Children's Spontaneous Facial Expression Recognition Using Deep Learning Methods

Author

Unqua Laraib

Regn Number

FALL 2018-MS-18(CSE) 00000275471

MS-18 (CSE)

Thesis Supervisor:

Dr. Arslan Shaukat

DEPARTMENT OF SOFTWARE ENGINEERING

COLLEGE OF ELECTRICAL AND MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCE AND TECHNOLOGY

ISLAMABAD

SEPT, 2022

Children's Spontaneous Facial Expression Recognition Using Deep Learning Methods

Author

Unqua Laraib

FALL 2018-MS-18(CSE) 00000275471

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

MS Software Engineering

Thesis Supervisor:

Dr. Arslan Shaukat

Thesis Supervisor's Signature: _____

DEPARTMENT OF COMPUTER & SOFTWARE ENGINEERING

COLLEGE OF ELECTRICAL & MECHANICAL ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY, ISLAMABAD

SEPT, 2022

DECLARATION

I certify that this research work titled "*Children's Spontaneous Facial Expression Recognition Using Deep Learning Methods"* is my own work under the supervision of Dr. Arslan Shaukat. This 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

Unqua Laraib

FALL 2018-MS-18(CSE) 00000275471

_____

Signature of Supervisor

LANGUAGE CORRECTNESS CERTIFICATE

This thesis is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the University for MS thesis work.

_____

Signature of Student

Unqua Laraib

FALL 2018-MS-18(CSE)

00000275471

_____

Signature of Supervisor

COPYRIGHT STATEMENT

ACKNOWLEDGEMENTS

All praise to Allah Almighty who is the Most Merciful and the Most Benevolent. Without a doubt I would have not been able to achieve anything without His help and countless blessings. Indeed He eased my way and gave me guidance, strength and patience throughout this journey. Truly, none is worthy of praise but ALLAH Almighty.

After ALLAH Almighty, I am eternally grateful to my family and would like to express my genuine gratitude toward them for being there for me when I needed support and motivation. I am especially grateful to my mother for her moral support, encouragement and prayers that stayed with me throughout.

I would like to pay special thanks to my respected supervisor Dr. Arslan Shaukat for his guidance, continuous assistance, inspiration, and doing more than what a student could ask for: boosting my morale and helping me stay strong and motivated for my research. My co-supervisor, Dr. Rizwan Ahmed Khan deserves equal amount of gratitude, as without his help and invaluable guidance this would have not been possible.

I want to acknowledge my entire thesis committee: Dr. Usman Akram and Dr. Farhan Hussain for their cooperation and helpful suggestions. Their guidance means a lot to me.

Lastly, I would like to thank my seniors and friends who guided, encouraged and supported me in my research.

Dedicated to my exceptional parents: ***Maj(R) Muhammad Siddiq & Zahida Parveen,*** *adored siblings and wonderful friends whose tremendous support and cooperation led me to this accomplishment*

ABSTRACT

For a child to incorporate himself into the society as a productive individual, it is vital that his emotional development is sound. In recent years, the emergence of several real-world applications has shifted the focus of technology towards enabling machines to decode and understand the emotional signals of human beings. However, in research, very often emotion recognition is limited to adults, not considering the fact that children can develop an awareness of various emotions at a very early stage, thus, the ability of machines, to distinguish various facial expressions of children, still needs to be explored. In the absence of a standardized database, it yet remains a challenge. Thus a set of benchmarks are required to establish a standardized comparison. In this paper, a system based on convolutional neural networks has been proposed to automatically recognize children's expressions. For this purpose, a video dataset for Children's Spontaneous facial Expressions (LIRIS-CSE) has been used. In our proposed system, pre-trained Convolutional Neural Network (CNN) such as VGG19, VGG16, and Resnet50 have been deployed as feature extractors and then models such as Support Vector Machine (SVM) and Decision Tree (DT) have been used for classification. We have tested their strength with various experimental setups such as 80-20% split, K-Fold Cross Validation (K-Fold CV), and Leave One out Cross-Validation (LOOCV), for both image-based and video-based classification approaches. Our research has achieved a promising classification accuracy of **99%** for image-based classification via features of all three networks with SVM using 80-20% split and K-Fold CV. Video-based classification results have been reported as well, where we have managed to achieve **94%** accuracy via features of VGG19 with SVM using LOOCV. Our achieved results are better as compared to the original work, where they have achieved an average image-based classification accuracy of **75%** on their designed LIRIS-CSE dataset.

Keywords – Facial Expression Recognition, Classification, CNN, Feature Extraction, SVM, DT.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

CHAPTER 1: INTRODUCTION

In this chapter we provide a detailed introduction of our research work which includes motivation, problem statement, research contributions, applications and structure of the thesis.

For the past few decades, there has been a great deal of advancement in technology, revolutionizing the world for the better good of humanity. This technological boom shows no sign of stopping and continues to improve and automate human lives, proving that there is always more room for improvement. What started almost two million years ago, tools made out of stone are considered the world's first technological invention. Later on, the creation of transistors which is considered the 20th century's most important invention, paved the way for advancement in technology and digitization. In the 1980's began the digital revolution with the Internet and mobile phones and so did the human's reliance on machines and computing power. This resulted in automation of our daily tasks and productive competence. However, in recent years the concept of Artificial Intelligence (AI) has emerged which differs from the traditional automated systems that can only perform a certain task by following a set of instructions given to it. With artificial intelligence, the machines are now able to perform a task based upon its learning and imitate the human's ability to learn. The potential AI holds is mind-blowing and it has already found its place in numerous fields such as Robotics, computer vision, machine learning, knowledge reasoning, games and much more. AI is progressing at a rapid speed continuing to amaze the world.

1.1 Motivation

Humans are social animals and in order to survive in society we must communicate, interact and socialize with one another. In order to effectively convey our message we need to express emotions. Emotions either expressed facially or added in speech or both can play a vital role in delivering the message effectively. However, in humans, this ability does not come naturally, it cultivates with time and it is very important to develop an understanding of emotions at an early stage so that we can manage to control them, express ourselves better and live a healthy life altogether. For different age groups, the understanding and interpretation of emotions varies. For children, especially, it is very important to develop the

understanding of emotions early on so they can become an active part of the society, interact and socialize better. At an early stage they may require help from adults in recognizing and naming emotions. With the passage of time children learn to understand *other's* and express *their* emotions. It is crucial to make children understand what they are feeling and this can be achieved via emotional language. For example if the child has a smile on his/her face, adults can use this language to tell them that they are feeling happy etc. As they grow, they are able to recognize emotions without the adult's help. Studies show that those children who can understand and manage their emotions are likely to have good communication skills, better interpretation of messages, improved responsiveness and control impulses. Considering the importance of these emotions it is equally significant for machines to completely understand and comprehend the message given to them which can greatly improve the way humans interact with them. As discussed, emotions can change the notion of the message entirely, so if machines are able to decode and distinguish between the various kinds of emotions it can revolutionize the Human Computer Interaction (HCI), resulting in advancement in various relevant fields such as Robotics, Mobile computing and much more. Facial emotion recognition (FER) is one of the most noteworthy topics in the area of Computer Vision, Artificial Intelligence (AI), Entertainment, Human-Computer Interaction (HCI), Advanced Driver Assistance Systems (ADASs), Virtual Reality (VR), and Augmented Reality (AR). In short, FER is an active area of research and has found its application in almost every field still having room for more contributions by researchers. As we become more and more dependent on machines, these man made contributions can come up with such intelligent technology which has the ability to not only comprehend the message but also learn from it.

1.2 Problem Statement

One of the active areas of research where a lot of work is being done is enabling machines to decipher the emotions like humans. The task of distinguishing among various emotional states is not as easy for machines as it is for humans since we have the ability to discern and decode emotions of any person who is in front of us. For machines this is a yet a hard task because they are affected by many factors that, if not taken into account can affect the performance of the system. These factors include presence of noise, occlusions, un-evenness

in pose, lightning, and variations in presenting expression by subjects across different areas of the world [1]. Another issue that can't be ignored is the lack of research work done for emotion recognition in children. For adults, emotion recognition is a well-researched area as there are a number of quality datasets pertaining to adult's facial emotions for the machines to be trained and tested on. In the absence of standardized databases of children's emotions, it yet remains a challenge for machines. Thus a set of benchmarks are required to establish a standardized comparison. Coming up with such complex and robust systems that have been trained on emotional datasets for children is necessary so they can perform remarkably well even under the effect of various factors.

1.3 Aims and Objectives

Aims and objective of our research are as follows:

- A comparative study of state of the art technologies and recent progress made in the field of emotion recognition in children.
- Visual feature extraction using state of the art algorithms or methodologies.
- Classification of basic emotional states based on visual features extracted from a video dataset.

1.4 Structure of Thesis

The structure of the thesis can be viewed in Figure 1.1.

Chapter 2 discusses the significance of emotions in children and how they are categorized and recognized. Moreover, basic emotion recognition model and techniques are covered and neural networks have been discussed in detail.

Chapter 3 discusses the contributions made in the field of emotion recognition in adults and children. Various emotional databases for children used in research and the traditional and CNN based techniques deployed for emotion recognition have been reviewed as well.

Chapter 4 explains the proposed methodology and strategy for extraction of useful features

for image and video based classification methodologies.

Chapter 5 discusses the carried out experiments and the database used.

Chapter 6 covers the existing limitations, the possible future work and conclusion of the thesis.

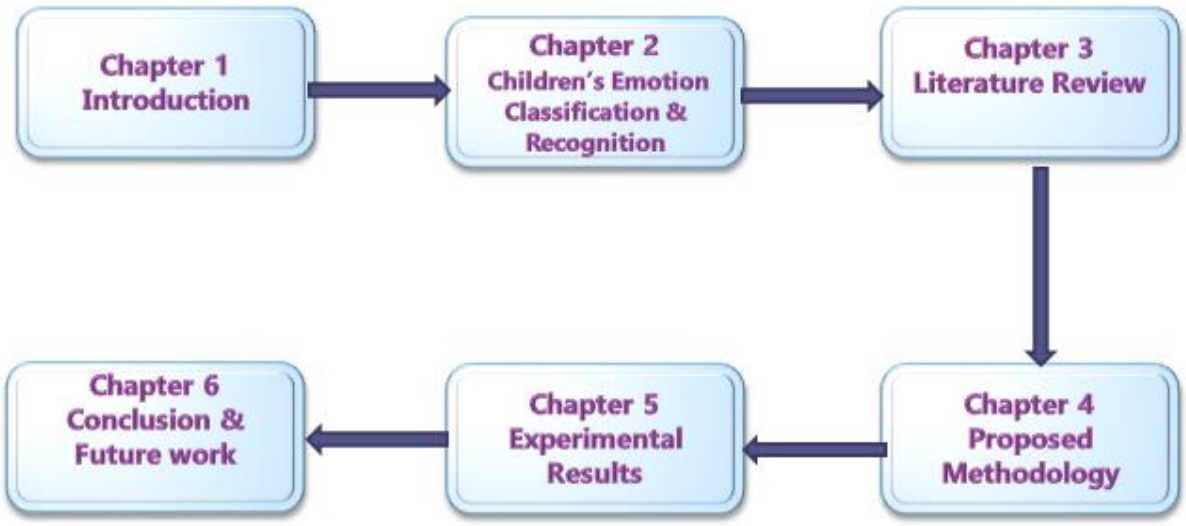

Figure 1.1 Structure of Thesis

Chapter 2

Children's Emotion Classification and Recognition Models

Chapter 2: CHILDEREN'S EMOTION CLASSIFICATION AND RECOGNITION MODELS

An emotion is a feeling, a state of mind or simply a physiological state that can be induced in a child due to any of the several reasons. The emotional states of a child are dependent on personal thoughts, perception and attitudes of others or simply the situation they are part of. Just like adults, children also express their emotions either through facial expressions or speech. Emotions however can be much more complex and difficult and can change how a child perceives the world and others perception of him/her.

2.1 Emotion Classification

Emotions can be classified based upon two different views. First view is about discreteness and independence of each emotional state. In simple words the first idea is of the view that each emotion is independent of each other and it is not affected by another state. That is there is no interconnection among the various states. The second view says that all emotional states are interconnected and can be grouped based upon dimensions.

2.1.1 Basic Emotional States

For a child to incorporate himself into the society as a productive individual, it is vital that his emotional development is sound. Cognition begins at a very early stage in children as they pick up behavior and respond to various stimuli in a specific manner. Similarly, they can understand the emotional states presented by others and the message behind it. This understanding has classified the emotional states into distinct categories. Ekman et al. [2] presented in his research that emotional states can be divided into six basic categories namely anger, disgust, fear, happiness, sadness and surprise. These basic states are distinct, having unique characteristics about them. Furthermore, the research in [2] proposed that each emotional state is a discrete entity and can be presented in terms of degrees.

2.1.2 Multi-Dimensional Representation

Researchers have represented emotional states in the form of dimensions rather than presenting them as discrete categories. The reason behind this dimensional representation is that emotional distance between different experiences can be visually displayed and

explained in a better way. Thayer et al. [3] gave a multi-dimensional representation of the emotional states where each state is mapped onto a two dimensional plane, moreover weighing each emotion in terms of its activation-valance also mapped into two dimensions. [3] Further explains that activation is the degree of excitation of an emotion where valence represents the degree of positivity and negativity of an emotional experience. Figure 2.1 shows a two dimensional activation-valence emotional plane.

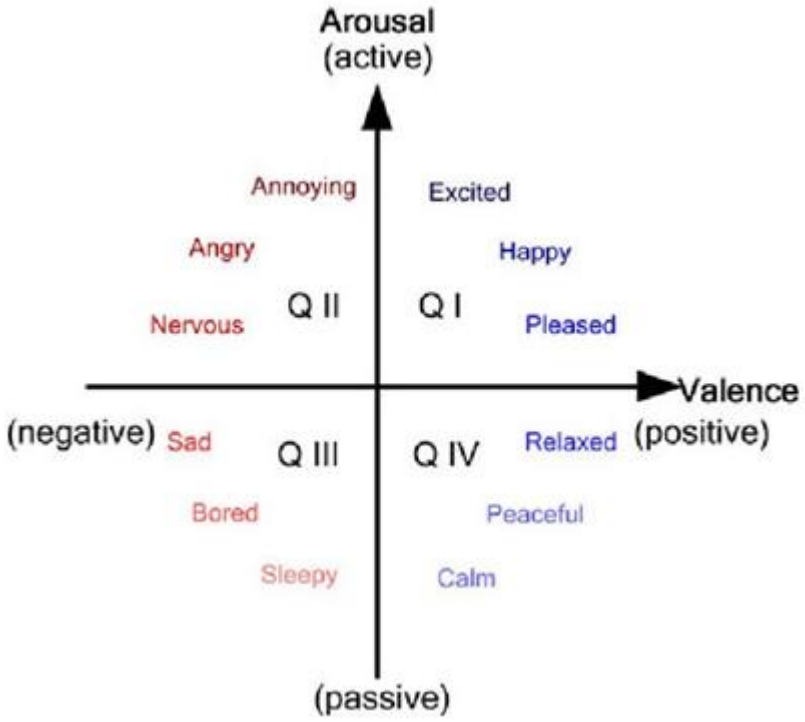

Figure 2.1 Valence Activation 2D Emotional Plane [3]

2.2 Affective Computing

Affective computing is the study of creating intelligent and smart systems that can not only decode, comprehend and understand but also imitate emotions [4]. Rosalind Picard [5] has explained in his research, that communication; decision making and learning are dependent and based upon emotions [6]. Affective computing is based upon the motivation of enabling machines to understand human emotions naturally and thus make rational decisions and give responses based upon that understanding [7]. This can be made possible with robust systems which are able to extract meaningful patterns and useful information from the input data and

then apply various machine learning techniques and algorithms to process the extracted information. Various patterns include speech recognition, Facial Expression Recognition (FER), and Natural Language Processing (NLP). Our research, in particular, focuses on facial expression recognition.

In a nutshell, affective computing aims to enable machines to interpret and understand different emotional states and 'learn' from it. Based upon learning, machines should be smart enough to make reasonable, polite and rational decisions.

2.3 Visual Emotion Recognition System

After complete understanding of emotions has been developed, researchers began to come up with ideas and techniques that has enabled machines to learn and predict the emotional states automatically either through facial expressions, postures or even speech as these are the basic mediums through which emotions are conveyed [8]. Researchers have implemented one or all of these modalities in their research to predict the correct emotion [9]. Similarly, in our research we have adopted visual content to recognize emotions from video data. Figure 2.2 shows a typical visual FER system.

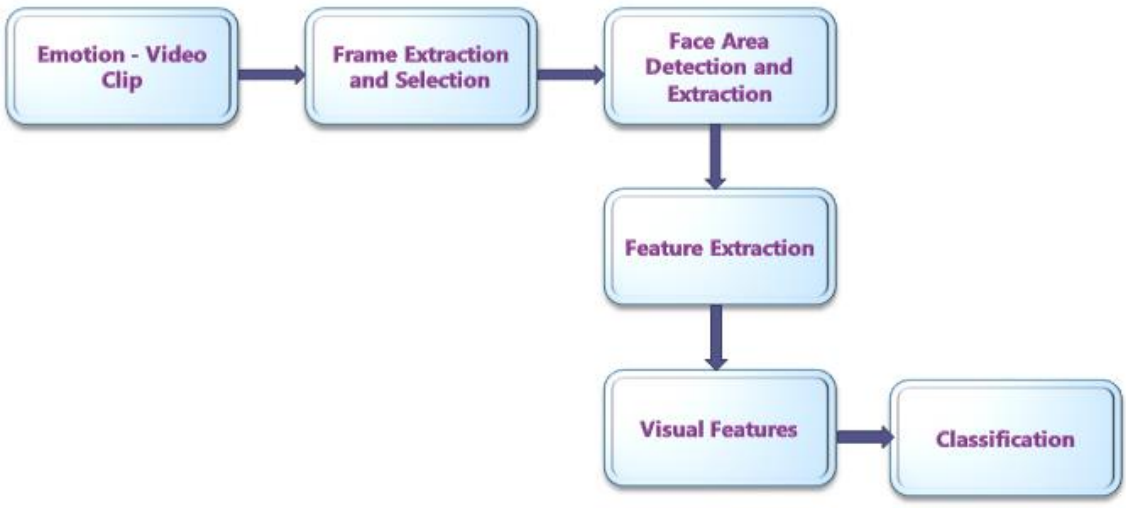

Figure 2.2 Visual Emotion Recognition System

Firstly, a video clip is given as input to the visual FER system which extracts multiple

frames or images from it (each video clip). Only useful frames are further passed on to avoid unnecessary processing. The second step is detection and extraction of just the face area from these frames. The next step is the extraction of useful features from these frames having face area only. In the last step classification takes place with a prediction model which is trained on these features and has learned to predict a specific category or a label which in our case is one of the basic emotional states.

2.3.1 Visual Corpus

In the recent years, there has been a lot of research work in the computer vision and machine learning domain which has led to creation of several datasets which are publically available for experimentation for researchers or they can create their own. We too have employed a publically available visual dataset for children's emotions in our research. There are three major categories of datasets. These types are as follows:

- **Spontaneous**

In this type of datasets, the emotions exhibited by the subjects are natural and not posed. The subjects display emotions in response to any sort of interaction. However expressions in these datasets are very hard to classify due its spontaneous nature. The subjects can change their position or posture making it challenging to extract useful features from the frames. Examples of such dataset are LIRIS-CSE, EmoReact, AFEW, BAUM-1s, BAUM-2s, SEWA and RECOLA etc.

- **Induced**

In this type the emotions are induced in the subjects through some source. The induction of the emotions is in response to any source like a video clip from a movie or a cartoon (for children). Examples of such datasets are JEMIme, SAVEE etc.

- **Posed**

Posed datasets are the easiest to test as they contain videos or images of actors who pose for the camera and give a certain expression following a script. These datasets are not as challenging because the actors are directly facing the camera with proper lighting and

background; moreover their expressions are highly exaggerated making it easier to extract quality visual features and work on them. Radbound, NIMH-ChEFS, DartMouth, CAFÉ, eNTERFACE'05 and RML are examples of posed datasets.

In our research we have selected the LIRIS-CSE dataset for our emotion recognition system. LIRIS-CSE falls in the spontaneous category in which emotions presented by the subjects are 100% spontaneous.

2.3.2 Feature Extraction

In this stage, useful visual features are extracted from videos presented in the dataset. These features are extracted from frames as they were extracted in the pre-processing stage. After extraction of features, these are passed onto a model for classification.

- **Visual Features**

For feature extraction, the prerequisite is a set of frames which contains face area only. From those frames, visual features, which are a representation of the data, are extracted. Generally, facial features are divided into two main categories: Geometric and Appearance features [10]. Geometric features hold information about specific areas of the face such as eyes, eyebrows, nose and mouth and their corner points. Whereas appearance based features deal with the entire face area or a certain region instead of focusing on a specific part of the face. Geometric features based methods are challenging to deploy as it requires precise facial feature detection which can be very difficult to handle at times. Still, these features are used by researchers in their research. For appearance based features, Gabor filters can be deployed to extract them; this is one method [11].

For feature extraction from images, in general, there are two main approaches. First the traditional approach which focuses on extracting hand-crafted features using methods such as LBP, SIFT and Gabor wavelets etc. These hand crafted features are manually calculated as this involves a set of features which are first defined and then extracted. These traditional approaches are still being used by researchers e.g. LBP which is less time consuming [12]. The other approach is representation learning which automatically extracts features from the input data. This method involves extraction of features using an effective machine learning technique which has the ability to learn by training. It can then make a decision about which

features should be extracted [13]. These deep learning methods have overcome the traditional or conventional hand crafted feature extraction methods for emotion classification and have proven their worth [14]. This is achievable with Convolutional Neural Networks (CNN) which is a combination of feature extraction and classification [15, 16].

Since the dataset being used in our research is a spontaneous one which can be very tricky when it comes to feature extraction, we have deployed state of the art representation learning techniques.

2.3.3 Classification

In the classification stage, various models are trained and tested on the extracted features by using either deep learning methods such as CNN or traditional models such as SVM or Decision Tree. In our research for classifications we have opted for two models namely Support Vector Machine (SVM) and Decision Tree (DT). Classification stage tells us how our models performed based on the predictions accuracy.

2.4 Machine Learning

In this ever-growing digital age where humans were becoming more and more dependent on computers, there was need for machines to do more than just follow set of pre-defined instructions. Thus the concept of Machine Learning (ML) was introduced. This term was first presented by Arther Samuel in 1959. Machine learning is the study and development of systems which does not need to be programmed to follow certain instructions rather it has the ability to learn from input data. It finds an algorithm that can learn and improve with training without any explicit programming. This involves decision making or predictions about the input data hence the term predictive analytics is also given to machine learning [17]. A field related to machine learning is known as Artificial Intelligence (AI) which is the development of intelligent systems that is capable of thinking and making decisions like humans do. Both fields are highly interlinked and need to work together to create intelligent systems. Machine learning is the subset of AI which can be divided into three main categories. Figure 2.3 shows the sub types of machine learning which are explained below:

- **Supervised learning**

In supervised learning, the inputs given to the system are labeled i.e. the inputs are given proper class names which shows that this particular input sample belongs to this particular class. Examples of supervised learning are Regression and classification models.

- **Un-supervised learning**

In unsupervised learning, the inputs given to the system are not labeled with class names and thus the model or classifier trains on unlabeled data to find out sequences and structures in the data. Although this type of learning can be used to solve more complex problems yet it can be more unpredictable and tricky at times. Cluster analysis comes under unsupervised learning.

- **Reinforcement learning**

In this type of learning, there exists an agent whose actions and behaviors are observed and taken into consideration under various situations in order to achieve a specific goal. The agent is given either a prize or is punished based upon how he performed in a given scenario. In a nutshell the agent learns to make a sequence of decisions under complex scenarios. Chess game is an example of reinforcement learning.

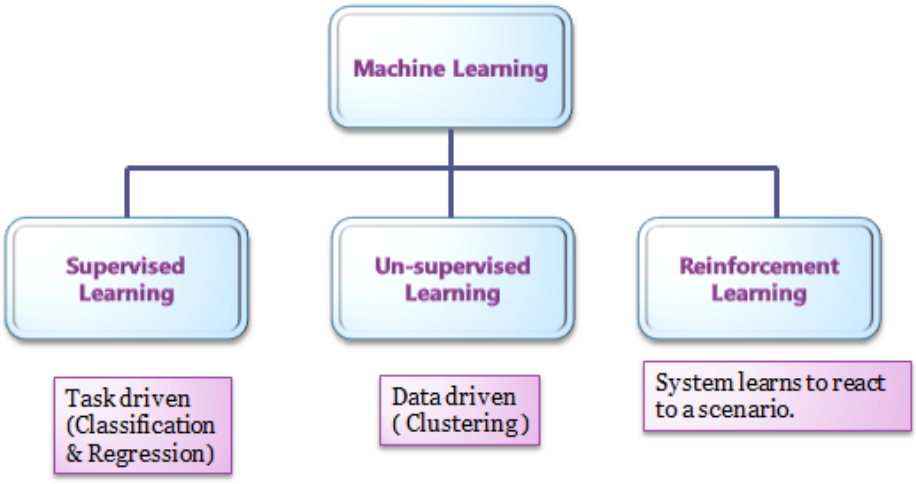

Figure 2.3 Machine Learning Sub-Types [17]

Our research falls under supervised learning category as the input data videos are labeled

with unique emotional states as class names.

2.5 Classic Machine Learning Algorithms

In this section, we discuss various classical machine learning algorithms. The performance of these models is dependent on the type, distribution and size of training data, type and number of features, complexity etc. Thus, the algorithms discussed below may not perform efficiently in all cases; rather there are so many factors that should be taken into account before selecting a specific algorithm.

2.5.1 Linear Regression

When getting started with machine learning, linear regression is a good starting point. Linear regression is a machine leaning, regression based model that assigns weight parameters, theta (θ) for all training features. In the beginning of the training process, random values are initialized to theta; however these values are updated in correspondence to the feature throughout the process such that the loss is diminished. Finally, the predicted output is a linear function of theta coefficients and features. The predicted output is depicted as:

$$h\theta = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots \quad (2.1)$$

2.5.2 Logistic Regression

Logistic regression is a popular classification model. It makes use of logistic function to generate a binary output model. Logistic regression finds a probability $0 \leq x \leq 1$ of binary 0 (failure) or 1 (success). This model works very similar to linear regression and calculates a linear output. One of the most commonly used logistic functions is sigmoid function. Similar to linear regression, the value z can be presented as:

$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots \quad (2.2)$$

$$h(\theta) = g(z) \quad (2.3)$$

$$\mathrm{g(z)} = \frac{1}{1+e^{-z}} \quad (2.4)$$

Here $h(\theta)$ relates to P(y=1|x) that is output probability to be 1 given input x where P(y=0|x) equals $1 - h(\theta)$.

2.5.3 K-Nearest Neighbors

K- Nearest Neighbor (KNN) is one of simplest, supervised and non-parametric method deployed for classification and regression problems. In this method, the test data point is assigned a label based on neighborhood exploration i.e. a distance of the test point from all k neighbors is calculated and the category of nearest neighbor is assigned to the test data point. KNN is a lazy model and the value of K is always an odd number.

2.5.4 Decision Tree

Decision tree is made up of independent variables where each node has a condition over a feature. Based on the condition, it is decided which node to navigate to and the output is predicted after the leaf node is reached. The condition of the nodes is selected based upon entropy/information gain. For example CART is an algorithm for selecting conditions that uses Gini index as classification or splitting criteria.

$$gini\ index = 1 - \sum p_t^2 \tag{2.5}$$

Decision tree does not require data preprocessing and provides logical description over the prediction. Moreover it includes several hyper-parameters such as criterion, max-depth, min-samples split and min-samples leaf. However, decision tree is prone to outliners and may become extremely complex.

2.5.5 Support Vector Machine (SVM)

SVM is a supervised machine learning algorithm most commonly used for image, text classification or regression problems. SVM uses a kernel function for complex problems and outliers are wisely dealt with by using soft margin constant C. SVM has two variants; linear and non-linear SVM.

The linear SVM is used when the problem space is or must be linearly separable. A hyper-plane is generated by the model which maximizes the margin between the support vectors. If

there are N features, the generated hyper-plane would be an N-1 dimensional subspace. Linear SVM uses no kernel. In order to maximize the margin, ||w|| needs to be minimized where w depicts a set of weight matrices. The optimization equation will be written as:

$$Minimize \ \frac{||w||^2}{2} \tag{2.6}$$

The non-linear SVM is used when the data is not linearly separable. It uses a kernel function to generate a hyper-plane such that the labels are distributed in a way so the training data is linearly separable. Later, the labels are classified with a linear curve in the hyper-plane, out-putting a non-linear solution. The equation of linear SVM is changed only by introducing a new kernel.

$$Minimize \ \frac{||w||^2}{2} + C \sum_i \zeta_i \tag{2.7}$$

2.5.6 Random Forest

A random forest is an ensemble of several decision trees which are combined together to generate a robust and accurate model. Random forest can be used for classification and regression problems via bagging method with an ensemble of decision trees. In order to calculate classification results, majority voting is used by the model whereas in regression problems the mean is calculated. Random forest deals with over fitting proficiently and can work with binary, categorical and continuous features. However, the model can become quite complex and computationally expensive as it grows larger.

2.6 Artificial Neural Networks

In 1943, Warren McCulloch and Walter Pitts gave the concept of Artificial Neural Network (ANN) along with a computation model [18]. As the name suggest an Artificial Neural Network is a framework based on human nervous system. It works the same way our brain processes information i.e. with a network of neurons that are mutually connected through links. ANNs use artificial neurons or connected units arranged in multiple layers to enable the machines to learn and make reasonable decisions like humans. An ANN is composed of following things:

- **Input layer**

The input layer comes at the start of a neural network, bringing the input data into in the framework for further processing. This input data is received by a bunch of neurons present in the input layer. The neurons are also known as nodes.

- **Hidden layer**

The hidden layer also comprises of neurons. Its job is to apply some kind of functions or transformations on the input data passed on from the input layer, to convert it in a form that can be helpful for output layer to make a correct decision.

- **Activations and Weights**

Each neuron has a specific value know as its activation and that neuron is ‘lit up’ when that number is a higher number. Weights are also very important for transforming the node’s value into a number that is more suitable for the final output. Higher the weight, more the impact it will have on the neuron’s value or activation. Each neuron of one layer is fully connected to the neurons of the subsequent layer and the links between these layers have weights assigned to them.

In short, when a neural network takes the input, the nodes of the input layer hold those input values and passes them onward to the hidden layer. The hidden layer does some sort of processing on those values and passes it on to the next hidden layer; this means that the output of the previous layer becomes input of the subsequent layer until the final output layer is reached which makes the final decision. Figure 2.4 presents the structure of an ANN.

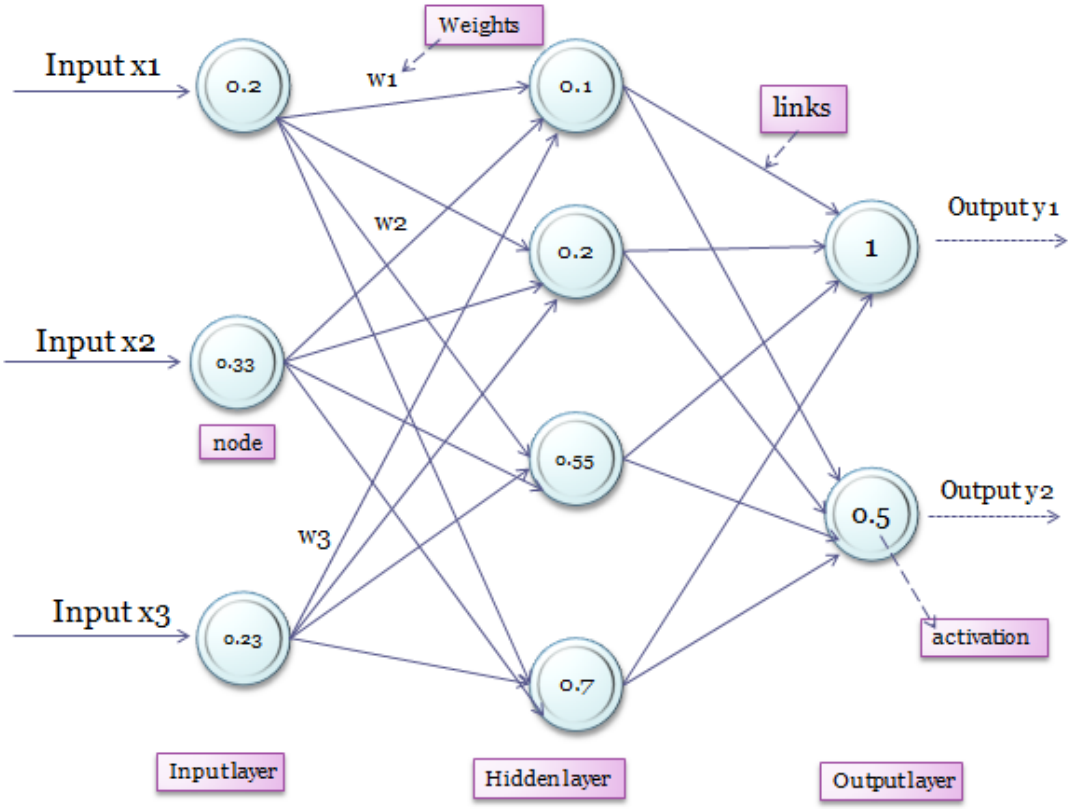

Figure 2.4 Structure of an Artificial Neural Network [18]

It took ten years to implement the very first ANN [19] model. The first model of a simple neural network was called Perceptron, which had the ability to train on a supervised data and make decisions based upon its learning. The architecture of Perceptron involves an input layer consisting of input nodes holding input values as x_1, x_2,…,x_n. Whereas w_1, w_2,…,w_n shows the weights on the corresponding links. The value of each node is basically a weighted-sum to which a bias (θ) is added to make the model fit the data. Bias is a constant that makes the neuron act more as linear functions and also adjusts the output.

$$y = f(t) = \sum_{i=1}^{n} X_i * W_i + \theta \qquad (2.8)$$

The result of *f (t)* is the input to the "activation function" such as "Binary step function" since perceptron is a binary classifier. This means that the output can either be 1 or 0. At the end of the process, the predicted values by the network are compared with actual values and error value is generated. If the error rate is high it must be reduced and that is accomplished with back-propagation. It is a process of adjusting and updating the weights until the error is

reduced to the minimum. Figure 2.5 shows the architecture of a Perceptron [19].

$$output = \begin{cases} 0, & y < 0 \\ 1, & y \geq 0 \end{cases} \tag{2.9}$$

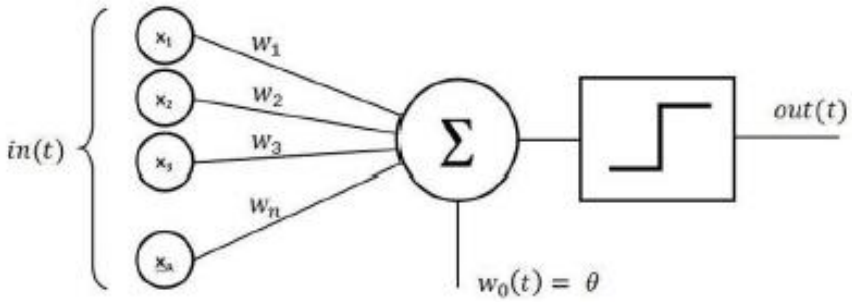

Figure 2.5 Architecture of a Perceptron [19]

Although the creation of ANN was revolutionary but it was faced with many shortcomings and drawbacks which needed attention. Minsky and Papert discussed the limitations of a perceptron and stated that by making the architecture of perceptron more complex its performance would still not improve. However these limitations have been gradually overcome with the increase in computational power of hardware and CPUs and also with the increasing popularity of the concept known as Big Data. The increase in the availability of data, data analytics and data science acted as the turning point in Machine learning which eventually gave rise to Deep Learning, leaving ANNs behind.

2.7 Deep Learning

Just like machine learning, deep learning also come under the umbrella of artificial intelligence. Here the word deep denotes the several hidden layers available in the network. It also follows the basic ANN structure but uses several layers to transform the data and for extraction of useful features from the data given as input [20]. The deep learning systems do not need extracted features given to them explicitly in fact they are designed to extract useful features on their own. For feature extraction, the model can use lower or earlier layers that focus on features like shapes, edges, blobs, corners or can use higher or later layers that are concerned about extraction of specific features such as small objects, faces or letters etc.

As we go deeper in to a deep learning model's architecture, the layers transform the data into a more abstract form. For example when an image is given as input to the network, the matrix of pixels is the activation of neurons in the input layer. The next set of earlier layers extract generic features such as edges or corners, the second set of earlier layer may arrange the features into useful form such as small objects or contours. The later layers could be responsible for detecting more specific features such as eyes, lips or nose and the final layer will classify the entire object which was given as input to the network.

2.8 Convolutional Neural Networks

Introduced in the 1980s, Convolutional Neural Network (CNN) is a class of deep neural network. CNNs are established and reliable form of Perceptron having several layers. These 'multi-layer perceptron' means a fully connected network in which a neuron of one layer is linked with all neurons of the layer next to it. Although the structure of fully connected networks may seem complex but it is very useful when it comes to avoiding over-fitting, a process in which the model is unable to generalize the data. This happens when a model learns the given training data alongside noise with such great detail that when an unseen data i.e. testing data is given to it, the model performs poorly. This means that the model will only perform well when it is given the data it has already seen otherwise not.

The input of CNNs is mostly visual data such as images, where they are used for classification, detection or recognition of these images. Based upon the architecture of a CNN, they are also known as shift-invariant as they involve sharing of weights and happen to be translation invariant as well. Translation-invariant means no matter how the inputs of the input is translated or moved, the CNN will still give the same classification or recognition results as it did on the original image.

A CNN is composed of following components: Convolutional layers, weights, receptive fields. Pooling layers, Dropout, ReLU layer for activation, Stochastic Gradient Descent, fully connected layers and loss layer.

2.8.1 Convolutional Layer

A CNN comprises of several convolutional layers hence the name Convolutional Neural

Network. These layers make use of Kernels to apply convolutional operation on the images. Kernels are also called as filters as they filter out any unnecessary information which is to be passed to the subsequent layers. The convolutional operation requires two things, one is the matrix of pixel values which represents the input image and the second thing is a kernel. When these kernels are applied on to the input matrix, convolution operation takes place and this result in a set of features like edges, corners or small objects like an eye.

A convolutional layer is given a tensor object as an input. A tensor is basically a matrix which is made up of the following:

$$(number\ of\ samples)\ X\ (sample\ width)\ X\ (sample\ height) X\ (sample\ depth) \quad (2.10)$$

Here sample represents an image and depth shows the no. of channels of the image weather the image is a grey-scale or a colored one. If the image is a colored one then its depth for each pixel will be 3, representing the 3 channels i.e. RGB which stands for Red, Green and Blue. If the image is grey-scale then its channel or depth will be 1. In later layers, the abstract color representations may act as RGB color channels, increasing the number of channels by more than 3 where each pixel give valuable information regarding the transformed image.

After convolution operations are applied on the input image or a tensor to be specific, it outputs a feature map. A feature map has the following shape:

$$(number\ of\ samples)\ X\ (feature\ map\ width)\ X\ (feature\ map\ height) X\ (feature\ map\ depth) \quad (2.11)$$

Now we discuss the attributes of a convolution layer which are as follows:

- A convolution kernel or a filter having shape *(filter width) x (filter height).* If the input image is a colored image then the kernel's shape will be 3x3x3 where 3x3 is the width and height where 3 shows the number of color channels. Whereas if the image is a grey scale one then kernel shape will be 3x3x1 where 3x3 shows the width and height and 1 depicts the color channel.
- Number of channels of the input and output.

- Depth i.e. number of channels of the convolution filter should match the depth or number of channels of the input image.

Figure 2.6 depicts the movement of a kernel over an image [21].

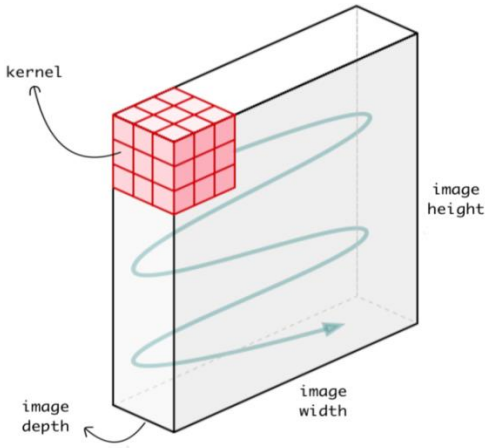

Figure 2.6 Movement of a Convolutional Filter [21]

The number of generated feature maps is equal to the number of kernels applied on to the input data. For example if 7 filters are applied on the input image then we will have 7 feature maps. The feature maps are different from each other and helpful in learning new features. Figure 2.7 shows convolution of an input image using an edge detector.

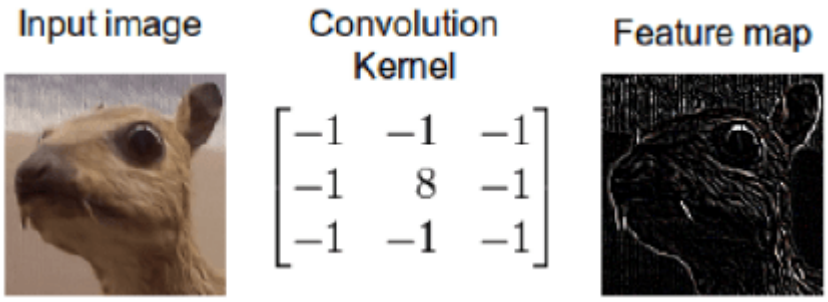

Figure 2.7 Convolution using Edge Detector Kernels [22]

The depth of the image and that of the kernel must be equal e.g. if the depth of RGB image is 3 then the kernel's depth must be 3 as well. The convolutional operation is applied on all

the channels of the image and the results of these operations are added together along with a bias term, this generates a compacted feature map with depth of 1 i.e. one channel. Output of the convolutional operations can be attained in the following two ways:

- Valid padding: this means no padding. If no padding is used then the feature map produced has a reduced size compared to the input.
- Half padding: this is also known as same padding. This is done so that the output feature map has the same dimension as the original input and it is not reduced. Figure 2.8 represents a half padding operation.

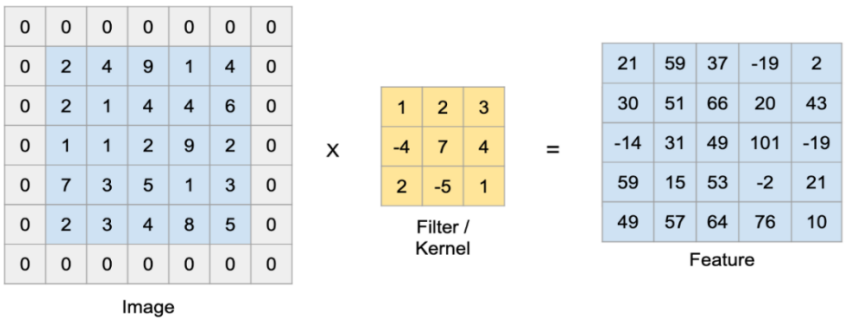

Figure 2.8 Half Padding [21]

In figure 2.8 it can be seen that when a kernel having size 3x3 is applied onto a 5x5 image with half padding, the size of the output feature map is the same as the input i.e. 5x5. However if no padding was done then the dimension of the feature map will reduce to 3x3.

2.8.2 Weights

Weights grasp a huge importance in a neural network. A weight is basically a number, changed or updated in back-propagation process to reduce the error. Weight-sharing or in simple words reusing similar weights is a very useful part of a neural network as the translation-invariant property of an input object can be explored with this. In simple words, no matter what the location, the extracted features always hold a significant meaning. For example in an image of a cat, the cat's position is irrelevant as the network will still be able to tell, to which class the object in the image belongs. Figure 2.9 shows the translation-invariant property of an image.

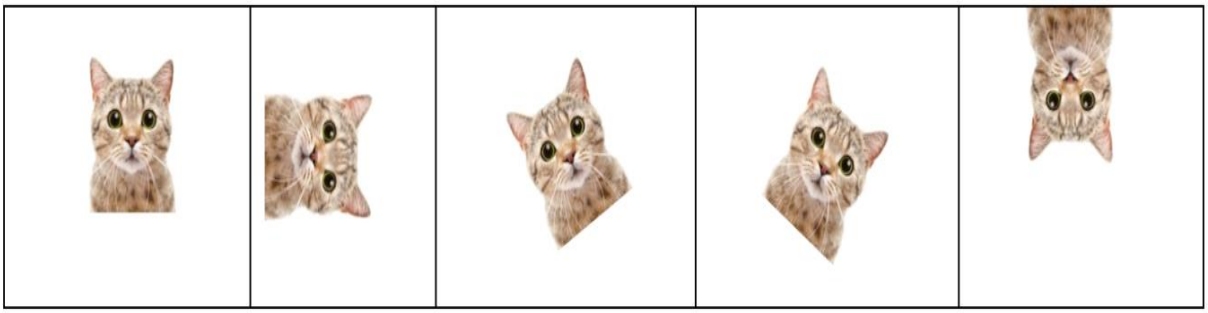

Figure 2.9 Translation Invariance of a Cat Image [22]

The reason behind the translation invariant property of the input is that after a convolution operation takes place, a plane or a feature map is generated [22] as a same filter is used through the whole input. Different weights of the kernels allow them to extract different kinds of features. Resultantly, the extracted feature, regardless of its position belongs to the entire feature map. During the convolution process, the weights of the kernel remain the same.

2.8.3 Local Receptive Field

Usually the input given to network is of very high dimension which makes it hard to work on, specifically, to link all the nodes of a layer to the nodes of the next hidden layer. So for that reason, a limited area is chosen at a time for convolution with the filter. This limited or segmented region is called as local receptive field of a particular neuron. The filter size and the receptive field are always equal. In figure 2.10 we can see that the input has a size of 32x32 and 3 channels and the applied filter has size 5x5x3 (the filter and input's depth should match) thus each node of the particular convolutional layer will be linked with a 5x5x3 region, creating 75 connections +1 bias (5*5*3=75+1) in total.

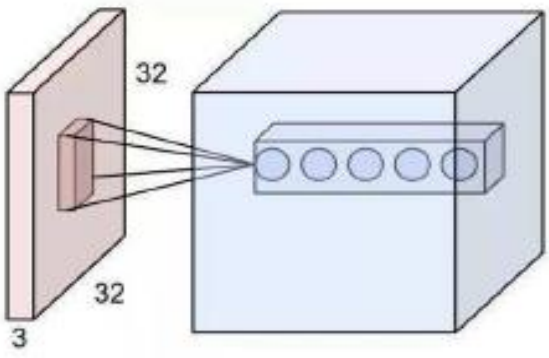

Figure 2.10 Local Receptive Field of 5x5x 3 Dimensions [22]

2.8.4 Pooling Layer

In most networks each convolutional layer is followed by a pooling layer. Pooling layers are extremely important component of a convolutional network. These set of layers reduce the dimensionality of the data making computation easy. The data is down sampled in a nonlinear fashion such that the outputs from multiple neurons of a layer are reduced to a single unit for the next layer. Generally there are two types of pooling: Global pooling which focuses on all the nodes of a convolutional layer and Local pooling which considers segmented areas, normally a window of size 2x2 [23]. The pooling process helps in extracting high level features which are position wise and rotationally invariant. There are two methods of doing pooling operations as shown in figure 2.11. These methods are explained below:

- **Max pooling**

As the name suggests, this methods picks up the largest value from the area of the image covered by the kernel. This method is highly preferred over the other type of pooling as it not only reduces dimensionality but noise as well.

- **Average pooling**

Also known as mean pooling, this method calculates the average or mean of the values in the area the kernel has covered. This method is useful in reducing the dimensionality.

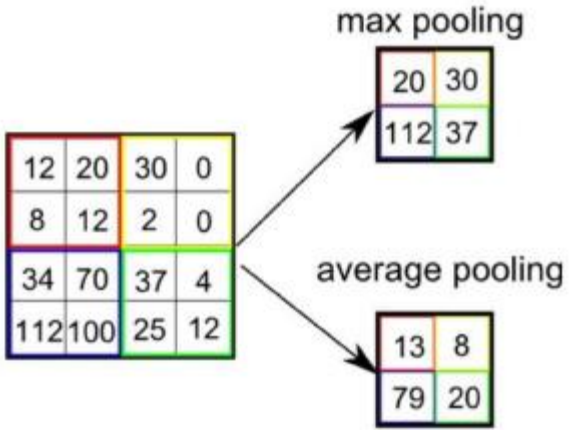

Figure 2.11 Max and Average Pooling Operations [23]

The convolution operations followed by pooling operations extract useful feature representations of the entire image. Increasing the number of convolution and pooling layers results in more detailed features but also costing more computationally. However pooling operations play a significant role in reducing over fitting.

2.8.5 Dropout

Sometimes a neural network tends to over-fit. To avoid this problem we need regularization which can be achieved using dropout layers. In some cases, the model gets trained on the training data so well, learning about every little detail such that it fails to generalize on the test data. This is known as over-fitting and to avoid that we use dropout. With dropout we basically 'drop' the neurons with a high activation value, lowering their impact to the minimum. Dropping some nodes allows other nodes to learn the features in a better way automatically.

2.8.6 Stochastic Gradient Descent

Gradient decent is an optimization process that finds out the values for parameters in order to diminish the cost function. One method for reducing the error is known as 'Batch Gradient Decent' that considers all training sample in one epoch and updates their parameters i.e. weights and biases. However this is a very slow and costly process when the number of training sample is huge. Thus we use a better method known as 'Stochastic Gradient Decent' or SGD which picks up the parameters of a single training sample and

updates them in a single forward pass. This approach is useful when our train data is big. A third approach is called as Mini batch Gradient Decent which takes in to consideration a small number of samples in a forward pass and then fixes the parameters to reduce the cost function.

2.8.7 Batch Normalization

Batch normalization is a technique of standardizing or scaling the inputs which as a result improves the overall performance of the network. During the training process as the weights gets updated, if a weight has an extremely large value then this will cause an imbalance in the entire network. For this problem, the solution is to use batch normalization which can be applied to selected layers; the output achieved after activation function from these layers is normalized. This technique is useful in enabling the layers to learn more independently adding some new learnable parameters which ensure that the weights do not have extremely high or low values that can cause imbalance. This practice speeds up the training process [24]. As compared to Dropout, this technique is better as it causes less information loss. For optimum performance by the network, both techniques should be used together.

2.8.8 ReLu Layer

In an artificial neural network's architecture each node in a layer must be applied with a function known as an 'Activation function' which resolves if node will fire or not. For long, time activation function namely 'Step function' was used but due to its binary nature it misclassified many input values and couldn't give good approximation for error. A better approach is using a sigmoid function which reduces the output value between 0 and1. Sigmoid function is proven to be more suitable for small datasets however in case of huge networks sigmoid function is not appropriately scaled [25] as it leads to huge number and high computational costs [26]. Another noteworthy issue relevant to sigmoid function was 'Vanishing Gradient' problem. Such high value of gradient prevents the learning process of the network.

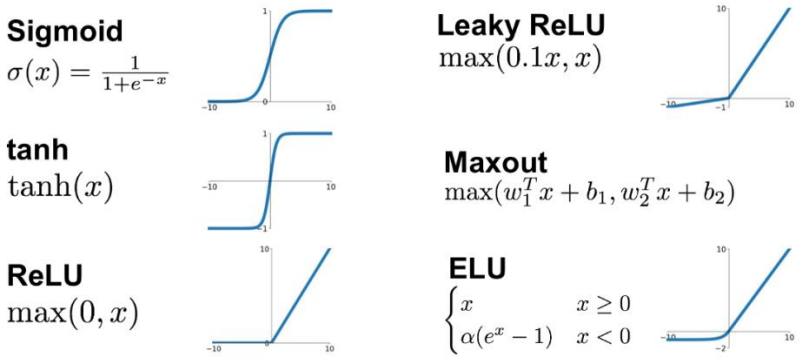

Figure 2.12 Different types of Activation Functions [25]

As compared to Sigmoid, ReLU is an activation function which didn't undergo the vanishing gradient problem. ReLU which stands for Rectified Linear Unit provides good approximation of error and costs less computationally. Krizhevsky et al. [27] explained in their research that ReLU requires less number of epochs to converge by using SGD by a factor 6. However, ReLU also has its short comings and that is many nodes fail to activate due to distribution of input which is below zero. Figure 2.13 shows a ReLU activation function. ReLU is presented as:

$$f(x) = \max(0, x) \tag{2.12}$$

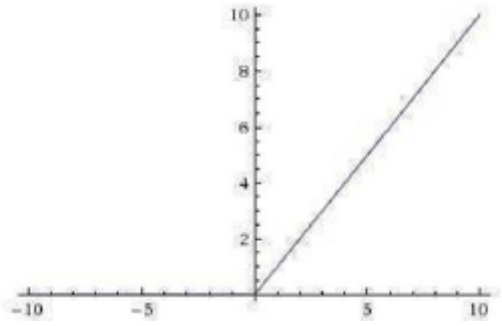

Figure 2.13 Activation Function: ReLU [27]

For training of Deep Neural Network models researchers need a GPU which stands for Graphic Processing Unit. The time taken to train a model is dependent on many things such as the network's structure or topology and the input data etc. However a GPU reduces the

time required to train a model as compared to a CPU [28]. Even with a GPU, a network still may take a lot of time to train but it's better to use it as compared to a CPU as it trains 10 times faster [29]. The reason a GPU works better is due to its powerful architecture having thousands of cores that work in parallel to each other. This manages multiple tasks in a single time synchronously. However many researchers still use CPU due to some limitations but a GPU is always recommended.

2.8.9 Fully Connected Layer

The fully connected or FC layers makes the last few layers of the neural network. This layer is required to make the final classifying decisions. FC layers make high level decisions upon getting input from convolution and pooling layers. Each node in the FC layer is connected to all the nodes of the layer preceding it. A fully connected layer's receptive field is the entire previous layer.

2.8.10 Loss Layer

Loss layer is the last layer of a network which gives the final classification output labels and classification error. The accuracy shows how accurately a model performed and the error represents the number of wrongly predicted labels. A predicted label is considered incorrect or misclassified if it is different from the ground truth i.e. actual label. There are various kinds of Loss functions; the most commonly used is Softmax loss function. In a nut shell or a softmax function or any other loss function estimates how far a predicted value is from the actual value.

2.9 Combination of Traditional and Deep Learning Techniques

So far we have discussed two methodologies for feature extraction and classification. The first methodology makes use of hand crafted features and classifies via a model. The second approach is deep learning based that makes use of a network such as a CNN that has the ability to extract useful features and then performs classification after learning from them. We have deployed a combined approach that makes use of both methodologies i.e. using a CNN for extraction of useful features and then using a traditional model such as SVM or Decision Tree for classification. This approach is less mainstream yet extremely useful for exploring the power of DL especially when the dataset is relatively small. It takes less time

and effort in training a network from scratch as it needs just a single pass from the training data and is most useful if we don't own a GPU. This involves using a pre-trained CNN which is trained on a popular dataset; ImageNet.

CNN's have the ability to extract complex features that represent the entire image. As CNN is a combination of two parts; feature extraction and classification. The Feature extraction part can use either earlier layers that extract generic features such as shapes, blobs, edges etc. or later layers that extract more specific features like contours or small objects. Instead of using classification part of a CNN, which usually uses a softmax activation function, these extracted features are fed to a classifier like SVM or Decision Tree for classification; this is one form of using a previously trained network. Another approach known as *Transfer Learning* is the ability of a system to apply the knowledge gained in a previous task to a newly assigned task for the purpose of object detection and classification etc. [30]. In this approach the pre-trained net is fine-tuned in accordance to the experimental setup. Nowadays, most of the networks are trained on ImageNet database [31]. With ImageNet, these networks are trained on more than a million images and have learned to classify up to 1000 classes including various animals, objects and many shapes. Studies show that using a pre-trained network is beneficial in achieving quality results [32]. However, if the architecture of the network is too complex, the extracted features may not be suitable for a comparatively simpler task. Similarly, if the target dataset is small and similar to the dataset on which the network is pre trained, network fine-tuning could cause over fitting [32]. There are numerous factors that should be considered when choosing a methodology for recognition or classification problems. The size of the dataset and how similar it is to the original dataset (the dataset on which the model was trained on) plays are very crucial role in shaping the end results. There are a few scenarios to be considered:

- ***Dataset is small and similar to the original dataset*:** in this case more specific features from deeper layers are more useful.
- ***Dataset is small and different from the original dataset**:* in this case generic features from earlier layers are more useful for the task.
- ***Dataset is big and similar to the original dataset:*** in this case fine tuning a CNN is a good option as the target dataset is expected to have more versatile samples.

- ***Dataset is big and different from the original dataset:*** in this training a CNN form scratch is a good option.

Chapter 3

LITERATURE REVIEW

CHAPTER 3: LITERATURE REVIEW

An emotion, either displayed through facial expressions or added in speech can deliver the message efficiently and in a better way. In research, facial emotion recognition, either done through analysis of video data or audio data or both, is a very active area in the field of computer vision. Researchers have made many contributions in this field to enable technology help humans to communicate proficiently. The technology these days is able to decode the facial expression being delivered and discard any emotion that may cause confusion. The ability of human-centered computing interfaces, to comprehend, understand the meaning of and to react to human expressions is one of the most important and challenging jobs [33, 34]. Current studies, in field of emotion recognition have mainly focused on adult facial expressions. Several traditional and deep learning approaches have been deployed for recognition of facial emotions using databases containing adult faces, as there is ample data on adult emotions. However there is a lack in literature on FER in children due to many reasons; one of them is the dearth of children emotional datasets. State of the art approaches have not yet been trained with standardized databases that provide 100% "unbiased" or spontaneous and natural emotions of children [35]. The existing databases also offer unevenness in pose, lightning and variations in expression by subjects across different areas of the world [36]. Similar issue with publicly available databases is that it only contains adult facial expressions and ignores the children's [37, 38]. Therefore, limited work has been done on spontaneous child facial expression recognition [39]. In this section we have discussed the contributions made in the field of emotion recognition in adults and children.

3.1 Children Datasets

In order to draw accurate conclusions for understanding children's way of giving expressions it is important to consider many factors that differentiate datasets from each other, for example, number of participants or volunteers, modality (audio or video), age group, elicitation techniques (posed or spontaneous), the number of categories and the environment in which the participants gave the emotion.

In NIMH Child Emotional Faces Picture Set (NIMH-ChEFS) [40], the considered emotions are ‘anger’, ‘fear’, ’happy’ and ‘sad’ with averted and directed gaze. The age group of volunteers in this database is between 10-17 years. The Dartmouth Database [41] of Children’s faces considers 6 basic universal emotions expressed by children having age about 6 to 16 years. All the children are Caucasian, 40 of them are boys and 40 are girls. Child Affective Facial Expression (CAFE) [42] is an image dataset, having 1,192 frames in total. This dataset considers 7 emotions i.e. 6 basic and one neutral. The age group of the culturally diverse volunteers is around 2-8 years. The Radboud Face Database (RaFD) [43] is a picture dataset that considers eight emotional states including anger, disgust, fear, happiness, sadness, surprise, dislike and neutral presented by 67 children. These children include Caucasian children of age group between 8-12 years (the dataset also includes Caucasian males and females and Moroccan Dutch males). Children showed the emotions with three different gaze directions, and five camera angles were used simultaneously to take pictures. The EmoReact [44] is a multimodal dataset containing emotions of children having age group of 4-14 years. In total there are 1102 audio and video clips in the dataset and 17 emotional states. This includes six universal emotions and nine complex emotions, neutral, valence, uncertainty and frustration. The JEMImE dataset [45] falls in the posed category where emotions were elicited via two methods: 1) children were asked to give or pose for a certain emotion, 2) copying an avatar that produces certain emotions. This dataset contains 3768 videos of 3 seconds each. The number of volunteers is 157 having ages around 6-12 years. This dataset contains 4 different kinds of emotions: anger, happiness, neutral and sadness. **LIRIS-CSE** [46] is an emotional dataset for children that contain 208 videos of culturally varied children. The database has been created; involving videos of 12 children, 5 are male and 7 female of age group 6 to 12 years (average 7.3 years).

In order to assess and benchmark various algorithms designed for facial emotion recognition problems, it is vital that there exist standardized databases. By making use of such databases in experimentation, researches are able to find the short comings present in various FER algorithms. The existing emotional datasets for children displays exaggerated emotions of children with proper camera positions and illumination setups, which is different compared to the spontaneous expressions children give in real life. LIRIS-CSE dataset presents 100% spontaneous emotions of children which can be used as a benchmark for the task of Facial

Emotion Recognition (FER) in children. Table 3.1 represents the comparison of various emotional datasets for children.

Table 3.1: Comparison of Various Emotional Datasets for Children

Dataset	No. of children	Age	Gender	Modality	No. of Videos/ Images	No.of Label	Elicitation	Setting	Labels
NIMH-ChEFS [40]	60	10-17	66% F	Image	-/482	5	Posed	Lab	Category
DartMouth [41]	80	6-16	50% F 50% M	Image	-/640	8	Posed	Lab	Category+ Intensity
CAFÉ [42]	154	2-8	58% F	Image	-/1192	5	Posed	Lab	Category
Radbound [43]	10	8-12	60% F	Image	-/80	8	Posed	Lab	Category
EmoReact [44]	63	4-14	51% F	Audio/Vid	1102 Videos	17	Spontaneous	Unconstrained	Category
JEMIme [45]	157	6-12	48% F	Audio/Vid	3768 Videos	4	Posed + limit	Unconstrained	Category + Quality
LIRIS-CSE [46]	12	6-12	58% F	Video	208/26k	6	Spontaneous	Unconstrained	Category

3.2 Advancements in Emotion Recognition in Children

In order to draw accurate conclusions for understanding children's way of giving expressions, researchers have conducted studies on facial emotion recognition in children by using the existing emotion databases. These studies have been conducted through deep analysis of a child's psyche and way of interpreting facial emotions. This is accompanied with experimental evaluations using traditional and deep learning methodologies such as Convolutional Neural Networks (CNN). The traditional feature extraction approach is often compared with CNN based feature extractors. The difference between the two is that

traditional feature base techniques focus on using hand crafted features, whereas CNNs can learn such features automatically during training. These studies not only put forth the significance but the weaknesses that need to be dealt with in this field.

Khan et al [46] proposed a novel emotion dataset for children, namely LIRIS-CSE which considers six emotions; happy, sad, anger, disgust and fear. They used a pre-trained VGG16 with transfer learning approach. The last fully connected layer of their VGG16 architecture was interchanged with a dense layer with five outputs. The authors performed image based classification only using 80% of the frames for training and 10% of frames for validation process. Their system achieved an average accuracy of 75%. Uddin *et al* [47] focused on recognition of dynamic facial expressions from videos by deploying their proposed approach on four datasets Oulu-CASIA, AFEW, CK+, and LIRIS-CSE. They introduced a novel approach based on Spark distributed computing environment, for efficiently processing the video data. They presented a dynamic feature descriptor known as LDSP-TOP that gives an established description of face dynamics. Moreover, for capturing additional features of a face, they designed a 1D CNN that consisted of residual connections. Lastly, for learning spatio-temporal features, they used a long short term memory auto-encoder. Their proposed methodology performed remarkably well on three of the above mentioned datasets, by achieving accuracies of 86.6%, 98.6%, and 84.2% on Oulu-CASIA, CK+, and LIRIS-CSE datasets respectively. On AFEW dataset, their proposed method did not perform well achieving a classification accuracy of 50.3%. C. Florea *et al* [48] presented a method which used labeled and unlabeled data for the task of facial expression recognition by combining transfer learning and semi-supervised learning with the proposed framework known as Annealed Label Transfer (ALT). They made use of AlexNet and VGG16 networks in their work as well. They setup four scenarios for evaluation, the first two are facial expressions in the wild and the other two are FER in children along with anxiety based expressions in the wild. They used RAF-DB and FER+ datasets (for labeled data) and MegaFace dataset (for un-labeled data) for first two scenarios. They reported the performance of AlexNet with ALT on CAFÉ dataset in a purely supervised manner, where it achieved 99.29% accuracy. To summarize, ALT used learned knowledge and transferred it to a labeled dataset in the wild to unlabeled datasets for generation of pseudo labels. Zhao *et al* [49] proposed a Mobile Edge Computing (MEC)

based hierarchical emotion recognition model. Their system used a pre-trained feature extraction and localization module on remote cloud. The difficulty imposed by environmental issues has been addressed by a mechanism called as perturbation aware defense. Using proposed MEC based hierarchical emotion recognition model with VGG16 in conjunction with localization module, they achieved 95.67% accuracy on LIRIS-CSE dataset.

Alejandro *et al* [50] considered six basic emotions for classification of facial expressions in children using NAO robot. In their study they compared AFFDEX SDK and a CNN with Viola Jones which was trained on AffectNet and tuned on NIMH-ChEF dataset with transfer learning approach. Moreover they tested their system by comparison on another dataset, CAFÉ. Lastly, they compared both systems with NAO robot using subsets of AMFED and EmoReact datasets. Guiping Yu [51] proposed a face and emotion recognition multi-modal system based on deep learning methods for monitoring the emotions of preschool children. Their model combined face, context and action for emotion recognition. They used FLAW, FER2013 and CK+ datasets for face recognition. Their work specified that face tracking algorithms can make the system more resilient to complications such as pose, angle and dynamic blur. Thus they proposed an algorithmic model that detects face blur. They combined LSTM and CNN model algorithms. Resultantly, they used VGG19 network in their proposed system. Weiqing *et al* [52] combined the online courses platforms with a deep learning CNN based model for constructing a framework for analyzing student's emotions during online classes. The goal of their system is that teachers change their teaching style by reading the results presented in the form of a histogram. Their model was trained on CK+, Jaffe and FER2013 datasets while considering eight basic emotions. They also used online augmentation techniques to increase number of images in the dataset. The proposed framework showed good performance and proved that it is likely to do well in practical applications. Megan *et al* [53] showed the benefit of using transfer learning for learning the general facial expressions from adult faces for performing multi-class classification on children's facial expressions. In their research they deployed closed-mouth subset of CAFÉ and CK+ dataset. Initially, they performed preprocessing on the images of the datasets. They designed a CNN model and trained it for classifying six basic emotions including neutral emotion in children and adults. Amir *et al* [54] developed an affect

recognition system for young children using a deep convolutional neural network. Their DCNN is an emotion recognition prototype that effectively extracts refined facial features. For training of their model they used enhanced FER dataset (FER+) which consists of 35887 facial images, considering eight emotions namely; anger, disgust, fear, happiness, neutral, surprise, sadness and contempt. They tested their model with kindergartner's children as they naturally interact with each other in a classroom. The prototype achieved an effective prediction accuracy of 93%.

Awatramani *et al* [55] focused on emotion recognition ability of children with autism spectrum disorder. In their work, they explained how this disorder can pose behavioral challenges for children and how they suffer from inability to recognize emotions. They implemented a basic CNN for teaching children with ASD to recognize emotions. Their system achieved an accuracy of 67.50% on an existing dataset.

Qing *et al* [56] performed facial expression recognition on babies. They introduced a novel dataset namely BabyExp. This dataset contains 12000 images of babies having ages around 1-2 years. The dataset presents three expressions happy, normal and sad. This dataset acts as a benchmark for facial expression recognition on babies and paves way for further research. They tested their dataset by proposing a feature guided CNN, where they also introduced a new loss function named; distance loss for optimization of inter class distance. Their methodology accomplishes a promising accuracy of 87.90% on BabyExp dataset.

Arnaud *et al* **[**45**]** focused on understanding facial emotions especially in children by proposing classification and regression systems. It was specified that existing databases for FER are limited in terms of number of participants, recording environment or simply the nature of annotation. They created their own children's FE dataset, JEMImE, which included many modalities. The modalities are marked with categorical and qualitative FEs. Furthermore, for classification and regression they deployed a FER pipeline, which uses a random forest trained on mixture of geometric and appearance features. The results proved that random forest models which are trained on JEMImE dataset generalize much better on children's data, for classification and regression.

Due to CNNs amazing ability to perform well and give robust results even for complex

tasks, many researchers are inclined to use them to image feature extraction, classification and recognition problems.

3.3 Emotion Recognition Based on Classical Methods

For emotion recognition, literature provides two kinds of approaches. One is classical or traditional approach and the other is machine learning based approaches. The two are often compared where the difference is that traditional feature base techniques make use of hand crafted features, whereas CNNs can learn such features by training. In traditional approach, classifiers such as MLP (Multi-layer Perceptron Model), SVM (Support Vector Machines) and KNN (k-Nearest Neighbors) makes use of handcrafted features such as Texture and Face Landmark features, Eigen vectors, HoG (Histogram of Oriented Gradients) and Gradient Feature Mapping etc. which are extracted by deploying methodologies like LBP (Local Binary Patterns), Gabor filters, Eigen Faces, LDA (Linear Discriminant Analysis), and PCA (Principal Component Analysis). Moreover researchers have used uni-model or bi-model systems for emotion recognition. Uni-model systems considered either visual data or audio data but not both. The bi-model on the other hand considers both modalities (audio, video/image) even sometimes textual data. The combination of these modalities has been tremendously improved [57, 58].

In this section we discuss various uni and bi model emotion recognition systems based on classical methods.

Shan *et al.* [59] used LBP (Local Binary Patterns) for estimation of facial emotions. In his person- independent FER system multiple machine learning models were put to use. Seven emotion states were considered from datasets such JAFEE and MMI. LBP showed promising results for emotion recognition system. Moreover evaluation of discriminative and useful LBP features was done by using Adaboost technique. The boosted LBP features were inputted to a SVM and its classification results were compared with regular LBP features. Eventually, SVM with RBF kernel on boosted LBP features gave optimum results.

Dhall *et al.* [60] used Pyramid Histogram of Gradients (PHOG) with Local Phase Quantization (LPQ) for evaluation of visual features. Two datasets, SSPNET and GEMEP-

FERA were used for testing and classification was done using SVM and largest margin nearest neighbor classifiers.

Shaukat *et al.* [61] used a combination of combination of Scale Invariant Features Transform (SIFT), Discrete Cosine Transform (DCT) and Gabor wavelets for evaluation of features for his facial emotion recognition system. In his experiment pre-processing stage involved detection of face area which was set to uniform size. This process was done for every image in the JAFEE dataset. SVM with radial basis kernel was given a concatenated set of extracted features. Their approach gave favorable and promising experimentation results.

Oussalah *et al.* [62] created a fuzzy facial emotion recognition system. In his experiment several pre-processing stage improvements were involved that included technique for face tracking such as a combination of CAM-shift, PCA, skin area detection using elliptical model and adaptive-thresh holding. TFEID dataset was used for experimentation and gave promising results when comparison with Bayes' classifier.

Khan *et al.* [63] performed facial feature analysis by developing pyramids of LBP. This variant sets Local Binary Patterns in a pyramidal form so that the face region is split into finer sub regions repetitively by doubling the division at each iteration level. For a specific expression, the most vital face regions were determined by conducting psyco visual experiment. Results show that specific face area is more vita for a specific emotion. For classification SVM, Random forest, 2 nearest neighbor (2NN), C4.5 Decision tree and Naïve Bayes Classifiers were used and 2NN showed best results among the other classifiers.

Wang et al. [64] deployed kernel based methods in their research for identification of emotions from audio-video data. A 'Hamming window' with size 512 points and between the window frames an overlap of 50% was used. Audio features were extracted from the frames such as 'MFCC' coefficients, power and pitch. Also, 'Planer envelop' method with tuned parameters was used for detecting face region in 'HSC color space'. On the detected face area Gabor filters with filter bank having eight orientation and five scales was used for feature extraction from these face areas. Down sampled Gabor coefficients and 'Principal Component Analysis' were applied onto the feature set to reduce its dimensionality and

computation complications. In order to map features of a multi-modal into a only one subspace, ‘Kernel Matrix Fusion (KMF)’ is deployed. For all modalities, Kernel matrices are separately fashioned, moreover unsupervised kernel PCA (KPCA), transformed kernel PCA features and supervised kernel Discriminant Analysis (KDA) are given to the HMM classifier. All the mentioned methodologies were applied on RML dataset, which proved to do better than other methodologies such as CCA. KCFA, score level and feature level fusion.

Haq et al. [65] created a dual-model system for emotion recognition on SAVEE dataset. Features were extracted separately for both audio-video data. Fusion process was also tested for both feature and ‘score level stage’. From the audio data, features like pitch, energy, MFCC and duration were extracted. When it comes to visual data, features on the basis of ‘2d marker coordinates’ positions on face area were extracted. After this techniques such as Plus 1-Take Away r algorithm based upon Mahalanobis and Bhattacharyya distance having selection on basis of KL-divergence were deployed for selecting distinct features. Moreover PCA and LDA were used for dimension reduction of selected features and were given to a Gaussian classifier. Strategies such as feature level and score level fusion were tested and high accuracies were achieved, along with visual recognition and decision level fusion approaches giving better results compared to audio-recognition and audio-feature level fusion in respective order.

Rashid et al. [66] in their research made use of spatial temporal features which were extracted from dimensionally reduced visual streams using PCA. For selecting audio feature representatives MFCC and prosodic features were identified. For both kinds of features in Euclidean space a codebook was created this is after PCA was applied. SVM was used for classification of multiple emotional classes. The final classification results were calculated based upon the predictions from all the classifiers by Bayes Sum Rule (BSR). It was proved that visual features performed better than audio features but fusion of both features gave the best results.

Table 3.2 Summary of Recent Multi Model Emotion Recognition Systems

Author	Dataset	Modality	Features Extracted	Classification strategy
Shan et al. [59]	JAFEE MMI	Uni model	V: Boosted LBP	V: SVM (RBF)
Dhall et al. [60]	SSPNET GEMEP-FERA	Uni-modal	V: PHOG with LPQ	V: SVM, Largest Margin
Shaukat et al. [61]	JAFEE	Uni-modal	V: Scale Invariant Feature Transform (SIFT), Gabor Wavelets, DCT	V: SVM (RBF)
Oussalah et al. [62]	TFEID	Uni-modal	V: CAM shift ,PCA, Adaptive thresh holding	V: Fuzzy recognition model
Khan et al. [63]	Cohn-Kanade MMI	Uni-modal	V: PLBP (Pyramids of LBP)	V: SVM, 2NN, RF, DT, NB
Wang et al. [64]	RML	Bi-modal	A: MFCC, Pitch and power V: Gabor Filters with PCA	AV: HMM(KPCA,KLDA input)
Haq et al. [65]	SAVEE	Bi-modal	A: MFCC, pitch, energy, duration V: 2D marker Coordinates	AV: Gaussian Classifier (PCA input)
Rashid et al. [66]	eNTERFACE'05	Bi-modal	A: MFCC, prosodic V: Spatial-temporal	A: SVM V: SVM AV: BSR

3.4 Machine Learning Based Emotion Recognition

Zhalehpour et al. [67] offered a paper BAUM-1s which presents a dual modality dataset which takes into consideration six basic emotional classes. For this dataset, 31 volunteers participated of which 17 were female. Along with BAUM-1, Zhalehpour et al. also deployed eNTERFACE dataset in their system and compared the results of two datasets. Techniques such as Maximum Dissimilarity (MAX-DIST) were deployed for selecting peak frame from the video input. The peak frame was selected on the basis of maximum distance or dissimilarity. Furthermore visual features such as Linear Phase Quantization (LPQ) and Patterns Oriented Edge Magnitude (POEM) were extracted from these datasets. LPQ and LBP are similar in terms of local histograms they generate upon which feature vectors are based. It is presented that LPQ outperforms LBP. Moreover the POEM features were considered as they are more robust and resistant to pixel intensities and consider gradient

magnitudes only, providing both, local and global information, compared to LBP on the other hand, which only gives local information. Additionally audio features were also considered such as 'Mel-frequency Cepstral Coefficients (MFCC)' was created using 12 order filters and 'Relative Spectral Feature based on Perceptual Linear Prediction (RASTA-PLP)' was created 20 order filters along with a 50% overlap ratio and a 25 window size. For the coefficients already acquired, 1^{st} and 2^{nd} order derivative were calculated along with many statistic functions were applied so that the finalized features vector has 675 distinctive features in it, further which was used for classification using Support Vector Machine (SVM). In their research SVM was used for both audio and video features. In order to avoid the curse of dimensionality affect linear kernel was deployed for video features and radial basis kernel was used for classification of audio features. Fusion was done on the outputs given by both classifier with different kernels based upon weighted product rule, the given confidence values for a video sequence from each classifier are multiplied and that label is selected which has the maximum product value.

Huibin Li et al. [68] presented an original CNN which works with deep fusion (DF-CNN). In their research, facial emotion recognition was done by considering 2D and 3D visual data. Their deep fusion CNN comprises of 3 sub units, first unit is for feature extraction, second unit is for fusion and last unit uses softmax layer for classification. For representation of facial attributes the 3D facial image was divided into 6 2D maps. Those attributes include normal and texture mapping, curvature map and geometric shape of the face. After this, these attributes are joined and input to the DF-CNN for fusion and extraction of features. As a result 32-dimension facial representation vector was returned. For prediction of emotions two methods were selected; one linear SVM which utilizes 32-D vector consisting of fused features, second Softmax classification which makes use of 6-D feature vector that represents six emotional states. The thing that makes their network different from other 3D networks is that it uses combination of feature learning and fusion learning in a single network. In their research they have used three datasets Bosphorus Subset, BU-3DFE Subset I, BU-3DFE Subset. Only 3D images are used in Bosphorus subset.

Kim et al. [69] performed emotion recognition, considering two views of an image, one the

object and the other background. They explained that emotion recognition is based upon the fact that each image holds important information presented in the form of facial expression. In order to improve an emotion recognition system's performance they have combined object information with background information. The input given to the deep neural network is a set of features extracted not only from object but background as well. In their research they deployed valence and arousal model for prediction of emotions by the network. Moreover they extracted four different kind of features; color, semantic, local and object features. These features are normalized to [0, 1]. Kim et al. created their own dataset for which images were taken from Flicker. For the database the images were searched with keywords i.e. basic emotional states such as angry, happy, fear etc. In the beginning 20,000 plus images were collected by the search but those images which don't represent any emotional state were manually thrown out. For assignment of values to emotions in their valence and arousal model they have deployed Amazon Mechanical Turk (AMT) for all the images in the dataset. Their model contained five layers; an input layer, an output layer and three hidden layers.

Zhang et al. [70] created an emotion classification system based upon Deep Belief Networks (DBN). Their model employs two stage learning; the first stage focuses on training of audio visual network and the second one focuses on fusion network. In the first, the audio network is fine-tuned via pre-trained AlexNet and visual network is fine-tuned using pre-trained C3D-Sports-1M. In the second stage target emotions dataset is used for training DBN fusion network. A one dimensional audio is changed into 3 channels of 'Mel-spectrogram of 64 x 64 x 3 dimensions' and inputted to the CNN. Mel-spectrogram's size can be changed accordingly to the input of the current convolution networks pre trained on image databases. 3D CNN is used for extraction of visual features from video data which are divided in to small segment each having 16 frames. This is followed by face area detection from each frame via Viola Jones real time face detector. Eye distance from the detected face area is calculated and normalized to a fixed 55 pixels value. From each frame a RGB image of size 150x110x3 is separated based upon the set of eye distance values. The size of this RGB image is reduced to 227x227x3 for the purpose of fine tuning the input when it is given to the pre-trained 3D-CNN. For fusion of networks the Deep Belief Network is used which is created using two RBMs stacked on top of each other and trained in 2 stages. The first stage

focuses on unsupervised pre-training by using a greedy layer wise training algorithm in the bottom-up fashion. After pre-training is done, network parameters are optimized by initializing and fine tuning the RBMs. For second stage training audio and video network parameters are fixed whereas fusion network parameters are updated for correct predicted values. After training of fusion network is complete, for each audio visual segment feature representation is obtained. Average pooling is applied as each segment is different in length so as to achieve uniform global feature representations. In their research various fusion techniques were taken into consideration for feature, score and decision level fusion. The impact of these fusion techniques on classification rate was compared with the DBN network based fusion. For this experiment RML, eNTERFACE and BAUM-Is datasets were chosen and linear SVM was deployed for classification. The results achieved from DBN fusion network out-performed other fusion techniques.

S.Zhang et al. [71] focused on audio based emotion recognition system using a Deep Neural Network that performs emotion classification using discriminant temporal pyramid matching. From the audio files MFCC features were extracted from which 3 channels static was extracted; delta plus delta-delta similar to an image's RGB channels which makes it acceptable to be inputted to their proposed DCNN. Further, these features are spitted into number of slices which are over-lapping. These slices are inputted to the DCNN in order to learn segment-level features. The researchers made use of a pre-trained AlexNet to train on segment level features; their AlexNet is already trained on ImageNet dataset. These segment level features are further divided by utterances. Moreover, DTPM is used for combining segment and utterance level features for further classification by linear SVM. In their research public datasets such as BAUM-1s, EMO-DB, RML and eNTERFACE-05 has been put to test and their methodology gave promising results on the above mentioned datasets.

J. Zhao *et al.* [72] presented a deep convolutional neural network in which they combined two convolutional networks. Their mode consists of two parts or branches; the first branch is a one dimensional CNN which learns features from audio clips whereas the two dimensional CNN in second branch is used to learn MFCC features. From both branches the extracted features are combined for processing which is followed by FC layer and softmax classification. The 1D CNN in first branch is composed of six convolutional (1D) and max

pooling layers (1D) and two dense layers or FC layers. 2D-CNN in the second branch consists of four convolutional (2D) and two max pooling (2D) layers and two FC layers. After feature extraction, FC layers in both 1D and 2D CNN are deleted and the networks are combined. The reason for margining the networks is to learn multidimensional data features via dimensional networks. Experimentation was performed on EMO-DB and IMOCAP datasets. Speaker dependent and independent validation techniques were used for cross validation of the merged architecture. Their proposed methodology showed good results on both datasets.

B.Yang et al. [73] performed emotion recognition by creating a DNN which focuses on weighted mixture approach that extracts useful features. For extraction of these features preprocessing was done on the data e.g. face detection, rotation and data augmentation. Two channels were considered; one gray scale and second local binary pattern, for processing by the model. The images of gray scale channel were extracted by using a fine tuned VGG16 and local binary pattern are extracted using a CNN. The extracted features from the two channels are combined into a weighted mixture model. Softmax classification was used for calculation of final output. CK+, Oulu-CASIA and JAFFE datasets were used for testing and validation of their model. Their methodology presented highest recognition results ever achieved on all three datasets.

P. Tzirakis *et al.* [74] considered a dual modality system considering both audio and video data. They divided audio signals into six second long sequences. This is followed by preprocessing which observes the deviation in loudness of each speaker's speech. The sequences were preprocessed such that the unit variance and means are zero. LSTM layers with depth two were used on top of their CNN in order to handle a speech's temporal nature. Max pooling was applied on two properties one across time and other channels. A Residual network having 50 layers was used for video data; the layers were inputted pixel values which were achieved by cropping the actor's facial area from the video. The authors used a pre-trained ResNet50 which was fine tuned. An RNN with LSTM layers having depth two and 256 cells was used for training on the extracted features. These layers were used for handling temporal data. Eventually obtaining both kinds of features (audio and video features), both network's LSTM layers were removed and the obtained feature vectors of

audio video modality were combined and fed to a RNN having 2 LSTM layers with 256 layers each. RECOLA a spontaneous dataset was used for validation, from which improved recognition rates were obtained.

Table 3.3 Summary of Emotion Recognition Systems Based on Machine Learning Methods

Author	Dataset	Modality	Extracted features	Classification methodology
Zhalehpour et al. [67]	BAUM-1s, eNTERFACE'05	Bi-modal	A: MFCC, RASTA-PLP V: LPQ, POEM,	AV: SVM
Huibin Li et al. [68]	BU-3DFE Subset I, BU-3DFE Subset II, and Bosphorus Subset	Bi-modal	A: MFCC V: geometric map, curvature map, normal map, texture map	AV: DCNN Softmax, SVM
Kim et al. [69]	Created own dataset from Flickr	Uni-modal	V: color features, local features, semantic, object feature	V: CNN
Zhang et al. [70]	RML, eNTERFACE, BAUM-Is	Bi-modal	A: MFCC, RASTA-PLP V: CNN Based feature extraction	A: AlexNet V: C3D Sports Fusion: DBN with RBM
S.Zhang et al. [71]	BAUM-1s, EMO-DB, RML, eNTERFACE-05	Uni-modal	A: MFCC	A: AlexNet DCNN, DTPM, SVM
J. Zhao et al. [72]	EMO-DB, IMOCAP	Uni-modal	A: Raw audio, MFCC	A: 1D CNN, 2D CNN
B. Yang et al. [73]	CK+, Oulu-CASIA, JAFFE	Uni-modal	V: LBP, data augmentation, face detection, rotation	V: VGG-16 CNN, Softmax
P. Tzirakis et al. [74]	RECOLA	Bi-modal	A: Raw Audio V: CNN base feature extraction	A: RNN with one LSTM layer V: ResNet-50, two LSTM layers

3.5 Limitation and Gaps

Emotion recognition is one of the most challenging tasks for a machine as it includes many factors that need consideration and if not taken into account can affect the performance of the system. Due to the subjective nature of emotions there is a risk of biasness in machines e.g. a system might assign a specific emotion to a certain ethnicity in most cases and fails to

classify other. A machine may also fail to understand the ethnic, cultural and racial differences among people in expressing emotions. Moreover, it is also tough for machines to comprehend the differences that exist in expressing emotions between a child and an adult. Furthermore, this field is also affected by many other factors such as presence of noise, occlusions, un-evenness in pose, lightning, and variations in presenting expression by subjects across different areas of the world. Another issue that can't be ignored is the lack of research work done for emotion recognition in children. For adults, emotion recognition is a well-researched area as there are a number of quality datasets pertaining to adult's facial emotions for the machines to be trained and tested on. In children, this is a challenging task because of the obstructions and non-frontal poses that children give as they move around and fiddle a lot as compared to adults.

In literature researchers often consider the six basic emotions whereas children tend to combine various emotions together and display a blended set of emotions. In their paper, Khan *et al.* [46] discussed six basic emotions plus four mixed emotions namely Happy-aloud, Fear-surprise, Happy-disgust, and Happy-surprise. Moreover, in emotional datasets there is an imbalance among various negative and positive emotions presented by children for example the samples of happy emotion may exceeds samples of other emotions. Lastly, a significant issue is the lack of publicly available datasets for children whereas for adults there are a number of quality datasets available for researchers and thus considerable amount of research work has been performed while facial emotion recognition in children remains unexplored.

Chapter 4

PROPOSED METHODOLOGY

CHAPTER 4: PROPOSED METHODOLOGY

In this chapter we discuss the proposed methodology in detail which encompasses techniques for extraction of useful features and the methods used to classify them. First of all, the methods set out for feature extraction and strategies for classification. Figure 4.1 shows the summary of our approach.

Our emotion recognition system is a uni-model system as it considers only visual content from the video clips provided in the emotional corpuses. In the initial stage preprocessing is required so to make it suitable so that it can be provided as input to the neural network. In preprocessing stage, frames have been extracted from the video corpuses and face area is then extracted from those frames. Next distinct features sets are extracted from the frames using three networks namely VGG16, VGG19 [75] and Resnet50 [76]. This step has been carried out for video and image based classification, however the methodology deployed for division of samples i.e. frames into train and test sets in both approaches is different. The extracted features from these frames are fed to train two image classifiers namely Support Vector Machine (SVM) [77, 78] and Decision Tree [79, 80] to test the representational power of these features. The proposed methodology is explained in detail in the following section.

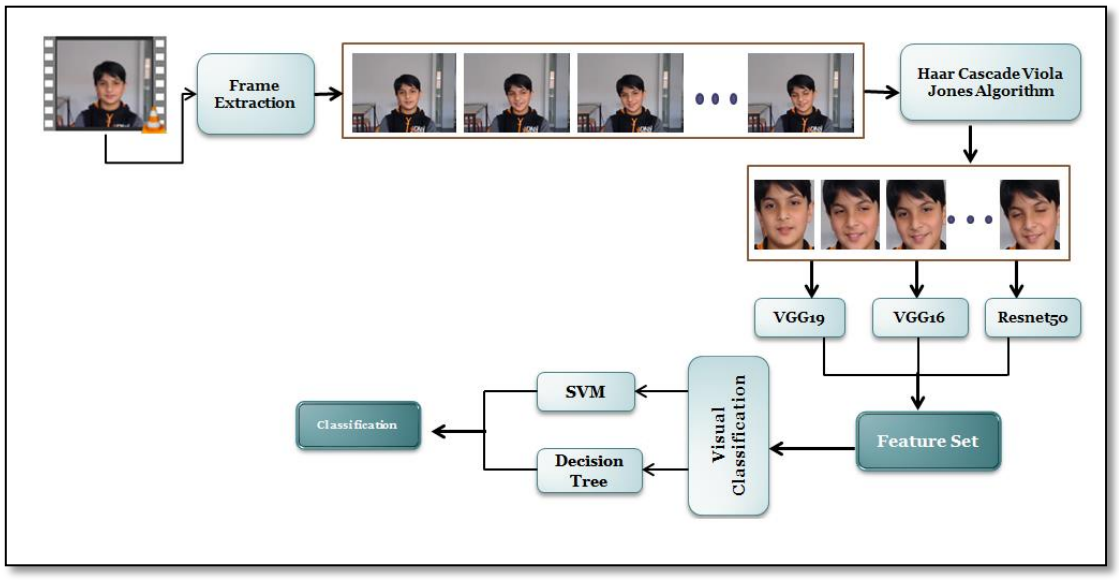

Figure 4.1 Proposed Visual Emotion Recognition System

4.1 Visual Modality

In emotion recognition systems, researchers often consider audio and visual modalities. For performing emotion recognition with visual modality it is important to carry out a number of pre-processing steps (frame extraction, face detection), feature extraction and classification. In our experiment, most of the steps are practically similar for both approaches: image based and video based classification. The detail of each step is described below.

4.2 Pre-processing

LIRIS-CSE is a video dataset having 208 videos in total. Out of 208, 185 videos are of 5 basic emotions namely disgust, fear, happiness, sad and surprise and the remaining 23 videos are of mixed emotions namely happy-aloud, fear-surprise, happy-disgust, and happy-surprise. Since the emotion anger has a single video so it has not been considered in our experiment Moreover, the mentioned mixed emotions have also not been considered.

4.2.1 Frame Extraction

Since the dataset contains videos, so the first step is frame extraction. The input provided to the system is in the form of small video clips each of 2 to 3 seconds, which shows a child conveying an emotional state. After extraction, we have approximately 19,000 frames in total.

4.2.2 Face Area Detection

Second step of this pre-processing stage is face area detection. In this step we remove the unnecessary background and extract only the face area from these frames via *Haar Cascade Classifier* [81]. Haar Cascade is an object detection algorithm based on Viola Jones algorithm. Viola Jones algorithm makes use of Haar features which represents a weak leaner or classifier. Figure 4.2 shows a set of Haar features.

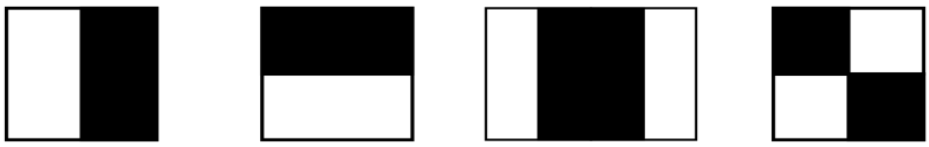

Figure 4.2 Haar Features [81]

This approach contains an ensemble of weak learners. Open CV toolkit contains pre-trained Haar cascade classifier so it has been used for face detection from our extracted frames. Figure 4.3 shows steps followed in visual pre-processing stage.

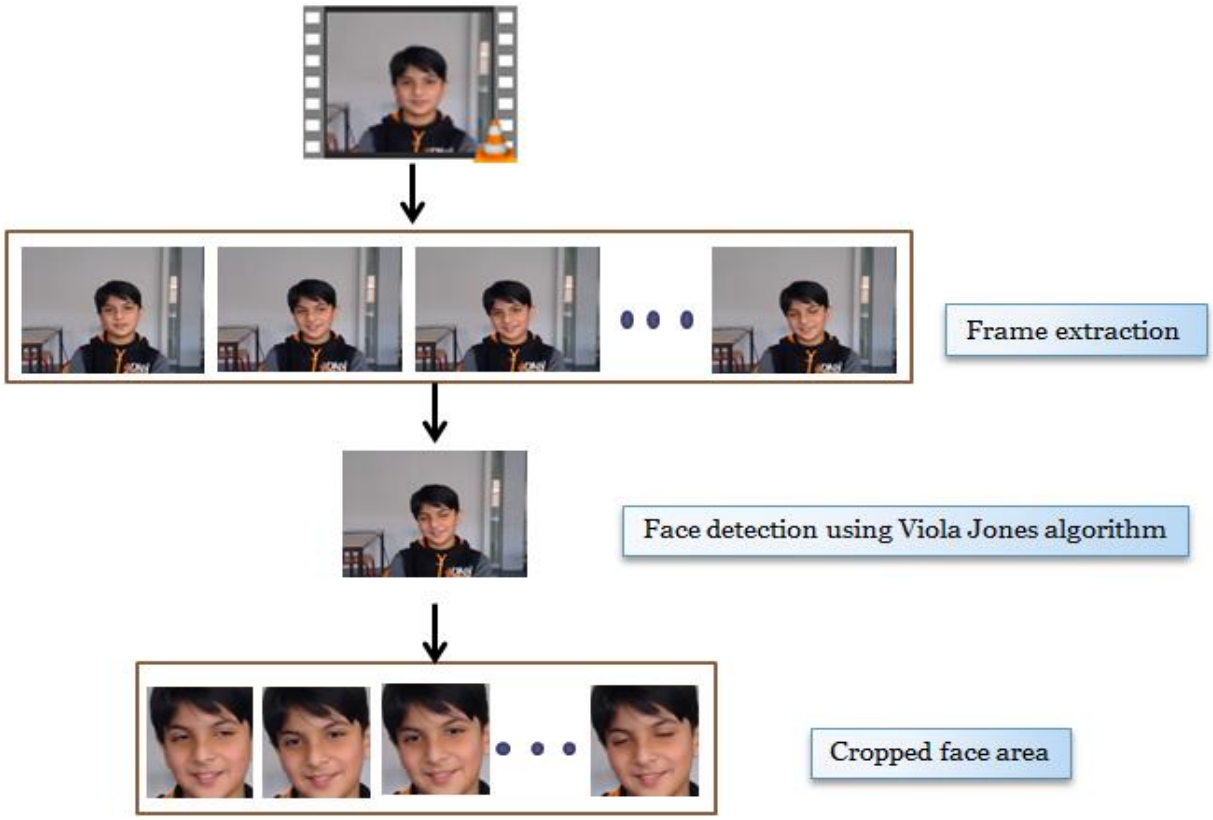

Figure 4.3 Visual Data Preprocessing

4.3 Image based approach

In image based approach, the new set is divided into two subsets for training and testing the classifier. In this approach, each frame is considered separately or independently labeled and the decision is made on each frame individually.

4.4 Video based approach

In video based approach, the *videos* have been divided into train and test sets. In this approach whole video is considered for labeling i.e. majority voting based decision is made on all the frames of a single complete video. Unlike image based approach, the frames are not considered independently; instead they belong to that specific video from which it was extracted. This helps in preserving the sequence and the temporal information of a video.

4.5 Feature Extraction

This section explains the method to extract learned image features from the data-subsets via pre-trained CNN. The proposed methodology has been implemented in MATLAB R2019b and pre-trained CNNs have been downloaded and installed from Deep Learning Toolbox provided by MATLAB. The images extracted in previous stage, are initially of different sizes and gray scale in color, but the pre-trained networks require input image to be of size 224 x 224 x 3, where 3 is the number of color channels. Input images have been resized before they are fed to the network. We conduct our first experiment using VGG19, then VGG16 and lastly ResNet50. These networks have been trained on more than a million images and can classify up to 1000 classes.

The reason for selecting the CNNs as feature extractors is because of their amazing ability to extract useful features from the data, based upon their learning [82]. VGG-Nets such as VGG16 and VGG19 are considered state of the art and they have an appealing framework due to their uniform architecture. Another type of residual network known as Resnets is also very effective in training deep networks as it stacks additional layers and builds a deeper network. This network makes use of skip connections and solves the *vanishing gradient* problem [83, 84]. Fortunately, pre-trained models of these networks are publically available making it practical for researchers to test even small sized databases. This is accomplished with using a previously trained network that is qualified to extract useful features [85] and can be put to use in any of the three following ways: *classification, feature extraction and transfer learning*. Table 4.1 gives a brief description of the ways a pre-trained network can be used.

Table 4.1 Uses of a Pre-Trained Network

Task	Explanation
Classification	A pre-trained network can be used directly for a classification task. The network will return a predicted label and class probabilities of the test image after classification.
Feature Extraction	A pre-trained network can be used as a feature extractor. These features can then be used to train a classifier such as SVM or Decision Tree to learn a new task.
Transfer Learning	The layers of a pre-trained network can be *fine-tuned* on a new/smaller dataset to solve a different classification problem. Fine-tuning a network is very useful approach as it is much cheaper and a faster option compared to training a network initially from scratch.

When using a pre-trained network as a feature extractor, instead of propagating through the entire network, propagation is ceased at a certain layer that has been already specified. Basically the network is 'chopped of' at the specified layer. Then feature values are extracted from that layer and only those feature values are considered as feature vectors. The Feature extraction part can use either earlier layers that extract generic features such as shapes, blobs, edges etc. or later layers that extract more specific features like contours or small objects. Studies show that using a pre-trained network that has already learned to extract useful and powerful features from data, as a *feature extractor*, is not only comparatively much faster and cheaper compared to training a network from scratch but is beneficial in achieving quality results [32].

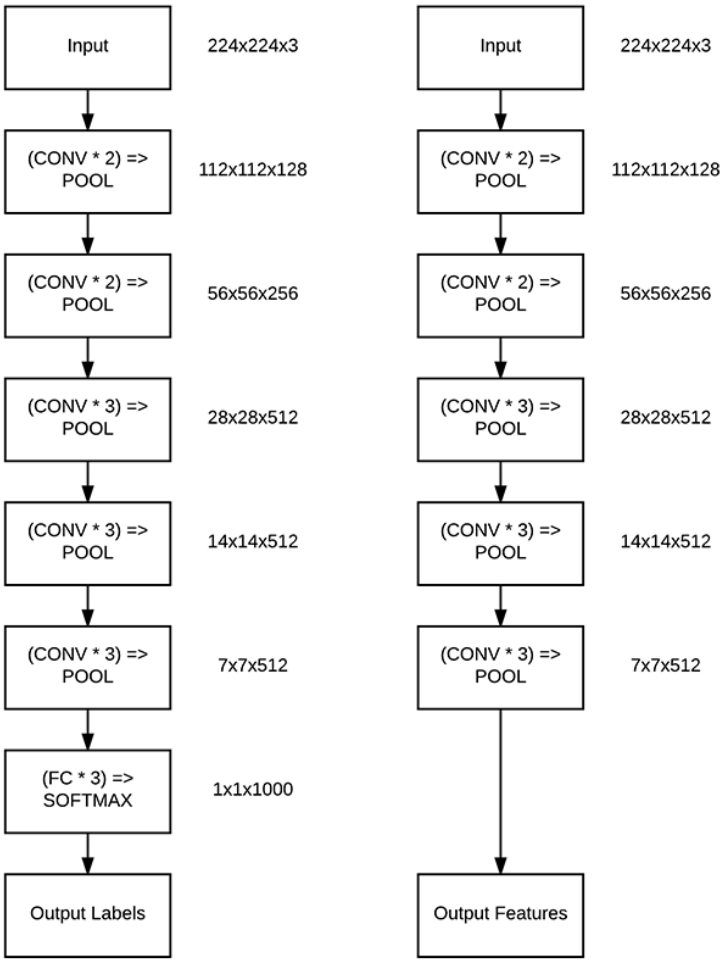

Figure 4.4: Left: Original VGG16 Network. Right: Removed FC Layers from VGG16[88]

Another approach known as *Transfer Learning* is the ability of a system to apply the knowledge gained in a previous task to a newly assigned task for the purpose of object detection and classification etc. [86]. In this approach the pre-trained net is fine-tuned in accordance to the experimental setup. Nowadays, most of the networks are trained on ImageNet database [87]. Figure 4.5 presents a snapshot of ImageNet dataset. With ImageNet, these networks are trained on more than a million images and have learned to classify up to 1000 classes including various animals, objects and many shapes. However, if the architecture of the previously trained network is too complex, the extracted features may not be suitable for a comparatively simpler task. Similarly, if the target dataset is small and similar to the dataset on which the network is pre trained, network fine-tuning could cause

over fitting as there is not much data available from which the data can learn from [32].

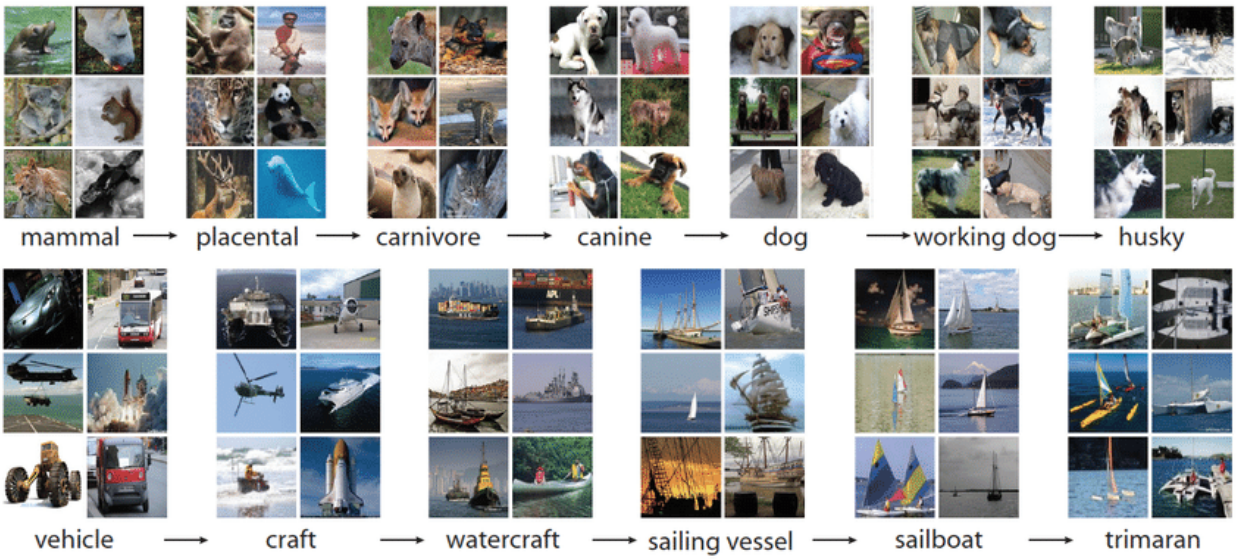

Figure 4.5 Snapshot of ImageNet Dataset [87]

4.5.1 VGG16

VGG16 is a convolutional neural network that we have deployed used in our work [88]. A pre-trained VGG16 model is provided in MATLAB's Deep Learning Toolbox. This network has been trained on millions of images from ImageNet dataset. In order to load the pre-trained network in MATLAB, following command can be used:

```
net = vgg16
```

The pre-trained network has 41 layers, having 16 layers with learnable weights. There are 13 convolutional layers, and 3 fully connected layers. The network takes an input image of size 224-by-224. In our work, we have considered a fully connected layer **'fc7'**of a pre-trained VGG16 which returns a feature vector of size 1x1x4096. The network's architecture can be viewed using:

```
net.Layers
```

VGG16 is a popular network created by Visual Geometry Group at Oxford**'s**. VGG16 is a quite large network having approx. 138 million parameters; however its architecture is very

simple as the convolution and pooling operations throughout the network remain uniform. The beauty of VGG16 is that it has a much simpler network despite the fact that there a so many hyper-parameters. Here, 16 indicate the number of layers with parameters. Throughout the network, convolution operations are performed via 3x3 filters, 1 stride and same padding. Similarly, in all pooling layers, the filter is 2x2, 2 stride and same padding. As the network starts, the first two layers are convolutional layers which apply 64 filters of size 3x3 on the input image of size 224x224x3 which results in 224x224x64 volume. Here we see no change in the output because of same convolutions being applied. Next pooling layers are applied which reduce the dimensions of the volume i.e. 224x224x64 has reduced to 112x112x64. Then two convolution layers with 128 filters are applied and after that a pooling layer reduces the volume to 56x56x128. Next comes three convolution layers with 256 filters and a pooling layer (which outputs a volume of size 28x28x256). Next comes three convolution layers with 512 filters and a pooling layer (outputs 14x14x512). At this point the authors agreed to fix the number of filters to 512, thus they remain the same for the rest of the network. Then three convolution layers with 512 filters and a pooling layer (7x7x512). Lastly there are three fully connected layers and a softmax layer with 1000 classes. The only drawback of VGG16 net is the huge number of learnable parameters that are to be trained. The architecture of VGG16 network is show in Figure 4.6.

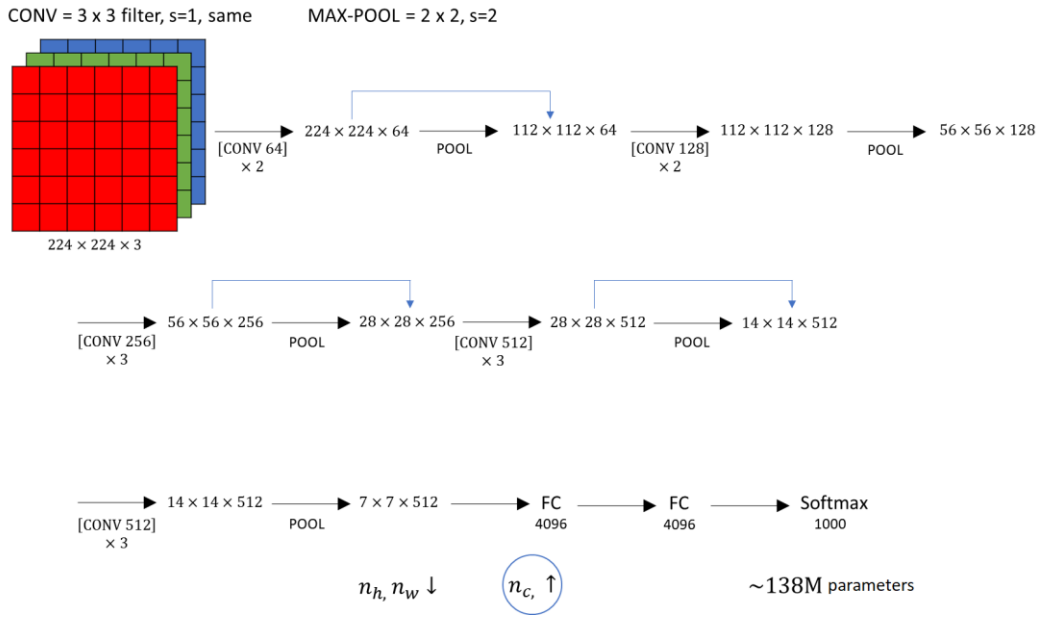

Figure 4.6 Architecture of VGG16 [88]

4.5.2 VGG19

We have used pre-trained *VGG19* which is a variant of VGG16. This model is also provided in MATLAB's Deep Learning Toolbox and is trained on ImageNet dataset. VGG19 consists of 47 layers in total having 19 layers with learnable weights [89]. It is composed of 16 convolutional layers with stride [1 1] and 3 fully connected layer. The architecture of VGG19 is provided in the Figure 4.7. Here, we have considered the output of two layers; a fully connected layer **'fc8'** and a max-pooling layer **'pool5'** as features, of sizes 1x1x1000 and 7x7x512 respectively. VGG19 is created by the same group of people who created VGG16. The architecture of both networks is quite similar where VGG19 is comparatively bigger in size than VGG16. However in most cases both networks perform almost the same hence researchers tend to use VGG16 instead.

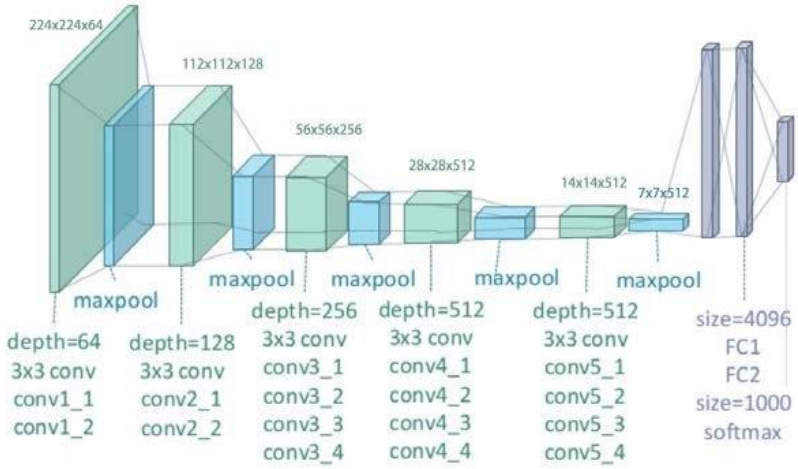

Figure 4.7 Architecture of VGG19 [89]

4.5.3 Resnet50

Resnet50 is another powerful Convolutional Neural Network that we have deployed in our research [90]. Similar to the networks discussed above, Resnet50 is also available in MATLAB's Deep Learning Toolbox and is trained on ImageNet dataset. It has 177 layers in total and 50 layers with learnable weights. The network's architecture is provided in the Figure 4.8 below. We have considered the output feature vector, of size 1x1x1000, from the last fully connected layer **'fc1000'.**

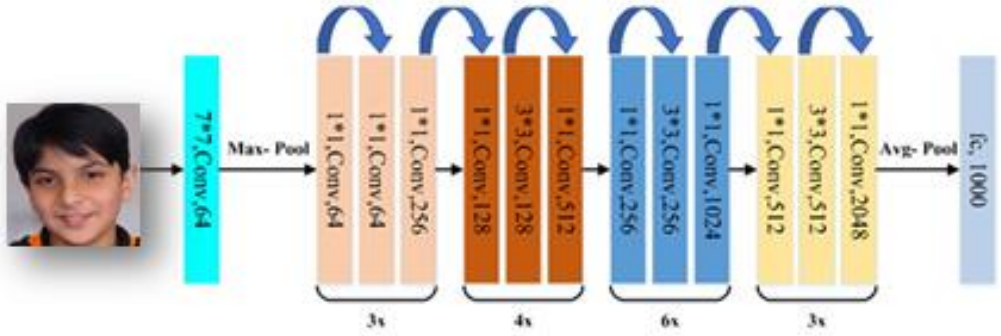

Figure 4.8 Architecture of Resnet50 [90]

The features from all three networks are forwarded to models i.e. SVM and decision tree for classification. The layer 'fc8' of VGG19 has 4097000 learnable parameters but layer 'Pool5' is a max pooling layer which has zero learnable parameters. In most networks, pooling layers are just used for reducing the dimension size of the image thus lessening the computational cost of training the network but there are no learnable parameters in these layers which affect the back propagation process [91]. For VGG16, fully connected layer (fc7) contains 16781312 parameters. Similarly, the fully connected layer (fc1000) of Resnet50 contains 2049000 parameters. Feature vector (FV) size and the learnable parameters of the selected layers from all three networks are shown in Table 4.2.

Table 4.2: Feature Vector (FV) Size Per Layer

Network	Layer	Type	FV Size	No of Parameters	Total No.of Parameters	Detail
VGG16	'fc7'	Fully Connected	4096	Weights:4096x4096 Biases: 4096x1 16781312	138 Million	Fully connected layer
VGG19	'fc8'	Fully Connected	1000	Weights:1000x4096 Biases: 1000x1 4097000	144 Million	Fully connected layer
VGG19	'pool5'	Max Pooling	25088	Zero	144 Million	2x2 max-pooling with stride [2 2] and padding [0 0 0 0]
ResNet50	'fc1000'	Fully Connected	1000	Weights: 1000x2048 Biases : 1000x1 2049000	25.6 Million	Fully connected layer

4.6 Classification

LIRIS-CSE is a relatively smaller dataset in terms of number of videos. So instead of training the network entirely from scratch on this dataset, we have used the extracted features from these networks to train two different types of classifiers i.e. Support Vector Machine (SVM) and Decision Tree (DT). The above mentioned models for classification task is a good choice, due to their robustness and ability to perform well in case of complex dataset, for instance, SVM performs effectively in case of high dimensional data i.e. the number of samples are less than number of dimensions. In terms of performance, there doesn't exists much difference in using a CNN's softmax function for classification or using above mentioned classifiers with features extracted by a CNN as it is empirically proven that SVM performs well with features from a CNN [92] especially when the dataset is of small size [32].

4.6.1 Support Vector Machine (SVM)

A multi-class SVM is a combination of multiple binary SVM which performs classification on five emotional states. When labeled data is provided to SVM for training, it generates a hyper plane that categorizes various observations. In SVM, the hyper plane acts as a decision boundary and the goal is to maximize the margin between support vectors, which are data points closer to the hyper plane. The hyper plane having w and b parameters is presented as:

$$f(x) = sign(w^t x + b) \tag{4.1}$$

If the data is linearly separable such a margin is good, where a soft margin is used to solve the problem of linearly inseparable data by introducing a positive slack variable with the hyper plane:

$$\xi_i$$

In case the data is not linearly separable (x_i, y_i) then the hyper plane is given as:

$$min_{w,\xi}\frac{1}{2}w^T w + C\sum_{i=1}^{N}\xi_i$$

$$\text{subject to } y_i(w^T\varphi(x_i) + b) \geq 1 - \xi_i \qquad \text{for } i = 1, \ldots., N, \tag{4.2}$$

Where the nonlinear parameter is presented by φ which transforms the data into a higher dimensional space whereas the tradeoff between maximizing the margin and minimizing the error by *C*. In case of dual space, the equation above becomes:

$$\max_\alpha L(\alpha) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j y_i y_j \varphi(x_i)\varphi(x_j)\ ,$$

subject to

$$\sum_{i=1}^{N}\alpha_i y_i = 0, \qquad 0 \leq \alpha_i \leq C\ . \tag{4.3}$$

The optimal hyper plane is provided by Lagrange multiplier α that combines two objectives in:

$$f(x) = sign\left(\sum_{i\in SV} w_i' K(x, x_i) + b\right), \tag{4.4}$$

Where $w_i' = \alpha_i y_i$ presents a non-zero value and Kernel function is shown as $K(x, x_i)$.

In our experiment, we have used a multi-class model for SVM known as *fitcecoc* [93] provided by MATLAB. This function uses K (K – 1)/2 binary support vector machine (SVM) models or binary learners and a one-versus-one coding design, where K is the number of unique class labels (levels). A coding design presents a matrix in which elements denote which classes are trained by the binary learner. In one-versus-one coding design each binary learner has one positive class, one negative and rest is ignored. A one-versus-one coding design is a heuristic method for solving multi-class classification problems using binary classification algorithms. This design divides a dataset having multiple classes into a dataset, one for each class versus every other class. For example, for the 5 classes in our experiment: disgust, fear, happy, sad and surprise the one-vs.-one will be divided it into ten

binary classification datasets:

- **Binary classification problem1**: disgust vs. fear
- **Binary classification problem2**: disgust vs. happy
- **Binary classification problem3**: disgust vs. sad
- **Binary classification problem4:** disgust vs. surprise
- **Binary classification problem5:** fear vs. happy
- **Binary classification problem6:** fear vs. sad
- **Binary classification problem7:** fear vs. surprise
- **Binary classification problem8:** happy vs. sad
- **Binary classification problem9:** happy vs. surprise
- **Binary classification problem10**: sad vs. surprise

The formula:

$$K(K-1)/2 \tag{4.5}$$

Can also be written as $\frac{\text{No.of classes x (No.of classes}-1)}{2}$ where the calculated answer represents the number of models for each binary classification problem. Each model predicts a class and the model having maximum number votes is predicted by this coding design.

4.6.2 Decision Tree

For classification and prediction problems, a decision tree is a commonly used model. The structure of a decision tree is a tree like flowchart; it consists of root node, branches and leaf node. The root node is the parent node in the tree, located at the top. In a decision tree, each internal node indicates a test on a feature and every branch shows the result of the test, where the terminal or leaf node holds the class label. In our experiment, we have used *fitctree*, which is a binary decision tree for multiclass classification provided by MATLAB. Based on the features given as input to the model, the model gives a fitted binary decision tree which splits the branch nodes based on the input features. Figure 4.9 shows structure of decision tree.

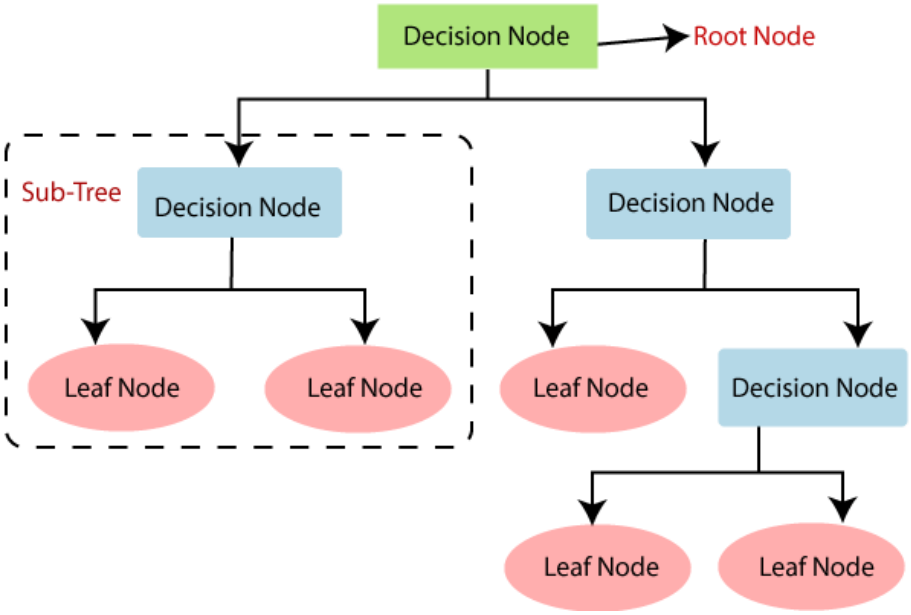

Figure 4.9 Structure Of Decision Tree [79]

In the next step these features which are already robust and discriminative are inputted to these classifiers for classification**.**

Chapter 5

EXPERIMENTAL RESULTS

CHAPTER 5: EXPERIMENTAL RESULTS

This chapter presents our experimental results on four feature sets from VGG19, VGG16 and Resnet50 with two classifiers SVM and DT. We have employed three kinds of experimental setups i.e. 80-20% split, K-Fold cross validation (CV) and leave one out cross validation (LOOCV) for both image based and video based approaches. All the experiments have been conducted on LIRIS-CSE dataset from which approximately 19,000+ frames were extracted from 185 videos.

5.1 Database Explanation

Several studies focus on emotion recognition in adults, overlooking the importance of FER systems for children. In order to contribute in that part, we have used a spontaneous emotional dataset for children: LIRIS-CSE [46].

5.1.1 LIRIS-CSE Dataset

Our study uses an emotional dataset LIRIS-CSE [46] that contains 208 videos of 12 culturally varied children showing natural expressions in two setups; one is classroom/lab and second is home setup. This dataset has been created, involving videos of 12 children, 5 male and 7 female of age group between 6 and 12 years (average 7.3 years). The kids have been shown animated emotional videos for inducing stimuli in children. The children's expressions have been recorded using a webcam attached at the top of a laptop along with a speaker output at 50 cm apart. Upon stimulation, the children have shown six basic emotions i.e. anger, disgust, fear, happy, sad and surprise. Although there are six basic universally accepted emotions including anger but during the creation of the dataset, the stimuli (movies, video clips) to induce '*negative'* emotions in young children have been selected with keeping many ethical and moral reasons in mind. So the selected stimuli should not have any long term negative impact on young children. Therefore there are more videos of '*positive'* emotions i.e. happiness, surprise and very few videos of negative emotions i.e. fear, sadness, disgust and only one video of anger. Even though children have not been shown any anger inducing video, still one child has been recorded displaying this emotion. Thus there is only one video of anger in the dataset. Figure 5.1 displays the

environmental setup, illumination changes and spontaneous expressions presented by children in LIRIS-CSE dataset.

Figure 5.1 Emotional Images of Children in LIRIS-CSE Dataset [46]

5.2 Image Based Results

For image based approach, we have tested the representational power of the extracted features by deploying; first the 80-20% split techniques then K-Fold cross validation.

5.2.1 80-20% split

In this methodology, the classifiers are trained on feature vectors extracted from 80% of the frames and then tested on remaining 20%. As discussed earlier, the division of the datasets in image based and video based approach is slightly different from each other. However, the 80-20% split ratio is being followed for both approaches.

In image based approach, the dataset is divided into two subsets for training and testing the classifier. The train set contains 80% of the frames and the test set contains the remaining 20%. The key point in image based approach is that the division of the frames is done randomly. Table 5.1 reports the results for frame or image based approach. In this setup, SVM with VGG16 features for image-based classification gives the best result, providing a classification accuracy of 99.9%. With VGG19 (fc8) and Resnet50 features, we have achieved classification accuracy of 99.5% and 99.8% respectively. DT has also performed remarkably well on the three set of features extracted from VGG16, VGG19 and Resnet50, giving classification accuracies of 97.8%, 97.5% and 96.9% respectively. This proves that, combination of SVM with VGG16 features is a very powerful one as it accomplishes extremely high classification accuracies.

5.2.2 K-Fold Cross Validation (CV)

In this experimental setup, we have deployed 10 fold cross validation. The dataset is randomly split in to 10 approximately equal parts. Out of 10 folds, 9 are used for training the classifier and the remaining one is used for testing. This process is repeated 10 times so that each of the 10 sub-samples is tested and the entire dataset is covered completely. Lastly the obtained accuracies are averaged and a final average accuracy is obtained.

In this approach, each bin contains approximately 1927 samples and each bin has been used for training and testing, by the classifier. We have achieved the highest classification accuracy of 99.8% with the combination VGG16 features and SVM as reported in Table 5.1. For the other two set of features by VGG19 and Resnet50 with SVM we achieved accuracy of 99.0% and 99.5% respectively. Similarly, DT also performed extremely well on the three set of features from VGG16, VGG19 and Resnet50, giving classification accuracies of 94.8%, 94.2% and 93.2% respectively.

Table 5.1 Classification Results for Image Based Approach

Image based approach		***Setup1***	***Setup 2***
Feature extractor	***Classifier***	***80%-20% split***	***K-Fold CV***
VGG16 (fc7)	*SVM*	99.9%	99.8%
	Decision Tree	97.8%	94.8%
VGG19 (fc8)	*SVM*	99.5%	99.0%
	Decision Tree	97.5%	94.2%
Resnet50 (fc1000)	*SVM*	99.8%	99.5%
	Decision Tree	96.9%	93.2%

5.3 Video Based Results

For video based approach, similar to image based approach we have deployed, the 80-20% split and K-Fold cross validation techniques and in order to improve the results we have used LOOCV.

5.3.1 80-20% split

In video based approach, the *videos* are divided into train and test sets i.e. 80% of the videos belong to train set and the remaining 20% belong to test set. The key point in video based classification is that the division of the frames is not random; rather all frames belonging to its respective video are in the same set. For this approach, the results are not as promising. It can be seen in Table 5.2. SVM gives a classification accuracy of 57% for VGG19 (pool5) features, 55% for both VGG16 and Resnet50 features. As expected, DT also performed poorly for this approach giving 47%, 52% and 45% accuracy for VGG19, VGG16 and Resnet50 features respectively.

5.3.2 K-Fold Cross Validation (CV)

In this setup, each bin consists of 18 videos and each bin has been used for training and testing by the classifier and thus every video sample is used. In Table 5.2 it can be seen, for video based approach, the results have slightly improved with this setup. SVM resulted in achieving classification accuracies of 73%, 69% and 71% for VGG19, VGG16 and Resnet50 features respectively. DT accomplished 64%, 65% and 63% for VGG19, VGG16 and Resnet50 features respectively.

5.3.3 Leave One Out Cross Validation (LOOCV)

This setup is deployed in order to improve the classification results off video based approach. As the name suggests, in leave one out cross validation, one instance of the dataset i.e. one video of a child presenting an emotion becomes part of the test set once, while all remaining instances are part of the training set (in each iteration). This process is repeated corresponding to the total number of instances (videos) in the dataset. This method has helped the SVM classifier in achieving improved classification accuracies of 94%, 91% and 93% for VGG19, VGG16 and Resnet50 features respectively. These results are given in Table 5.2. DT has also accomplished remarkable results of 91%, 89% and 88% for VGG19, VGG16 and Resnet50 features respectively.

Table 5.2 Classification Results for Video Based Approach

Video based approach		***Setup1***	***Setup 2***	***Setup 3***
Feature Extractor	***Classifier***	***80%-20% split***	***K-Fold CV***	***LOOCV***
VGG19	*SVM*	57.5%	73.5%	94.0%
	Decision Tree	47.5%	64.3%	91.8%
VGG16	*SVM*	55.2%	69.7%	91.5%
	Decision Tree	52.5%	65.9%	89.5%
Resnet50	*SVM*	55.0%	71.2%	93.2%
	Decision Tree	45.6%	63.4%	88.0%

In 80-20% split, image-based approach gave better results compared to video-based approach, as there exist sufficient frames of the same video in training and testing set but in video based all frames of the same video are present in either of two sets. In other words, video based approach considers a single video as one sample which led to insufficient amount of training samples. It can be seen in other two setups that accuracies had improved due to the sufficient amount of data available for training. For K-fold cross validation, such decent performance by the classifiers for image based approach was anticipated because it is a very powerful method as it covers the entire dataset. Lastly, LOOCV immensely improved the accuracies of video based approach as the number of train videos has been increased. All frames of a single video are used for testing in each iteration, while the frames of remaining videos are used for training. Majority voting is then applied to the testing frames of a video for decision. Entire dataset is tested using LOOCV.

To summarize, our proposed framework has achieved highest accuracy of **99%** for image-based classification via features of all three networks with SVM using 80-20% split and K-Fold CV. Table 5.3 reports the confusion matrix of 80-20% split using VGG16 features with SVM as a classifier, resulting 99.9% accuracy. The confusion matrix of this setup depicts only one misclassification of surprise emotion predicted as fear. Table 5.4 reports the confusion matrix of K-Fold CV using VGG16 features with SVM as a classifier, resulting 99.8% accuracy. From the confusion matrix it can be seen that emotion disgust

was misclassified as happy twice and four samples of fear were miss-predicted as happy and five as surprise. Similarly, seven samples of happy were predicted wrong; one as sad and six as surprise. Moreover, one sample of sad was miss-predicted as happy and four as surprise. Lastly, surprise suffered most misclassifications; four as fear, seven as happy and two as sad.

Similarly Table 5.5 reports the confusion matrix for video based approach, in which SVM has achieved the highest classification accuracy of 94% with VGG19 features using leave one out cross validation. The number of samples of each emotion represents the number of videos for that emotion i.e. majority voting based decision has been made on all the frames of a single video. It can be seen that classifier reported zero misclassifications for happy, whereas fear has most misclassified videos. Two samples of disgust have been misclassified as happy and surprise, whereas five samples of fear have been predicted wrong; one as disgust, one as happy and three as surprise. Lastly, three samples of surprise have been misclassified as happy and one as fear. In both setups, overall surprise suffered most misclassifications; especially surprise was mis-predicted as happy. This is understandable, as children tend to combine these emotions and present a mixed emotion.

Table 5.3 Confusion Matrix Image Based Approach. 80-20% split with VGG16 Features and SVM

	Disgust	**Fear**	**Happy**	**Sad**	**Surprise**
Disgust	167	0	0	0	0
Fear	0	709	0	0	0
Happy	0	0	1253	0	3
Sad	0	0	0	813	0
Surprise	0	1	0	0	911

Table 5.4 Confusion Matrix Image Based Approach. K-Fold CV with VGG16 Features and SVM

	Disgust	**Fear**	**Happy**	**Sad**	**Surprise**
Disgust	819	0	2	0	0
Fear	0	3495	4	0	5
Happy	0	0	6302	1	6
Sad	0	0	1	4082	4
Surprise	0	4	7	2	4539

Table 5.5: Confusion Matrix Video Based Approach. LOOCV with VGG19 Features and SVM

	Disgust	Fear	Happy	Sad	Surprise
Disgust	7	0	1	0	1
Fear	1	28	1	0	3
Happy	0	0	62	0	0
Sad	0	0	0	29	0
Surprise	0	1	3	0	48

Each setup resulted in different set of accuracies; however it is important to note that features representations have more important role in the overall performance. As discussed, we used pre-trained networks which have been trained on ImageNet dataset, which is very different from our dataset. In that case, it is more useful to extract generic representations rather than specific [32]. For image based approach, it has been observed that VGG16 performs better in comparison to the other two networks in every case. From the results of video based approach, it is observed that VGG19 features outperform VGG16 and ResNet50 features in majority of the experiments carried out. This is because the feature representations from VGG19 are more powerful, as we ceased propagation of the network at Pool5 layer. Aravind et al [32] also proved this in their study that on a smaller dataset, different from original dataset (on which it was trained); the features from an earlier layer deliver better accuracy. Furthermore, in all tests carried out, SVM beats DT with a huge difference, proving that it is one of the best classifiers available. Our experiments have also demonstrated that the combination of pre-trained networks as feature extractors with models like SVM and DT is a fine starting point to learn a new task and can guarantee promising results on a smaller dataset.

We have also compared our results with previous results computed on LIRIS-CSE dataset. Table 5.6 reports this comparison. Khan *et al* [46] achieved an average image-based classification accuracy of **75%** using VGG16 architecture based on transfer learning approach. Uddin *et al* [47] reported an accuracy of **84.2%** using deep spatio-temporal LDSP on spark. Zhao *et al* [49] used MEC based hierarchical emotion recognition model with VGG16 in conjunction with localization module and achieved **95.67%** accuracy on LIRIS-CSE dataset. In our work, we have performed image and video based classification on this dataset and have managed to achieve promising results. With frame based approach, our

achieved recognition rates are higher as compared to results reported in literature. This shows that our proposed framework is effective in recognizing facial expressions of the children from images present in LIRIS-CSE dataset.

Table 5.6 Comparison with other works on LIRIS-CSE

Author	Technique	Visual Recognition Rate
Khan et al [46]	Transfer learning • Pre trained CNN (VGG16), Fine tuning • 80-20% split	• **75%**
Uddin et al [47]	• LDSP-TOP • 1-D CNN • LSTM	• **84.2%**
Zhao et al [49]	• Mobile Edge Computing (MEC) • Pre-trained VGG16 for feature extraction • Localization module	• **95.67%**
Proposed method	Feature extraction • Pre trained CNN (VGG19(fc8, pool5), VGG16, Resnet50) • 80-20% split, KFold cross validation and LOOCV • SVM and Decision Tree	Image based classification (max) • **99.9%** (80-20% split) • **99.8 %** (KFold CV) Video based classification (max) • **57%** (80-20% split) • **73%** (KFold CV) • **94%** (LOOCV)

Chapter 6

CONCLUSION AND FUTURE WORK

CHAPTER 6: CONCLUSION AND FUTURE WORK

6.1 Conclusion

Facial emotion recognition is a very active area of research and still various frameworks are being deployed to improve the machine's ability to accurately classify several emotions presented by children in a pure and spontaneous fashion. In this paper, we have used pre-trained convolutional neural networks as feature extractors on LIRIS-CSE video dataset for facial expression recognition on five universal spontaneous emotions of children. Our experiments demonstrated that the features extracted from these networks are a powerful representation of the input data and they perform very well when classified using SVM and DT. The effectiveness of this approach is proven by our achieved results, as it outperformed the frameworks applied on the same dataset in literature for image-based approach. Using our framework, the difficulty posed by small datasets can be avoided especially using the feature representations from pre-trained networks. Our proposed methodology also proved its worth by performing extremely well for video based classification.

6.2 Contributions

- Fully automated image and video based emotion recognition system for children.
- Review and comparison of recent advancements and contributions made in emotion recognition domain.
- Detailed experimentation carried out on LIRIS-CSE dataset by using features from three different networks VGG16, VGG19 and Resnet50 and two different classifiers.
- Accomplished one of the highest classification recognition accuracies on the mentioned dataset.

6.3 Future Work

In future, our proposed framework can be tested on more varied children facial expression datasets. Several emotions presented by children other than the six universal emotions can also be considered for recognition. Dataset with mixture of several modalities, such as audio or video can also be used in future to achieve dynamic results.

REFERENCES

[1] Lucey, P., Cohn, J. F., Kanade, T., Saragih, J., Ambadar, Z., & Matthews, I. (2010, June). The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *2010 ieee computer society conference on computer vision and pattern recognition-workshops* (pp. 94-101). IEEE.

[2] P. Ekman, "An argument for basic emotions," *Cogn. Emot.*, vol. 6, no. 3–4, pp. 169–200, 1992.

[3] R. E. Thayer, *The psychobiology of mood and arousal*. Oxford University Press, Oxford, 1989.

[4] Tao, Jianhua; Tieniu Tan (2005). "Affective Computing: A Review". *Affective Computing and Intelligent Interaction*. LNCS 3784. Springer. pp. 981–995.

[5] R. W. Picard, "Affective Computing: Challenges," Int. J. Hum. Comput. Stud., 1995.

[6] P. N. Johnson-Laird and E. Shafir, "The interaction between reasoning and decision making: an introduction," Cognition, 1993.

[7] Heise, David (2004). "*Enculturating agents with expressive role behavior*". In Sabine Payr; Trappl, Robert (eds.). Agent Culture: Human-Agent Interaction in a Mutlicultural World. Lawrence Erlbaum Associates. pp. 127–142.

[8] D. Bolinger and D. L. M. Bolinger, *Intonation and its uses: Melody in grammar and discourse*. Stanford University Press, 1989.

[9] C.-H. Wu, J.-C. Lin, and W.-L. Wei, "Survey on audiovisual emotion recognition: databases, features, and data fusion strategies," *APSIPA Trans. Signal Inf. Process.*, vol. 3, 2014.

[10] Y.-L. Tian, T. Kanade, and J. F. Cohn, "Facial expression analysis," in *Handbook of face recognition*, Springer, 2005, pp. 247–275.

[11] Y. Wang and L. Guan, "Recognizing human emotional state from audiovisual signals," *IEEE Trans. Multimed.*, vol. 10, no. 5, pp. 936–946, 2008.

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

[13] Poria, S., Cambria, E., Bajpai, R., & Hussain, A. (2017). A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, *37*, 98-125.

[14] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture

classification with local binary patterns," IEEE Trans. Pattern Anal. Mach. Intell., vol. 24, no. 7, pp. 971–987, 2002.

[15] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86*(11), 2278-2324.

[16] P. A. Millan Arias and J. A. Quiroga Sepulveda, "Deep Learned vs. Hand-Crafted Features for Action Classification," *2018 IEEE First International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, Laguna Hills, CA, 2018, pp. 170-171.

[17] A. Saha, S. S. Rathore, S. Sharma and D. Samanta, "Analyzing the difference between deep learning and machine learning features of EEG signal using clustering techniques," *2019 IEEE Region 10 Symposium (TENSYMP)*, Kolkata, India, 2019, pp. 660-664.

[18] McCulloch, Warren; Walter Pitts (1943). "A Logical Calculus of Ideas Immanent in Nervous Activity". Bulletin of Mathematical Biophysics. 5 (4): 115–133. doi:10.1007/BF02478259. PMID 2185863.

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

[20] Deng, L.; Yu, D. (2014). "Deep Learning: Methods and Applications"(PDF). Foundations and Trends in Signal Processing. 7 (3–4): 1–199. doi:10.1561/2000000039.

[21] Graves, Alex; Eck, Douglas; Beringer, Nicole; Schmidhuber, Jürgen (2003). "Biologically Plausible Speech Recognition with LSTM Neural Nets" (PDF). *1st Intl. Workshop on Biologically Inspired Approaches to Advanced Information Technology, Bio-ADIT 2004, Lausanne, Switzerland*. pp. 175–184.

[22] 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.

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

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

[25] Ciresan, Dan; Ueli Meier; Jonathan Masci; Luca M. Gambardella; Jurgen Schmidhuber (2011). "*Flexible, High Performance Convolutional Neural Networks for Image Classification*" (PDF). Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence-Volume Volume Two. 2: 1237–1242. Retrieved 17 November 2013.

[26] Sergey Ioffe, and Christian Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift", 2nd March 2015.

[27] 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.

[28] 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.

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

[30] Pan, S. J., & Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, *22*(10), 1345-1359.

[31] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[32] Ravi, A. (2018). Pre-trained convolutional neural network features for facial expression recognition. *arXiv preprint arXiv:1812.06387*.

[33] Pantic, M. (2009). Machine analysis of facial behaviour: Naturalistic and dynamic behaviour. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *364*(1535), 3505-3513.

[34] Ko, B. C. (2018). A brief review of facial emotion recognition based on visual information. *sensors*, *18*(2), 401.

[35] Ekman, P., & Friesen, W. V. (1986). A new pan-cultural facial expression of emotion. *Motivation and emotion*, *10*(2), 159-168.

[36] Li, S., & Deng, W. (2020). Deep facial expression recognition: A survey. *IEEE transactions on affective computing*.

[37] Pantic, M., Valstar, M., Rademaker, R., & Maat, L. (2005, July). Web-based database for facial expression analysis. In *2005 IEEE international conference on multimedia and Expo* (pp. 5-pp). IEEE.

[38] LoBue, V., Baker, L., & Thrasher, C. (2018). Through the eyes of a child: Preschoolers’ identification of emotional expressions from the child affective facial expression (CAFE) set. *Cognition and Emotion*, *32*(5), 1122-1130.

[39] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).

[40] Egger, H. L., Pine, D. S., Nelson, E., Leibenluft, E., Ernst, M., Towbin, K. E., & Angold, A. (2011). The NIMH Child Emotional Faces Picture Set (NIMH-ChEFS): a new set of children's facial emotion stimuli. *International journal of methods in psychiatric research*, *20*(3), 145-156.

[41] Dalrymple, K. A., Gomez, J., & Duchaine, B. (2013). The Dartmouth Database of Children‟s Faces: Acquisition and validation of a new face stimulus set. *PloS one*, *8*(11), e79131.

[42] LoBue, V., & Thrasher, C. (2015). The Child Affective Facial Expression (CAFE) set: Validity and reliability from untrained adults. *Frontiers in psychology*, *5*, 1532.

[43] Langner, O., Dotsch, R., Bijlstra, G., Wigboldus, D. H., Hawk, S. T., & Van Knippenberg, A. D. (2010). Presentation and validation of the Radboud Faces Database. *Cognition and emotion*, *24*(8), 1377-1388.

[44] Nojavanasghari, B., Baltrušaitis, T., Hughes, C. E., & Morency, L. P. (2016, October). Emoreact: a multimodal approach and dataset for recognizing emotional responses in children. In *Proceedings of the 18th acm international conference on multimodal interaction* (pp. 137-144).

[45] Dapogny, A., Grossard, C., Hun, S., Serret, S., Grynszpan, O., Dubuisson, S., ... & Bailly, K. (2019). On Automatically Assessing Children's Facial Expressions Quality: A Study, Database, and Protocol. *Frontiers in Computer Science*, *1*, 5.

[46] Khan, R. A., Crenn, A., Meyer, A., & Bouakaz, S. (2019). A novel database of children's spontaneous facial expressions (LIRIS-CSE). *Image and Vision Computing*, *83*, 61-69.

[47] Uddin, M. A., Joolee, J. B., & Sohn, K. A. (2021). Dynamic Facial Expression Understanding Using Deep Spatiotemporal LDSP On Spark. *IEEE Access*, *9*, 16866-16877.

[48] Florea, C., Florea, L., Badea, M. S., Vertan, C., & Racoviteanu, A. (2019, September). Annealed Label Transfer for Face Expression Recognition. In *BMVC* (p. 104).

[49] Zhao, Y., Xu, K., Wang, H., Li, B., Qiao, M., & Shi, H. (2021). MEC-Enabled Hierarchical Emotion Recognition and Perturbation-Aware Defense in Smart Cities. *IEEE Internet of Things Journal*.

[50] Lopez-Rincon, A. (2019, February). Emotion recognition using facial expressions in children using the NAO Robot. In *2019 International Conference on Electronics, Communications and Computers (CONIELECOMP)* (pp. 146-153). IEEE.

[51] Yu, G. (2021). Emotion Monitoring for Preschool Children Based on Face Recognition and Emotion Recognition Algorithms. *Complexity*, *2021*.

[52] Wang, W., Xu, K., Niu, H., & Miao, X. (2020). Emotion Recognition of Students Based on Facial Expressions in Online Education Based on the Perspective of Computer Simulation. *Complexity*, *2020*.

[53] Witherow, M. A., Samad, M. D., & Iftekharuddin, K. M. (2019, September). Transfer learning approach to multiclass classification of child facial expressions. In *Applications of Machine Learning* (Vol. 11139, p. 1113911). International Society for Optics and Photonics.

[54] Farzaneh, A. H., Kim, Y., Zhou, M., & Qi, X. (2019, June). Developing a deep learning-based affect recognition system for young children. In *International Conference on Artificial Intelligence in Education* (pp. 73-78). Springer, Cham.

[55] Awatramani, J., & Hasteer, N. (2020, October). Facial Expression Recognition using Deep Learning for Children with Autism Spectrum Disorder. In *2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)* (pp. 35-39). IEEE.

[56] Lin, Q., He, R., & Jiang, P. (2020). Feature Guided CNN for Baby‟s Facial Expression Recognition. *Complexity*, *2020*.

[57] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli, "Multimodal fusion for multimedia analysis: a survey," *Multimed. Syst.*, vol. 16, no. 6, pp. 345–379, 2010.

[58] U. G. Mangai, S. Samanta, S. Das, and P. R. Chowdhury, "A survey of decision fusion and feature fusion strategies for pattern classification," *IETE Tech. Rev.*, vol. 27, no. 4, pp. 293–307, 2010.

[59] 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.

[60] A. Dhall, A. Asthana, R. Goecke, and T. Gedeon, "Emotion recognition using PHOG and LPQ features," in *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, 2011, pp. 878–883.

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

[62] M. Oussalah and S. Wang, "Fuzzy emotion recognition model for video sequences," in *Image Processing Theory, Tools and Applications (IPTA), 2012 3rd International Conference on*, 2012, pp. 127–132.

[63] 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.

[64] 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.

[65] 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.

[66] 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.

[67] 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.

[68] Li, H., Sun, J., Xu, Z., & Chen, L. (2017). Multimodal 2D+3D Facial Expression Recognition with Deep Fusion Convolutional Neural Network. IEEE Transactions on Multimedia, 19(12), 2816–2831. doi:10.1109/tmm.2017.2713408

[69] Kim, H.-R., Kim, Y.-S., Kim, S. J., & Lee, I.-K. (2018). Building Emotional Machines: Recognizing Image Emotions through Deep Neural Networks. IEEE Transactions on Multimedia, 1–1. doi:10.1109/tmm.2018.2827782

[70] 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.

[71] S. Zhang, S. Zhang, T. Huang and W. Gao, "Speech Emotion Recognition Using Deep Convolutional Neural Network and Discriminant Temporal Pyramid Matching," in *IEEE* 92 *Transactions on Multimedia*, vol. 20, no. 6, pp. 1576-1590, June 2018, doi: 10.1109/TMM.2017.2766843.

[72] J. Zhao, X. Mao and L. Chen, "Learning deep features to recognise speech emotion using merged deep CNN," in IET Signal Processing, vol. 12, no. 6, pp. 713-721, 8 2018, doi: 10.1049/iet-spr.2017.0320.

[73] B. Yang, J. Cao, R. Ni and Y. Zhang, "Facial Expression Recognition Using Weighted Mixture Deep Neural Network Based on Double-Channel Facial Images," in IEEE Access, vol. 6, pp. 4630-4640, 2018, doi: 10.1109/ACCESS.2017.2784096.

[74] P. Tzirakis, G. Trigeorgis, M. A. Nicolaou, B. W. Schuller and S. Zafeiriou, "End-to-End Multimodal Emotion Recognition Using Deep Neural Networks," in IEEE Journal of Selected Topics in Signal Processing, vol. 11, no. 8, pp. 1301-1309, Dec. 2017, doi: 10.1109/JSTSP.2017.2764438.

[75] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[76] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[77] Ding, S. F., Qi, B. J., & Tan, H. Y. (2011). An overview on theory and algorithm of support vector machines. *Journal of University of Electronic Science and Technology of China*, *40*(1), 2-10.

[78] Tian, Y., Shi, Y., & Liu, X. (2012). Recent advances on support vector machines research. *Technological and economic development of Economy*, *18*(1), 5-33.

[79] Song, Y. Y., & Ying, L. U. (2015). Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry*, *27*(2), 130.

[80] Liu, H., Cocea, M., & Ding, W. (2017, July). Decision tree learning based feature evaluation and selection for image classification. In *2017 International Conference on Machine Learning and Cybernetics (ICMLC)* (Vol. 2, pp. 569-574). IEEE.

[81] Cuimei, L., Zhiliang, Q., Nan, J., & Jianhua, W. (2017, October). Human face detection algorithm via Haar cascade classifier combined with three additional classifiers. In *2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI)* (pp. 483-487). IEEE.

[82] Shaheen, F., Verma, B., & Asafuddoula, M. (2016, November). Impact of automatic feature extraction in deep learning architecture. In *2016 International conference on digital image computing: techniques and applications (DICTA)* (pp. 1-8). IEEE.

[83] Li, B., & Lima, D. (2021). Facial expression recognition via ResNet-50. *International Journal of Cognitive Computing in Engineering*, *2*, 57-64.

[84] Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

[85] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, *115*(3), 211-252.

[86] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*, 85-117.

[87] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, *115*(3), 211-252.

[88] Tammina, S. (2019). Transfer learning using vgg-16 with deep convolutional neural network for classifying images. *International Journal of Scientific and Research Publications (IJSRP)*, *9*(10), 143-150.

[89] Ramalingam, S., & Garzia, F. (2018, October). Facial expression recognition using transfer learning. In *2018 International Carnahan Conference on Security Technology (ICCST)* (pp. 1-5). IEEE.

[90] Akhand, M. A. H., Roy, S., Siddique, N., Kamal, M. A. S., & Shimamura, T. (2021). Facial Emotion Recognition Using Transfer Learning in the Deep CNN. *Electronics*, *10*(9), 1036.

[91] Scherer, D., Müller, A., & Behnke, S. (2010, September). Evaluation of pooling operations in convolutional architectures for object recognition. In *International conference on artificial neural networks* (pp. 92-101). Springer, Berlin, Heidelberg.

[92] Agarap, A. F. (2017). An architecture combining convolutional neural network (CNN) and support vector machine (SVM) for image classification. *arXiv preprint arXiv:1712.03541*.

[93] Hearst, M. A., Dumais, S. T., Osuna, E., Platt, J., & Scholkopf, B. (1998). Support vector machines.

IEEE Intelligent Systems and their applications, *13*(4), 18-28.