Design and Implementation of Automatic Classroom Attendance System based on Face Recognition

Feng Song [1, 2, a], Ruige Liu [3, b]

[1] College of Institute of Information Engineering, Binzhou University, Binzhou, 256603, China;

[2] Binzhou Intelligent Manufacturing Engineering Technology Research Center, Binzhou University, Binzhou, 256603, China;

[3] College of Mechatronics Engineering, Binzhou University, Binzhou 256603, China.

[a] songfeng163@126.com, [b] lrg0903@163.com

Abstract

Traditional manual sign-up has the problems of generation check-in, time-consuming and laborious, and fingerprint, iris and other bio-check-in technologies are difficult to use and inconvenient to use. In order to solve the above problems, we developed an automatic classroom attendance system. The system is based on face recognition and developed in Visual Studio 2017 integrated development platform, using Emgu CV image library and Entity Framework. Several face recognition algorithms used in the system and Emgu CV image processing library are introduced. The implementation steps and key functions of the system are introduced. The system realizes the functions of multi-scale automatic detection, recognition and attendance in real-time video, improves the efficiency of course attendance, and has high popularization and application value.

Keywords

Classroom Attendance, Automatic Attendance, Face Detection, Face Recognition.

1. Introduction

College classroom is the main place for students to learn knowledge, develop their abilities and improve their comprehensive qualities [1]. However, the phenomenon of College Students' absence from class is becoming more and more serious. High attendance rate is the basic guarantee of teaching effect. Classroom attendance is a very effective measure to ensure students' attendance rate. Attendance records can let students know their own learning attendance process, but also let teachers and school managers know the students' learning attendance situation. Nowadays, the most common way of attendance in class is to use the traditional way of teacher's name-calling according to the list of students. However, because teachers teach more classes and have more students in each class, it is difficult for teachers to recognize all the students, resulting in the situation of substitute attendance and check-in from time to time, and the way of manual naming will also result in the waste of classroom time. In order to solve the problem of manual signature, some attendance systems using biometrics, such as fingerprint [2], iris [3], have appeared. However, fingerprint and iris recognition technology require professional equipment on the one hand, and contact information collection and inspection on the other hand, which is inefficient and inconvenient in use. The non-contact characteristics of face recognition technology make it very convenient to use. The face recognition technology which is developing continuously can now carry out real-time recognition with very high accuracy. In order to solve the problems of traditional roll call, fingerprint, iris and other roll call methods, we design an Automatic Classroom Attendance System based on Face Recognition, hereinafter referred to as ACASFR. The system uses Visual Studio (VS) 2017 integrated development

platform, uses Emgu CV image processing library to collect, train and build the face database of students in class. Using Entity Framework (EF) ORM framework, all information of the system is stored in MS SQL Server 2014 database. The system realizes the management and maintenance of teachers' information, teachers' semester course information, class information, students' information in classes, students' information in courses, and the inquiry and statistics of teachers' class attendance and attendance history..

2. Face Recognition Algorithms and Image Processing Library Used in the System

ACASFR system uses several commonly used face recognition algorithms, and uses Emgu CV image processing library to detect and recognize faces in real-time video images. This section briefly introduces several face recognition algorithms and Emgu CV library used in the system.

2.1 EigenFace

EigenFace is the name of a set of feature vectors used in computer vision problems of face recognition. The EigenFace recognition method was developed by Sirovich and Kirby (1987) [4] and used by Matthew Turk and Alex Pentland in face classification [5]. The eigenvector is derived from the covariance matrix of the probability distribution in the high-dimensional vector space of the face image. EigenFace itself constitutes the base set of all the images used to construct the covariance matrix. The process of implementing EigenFace is actually the process of Principal Component Analysis (PCA). It is used to simplify the processing of the problem. The process is as follows: In the two-dimensional space coordinates, a unit vector U is found to maximize the sum of projections of all data on U. In this way, the data can be separated as far as possible, and then the training sample and the test sample are projected onto the unit vector U. The Euclidean distance between the sample and each projected face is calculated, and the number of the sample with the smallest Euclidean distance is the face with the greatest probability.

2.2 FisherFace

Fisherface was invented by Ronald Fisher [6]. Fisherface is based on Linear Discriminant Analysis (LDA) theory, which is similar to the PCA used in EigenFace, is a method of mapping the original data into low-dimensional space. LDA and PCA start with the whole data. The steps of LDA dimension reduction are as follows:

1) Calculate the mean vector of each class sample in the data set.

2) The inter-class divergence matrix S_B and the intra-class divergence matrix S_W are calculated by the mean vector.

3) Solve the eigenvalue of the S_W^(-1) S_B W=λW and get the eigenvectors and eigenvalues of S_W^(-1) S_B.

4) The eigenvectors are arranged in descending order according to the size of eigenvalues, and the projection matrix W is composed of the first K eigenvectors.

5) Projection of sample points into a new subspace by D*K dimension eigenvalue matrix, Y=X*W.

Overall, the central idea of LDA is to maximize the distance between classes and minimize the distance within classes. Since LDA utilizes class member information and extracts a set of eigenvectors, the set of eigenvectors emphasizes the difference of different faces rather than the change of illumination, facial expression and direction. Therefore, compared with EigenFace, the Fisherface method is more insensitive to the changes of illumination and facial pose, which is helpful to improve the recognition effect..

2.3 LBPHface

LBP is short for Local Binary Pattern. LBP is an operator used to describe local features of images. LBP features have obvious advantages such as gray scale invariance and rotation invariance. It was proposed by T. Ojala, M. Pietik Kainen and D. Harwood [7][8] in 1994. Because of its simple

calculation and good effect, LBP features have been widely used in many fields of computer vision. LBP features are well known for their applications in face recognition and target detection.

LBPH is short for Local Binary Patterns Histograms. LBPH combines LBP features with spatial information of images. This representation method was proposed by Ahonen et al. in paper [9]. They divide the LBP feature image into m local blocks and extract the histograms of each local block. Then these histograms are connected in turn to form the statistical histogram of the LBP feature, that is, LBPH.

2.4 CascadeClassifier

Cascade classifier is a strong classifier of polynomial level by connecting several classifiers. The cascade classifier should be trained before it is used. How to train? To train with the eigenvalue of the target, for the face, the Haar feature is usually used to train.

Haar feature was first proposed by M. Oren, C. Papageorgious and others in the paper "Pedestrian detection using wavelet templates" [10]. Following improvements and developments, C.H. Messom and A.L. Barczak proposed the calculation method of integral histogram accelerating Harr feature [11], R. Lienhart, J. Maydt and others proposed several template types of Haar feature [12], and gradually formed Haar classifier in OpenCV. Haar feature, also known as Haar-like feature, is a simple and efficient image feature based on Haar wavelet with similar intensity differences in rectangular regions.

2.5 Emgu CV Image Processing Library

Emgu CV is a cross platform .Net wrapper to the OpenCV image processing library. The full name of OpenCV is Open Source Computer Vision, It was designed especially for computational efficiency with strong focus on real time applications. It is written in optimized C/C++, and can take advantage of multi-core processing. In Image processing it has been a great boon for the developers [13]. EmguCV allows OpenCV functions to be invoked from .NET compatible languages such as C, VB, VC++, Ironpython, etc. The wrapper can be compiled by Visual Studio, Xamarin Studio and Unity, and can run on Windows, Linux, Mac OS X, iOS, Android and Windows Phone.

OpenCV image library encapsulates some commonly used image processing algorithms, such as EigenFace, FisherFace, LBHFace, Cascade Classifier mentioned above, which have been implemented in OpenCV library. When using, the corresponding classes and methods can be called. Emgu CV packages the OpenCV library, so the above algorithms can be easily called in the Emgu CV library to realize the functions of face detection, model training and face recognition.

3. System Architecture Design

ACASFR system can use any type of camera, using several kinds of face recognition algorithms and Emgu CV image database introduced in Section 2, to automatically detect, recognize and attend work from real-time video. The core functions of ACASFR system are as follows:

1) Establishment and maintenance of department information, class information and teachers' semester course information;

2) Create basic information of middle school students in class and collect face image information;

3) Automatic race detection in real-time monitoring video;

4) Face recognition using several classes of face recognition algorithms described in section 2;

5) According to the recognized face, the corresponding student information is queried in the database and the attendance status is recorded;

6) Administrators and teachers can view attendance history and attendance statistics. Students can view their attendance history information.

Considering the convenience and validity of system management, the functions of administrator user information management, teacher user information management, system basic use information management, some commonly used dictionary information management, student image information

acquisition and face recognition model training are also added. The overall structure design of ACASFR system is as follows:

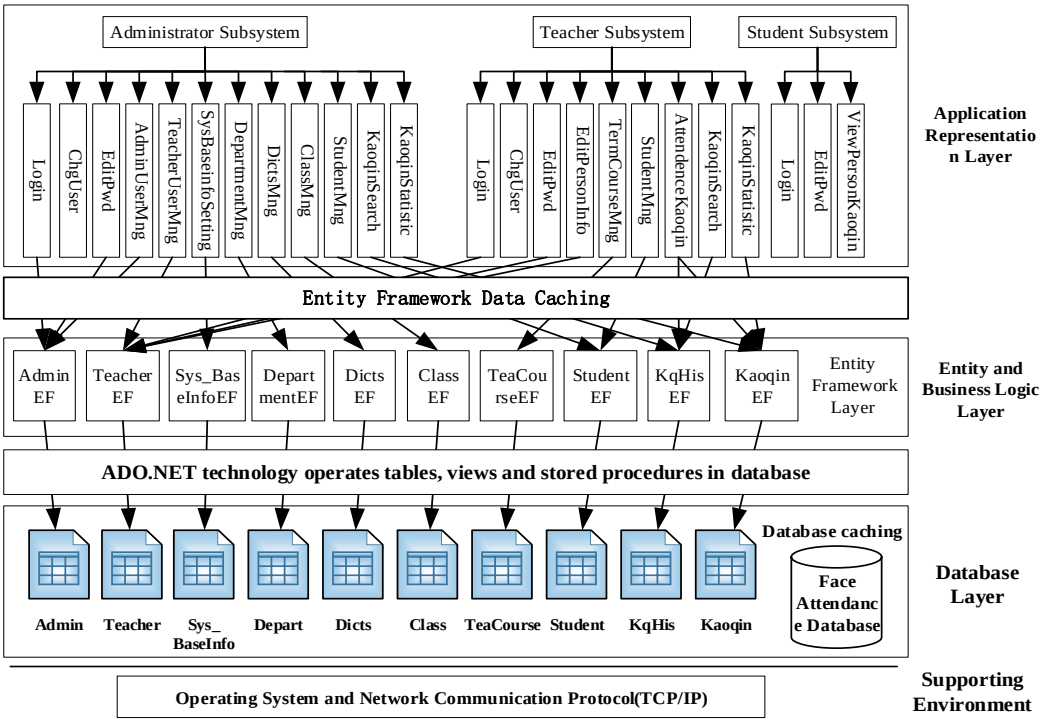

Fig. 1 ACASFR System Architecture Design

The ACASFR system is logically composed of four layers, from the bottom to the top, respectively:

1) Environment layer: The system uses C/S structure, runs on the operating system, communicates with the database server using the network communication protocol (TCP/IP);

2) Database Layer: Using SQL Server 2014 database management system for database management, this level provides data tables and query views used by ACASFR system;

3) Entity and Business Logic Layer: Based on EF Framework, this layer realizes mapping of data tables in the database to entity classes and generates corresponding business logic classes. After mapping, several kinds of operations of CRUD in the database can be accomplished by operating EF entity classes and business logic classes. EF Framework is based on ADO.NET technology. EF Framework can convert operations of EF business classes into the operation of data tables in database by ADO.NET technology. This can improve the efficiency of development. Using EF framework, the update of data table structure in database can be quickly updated to EF entity class and business class, which further improves the efficiency of development.

4) Application Representation Layer: This layer is the realization of each specific function. It implements CRUD operation of data table in database by calling entity class and business class based on EF framework. When querying, it can also use data cache of EF framework to improve query efficiency.

4. Realization of Key Functions of the System

The implementation procedure of ACASFR system is as follows:

1) A class attendance database named 'kaoqindb' is established in the database management system of SQL Server 2014, and various data tables and combined query views are created.

2) Create 'kaoqin' project in VS2017, add EF 5.x DbContext generator, select all data tables of kaoqindb database according to the wizard, and generate each EF entity class and business logic class.

3) Create various interactive interfaces, complete the basic module functions of administrator information, Department information, class information, system basic information, teacher information, teacher semester course information, commonly used dictionary table information, etc.

4) The basic information function of students is completed in the student management module, and the facial information of students is added. The facial information can be collected by camera, or the head photo of Student ID Card can be used to store the image information in the database in binary mode.

5) The class face information is downloaded from the database to the local computer after collecting the basic information of students and their face image information. At the same time, several commonly used algorithms described in Section 2 are used to train face models automatically and generate local face model data.

6) ACASFR system calculates the information of the course schedule on that day according to the start date of the current term set in the teachers' term course information and the basic information of the system in class. Teachers can use automatic attendance and manual attendance by clicking on their own timetable. Real-time video in camera is used for automatic attendance, and multi-person face checking and attendance are carried out online. The detected students are marked as normal attendance. The undetected students can use manual attendance module to attend.

7) School managers and teachers can check the attendance statistics of each course in time, and students can also check their attendance history.

Real-time video surveillance and timing image sampling, training of face classifier, multi-scale face detection, face recognition and classroom attendance are the key technologies in ACASFR system. The following are described in detail.

4.1 Real-Time Video Surveillance and Timing Image Sampling

In the process of collecting and establishing students' face information database and attendance in class, real-time video surveillance and timing image sampling are needed. The workflow of this process is as follows:

1) Setting the fixed time interval of clock 1 is 0.1s, which ensures no infinite loop and facilitates the subsequent timing image sampling;

2) Define the instance cap object of the global VideoCapture class and open the camera through the instantiation of new VideoCapture();

3) If the camera is turned on successfully, the clock 1 event is activated. If the camera is not turned on successfully, cap.Dispose() is used to release the camera object;

4) In the event of clock 1, a video image is captured by Mat frame = cap. QueryFrame() method. The system will automatically take advantage of the time interval of 0.1s set by the previous clock to continuously sample the image;

5) If the frame object is not empty, it will enter the process of face detection and recognition.

The workflow of the process is shown in Fig 2.

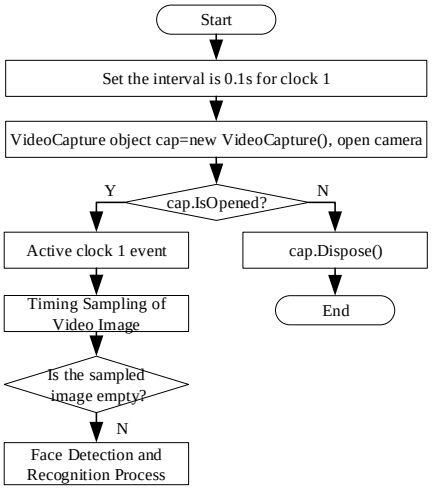

Fig. 2 Real-time video surveillance and timing image sampling process

4.2 Training of Face Classifier

After the collection and establishment of the student's face information database, the training of face classifier should be carried out. The process of training face classifier is as follows:

1) Define and instantiate the vectors stored in the training image using VectorOfMat images = new VectorOfMat() statement

2) Define and instantiate the vectors that save the training image labels by using the VectorOfInt labels = new VectorOfInt() statement;

3) Traverse the folder where the students' face images are stored, store all the students' images (the width and height of the images are 100 pixels) in images, and store the students' ID in labels;

4) Define a face recognizer with FaceRecognizer model and instantiate it with an algorithm in sections 2.1-2.3. Take FisherFaceRecognizer () as an example to instantiate the face recognizer model;

5) Using model.Train (images, labels) method to train face recognition model;

6) Use the mode.write ("pic/faceData.xml") method to store

the trained model data in the faceData.xml file of the pic folder.

The training process of face classifier is shown in Fig 3.

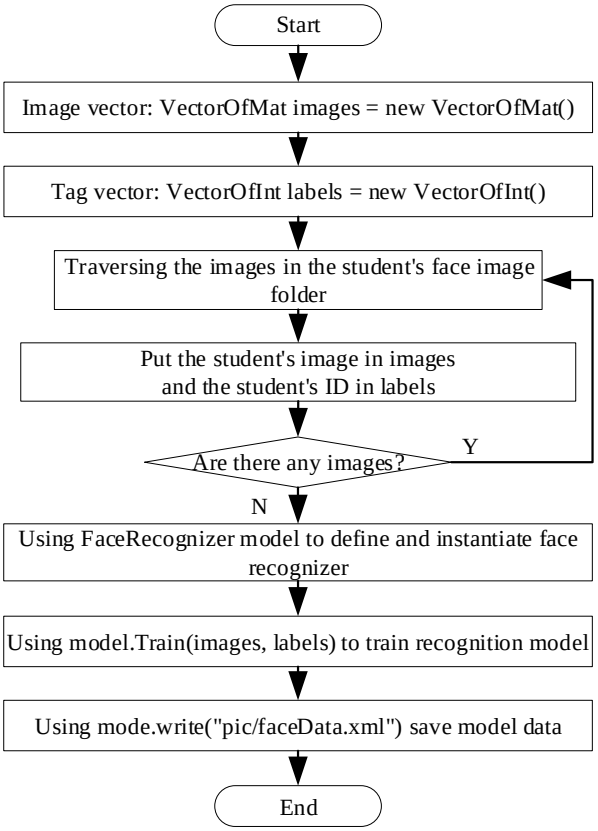

Fig. 3 Training flow of face classifier

4.3 Multiscale Face Detection

Emgu CV encapsulator provides a multi-scale face detection method, which can detect multiple face information in an image. The flow of multi-scale face detection in ACASFR system is as follows.

1) An example of cascade classifier is generated by using CascadeClassifier face_cascade = new CascadeClassifier ("haarcascade_frontalface_default.xml") statement.

2) Using CvInvoke.CvtColor (img, frameGray, ColorConversion. Bgr2Gray) statement to convert the color image sampled in Section 4.1 into gray image;

3) Using CvInvoke. GaussianBlur (frameGray, frameGray, new Size(3, 3), 0, 0) statement for Gaussian filtering to reduce noise in gray imag;

4) Using Rectangle[] facesDetect = face_cascade. DetectMultiScale(frameGray, 1.3, 2, new Size (80, 80)) statement for multi-scale face detection, and the detection results are stored in a rectangular array for the next face recognition.

The flow of multi-scale face detection is shown in Fig 4.

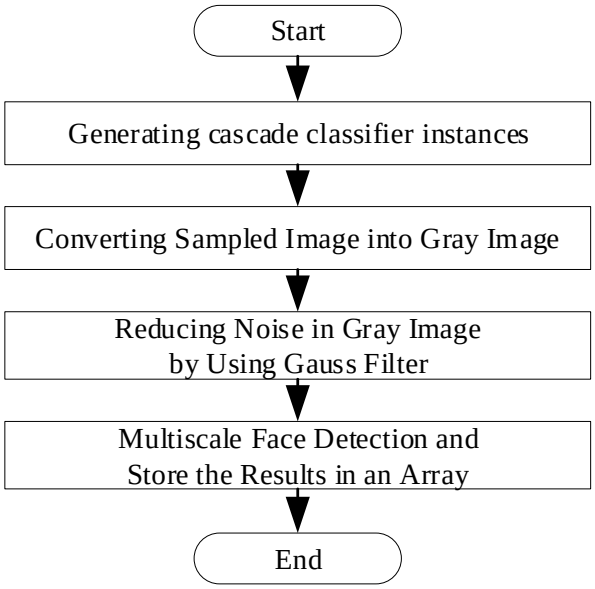

Fig. 4 Multiscale Face Detection Process

4.4 Face recognition

The process of face recognition is as follows:

1) Traversing through the rectangular array facesDetect detected in Section 4.3 defines the current traversal variable as Rectangle face;

2) Use CvInvoke.Rectangle(currentImage, face, new MCvScalar(0, 255, 0), 2) statement to mark a green rectangular border for faces;

3) Using Mat imageFace = new Mat(frameGray, face).Clone() statement to copy and generate a gray face area, gray face is used in recognition.

4) The image is adjusted to the same width and height as the training model, with a value of 100 pixels by using the CvInvoke.Resize (imageFace, imageFace, new Size(100, 100)) statement.

5) Define the recognition result class by using the FaceRecognizer. PredictionResult pr = new FaceRecognizer.PredictionResult() statement.

6) FaceRecognizer model is used to define the recognition model, and an algorithm in sections 2.1-2.3 is used to instantiate the model. The recognition classes selected here are identical to those in Section 4.2 training model. Otherwise, errors will occur, so Fisher FaceRecognizer () is used as an example to instantiate the model.

7) The recognition result is obtained by pr = model. Predict (image Face). The result contains two key values: the ID of the recognized face and the confidence distance between the face and the face in the model base. When the confidence distance is higher than a certain threshold, the recognition is considered to be unsuccessful. Otherwise, the recognition is considered successful and enters the classroom attendance process.

The process of face recognition is shown in Fig 5.

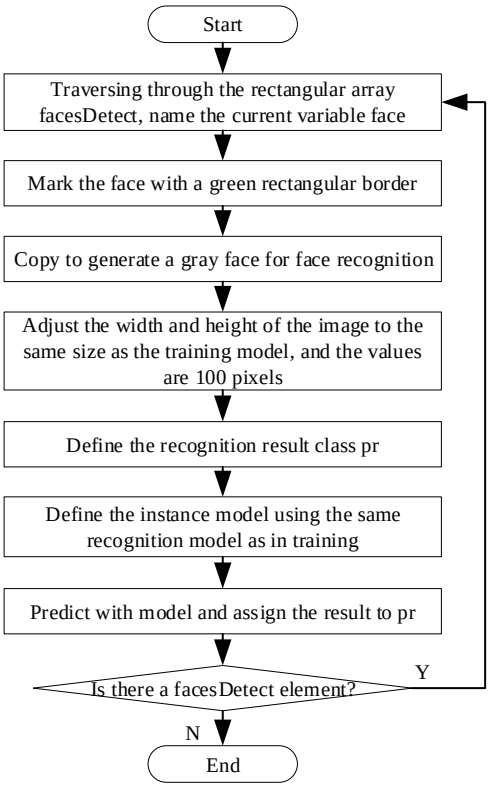

Fig. 5 Face Recognition Process

4.5 Attendance in Class

The process of attendance in class is introduced after the ID and reliability distance of face are obtained in Section 4.4. When the confidence distance is lower than the set threshold, the recognition is considered effective. At this time, the ID of the face is used to query the corresponding student information in EF framework. It should be noted that in the process of recognition, repeated queries of data will bring great pressure to the face recognition program and database management system if they have been communicating with the database. Therefore, this paper adopts the method of loading a list of global data named listStu in the current interface when the face attendance is initialized, which is used to store students' information in the open system. All queries in the process of identification are interactive queries for this listStu, no longer connecting and accessing the database, which improves the running efficiency of the program. After inquiring and finding students' information, we should update the students' information. At this time, we should connect the database to update the students' attendance status in real time, and mark the students' attendance status as normal attendance. For students who fail to attend automatically, the system defaults to truancy. For students who fail to attend automatically, teachers can take manual attendance in the system.

5. Attendance Function Test

After completing the ACASFR system, test the attendance function of the system. This section mainly tests two functions: automatic attendance and manual attendance.

Automatic attendance interface can automatically open the camera for real-time monitoring, sample video images by clock timer, multi-scale face detection, face recognition according to the set threshold, use green rectangular box to mark the position of face, and query student information in the database according to the corresponding student ID of the recognized face. Students' names are marked on the top of the green rectangular box, the student numbers and names identified are displayed in the information prompt box on the upper side of the window, and the normal attendance information is stored in the database. The automatic attendance function can quickly detect and complete the attendance of most students. The effect of the automatic attendance interface is shown in the figure below.

Fig. 6 Automatic Check Attendance

For student information not detected by the system, teachers can use the manual attendance function provided by the system to confirm attendance. The manual attendance function interface provides students' image information and various attendance states: Attendance, Truancy, Sick, Absence, Late, Leave Early. Teachers can choose a state according to the actual attendance of students, and then click the "Confirm Attendance" button to complete attendance. The manual attendance function ensures the accuracy and integrity of attendance information with human participation. The manual attendance interface is shown in the following figure.

Fig. 7 Manul Check Attendance

6. Conclusion

The ACASFR system developed in this paper solves the problem of traditional manual roll-call, which is time-consuming and laborious. It also solves the difficult and inconvenient problems of fingerprint and iris contact roll-call system, and can effectively improve the attendance rate in class. The characteristics of the system are as follows:

1) Any camera can be used for non-contact face detection and face recognition, which improves the convenience of use and reduces the cost of development and use;

2) Automatic face detection, recognition and attendance from real-time surveillance video can improve attendance efficiency;

3) It can identify and attend multiple people at the same time, and improve the efficiency of the system.

4) Statistical analysis function facilitates school administrators and teachers to grasp students' attendance situation in time, and also enables students to know their attendance situation in time.

Acknowledgements

This paper was supported by Scientific Research Foundation Project of Binzhou University (No. BZXYG1519), the Double Service Project of Binzhou University in 2017.

References

[1] Wu Xiangmei, Deng Yuanyuan, Pei Fenghua, Yu Songson, Pai Jiahui. The Design and Implementation of Mobile Classroom Attendance System based on the Face Recognition[J]. Journal of Software, 2018(1):5-8. (In Chinese)

[2] Mohamed and C. Raghu, "Fingerprint attendance system for classroom needs," in India Conference (INDICON), 2012 Annual IEEE. IEEE, 2012, pp. 433–438.

[3] S. Kadry and K. Smaili, "A design and implementation of a wireless iris recognition attendance management system," Information Technology and control, 2007, vol. 36, no. 3, pp. 323–329.

[4] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," J. Opt. Soc. Am. A, Vol. 4, No. 3, March 1987, 519-524.

[5] M. A. Turk and A. P. Pentland, ―Face Recognition Using Eigenfaces‖, [J] Proc. Of IEEE Conf. on Computer Vision and Pattern Recognition, June 1991, p. 586-591.

[6] R.A. Fisher, The Use of Multiple Measures in Taxonomic Problems[J], Ann. Eugenics, vol. 7, pp. 179-188, 1936.

[7] T. Ojala, M. Pietikäinen, and D. Harwood, Performance evaluation of texture measures with classification based on Kullback discrimination of distributions[J], Proceedings of the 12th IAPR International Conference on Pattern Recognition (ICPR 1994), vol. 1, pp. 582 - 585.

[8] T. Ojala, M. Pietikäinen, and D. Harwood, A Comparative Study of Texture Measures with Classification Based on Feature Distributions[J], Pattern Recognition, 1996, vol. 29, pp. 51-59.

[9] Ahonen, T., Hadid, A., and Pietikainen, M. Face Recognition with Local Binary Patterns [J]. Computer Vision- ECCV 2004 (2004), 469–481.

[10] Michael Oren, Constantine Papageorgiou, Pawan Sinha, Edgar Osuna, Tomaso Poggio. Pedestrian Detection Using Wavelet Templates [J], cvpr, July 1997.

[11] C.H. Messom, A.L. Barczak. High Precision GPU based Integral Images for Moment Invariant Image Processing Systems. Electronics New Zealand Conference (ENZCON'08), 2008.

[12] Rainer Lienhart and Jochen Maydt, An Extended Set of Haar-like Features for Rapid Object Detection [J]. IEEE ICIP 2002: p.900-903.

[13] Sneha Ramhari Suryavanshi, Lata J. Sankpal. Monitor Student's Presence in Classroom [J]. International Journal of Computing and Technology, 2016, 3(8):422-427.