. First, we run a sigmoid layer which decides what parts of the cell state we're going to output. Then, we put the cell state through tanh (to push the values to be between −1 and 1) and multiply it by the output of the sigmoid gate, so that we only output the parts we decided to.

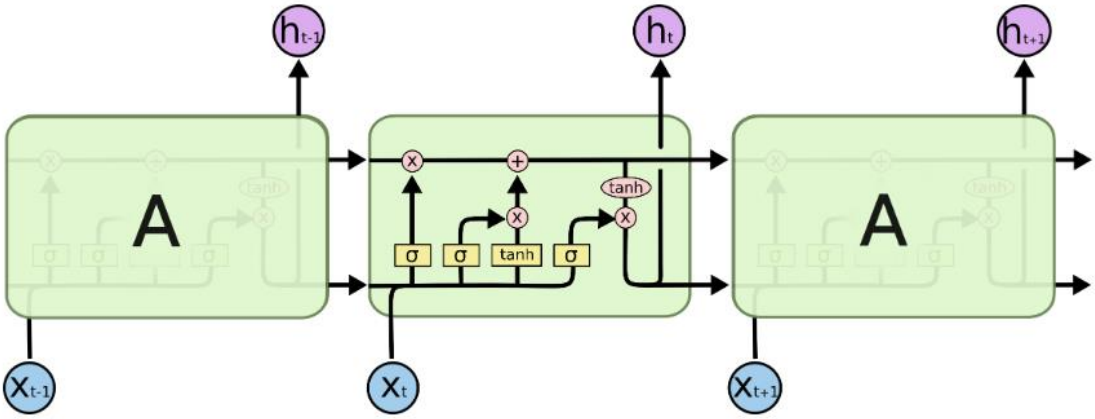

Figure 4-6 The repeating module in an LSTM contains four interacting layers [71]

4.2.2 Neural Network Parameters

The proposed architecture is implemented using Keras library in python. These libraries provide numerous methods and models for implementing CNN in Tensor-flow as backend. CNN network in Keras can be implemented in both CPU and GPU. We have used GPU GT 940m for training and testing.

In network those parameters which can be tuned are tuned for the best possible results. RELU and Soft-max are used as neuronal activation parameters, Stochastic gradient descent (SGD) as optimization parameter and Glorot and uniform as initialization parameter for different layers of network. Batch size is fixed to 10 and number of epoch is 2. Weights of the kernels in our proposed network are initialized randomly using normal mode initialization. In all the layers, Bias value is fixed to be zero. Stride of 2 is used in max-pooling and stride of 1is used in convolution layers.

4.2.3 Neuronal Activation

The purpose of the neuronal activation function is to control the output of the neurons in neural network. There are many types of neuronal function e.g maxout, RELU, tangent etc. The we have used in our proposed network is RELU which is a nonlinear function. The output of RELU function are only two values, zero and a positive value. This function finds the max value between zero and the given output.

$$\mathbf{F(x) = Max\ (0\ ,\ x).}$$

4.2.4 Regularizer

In neural network when system neuron learns, they try tune their weights according to input data. Neighbor neurons starts to depend on each other for specific data and if this goes on system become over fitted for specific data which results in a fragile model. To address this problem Regularizer are used. Their purpose is to reduce the overfitting in model. The Dropout Regularizer [20] is used for our proposed network. This Regularizer selects neurons randomly and drop them. This way dependency of neurons on each other is minimized. Remaining neurons will have to update their weights independently. In our proposed neural network high dropout value of 0.5 is used throughout the network to reduce the chance of overfitting.

4.2.5 Optimizer

Purpose of the optimizer to distribute weights throughout network by using the loss function at the output of network. Stochastic gradient descent (SGD) [21] is used as optimizer and Categorical Cross entropy as loss function in our network

4.2.6 Rectified Linear Unit

An ANN architecture necessarily has the activation function of a unit (neuron). Since the early days of ANN, researchers have been using different functions. But a good error approximation is not possible due to step function's binary nature.

The sigmoid functions were utilized to overcome this challenge. They were used to provide promising results for small networks. However, it was not appropriate to scale sigmoid function

on large networks [20]. As it could lead to huge numbers, the cost for computations was too high [21]. The gradient vanishing problem was among other significant issues with sigmoid function. The prevention of learning occurs due to high value of gradient value [22][23].

Compared to previous common activation functions, the rectified linear unit function (ReLU) provided benefits in this situation. It did not suffer from the gradient vanishing problem and provided a good error approximation but it was used to cost less. The figure 2.4 displays ReLU and it is also mentioned below:

$$\mathbf{f(x) = max(0, x)}$$

The research conducted by Krizhevsky et al. [20] elaborated that the use of ReLU lowered the epochs's number required to converge when using Stochastic Gradient Descent by a factor of 6. There is a major drawback of ReLU's use which is the weakness when input distribution is below zero. It is due to the reason that neuron will not activate by any data point. Use of GPU. There are practical reasons to train deep networks with the help of a GPU.

To reduce training time compared to CPU training is the main reason [24]. Though the speed depends on the network topology, the use of GPU provides 10 times faster speed [25].

The way of processing different tasks is what differentiates GPU from CPU. A few cores are used in CPUs to execute sequential serial processing. However, GPU represents a mighty parallel architecture. To manage several tasks at the same time, thousands of tiny cores work in harmony in this architecture.

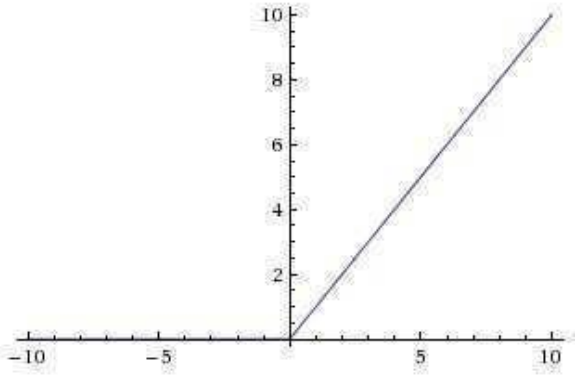

Figure 4-7: Rectified Linear Unit (ReLU) [74]

From the above discussion, it is clear that the Deep Learning works better with GPU rather than CPU. However, most of projects still make use of CPU due to different reasons

4.3 Post Processing

After passing from neural network, each frame is assigned an emotion. In this step those all those individual emotions are counted and then stored in a form of array. Array contains counts against all seven emotions. The emotion consisting of maximum count is assigned to the video.

Chapter 5: EXPERIMENTAL RESULTS

In this chapter, we present the outcome of our experiments conducted in our audio-visual emotion recognition system. We employed two datasets namely RAVDESS and SAVEE.

5.1 Databases Explanation:

Several recent visual emotion recognition methods have employed posed databases. We have further tried to contribute in that aspect by using three posed datasets in our research. The chosen databases include RAVDESS and SAVEE 14 and there details are mentioned in the following section.

5.1.1 Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS [68]):

Many emotional datasets are available, out of which we selected RAVDESS to carry out our research. RAVDESS dataset contains seven emotional states (anger, disgust, fear, happiness, sadness, surprise, calm) along with neutral, acted by 24 different subjects out of which 50% were male and 50% were female. Every emotion consists of two intensities along with neutral expression. Three modalities are available for each actor, audio-only (AO), video-only (VO), and audio-visual (AV). A total of 1248 video-only (VO) samples are present in this dataset. The language of dataset is North American English. Each video comprises of 30 frames per second with 1920x1080 resolution. White background was used to ensure easy and accurate face detection.

5.1.2 Surrey Audio-visual Expressed Emotion Database (SAVEE [69]):

The final database that we have employed in this research is SAVEE database and contains recordings of four subjects (all male) having an age of about 27 to 31 years 14 . The speakers are British and a total of 480 videos are recorded in English language. Six basic emotion states namely,

anger, disgust, fear, happiness, sadness and surprise are recorded along with a neutral state. The neutral state is expressed in 120 different samples whereas 60 different samples are recorded for each of the remaining emotional states. Emotional and text prompts were displayed on a monitor in front of the actors during the recording process. These prompts included a video along with three images and for every emotion; the text prompts were divided into three groups in order to avoid fatigue. The recording was checked in order to guarantee that the emotions displayed were accurate. For that purpose, the samples were evaluated by 10 subjects and the acting performance was measured under audio, visual and audiovisual conditions separately.

Figure 5-2 Sample images from RAVDESS dataset [15]

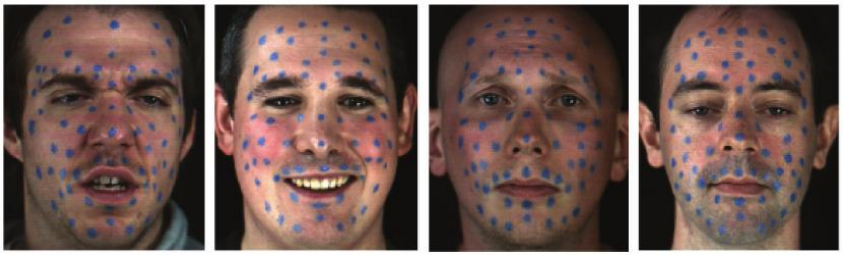

Figure 5-1 Sample images from SAVEE dataset [14]

5.2 Evaluation for Video Modality:

After the detailed description of the databases in the previous section, we now move ahead and explain how the experiment was performed and the results were evaluated.

5.2.1 Cross Validation:

Emotion detection results are evaluated using K-fold Cross Validation. Evaluation is done on all seven classes i.e., angry, disgust, fear, happy, sad, surprise and neutral. Two values for k folding are used for SAVEE dataset.. 3 fold and 15 fold.

5.2.2 Results of Proposed Architecture without LSTM:

In 3 fold, the dataset is divided into 3 parts. 33.3 % is used for testing and 66.6% for training. Similarly, in 15 fold, dataset is divided in 15 parts. 6.66% is used for testing and 93.33% for training. The proposed architecture is trained and tested in HP Core i5 system with 16Gb RAM and 2GB dedicated video ram of NVIDIA Geforce 940m. Average accuracy for 3-Fold is 98.8% and Average accuracy for 15-Fold is also 98.8%.

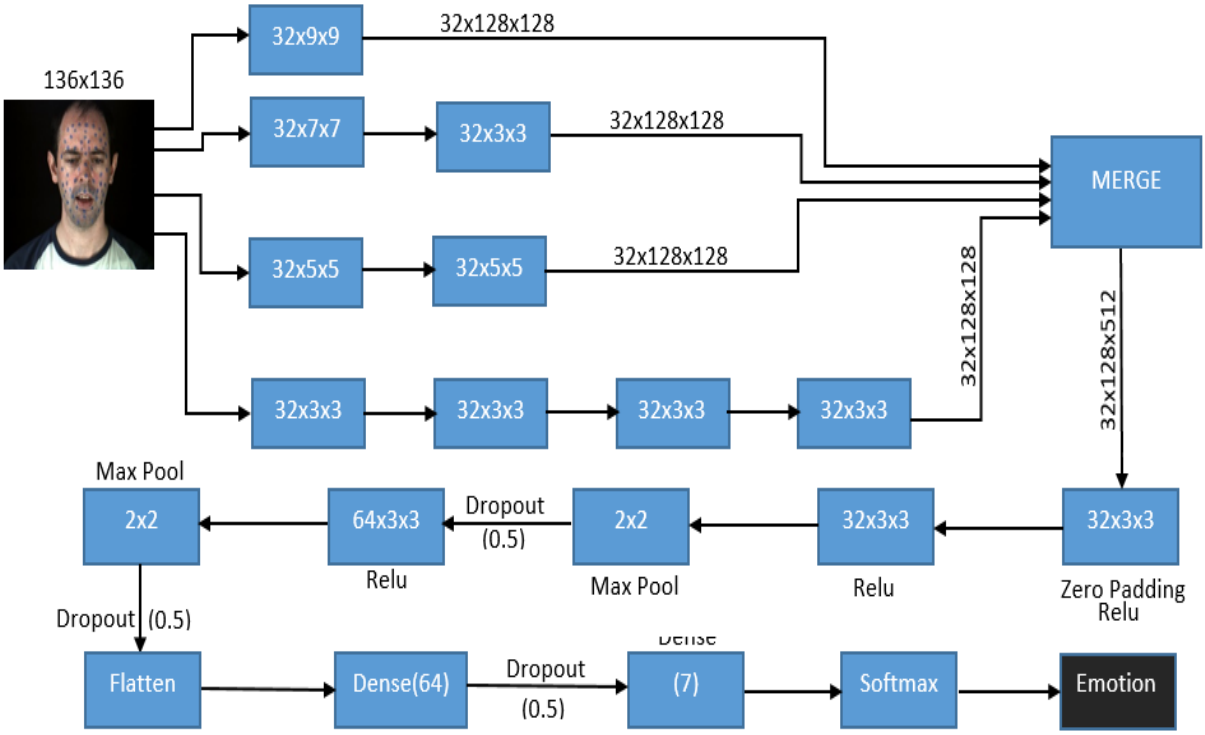

Figure 5-3 Proposed Non-Sequential Deep Convolutional Neural Network without LSTM

Table 5-1: Fifteen fold Cross Validation Result for SAVEE

K-Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
1	100	100	100	100	100	100	100	100%
2	100	100	100	100	100	100	100	100%
3	100	100	100	100	100	100	100	100%
4	100	100	100	100	100	100	100	100%
5	100	100	75	100	100	75	100	92.85%
6	100	100	100	100	100	100	100	100%
7	100	100	100	100	100	100	100	100%
8	100	100	100	100	100	100	100	100%
9	100	100	100	100	75	100	100	96.42%
10	100	75	100	100	100	100	100	96.42%
11	100	100	75	100	100	100	100	96.42%
12	100	100	100	100	100	100	100	100%
13	100	100	100	100	100	100	100	100%
14	100	100	100	100	100	100	100	100%
15	100	100	100	100	100	100	100	100%
Average Accuracy								**98.80%**

Table 5-2: Three Fold Cross Validation Result for SAVEE (1st)

1st Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	19	1	0	0	0	0	0	95%
Disgust	0	19	0	0	0	0	1	95%
Fear	0	0	20	0	0	0	0	100%
Happy	0	1	0	19	0	0	0	95%
Sad	0	0	0	0	20	0	0	100%
Surprise	0	0	0	0	0	20	0	100%
Neutral	0	0	0	0	0	0	40	100%
Average Accuracy								**97.85%**

Table 5-3: Three Fold Cross Validation Result for SAVEE (2nd)

2nd Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	20	0	0	0	0	0	0	100%
Disgust	0	20	0	0	0	0	0	100%
Fear	0	0	19	0	1	0	0	95%
Happy	0	0	0	20	0	0	0	100%
Sad	0	0	0	0	20	0	0	100%
Surprise	0	0	0	0	0	20	0	100%
Neutral	0	0	0	0	0	0	40	100%
Average Accuracy								**99.28%**

Table 5-4: Three Fold Cross Validation Result for SAVEE (3rd)

3rd Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	20	0	0	0	0	0	0	100%
Disgust	0	20	0	0	0	0	0	100%
Fear	0	0	19	0	0	1	0	85%
Happy	0	0	0	20	0	0	0	100%
Sad	0	0	0	0	20	0	0	100%
Surprise	0	0	0	0	0	20	0	100%
Neutral	0	0	0	0	0	0	40	100%
Average Accuracy								**99.28%**

Table 5-5 Four Fold Cross Validation Result for RAVDESS (1st)

1st fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	37	3	2	3	2	1	0	77.08%
Disgust	1	42	1	1	0	3	0	87.5%
Fear	5	1	41	0	1	0	0	85.41%
Happy	0	0	0	47	0	1	0	97.91%
Sad	2	2	0	1	42	1	0	87.5%
Surprise	1	0	1	3	1	42	0	87.5%
Neutral	0	0	0	0	0	0	24	100%
Average Accuracy								**88.98%**

Table 5-6 Four Fold Cross Validation Result for RAVDESS (2nd)

2nd fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	38	3	2	1	1	3	0	79.16%
Disgust	1	44	1	0	0	2	0	91.66%
Fear	3	2	35	0	3	3	2	72.91%
Happy	2	0	0	45	0	1	0	93.75%
Sad	3	2	2	0	40	1	0	83.33%
Surprise	1	1	1	2	1	42	0	87.5%
Neutral	0	0	0	0	1	0	23	95.83%
Average Accuracy								**86.30%**

Table 5-7: Four Fold Cross Validation Result for RAVDESS (3rd)

3rd fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	43	0	3	0	1	1	0	89.58%
Disgust	1	41	5	0	1	0	0	85.41%
Fear	1	1	44	0	1	1	0	91.66%
Happy	0	4	0	42	0	2	0	87.5%
Sad	2	2	2	0	42	0	0	87.5%
Surprise	1	1	4	4	1	37	0	77.08%
Neutral	0	0	0	1	0	0	23	95.83%
Average Accuracy								**87.79%**

Table 5-8: Four Fold Cross Validation Result for RAVDESS (4^{th})

4^{th} fold	**Anger**	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**	**Recognition Rate**
Anger	40	0	6	0	1	1	0	83.33%
Disgust	2	43	3	0	0	0	0	89.58%
Fear	1	1	43	2	0	1	0	89.58%
Happy	1	1	0	43	0	3	0	89.58%
Sad	4	3	3	0	35	3	0	72.91%
Surprise	2	0	2	2	1	40	1	83.33%
Neutral	0	0	0	0	0	1	23	95.83%
Average Accuracy								**86.30%**

Table 5-9 Confusion Matrix of SAVEE without LSTM

	Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**	**Recognition Rate**
Anger	59	1	0	0	0	0	0	98.33%
Disgust	0	59	0	0	0	0	1	98.33%
Fear	0	0	58	0	1	1	0	96.66%
Happy	0	1	0	59	0	0	0	98.33%
Sad	0	0	0	0	60	0	0	100%
Surprise	0	0	0	0	0	60	0	100%
Neutral	0	0	0	0	0	0	120	100%
Average Accuracy								**98.80%**

Table 5-10 Confusion Matrix of RAVDESS without LSTM

	Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**	**Recognition Rate**
Anger	158	6	13	4	5	5	0	82.29%
Disgust	5	167	10	1	1	5	0	86.97%
Fear	10	5	163	2	5	5	2	84.89%
Happy	3	5	0	177	0	7	0	92.18%
Sad	11	9	7	1	159	5	0	82.81%
Surprise	5	2	8	11	4	161	1	83.85%
Neutral	0	0	0	1	1	1	93	96.87%
Average Accuracy								**87.12%**

As we can see from confusion matrix of SAVEE dataset, very high accuracy of 98.8% is achieved which is better than all accuracies achieved using neural networks up till now. Most misclassified emotion in SAVEE dataset is fear which is misclassified for sad and surprise emotion. Accuracy of 87.12% is achieved for RAVDESS dataset which is also highest accuracy achieved using neural networks. The most misclassified emotion in RAVDESS is anger which is misclassified as fear mostly. Neutral emotion is best classified in both datasets. By looking at the individual emotions predictions of each frame it is found that there was no dependency among frames of videos. Every frame is independent and the final emotion assigned to video is not assigned with strong confidence. In few cases, emotions were correctly predicted by advantage of only few frames.

5.2.3 Results of Proposed Architecture with LSTM:

By adding LSTM layer in our model, time factor is included in classification process. In previous model out of each frame of video was independent of each other for example in happy video every frame is treated differently and doesn't depend on each other output which seems logically wrong. In video every consecutive frame shows relationship with previous frames. This has been achieved using LSTM layer in which up to N point in time, every output depends on its all previous outputs.

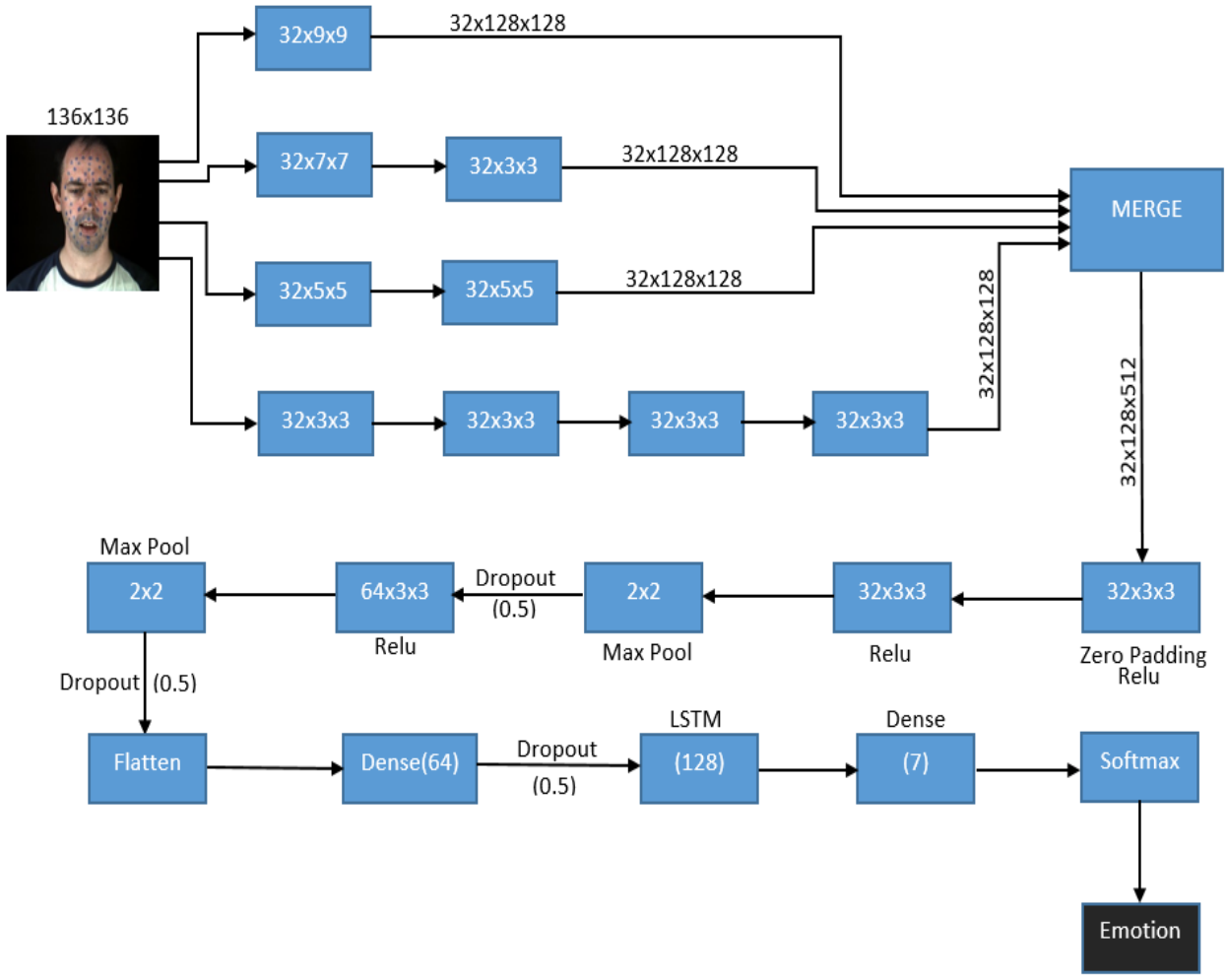

Figure 5-4 Proposed Non-Sequential Deep Convolutional Neural Network with LSTM

Table 5-11: Three Fold Cross Validation Result for SAVEE with LSTM (1st)

1st Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	20	0	0	0	0	0	0	100%
Disgust	0	20	0	0	0	0	0	100%
Fear	0	0	20	0	0	0	0	100%
Happy	0	0	0	20	0	0	0	100%
Sad	0	0	0	0	20	0	0	100%
Surprise	0	0	0	0	0	20	0	100%
Neutral	0	0	0	0	0	0	40	100%
Average Accuracy								**100%**

Table 5-12: Three Fold Cross Validation Result for SAVEE with LSTM (2nd)

2nd Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	20	0	0	0	0	0	0	100%
Disgust	0	20	0	0	0	0	0	100%
Fear	0	0	19	0	1	0	0	95%
Happy	0	0	0	20	0	0	0	100%
Sad	0	0	0	0	20	0	0	100%
Surprise	0	0	0	0	0	20	0	100%
Neutral	0	0	0	1	0	0	39	97.5%
Average Accuracy								**98.92%**

Table 5-13: Three Fold Cross Validation Result for SAVEE with LSTM (3rd)

3rd Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	20	0	0	0	0	0	0	100%
Disgust	0	20	0	0	0	0	0	100%
Fear	0	0	20	0	0	0	0	100%
Happy	0	0	0	20	0	0	0	100%
Sad	0	0	0	0	20	0	0	100%
Surprise	0	0	0	0	0	20	0	100%
Neutral	0	0	0	0	0	0	40	100%
Average Accuracy								**100%**

Table 5-14: Four Fold Cross Validation Result for RAVDESS (1st)

1st Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	37	3	3	0	2	3	0	77.08%
Disgust	2	42	3	0	0	1	0	87.5%
Fear	5	2	38	1	1	0	1	79.16%
Happy	1	0	0	47	0	0	0	97.91%
Sad	4	1	0	1	41	1	0	85.41%
Surprise	1	1	2	2	1	40	1	83.33%
Neutral	0	0	0	0	0	1	23	95.83%
Average Accuracy								**86.60%**

Table 5-15: Four Fold Cross Validation Result for RAVDESS (2nd)

2nd Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	36	0	2	3	5	2	0	75%
Disgust	0	42	5	0	1	0	0	87.5%
Fear	0	2	43	0	2	1	0	89.58%
Happy	1	0	0	45	2	0	0	93.75%
Sad	7	2	1	0	37	0	1	77.08%
Surprise	1	1	3	0	0	42	1	87.5%
Neutral	0	0	0	0	0	0	24	100%
Average Accuracy								**87.20%**

Table 5-16: Four Fold Cross Validation Result for RAVDESS (3rd)

3rd Fold	Anger	Disgust	Fear	Happy	Sad	Surprise	Neutral	Recognition Rate
Anger	42	0	3	0	2	1	0	87.5%
Disgust	1	42	5	0	0	0	0	87.5%
Fear	0	1	44	1	1	1	0	91.66%
Happy	0	0	1	47	0	0	0	97.91%
Sad	1	4	3	0	40	0	0	83.33%
Surprise	2	3	5	3	1	34	1	70.83%
Neutral	0	0	0	0	0	0	24	100%
Average Accuracy								**88.39%**

Table 5-17: Four Fold Cross Validation Result for RAVDESS (4^{th})

4^{th} Fold	**Anger**	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**	**Recognition Rate**
Anger	41	0	5	1	1	0	0	85.41%
Disgust	1	42	4	1	0	0	0	87.5%
Fear	3	0	43	2	0	0	0	89.58%
Happy	0	0	1	45	0	2	0	93.75%
Sad	5	3	0	0	39	1	0	81.25%
Surprise	1	0	4	2	1	39	1	81.25%
Neutral	0	1	0	0	0	0	23	95.83%
Average Accuracy								**87.79%**

Table 5-18: Confusion Matrix of SAVEE with LSTM

	Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**	**Recognition Rate**
Anger	60	0	0	0	0	0	0	100%
Disgust	0	60	0	0	0	0	0	100%
Fear	0	0	59	0	1	0	0	98.33%
Happy	0	0	0	60	0	0	0	100%
Sad	0	0	0	0	60	0	0	100%
Surprise	0	0	0	0	0	60	0	100%
Neutral	0	0	0	1	0	0	119	99.16%
Average Accuracy								**99.64%**

Table 5-19: Confusion Matrix of RAVDESS with LSTM

	Anger	**Disgust**	**Fear**	**Happy**	**Sad**	**Surprise**	**Neutral**	**Recognition Rate**
Anger	156	3	13	4	10	6	0	81.25%
Disgust	4	168	17	1	1	1	0	87.5%
Fear	8	5	168	4	4	2	1	87.5%
Happy	2	0	2	184	2	2	0	95.83%
Sad	17	10	4	1	157	2	1	81.77%
Surprise	5	5	14	7	3	155	4	80.72%
Neutral	0	1	0	0	0	1	94	97.91%
Average Accuracy								**87.49%**

Accuracy achieved on SAVEE dataset using LSTM layer is 99.64% which shows improvement of 0.84% over previously proposed model. Only one video from fear and neutral emotion is misclassified. Accuracy of RAVDESS dataset is 87.49% which is 0.37% better. It can be seen that, there is not much difference between accuracies both proposed models, but if we carefully look at results of individual frames it can be seen that using LSTM model, every emotion assigned to videos is assigned with very strong confidence. All those cases in which previous model predicted correct emotion with confidence of 50% to 60% were increased to 90% plus in LSTM model. The graphs shown below confirms this statement by clearly showing that curve of LSTM model is better than previous model.

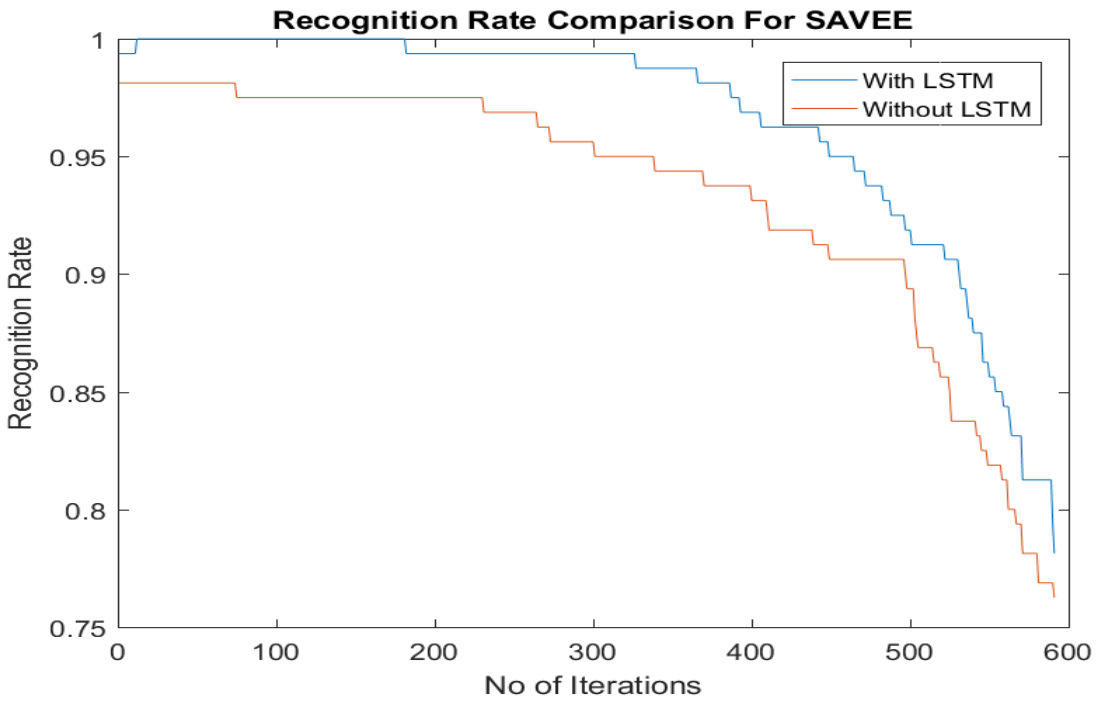

Figure 5-5 Frame wise Recognition Rate Comparison of SAVEE Dataset for both models.

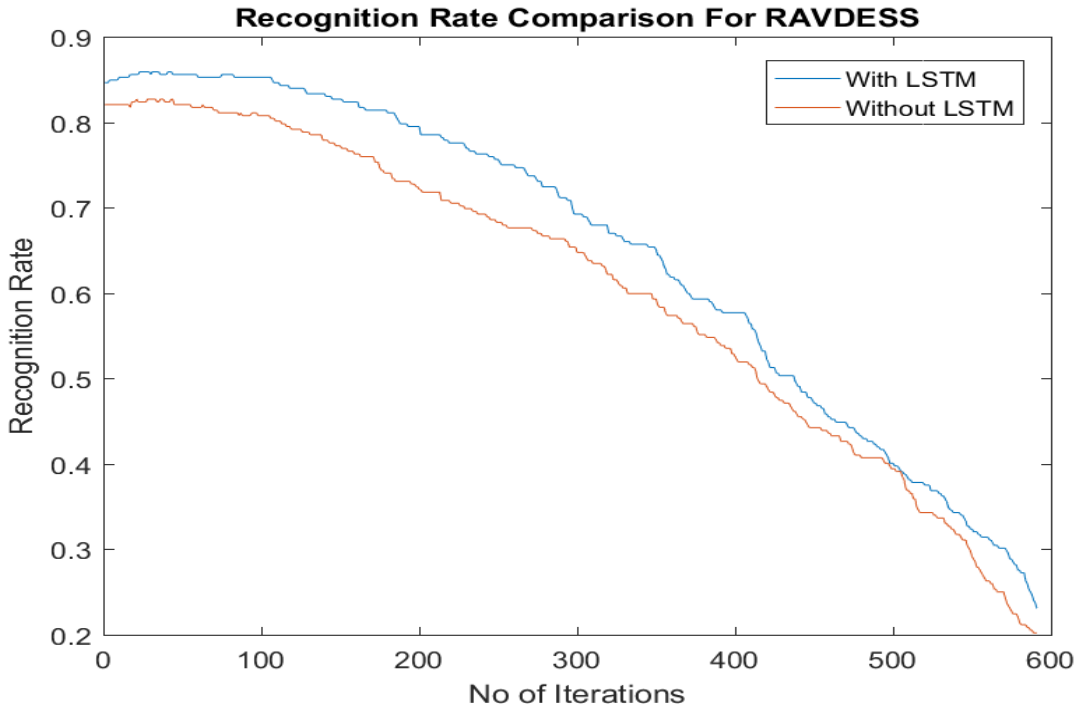

Figure 5-6 Frame wise Recognition Rate Comparison of RAVDESS Dataset for both models

Table 5-20: Comparison of visual recognition results

Dataset	**Refs**	**Recognition Rate (%)**
SAVEE	*Fuzzy ARTMAP neural network.* [16]	93.75
	Crosschannel CNN. [18]	93.9
	Our CNN Without LSTM	**98.8**
	Our CNN With LSTM	**99.64**
RAVDESS	*Deep Complete-CCA* [67]	66.69
	Domain Classification Lower face [54]	78.61
	Our CNN Without LSTM	**87.12**
	Our CNN With LSTM	**87.49**

On comparing our approach with the other recent recognition models as shown in Table 5-20 above, we can safely say that our CNN model outperforms almost every other model and offers near perfect recognition rates. Our proposed CNN model with LSTM further outperforms CNN model without LSTM in both SAVEE and RAVDESS by 0.84% and 0.37% respectively. Overall accuracy difference between both models is not significant but if we look at the graph on Fig:5-5 and Fig;5-6 then it can be clearly seen that LSTM model outperform simple CNN model in frame wise emotion recognition. LSTM model assign emotion to video with much more confidence than simple CNN model.

Chapter 6: CONCLUSION AND FUTURE WORK

6.1 Conclusions:

The results show that adding time factor as a feature in training improve our classification instead of using each frame of video independently. Time factor is added in our model by using LSTM layer which holds the output until last frame of the video.

Visual features were computed from each frame by mean of a non-sequential deep convolutional neural network. This model consists of four channels which cover both local and global features. These four channels are merged together and are further followed by convolution, max-pooling, relu, dense and dropout layers. Output emotion of each frame is stored in an array in which max value of particular emotion out of seven emotions is assigned to whole video.

For training and testing 3 ,4 and 15-fold Cross validation were used. The experimental results were based on SAVEE and RAVDESS daaset. The recognition rate on SAVEE dataset is 99.64% which shows improvement over previous state of art results by 5.89%, 5.74% respectively. The recognition rate on RAVDESS dataset is 87.49 % which shows improvement over previous state of art results by 20.8 %, 8.88% respectively. Most misclassified emotion in our system was fear.

6.2 Contributions:

- Fully automated Video emotion recognition from video clips.
- Review and comparison of recent recognition systems designed for emotion classification.
- Detailed experiments on two different datasets using two models (with and without LSTM).
- Achieving one of the highest recognition rates on all two employed datasets using CNN model. (RAVDESS, SAVEE).

6.3 Future Work:

For future work we plan to use audio features as 2d images and train our model on these images. After that combined set of images from both audio and video frames can be used for classification. This way feature level fusion will be tested and decision level fusion too. Dataset used in this research are created in a controlled environment e.g. fixed background which never change. This helps in face detection and background separation but this is not the case in real life. Different videos are made in different scenarios. We will focus on those databases which contain real life images and classify on those images. And finally, new networks are introduced such as Capsule Networks which takes only few images as input for training like human being and gives very good classification. This network can also be used for emotion classification.

References

[1] R. W. Picard, “Affective Computing: Challenges,” *Int. J. Hum. Comput. Stud.*, 1995.

[2] P. N. Johnson-Laird and E. Shafir, “The interaction between reasoning and decision making: an introduction,” *Cognition*, 1993.

[3] M. Clynes, *Sentics: The touch of emotions*. 1977.

[4] P. S. Bellet and M. J. Maloney, “The Importance of Empathy as an Interviewing Skill in Medicine,” *JAMA J. Am. Med. Assoc.*, 1991.

[5] J. F. Dovidio, J. A. Piliavin, D. A. Schroeder, and L. A. Penner, *The social psychology of prosocial behavior*. Psychology Press, 2017.

[6] P. Ekman, “Facial action coding system (FACS),” *A Hum. face*, 2002.

[7] I. A. Essa, “Analysis, interpretation and synthesis of facial expressions,” Massachusetts Institute of Technology, 1995.

[8] Y. Yacoob and L. Davis, “Computing spatio-temporal representations of human faces,” research directed by Dept. of Computer Science.University of Maryland at~…, 1994.

[9] J. M. Ben\’\itez, J. L. Castro, and I. Requena, “Are artificial neural networks black boxes?,” *IEEE Trans. neural networks*, vol. 8, no. 5, pp. 1156–1164, 1997.

[10] S. S. Haykin, S. S. Haykin, S. S. Haykin, and S. S. Haykin, *Neural networks and learning machines*, vol. 3. Pearson Upper Saddle River, 2009.

[11] W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, 1943.

[12] D. O. Hebb, *The organization of behavior: A neuropsychological theory*. Psychology Press, 2005.

[13] F. Rosenblatt, “The perceptron: a probabilistic model for information storage and organization in the brain.,” *Psychol. Rev.*, vol. 65, no. 6, p. 386, 1958.

[14] N. R. Hecht and others, “Theory of the backpropagation neural network,” in *International*

Joint Conference on Neural Networks, 1989, vol. 2, pp. 593–605.

[15] M. Minsky and S. A. Papert, *Perceptrons: An introduction to computational geometry*. MIT press, 2017.

[16] T. J. Sejnowski and C. R. Rosenberg, "NETtalk: a parallel network that learns to read aloud, Neurocomputing: foundations of research." MIT Press, Cambridge, MA, 1988.

[17] Y. LeCun *et al.*, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.*, vol. 1, no. 4, pp. 541–551, 1989.

[18] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Process. Mag.*, vol. 29, 2012.

[19] I. Sutskever, O. Vinyals, and Q. V Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[21] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, 2011, pp. 315–323.

[22] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, 2013, vol. 30, no. 1, p. 3.

[23] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.

[24] A. Coates, B. Huval, T. Wang, D. Wu, B. Catanzaro, and N. Andrew, "Deep learning with COTS HPC systems," in *International conference on machine learning*, 2013, pp. 1337–1345.

[25] O. Yadan, K. Adams, Y. Taigman, and M. Ranzato, "Multi-gpu training of convnets," *arXiv Prepr. arXiv1312.5853*, 2013.

[26] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey

striate cortex," *J. Physiol.*, vol. 195, no. 1, pp. 215–243, 1968.

[27] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol. Cybern.*, vol. 36, no. 4, pp. 193–202, 1980.

[28] P. J. Werbos, "Applications of advances in nonlinear sensitivity analysis," in *System modeling and optimization*, Springer, 1982, pp. 762–770.

[29] S. Linnainmaa, "The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors," *Master's Thesis (in Finnish), Univ. Helsinki*, pp. 6–7, 1970.

[30] S. Linnainmaa, "Taylor expansion of the accumulated rounding error," *BIT Numer. Math.*, vol. 16, no. 2, pp. 146–160, 1976.

[31] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," 1985.

[32] M. Andrychowicz *et al.*, "Learning to learn by gradient descent by gradient descent," in *Advances in Neural Information Processing Systems*, 2016, pp. 3981–3989.

[33] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The RPROP algorithm," in *Proceedings of the IEEE international conference on neural networks*, 1993, vol. 1993, pp. 586–591.

[34] P. J. Werbos and others, "Backpropagation through time: what it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[35] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, and others, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[36] T. Dettmers, "Understanding convolution in deep learning," *Retrieved March*, vol. 25, p. 2018, 2015.

[37] M. Hoai12, "Regularized max pooling for image categorization," 2014.

[38] Y. Wang, L. Guan, and A. N. Venetsanopoulos, "Kernel cross-modal factor analysis for information fusion with application to bimodal emotion recognition," *IEEE Trans. Multimed.*, vol. 14, no. 3, pp. 597–607, 2012.

[39] S. Haq, T. Jan, A. Jehangir, M. Asif, A. Ali, and N. Ahmad, "Bimodal human emotion classification in the speaker-dependent scenario," *Pakistan Acad. Sci. Islam.*, vol. 27, 2015.

[40] M. Rashid, S. A. R. Abu-Bakar, and M. Mokji, "Human emotion recognition from videos using spatio-temporal and audio features," *Vis. Comput.*, vol. 29, no. 12, pp. 1269–1275, 2013.

[41] W.-Y. Chang, S.-H. Hsu, and J.-H. Chien, "FATAUVA-Net: An integrated deep learning framework for facial attribute recognition, action unit detection, and valence-arousal estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 17–25.

[42] R. KalaiSelvi, P. Kavitha, and K. L. Shunmuganathan, "Automatic emotion recognition in video," in *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)*, 2014, pp. 1–5.

[43] J. Grobova, M. Colovic, M. Marjanovic, A. Njegus, H. Demire, and G. Anbarjafari, "Automatic hidden sadness detection using micro-expressions," in *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, 2017, pp. 828–832.

[44] S. Zhalehpour, O. Onder, Z. Akhtar, and C. E. Erdem, "BAUM-1: A spontaneous audio-visual face database of affective and mental states," *IEEE Trans. Affect. Comput.*, vol. 8, no. 3, pp. 300–313, 2017.

[45] K. P. Seng, L.-M. Ang, and C. S. Ooi, "A Combined Rule-Based & Machine Learning Audio-Visual Emotion Recognition Approach," *IEEE Trans. Affect. Comput.*, vol. 9, no. 1, pp. 3–13, 2018.

[46] F. Cid, L. J. Manso, and P. Núnez, "A Novel Multimodal Emotion Recognition Approach for Affective Human Robot Interaction'," *Proc. FinE*, pp. 1–9, 2015.

[47] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on local binary patterns: A comprehensive study," *Image Vis. Comput.*, vol. 27, no. 6, pp. 803–816, 2009.

[48] A. Dhall, A. Asthana, R. Goecke, and T. Gedeon, "Emotion recognition using PHOG and LPQ features," in *Face and Gesture 2011*, 2011, pp. 878–883.

[49] A. Shaukat and K. Chen, “Towards automatic emotional state categorization from speech signals,” in *Ninth Annual Conference of the International Speech Communication Association*, 2008.

[50] A. Shaukat, M. Aziz, and U. Akram, “Facial Expression Recognition Using Multiple Feature Sets,” in *2015 5th International Conference on IT Convergence and Security (ICITCS)*, 2015, pp. 1–5.

[51] R. A. Khan, A. Meyer, H. Konik, and S. Bouakaz, “Framework for reliable, real-time facial expression recognition for low resolution images,” *Pattern Recognit. Lett.*, vol. 34, no. 10, pp. 1159–1168, 2013.

[52] F. Ping, J. Dongmei, W. Fengna, R. Ilse, and S. Hichem, “Manifold analysis for subject independent dynamic emotion recognition in video sequences,” in *2009 Fifth International Conference on Image and Graphics*, 2009, pp. 896–901.

[53] S. Haq, P. J. B. Jackson, and J. Edge, “Speaker-dependent audio-visual emotion recognition.,” in *AVSP*, 2009, pp. 53–58.

[54] B. Zhang, G. Essl, and E. M. Provost, “Recognizing emotion from singing and speaking using shared models,” in *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2015, pp. 139–145.

[55] J. Yan *et al.*, “Convolutional neural networks and feature fusion for bimodal emotion recognition on the emotiW 2016 challenge,” in *2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2017, pp. 1–5.

[56] L. Chao, J. Tao, M. Yang, Y. Li, and Z. Wen, “Long short term memory recurrent neural network based encoding method for emotion recognition in video,” in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2752–2756.

[57] M. Paleari, B. Huet, and R. Chellali, “Towards multimodal emotion recognition: a new approach,” in *Proceedings of the ACM international conference on image and video retrieval*, 2010, pp. 174–181.

[58] K.-C. Huang, H.-Y. S. Lin, J.-C. Chan, and Y.-H. Kuo, “Learning collaborative decision-

making parameters for multimodal emotion recognition," in *2013 IEEE International Conference on Multimedia and Expo (ICME)*, 2013, pp. 1–6.

[59] C. Fadil, R. Alvarez, C. Martinez, J. Goddard, and H. Rufiner, "Multimodal emotion recognition using deep networks," in *VI Latin American Congress on Biomedical Engineering CLAIB 2014, Paraná, Argentina 29, 30 & 31 October 2014*, 2015, pp. 813–816.

[60] D. Gharavian, M. Bejani, and M. Sheikhan, "Audio-visual emotion recognition using FCBF feature selection method and particle swarm optimization for fuzzy ARTMAP neural networks," *Multimed. Tools Appl.*, vol. 76, no. 2, pp. 2331–2352, 2017.

[61] S. Zhang, S. Zhang, T. Huang, W. Gao, and Q. Tian, "Learning Affective Features With a Hybrid Deep Model for Audio--Visual Emotion Recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 10, pp. 3030–3043, 2018.

[62] F. Noroozi, M. Marjanovic, A. Njegus, S. Escalera, and G. Anbarjafari, "Audio-visual emotion recognition in video clips," *IEEE Trans. Affect. Comput.*, 2017.

[63] J. Xue, Z. Luo, K. Eguchi, T. Takiguchi, and T. Omoto, "A Bayesian nonparametric multimodal data modeling framework for video emotion recognition," in *2017 IEEE International Conference on Multimedia and Expo (ICME)*, 2017, pp. 601–606.

[64] N. Ronghe, S. Nakashe, A. Pawar, and S. Bobde, "Emotion recognition and reaction prediction in videos," in *2017 Third International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, 2017, pp. 26–32.

[65] B. Cheng *et al.*, "Robust emotion recognition from low quality and low bit rate video: A deep learning approach," in *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*, 2017, pp. 65–70.

[66] P. Barros and S. Wermter, "Developing crossmodal expression recognition based on a deep neural model," *Adapt. Behav.*, vol. 24, no. 5, pp. 373–396, 2016.

[67] Y. Liu, Y. Li, and Y.-H. Yuan, "A Complete Canonical Correlation Analysis for Multiview Learning," in *2018 25th IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 3254–3258.

[68] S. R. Livingstone and F. A. Russo, “The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English,” *PLoS One*, vol. 13, no. 5, p. e0196391, 2018.

[69] P. Jackson and S. Haq, “Surrey audio-visual expressed emotion (savee) database,” *Univ. Surrey Guildford, UK*, 2014.

[70] Tim Dettmers. Deep learning in a nutshell: History and training. https://devblogs.nvidia.com/parallelforall/ deep-learning-nutshell-history-training, 2015. [Online; Accessed 15 Feb 2019].

[71] Understanding LSTM Networks. http://colah.github.io/posts/2015-08-Understanding-LSTMs/ [Online; Accessed 15 Feb 2019].

[72 Rensselaer Polytechnic Institute. Automatic facial action units recognition. https://www.ecse.rpi.edu/~cvrl/tongy/aurecognition.html, 2015. Online; Accessed 15 March 2019]

[73] Conner DiPaolo. Perceptron. https://github.com/cdipaolo/goml/tree/ master/perceptron, 2015. [Online; Accessed 15 March 2019]

[74] Andrej Karpathy. Cs231n convolutional neural networks for visual recognition. http://cs231n.github.io/convolutional-networks/, 2015. [Online; Accessed 15 March 2019].