Face Recognition for Security in Pawn Shops

Santhosh Kumar[1] R. Sandeep[2] J. Subiksha[3] B. Vishnu Priyen [4] M.Yamini[5]
[1,]Assistant Professor , [2345] UG Scholar, Department of ECE
SNS College of Technology
Coimbatore, India

Abstract : In the recent scenario, biometric technique is used in many fields such as mobile phone unlocking systems, medical centres. In this project we have proposed a face recognition for security in pawn shops. Usually a biometric system such as fingerprint scanning along with the gates are used for safety lockers of pawn shops. A face recognition algorithm is a computer based application which is capable of comparing the face of a authorized person with the image of a person that is already fed in the database. The locker system opens only if the face images are matched. This new method of using face detection technique will be useful for a more secure system by reducing thefts to a greater extent and also it helps to reduce the time taken by reducing.

I. INTRODUCTION

A smart city is an urban area that uses different types of electronic data collection or sensors to supply information used to manage assets and resources efficiently. This includes data collected from citizens, devices and assets that is processed and analyzed to monitor and manage traffic and transportation systems, power plants, water supply networks, waste management, law enforcement, information systems, schools, libraries, hospitals, and other community services. The smart city concept integrates ICT, and various physical devices connected to the network (IOT) to optimize the efficiency of city operations and services and connect to citizens. Smart city technology allows city officials to interact directly with both community and city infrastructure and to monitor what is happening in the city and how the city is evolving.

Some face recognition algorithms identify facial features by extracting landmarks, or features, from an image of the subject's face. For example, an algorithm may analyze the relative position, size, and/or shape of the eyes, nose, cheekbones, and jaw. These features are then used to search for other images with matching features. Other algorithms normalize a gallery of face images and then compress the face data, only saving the data in the image that is useful for face recognition. A probe image is then compared with the face data. One of the earliest successful systems is based on template matching techniques applied to a set of salient facial features, providing a sort of compressed face representation.

Recognition algorithms can be divided into two main approaches, geometric, which looks at distinguishing features, or photometric, which is a statistical approach that distills an image into values and compares the values with templates to eliminate variances. Popular recognition algorithms include principal component analysis using eigenfaces, linear discriminant analysis, elastic bunch graph matching using the Fisher face algorithm, the hidden Markov model, the multilinear subspace learning using tensor representation, and the neuronal motivated dynamic link matching.

II. EXISTING SYSTEM

The facial expression recognition system applies a band let system to a face image to extract sub-bands as in figure 1. Then, a weighted center symmetric local binary pattern is applied to each sub-band block by block. The CS-LBP histograms of the blocks are concatenated to produce a feature vector of the face image. An optional feature selection technique selects the most dominant feature which are then fed into two classifiers: Gaussian mixture model and support vector machine. The score of these classifiers are fused by weight to produce a confident score, which is used to make decisions about the facial expression's type. Several experiments are performed using a large set of data to validate the proposed system. Experimental results show that the proposed system can recognize facial expressions with 99.95% accuracy.

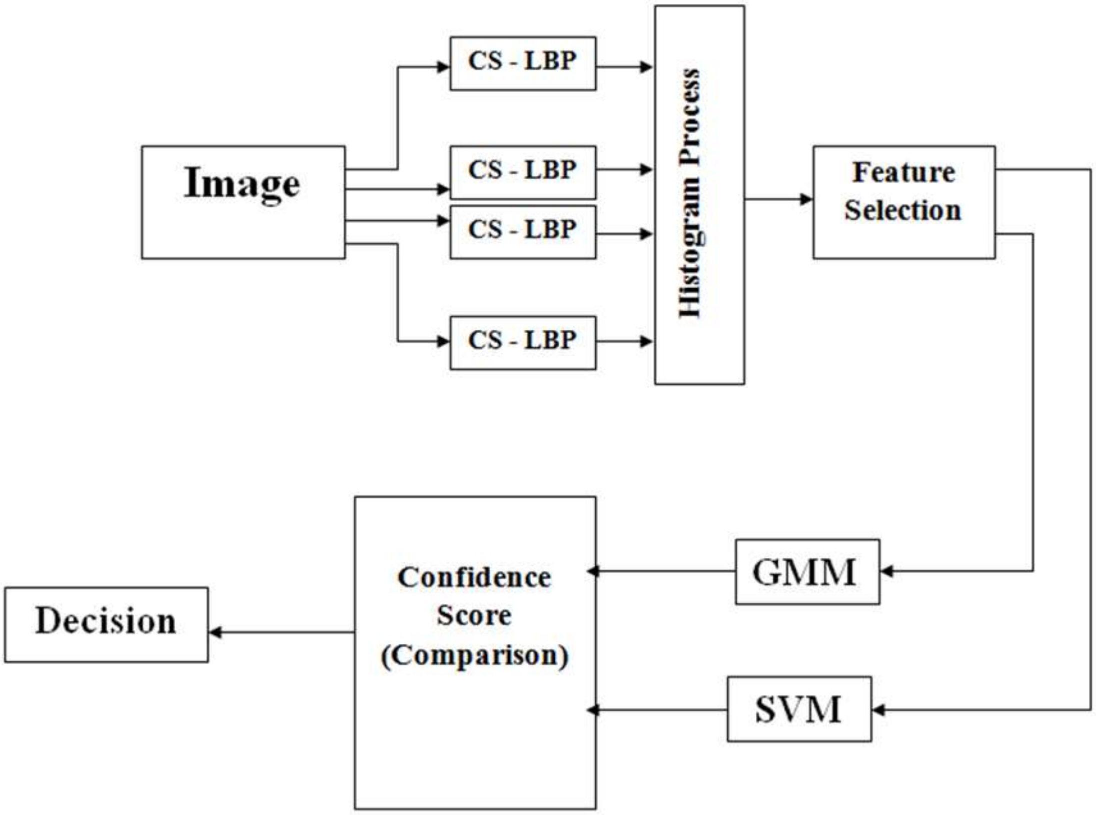


Figure 1 block diagram of facial expression recognition system

Figure 2 shows a general healthcare framework in a Smart City. There are many smart homes in the city, each equipped with various required smart devices, such as smart cameras, smart appliances, smart video, smart phones, smart alarm systems, smart switches, smart locks, and so on. In the figure, we see the flow of data, decisions, and actions in healthcare in the Smart City. The sensors capture signals or data from a resident in the smart home. These signals are transferred to the cloud for processing. A cloud manager handles authentication and access issues, while a cloud server processes the signal and makes a decision. The decision is then passed to certain registered hospitals, doctors, and caregivers. The final decision comes from the doctor, who then alerts the caregivers, traffic managers, and hospitals to take appropriate actions.

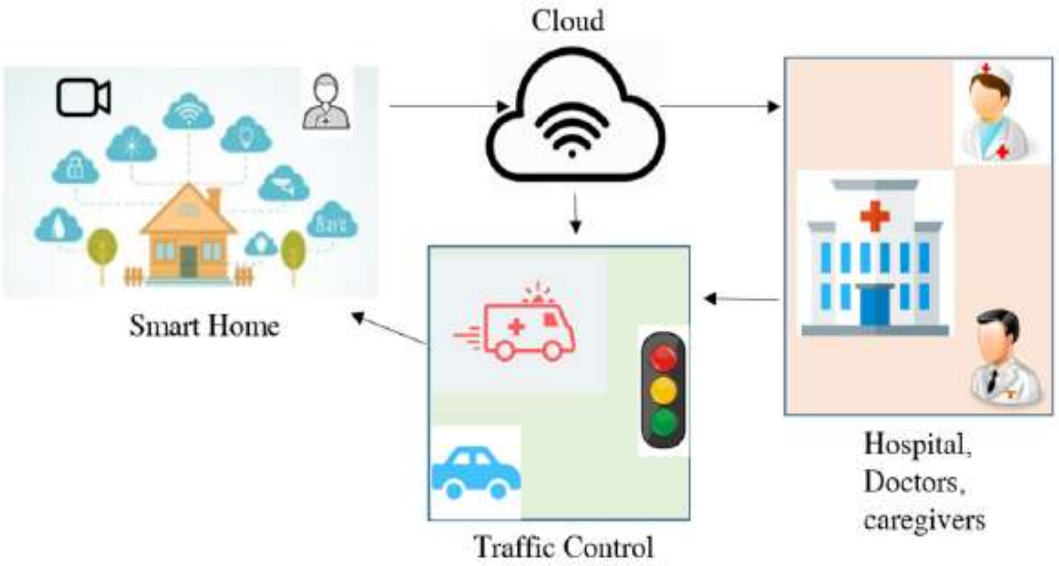


Figure 2 framework of smart city for healthcare

People often need to borrow small amounts of money that other lenders are not willing to provide. For this need the people sometimes prefer pawn shops for small loan amounts. The basic working of the pawn shop to secure the item is to keep it in locker. For that there are two gates before the locker with fingerprint. Whenever the authorized person enters he/she has to keep his/her finger in the fingerprint scanner for security purpose and then open the gates with the key. This procedure repeats for two times as there are two gates present before the locker. Then the pawned item is kept/taken from the locker.

III. Proposed System

In telecommunication and computer science, serial communication is the process of sending data one bit at one time, sequentially, over a communication channel or computer bus as shown in figure 3 In order to make two devices communicate, whether they are desktop computers, microcontrollers, or any other form of integrated circuit, we need a method of communication and an agreed-upon language. The most common form of communication between electronic devices is serial communication. Communicating serially involves sending a series of digital pulses back and forth between devices at a mutually agreed-upon rate. The sender sends pulses representing the data to be sent at the agreed-upon data rate, and the receiver listens for pulses at that same rate. This is what's known as asynchronous serial communication. There isn't one common clock in asynchronous serial communication; instead, both devices have their own clock and agree on a rate to which to set their clocks.

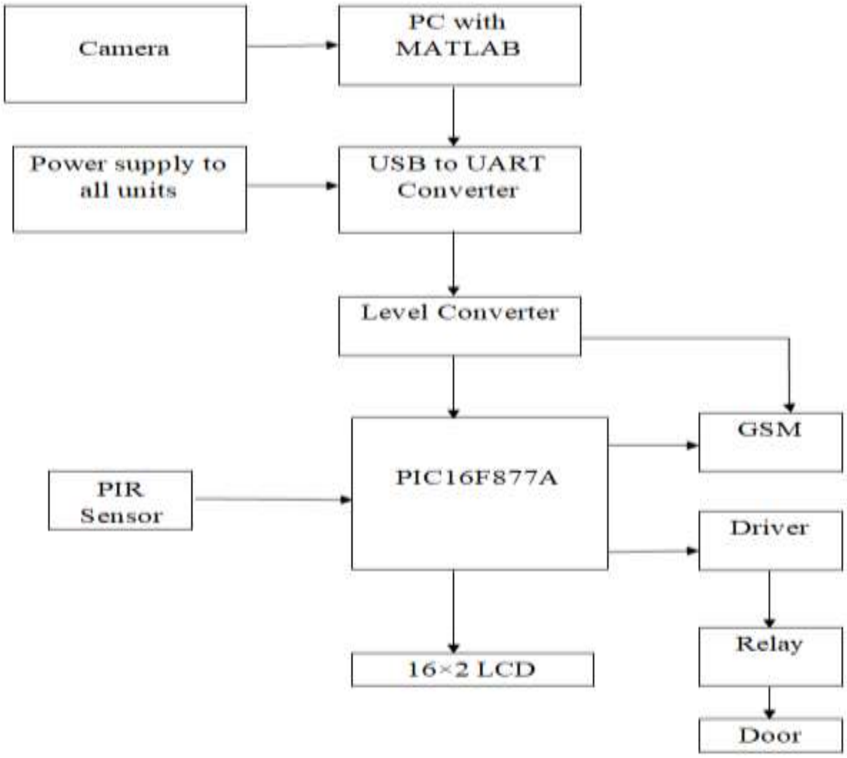


Figure 3 block diagram of proposed system

For example, let's say two devices are to exchange data at a rate of 9600 bits per second. First, we would make three connections between the two devices:

- a common ground connection, so both devices have a common reference point to measure voltage by;
- one wire for the sender to send data to the receiver on (transmit line for the sender);
- One wire for the receiver to send date to the sender on (receive line for the sender).

Now, since the data rate is 9600 bits per second (sometimes called 9600 baud), the receiver will continually read the voltage that the sender is putting out, and every 1/9600th of a second, and it will interpret that voltage as a new bit of data. If the voltage is high (+5V in the case of Wiring/Arduino, the PIC, and BX-24), it will interpret that bit of data as a 1. If it is low (0V in the case of Wiring/Arduino, the PIC, and BX-24), it will interpret that bit of data as a 0.

By interpreting several bits of data over time, the receiver can get a detailed message from the sender at 9600 baud, for example, 1200 bytes of data can be exchanged in one second. If you have a home computer and a modem, you've seen serial communication in action. Your computer's modem exchanges information with your service provider's modem serially.

IV. Software required

The software used in our project is MATLAB, Matrix Laboratory. It is a multi paradigm numerical computing environment. A proprietary programming language developed by MathWorks, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, C#, Java, Fortran and Python.

Although MATLAB is intended primarily for numerical computing, an optional toolbox uses the MuPAD symbolic engine, allowing access to symbolic computing abilities. An additional package, Simulink, adds graphical multi-domain simulation and model based design for dynamic and embedded systems.

MATLAB can call functions and subroutines written in the programming languages C or Fortran. A wrapper function is created allowing MATLAB data types to be passed and returned. The dynamically loadable object files created by compiling such functions are termed "MEX-files" (for MATLAB executable). Since 2014 increasing two-way interfacing with Python is being added.

Libraries written in Perl, Java, ActiveX or .NET can be directly called from MATLAB, and many MATLAB libraries (for example XML or SQL support) are implemented as wrappers around Java or ActiveX libraries. Calling MATLAB from Java is more complicated, but can be done with a MATLAB toolbox which is sold separately by MathWorks, or using an undocumented mechanism called JMI (Java-to-MATLAB Interface), (which should not be confused with the unrelated Java Metadata Interface that is also called JMI). Official MATLAB API for Java was added in 2016.

As alternatives to the MuPAD based Symbolic Math Toolbox available from MathWorks, MATLAB can be connected to Maple or Mathematica.

Libraries also exist to import and export MathML.

V. HARDWARE REQUIRED CONVERTER

The converter used here is serial RS232 converter. For the data transmission above, a high voltage indicates a bit value of 1, and a low voltage indicates a voltage of 0. This is known as true logic. Many serial protocols use inverted logic, meaning that a high voltage indicates logic 0, and a low voltage indicates logic 1. It's important to know whether your protocol is true or inverted. For example, RS-232, described below, uses inverted logic.

The RS-232 standard defines voltages and general baud rate ranges for serial communications between devices using it. We won't be getting the voltages exactly right, but for most applications, we'll be close enough. Until recently, most desktop computers had an RS-232 or similar serial port. Now, many desktop computers are shifting to other forms of serial communication such as USB, or Universal Serial Bus, and Firewire, which allow for more flexible configurations and faster data rates. the RS-232 standard is still very common in other devices, though, as it is cheaper to use than USB, simpler to implement, consumes less power, and provides more than adequate speeds for exchanging control data (i.e. data that allows one device to control another).UART (Universal Asynchronous Receiver Transmitter) or USART (Universal Synchronous Asynchronous Receiver Transmitter) are one of the basic interfaces which you will find in almost all the controllers available in the market till date. This interface provides a cost effective simple and reliable **communication** between one controller to another controller or between a controller and PC.

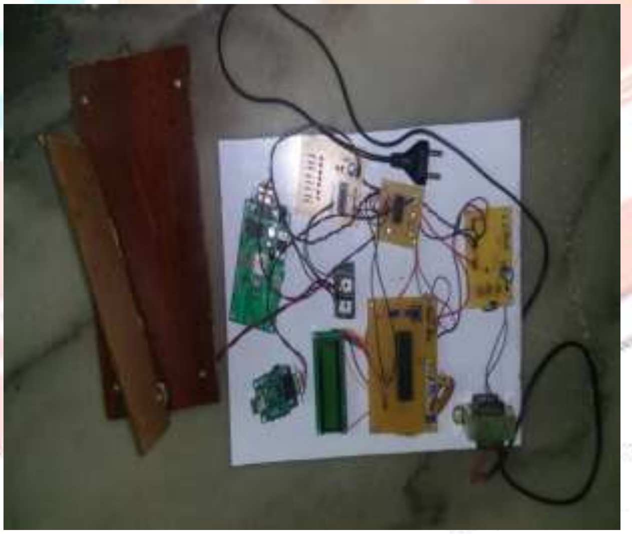


Figure 4 RS232 converter

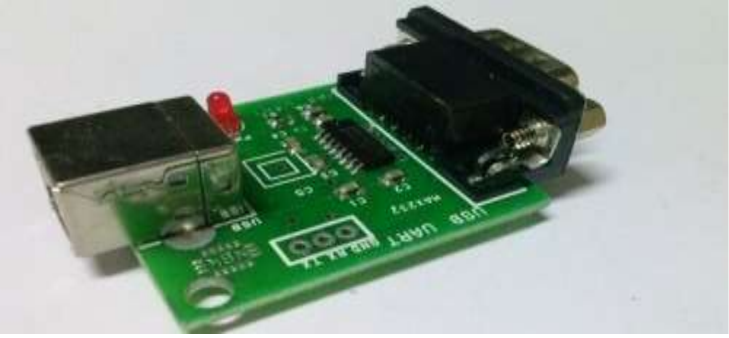
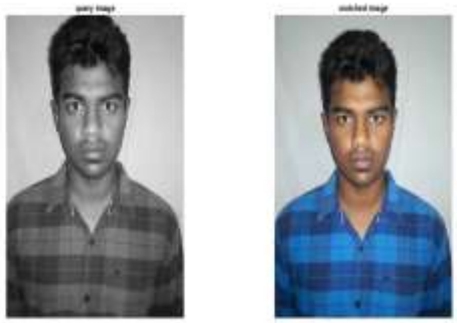


Figure 5 Hardware

VI. Results



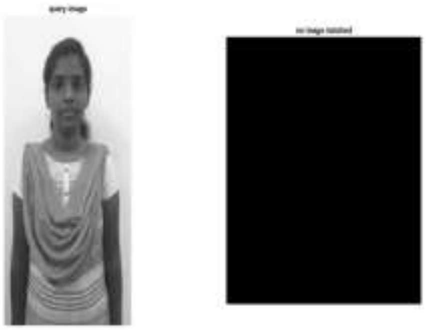


Figure 5 matched image

Figure 6 mis-matched image

VII. Conclusion

Thus using the classifiers the images are compared and then it is used for security purpose in pawn shops.

VIII. Future Work

This system, for the mismatched image the message is sent to the mobile and this may take some time or else if network has issues it would be not effective to have this idea. So if there is a mismatched image it would be effective to have a alarm sound. With this idea it can be implemented in many real time applications for secured network.

References

[1] H. Qayyum, M. Majid, S. M. Anwar, and B. Khan, ``Facial expression recognition using stationary wavelet transform features,'' Math. Problems Eng., vol. 2017, 2017, Art. no. 9854050.

[2] L. Y. Mano et al., ``Exploiting IoT technologies for enhancing health smart homes through patient identi_cation and emotion recognition,'' Comput.Commun., vols. 89_90, pp. 178_190, Sep. 2016.

[3] H.Muthusamy, K. Polat, and S. Yaacob, ``Improved emotion recognitionusing Gaussian mixture model and extreme learning machine in speech and glottal signals,'' Math. Problems Eng., vol. 2015, 2015, Art. no. 394083.

[4] Q. Mao, M. Dong, Z. Huang, and Y. Zhan, ``Learning salient features for speech emotion recognition using convolutional neural networks,'' IEEE Trans. Multimedia, vol. 16, no. 8, pp. 2203_2213, Dec. 2014.

[5] A. Zanella, ``Internet of Things for smart cities,'' IEEE Internet Things J.,vol. 1, no. 1, pp. 22_32, Feb. 2014.

[6] S. Mallat and G. Peyré, ``A review of bandlet methods for geometricalimage representation,'' Numer. Algorithms, vol. 44, no. 3, pp. 205_234,2007.

.

[7] A.Karthikeyan, S.Sai Gokul, P.Shalini, R.Sowmeya. R.Vinu Varsha, "An Awarding Point Technique in Wi-Fi Sharing System" in International Journal of Creative Research Thoughts (IJCRT) Volume 6, Issue 1, January 2018, pp 1267- 1273 ISSN: 2320-2882.

[8] A.Karthikeyan, S.Vigneshwaran, A.Santhosh kumar, S.Srikanth "Energy saving system using embedded system for street light controller" in International Journal of Advance Research in Science and Engineering (IJARSE), Volume:6, Issue:1, January 2017,pp 193-199 , ISSN:2319-8354.

[9] A. Karthikeyan, Merlin Asha. M, G. Naveen Balaji, S. Mythili, N. Thillaiarasu "An Efficient Brain Tumor Detection Algorithm based on Segmentation for MRI System" in International Journal of Trend in Scientific Research and Development (IJTSRD), Volume:2 , Issue : 2, January –February 2018, pp 1353- 1358,ISSN:2456-6470.

[10] A.Karthikeyan, P.Saranya, N.Jayashree "An Efficient VLSI Architecture For 3D DWT using Lifting Scheme" in International Journal of Engineering Science and Innovative Technology (IJESIT) Volume 2, Issue 1, January 2013, pp 292-298 ISSN : 2319-5967.