Design of a Real-Time Human Emotion Recognition System

D. V. Ashwin, Abhinav Kumar, and J. Manikandan(✉)

Department of ECE, Crucible of Research and Innovation (CORI), PES University, 100-Feet Ring Road, BSK Stage III, Bangalore 560085, India
ashwindv75@gmail.com, abhinav.kumar685@gmail.com, manikandanj@pes.edu

Abstract. Emotion recognition systems are in huge demand to understand the emotion of human towards human, animals, computers, machines and systems. This has influenced such systems to be employed for applications in various domains such as website customization, study of audience reaction in theaters, gaming, software engineering, education and many more. In this paper, a real-time emotion recognition system is designed which is capable of recognizing six human emotions of any person in front of the camera, without any prior information about the person. This is achieved using a combination of both geometric and appearance based features. In order to assess and enhance the performance of the proposed design, the system is tested using standard CK+ datasets too. The system designed is evaluated using three different classifiers and their results are reported. Maximum accuracy of 98.73% is achieved by the system. The proposed system is designed using open source software and can be used for various IoT based applications too.

1 Introduction

Human emotions are considered as one of the important mediums through which vital information can be retrieved for various applications. Humans express their emotions in different ways, namely facial expression, body language, tone of voice, words uttered etc. [1, 2]. The facial expressions are very prominent as reported in [3], and the ability to recognize human emotions using a real-time automated system can provide significant impact on several areas like marketing, gaming, e-learning, entertainment and other applications that involve human computer interaction.

Design of emotion recognition systems for various applications is reported in literature for Human Computer Interactions (HCI) in [4–8] and many more. In this paper, a novel attempt is made to detect human emotions using facial expressions that doesn't require any prior information about the person under test. In other words, the system designed is trained only once and it works fine in detecting human emotions of any person sitting in front of the system, without any further training. The proposed system is designed using open source software (OSS) namely OpenCV, dlib and skimage. It employs a combination of geometry-based features proposed in [9] and appearance-based features proposed in [10] for better recognition accuracy.

N. Kumar and A. Thakre (Eds.): UBICNET 2017, LNICST 218, pp. 177–188, 2018.
https://doi.org/10.1007/978-3-319-73423-1_16

The organization of the paper is as follows. The motivation for proposed work is reported in Sect. 2. The design of proposed real-time human recognition system is explained in Sect. 3. The performance evaluation of system designed is reported in Sect. 4 followed by Conclusion and References.

2 Motivation

The work of Darwin in 1872 [11] attracted the attention of many behavioural scientists towards emotion recognition and nearly after a century in 1978 Suwa made the first attempt to automate the process of analyzing facial expressions in [12]. Lot of work has been carried out in this domain and a detailed survey of the existing work can be found in [13–15]. A few notable limitations of emotion recognition systems include the dependency on prior knowledge such as person specific neutral expression, location of eyes as in [16, 17], use of time and memory intensive algorithm like Gabor-wavelet algorithm in [18], use of algorithms like optical flow used in [19], which are sensitive to factors which cannot be easily controlled like background lighting, use of methods which has limitations on generalizing the system for other datasets [7, 8]. These limitations motivated the design of proposed system, which is completely automated and does not require any prior knowledge. The system is fast and the performance is not affected by factors like background lighting, tilted faces, face size, colour of skin etc. The system works fine with real-time input as well as for standard datasets.

3 Proposed System

The block diagram of proposed real-time human emotion recognition system is shown in Fig. 1 and the working of each block is explained in detail in following subsections. It may be observed from Fig. 1 that two features (Geometrical and Local binary patterns (LBP)) are extracted in parallel for the system designed.

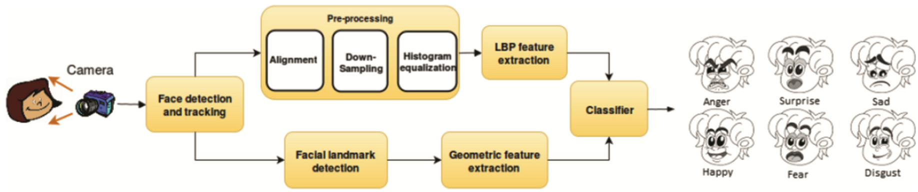


Fig. 1. Block diagram of proposed human emotion recognition system

A Face Detection and Tracking

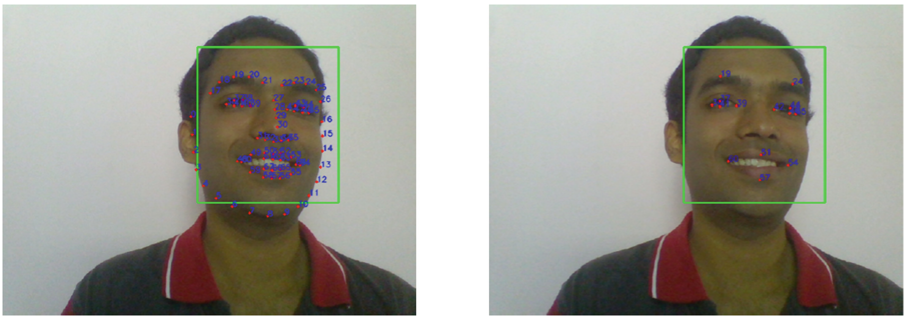
An image of size 1280 × 720 pixels is first captured using laptop camera and the system detects a face in given image. In case, no face is available in the image, the system will continue to capture images till a face is detected in the image. The face detection is accomplished using Viola-Jones algorithm proposed in [20] and is implemented using OpenCV [21] for proposed system. Viola-Jones algorithm is a Haar feature based cascade classifiers and it comprises of four major steps namely converting image into integral image, extracting haar features from integral image, selecting important features using adaboosting algorithm [22] and finally using the selected features in a cascade form to detect faces in a given image. Viola Jones algorithm detects all the faces in an image, whereas the system focuses on the face which is closest to the camera. The input and output of face detection and tracking block is shown in Fig. 2. The green rectangle tracks the movement of face in the video. It was observed during implementation that the face detection rate is slow and hence correlation tracker proposed in [23] and implemented in dlib [24] is used for the system to enhance the speed of tracking.

Input Image | Image with Face Detection

Fig. 2. Performance of face detection block

B Facial Landmark Detection

Ensemble of regression trees method proposed in [25] is used to detect the facial landmarks on the filtered face as shown in Fig. 3. The ensemble of regression trees method is implemented in dlib library. The pre-trained model in dlib library provides 68 points as shown in Fig. 3a, whereas only a sub set of 14 points is used in proposed work to extract the geometric features as shown in Fig. 3b.



(a) 68 Facial Landmarks | (b) 14 Facial Landmarks

Fig. 3. Performance of facial landmark detection block

C Geometrical Feature Extraction

The following nine geometric features are extracted from the 14 points obtained from the facial landmark detection block. Let $d(p_i, p_j)$ denote the Euclidean distance between the points with index i and j respectively and g_i be the ith geometrical feature extracted.

Distance between eye lid and eyebrow is computed as

$$g_1 = \frac{d(p_{19}, p_{39}) + d(p_{24}, p_{42})}{2} \quad (1)$$

Distance between eye lid and lip is computed as

$$g_2 = \frac{d(p_{39}, p_{51}) + d(p_{42}, p_{51})}{2} \quad (2)$$

Width of the mouth is computed as

$$g_3 = d(p_{54}, p_{60}) \quad (3)$$

Height of the mouth is computed as

$$g_4 = d(p_{51}, p_{57}) \quad (4)$$

Ratio of mouth width to mouth height is computed as

$$g_5 = \frac{g_3}{g_4} = \frac{d(p_{54}, p_{60})}{d(p_{51}, p_{57})} \quad (5)$$

Distance between eye brows is computed as

$$g_6 = d(p_{19}, p_{24}) \quad (6)$$

Distance between eye lids of an eye is computed as

$$g_7 = \frac{d(p_{37}, p_{41}) + d(p_{44}, p_{46})}{2} \quad (7)$$

Average of half of upper lip length is computed as

$$g_8 = \frac{d(p_{51}, p_{60}) + d(p_{51}, p_{54})}{2} \quad (8)$$

Average of half of lower lip length is computed as

$$g_9 = \frac{d(p_{57}, p_{60}) + d(p_{57}, p_{54})}{2} \quad (9)$$

The geometrical features extracted from 14 facial landmark points are shown in Fig. 4 with white lines connecting the facial landmark points. All the above mentioned geometrical features are sensitive to scale variations and hence an evaluation of normalization methods is carried out using features such as width of the rectangle around face, height of rectangle around face, area of rectangle around face, ratio of width to height of rectangle around face. The normalizing equation is formulated as

$$f_i = \frac{g_i}{D} \tag{10}$$

where i = 1, 2, …, 9. Let us consider H and W to be the height and width of the rectangle around face then the normalizing parameter D in (10) is given as

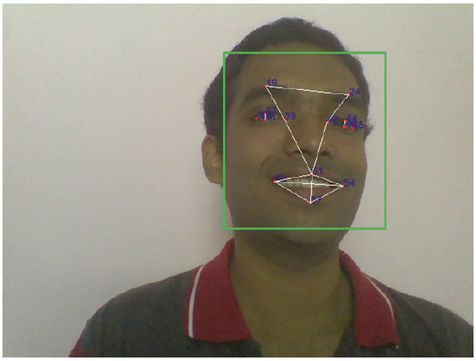


Fig. 4. Extraction of geometrical features from face

$$\mathrm{D} = \mathrm{H}\ \textit{for height normalization} \tag{11}$$

$$\mathrm{D} = \mathrm{W}\ \textit{for height normalization} \tag{12}$$

$$\mathrm{D} = \mathrm{H} \times \mathrm{W}\ \textit{for height normalization} \tag{13}$$

$$\mathrm{D} = \mathrm{H/W}\ \textit{for height normalization} \tag{14}$$

$$\mathrm{D} = 1\ \textit{for height normalization} \tag{15}$$

In order to assess and finalize among the above mentioned normalizing methods, a performance evaluation experiment was carried out as follows. The experiment involved a person moving towards the web-cam of a laptop from a distance with same expression on the face. It is well known that when the person is away from the camera, the size of detected face in that particular frame will be small and hence the Euclidean distance between the selected facial landmarks will be small. Similarly, when the person moves towards the camera, the size of detected face increases and hence Euclidean distance between the selected facial landmark also increases. During this experiment, all the nine geometrical features were extracted from all the frames of the video sequence. A total of 1474 frames/image were obtained during the process, out of which 266 images were having the emotion of anger, 273 images for disgust, 231 images for fear, 224 images for happy, 252 images for sad and 228 images for surprise. The above mentioned normalizing approaches were individually applied to all the nine features. Average deviation was computed for each case and all the nine features using the equation

$$\delta_{avg} = \frac{1}{9}\sum_{k=1}^{9}\frac{\sigma_k}{\mu_k} \times 100 \quad (16)$$

where σ_k represents standard deviation of k^{th} feature given as

$$\sigma_k = \frac{\sum_{i=1}^{n}(g_k^i - \mu_k)^2}{n} \quad (17)$$

where μ_k represents the mean of k^{th} feature and is given by

$$\mu_k = \frac{1}{n}\sum_{i=1}^{n} g_k^i \quad (18)$$

where n represents number of frames in the video sequence, g_k^i represents value of k^{th} feature in i^{th} frame.

The performance evaluation of normalizing methods was carried out using LDA classifier and Leaving-out-one method to obtain the recognition accuracy and the results are reported in Table 1. It is observed and concluded from Table 1 that Height and Width normalization gave best results with minimum deviation and best recognition accuracy. Hence, width normalization is considered for the proposed system with normalizing parameter D as the width of eye given as

Table 1. Evaluation of different normalization methods

SI No.	Normalizing method	Deviation (δ_{avg}) in %	Accuracy in %
1	UnNormalized	17.65	99.72
2	AreaNormalized	19.75	99.66
3	RatioNormalized	17.66	99.72
4	HeightNormalized	5.89	99.79
5	WidthNormalized	5.87	99.79

$$D = \frac{d(p_{36}, p_{39}) + d(p_{42}, p_{45})}{2} \quad (19)$$

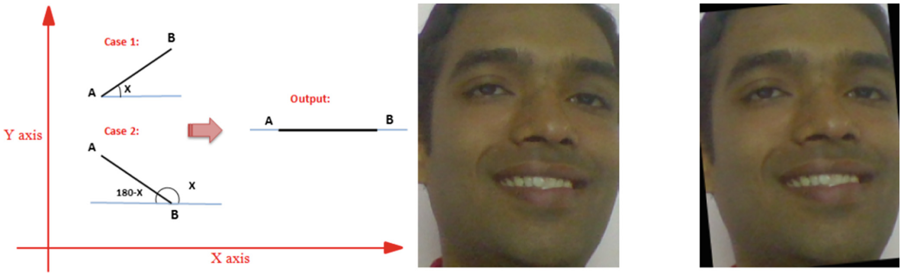
and the nine geometric features are normalized as

$$f_1 = \frac{g_1}{2D}\ f_2 = \frac{g_2}{D}\ f_3 = \frac{g_3}{D}\ f_4 = \frac{g_4}{D}\ f_5 = \frac{g_5}{D}\ f_6 = \frac{g_6}{2D}\ f_7 = \frac{g_7}{0.5D}\ f_8 = \frac{g_8}{D}\ f_9 = \frac{g_9}{D} \quad (20)$$

Different features were scaled by different values to make sure that all the normalized features lie in the same range avoiding feature dominance i.e., it removes the possibility of any feature with a large value dominating other features with relatively smaller values.

D Pre-processing for LBP Feature Extraction

In order to perform pre-processing for LBP feature extraction, the face is cropped and considered as Region of Interest (ROI). After cropping, the face alignment is performed by rotating the cropped image in such a way that the line joining points 36 and 45 are parallel to the horizontal reference line. Illustration of rotating a line to make it parallel to horizontal line is shown in Fig. 5, where point A and B can be considered as landmark points 36 and 45 respectively. The angle of rotation is calculated as



Procedure employed for Face alignment Cropped Input Face Aligned Face

Fig. 5. Illustration of face alignment

$$\theta = \tan^{-1}\left(\frac{p_{45}^{y} - p_{36}^{y}}{p_{45}^{x} - p_{36}^{x}}\right) \tag{21}$$

where p_{45}^{x}, p_{45}^{y}, p_{36}^{x}, p_{36}^{y} represents x and y co-ordinates of point p_{45} and p_{36} respectively. The results of input cropped face and its alignment are shown in Fig. 5.

Next step is to down-sample the aligned face to 108 × 147 pixels as shown in Fig. 6. This ensures that different parts of the face share almost the same location irrespective of input face. The resizing is followed by histogram equalization to improve the contrast of the image as shown in Fig. 6.

Resized image Histogram equalized image

Fig. 6. Preprocessing of cropped input face

E Local Binary Patterns (LBP)

Local Binary Pattern (LBP) operator compares the pixel value at any point with its neighbouring pixel values to generate a binary number representing the pattern. The histogram of LBP image is a robust feature descriptor against variations in illumination. Uniform LBP ($LBP^{u2}_{8,2}$) approach proposed in [26] is used after dividing the 108 × 147 pixels face images into blocks of pixel size 18 × 21, providing a better trade-off between recognition performance and feature vector length. The input face images are divided into 42(6 × 7) blocks as shown in Fig. 7 and LBP features of length 59 are extracted from each block. The features extracted from each block are concatenated to obtain the LBP histogram of length 2478(59 × 42). The LBP feature extraction for proposed system is implemented using LBP function with 'uniform' method available in scikit-image [27] to determine the pattern. The dimensionality reduction of LBP features is performed using PCA (Principal Component Analysis).

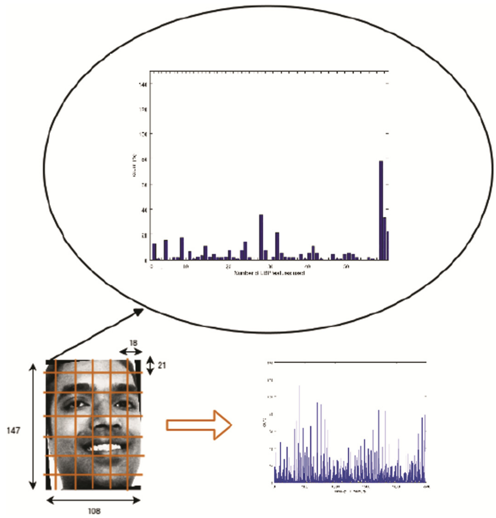


Fig. 7. LBP feature extraction

Implementation of PCA in scikit-learn [28] was used for the proposed work. It was observed during analysis that PCA of feature size 80 gave best results and hence the same is used in this work.

F Classifier Design

The features extracted from input image includes 2478 appearance based features and 9 geometric features. Different combinations of input features are fed to classifiers for assessing the performance of the system designed. The three different classifiers considered are K-Nearest Neighbour (KNN), Linear Discriminant Analysis (LDA) and Support Vector Machines (SVM). These classifiers are implemented using scikit-learn library proposed in [28]. The different architectures and combination of inputs considered for evaluation in this work are shown in Fig. 8.

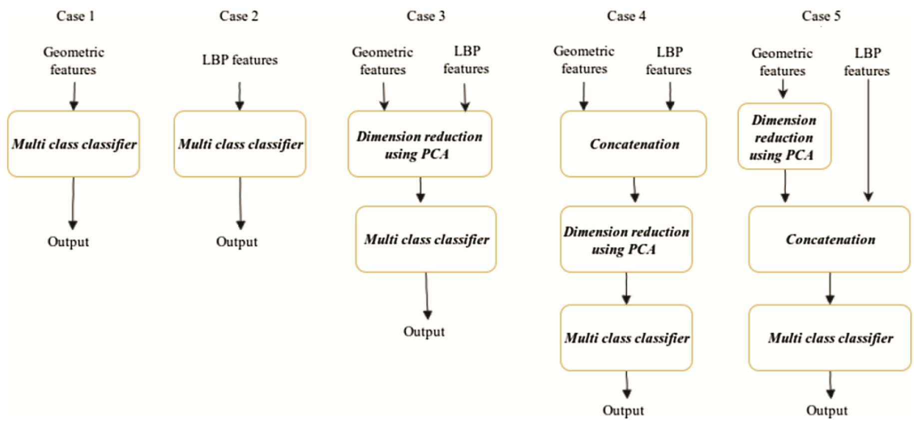


Fig. 8. Different architectures considered for proposed work

4 Experimental Results

The proposed human emotion recognition system consists of a laptop with the software running on it and the system is capable of detecting the emotion of any person sitting in front of it as shown in Fig. 9, without any additional training. The results obtained for different emotions using the system are also shown in Fig. 9 and satisfactory performance was observed on real-time datasets.

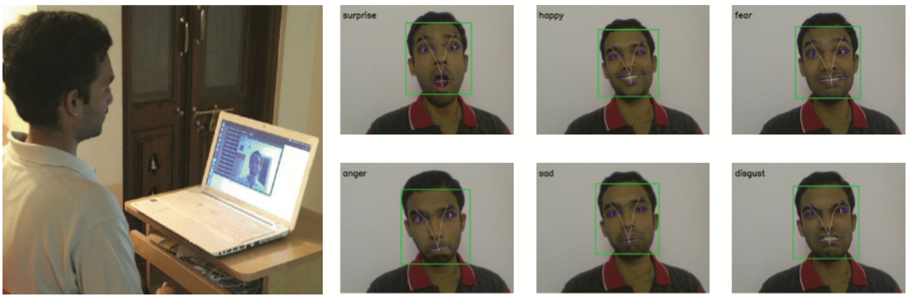


Fig. 9. Hardware setup for testing and the results obtained from the system designed

In order to quantify the performance of the proposed system using different classifiers and architectures, experiments were carried out on extended Cohn-Kanade dataset (CK+) which is one of the most widely used data base for facial expression recognition. The database consists of 593 sequences of 123 subjects. Each image sequence starts with neutral expression and ends with a peak expression. The offered peak expression is fully coded by Facial Action Coding System (FACS) using FACS investigator guide. After applying perceptual judgement to the facial expression labels, only 327 of the sequences were for the human facial expressions: 45 for anger (An), 18 for contempt (Co), 59 for disgust (Di), 25 for fear (Fe), 69 for happiness (Ha), 28 for sadness (Sa) and 83 for surprise (Su). Out of these datasets, only 309 images corresponding to the six basic emotions considered by our system i.e., anger, disgust, fear, happy, sad and surprise are used. The results obtained from these datasets for different architectures are reported in Table 2 using the leaving-one-out method for classification and it is observed that Case 5 using LDA classifier gave best results. It may be noted that this experimentation is not mandatory for real-time system designed and is carried out only to finalize the features and classifier to be used for the real-time system.

Table 2. Performance evaluation of proposed real-time emotion recognition system for CK+ dataset

Emotion	CASE 1			CASE 2			CASE 3			CASE 4			CASE 5		
	KNN	LDA	SVM	KNN	LDA	SVM	KNN	LDA	SVM	KNN	LDA	SVM	KNN	LDA	SVM
Anger	60.00	68.88	73.33	57.77	64.44	73.33	55.56	73.33	77.33	55.56	73.33	77.78	55.55	75.56	77.78
Disgust	75.00	83.33	78.33	43.33	86.67	83.33	50.00	88.33	83.33	50.00	88.33	83.33	50.00	88.33	83.33
Fear	64.00	96.00	76.00	16.00	76.00	68.00	32.00	80.00	68.00	12.00	80.00	68.00	12.00	84.00	68.00
Happy	94.12	98.53	92.65	69.13	98.53	95.06	63.24	95.59	97.10	63.23	95.59	89.70	63.23	97.10	89.71
Sad	14.28	42.86	42.86	3.57	46.43	42.86	3.57	71.43	42.86	3.57	71.43	57.14	3.57	67.88	57.14
Surprise	98.73	98.73	96.20	77.22	97.47	94.94	78.48	96.20	94.94	78.48	96.20	88.61	78.45	98.73	88.61
Accuracy	**76.72**	**85.9**	**81.97**	**54.10**	**84.26**	**82.51**	**55.41**	**87.54**	**83.55**	**53.76**	**87.54**	**81.63**	**53.76**	**88.86**	**81.64**

The proposed system is designed using a Toshiba laptop with following configuration: Quadcore AMD Processor operating at 1.5 GHz with 8 GB RAM and 1 MB L2 cache. Details about the time taken by various blocks in the system are reported in Table 3. Face detection takes 275 ms, but it is performed only once every 2 s. Once a face is detected, only face tracking is performed which takes only 60 ms and this significantly improved the performance of our system. The computation time can be further reduced on using a system with better configuration.

Table 3. Time taken by various steps

SI No.	Step	Time taken in ms
1	Face detection	275
2	Face tracking	60
3	Landmark detection	7.6
4	Geometrical feature extraction	2
5	Pre-processing	0.45
6	LBP feature extraction	45
7	Classifier	0.001

5 Conclusion

Design and implementation of a real-time human emotion recognition system is proposed in this paper. The entire system is designed using freely available open source softwares and libraries only, motivating the readers to design their own system. The system works well with low resolution images as input too, which can be easily obtained from webcam, security cameras etc., which will be useful for IoT based applications, where most of the images captured have lesser resolution. The system designed is real-time, fully automated, doesn't require any prior information about the person in front of the camera, independent of factors like background lighting and needs just one time training. The system designed has potential applications for gaming, marketing, e-learning etc.

References

1. Izard, C.E.: Human Emotions. Springer Science & Business Media, Berlin (2013)
2. Mauro, R., Sato, K., Tucker, J.: The role of appraisal in human emotions: a cross-cultural study. J. Pers. Soc. Psychol. **62**(2), 301 (1992)
3. De la Torre, F., Cohn, J.F.: Visual analysis of humans: looking at people. In: Moeslund, T., Hilton, A., Krüger, V., Sigal, L. (eds.) Facial Expression Analysis, pp. 377–409. Springer, London (2011). https://doi.org/10.1007/978-0-85729-997-0_19
4. Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., Taylor, J.G.: Emotion recognition in human-computer interaction. Sig. Process. Mag. IEEE **18**(1), 32–80 (2001)
5. Anderson, K., McOwan, P.W.: A real-time automated system for the recognition of human facial expressions. IEEE Trans. Syst. Man Cybern. Part B (Cybern) **36**, 96–105 (2006)
6. Bartlett, M.S., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I., Movellan, J.: Recognizing facial expression: machine learning and application to spontaneous behavior. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), vol. 2, pp. 568–573. June 2005
7. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on local binary patterns: a comprehensive study. Image Vis. Comput. **27**(6), 803–816 (2009)
8. Saeed, A., Al-Hamadi, A., Niese, R., Elzobi, M.: Frame-based facial expression recognition using geometrical features. Adv. Hum.-Comput. Interact. **2014**, 4 (2014)
9. Thanh Do, T., Hoang Le, T.: Facial feature extraction using geometric feature and independent component analysis. In: Richards, D., Kang, B.-H. (eds.) PKAW 2008. LNCS (LNAI), vol. 5465, pp. 231–241. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-01715-5_20
10. Shan, C., Gong, S., McOwan, P.W.: Facial expression recognition based on local binary patterns: a comprehensive study. Image Vis. Comput. **27**(6), 803–816 (2009)
11. Darwin, C.: The Expression of the Emotions in Man and Animals. John Murray, London (1872)
12. Suwa, M., Sugie, N., Fujimora, K.: A preliminary note on pattern recognition of human emotional expression. In: International Joint Conference on Pattern Recognition, vol. 1978, pp. 408–410. (1978)
13. Pande, S., Shinde, S.: A survey on: emotion recognition with respect to database and various recognition techniques. Int. J. Comput. Appl. **58**(3), 9–12 (2012)
14. Hemalatha, G., Sumathi, C.: A study of techniques for facial detection and expression classification. Int. J. Comput. Sci. Eng. Surv. **5**(2), 27 (2014)
15. Fasel, B., Luettin, J.: Automatic facial expression analysis: a survey. Pattern Recogn. **36**(1), 259–275 (2003)
16. Lucey, P., Cohn, J.F., Kanade, T., Saragih, J., Ambadar, Z., Matthews, I.: The extended cohn-kanade dataset (ck+): a complete dataset for action unit and emotion-specified expression. In: 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops, pp. 94–101. IEEE (2010)

17. Niese, R., Al-Hamadi, A., Farag, A., Neumann, H., Michaelis, B.: Facial expression recognition based on geometric and optical flow features in colour image sequences. Comput. Vis. IET **6**(2), 79–89 (2012)
18. Bartlett, M.S., Littlewort, G., Frank, M., Lainscsek, C., Fasel, I., Movellan, J.: Recognizing facial expression: machine learning and application to spontaneous behavior. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, vol. 2, pp. 568–573. IEEE (2005)
19. Yeasin, M., Bullot, B., Sharma, R.: From facial expression to level of interest: a spatio-temporal approach. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004. vol. 2, pp. II–922. IEEE (2004)
20. Viola, P., Jones, M.J.: Robust real-time face detection. Int. J. Comput. Vis. **57**, 137–154 (2004)
21. Bradski, G.: The OpenCV library. Dr. Dobb's J. Softw. Tools **120**, 122–125 (2000)
22. Freund, Y., Schapire, R.E.: A short introduction to boosting. J. Jpn. Soc. Artif. Intell. **14**, 771–780 (1999)
23. Danelljan, M., Häger, G., Khan, F., Felsberg, M.: Accurate scale estimation for robust visual tracking. In: British Machine Vision Conference, Nottingham, 1–5 September 2014. BMVA Press (2014)
24. King, D.E.: Dlib-ml: a machine learning toolkit. J. Mach. Learn. Res. **10**, 1755–1758 (2009)
25. Kazemi, V., Sullivan, J.: One millisecond face alignment with an ensemble of regression trees. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867–1874 (2014)
26. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 2037–2041 (2006)
27. van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T., Scikit-Image Contributors: Scikit-image: image processing in Python. PeerJ **2**, e453 (2014)
28. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**(1), 2825–2830 (2011)