

Automatic Attendance System Using Face Recognition Technique



Rakshanda Agarwal, Rishabh Jain, Rajeshkannan Regunathan
and C. S. Pavan Kumar

Abstract Recognition of human face is an important domain in unique identification of humans. It is currently being widely used in many industrial applications, such as video monitoring systems, human–computer interaction, and automatic gate control systems and for securing networks. Every university uses some method of attendance to keep a record of the number of students or people who attended that particular lecture. This paper delineates a method for taking attendance of people in a classroom which integrates the face recognition technology using local binary patterns histograms (LBPH) algorithm, along with face detection by Haar feature-based cascades and distance-based clustering. The proposed system records the attendance of the people in a classroom environment autonomously and provides the user with an output as a spreadsheet describing the attendance.

Keywords Face detection · Face recognition · Clustering · Attendance

1 Introduction

Taking attendance by calling every student’s name or roll number consumes around 10–15 min of time. This being a taxing job for both teachers and students, and a new methodology needs to be implemented. This saved time and can be used for other important tasks such as teaching, doubt clarification. Calling attendance normally has

R. Agarwal · R. Jain · R. Regunathan (✉) · C. S. Pavan Kumar
School of Computer Science and Engineering,
Vellore Institute of Technology, Vellore 632014, Tamil Nadu, India
e-mail: rajeshkannan.r@vit.ac.in

R. Agarwal
e-mail: rakshandapramod.agarwal2014@vit.ac.in

R. Jain
e-mail: rishabhjain.2014@vit.ac.in

C. S. Pavan Kumar
e-mail: pavan540.mic@gmail.com

© Springer Nature Singapore Pte Ltd. 2019

A. J. Kulkarni et al. (eds.), *Proceedings of the 2nd International Conference on Data Engineering and Communication Technology*, Advances in Intelligent Systems and Computing 828, https://doi.org/10.1007/978-981-13-1610-4_53

many other drawbacks also; they are marking false attendance, missing attendance. All these issues create problems for the faculty. A proper way of handling such issues is machine vision. It uses image processing, which is a way to manipulate images using mathematical functions and by higher dimensional signal processing techniques to which the input can be an image, series of images or a video while the output can be provided in the form of an image. These processes are generally digitally performed, but it can also be done via optical and analog devices [1, 2].

To take attendance through a video input, the video must first be divided into frames and faces must be extracted. Now in these extracted faces, similar faces are clustered together through basic clustering algorithm. Once clustering is successful, we have a training database which is trained, and the clusters are matched with this database. If a match is found attendance is marked and if no match is there, the new images from the input cluster are appended to the database to make out database stronger and more efficient. This paper comprises of image processing and machine learning techniques that have been used to achieve our target that is an automatic attendance system which is used to take attendance with ease and accuracy, providing us the attendance list of students. This will not only save time but will also solve the above-mentioned issues.

2 Paper Preparation

Face recognition is achieved in various steps as described in Fig. 1 (first author's (Rakshanda Agarwal) image) which include face detection and face registration, learning and training phases, clustering or classification of images and then finally accessing the database to recognize the face.

2.1 Face Detection

Detecting a face marks the onset of human face recognition. Using face detection, we can determine the coordinates and scale of face in the given input frame. Face detection can be difficult at times because face patterns have different appearances. A few factors that cause variations are expressions, skin color, or common objects such as glasses or mustache. One of the main factors is lighting changes that also can affect face detection [3].

Face detection is derived from object detection using Haar feature-based cascade classifier which was proposed by Paul Viola and Micheal Jones. This is a machine learning-based approach. To detect a face, we need a lot of positive and negative images, i.e., images with and without faces. Once we get these faces, we need to extract features from it as shown which are used to classify images. Each feature when applied to the training set a best threshold is calculated, which is then used to

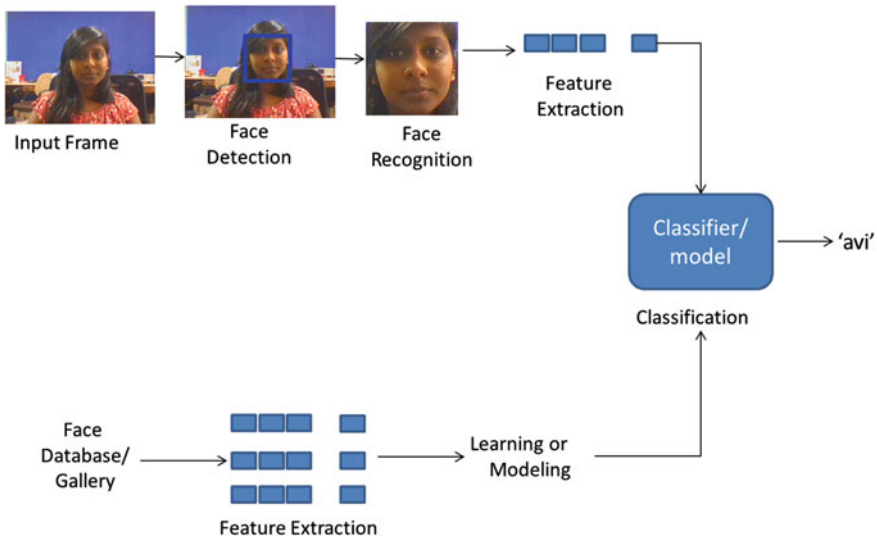


Fig. 1 Steps in face recognition (First author: Rakshanda Agarwal)

classify the face as positive or negative. This process continuous recursively until the required error rate or accuracy is achieved [4, 5].

2.2 Face Recognition

Face recognition for computer isn't as simple as it is for humans. Face recognition for computers is based on geometric features which we discussed in the face detection section above. There are various approaches to face recognition which include eigenfaces, fisherface, and local binary pattern histogram [6]. Our main solution here is obtained through local binary pattern histogram (LBPH). The main objective is to encapsulate this structure described by the local features in the image by pixel comparison to its neighboring pixels. To compute value for each pixel, compare the pixel to its eight neighbors and follow the pixels in a circular fashion, if the center pixel has a greater value in comparison with the neighbor, then give "0", else give "1". This gives us an eight digit binary number. Compute the histogram, for each combination formed. Normalize (concatenate) the histogram for every cell, this provides the feature vector for the entire face under process [7, 8].

The equation of the LBP operator is as follows:

$$\text{LBP}(x_c, y_c) = \sum_{p=0}^{p-1} 2^p s(i_p - i_c) \quad (1)$$

where

(x_c, y_c) is the central pixel, i_c is the intensity of central pixel, and i_p is the intensity of neighbor pixel.

The function $s(x)$ is defined as

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases} \quad (2)$$

2.3 Clustering

The process of grouping objects into sets in such a way that in a group is called cluster, and the objects are similar or have common properties in contrast to those in other groups or clusters. One common method is the k -means clustering algorithm. This algorithm has various applications in data mining, data compression, pattern recognition, and pattern classification. In k -means clustering, k data points are classified into the groups or clusters in order to reduce the geometric mean square distance between the data point and its nearest center [9].

3 Related Work

Computer vision is a vast branch which has object detection and recognition as important aspects. Face detection and recognition is one of the foremost applications. Machine learning is also equally important for these computer vision methods. All these concepts are interrelated to each other and hence can be used in various aspects.

Support vector machine (SVM) is a machine learning technique that can also be applied to computer vision. An algorithm to decompose data is proposed that can be implemented to train SVMs on datasets of higher magnitude and guarantees optimality. Its applicability is demonstrated in systems primarily detecting faces. SVM is used since it has a well-founded mathematical point of view that follows the risk minimization principle and can handle high-dimensional input vectors, they are appropriate in computer vision [10].

Face detection has an important application in human–computer interaction, video surveillance, etc. A new algorithm is proposed to detect colored faces in different illumination conditions as well as composite backgrounds. Based on color transformations, this method distinguishes skin regions over the whole image and then produces a face based on the position of these patches of skin. The difficulty of detecting faces under low and high luminescence is overcome by applying a non-linear transform. Thus, this detection method is better than the original one and has shown great results [11].

Another method of frontal face detection is through use of multilayered neural networks. A network is retinal connected to examine a small window of a face, and then it decides whether each window contains a part of a face or not. This process works by initiating multiple neural networks over to all portions of input images. This procedure can detect between 77.9 and 90.3% of faces with an acceptable number of false detections [12].

This method creates a model of the human face pattern by a few “face” and “non-face” clusters. A distribution-based model is built for face patterns and distance parameters are used for learning and to distinguish between “face” and “non-face” clusters. The distance matrix that is used to compute difference feature vectors and the “non-face” vectors included are both critical for the success of our system [3]. A mosaic approach for detection of human face consists of the higher two levels of the system architecture. While the lower level is an improved edge detection method, this method is efficient when the image size is unknown and can be used for black and white images without any prior info [13].

There are multiple face recognition methods: one of them is based on PCA, i.e., principal component analysis and LDA, i.e., linear discriminant analysis. In step one, a face image is projected to face subspace from original vector space through PCA, and in the second step, best linear classifier is obtained using LDA. The basic idea is to improve generalization of LDA when only a few samples per class are present. This hybrid classifier provides a useful framework for image recognition using PCA and LDA [14].

Eigenfaces are a method for recognition of faces and an approach to detect and identify human faces. This approach first tracks the human skull and then distinguishes the entire person by associating the facial features. This framework helps to detect and recognize new faces in an unsupervised manner. Also, it is efficient and relatively simple and has been observed to perform well in a sort of restricted type of environment [15].

Recognizing the frontal face with varying expressions, illumination, occlusion, and disguise is a big problem in face recognition as the results are not accurate and always different. A new method from sparse representation offers a solution to such problems. A clustering algorithm for recognition of a face is proposed and solves two main issues of face recognition: robustness to occlusion and feature extraction. The concept of sparse representation enables decision over the degree of occlusion that can be handled by the recognition algorithm and ways to maximize robustness to occlusion by selecting appropriate training images [16].

Using elastic bunch graph matching to recognize human faces from a large database wherein faces are treated as categorized graphs built using Gabor wavelet transformations. The new image distributions are mined by elastic graph matching methods and then can be matched by a similarity function. This structure is generic, flexible and is designed to recognize the members of a known group of objects. This also works on images which included mirror images and works great with faces of same pose [17].

A two way clustering is applied for data analysis of gene microarray data. Its chief purpose is to recognize the gene subgroup and model, so a stable partition emerges

whenever anyone of them is utilized to classify the other. An iterative clustering method is used to perform such search. This process is used to create small groups of genes that can be used as features to cluster subsets of the samples. This is achieved through a new algorithm known as CTWC—coupled two-way clustering [18]. It is also applied to analyze a dataset comprising of feature attribute patterns of different forms of cells. This classification method also helped in classifying cancerous and non-cancerous tissues. Two-way clustering can be used for both grouping genes in functionally similar groups and in grouping tissues based on the gene feature expression [19].

4 Methodology

We recognize faces through a video frame and consequently mark the attendance and update our recognizer, i.e., database. This process has been clearly described in this section of the paper. Firstly, the input to the system is a short video of people sitting in an area such as a classroom and our initial student database. For optimal results, the input parameters are as follows: People are looking toward the camera, the faces are in an unobstructed alignment to the camera, the camera should be kept at one's shoulder height, there should be proper lighting in the area, especially over the face, and the frame rate should be high preferably in the range of 30–60 frames per second (FPS) [20].

The video input is currently in RGB format with high FPS and hence has high volume. This makes the system processing time high which needs to be reduced. Several processes are optimized to achieve this. The first process is to convert each input to grayscale reducing one of the dimensions of the data to one-third. The second step is to reduce the data by physically reducing dimension values by removing the major part of the input video feed, i.e., by extraction of faces from the video feed. This will immediately reduce the data feed size. To extract faces Haar feature-based cascade classifier is used which uses machine vision to efficiently find the faces. These faces are resized to obtain data normalization providing us with better results. Inter-cubic interpolation is used to resize the images [21] and to increase the features like in Fig. 2 (The image is taken from opencv webpage which is an open-source platform.) that are recognized by the LBPH algorithm providing better features to be found faster.

A main machine vision technique is the local binary patterns histogram (LBPH) algorithm which is used here for both the steps that are clustering and face detection. Both steps improve the accuracy of the system. Clustering of images is done by matching images with the total set, giving one cluster of images for each person. Since this is done for each video input, we do not save the recognizer but build it dynamically [2].

After obtaining cluster of images, we match each cluster with the recognition database of the group. We obtain labels for our cluster and if some label is found to be missing, the user is asked to enter the label for the cluster. This happens broadly

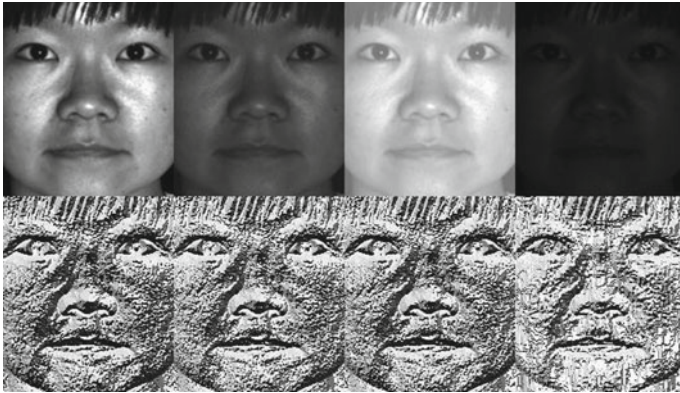


Fig. 2 Grayscale representation of LBPH features (from Opencv: free to download/open-source page mentioned in citation) [22]

for two conditions. First being when the person is scanned by the database for the first time, i.e., there is no data present about the person initially or the second case being that the features present in the new images are different in structure than what was observed before, i.e., the person could be present, but with a different facial feature or different lighting condition.

In the first case, a new label provided by the user is used to train the recognizer along with images from the cluster with various features. This increases the database length by proving a new class. While in the second case, the user provided tag is updated with these new features increasing the feature density of the recognizer leading to accurate output for a wider range of input. This provides an update to the database when the label is already present in the database, by appending this cluster to the database with same label. This will improve the efficiency of our recognizer and hence recognizing all the faces in the input video. This system has various advantages like improving the result over time while reducing the user intervention. The processes return high processor usage efficiency (PUE) when compared to other trivial methods. This ultimately leads us to achieve our goal.

4.1 Algorithm

Automatic_Attendance()

1. Create AllFaces = []
2. Load global face recognizer and Input Video
3. Open first frame in video and convert frame into grayscale
4. Detect all faces in the frame
5. For i in faces set AllFaces[i] = face

6. If next frame exists open next frame and go back to step 5
7. Create a local face recognizer and initialize TagValue = 0
8. Create a new cluster indexed by TagValue
9. Train local recognizer with top of AllFaces and TagValue
10. Remove top from AllFaces and set AllFaces[0] = NULL
11. Move to next element of AllFaces
12. Predict confidence of image by local recognizer
13. If (confidence < threshold) add image to current cluster, update local recognizer with image and remove image from AllFaces
14. If next element exists in AllFaces goto step 14
15. Increment TagValue by 1
16. If AllFaces != NULL goto step 11
17. Initialize index = 0
18. Open cluster with current index value
19. Predict each image in current cluster with global recognizer
20. if (image_confidence > threshold) Display an image from current cluster, ask user to enter Student ID and update global recognizer with images in current cluster and Student ID.
21. Else: Record the predicted tag as Student ID, update global recognizer with images giving confidence higher than threshold with Student ID.
22. Increment index value by 1
23. If cluster[index] exists goto step 21
24. Save global face recognizer

5 Conclusion and Future Work

Students' attendance being the foremost important task in every university is responsible for a huge amount of time consumption. Manually marking students' attendance has various drawbacks such as missing attendance, losing attendance sheet, and most importantly proxy issue. All these issues can be eradicated through our system. The only problem that our system faces is memory consumption, but since it reduces time and energy memory consumption which is not an issue. Are future endeavor includes converting this system to a software or an application so that it can be used throughout every university. We will also be working on reducing the overall time and space the system requires in execution, so that our system can be 100 percent accurate.

Declaration Images used in the paper belong to first author of the paper, and she has given her consent for use of the images. Authors take full responsibility for consequences arising from this in the case in future.

References

1. Gonzalez RS, Wintz P Digital image processing
2. Gu G, Perdisci R, Zhang J, Lee W (2008) BotMiner: clustering analysis of network traffic for protocol-and structure-independent Botnet detection. In: USENIX security symposium 2008 Jul 28, vol 5, no 2, pp 139–154
3. Sung KK, Poggio T (1998) Example-based learning for view-based human face detection. *IEEE Trans Pattern Anal Mach Intell* 20(1):39–51
4. © Copyright 2013, Alexander Mordvintsev & Abid K. Revision 43532856. http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_objdetect/py_face_detection/py_face_detection.html
5. Viola P, Jones MJ (2004) Robust real-time face detection. *Int J Comput Vis* 57(2):137–154
6. Wagner P (2012) Face recognition with python. Tersedia dalam: www.bytestfish.de (diakses pada 16 Februari 2015). 2012 Jul 18
7. ©Copyright 2011–2014, opencv dev team. http://docs.opencv.org/2.4/modules/contrib/doc/face_cerc/facerec_tutorial.html#local-binary-patterns-histograms
8. Ahonen T, Hadid A, Pietikainen M (2006) Face description with local binary patterns: application to face recognition. *IEEE Trans Pattern Anal Mach Intell* 28(12):2037–2041
9. Kanungo T, Mount DM, Netanyahu NS, Piatko CD, Silverman R, Wu AY (2002) An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Trans Pattern Anal Mach Intell* 24(7):881–892
10. Osuna E, Freund R, Girosit F (1997) Training support vector machines: an application to face detection. In: 1997 IEEE computer society conference on 1997 Jun 17 computer vision and pattern recognition, 1997. Proceedings, pp 130–136. IEEE
11. Hsu RL, Abdel-Mottaleb M, Jain AK (2002) Face detection in color images. *IEEE Trans Pattern Anal Mach Intell* 24(5):696–706
12. Rowley HA, Baluja S, Kanade T (1998) Neural network-based face detection. *IEEE Trans Pattern Anal Mach Intell* 20(1):23–38
13. Yang G, Huang TS (1994) Human face detection in a complex background. *Pattern Recogn* 27(1):53–63
14. Zhao W, Chellappa R, Krishnaswamy A (1998) Discriminant analysis of principal components for face recognition. In: Proceedings of third IEEE international conference on 1998 Apr 14 automatic face and gesture recognition, 1998, pp 336–341. IEEE
15. Turk MA, Pentland AP (1991) Face recognition using eigenfaces. In: IEEE computer society conference on 1991 Jun 3 computer vision and pattern recognition. Proceedings CVPR'91, pp 586–591. IEEE
16. Wright J, Yang AY, Ganesh A, Sastry SS, Ma Y (2009) Robust face recognition via sparse representation. *IEEE Trans Pattern Anal Mach Intell* 31(2):210–227
17. Wiskott L, Krüger N, Kuiger N, Von Der Malsburg C (1997) Face recognition by elastic bunch graph matching. *IEEE Trans Pattern Anal Mach Intell* 19(7):775–779
18. Getz G, Levine E, Domany E (2000) Coupled two-way clustering analysis of gene microarray data. *Proc Natl Acad Sci* 97(22):12079–12084
19. Alon U, Barkai N, Notterman DA, Gish K, Ybarra S, Mack D, Levine AJ (1999) Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci* 96(12):6745–6750
20. © 2017 Alamy Ltd. All rights reserved. <http://www.alamy.com/stock-photo-school-photograph-of-junior-girls-sitting-at-their-desks-with-open-52729837.html>
21. Maeland Einar (1988) On the comparison of interpolation methods. *IEEE Trans Med Imaging* 7(3):213–217
22. ©Copyright 2011–2014, opencv dev team. http://docs.opencv.org/2.4/modules/contrib/doc/face_cerc/facerec_tutorial.html#id22