Face Recognition System in Cell Phones Based on Text Message Service

Suman Deb, Sourav Biswas, Chinmoy Debnath,
Suddhasatta Dhar, and Priyanka Deb

Department of Computer Science and Engineering. National Institute of Technology Agartala, Jirania, Tripura - 799 055
{sumandebcs,souravbiswas.cs,suddha14}@gmail.com,
{chinmoy_cse,priyanka_nita.cse07}@ymail.com

Abstract. Face Recognition is a task human performs remarkably easily and successfully. This technology being easy to use, non-intrusive and simple to implement with less hardware requirements has a great advantage over other conventional biometrics. The goal of our work is to present face recognition with the help of sms/text messaging service on mobile platform to enlarge the scope of face recognition in the field of social security. This paper is aimed to help people recognizing any suspects or criminals as well as informing the concerned authorities with the help of sms. The use of mobile based platform gives the portability and versatility to this project. We plan to introduce the sms based face recognition technique which makes the work more scalable and also eliminates the use of GPRS or any other similar services, which is availed by less number of people compared to the total number of cell phone users.

Keywords: SMS, non-invasive, principle component analysis, eigen face.

1 Introduction

Face Recognition has been an emerging technology for last fifteen years or more. The technique is easy to use, non-invasive as well as non-intrusive. As a result it found uses in passport authentications, time attendance systems, control access systems and more.

Despite of all the advantages, this technology has its own limitations. To build up a realistic face recognition system, training sets and a huge database of facial images and large processing power for parsing and other operations is required. This has been the primary obstacle in using this technique in mobile platforms. In this work we provided a cost-effective solution for using face recognition technology on cell phones for the purpose of social security.

1.1 Face Recognition in Cell Phones

The main purpose of this work is the implementation of recognition techniques of human faces in cell phones with a view of enhancing the social security. We have

N. Meghanathan et al. (Eds.): CCSIT 2012, Part III, LNICST 86, pp. 134–141, 2012.

selected Android OS as the mobile OS due to its adequate knowledgebase for developers. The inbuilt face detect class can detect faces efficiently from a given picture which acts as an aid to our face recognition system. A remote centralized database connected with a remote processing has been used to store the personal information related to faces. The processing unit accepts the detected face from the cell phone, processes and recognizes the face and give a relevant feed back to the sender.

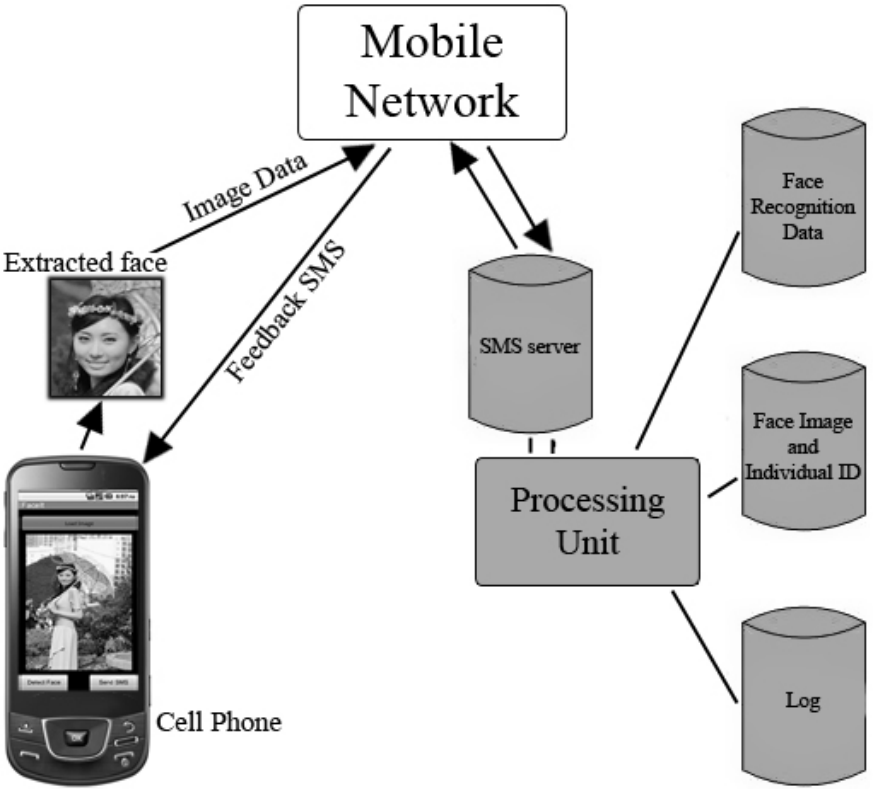

Fig. 1. Implementation of face recognition technology in cell phone

1.2 SMS Based Face Recognition Technology

This technique implies that any type of communication between the cell phone and the remote computer will be through sms/text message service. We have selected text messaging as a data application in this paper, due to its wide range user, its omnipresence and cost effectiveness. About 74% of all mobile phone subscribers use this service as a means to communicate. Text messaging service is available to almost any type of standard GSM network. Compared to any other means of communication

available in mobile network, this service is one of the cheapest and fastest ways to communicate. Our remote computer is connected with a sms server to communicate with the mobile phone.

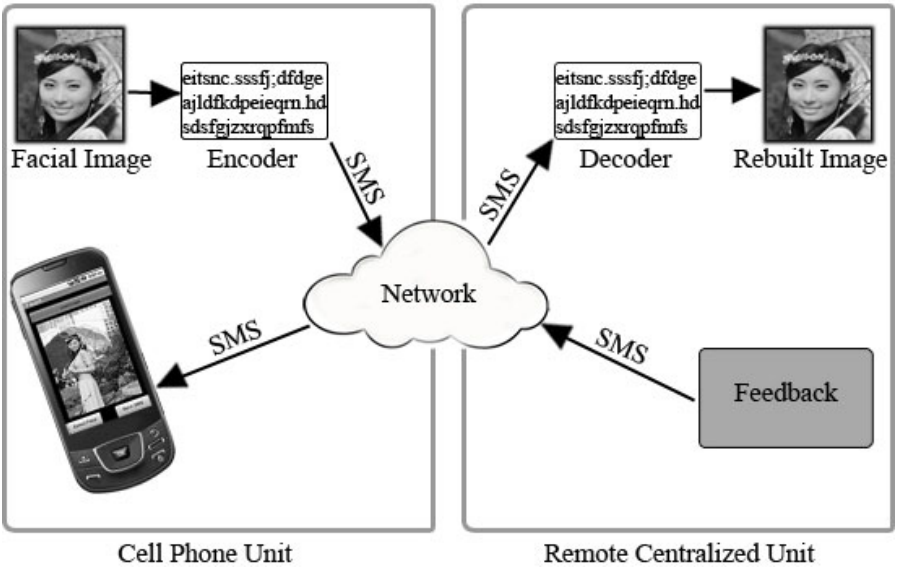

Fig. 2. Communication via text messages

2 Implementation of Face Recognition Technique

The face recognition system in this paper is a modular system. It consists of four primary modules.

2.1 Face Detect

This is the only module installed on the mobile platform. This module processes an image taken by a cell phone camera, to detect a face in the photograph. If a face is detected then the facial image is cropped and resized to 50X50 pixels size for optimized storage.

Fig. 3. Cropped facial images of size 50X50 pixels

The image is then compressed and encoded to a text message and is sent to the remote computer which is connected with a centralized database.

Most of the popular mobile OS has inbuilt Face Detect class/function. We have used the Android's Face Detect class for this module.

2.2 Image Processing

This module on the remote computer, decodes the sms received from the sms server, and rebuilds the facial image. The module adjusts the contrast and brightness of the image to a threshold value for better extraction.

2.3 Face Extraction

OpenCV HaarCascade method is implemented to extract face from the image received from the previous image. A classifier is loaded in this method which scans the image for any region which is likely to be face. The classifier returns "1" if face is found, or else it returns "0".

Next step performs the PCA algorithm on a training set, which has been created from the facial images in the database. Principal Component Analysis algorithm analyses data by performing dimensionality reduction where the projection of the original image onto a lower dimensional space takes place. The PCA subspace is calculated and the training images are converted to points in the subspace. All these data are then stored.

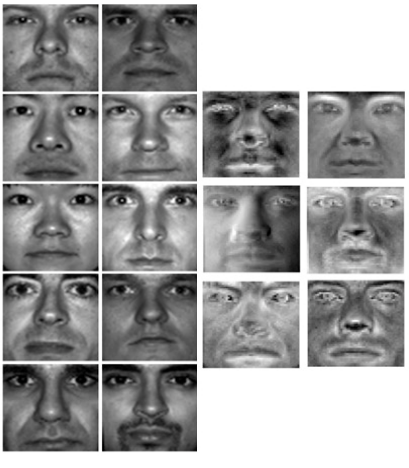

Fig. 4. Face extraction by Eigen's face method using training sets

2.4 Recognition Implementing Eigen's Method

This module loads all the images from the database and projects each image onto the PCA subspace. Then the closest projected training image is located. If the closeness is beyond a threshold value, then the result is taken as a positive match, and a relevant feedback text message is sent to the sender of the image.

3 Image to Text Conversion

In this paper, the two dimensional matrix consisting of the average RGB value of each pixel has been converted to a string array which goes through a proposed lossless string compression algorithm for the sake of reducing the output string size to 160 characters so as to accommodate in a single text message. This will avoid broken sms error, thus ensures data integrity while transmitting. Our proposed data compression algorithm is a combination of Huffman coding and Base 64 coding.

3.1 Encoding

This technique is carried out in the cell phone. Huffman algorithm is first applied to the string array. A binary Huffman tree is built. Then the output string is encoded with Base 64 and the Huffman tree is appended with the output separated by a separator (a special character we used for separating the string from the Huffman tree). Then a character counter checks whether the whole string is under 160 characters. If the check returns negative, then the string is passed to the Huffman function again. This method will iterate until a string with 160 characters or less is produced. The number of iterations will be appended in the end before sending the final string to the remote computer through a sms server.

3.2 Decoding

This algorithm is implemented on the remote computer to rebuild the image from the received text message. The first step is to separate the Huffman tree from the string. Then Base 64 decoding is performed and subsequently Huffman decoding is performed by the separated Huffman tree. In this way after the same number of iterations, the two dimensional pixel matrix can be obtained from which the image can be rebuilt.

4 Testing

We performed black box and white box testing on a database of face image. The efficiency of the recognition is quiet high for frontal faces which decreases gradually on rotating the face.

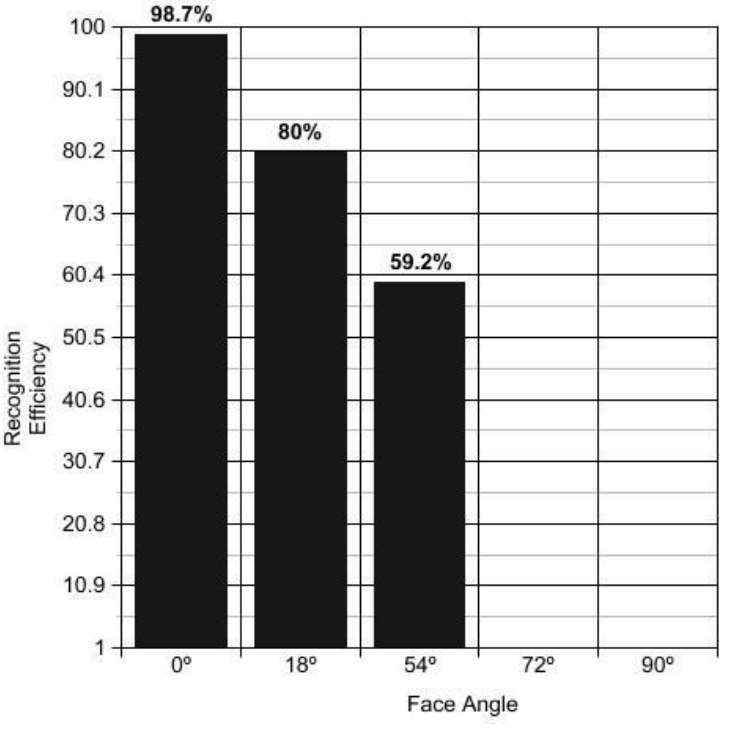

Fig. 5. Face reconition efficiency (under balanced illumination and noiseless background)

The formula of sms compression efficiency is

$$\text{Efficiency} = \frac{(\text{Total char in original SMS} - \text{Total char in compressed SMS})}{\text{Total char in original SMS}}$$

The image data we used for compression is as in Fig: 6.

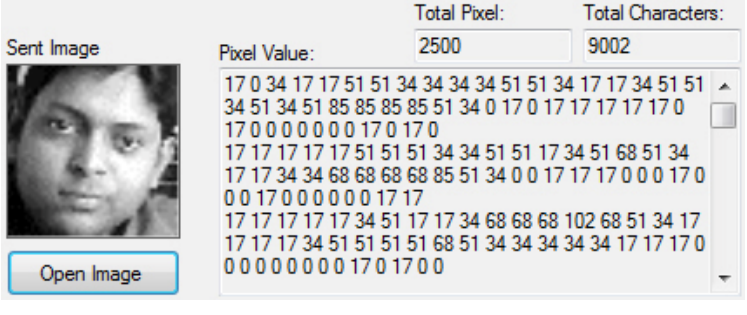

Fig. 6. Extracted pixel value

And the compressed text string with appended header and footer is depicted below.

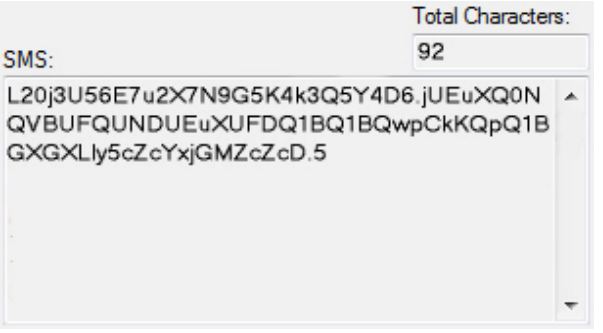

Fig. 7. Output text string

Thus efficiency for the above 50X50 pixel image is.

$$(9002-92)/9002 = 98.97\% \quad (1)$$

Efficiency for this algorithm is directly proportional to the length of the input string since the number of strings in output of Base 64 will always be within 64 characters. Thus though the compression algorithm is not much efficient for shorter string but for longer strings the algorithm yields more efficiency.

5 Conclusion

In this paper, we have proposed a way in which an individual with a camera phone can easily know the identity of a suspected subject instantly and can help enhance law enforcement. The system extracts the facial image from a photograph and sends it to a remote computer via a sms server. The remote computer then uses the recognition technique to match the face with the database and gives a feedback to the source mobile number. The result of our preliminary experiment shows improved performance in the face detection and the sms compression to the traditionally implemented systems. Current work is focused on the face detection algorithms on mobile platform and the sms compression algorithm. The iterations in the sms compression algorithm add a delay in the process. We propose a threshold value to limit the number of iteration to certain point.

In further work, we intend to improve face recognition effectiveness by implementing other algorithms. On the other hand, the proposed sms compression algorithm can be used in a completely new dimension of text messaging and sending multimedia messages via text messages.

References

1. Huffman, D.A.: A Method for the Construction of Minimum-Redundancy Codes. In: Proceedings of the I.R.E., pp. 1098–1102 (September 1952) Huffman's original article
2. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 2nd edn., Section 16.3, pp. 385–392. MIT Press and McGraw-Hill (2001)
3. Pissarenko, D.: Eigenface-based facial recognition (2003)
4. Acharya, T., Ray, A.: Image Processing: Principles and Applications. Wiley (2005)
5. Nelson, M., Jean, L.G.: The Data Compression Book, 2nd edn. IDG Books Worldwide, Inc., Cambridge (1995)
6. Hashemian, R.: Direct Huffman Code and Decoding Using The Table of Code-Lengths. Northen Illionis University (2005) (unpublished thesis)
7. Bradski, G., Kaehler, A.: O'Reilly Learning OpenCV, 1st edn. (2008)