Heaven's Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Rajshahi University of Engineering & Technology, Bangladesh

Facial Expression Recognition Based on LBP and CNN: A Comparative Study Using SVM Classifier

Author

Mir Mursalin Hossain

Roll No. 133047

Department of Computer Science & Engineering

Rajshahi University of Engineering & Technology

Supervised by

Dr. Md. Rabiul Islam

Professor

Department of Computer Science & Engineering

Rajshahi University of Engineering & Technology

ACKNOWLEDGMENT

At first I would like to pay my heartiest gratitude to almighty Allah who give me the scope and enthusiasm for successful completion of my thesis work.

I would like to express my special appreciation and thanks to my respected teacher and thesis supervisor **Prof. Dr. Md. Rabiul Islam**, Head, Department of Computer Science & Engineering, Rajshahi University of Engineering & Technology. For his relentless patience, constant inspiration, motivation and friendly guidance during the progress of work I have been able to finish this thesis. His advice and encouragement on both research as well as on my career has been priceless.

I am also very thankful to our respected teachers. My deepest sense of gratitude for their valuable suggestions, extending facilitations and inspiration in the successful completion of my thesis work. I would like to thank all the officers and staffs of Department of Computer Science & Engineering.

Finally, I express my cordial thanks and gratitude to my beloved parents, friends and well-wishers for their blessings and constant support throughout the work.

November, 2018 Mir Mursalin Hossain

RUET, Rajshahi

Heaven's Light is Our Guide

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Rajshahi University of Engineering & Technology, Bangladesh

CERTIFICATE

*This is to certify that this thesis report entitled "**Facial Expression Recognition Based on LBP and CNN: A Comparative Study Using SVM Classifier**" submitted by **Name: Mir Mursalin Hossain**, **Roll: 133047** in partial fulfillment of the requirement for the award of the degree of Bachelor of Science in Computer Science & Engineering of Rajshahi University of Engineering & Technology, Bangladesh is a record of the candidate own work carried out by him under my supervision. This thesis has not been submitted for the award of any other degree.*

Supervisor

_____

Dr. Md. Rabiul Islam
Professor
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology
Rajshahi-6204

External Examiner

_____

Suhrid Shakhar Ghosh
Lecturer
Department of Computer Science & Engineering
Rajshahi University of Engineering & Technology
Rajshahi-6204

ABSTRACT

Facial Expression Recognition are the machine learning problem which aims to improve human-computer interaction, data-driven animation and many more. Many methods are being implemented for this problem but the overall performance yet not satisfactory. In this thesis work, Local Binary Pattern (LBP) and Convolutional Neural Network (CNN) is implemented as feature extractor for Support Vector Machine (SVM) classifier. LBP has many promising performance on texture classification and CNN is the current hype for solving machine learning problem. SVM is one of the most favorite classifier for classification and with correct feature extraction, it gives amazing result. This work aim to highlight on using this method jointly to find their performance for facial expression recognition.

Experiment is conducted on four datasets for both LBP based SVM and CNN based SVM. Different training, validation and testing set is used to get overview of their overall performance. The comparative discussion of implementing this method with same structure of experiment highlights their possibilities and limitations for automated facial expression recognition system.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

The human face can express many emotions without saying a word. It is one of the most powerful and natural means for human beings to communicate their emotions. And unlike different forms of nonverbal communication, facial expressions are universal. In this era of technology, Automatic facial expression recognition and analysis is an interesting and challenging problem which has enormous impact on automation of modern society.

1.2 Overview

Facial expressions and our actions are non-verbal means of communication which comprise of 93% human communicating emotions, of which facial gestures and human actions have 55% role [1]. Facial expressions represent our emotion in our face. So, human emotions can be easily analyzed through face image and computers can be able to interact naturally with the user, in the same way that humans interact with other humans. That's why there has been a growing interest in this field.

Ekman and Friesen [2] carried out research that indicates facial expressions are universal and innate in all race, gender and age. Adding neutral emotion there are seven basic emotions. These include neutral, anger, disgust, fear, happiness, sadness and surprise. Though present work defines 21 distinct emotion categories [3], all other emotions are a result of the heterogeneity of these emotions.

1.3 Types of Facial Expression:

Basic facial expressions are discussed below: [4]

- **Neutral:** A neutral or blank expression is a facial expression characterized by neutral positioning of the facial features which indicates lack of strong emotion.
- **Anger:** This is a universal emotion which is usually demonstrated by eyebrows squeezed together, forming a crease, with eyelids tight and straightened.
- **Disgust:** An universal emotion generally expressed by pulling eyebrows down, wrinkling nose tightening the lips.
- **Fear:** Fear is generally expressed with widened eyes and slanted eyebrows that go upward. One's mouth is usually slightly open.
- **Happiness:** An universal emotion expressed with a smile and crescent-shaped eyes that may be demonstrated even by infants.
- **Sadness:** Sadness is an universal emotion usually expressed by a frown and upward slanting of the eyebrows. It is usually related with feelings of helplessness and loss.
- **Surprise:** Surprise is generally demonstrated by widened eyes and a gaping mouth. This emotion is also braced to shock and fear.

1.4 Application

Facial expression recognition has applications in different sectors. For example:

- **Education:** Real-time learner responses and engagement to the educational content is a great source of measurement for effectiveness of lecture.
- **Marketing:** It is a great way for the business companies to analyze how customers respond to their ads, products, packaging and store design.
- **Gaming:** With the introduction of virtual reality gaming is close to real life experience. Facial expression recognition can play a vital role to improve the gaming experience.

- **Security:** It can help to identify suspicious behavior in crowd and can be used to pre-emptively stop criminals and potential terrorists.
- **Health-care:** It can be helpful in the automation of medical service. Both physical and mental health can be analyzed through this application.
- **Customer Service:** Manging customer service can be more effective using facial expression recognition system. Analyzing customer feedback and computer response will ensure human computer interaction in real life.

Automotive facial expression recognition system is gradually been introduced in various application of computer science, engineering, psychology and neuroscience. It has many other application in paralinguistic communication, clinical psychology, psychiatry, neurology, pain assessment, lie detection, intelligent environments and multimodal human computer interface (HCI). There have been many reports like MarketsandMarkets [5] which estimate the emotion detection and recognition market to grow from US$6.72 billion in 2016 to $36.07 billion by 2021, at a compound annual growth rate (CAGR) of 39.9 percent from 2016 to 2021.

1.5 Objectives

The objectives of this research is to focus on comparative study of performance using Support Vector Machine (SVM) as classifier for Local binary Pattern (LBP) and Convolutional Neural Network (CNN) feature extraction technique. It will give us an overview of the effectiveness of these methods on various dataset of facial expression image. In this research, matters that are considered are the size and type of dataset, performance and drawbacks of the methods and potential improvement suggestions.

1.6 Thesis organization

Organization of this thesis work is divided into the following chapters. It highlights the structure of the thesis representation.

Chapter 1: Introduction - This Chapter highlights on a brief view of automotive facial expression recognition system.

Chapter 2: Literature Review - It holds current knowledge including theoretical findings and methodological contributions on title topic.

Chapter 3: Background Study - This Chapter includes common structure of facial expression recognition system, discussion about datasets and different methodology.

Chapter 4: Comparative Study of LBP and CNN on SVM classifier - This Chapter holds the details of this two methodology for facial expression recognition system. Their benefits and drawbacks are discussed here.

Chapter 5: Implementation - This Chapter broadly discuss the implementation technique of both methods. The environmental setup, tools, pre-processing, feature extraction and classification details are found here.

Chapter 6: Result & Performance Analysis - This Chapter includes analytical data of performance and result of implemented method of facial expression recognition system.

Chapter 7: Conclusion & Observation - This Chapter contains a brief discussion of total working procedure, its limitation and future work.

Chapter 2

Literature Review

2.1 Introduction

The study of facial expression recognition had been an attractive field for recent years in the area of human computer interaction. Though many improvement has been achieved recognizing facial expression with a high accuracy remains difficult due to the subtlety, complexity and variability of facial expressions. Various approaches has been introduced to achieve better performance and broaden our outlooks. Now, here facial expression recognition related previous works tied to this thesis title will be discussed.

2.2 Related Works

$\Rightarrow$ Caifeng Shan, Shaogang Gong, Peter W. McOwan published an comprehensive study on facial expression recognition based on local binary patterns. [6]

Contributions:

- Comprehensive empirical study of facial expression recognition based on local binary patterns features was performed.
- Different machine learning methods were systematically examined on several facial expression databases.
- Boosted-LBP was formulated to extract the most discriminant LBP features.

- LBP features were investigated for low-resolution facial expression recognition.

Limitations:

- Experiment were performed on posed facial expression dataset only.
- No detailed explanation was given about the step by step process for the findings of the experimented result.
- Good accuracy is only found using proposed method when test is performed on same dataset by which it was trained.

⇒ Dan Duncan, Gautam Shine, Chris English proposed transfer learning on the fully connected layers of an existing pretrained convolutional neural network. [7]

Contributions:

- Variety of datasets as well as their own unique image dataset was used to train the model.
- Effectiveness of their method was showed for live video stream.
- A system was evolved for detecting human emotions in different scenes, angles, and lighting conditions in real-time.
- Classification accuracy of VGG_S Convolutional Neural Network (CNN) was improved by using transfer learning.

Limitations:

- Large difference was found in train and test accuracy.
- Their method works on gray scale image but their pretrained convolutional neural network used RGB image for training.
- Subtle difference between the emotions 'disgust' and 'fear' led to downfall of performance.

⇒ Henry Medeiros, Valfredo Pilla Jr, Andr'e Zanellato, Cristian Bortolini suggested using CNN for feature extraction and SVM for classification. [8]

Contributions:

- The relatively new application of Convolutional Neural Network (CNN) as feature extractor is used.
- It will be helpful to find the effectiveness of deep neural network as feature extractor.
- A pretrained Alexnet CNN was used which permitted the use of a small dataset to train the SVM classifier only.

Limitations:

- Method was applied only on single Extended Cohn-Kanade public dataset.
- Only three emotional states: Aversion, Happiness and Fear are considered for experiment.
- As pretrained CNN was trained using Alexnet which uses RGB image, three channel representation is obtained by replicating the image into the other two channels.

⇒ Yichuan Tang proposed deep learning using linear support vector machines. [9]

Contributions:

- For Classification consistent advantage of replacing the softmax layer with a linear support vector machine was demonstrated.
- Minimization of learning was seen in margin-based loss instead of the cross-entropy loss.
- Proposed model was implemented on various dataset to give an clear overview of performance of the model.

Limitations:

- This experiment is shown mainly to demonstrate the effctiveness of the last linear SVM layer vs the softmax.
- Exploration of other commonly used tricks such as Dropout, weight constraints, hidden unit sparsity, adding more hidden layers and increasing the layer size was not shown.

⇛ Felix Juefei-Xu, Vishnu Naresh Boddeti, Marios Savvides proposed local binary convolutional neural networks. [10]

Contributions:

- The LBC layer is shown to afford significant parameter savings, 9x to 169x in the number of learnable parameters compared to a standard convolutional layer.
- CNNs with LBC layers, called local binary convolutional neural networks (LBCNN) was seen to be achieving performance parity with regular CNNs on a range of visual datasets.
- The lower model complexity of LBCNN is found to be attractive option for learning with low sample complexity.

Limitations:

- The idea came from complete binarization of CNNs which eads to performance loss in comparison to real-valued network weights.
- Different parameters and configurations of the LBP formulation can result in drastically different feature descriptors which leads to ups and downs of performance.

2.3 Conclusion

Works in this thesis book are closely related to the above works in the field of facial expression recognition.

Chapter 3

Background

3.1 Introduction

Facial expression recognition has a vast area of application in real life situations. Various machine learning methods have been applied to achieve better performance. Similar to all other machine learning problem it has three general solving steps which are data pre-processing, feature extraction and classification. However it is necessary to discuss the algorithms and techniques used in these steps for facial expression recognition. To keep our overview of background knowledge more centered to our work we will only discuss about facial expression recognition in sill images which contains facial expression of single person. We will also discuss about available datasets for this purpose.

3.2 Face Detection & Data pre-processing

Detecting face in the input image and image pre-processing for various requirement is the starting step of facial expression recognition. Successful implementation of this step is crucial for better accuracy.

3.2.1 Face Detection

Face detection in still images is an important problem in machine vision and is often the first step in many vision based application that involve interaction between human and machine. It can be formulated as follows. When an arbitrary image is given, the goal of face detection is to determine whether or not there are any faces in the image and if present, return the location and

extent of each face. There are several face-detection techniques which are broadly classified as knowledge-based, feature-based, template-matching, appearance-based, part-based etc [11].

Knowledge-based methods is based on rules related to human facial features. Random labeled graph matching and Color Information based techniques are feature-based. In template-matching-based methods standard face pattern is utilized and the correlation values with the standard patterns are computed for the face contour, eyes, nose and mouth independently on a given input image. Face detection in appearance-based methods is generally depends on finding the differences between face and non-face patterns.

3.2.2 Image pre-processing

Given face image is needed to be pre-processed for the next steps. Detected face area may needed to be cropped. Face alignment may be necessary. Depending on the required size of the image for next image given image may needed to be resized. Nearest-neighbor interpolation, bilinear, bi-cubic etc. are popular choice of algorithm for resizing. Color adjustment and re-sampling can be used for illumination balance. Image conversion to different format or type and image segmentation may be necessary according to the requirement of feature extraction and classification method.

3.3 Feature Extraction

In machine learning, pattern recognition and image processing, feature extraction starts from an initial set of measured data and builds derived values intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps and in some cases leading to better human interpretations. Feature extraction is related to dimensionality reduction.

When the input data to an algorithm is too large to be processed and is suspected to be redundant, then it can be transformed reduced set of features. This process is called feature selection. The selected features are expected to certain the relevant information from the input data so that desired task can be performed by using this reduced representation instead of complete initial data. In Facial expression recognition, extracted features are then feed to classification steps.

3.3.1 Common Feature Extraction Methods

There are two common approaches to extract facial features: geometric feature-based methods and appearance-based methods [12].

In the geometric feature-based approach the primary step is to localize and track a dense set of facial points in the image. Most geometric feature-based approaches use the active appearance model (AAM) or its different variations, to track a dense set of facial points. The locations of these facial landmarks are then used in different ways to extract the shape of facial features and movement of facial features, as the expression evolves [13]. However, the geometric feature-based methods generally requires accurate and reliable facial feature detection and tracking, which is difficult to accommodate in many situations. Gauss–Laguerre wavelet textural feature fusion, Image-Ratio feature selection, Multiple Image Characterization Techniques etc. are example of this methods.

Appearance based features describe the change in face texture when particular action is performed such as wrinkles, bulges, forefront, regions surrounding the mouth and eyes. Image filters are used and applied to either the whole face or specific regions in a face image to extract a feature vector [14]. Principal Component Analysis (PCA), Local Gabor Filter Bank with PCA plus LDA, Local Directional Pattern Variance (LDPV), Sparse Representation with Multiple Gabor filters, selected facial patches, Local Binary Patterns techniques are example of this methods.

3.3.2 Other Feature Extraction Methods

With many study of deep learning in recent years, there have been research going on using deep learning methods as feature extractor with other classification methods. Researchers [15] found deep learning method like Convolutional Neural Network (CNN) can be used as feature extractor for various application [16] [17]. Well it turns out Donahue et al. [18], Zeiler and Fergus [19] and Oquab et al. [20] have suggested that generic features can be extracted from large CNN and provided some initial evidence to support this claim. For example, removing the softmax as the classifier from CNN's last layer will make CNN as feature extractor which then can be applied in classifier like Support Vector Machine (SVM).

3.4 Classification

Classification is a field of research to classify things/objects/images/sound/text etc. using machine learning/Statistical Learning techniques. In machine learning, classification is a technique comes under supervised learning and it has significant difference with other learning techniques like regression and clustering. There are three major machine learning techniques: supervised learning, unsupervised learning and semi-supervised learning.

In supervised learning, there are prior knowledge about input dataset and it's corresponding label. So, machine is trained first and then based on that knowledge it tries to find the the label of the input. Classification and regression are supervised learning method. The difference between classification and regression is in classification prediction value tends to be category but in regression prediction value tends to be a continuous.

In unsupervised learning, there are no prior knowledge about input data and machine are left to their own devises to discover and present the interesting structure in the data. Unsupervised learning problems can be grouped into clustering and association problems. In clustering problem, inherent groupings of the data is to be discovered but association problems, describing large portions of data by rules are necessary.

Semi-supervised learning problems are where large amount of input data with only some of labeled data are provided. Challenge is to find label of rest of data and new input data.

As in facial expression recognition problem, we have categorical data and we first want to train our machine by labeled data and then find label of new input data, it is a classification problem. We will discuss further about classification below.

3.4.1 Common Classification Algorithms

⇒ **Logistic Regression:**

It is a machine learning algorithm for classification where the probabilities describing the possible outcomes of a single trial are modeled using a logistic function. It is a statistical method

for analyzing a data set in which there are one or more independent variables that determine an outcome for that data.

- **Advantages:** Logistic regression is designed for classification), and is most useful for understanding the influence of several independent variables on a single outcome variable.
- **Disadvantages:** But it works only when the predicted variable is binary, assumes all predictors are independent of each other, and assumes data is free of missing values.

⇛ **Naive Bayes Classifier:**

It is a classification technique based on Bayes Theorem with an assumption of independence between every pair of features. This algorithm requires a small amount of training data to calculate the necessary parameters.

- **Advantages:** Naive Bayes classifiers are extremely fast compared to more sophisticated methods.
- **Disadvantages:** But Naive Bayes is is known to be a bad estimator.

⇛ **k-Nearest Neighbor:**

The k-nearest-neighbors algorithm is a classification algorithm which takes a bunch of labeled points and uses them to learn how to label other points. To label a new point, it searches at the labeled points closest to that new point.

- **Advantages:** This algorithm is simple to implement, robust to noisy training data, and effective for large training dataset.
- **Disadvantages:** Need to determine the value of no of neighbors and the computation cost is high.

⇛ **Decision Trees:**

it builds a tree like structure. When a dataset of attributes together with its classes is given, a decision tree produces a sequence of rules that can be used to classify the data.

- **Advantages:** Decision trees can handle both categorical and numerical data.
- **Disadvantages:** Decision tree can create complex trees that do not generalize well and unstable.

⇛ **Random Forests:**

Random forest classifier is a meta-estimator that fits a number of decision trees on various sub-samples of datasets and uses average to improve the predictive accuracy of the model and controls over-fitting.

- **Advantages:** Reduction in over-fitting and random forest classifier is generally more accurate than decision trees in most cases.
- **Disadvantages:** It has slow real time prediction and it is difficult to implement and complex algorithm.

⇛ **Support Vector Machine:**

Support vector machine is a representation of the training data as points in space separated into categories by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the line they fall.

- **Advantages:** In high dimensional spaces, it is effective and uses a subset of training points in the decision function so it is also memory efficient.
- **Disadvantages:** It does not directly provide probability estimates.

⇛ **Neural Networks:**

A neural network consists of units (neurons), arranged in layers, which convert an input vector into some output. Each unit takes an input, applies a function to it and then passes the output on to the next layer and final layer gives output. It is modeled according our brain structure.

- **Advantages:** Scales well to larger dataset and various type of input dataset.
- **Disadvantages:** It has hardware dependency and hard to explain the cause of the result.

3.5 Datasets

A facial expression dataset is a collection of images or video clips with facial expressions of a range of emotions. Well-labeled media content of facial behavior is essential for training, testing and validation of algorithms for the development of expression recognition systems.

There are a number of datasets for facial expression recognition. Many contains both images and videos. Dataset can be either posed or non-posed. Some of them are race, age and gender specific. Some dataset contains facial expression image containing beard, eyeglass and makeup. Dataset may have only frontal or both frontal and side-viewed facial expression images. Many dataset have more or few labeled images other than our six basic emotions.

For our work purpose we need frontal face image with all six basic emotion labels with no beard, eyeglass and makeup. Below four are publicly available facial expression database which can be used for our works.

$\Rightarrow$ **Extended Cohn-Kanade (CK+) Dataset:** [21]
It is an extended version of Cohn-Kanade (CK) database published in 2000. It's extended version with emotion and AU labels, along with the extended image data and tracked landmarks was published 2010. It is one of the most common dataset for facial expression recognition. It also contains contempt labeled emotion with six basic emotions and neutral.

- **Type:** Posed, Grown Male & Female of all race
- **Resolution:** $640 * 490$
- **Color:** RGB
- **Number of subjects:** 123
- **Number of images:** 593
- **No of Facial expressions:** 08
- **Facial expression:** neutral, anger, disgust, fear, happiness, sadness, surprise and contempt

Figure 3.1: Example image of CK+ Dataset

⇛ **Japanese Female Facial Expression (JAFFE) Dataset:** [22]

It is another popular dataset become available in 1998. Only Japanese female facial expressors expressed in total of 7 facial expression which was then rated on 6 emotion adjectives by 60 Japanese subjects. The photos were taken at the Psychology Department in Kyushu University.

- **Type:** Posed, 10 Japanese grown female models
- **Resolution:** $256 * 256$
- **Color:** Gray
- **Number of subjects:** 10
- **Number of images:** 213
- **No of Facial expressions:** 07
- **Facial expression:** neutral, anger, disgust, fear, happiness, sadness and surprise

Figure 3.2: Example image of JAFFE Dataset

⇒ **The Karolinska Directed Emotional Faces (KDEF) Dataset:** [23] [24]

The Karolinska Directed Emotional Faces (KDEF) is a set of totally 4900 pictures of human facial expressions of emotion. The material was developed in 1998. The set contains 70 individuals, each displaying 7 different emotional expressions, each expression being photographed (twice) from 5 different angles.

- **Type:** Posed, 35 male and 35 female of age between 20-30
- **Resolution:** $562 * 762$
- **Color:** RGB
- **Number of subjects:** 70
- **Number of images:** 4900
- **No of Facial expressions:** 07
- **Facial expression:** neutral, anger, disgust, fear, happiness, sadness and surprise

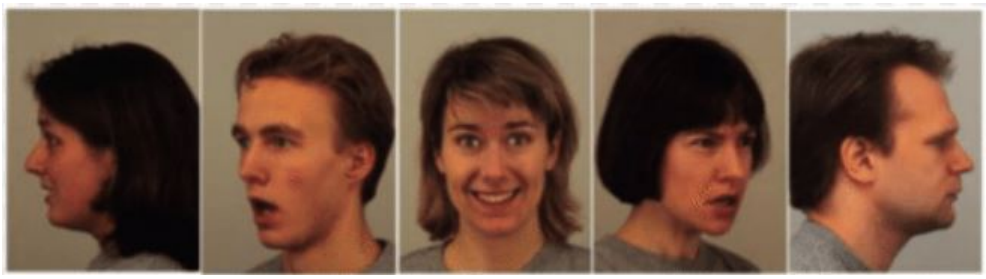


Figure 3.3: Example image of KDEF Dataset

⇒ **FER2013 Dataset:** [25]

The dataset consists of 48x48 pixel grayscale images of faces. it is actually published first for a kaggle competition of Facial Expression Recognition Challenge. Most of the images were collected from internet which are freely available and then labeled manually. So, it is a non-posed dataset.The training set consists of 28,709 examples. The public test set has 3,589 examples. The final test set consists of another 3,589 examples. Dividing dataset with training and test set is significant as it is absent in other datasets. It has also lower resolution than other datasets.

- **Type:** Non-Posed
- **Resolution:** $48 * 48$
- **Color:** Gray
- **Number of subjects:** unknown
- **Number of images:** 4900
- **No of Facial expressions:** training - 28709, public test - 3589, private test - 3589
- **Facial expression:** neutral, anger, disgust, fear, happiness, sadness and surprise

Figure 3.4: Example image of FER2013 Dataset

3.6 Conclusion

Background study of the thesis work has profound impact on developing deeper knowledge about the subjected topic. it is a way to dive in theoretical and experimental work for the thesis.

Chapter 4

Comparative Study of LBP and CNN baseed on SVM classifier

4.1 Introduction

There are works showing the effectiveness of Local Binary Pattern (LBP) and Convolutional Neural Network (CNN) as feature extractor. C. Shan, S. Gong and P. W. McOwan showed did a comprehensive study on some feature extractor for facial expression recognition system. [6] They found LBP feature extractor on SVM classifier and found good result for CK+ and JAFFE dataset. However they have not showed their effectiveness on other dataset.

Again, using Convolutional Neural Network (CNN) as feature extractor with SVM classifier has been used in application like image classification. [17] So, it's effectiveness for facial expression recognition is an ongoing research. This thesis work shows the comparative study of LBP based SVM and CNN based SVM with same dataset and following same procedure for image processing and classification. So, the comparative effectiveness of both method will be clear.

4.2 Work Flow

Work flow is the summary of procedure or steps taken for a work. It helps to understand the whole work. The below diagram shows the thesis work flow.

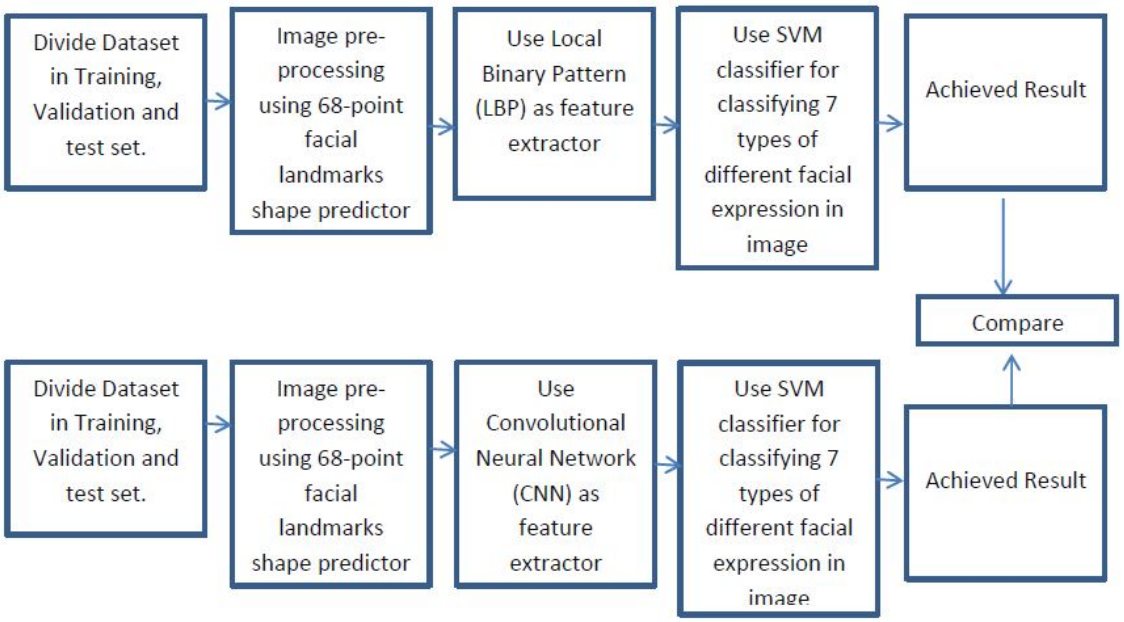


Figure 4.1: Thesis work flow diagram

For our thesis work, datasets, image pre-processing method, classifier's parameters need to be specific. Otherwise, result will not show ideal comparison. Deeper knowledge of both feature extracting method of Local Binary Pattern (LBP) and Convolutional Neural Network (CNN) is also necessary.

4.3 Training, validation and testing dataset

We can use Extended Cohn-Kanade (CK+), Japanese Female Facial Expression (JAFFE), Karolinska Directed Emotional Faces (KDEF) and FER2013 Dataset for our purpose. However, the resolution of FER2013 dataset is only $48 * 48$. So, it can be used to measure the performance of both method for low illumination. It has also dedicated public and private dataset which can be used for validation and testing.

All CK+, JAFFE, KDEF has higher resolution. We can resize all of them to $256 * 256$ resolution while cropping for face image. So, computation cost will be lower. JAFFE and FER2013 dataset has Grayscale image. CK+ has both RGB and Grayscale image. KDEF contains only RGB image. For simplicity all of them is needed to be in Grayscale image.

Moreover, CK+ dataset has in total 8 emotion label. But we want to detect only 7 emotion label. So, CK+ dataset was modified. KDEF is also modified as it has only 490 image with frontal face image without repetition.

⇛ **Used training, validation and testing dataset:**
For CK+, JAFFE and KDEF there is no validation set and testing set given. So, training, validation and testing dataset is choosed among them. With most number of images among them KDEF is choosed as training set while JAFFE was set as validaion set and CK+ as testing set.

On the other hand, FER2013 has very lower resolution comparative to these datasets. So, another experiment is done with this dataset with it's training, validation and testing set. For both of the dataset settings same steps are taken and comparison on result is discussed.

4.4 Facial landmark detection and processing

Facial landmark detection means facial feature detection which is also referred to as facial key point detection. To be able to facial expression recognition we first need to identify a person in an image and find where his face is located. Therefore, face detection — locating a face in an image and returning a bounding rectangle / square that contains the face is necessary for facial expression recognition. Paul Viola and Michael Jones [27] successfully introduced Viola and Jones face detector for this task.

We can find more accurate face detection when the location of different facial features can be found. Facial landmarks are used to localize and represent salient regions of the face, such as eyes, eyebrows, nose, mouth, jawline etc. Facial landmarks have been successfully applied to face alignment, head pose estimation, face swapping, blink detection, face morphing and much more.

Detecting facial landmarks is a subset of the shape prediction problem. We used the facial landmark detector included in the dlib library [26]. It is an implementation of [27] using iBUG

300-W dataset [28] to pre-train.

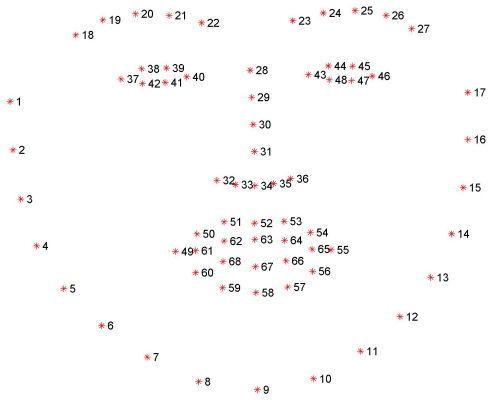


Figure 4.2: The 68 point landmarks on a face image

⇛ **Used method for facial landmark detection and image pre-processing:**
In the thesis work dlib's pre-trained facial landmark detector is used. For each of the face detections, it giving us the 68 (x, y)-coordinates that map to the specific facial features in the image. So an array with shape (68, 2) is found which helps to face detect, face alignment and choosing the boundary of cropping images. Cropped image is resized to $256 * 256$ resolution. It was also pushed along with LBP and CNN feature extractor to the classifier.

4.5 Local Binary Pattern (LBP)

Local binary patterns (LBP) is a type of visual descriptor used for classification in computer vision which was first described in 1994. [29] [30]

The primary step of constructing the LBP texture descriptor is to convert the image to grayscale. For every pixel in the grayscale image, we select a neighborhood of size r surrounding the center pixel. A LBP value of that image is then calculated for this center pixel and stored in the output 2D array with the same width and height as the input image.

For example, the operator labels the pixels of an image by thresholding a 3*3 neighborhood of each pixel with the center value and considering the results as a binary number when all

the neighborhood is in 1 pixel distance. For this there are 2^8 possible value and so 256-bin histogram of the LBP labels computed over a region is used as a texture descriptor. The derived binary numbers are called Local Binary Patterns or LBP codes (as shown in Fig. 4.3 - Fig 4.6). It codify local primitives information regarding input image.

Figure 4.3: The original face image and the cropped image [6]

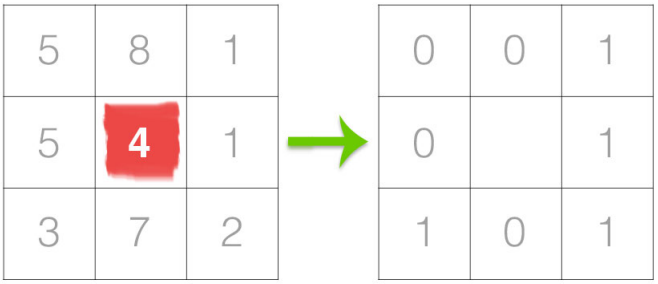


Figure 4.4: Getting LBP value from 8-pixel neighborhood [31]

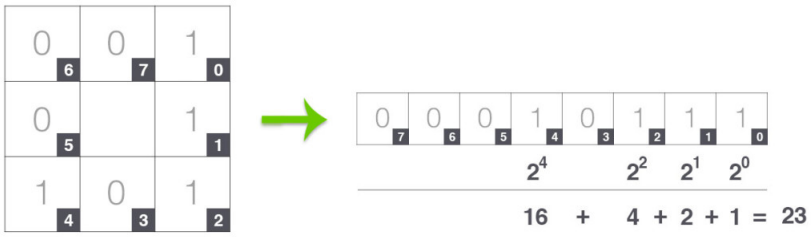


Figure 4.5: LBP with 8-bit binary neighborhood of the center pixel visualization [31]

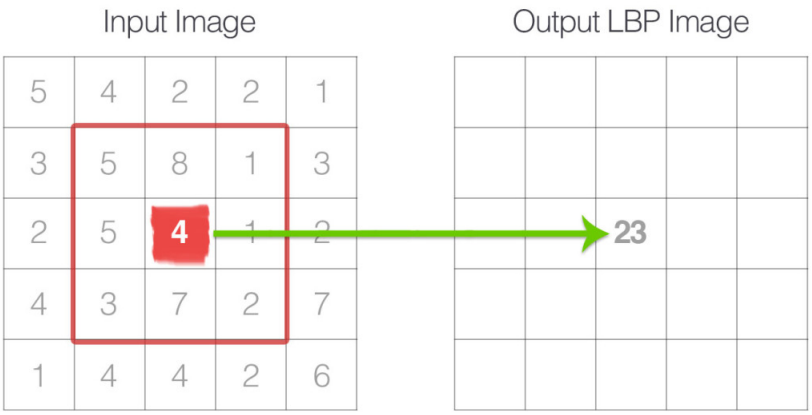


Figure 4.6: Calculated LBP value is then stored in an output 2D array [31]

The limitation of the basic LBP operator is its small 3*3 neighborhood as it can not capture necessary features with large scale. So, the operator later was extended to use neighborhood of different sizes. An extension to the original LBP implementation is popularly used which handle variable neighborhood sizes. To account for variable neighborhood sizes, two parameters were introduced. One is the number of points p in a circularly symmetric neighborhood and other one is the radius of the circle r, which allows to account for different scales.

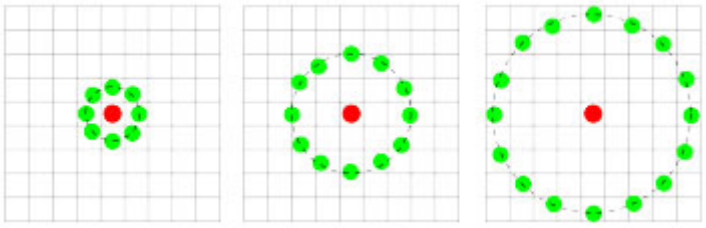


Figure 4.7: Three neighborhood examples with varying p and r used to construct circular Local Binary Patterns [31]

The LBP operator produces 2^p different output values which is often considered computational expensive for large dataset. So, the concept of LBP uniformity came to light. A LBP is considered to be uniform if it has at most two 0-1 or 1-0 transitions. For example, 00000000, 001110000 and 11100001 are uniform patterns but 00111101 is not.

There are p + 1 uniform patterns for for a LBP with p number of points when rotationally invariant features are required. When LBP features do not encode rotation information for method

non-rotational invariant uniform method, the dimension of the histogram becomes p*(p-1)+3. Rotation variation is caused by rotation of the camera or captured images. This LBP histogram contains information about the distribution of the local micro-patterns, such as edges, spots and flat areas, over the whole image, so can be used as feature extractor.

⇛ **Used method for LBP feature detection:**
In the thesis, for all dataset of image, 24 number of points with radius is choosed for circular Local Binary Pattern (LBP). Non-rotational invariant uniform method technique is used feature extraction. So an histogram of dimension 26 was used as feature extractor along with facial landmark array found in image processing method. The feature vector was normalized to have the value in range 0 to 1.

4.6 Convolutional Neural Network (CNN)

In neural networks, Convolutional neural network (ConvNets or CNN) is one of the popular way to do images recognition, images classifications. Objects detections, recognition faces etc. are some of the areas where CNN are widely used. In many research, the effectiveness of CNN as feature extractor has came to light.

Convolutional Neural Networks are inspired by the brain. Convolutional Neural Networks have a different architecture than general Neural Networks. Regular Neural Networks transform an input by putting it through a number of hidden layers. Every layer is made up of a set of neurons, where each layer is fully connected to all neurons of the previous layer. Finally, last fully-connected layer works as the output layer which represent the predictions.

Convolutional Neural Networks are a bit different. First of all, the layers are organized in 3 dimensions: width, height and depth. A normal color image as a rectangular box whose width and height are measured by the number of pixels along those dimensions. Depth layers are referred to as number of color channels of the image.

Further, the neurons in one layer do not connect to all the neurons in the next layer but only

to a small region of it. Lastly, the final output will be reduced to a single vector of probability scores, organized along the depth dimension.

CNN have two components:

- **The Hidden layers/Feature extraction part:** In this part, the network will perform a series of convolutions and pooling operations during which the features are detected. As an example if the input is a picture of a zebra, this is the part where the network would recognize its stripes, two ears and four legs.
- **The Classification part:** In this part, fully connected layers will serve as a classifier on top of these extracted features. They will assign a probability for the object on the image being what the algorithm predicts it is.

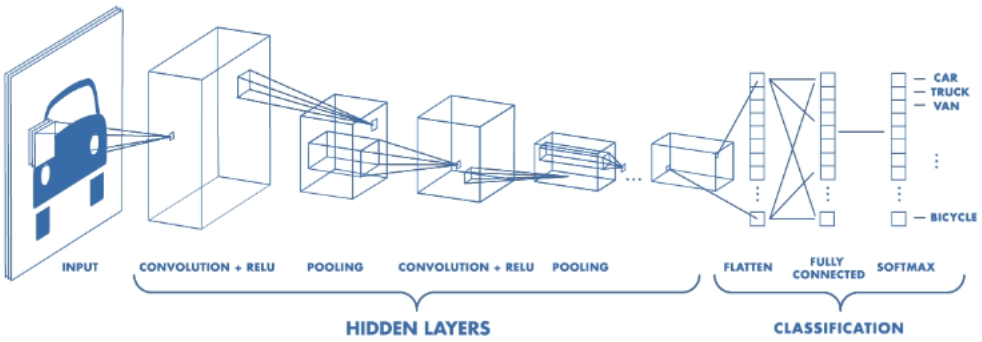


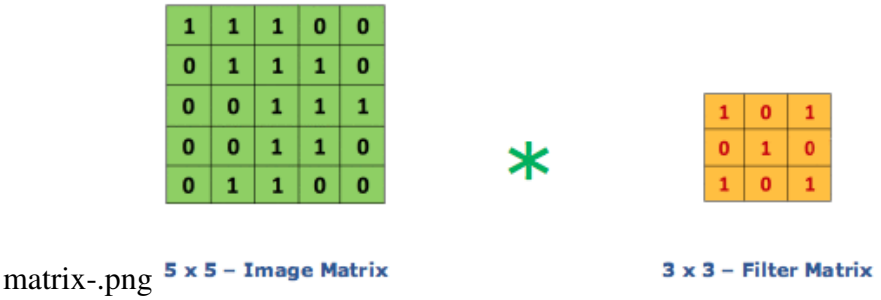
Figure 4.8: Parts of CNN with example network [32]

4.6.1 Feature extraction part

Rather than focus on one pixel at a time, a convolutional net takes in square patches of pixels and passes them through a filter. That filter is also a square matrix smaller than the image itself, and equal in size to the patch. It is also called a kernel, The job of the filter is to find patterns in the pixels. These are the important parameters for Feature extraction part in CNN.

- **Convolution Layer:** Convolution is one of the main building blocks of a CNN. The convolution is performed on the input data with the use of a filter or kernel which then

produce a feature map. We execute a convolution by sliding the filter over the input. At every location, a matrix multiplication is performed and sums the result onto the feature map.



matrix-.png

Figure 4.9: Example $5 * 5$ input image and $3 * 3$ convolution filter [33]

‘

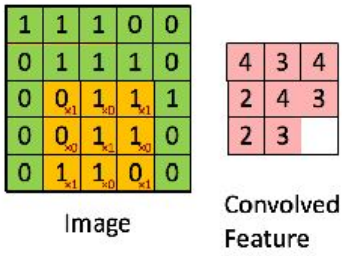


Figure 4.10: Output matrix pf image for convolved feature [33]

- **Pooling layer:** After a convolution layer, it is common to add a pooling layer in between CNN layers. It can also be called sub-sampling or down-sampling which reduces the dimensionality and computation in the network of each map but retains the important information. Spatial pooling can be of different types as like Max Pooling, Average Pooling, Sum Pooling etc. The function of pooling is to continuously reduce the dimensionality to reduce the number of parameters and computation in the network. This shortens the training time and controls over-fitting.

- **Strides:** Stride is the number of pixels shifts over the input data matrix. When the stride is 1 then the filters is moved to 1 pixel at a time. When the stride is 2 then the filters is moved to 2 pixels at a time and so on.

- **Padding:** Sometimes filter does not fit perfectly fit the input image. Then the solution is either pad the picture with zeros (zero-padding) so that it fits or drop the part of the image where the filter did not fit.

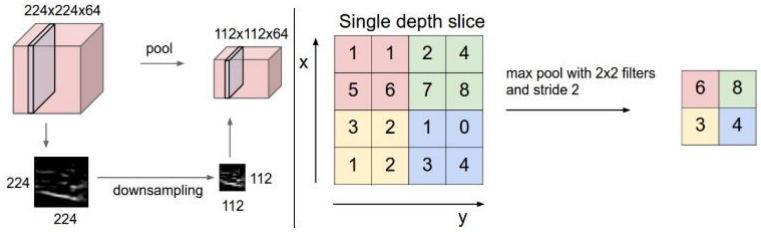


Figure 4.11: Example pooling with stride in sample data [34]

4.6.2 Classification part

After the convolution and pooling layers, classification part consists of a few fully connected layers. However, these fully connected layers can only accept 1 Dimensional data. So, input data is flatten in one vector. This part has the same principle as a regular Neural Network.

Training a CNN works in the same way as a regular neural network. There are also dropouts between some layers, the dropout layer is a regularizer that randomly sets input values to zero to avoid over-fitting. With the fully connected layers, all these features are combined together to create a model. Finally, with an activation function such as softmax or sigmoid to output is classified. Instead of using activation function like softmax or sigmoid, we can take the input of these layer and can use it in Support Vector Machine (SVM) to get our variant of model.

$\Rightarrow$ Used model for CNN to use as feature extractor:
Our input is grayscale image with $48*48$ resolution or $256*256$ resolution. the value of conv2D function declares dimensionality of output space, kernel size and activation function which can be linear or non-linear. Normalization happens by adjusting the input layer and scaling the activations. Flattening is done before entering dense layer which is fully connected layer. For the final output softmax was used.

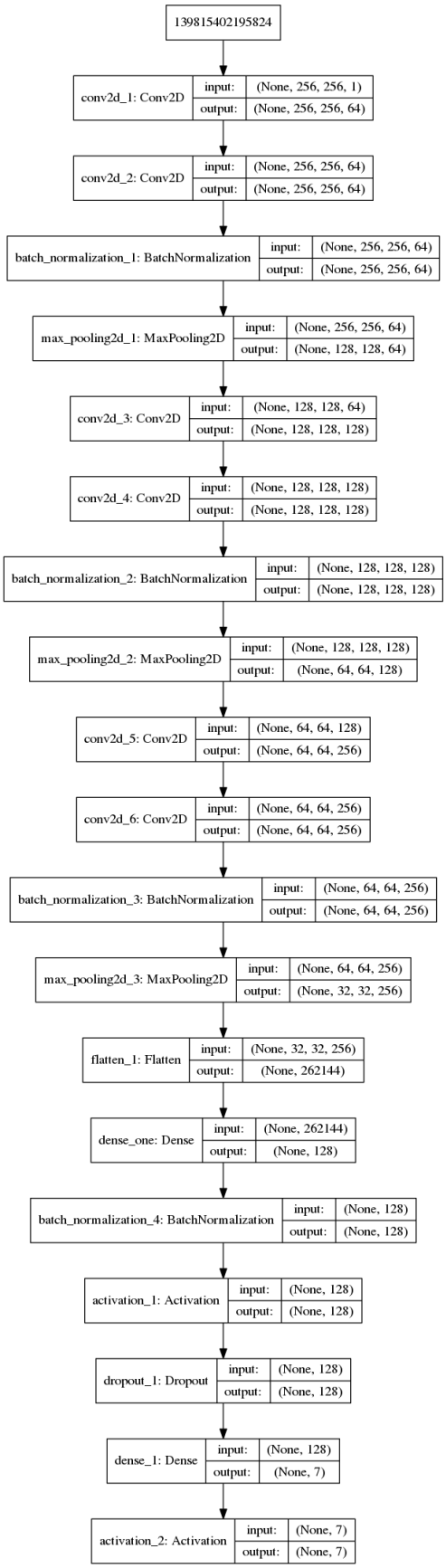


Figure 4.12: Example diagram of used CNN model

In our thesis, we took the input of first dense layer of output vector 128. We then used it as feature vector for SVM along with facial landmark array. So, our input feature vector has become of lenth 264. Which was then passed to SVM for classification.

4.7 Support Vector machine (SVM)

Support Vector Machine (SVM) is a supervised machine learning algorithm which can be used for classification challenges. It is based on the concept of decision planes that define decision boundaries. A decision plane is one that separates between a set of objects having different class memberships. In this algorithm, each data item is plotted as a point in n-dimensional space (where n is number of features) with the value of each feature being the value of a particular coordinate. Then, classification is done by finding the hyper-plane that differentiate the classes very well.

An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. Maximizing the distances between nearest data point (either class) and hyper-plane helps to decide the right hyper-plane. This distance between hyper-planes is called as Margin.

SVM is mostly useful in non-linear separation problem. It does some extremely complex data transformations based on the given labels.In SVM, it is easy to have a linear hyper-plane between these two classes. Although, extensions have been developed for regression and multi-class classification.

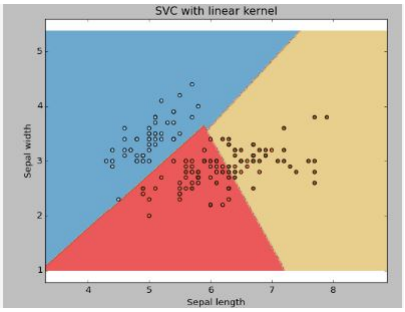


Figure 4.13: Example of application of SVM for multi-class classification [35]

It works really well with clear margin of separation and effective in high dimensional spaces. It is even usable in cases where number of dimensions is greater than the number of samples. Though It's performance drops when large data set is applied and for that the required training time is higher. It also doesn't works well when the data set has more noise i.e. target classes are overlapping.

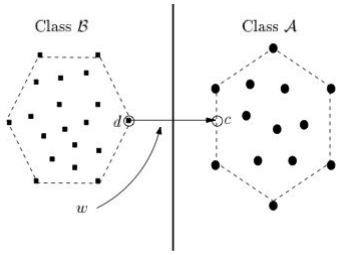


Figure 4.14: How SVM uses two closes point to determine separating plane [36]

The support vector machine searches for the closest points , which it calls the "support vectors". it is calculated "support vector machine" due to the fact that points are like vectors and that the best line is "supported by" the closest points. When SVM finds the closest points, the SVM draws a line connecting them. The support vector machine then declares the best separating line to be the line that bisects and is perpendicular to the connecting line. The support vector machine gets better because when a new sample comes as there is already a line that keeps classes as far away from each other as possible.

4.7.1 Parameters

It In practice, real data is messy and cannot be separated perfectly with a hyperplane. There are some tuning parameters can be used to get better performance which depends on dataset. This flexibility of support vector machines does come at the price of cost of computation. Here are some important parameters:

- **Kernels:** It is usually used to refer as the kernel trick. It is a method of using a linear classifier to solve a non-linear problem. It entails transforming linearly inseparable data like to linearly separable ones. The kernel function is that which is applied on each data instance to map the original non-linear observations into a higher-dimensional space in which they become separable.

It The learning of the hyperplane in linear SVM is done by transforming the problem using some linear algebra. Unfortunately with a large number of attributes in a dataset, it is difficult to know which kernel would work best. The most commonly used kernels are linear, polynomial (poly) and radial basis functions (rbf).

- **C:** It is penalty parameter of the error term. It also controls the trade off between smooth decision boundary and classifying the training points correctly.
- **gamma:** It is kernel coefficient for rbf, poly and sigmoid. Higher the value of gamma, SVM will try to exact fit the as per training data set and cause over-fitting problem.
- **max_iter:** Number of maximum iteration will be used to find the solution.
- **decision_function_shape:** It can be one vs one (ovo) and one vs rest (ovr). For multi-class problem ovr is used.

⇛ **Used method for clasification using SVM:**
We have done cross-validation on small dataset and tried to choose best parameter. For most of the parameters default value provided by SVC function in python is okay and we tuned some parameters in expectation of better performance. We fitted our training dataset found by feature extraction. SVM has done multi-class classification based on provided data and it's label.

Validation test set was used by CNN feature extractor at the time of CNN model training. Both validation and testing set is then evaluated according to the found fitted SVM model. Resulted Confusion matrix is a score visualizer that takes a fitted classifier and a set of test dataset which returns a report showing how each of the test values predicted classes compare to their actual classes.

Resulted classification report displays the precision, recall, F1, and support scores for the model. The definition of precision is the ability of a classifier not to label an instance positive that is actually negative. Whereas recall is the ability of a classifier to find all positive instances. The F1 score is a weighted harmonic mean of precision and recall for which the best score is 1.0 and the worst is 0.0. And support is the number of actual occurrences of the class in the specified dataset.

4.8 Conclusion

The detailed procedure of work flow diagram is discussed in this chapter. All the major steps taken is highlighted with background study of them. Later chapters follow the method annotated in this chapters.

Chapter 5

Implementation

5.1 Introduction

The present chapter aims to explain how the procedures for the comparison of LBP and CNN based SVM follows. Implementation is done according to work-flow diagram (Figure 4.1) discussed in previous chapter. The hardware and software specification which were used to implement this procedure are given.

5.2 Hardware Specification

Hardware Specification is the overview of hardware support to continue the experiment. As difference of hardware may make difference in the output result. Specially, time and memory complexity depends on hardware.

For the experiment purpose two types of hardware support was used. As one performed on own machine and other was performed online by creating kernels in Kaggle. [37] Kaggle gives free 6 hour to run a kernel which was used as code snippet to run large dataset for these thesis work.

Table 5.1: Local Hardware Specification

System Model	HP 15 Notebook PC
Processo r	Intel(R) Core(TM) i3-4010U CPU @ 1.70 GHz
Memory	8.00 GB
System Type	64-bit Operating Suystem, x64-based processor

Table 5.2: Kaggle Hardware Specification

System Model	Kaggle kernel environment
GPU	NVidia K80
Memory	14.00 GB
Disk	6 GB

5.3 Software Framework

Source code for experiment is run on specific software framework. Here are the summary of it.

Table 5.3: Software framework specification

Operating System	Ubuntu
Language	Python 3.6.5
Editor	JetBrains PyCharm Community Edition 2017.3.4

5.4 Input Databases

Dataset was modified. We omitted the contempt emotion label in CK+ dataset (Figure 3.1). Only frontal face image is taken from KDEF dataset (Figure 3.3). Here is the list of facial expression images for each emotion label of all four dataset used.

Table 5.4: Dataset Details

Name	Resolution	Total	Neutral	Anger	Disgust	Fear	Happiness	Sadness	Surprise
CK+	$640 * 490$	327	50	45	59	24	59	28	62
JAFFE	$256 * 256$	213	30	30	25	32	31	31	30
KDEF	$562 * 762$	490	70	7	70	70	70	70	70
FER2013	$48 * 48$	28709	4965	3995	436	4097	7215	4830	3171

There are 28709 training image in FER2013 (Figure 3.4). it has also validation and test set. Here are the emotion label list of those.

Table 5.5: FER2013 Validation and Testing Dataset Details

Name	Total	Neutral	Anger	Disgust	Fear	Happiness	Sadness	Surprise
Validation Set	3589	607	467	56	496	895	653	415
Testing Set	3589	626	491	55	528	879	594	416

5.5 Detailed discussion of the taken procedure

5.5.1 Dataset processing

CK+ dataset comes with 8 Emotion_labels, FACs_labels, Landmarks and Extended cohn-kanade-images folder. For each individual there was number of facial expression but most of them have all facial expression image. Modified dataset was prepared manually by taking all facial expression image with right dedicated labels stored in Emotion_labels folder.

JAFFE contains facial expression for 10 Japanese women by taking each expression 3 times. Though not all emotion labels have 30 image. Some emotion label contains less or more.KDEF contains 4900 image. Only frontal image of 70 individuals is taken in modified dataset.

FER2013comes as CSV file with emotion, pixels and Usage. The Usage with ”Training” is

taken as training dataset, usage ”PublicTest” is used as validation set and ”PrivateTest” is used as testing set.

5.5.2 Image pre-]processing

- All the image dataset is transformed into grayscale.
- CK+ and KDEF dataset is resized into $256 * 256$.

Figure 5.1: Example image resizing (from 562*762 to 256*256) and grayscaling in KDEF dataset

- Facial image is detected, cropped and aligned using 68-point facial landmark. Landmark vector of shape $(68, 2)$ is saved for each dataset.
- To remove the difference with emotion label in different dataset emotion labels is changed where 0=Neutral, 1=Anger, 2=Disgust, 3=Fear, 4=Happiness, 5=Sadness and 6=Surprise.
- CK+, JAFFE and KDEF dataset is modified in the version of CSV file containing emotion label and flatten pixel value. So there are $256 * 256$ or 65536 pixel value and an emotion label is assigned per image of expression dataset.

5.5.3 Feature Extraction

LBP Feature Extraction

- A number of configuration of neighborhood point and radius was experimented for custom small dataset. Checking the performance of each pair for classification using SVM $(24, 8)$ seems good. No of neighborhood point was set as 24 and radius was taken 8 for nri_uniform method of LBP.

- All the image dataset is passed to LBP feature extractor for taken parameters. So for each image there is a histogram of dimension 26.

- Flatten landmark vector of $(68, 2)$ is concatenated with LBP features. So, the size of features for per image becomes 162.

- So, the feature vector shape for CK+ is $(327, 162)$, JAFFE is $(213, 162)$, KDEF is $(490, 162)$ and FER2013 is $(28709, 162)$. For both validation and testing dataset of FER2013 shape of feature vector is $(3589, 162)$.

CNN Feature Extraction

- Pixel value of training and validation set is normalized between the value of 0 and 1. So, all the pixel value is divided by 256 and stored accordingly.

- One hot encoding is performed on emotion label. So, each image emotion is now represented in a vector of length 7. For example emotion label 5 becomes $[0000010]$

- input data is passed to CNN model (Figure 4.12) to fit with validation data is specified. Number of epochs is also set and callback is used for ModelCheckpoint.

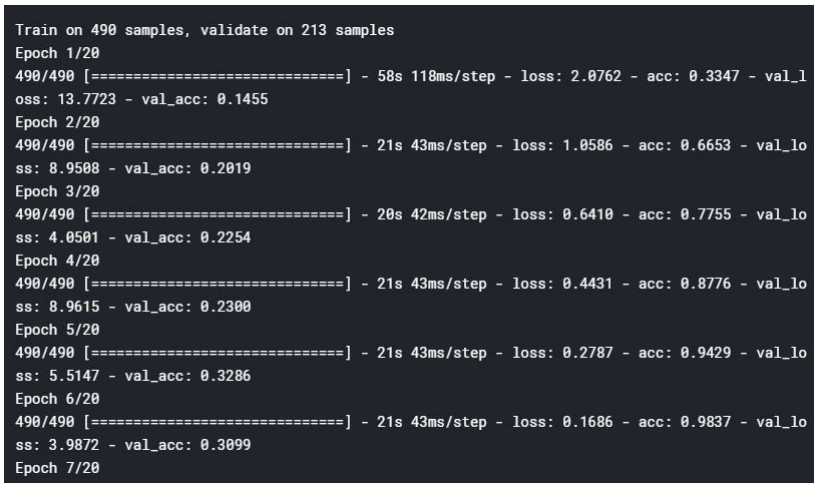


Figure 5.2: Snapshot of CNN training

- Input data of first dense model is recovered from saved model. It makes a vector of length 128. The shape depends on the number of image in training set.
- This feature vector is concatenated with flatten landmark vector which makes feature vector length 264.

5.5.4 Classification

- We implemented our feature vector on Support Vector Machine of both linear and non-linear kernel. For non-linear kernel we used 'rbf'.
- We performed cross validation on custom small dataset and found a good set of parameters for our work.
- For linear kernel, value of C is choosed 100. Other default parameter of LinearSVC function provided by sklearn library was okay.
- For 'rbf' kernel, random_state is selected as 0, epochs value as 10000 and decision function is selected 'ovr'. Other default parameter of SVC function provided by sklearn library was okay.
- All the training dataset is fitted in both version of SVM.
- Accuracy with confusion matrix and classification report is taken for further analyzing.

5.6 Source Codes

Source code gives the complete outlook of experimental work. There are number of codes is used in this experiment. All modified CK+, JAFFE, KDEF and FER2013 dataset with Landmarks and LBP feature vector is uploaded in Kaggle. As it provides GPU, so calculation is faster for large dataset like FER2013.

- Codes for resizing dataset and making them grayscale.
- Codes for face alignment, cropping.
- Codes for getting LBP features and landmark vector.
- Codes for application of LBP and CNN feature extractor on SVM while KDEF is trained.
- Codes for application of LBP and CNN feature extractor on SVM while FER2013 is trained.

All the codes are uploaded online can be visualized in Kaggle [37] and Github [38].

5.7 Conclusion

Implementation details are necessary for total insight of thesis work. It helps to understand and possibly recreate or improve the current work.

Chapter 6

Result and Performance Analysis

6.1 Introduction

LBP based SVM and CNN based SVM is implemented. Comparative result and performance analysis for these method gives us the overall idea about these methods. Here, labels are represented as 0=Neutral, 1=Anger, 2=Disgust, 3=Fear, 4=Happiness, 5=Sadness and 6=Surprise.

6.2 Result

6.2.1 LBP based SVM for FER2013 dataset

- **Training Dataset:** FER2013 Training Dataset (Table 5.4)
- **Validation Dataset:** FER2013 PublicTest Dataset
- **Testing Dataset:** FER2013 Private Dataset

	precision	recall	f1-score	support
0	0.41	0.48	0.44	607
1	0.48	0.28	0.35	467
2	0.94	0.27	0.42	56
3	0.39	0.20	0.26	496
4	0.54	0.77	0.63	895
5	0.35	0.39	0.37	653
6	0.74	0.57	0.64	415
avg / total	0.48	0.48	0.46	3589

(a) Classifiction Report

```
[[290  29   0  31 111 137   9]
 [ 75 131   0  33 135  84   9]
 [  8   7  15   3  12  10   1]
 [ 98  38   0  98 112 110  40]
 [ 76  15   1  16 689  86  12]
 [131  38   0  47 171 253  13]
 [ 31  17   0  26  56  49 236]]
```

(b) Confusion matrix

Figure 6.1: Validation Dataset for rbf kernel

	precision	recall	f1-score	support
0	0.18	0.21	0.20	607
1	0.16	0.09	0.12	467
2	0.00	0.00	0.00	56
3	0.11	0.06	0.08	496
4	0.25	0.37	0.30	895
5	0.21	0.23	0.22	653
6	0.12	0.08	0.10	415
avg / total	0.18	0.20	0.18	3589

(a) Classifiction Report

```
[[129  46   2  51 212 110  57]
 [ 74  44   2  38 185  89  35]
 [  8   6   0   2  22  12   6]
 [104  29   1  30 202  89  41]
 [181  62   2  74 330 179  67]
 [138  47   2  43 227 151  45]
 [ 79  38   2  25 150  87  34]]
```

(b) Confusion matrix

Figure 6.2: Testing Dataset for rbf kernel

Table 6.1: Performace of LBP based SVM for FER2013 dataset

Kernel	**Time**	**Dataset**	**Accuracy**
linear	148.4 sec	Vlidation Dataset	40.3%
		Testing Dataset	18.1%
rbf	463.0 sec	Vlidation Dataset	47.7%
		Testing Dataset	20.0%

6.2.2 CNN based SVM for FER2013 dataset

- **Training Dataset:** FER2013 Training Dataset (Table 5.4)
- **Validation Dataset:** FER2013 PublicTest Dataset
- **Testing Dataset:** FER2013 Private Dataset

	precision	recall	f1-score	support
0	0.62	0.34	0.44	607
1	0.57	0.42	0.49	467
2	0.76	0.46	0.58	56
3	0.48	0.33	0.39	496
4	0.80	0.78	0.79	895
5	0.38	0.73	0.50	653
6	0.78	0.73	0.75	415
avg / total	0.62	0.58	0.58	3589

(a) Classifiction Report

```
[[209  32   1  42  59 250  14]
 [ 23 197   3  47  30 152  15]
 [  1  13  26   2   3  11   0]
 [ 21  33   1 164  21 225  31]
 [ 39  17   1  20 699  99  20]
 [ 35  45   2  43  42 479   7]
 [  9   6   0  23  20  55 302]]
```

(b) Confusion matrix

Figure 6.3: Validation Dataset for rbf kernel

	precision	recall	f1-score	support
0	0.64	0.35	0.45	607
1	0.55	0.42	0.47	467
2	0.74	0.52	0.61	56
3	0.47	0.32	0.38	496
4	0.78	0.77	0.78	895
5	0.38	0.72	0.50	653
6	0.78	0.71	0.74	415
avg / total	0.61	0.57	0.57	3589

(a) Classifiction Report

```
[[214  33   1  40  67 240  12]
 [ 21 196   5  47  31 150  17]
 [  0  11  29   1   3  12   0]
 [ 21  36   0 159  24 226  30]
 [ 43  19   2  20 692  99  20]
 [ 29  55   2  44  45 472   6]
 [  8   9   0  24  25  53 296]]
```

(b) Confusion matrix

Figure 6.4: Testing Dataset for rbf kernel

Table 6.2: Performace of CNN based SVM for FER2013 dataset

Kernel	**Time**	**Dataset**	**Accuracy**
linear	47.2 sec	Vlidation Dataset	57.8%
		Testing Dataset	57.3%
rbf	914.4 sec	Vlidation Dataset	31.9%
		Testing Dataset	25.0%

6.2.3 LBP based SVM for KDEF dataset

- **Training Dataset:** KDEF Dataset (Table 5.4)
- **Validation Dataset:** JAFFE Dataset
- **Testing Dataset:** CK+ Dataset

	precision	recall	f1-score	support
0	0.04	0.03	0.04	30
1	0.40	0.27	0.32	30
2	0.29	0.28	0.28	29
3	0.11	0.22	0.14	32
4	0.00	0.00	0.00	31
5	0.12	0.16	0.14	31
6	0.16	0.10	0.12	30
avg / total	0.16	0.15	0.15	213

(a) Classifiction Report

```
[[ 1  0  3  8  2  9  7]
 [ 7  8  1 10  0  1  3]
 [ 1  5  8  8  1  5  1]
 [ 9  1  1  7  3 10  1]
 [ 2  3  5 11  0  9  1]
 [ 6  1  4 11  1  5  3]
 [ 1  2  6 10  5  3  3]]
```

(b) Confusion matrix

Figure 6.5: Validation Dataset for rbf kernel

	precision	recall	f1-score	support
0	0.15	0.14	0.14	50
1	0.17	0.20	0.18	45
2	0.25	0.15	0.19	59
3	0.07	0.12	0.09	24
4	0.26	0.17	0.20	59
5	0.08	0.21	0.12	28
6	0.12	0.06	0.08	62
avg / total	0.17	0.15	0.15	327

(a) Classifiction Report

```
[[ 7  7  5  8  7 13  3]
 [ 3  9  5  8  4 10  6]
 [12  7  9  8  4 12  7]
 [ 3  2  2  3  2  8  4]
 [ 7 10  6  6 10 11  9]
 [ 4  4  3  6  4  6  1]
 [11 14  6  7  8 12  4]]
```

(b) Confusion matrix

Figure 6.6: Testing Dataset for rbf kernel

Table 6.3: Performace of LBP based SVM for KDEF dataset

Kernel	**Time**	**Dataset**	**Accuracy**
linear	1.4 sec	Vlidation Dataset	13.1%
		Testing Dataset	11.0%
rbf	0.2 sec	Vlidation Dataset	15.0%
		Testing Dataset	14.7%

6.2.4 CNN based SVM for KDEF dataset

- **Training Dataset:** KDEF Dataset (Table 5.4)
- **Validation Dataset:** JAFFE Dataset

	precision	recall	f1-score	support
0	0.16	0.23	0.19	30
1	1.00	0.03	0.06	30
2	1.00	0.17	0.29	29
3	0.42	0.25	0.31	32
4	0.93	0.42	0.58	31
5	0.22	0.68	0.34	31
6	0.65	0.80	0.72	30
avg / total	0.62	0.37	0.36	213

(a) Classifiction Report

```
[[ 7  0  0  0  0 16  7]
 [17  1  0  4  0  7  1]
 [ 6  0  5  3  0 15  0]
 [ 4  0  0  8  0 19  1]
 [ 3  0  0  1 13 12  2]
 [ 6  0  0  2  0 21  2]
 [ 0  0  0  1  1  4 24]]
```

(b) Confusion matrix

Figure 6.7: Validation Dataset for linear kernel

	precision	recall	f1-score	support
0	0.53	0.48	0.51	50
1	0.64	0.47	0.54	45
2	0.90	0.61	0.73	59
3	0.27	0.62	0.38	24
4	0.92	0.93	0.92	59
5	0.42	0.79	0.55	28
6	0.90	0.60	0.72	62
avg / total	0.72	0.64	0.66	327

(a) Classifiction Report

```
[[24  2  0 12  2  7  3]
 [ 5 21  4  7  1  7  0]
 [10  7 36  3  1  2  0]
 [ 2  1  0 15  1  4  1]
 [ 1  0  0  2 55  1  0]
 [ 1  2  0  3  0 22  0]
 [ 2  0  0 14  0  9 37]]
```

(b) Confusion matrix

Figure 6.8: Testing Dataset for linear kernel

- **Testing Dataset:** CK+ Dataset

Table 6.4: Performace of CNN based SVM for KDEF dataset

Kernel	**Time**	**Dataset**	**Accuracy**
linear	0.1 sec	Vlidation Dataset	37.1%
		Testing Dataset	64.2%
rbf	0.3 sec	Vlidation Dataset	14.6%
		Testing Dataset	8.6%

6.3 Performance Analysis

We have done a series of experiment on various dataset. It is seen that there are ups and downs of the performance depending on dataset and applied method. Classification report and confusion matrix gives us more describing outlook which will lead us to more accurate performance analysis.

Our multi-class datasets are scatter. it is little difficult to accurately classify all the facial expression label. It can be seen that below 2D plot of some datasets.

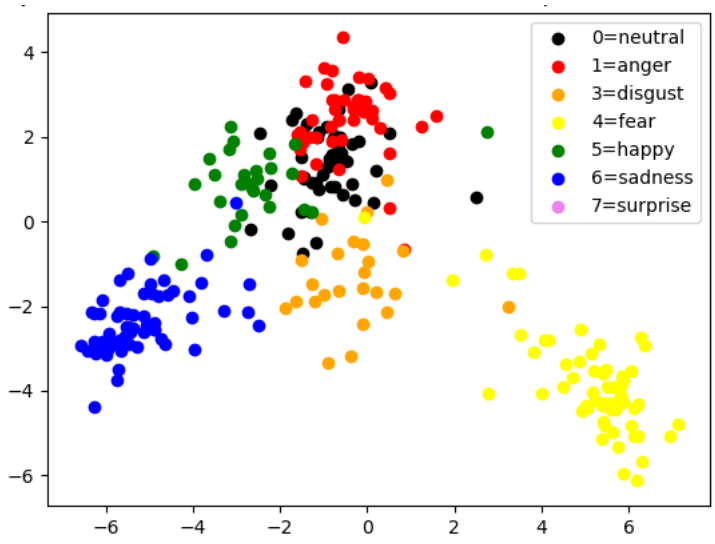


Figure 6.9: Normalized 2D plot of CK+ multi-class dataset

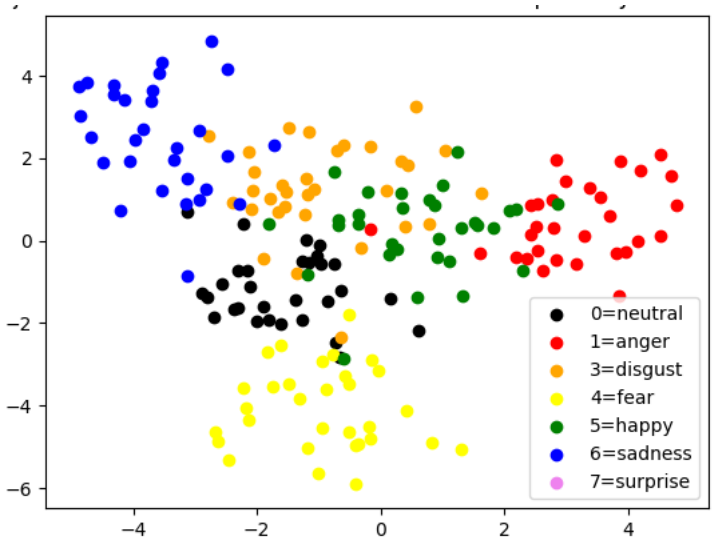


Figure 6.10: Normalized 2D plot of JAFFE multi-class dataset

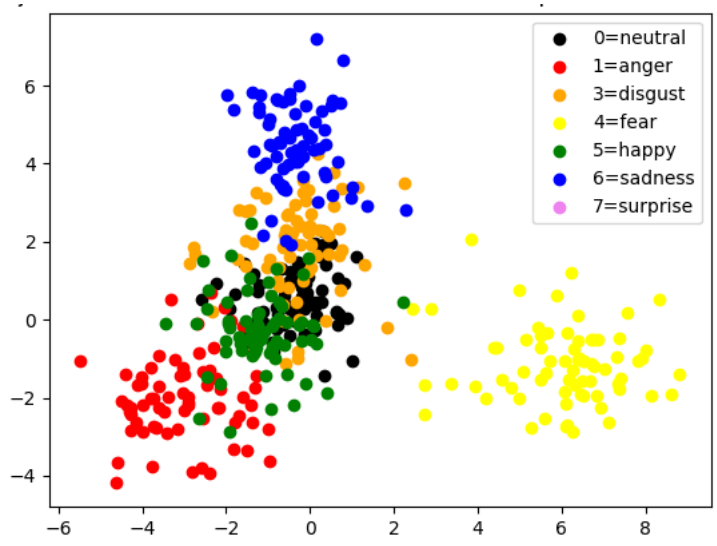


Figure 6.11: Normalized 2D plot of KDEF multi-class dataset

As per Table 6.1, it is seen that for LBP based SVM for FER2013 dataset, the highest performance found in our method 47.7%. For this method testing accuracy was only 20%. FER2013 dataset is a dataset of low resolution image. Though LBP is mostly illumination invariant here another case comes to highlight with lot of data there are lot of noise. And sometimes the performance drop of SVM happens because of high number of training data.

As per Table 6.2, CNN based SVM for FER2013 dataset gives a better result for both validation

and testing dataset as CNN works better when training dataset is large. Using CNN as feature extractor helps us to get much feature information which helped us to achieve better result when feature vector is fitted in SVM model.

As per Table 6.3, LBP based KDEF gives a much lower result. There are performance drop in both validation and testing dataset. It is to remind that both both validation and testing dataset are collected on different environment than KDEF. So, it is seen that how training with one dataset and testing in different lowers the performance. in most of the experiment [6] both training and testing is done on same dataset. Which may gives better result but our work highlight overall real world performance of a method.

As per Table 6.4, CNN based KDEF gives better result than previous for linear kernel. So, for different type of dataset as training and test set linear kernel is more appropriate as rbf kernel with CNN based SVM gives no improvement. it has comparatively higher performance in testing AKA CK+ dataset here. From confusion matrix, it is seen that it has less wrong labeling.

It is seen that rbf based SVM takes a much higher time compared to linear kernel for FER2013 dataset. For smaller dataset this time complexity does not gets noticed. It is to remind that to achieve CNN feature vector CNN model takes longer time than achieving LBP feature vector.

It is also seen that LBP based SVM works better when 'rbf' kernel is used and CNN based SVM works better when 'linear' kernel is used. CNN based SVM gives comparatively better result than LBP based SVM for facial expression recognition. Performance drop happens when testing is done on different dataset.

6.4 Conclusion

This chapter is total overview of the result found by our experiment. it gives us a comparative highlight about the performance of different method for different dataset.

Chapter 7

Conclusion

7.1 Summary

The thesis work highlighted different method for Facial Expression Recognition. Detecting 7 labeled facial expression is a multi-class problem. There was focus on implementing Support Vector Machine (SVM) for this task. Though, different feature extraction method is used and they are Local Binary Pattern (LBP) and Convolutional Neural Network (CNN). And number of different dataset is used to get true evaluation of each method. It gives us an general idea about performance of SVM and possiblities of LBP and CNN as feature extractor for facial expression recognition. Applying same structure of experimnt for every method and evaluating performance gives us comparative overview.

7.2 Drawbacks

Thesis work have exposed some drawbacks at the end of the thesis.

- The thesis worked failed to achiecve better result than any current method.
- It also doesn't concern about facial expression found by non-frontal face and video stream.

7.3 Future Scopes

This thesis work has many scope for improvement. According to drawbacks, many research scope are evolved that are provided below.

- Finding facial expression of non-frontal face.
- Finding expression through video stream
- Finding more accurate features for facial expression recognition.
- Evaluating performance for different illumination dataset of training and testing.

References

[1] J. S. Carton, E. A. Kessler, and C. L. Pape, "Nonverbal decoding skills and relationship well-being in adults," *Nonverbal Behavior 23(1)*, p. 91–100, Spring 1999.

[2] P. Ekman and W. V. Friesen, "Pictures of facial affect," *Consulting Psychologists Press*, 1976.

[3] S. Du, Y. Tao, , and A. M. Martinez, "Compound facial expressions of emotion," *PNAS*, 2014.

[4] enkiverywell, "Common facial expressions and their meaning," https://www.enkiverywell.com/facial-expressions-list.html.

[5] marketsandmarkets, "Emotion detection and recognition market," https://www.marketsandmarkets.com/PressReleases/emotion-detection-recognition.asp.

[6] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on local binary patterns: A comprehensive study," *Elsevier*, 2008.

[7] D. Duncan, G. Shine, and C. English, "Facial emotion recognition in real time," *Stanford*, 2016.

[8] V. P. J. Henry Medeiros, A. Zanellato, and C. Bortolini, "Facial expression classification using convolutional neural network and support vector machine," *Semanticscholar*, 2016.

[9] Y. Tang, "Deep learning using linear support vector machines," *arXiv:1306.0239*, 2015.

[10] F. Juefei-Xu, V. N. Boddeti, and M. Savvides, "Local binary convolutional neural networks," *IEEE*, 2017.

[11] W.-L. Chao, "Face recognition," *semanticscholar*, 2007.

[12] Y.-L. Tian, T. Kanade, and J. F. Cohn, "Facial expression analysis," *Springer*, p. Chapter 11, 2005.

[13] D. Ghimire and J. Lee, "Geometric feature-based facial expression recognition in image sequences using multi-class adaboost and support vector machines," *PMC*, 2013.

[14] N. C. Joy and D. P. J.C., "Feature extraction techniques for facial expression recognition systems," *Global Research and Development Journal for Engineering*, 2016.

[15] A. S. Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "Cnn features off-the-shelf: an astounding baseline for recognition," *arXiv:1403.6382*, 2014.

[16] T. Bluche, H. Ney, and C. Kermorvant, "Feature extraction with convolutional neural networks for handwritten word recognition," *IEEE*, 2013.

[17] A. F. M. Agarap, "An architecture combining convolutional neural network (cnn) and support vector machine (svm) for image classification," *arXiv:1712.0354*, 2017.

[18] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. D. Decaf, "A deep convolutional activation feature for generic visual recognition," *ICML*, 2014.

[19] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," *CoRR, abs/1311.2901*, 2013.

[20] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," *INRIA*, 2013.

[21] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," *IEEE*, 2010.

[22] kasrl.org, "The japanese female facial expression (jaffe) database," http://www.kasrl.org/jaffe.html.

[23] D. Lundqvist, A. Flykt, and A. Öhman, "The karolinska directed emotional faces – kdef," *Psychology section, Karolinska Institutet*, 1998.

[24] S. o. P. S. S. Karolinska Institutet, Department of Clinical Neuroscience, "Kdef & akdef," http://www.kdef.se/.

[25] P.-L. Carrier and A. Courville, "Challenges in representation learning: Facial expression recognition challenge," https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge/data.

[26] dlib, "Dlib c++ librarys," http://dlib.net/.

[27] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *IEEE*, 2001.

[28] iBUG 300-W dataset, "The 68 facial landmark coordinates," https://ibug.doc.ic.ac.uk/resources/facial-point-annotations/.

[29] T. Ojala, M. Pietikäinen, and D. Harwoods, "Performance evaluation of texture measures with classification based on kullback discrimination of distributions," *International Conference on Pattern Recognition, vol. 1*, pp. 582 – 585, 1994.

[30] ——, "A comparative study of texture measures with classification based on feature distributions," *Pattern Recognition, vol. 29*, pp. 582 – 585, 1996.

[31] pyimagesearch.com, "Lbp with 8-bit binary neighborhood of the center pixel visualization," https://www.pyimagesearch.com/2015/12/07/local-binary-patterns-with-python-opencv/.

[32] www.mathworks.com, "Introduction to deep learning: What are convolutional neural networks," https://www.mathworks.com/videos/introduction-to-deep-learning-what-are-convolutional-neural-networks--1489512765771.html.

[33] ——, "Introduction to deep learning: What are convolutional neural networks," https://medium.com/@RaghavPrabhu/understanding-of-convolutional-neural-network-cnn-deep-learning-99760835f148.

[34] cs231n.github.io, "Convolutional neural networks (cnns / convnets)," http://cs231n.github.io/convolutional-networks/.

[35] analyticsvidhya.com, "Understanding support vector machine algorithm from examples," https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/.

[36] K. P. Bennett and E. J. Bredensteiner, “Duality and geometry in svm classiers,” *ICML*, 2000.

[37] A. Mursalin, “Fer-landmarks-cnn-lbp-svm,” https://www.kaggle.com/ankur133047/kernels.

[38] Encryptioner, “Thesis-fer-based-on-lbp-cnn-on-svm,” https://github.com/Encryptioner/Thesis-FER-based-on-LBP-CNN-on-SVM.