Pose and Facial Expression Recognition on Multiple Yaw Angles

A MINI PROJECT

For the degree of

B.TECH

In

INFORMATION TECHNOLOGY

Submitted by:

Richa Vinian (IIT2015015)
Payal Prasad (IIT2015052)
Puja Kumari (LIT2015017)
Samriddhi Niranjan (LIT2015021)

Under the Guidance of:

Prof. Anupam Agrawal

IIIT-Allahabad

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY, ALLAHABAD – 211012 (INDIA)

November 19, 2017

CANDIDATES' DECLARATION

We hereby declare that the work presented in this project report of B.Tech (IT) 5th Semester entitled "**Pose and Expression Recognition on Multiple Yaw Angles"**, submitted by us at Indian Institute of Information Technology, Allahabad is an authenticated record of our original work carried out from July 2017 to November 2017 under the guidance of Prof. Anupam Agrawal.

Due acknowledgements have been made in the text to all other materials used. The project was done in full compliance with the requirements and constraints of the prescribed curriculum.

Place: Allahabad
Date:

Richa Vinian (IIT2015015)
Payal Prasad (IIT2015052)
Puja Kumari (LIT2015017)
Samriddhi Niranjan (LIT2015021)

CERTIFICATE FROM SUPERVISOR

I do hereby recommend that the mini project report prepared under my supervision by Richa Vinian (IIT2015015), Payal Prasad (IIT2015052), Puja Kumari (LIT2015017) and Samriddhi Niranjan (LIT2015021) titled "Pose and Expression Recognition on Multiple Yaw Angles" be accepted in the partial fulfilment of the requirements of the completion of 5th semester of Bachelor of Technology in Information Technology for Examination.

Date:
Place: Allahabad

Prof. Anupam Agrawal
IIIT-Allahabad

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to the Department of Information Technology, IIIT Allahabad for providing valuable help, discussion, suggestions and resources. We are especially thankful to Prof. Anupam Agrawal for his valuable support in providing us with the initial implementation of the topic so that we could further extend the work and make something valuable out of it.

Richa Vinian
Payal Prasad
Puja Kumari
Samriddhi Niranjan

Abstract

Research in the field of expression recognition has mainly been applied to the frontal view of face images only. Non-frontal facial expression recognition is important in many scenarios where the frontal view of the face may not be available. Some attempts have been made to work on expression recognition on non-frontal faces but most of these attempts have considered yaw angles only up to 45°. This is largely due to the limitations of the databases available. In this project, we describe an approach to recognize the pose and the expressions of the non-frontal faces. For classification, we used Multi-class Support Vector Machines (SVM) because of its ability to classify high dimensional data quickly. For feature extraction, we have done a comparative analysis of three approaches of expression classification for non-frontal images- LGBP without angle classification, LBP with angle classification and DCLBP. Among the three methods, best results were obtained on LBP with angle classification followed by LBP without angle classification and DCLBP. The methods were trained and tested on the benchmark dataset KDEF. Testing was also done on our own dataset.

Contents

1. Introduction

Automatic facial expression recognition from non-frontal views is a challenging research topic which has recently started to attract the attention of the research community [5]. According to a study, it has been observed that when two human beings interact with each other verbal cues provide only 7% of the meaning of the message and vocal cues about 38%. However, facial expressions provide 55% of the meaning of the message [4] i.e., a significant amount of information, during human interaction, is conveyed through facial expressions.

Psychological research has classified facial expression into 6 basic categories. These are: *joy*, *surprise*, *anger*, *fear*, *sadness* and *disgust*. In this project, we did a comparative analysis of facial expression recognition with and without pose prediction. We try to classify facial images, which are not necessarily frontal view of the faces into one of the above mentioned six categories.

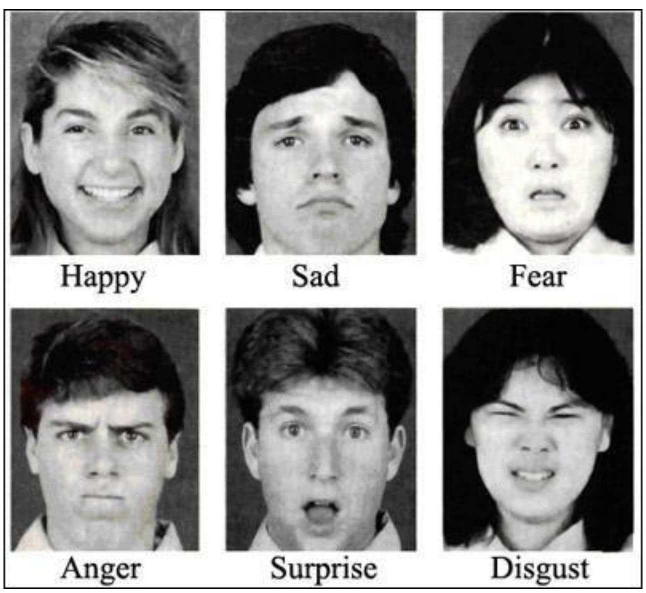


Fig.1 The six basic facial expressions.

The first and the most important step in facial expression recognition is to extract a good representation of the face from the image. Most of the datasets which have been used to do facial expression recognition have images which have been taken from in-front of the face.The second and last step for facial expression recognition is classification. For this purpose, we used multi-class Support Vector Machines (SVMs).

2. Motivation

Most of the work on facial expression recognition has been done using frontal facial images only. As compared to frontal view, only a small amount of work has been carried out to observe the effect of different poses on facial expression recognition. Psychology experiments have shown that even if a person's head pose changes by 15-degree, it results in statistically significant changes in how the other person perceives the emotion. It is natural to assume that frontal pose is optimal for facial expression recognition, as at this view the whole face is visible. However, little work has been carried out to investigate optimal view for facial expression recognition [4].

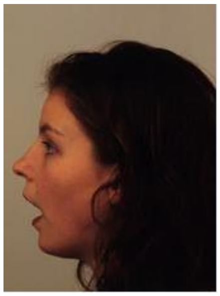
1. Problem Objective and Scope

The objective of our project was to develop a software system which takes an image of human face as an input and give the expression on the face as output. The input image may or may not be a frontal view of the face. It may be an image captured at an angle between that of profile view and front view. The output given by the system must be one of the following: *happy*, *surprise*, *anger*, *fear*, *sad* and *disgust.*

<u>Example Input 1</u>:

<u>Example Output 1</u>: Happy

<u>Example Input 2</u>:



<u>Example Output 2</u>: Surprise

4. Literature Survey

S. No.	Title of Paper	Conf./ Journal Name & Publisher	Year	Method	Merits/Challenges Dealt	Test Data	Demerits	Future scope
*** 1 ***	**Local binary patterns for multi-view facial expression recognition**	**ACM**	**2011**	**Different types of Local Binary Patterns (LBP), Support Vector Machines (SVM)**	**Illumination Variation, Low Resolution, Rotation Invariance, Computation Time, no manual Labelling on Images Required, Noise Tolerant**	**BU-3DFE, Multi-Pie**	**Slightly inaccurate with faces having facial hair and spectacles**	**Accuracy may be Improved**
2	A Novel Approach to Expression Recognition from Non-Frontal Images	IEEE	2009	Scale Invariant Feature Transform and ROU Technique, SVM	Rotation and Scaling Invariance.	BU-3DFE	Uses Preprocessed Input from database for computing SIFT features.	To use the real-time facial point detecting and tracking techniques
3	Local saliency-inspired binary patterns for automatic recognition of multi-view facial expression	IEEE	2016	Local Saliency-inspired Binary Pattern	The accurate recognition of MVFE from 2D face images is a challenge using a computationally inexpensive model.	Multi-Pie Database	An SVM classifier is adopted since it is a well understood classification technique that has been demonstrated to be effective for facial expression recognition.	Improve Accuracy
4	Gabor Filter Based Face Recognition Using Non-frontal Face Images	ACM	2015	Gabor Wavelet and Feature Extraction, SMO classifier, K-means clustering	Low Resolution, Computation time, Illumination Conditions, Manual Labelling of points not required	Multi-Pie Database	Flexible in terms of head-pose movement	-

<u>Note:</u> *** X * denotes base paper used for Implementation.**

5. Methodology

In this project, we have done a comparative analysis of three approaches of expression classification for non-frontal images. The second approach is the same as proposed by Moore and Bowden [4] but slightly optimized than the original. The first approach is a slight modification of the second approach. It does not involve angle classification like the second approach. The third approach involves DCLBP, which is a texture descriptor. This descriptor has never been used before to extract features from images to perform expression classification.

In order to images on the basis of expressions, the basic approach is to do it in a two-step manner. The two basic steps are Feature Extraction and Classification as shown in figure 5.1. For classification, we used Support Vector Machines (SVM) because of its ability to classify high dimensional data quickly. For Feature Extraction, we explored and compared 3 different approaches namely:

1. Local Gabor Binary Patterns (LGBP) without Angle Classification.
2. Local Gabor Binary Pattern (LGBP) with Angle Classification.
3. Diagonal Cris-Cross Local Binary Pattern (DCLBP).

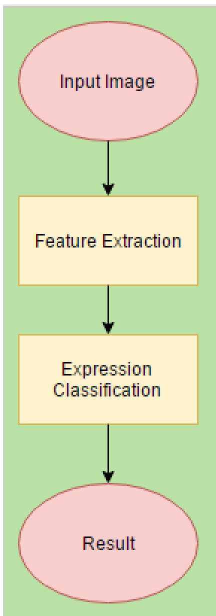


Fig 5.1 Flowchart of the Basic Approach.

5.1 Feature Extraction

In general, two categories of feature extraction methods exist which are used for facial expression recognition: geometric based methods and appearance based methods [4]. The features extracted using geometric based methods contains information about the location and shape of facial features whereas the features extracted using Appearance based methods contains information about the appearance change of the face which includes furrows, wrinkles, and bulges. Appearance based methods use different kinds of filters which are applied on the image to extract information from it. The features extracted using geometric based methods are sensitive to noise whereas the features extracted using Appearance based methods are less dependent on initialization and are capable of encoding micro patterns present in the skin texture which are important for facial expression recognition. All the methods described in this report are appearance based methods.

5.1.1 Local Binary Pattern (LBP)

LBP is an operator which is applied to an image pixel by pixel to extract features from it.LBPs have been successfully applied as a texture descriptor . Some of the properties of LBP which makes it an extremely important feature extraction technique are as follows. It is not computationally complex like many other techniques such as neural networks etc. It is tolerant to illumination change. The steps to compute LBP of an image are as follows. A 3 x 3 window of pixels is considered around a center pixel f_c. The neighboring pixels of the center pixel are labelled as f_p where $p \in \{0, . . ., 7\}$. These neighboring pixels are thresholded against the center pixel to obtain a binary string. An example of the calculation of LBP operator is shown in Fig. 5.2. The thresholding function $S(f_p - f_c)$ is given by equation (1).

$$S(f_p - f_c) = \begin{cases} 1 & \text{if } f_p \geqslant f_c \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

By multiplying each bit of binary string by 2^p, the LBP is computed as follows:

$$LBP = \sum_{p=0}^{7} S(f_p - f_c) 2^p \tag{2}$$

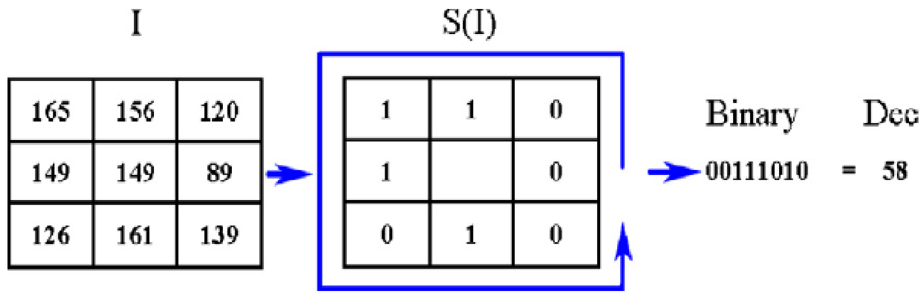


Fig 5.2 Shows calculation of LBP for a pixel with a neighborhood of 3x3 .

The histogram of the image, or any sub-region of it, obtained after application of LBP operator is used as a feature vector. This feature vector contains the necessary texture information about the image.

5.1.2 Uniform Local Binary Pattern

Uniform Local Binary Pattern is a particular case of Local Binary Pattern. A local binary pattern which has at most two bitwise transitions from 0 to 1 or 1 to 0 for a circular binary string is called as uniform local binary pattern. 00000110 and 1011111 are examples of uniform local binary pattern whereas 01010110 is an example of non-uniform local binary pattern. The notation LBP(P, R) is used to denote a neighborhood of P pixels around a center pixel and R denotes the radius. If R = 1 then P = 8; R = 2, P = 16 and so on. The LBP shown in Figure 5.2. is a non-uniform LBP with P = 8 and R = 1. It has been observed that for LBP(8, 1), out of 2^8 possible patterns in an image, more than 90% of all patterns are uniform local binary patterns [4]. Total number of uniform LBPs for P pixels surrounding the center pixel is given by (P - 1)P + 2. All the other patterns where U(x) > 2, are defined as non-uniform patterns. In mathematical notation, the uniform LBP could be written as follows

$$LBP^{u2}_{P,R} = \begin{cases} z \quad \text{if } U(LBP_{P,R}) \leqslant 2,\ LBP_{P,R} = I_z,\ I_z \in I, \\ \qquad \text{where } |I| = (P-1)P+1 \\ (P-1)P+2 \text{ otherwise} \end{cases} \tag{3}$$

where,

$$U(LBP_{P,R}) = ||S(f_{p-1} - f_c) - S(f_0 - f_c)|| + \sum_{p=0}^{P} ||S(f_p - f_c) - S(f_{P-1} - f_c)|| \tag{4}$$

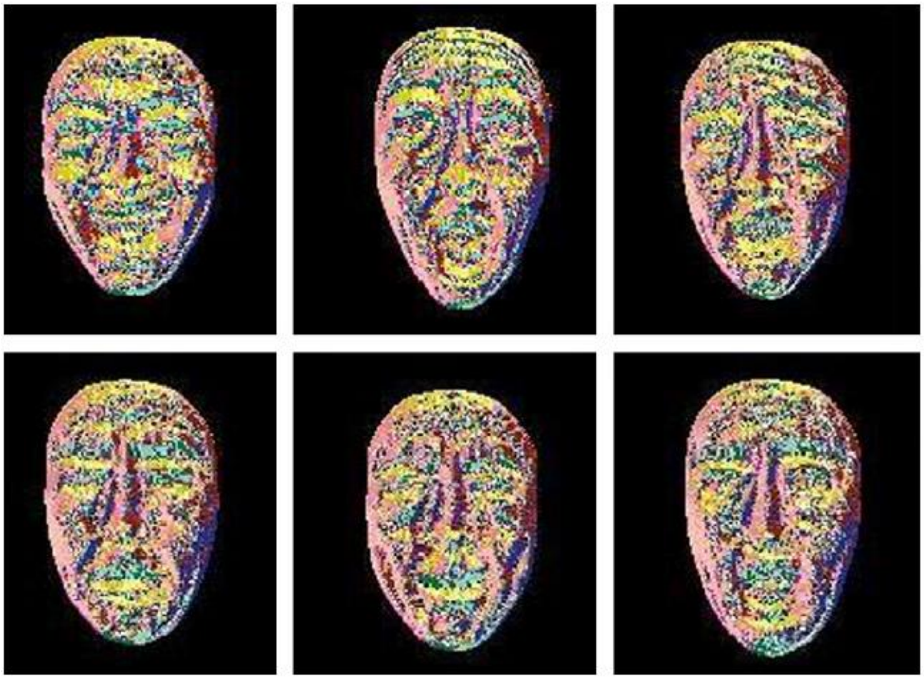


Fig 5.3: Uniform LBP of six basic expressions. From top to bottom and left to right: happy, surprise, fear, sad, disgust, anger.

Using uniform patterns for a neighborhood where P = 8, reduces the histogram from 256 to 59 bins (58 bins for uniform patterns and 1 bin for non-uniform patterns). The figure 5.3 shows colored uniform LBP images of all the six basic expressions.

5.1.3 Local Gabor Binary Pattern (LGBP)

Before applying local binary pattern (LBP) to an image, if Gabor filter is applied to the image, then the obtained LBP map is known as Local Gabor Binary Pattern (LGBP). Gabor filter is very useful in detecting edges from the image. The cells present in the visual cortex of the mammals could be modelled using gabor filters. This is why it is an extremely good technique for feature extraction, texture representation and texture discrimination. Application of gabor filter along with local binary pattern enhances the power of the histogram obtained from the image. The histogram obtained from the LGBP map contains more information about the features of the image than the LBP map alone.

In order to extract the LGBP feature vector of the image the following procedure is followed. The image is convolved with gabor filter to obtain the LGBP image as shown in equation (5).

$$G(\mu, v) = I(x, y) * \psi_{\mu,v}(z) \tag{5}$$

where:

$$\psi_{\mu,v}(z) = \frac{\left\|k_{\mu,v}\right\|^2}{\sigma^2} e^{\frac{-\left\|k_{\mu,v}\right\|^2 \|z\|^2}{2\sigma^2}} \left[e^{ik_{\mu,v}z} - e^{\frac{-\sigma^2}{2}}\right] \tag{6}$$

$$k_{\mu,v} = k_v e^{i\phi_\mu}, \quad k_v = 2^{-\frac{v+2}{2}}\pi, \quad \phi_{\mu-\mu\frac{\pi}{8}} \tag{7}$$

where z = (x, y), μ is the orientation and || . || is the norm operator. By using eight different orientations ϑ ∈ {0, . . ., 7}, 8 gabor filters are obtained. Each of these eight filters is applied to the original image. This gives us a set of 8 gabor kernels. Uniform local binary pattern is calculated for each kernel to obtain 8 LGBP maps. Feature vector is obtained from these maps by dividing each LGBP map into 9 x 9 = 81 blocks. A 59-bin histogram is computed for each block. All 81 histograms obtained from all the blocks of the maps are concatenated to form the feature vector as shown in figure 5.4.

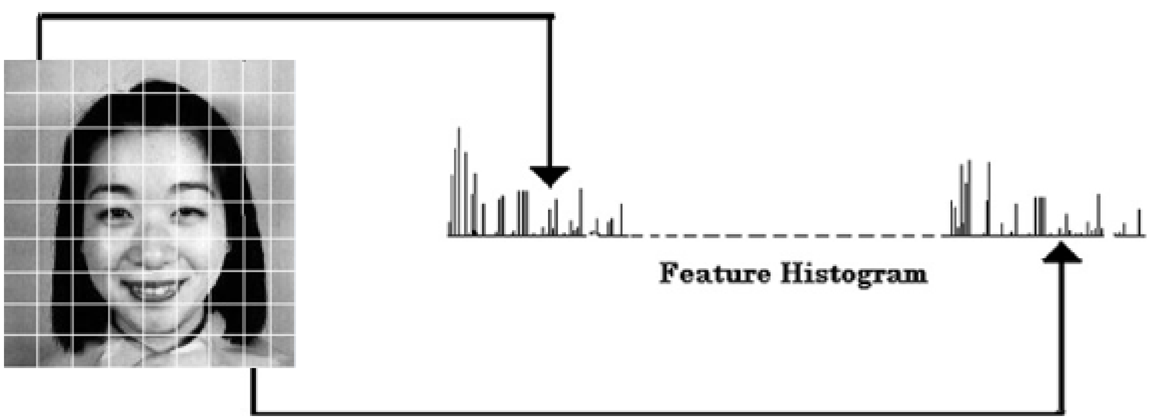


Fig. 5.4 Shows dividing the image into 81 blocks with histogram creation and concatenation.

The feature vector obtained contains information about the local image texture. This feature vector contains more information about the texture than the feature

vector of LBP. This procedure can me mathematically described using equations (8), (9), (10) and (11).
Equation (8) shows the complete feature vector HG(LBP^{xxx}) as a concatenation of different histograms (H_0, H_1, …) obtained from the LGBP maps.

$$HG(LBP^{xxx}) = (H_0, H_1, \ldots, H_{n-1}). \tag{8}$$

where the histogram of the r^{th} sub-block of LBP^{xxx} is computed by:

$$H_r = (h_{r,0}, h_{r,1}, \ldots, h_{r,u-1}) \tag{9}$$

where u is the total number of bins for feature LBP^{xxx} and h is defined as:

$$h_i = \sum_{x,y} I\{LBP^{xxx}(x, y = i)\}, \quad i = 0, 1, \ldots, u-1 \tag{10}$$

where i is the i^{th} bin of histogram h, h_i is the number of patterns in the image with LBP^{xxx} pattern i and

$$I(A) = \begin{cases} 1 & \text{if } A \text{ is true} \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

5.1.4 Diagonal Cris-Cross Local Binary Pattern (DCLBP)

The diagonal cris-cross LBP operator is a variant of the original LBP operator. This operator takes into account the value of the center pixel while thresholding values. Also, diagonal variations in the intensity of the pixel values is considered. The DCLBP operator is given by the mathematical formulation as follows:

$$DCLBP_{p,r}(N_c) = \frac{\left[\left(\sum_{k=0}^{|P|-1} \vartheta\left(\delta_{k,|P|+k}\right) \times 2^{p_k \in P}\right) + N_c\right]}{2} \qquad P = \{1, 3, 5, 7\} \qquad |P| = 4 \tag{12}$$

$$\delta_{k,|P|+k} = \left(N_k - N_{|P|+k}\right) \tag{13}$$

$$\vartheta(\delta) = \begin{cases} 0, & \delta < 0 \\ 1, & \delta \geq 0 \end{cases} \tag{14}$$

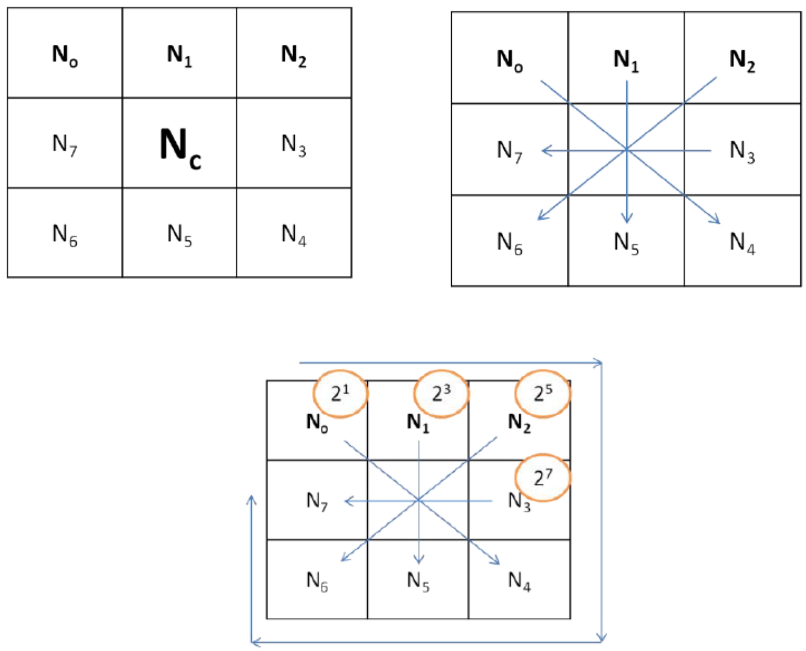


Fig. 5.5: Shows the notations associated and calculation of DCLBP. [1]

The steps to calculate DCLBP as shown in figure 5.5 are as follows:

1. Calculate the differences $N_0 - N_4$, $N_1 - N_5$, $N_2 - N_6$, $N_3 - N_7$.
2. For each positive difference assign the respective weights and add the results shown in the equation.
3. Finally, replace the center pixel value with the average of the computed value and the original value of the center pixel.

5.2 Method Flowcharts

<u>LGBP without pose classification:</u>

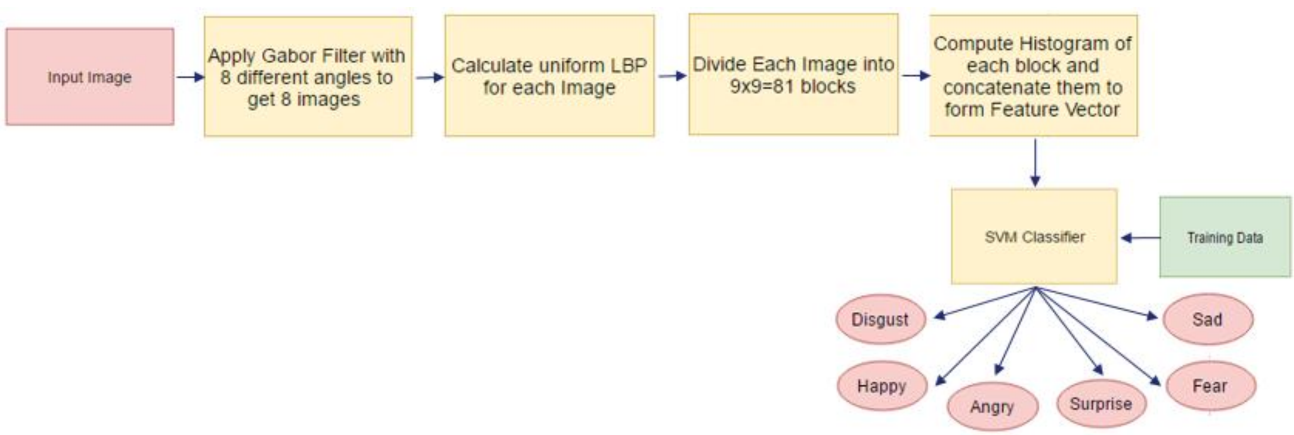


Fig. 5.6: Flowchart showing the procedure of Expression classification without angle

In this method, Gabor filters were applied to the input image. Gabor filters were applied by varying the orientation parameter of the Gabor Filter 8 times. This gave us a set of 8 images. Now, for each of these images, a corresponding LBP image was created using the LBP operator. Each of these obtained LBP images was divided into 9x9 = 81 blocks. The number of dimensions of this feature vector were 38,232 (= 59x81x8). Finally, this feature vector was passed on to the SVM classifier. The SVM classifier was previously trained by using similar feature vectors using labelled data.

LGBP with pose classification:

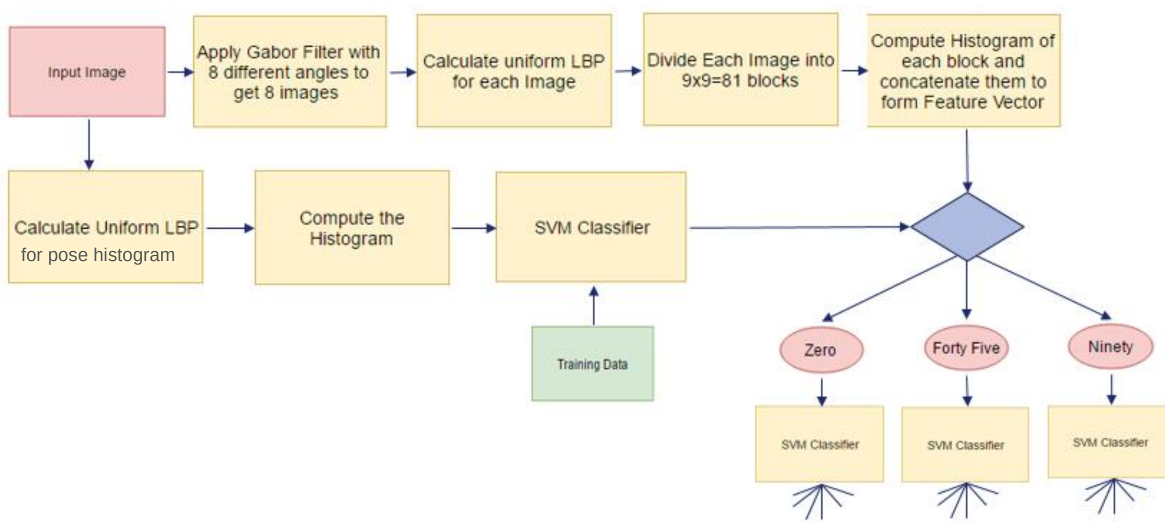


Fig. 5.7: Flowchart showing the procedure of Expression classification with angle classification.

This method is quite similar to the previous one. It is depicted by the flowchart in figure 5.7. The only difference here is the additional layer of angle classifier, which is added to classify the angle from which the image has been taken and also, the increase in number of SVM classifiers. This layer helps in improving the overall accuracy of the system. Since the KDEF dataset has images taken from 0, 45 and 90 degrees three classifiers were used in place of one, as was the case with the previous methodology. To classify the angle of the image, uniform LBP of the image was computed followed by the computation of its histogram. This histogram was passed on to the SVM classifier present in the lower branch of the flowchart to determine the category to which the image belongs. Depending upon the result of this classifier, the feature vector calculated in the upper branch of the flowchart was passed on to exactly one SVM classifier out of three.

Diagonal Cris-Cross LBP:

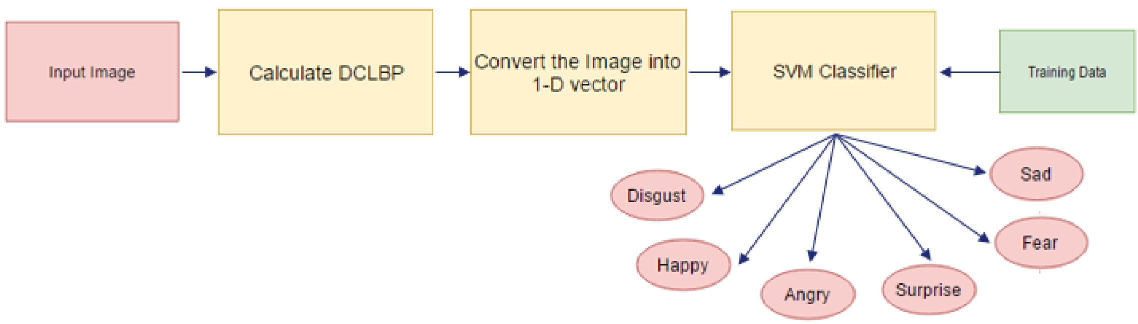


Fig. 5.8: Flowchart showing the procedure of Expression classification using DCLBP.

The flowchart in figure 5.8 depicts this method. In this method, DCLBP of the input image was computed. After this, the complete image was converted into a one-dimensional vector called as feature vector. This feature vector was then passed on to the SVM.

6. Software Requirements

Implementation Language: Python 3.5

Libraries: NumPy, OpenCV3, Scikit-learn.

IDE: Anaconda (conda), Spyder.

Hardware Requirements: 64-bit environment, Intel Core i7 CPU @ 2.60 GHz with 8 GB of RAM.

7. Implementation Details

7.1Modules

Training Phase Modules:

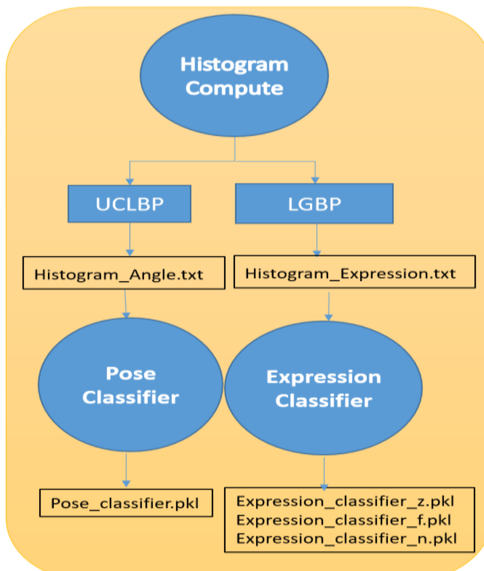


Fig 7.1: Software modules in the Training Phase

- Histogram compute module extracts features of face images from the training data and computes the histogram and stores it in text files as shown in the above image.
- Pose Classifiers uses the angle histograms created in the above step and builds the pose classifier.
- Expression Classifier uses the expression histograms computed in the first step and builds the expression classifier.
- Four classifier files are generated as the final output.

Testing Phase Modules:

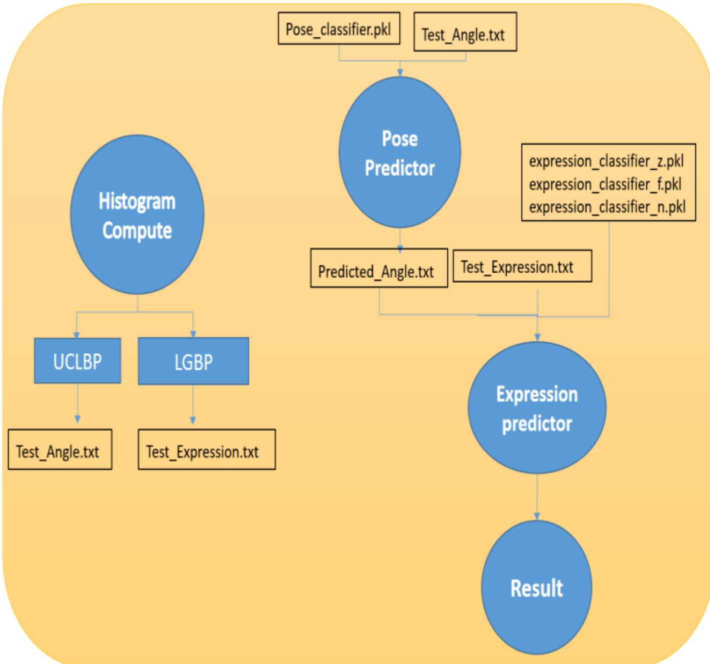


Fig. 7.2: Software modules for Testing Phase

- Histogram compute module extracts the features of face images from the testing data and computes angle and expression histograms and stores them in text files.
- Pose Predictor module uses the pose classifier generated in training phase and the angle histograms computed in the first step and predicts the pose of the test images. The predicted angles are stored in a text file.
- Expression predictor module uses the pose dependent expression classifiers generated in training phase and the expression histograms computed in the first step and predicts the expression of the test images.
- The predicted pose and expressions are displayed on the console.

7.2 Datasets

7.2.1 Benchmark Dataset- KDEF

KDEF is a set of 2,520 images of human facial expressions of emotion for training purpose. The set contains 70 individuals each displaying 6 different emotions photographed from three different angles.

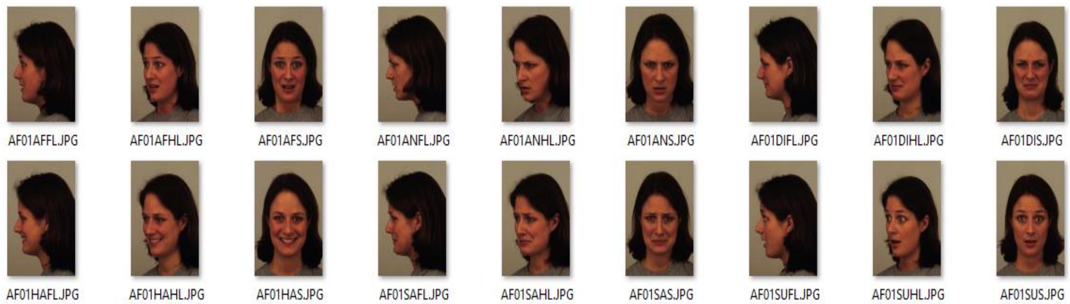


Fig. 7.3: Sample images from KDEF Dataset

7.2.2 Self-created Dataset

We created a set of 162 images for testing purpose. The set contains 9 individuals displaying 6 different emotions photographed from three different angles.

Fig. 7.4: Sample images from our Dataset

8. Results

A 10-fold cross validation was performed to determine the accuracy of all the three approaches. The KDEF dataset was divided into 10 equal sets with each set containing images of 14 persons with all their expressions from all the three angles (14 x 6 x 3 = 252 images in each set). The SVMs were trained on 9 sets and tested on one set. This procedure was applied 10 times with a different test set every time. Figure 8.1 shows the overall accuracy of the three methods described in the previous section.

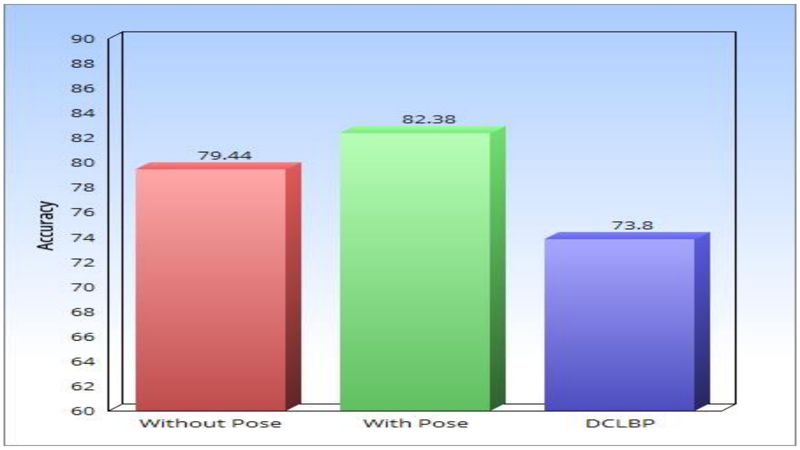


Fig. 8.1: Bar chart showing the accuracies of both the methods.

- Accuracy obtained using LGBP without angle classification was 79.44% and LGBP with angle classification was 82.38% and 73.80%.
- As we can see in figure 8.1, highest accuracy was obtained by using LGBP with angle classification followed by LGBP without angle classification.
- LGBP with angle classification performed better than LGBP without angle classification due to the added layer of classification of angle.
- The approach of using DCLBP as a feature extraction method performed the poorest among the three and therefore we conclude that DCLBP is not a better descriptor for facial expression recognition than LGBP.

The confusion matrices obtained from the three approaches are shown in figure 8.2, 8.3, 8.4 and 8.5:

	Zero	Forty Five	Ninety
Zero	**94.76**	3.33	0.95
Forty Five	4.76	**93.09**	5.95
Ninety	0.48	3.57	**92.86**

Fig. 8.2: Confusion matrix of angle classification for LGBP with angle classification.

	Happy	Sad	Surprise	Angry	Disgust	Fear
Happy	**92.86**	0.95	1.43	0.00	2.38	2.38
Sad	0.48	**83.81**	0.95	5.71	7.14	10.48
Surprise	0.00	0.95	**87.14**	0.48	0.48	10.48
Angry	1.43	5.24	0.48	**84.29**	7.14	5.24
Disgust	3.33	3.81	1.90	5.71	**79.05**	4.29
Fear	1.90	4.76	8.09	3.18	3.81	**67.14**

Fig. 8.3: Confusion matrix of expression classification for LGBP with angle classification.

	Happy	Sad	Surprise	Angry	Disgust	Fear
Happy	**92.86**	1.43	0.48	1.90	1.90	2.86
Sad	1.43	**62.85**	0.95	6.67	13.33	13.81
Surprise	1.43	3.81	**89.52**	0.48	0.95	9.52
Angry	0.95	6.19	0.95	**84.29**	4.76	2.38
Disgust	0.95	2.38	0.48	3.33	**74.23**	6.19
Fear	3.81	10.95	7.62	2.86	4.76	**65.71**

Fig. 8.4: Confusion matrix of expression classification for LGBP without angle classification.

	Happy	Sad	Surprise	Angry	Disgust	Fear
Happy	**95**	0.0	0.0	0.0	5.0	0.0
Sad	4.65	**65.00**	2.32	16.25	4.65	6.97
Surprise	0.02	6.38	**68.08**	0.0	2.12	23.40
Angry	4.34	10.86	4.34	**65.21**	6.52	8.69
Disgust	0.00	5.40	2.70	8.10	**81.08**	2.70
Fear	0.00	10.25	15.38	5.12	10.25	**58.97**

Fig. 8.5: Confusion matrix of expression classification for DCLBP.

The following inferences can be made from the matrices shown in figure 8.2, 8.3, 8.4 and 8.5.

- The overall accuracy for angle prediction is very high (93.57).
- The best performing angle is *"Zero"* followed by *"Forty-Five"*. This is because of the fact that, with increase in the angle, less portion of the face is available for feature extraction and therefore poorer is the extracted feature.
- The best performing expression is *"Happy"* followed by *"Surprise"*. This is because the changes in the appearance of the face are quite distinctive and unique than the other expressions.
- In figure 8.3 "*Fear*" expression has the least accuracy because of its similarity with "*Surprise*" and "*Sad*" expression.
- In figure 8.4 similarity between "*Disgust*" and "*Sad*" could also be seen along with the trends observed in figure 8.3.
- In figure 8.5 similarity between "Surprise" and "Sad" could be seen along with the trends observed in figure 8.4.

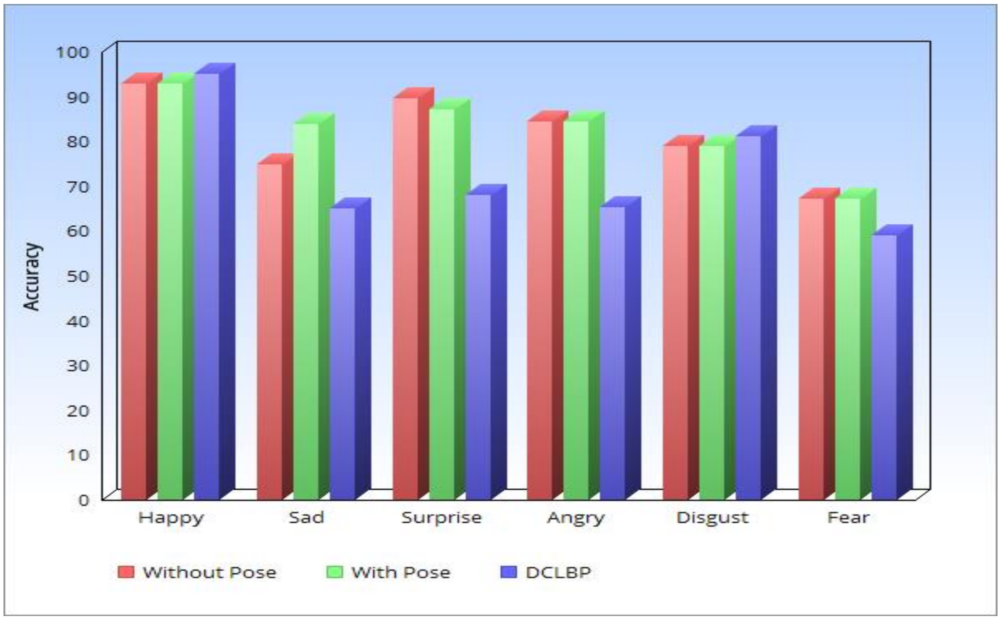


Fig. 8.6: Bar chart showing expression wise comparison of accuracies obtained using the three techniques.

- From figure 8.6 we can see that the expression "*Happy*" has performed well in all the three approaches and the difference among accuracies is quite low.
- "*Fear*" expression has performed the poorest in all the three approaches.
- Accuracy of "*Sad*" and "*Disgust*" expression is highest in LGBP with angle classification.
- Accuracy of "*Sad*" expression is very low using DCLBP approach as compared to the other two approaches.
- LGBP without pose classification outperforms LGBP with angle classification in "*Surprise*" expression.
- The accuracy of LGBP without angle classification is almost equal to LGBP with angle classification in "*Happy*", "*Anger*" and "*Fear*" expression.

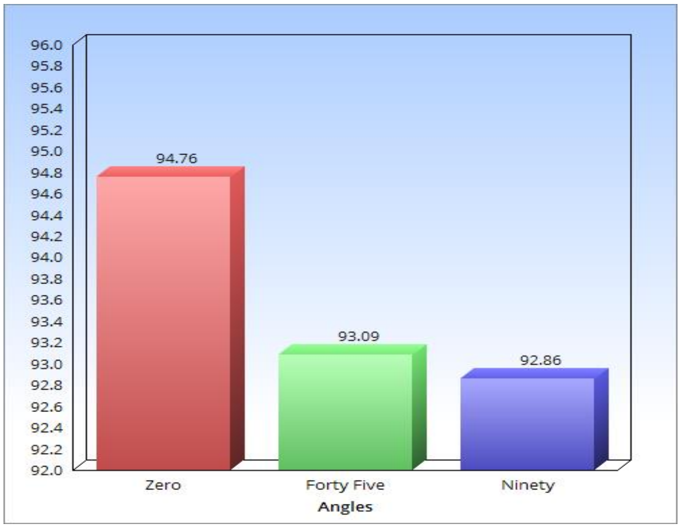


Fig. 8.7: Bar chart showing accuracy of angle prediction

We applied the best performing method i.e. LGBP with pose classification on our own dataset created at IIIT Allahabad and the results for testing were as follows:

- ✓ Accuracy for Angle Classification: **70.19%**
- ✓ Accuracy for Expression Classification: **20.97%**

We can see that the accuracy is low as compared to that of KDEF dataset. This is because of the following reasons:

1. The quality of images is low as compared to KDEF dataset.

2. KDEF was created under the following constraints:
    - Uniform T-shirt colour of the subjects
    - Soft even light for shooting expressions in multiple angles
    - Use of a grid to centre the subject’s face during shooting
    - Subjects seated at a distance 3meters from the camera
3. Since the images were clicked manually by hand, it was not possible to enforce the above constraints while creating our dataset, so the angle accuracy is also low.

9. Conclusion and Future Scope

Our project shows a comparative study among three methods used for classifying non-frontal facial images on the KDEF dataset. Feature extraction techniques like LGBP and DCLBP were evaluated. Multi-class SVMs were used to classify angle and expressions. An overall accuracy of 82.38% was obtained using LGBP with angle classification. LGBP with angle classification outperformed DCLBP by around 8% for expression recognition. The best performing expression “happy” achieved an accuracy of 92.86% in LGBP with angle classification approach. The poorest performing expression “fear” achieved an accuracy of 67.14%.

The limitations of the project is that it works only on three yaw angles of 0°, 45° and 90°. For practical applications, this should be extended such that poses and expressions at any arbitrary angles of the face is identified by the computer. This can be implemented using 3D face modelling for continuous angles. In case of face recognition, the methods of using 3D face model has been proven to achieve better recognition results than 2D facial images due to its robustness to the poses, scales and lighting variations. Hence, this can be extended to the problem of expression recognition. This is a complex project in itself and given the time constraints, we were unable to investigate further in this domain.

10. References

[1] Shahera Hossain, "Study on Co-occurrence-based Image Feature Analysis and Texture Recognition Employing Diagonal-Criss cross Local Binary Pattern", Web: http://hdl.handle.net/10228/5282

[2] B. Santra and D. Mukherjee, "Local saliency-inspired binary patterns for automatic recognition of multi-view facial expression", in Image Processing (ICIP), 2016 IEEE International Conference on, 2016.

[3] A. Majumder, S. Chowdhury and J. Sil, "Gabor Filter Based Face Recognition Using Non-frontal Face Images", in WCI 15 Proceedings of the Third International Symposium on Women in Computing and Informatics, Kochi, India, 2015, pp. 419-425.

[4] S.Moore and R. Bowden "Local binary patterns for multi-view facial expression recognition", Computer Vision and Image Understanding, vol. 115, no. 4, pp. 541-558, 2011.

[5] S. Liao, W. Fan, A.C.S. Chung, D.Y. Yeung, "Facial expression recognition using advanced local binary patterns, tsallis entropies and global appearance features", in: ICIP, 2006, pp. 665–668.

[6]W. Zheng, H. Tang, Z. Lin and T. S. Huang; Computer Vision, 2009 IEEE 12th International Conference, 2009.

Comments and Suggestions