

## **FINGER VEIN RECOGNITION BASED ON DEEP NEURAL NETWORKS**

Dr. Sudhakar R<sup>1</sup>, Thilagavathi S<sup>2</sup>, Abirami P<sup>3</sup>, Vigneshwaran M<sup>3</sup>, Kavini Sundari R<sup>3</sup>

<sup>1</sup>Head of Department, Department of ECE, Dr. Mahalingam college of Engineering and Technology, Tamil Nadu, India

<sup>2</sup>Assistant Professor, Department of ECE, Dr. Mahalingam college of Engineering and Technology, Tamil Nadu, India

<sup>3</sup>Students, Department of ECE, Dr. Mahalingam college of Engineering and Technology, Tamil Nadu, India

[abirami02032001@gmail.com](mailto:abirami02032001@gmail.com), [thilagavathi.sta@gmail.com](mailto:thilagavathi.sta@gmail.com)

**Abstract**— *In this paper finger vein recognition is achieved with more accuracy in comparison with other biometrics. The accuracy and working of other biometric systems like finger print, iris may be affected by the skin condition of people. The other potential risk in biometrics includes tracking the data from the public with or without their knowledge may also occur. To solve this kind of issue a method of finger vein recognition based on deep neural networks is proposed. This technique is used to analyze the patterns of blood vessels which are visible from the surface of the skin of fingers. The system can also provide the enhanced security from fraudulent activities by identifying the blood flow in the veins to ensure the person is alive or not. The main purpose of this project is to propose a method of finger-vein identification system based on Convolutional neural networks(CNN) which deals with finger-vein images of different quality. This system is used to produce stable and highly accurate performance and also improved authentication speed. The input image is compared with the images in the database. If it is matched with the existing image, the system will give the positive response for authentication otherwise vice versa . This system is used for Biometric Authentication at Hotels, Security purpose in Schools, Colleges and also secured unlocking in Mobile phones.*

**Keywords**—*component; finger vein recognition; convolution neural network; image feature extraction*

### I. Introduction

Finger-vein recognition has attracted the attention of the biometrics community [1],[3] in recent years. Finger vein recognition is one of the method of biometric authentication that uses pattern-recognition techniques it make use of images of human finger vein patterns which are present under the skin surface. Several biometric modalities have been employed for the verification task, these can be broadly classified into two categories: (i) Extrinsic modalities, i.e. face, fingerprint, and iris (ii) Intrinsic modalities i.e. finger vein, hand-vein, and palm-vein. A finger vein has some distinct advantages when compared to other conventional biometric traits like face, palm print and iris [5], [7]. It is proposed to adopt a CNN model pre-trained on ImageNet as a universal feature descriptor. This approach represents a hand vein image by concatenating the local features from different levels of the spatial pyramid and also proposed a hand-dorsal veinrecognition system employing VGG-16 and it is pre-

trained on a large-scale database, as a universal feature extractor.

Nowadays biometric identity theft is increased such as artificial fingerprints are made with gelatin, Silicone molds which may be misused by the unknown user to clone user(clone fingerprints) for the accessing purpose. To avoid this finger vein recognition using deep neural networks [9] is employed. Deep convolutional Neural Networks (CNNs) have demonstrated excellent performance for image classification. Conventional finger vein recognition systems perform recognition based on the finger lines extracted from the input Images or image enhancement and texture feature extraction from the finger vein images [11]. It employed [12] an improved deep network, merge CNN which uses several CNNs with short paths. Finger-vein recognition [13], a composite image generated using only a texture image was used as input in the CNN. CNN's were presented in [14] and achieved good performances on different finger-vein datasets.

Mostly, CNNs are used for feature extraction, and [15] for finger-vein image quality assessment. A CNN model pre-trained on ImageNet is adopted [16] to develop a CNN-based local descriptor named CNN-CO (CNN competitive order) that can exploit discriminative features for finger vein recognition. The anatomy and imaging analysis of the finger vein is explained [17]. A novel trainable Generalized-Mean (GeM) pooling layer that generalizes max and average pooling is demonstrated and show that it boosts retrieval performance [18]. Reduces missing pixels during extraction [19], and the method of sliding window summation is employed to filter and adjusted with self-built tools. The need for Finger vein recognition offers a form of identification that is much more accurate than older identity verifications technologies like digital fingerprinting. A residual learning framework for the training of networks that are substantially deeper than those used previously [20] so deep learning-based approaches have been successfully applied for computer vision and object tracking, speech recognition, and handwriting recognition. [2] proposed a secure hash message authentication code. A secure hash message authentication code to avoid certificate revocation list checking is proposed for vehicular ad hoc networks (VANETs).

Convolutional Neural Networks are proposed to reduce the number of parameters and adapt the network architecture specifically to vision tasks. Convolutional neural networks are usually composed by a set of layers that can be grouped by their functionalities. Using Finger vein recognition based on deep neural networks a system is created as it is used

for authentication process and achieved by matching the vein patterns if it.

II. DEVELOPMENT OF FINGER VEIN RECOGNITION SYSTEM

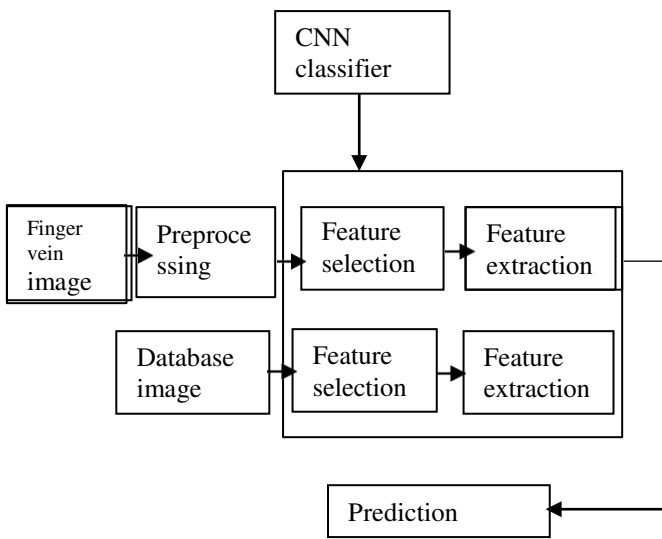


Figure 1. Block diagram of finger vein recognition system

The finger vein Recognition system is a method that specifies an individual’s vein pattern. Finger vein images are stored and first undergoes preprocessing method. After that CNN classifier prognosis the feature selection and feature extraction of finger vein image and data base image. After extracting features CNN comparison of both image prediction is made, if the sample images match with the input image it is predicted as authorized, if it does not match with sample images it is predicted as unauthorized.

A. Pre-processing

Pre-processing is the process of reducing the redundancy present in the captured images without reducing the details present in them. There are three steps in pre-processing,

- Reading the input image (Greyscale Image)
- Resizing an image
- Denoise (Removing noise)

a) Reading the input Image:



Figure 2. reading the input image

By reading the input image it provides an easy interface to know the absolute path where the images are stored.

b) Resizing an image:

Resizing an image is increasing or decreasing the total number of pixels and remapping can occur when corrected for lens distortion or rotating an image. There are 2 ways to resize image data to match the input size. Using the Scaling factor the height and width of the images are multiplied. It changes the spatial extents of the pixels to the aspect ratio.

Need for resizing: Basically, the pixels in an image is updated when it is resized. While processing input images it makes some unnecessary pixels. By resizing we can establish a base size for all images (360 widths, 480 heights, 3 pixels).

b) c) Removing noise:

A noise removal algorithm is a process of removing or reducing the noise from the image. The noise removal algorithm removes the visibility of noise by smoothing the entire image. Removing the noise also helps in preserving the details of the same image. Advantages of removing noise are more robust against noise and preserve fine structure to some extent, it also reduces noise in flat regions and preserves edges to a higher extent.

B. Convolution Neural Network

A CNN is a type of artificial neural network used in image recognition and processing the Pixel data. The Feature Selection and Feature Extraction are done by different layers of CNN which is shown in Fig 3.

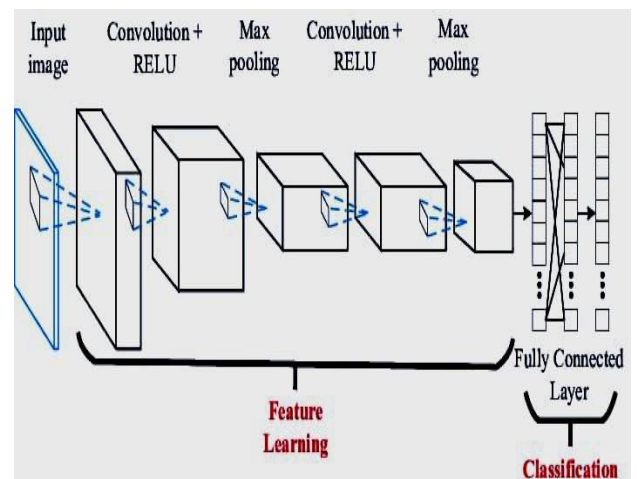


Figure 3. CNN Architecture

a) Convolution layer

The convolution layer is the key component of convolutional neural networks and this is the first layer. The purpose is to detect the presence of a set of features in the images received as input. This is done by convolution filtering. This layer receives an input image and also several

images and calculates the convolution of each of them with each filter. This filter corresponds exactly to the features we want to find in the images. [8] discussed because of various appealing focal points, agreeable correspondences have been broadly viewed as one of the promising systems to enhance throughput and scope execution in remote interchanges. [10] discussed that Helpful correspondence is developing as a standout amongst the most encouraging procedures in remote systems by reason of giving spatial differing qualities pick up.

*b) Pooling layer*

The pooling layer is another block of CNN are used to reduce the dimension of images. It reduce the number of parameters and computation in network. And also basically helps in the reduction of overfitting by compressing or generalizing the features of images.

*c) ReLU correction layer*

ReLU (rectified linear units) defined by  $ReLU(x) = \max(0, x)$ . where, 0 mentioned for negative values and x mention the value greater than zero. The function is linear for values greater than zero, means it has lot of desirable properties of linear activation function when training a neural network and it is nonlinear function when the output is in negative value. visually it looks like as shown in Fig 4.

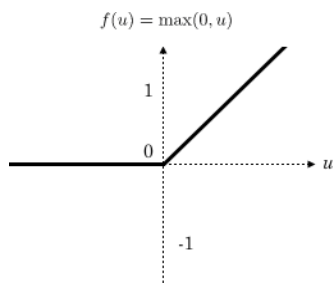


Figure 4. ReLU layer

The ReLU correction layer replaces all negative values received as inputs by zeros. It acts as an activation function.

*d) Fully connected layer*

The Fully connected layer receives an input vector and produces a new output vector. To achieve this, it applies a linear combination and then possibly an activation function to the input values received. And to calculate the probabilities, this layer multiplies each input element by weight, makes the sum, and then applies an activation function. The term fully connected explains the fact that each input value is connected with all output values. The CNN also learns weight values in the same way as it learns the convolution layer filters during the training phase by the backpropagation of the gradient. The relationship between the position of features in the image and a class is determined in this layer.

*C. Software Requirements*

In Arduino IDE, upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS, and Linux. Arduino IDE supports the programming languages such as C and C++. User-written code only requires two basic functions, starting the sketch and the main program loop, which is compiled to check if any errors are there and uploading the code into the Arduino IDE distribution. Python Programming language that aims for both new and experienced Programmers to be able to achieve their ideas. It is currently the most prevalent, and also well supported among the Programming languages. Computer Vision, allows the Computer to identify objects through digital images or videos. Implementing CV through Python supports developers to execute the idea easier. And also it has several advantages they are Ease of coding, Fast Prototyping, Vast Libraries for machine learning, it can be directly integrated with web frameworks.

Table 1: Function of Arduino pin

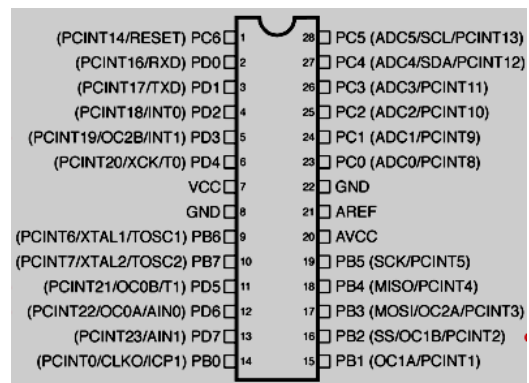


Figure 6. PIN diagram of Arduino UNO

S.NO	PIN NAME	FUNCTION
1	PC 6	reset
2	PD 0 – PD 4	Digital pin 0 –Digital pin 4
3	VCC	VCC
4	GND	Ground
5	PB 6 , PB 7	Crystal
6	PD 5 – PD 7	Digital pin 5 –Digital pin 7
7	PB 0 –PB 5	Digital pin 8 –Digital pin 13
8	AVCC	VCC
9	AREF	Analog reference
10	GND	Ground
11	PC 0 – PC 5	Analog input 0 – Analog input5

The table 1 describes about the name and function of Arduino pin.

Table 2: Features of Arduino

Micro controller	Atmega328p -8 bit AVR Family microcontroller.
Operating voltage	5v
Recommended input voltage	7-12v
Input voltage limits	6-20v
Analog input pins	6(A0-A5)
Digital I/O pins	14
DC current on I/O pins	40 mA
DC current on 3.3v pin	50 mA
Flash memory	32 KB
SRAM	1 KB

The table 2 describes about the features of Arduino UNO.

III. HARDWARE PARTS

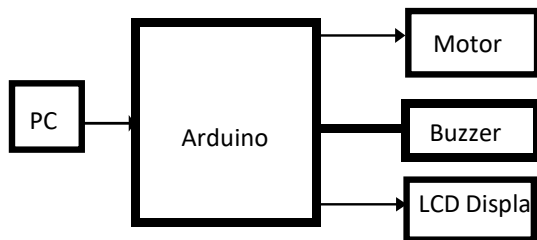


Figure7. Hardware block diagram

A. Arduino UNO

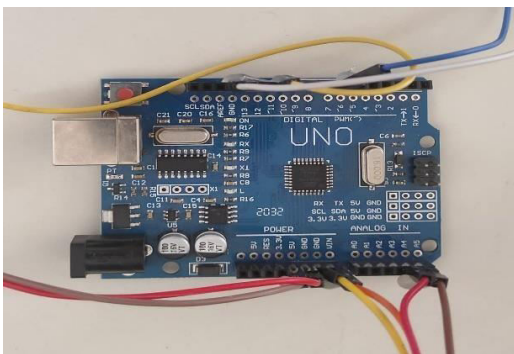


Figure 8.Arduino UNO

Arduino UNO is a microcontroller board and it is an 8bit ATmega328P microcontroller and also it consists of other components such as a crystal oscillator, serial communication, voltage regulator, etc. to support the microcontroller. It has 14 digital input/output pins (out of which 6 can be used as PWM outputs), 6 analog input pins, a USB connection, A Power barrel jack, an ICSP header, and a reset button.

B. Motor

The Fig 9 DC shows a motor which has some advantages such as higher starting torque, quick starting, stopping, reversing, variable speeds with voltage input also cheaper and easier to control than AC. A Relay is a switch that turns on or off based on an external electrical signal. And the connections are previously explained. With a positive response from PC, the motor will be in the ON condition.



Figure 9. DC Motor

C. Interfacing motor with Arduino microcontroller

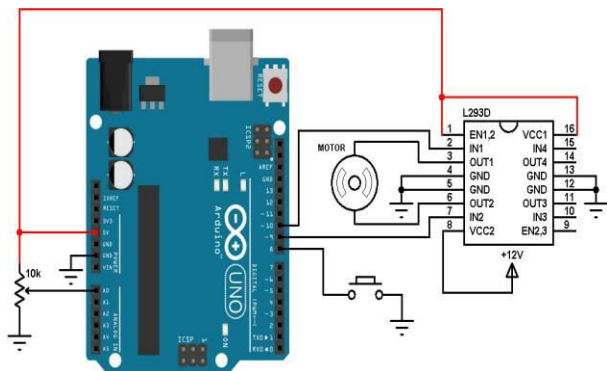


Figure10. Interfacing motor with Arduino microcontroller

The motor and relay are connected in the right way with a +12V power line to the Arduino board according to the arrangement shown in Fig 10. This shows the way to control the main voltage with the Arduino using a relay module. Generally closed configuration is used, meaning the current is flowing. By sending a signal from the Arduino to the relay part that opens the circuit and stop the current. An open configuration works in another way that relay is always open, by sending signal from the Arduino the circuit is closed. The right set consists of VCC and GND to power up the module, and input that control the bottom and top relays, respectively. The second set consists of GND, and VCC Pins. The VCC pin powers the electromagnet of the relay. The module has a VCC and pins connecting the jumper cap. The VCC and pins are connected with the jumper cap ON. That means the relay electromagnet is directly powered by the Arduino's power pin, so the relay module and the Arduino circuits are not physically isolated from each other. Without the jumper cap, we need to provide an independent power source to power up the relay's



electromagnet through the pin. Motor runs if the authentication process is successful.

*D. Buzzer*

The Fig 11 shows a Buzzer or beeper is an audio signaling device. The piezoelectric buzzer can produce a variety of pleasant sounds, analog sounds, and intermitted sounds. With a negative response from pc, the buzzer will be ON.



Figure 11. Buzzer

*E. Interfacing buzzer with Arduino microcontroller*

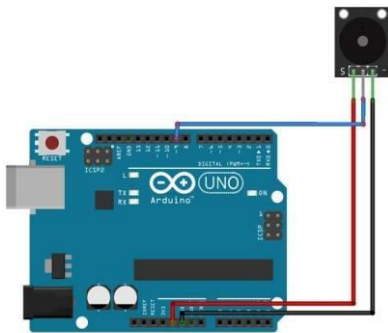


Figure 12. Interfacing buzzer with Arduino microcontroller

The buzzer works on the principle of the piezoelectric effect. The piezoelectric element is the main component of a buzzer. When an alternating current is applied to a piezoelectric disc it expands and contracts diametrically. As a result, vibrations are produced in the piezoelectric element. And it generates the sound of a particular frequency. In a magnetic buzzer, a pole consists of a ferromagnetic disc. The coil in the ferromagnetic disc acts as an electromagnet, and when current is supplied to the coil, the disc is attracted to the coil. If there is no current in the coil, the disc returns to its resting position. The vibrations of the disc are controlled by the weight of the disc. If the power supply is ON the current is generated in an oscillator. And then this current passes through the solenoid coil. Sound generates under the interaction of magnet and coil. Connect the Supply wire (positive) of the buzzer to the Digital Pin of the Arduino. Connect the Ground wire (negative) of the buzzer to Ground Pin on the Arduino. Sound is generated when there is a negative response in the authentication process.

*E. LCD display*

The Fig 13 shows Liquid crystal display has material that combines the properties of both liquid and crystals. They have a temperature range within which the molecules are almost as mobile as they would be in liquid, but are grouped in an order form similar to a crystal. More microcontroller devices are using smart LCDs to output visual information. LCD is lighter in weight for screen size. Energy-efficient because of lower consumption. The brightness range is too much wider to produce very bright images due to high peak intensity.

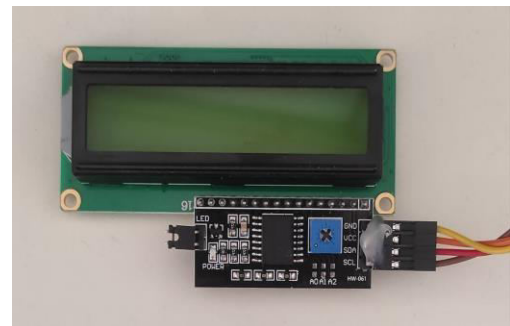


Figure 13. LCD display

*F. Interfacing LCD with Arduino microcontroller*

The circuit diagram shows the interfacing of liquid crystal display with the Arduino module. From the circuit diagram, it is observed that the RS pin of the LCD is connected to pin 12 of the Arduino. The Arduino's Pin 11 is connected to the enable signal pin of the LCD module. The LCD part and Arduino part are interfaced with the 4-bit mode in this project. Hence there are 4 input lines which are DB4 to DB7 of the LCD. This process is very simple, it requires fewer connection cables and also it can utilize the most potential of the LCD module. [4] proposed a novel method for secure transportation of railway systems has been proposed in this project. In existing methods, most of the methods are manual resulting in a lot of human errors. This project proposes a system which can be controlled automatically without any outside help. [6] discussed that the activity related status data will be communicated consistently and shared among drivers through VANETs keeping in mind the end goal to enhance driving security and solace. Along these lines, Vehicular specially appointed systems (VANETs) require safeguarding and secure information correspondences

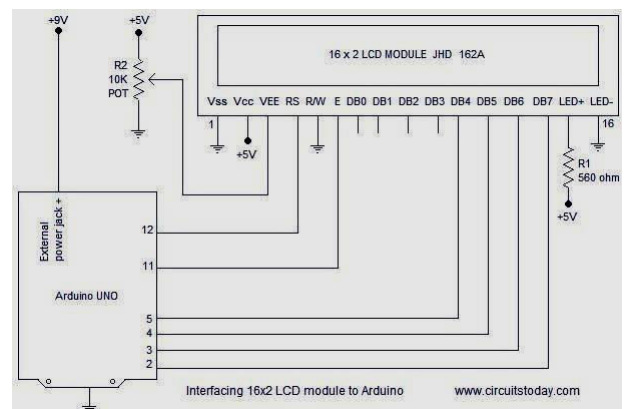


Figure 14. Interfacing LCD with Arduino microcontroller

## a) ALGORITHM

STEP 1: Start the program  
 STEP 2: Get necessary elements from LiquidCrystal\_I2C.h Header file  
 STEP 3: Here, if condition is used to find the finger vein matching  
 STEP 4: If condition true is satisfied, print authorised  
 STEP 5: When if condition fails , using else if condition is used to find mismatching finger vein  
 STEP 6: If condition true then print unauthorized and buzzer starts high to low sound  
 STEP 7: End of the program

```
// Set the LCD address to 0x27 for a 16 chars and 2 line display
LiquidCrystal_I2C lcd(0x27, 16, 2);
void setup(){
  Serial.begin(9600);
  pinMode(10,OUTPUT);
  pinMode(6,OUTPUT);
  lcd.begin();
  lcd.backlight(); }
void loop()
  if(Serial.available()){
    byte rcv = Serial.read();
    // Serial.println(rcv);
    // if rcv checks 'A'(matching finger vein)
    if(rcv == 'A'){
      lcd.clear();
      // then show authorized
      lcd.print("Authorized");
      // buzzer start at low
      digitalWrite(10,LOW);
      delay(3000);
      // buzzer end with high
      digitalWrite(10,HIGH);
      delay(1000);
    }
    // else if rcv 'B'(mismatching finger vein)
    else if(rcv == 'B'){
      // Serial.println("IN2");
      lcd.clear();
      // then show unauthroized
      lcd.print("Unauthorized");
      // buzzer start at high
      digitalWrite(6,HIGH);
      delay(2000);
      //buzzer end with low
      digitalWrite(6,LOW); }
  }
}
```

## IV. RESULTS AND DISCUSSION

The Fig 15 represents the hardware connections. The motor, relay, buzzer, and LCD Display are interfaced with the

Arduino microcontroller. The HW (Hi-Watt) battery is used to power the motor.

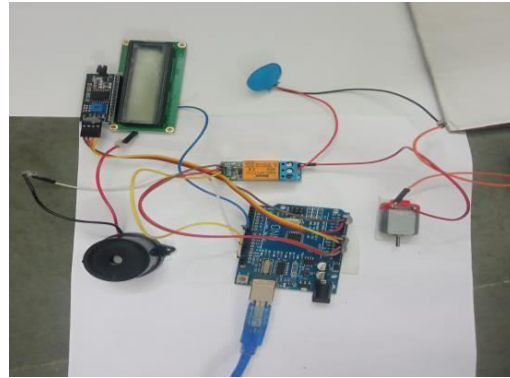


Figure 15: Hardware connection

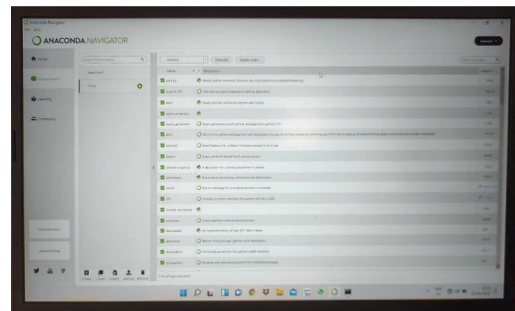


Figure 16. Anaconda Navigator

Using the Anaconda Navigator Tool, a terminal inflow the running part leads to the output window. After changing the directory the vein pattern folder is chosen which start running the output. By using Python f1.py, in which f1 includes all the sample vein patterns then the image browsing window will pop up as shown in Fig 17.



Figure17. Browsing result

The connection between the motor and Arduino is shown in Fig 18.

While browsing the image(FIG 17),if input image matches with database image,it will result as authenticated as shown in below FIG 19.

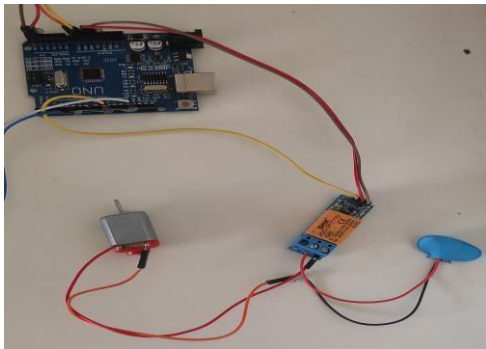


Figure 18: Arduino with DC motor

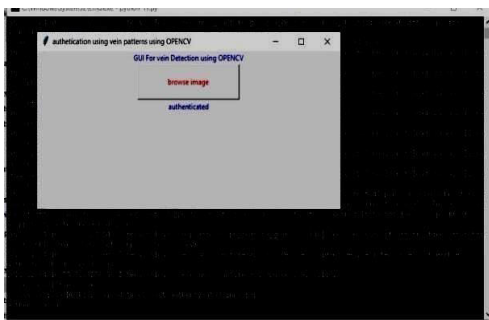


Figure 19. Authenticated result

Parallely in the hardware side, the authorised result will shown in the LCD Display and at the same time motor will run which is shown in FIG 20.

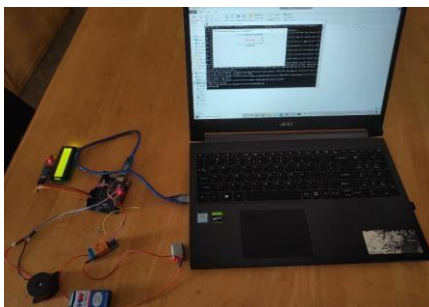


Figure 20. Authorized result

The connection between the Buzzer and Arduino is shown in Fig 21.

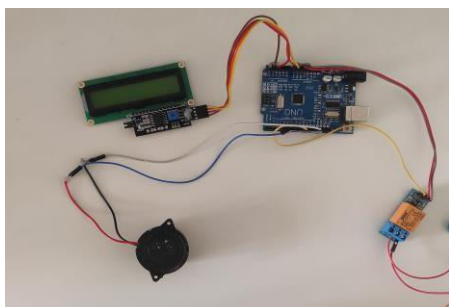


Figure 21. Arduino with buzzer

Similarly, while browsing the image(FIG 17) ,if input image does not matches with database image,it will result as unauthenticated as shown in below FIG 22.



Figure 22. Unauthenticated result

Parallely in the hardware side, the unauthorised result will shown in the LCD Display and at the same time buzzer will on which is shown in FIG 23.

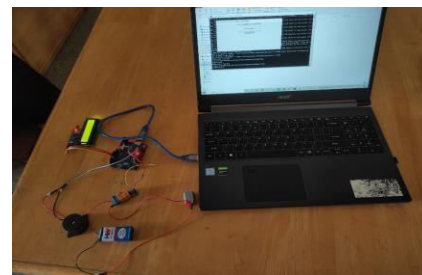


Figure 23. Unauthorized result in LCD

## V. CONCLUSION

In this paper, the finger vein recognition techniques can provide relatively good performance, they are strongly dependent upon the image quality of the analyzed finger-vein images. By using this finger vein recognition, we can achieve more accuracy in comparison with other biometrics like fingerprint, iris patterns or facial expression that can be used to identify an individual. And biometrics theft can be reduced by this finger-vein recognition. If the vein pattern matches it will result as authenticated otherwise unauthenticated. And the future scops are self-checkout without the need for cash or a card can be implemented by storing the public template for finger vein authentication using a near infrared camera on a tablet device attached to a shopping cart and one stop biometric authentication for hotels.

## VI. REFERENCES

- [1] W. Kang, Y. Lu, D. Li, and W. Jia, "From noise to feature: Exploiting intensity distribution as a novel soft biometric trait for finger vein recognition", IEEE Trans. Inf. Forensics Security, vol. 14, pp. 858-869, Apr. 2018.
- [2] Christo Ananth, M.Danya Priyadharshini, "A Secure Hash Message Authentication Code to avoid Certificate Revocation list Checking in Vehicular Adhoc networks", International Journal of Applied Engineering Research (IJAER), Volume 10, Special Issue 2, 2015,(1250-1254).

- [3] L. Yang, G. Yang, K. Wang, H. Liu, X. Xi, and Y. Yin, "Point grouping method for finger vein recognition", *IEEE Access*, vol. 7, pp. 28185-28195, 2019.
- [4] Christo Ananth, K.Nagarajan, Vinod Kumar.V., "A Smart Approach For Secure Control Of Railway Transportation Systems", *International Journal of Pure and Applied Mathematics*, Volume 117, Issue 15, 2017, (1215-1221).
- [5] L. Fei, B. Zhang, Y. Xu, W. Jia, J. Wen, and D. Huang, "Local discriminant direction binary pattern for palmprint representation and recognition", *IEEE Transactions on Circuits and Systems for Video Technology*, Volume: 30,2020.
- [6] Christo Ananth, Dr.S. Selvakani, K. Vasumathi, "An Efficient Privacy Preservation in Vehicular Communications Using EC-Based Chameleon Hashing", *Journal of Advanced Research in Dynamical and Control Systems*, 15-Special Issue, December 2017,pp: 787-792.
- [7] K. Nguyen, C. Fookes, A. Ross and S. Sridharan, "Iris recognition with off-the-shelf CNN features:A deep learning perspective", *IEEE Access*, vol. 6, pp. 18848-18855, 2018.
- [8] Christo Ananth, Dr. G. Arul Dalton, Dr.S.Selvakani, "An Efficient Cooperative Media Access Control Based Relay Node Selection In Wireless Networks", *International Journal of Pure and Applied Mathematics*, Volume 118, No. 5, 2018,(659-668).
- [9] Diaa M. Uliyan, Somayeh Sadeghi, Hamid A. Jalab, "Anti-spoofing method for fingerprint recognition using patch based deep learning machine" *ScienceDirect*, Jul 4, 2019.
- [10] Christo Ananth, Joy Winston.J., "SPLITTING ALGORITHM BASED RELAY NODE SELECTION IN WIRELESS NETWORKS", *Revista de la Facultad de Agronomía*, Volume 34, No. 1, 2018,(162-169). *Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1452–1464, Jun. 2018.
- [11] Kyong Jun Noh, Jiho Choi, Jin Seong Hong, and Kang Ryoung Park, "Finger-Vein Recognition Based on Densely Connected Convolutional Network Using Score-Level Fusion With Shape and Texture Images" *IEEE access*, Jun 4, 2020.
- [12] Ismail boucherit, Mohamed Ould Zmirli, Bakhtiar and Affendi Rosdi "Finger vein identification using deeply-fused Convolutional Neural Network" *Journal of King Saud University*, volume 34, 8 April 2020.
- [13] J. M. Song, W. Kim, and K. R. Park, "Finger-vein recognition based on deep Dense Net using composite image," *IEEE Access*, vol. 7, pp. 66845–66863, May 2019.
- [14] Y. Lu, S. Xie and S. Wu, "Exploring competitive features using deep convolutional neural network for finger vein recognition", *IEEE Access*, vol. 7, pp. 35113-35123, 2019.
- [15] H. Qin and M. A. El-Yacoubi, "Deep representation for finger-vein image quality assessment," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1677–1693, Aug. 2018.
- [16] Y. Lu, S. Xie and S. Wu, "Exploring competitive features using deep convolutional neural network for finger vein recognition", *IEEE Access*, vol. 7, pp. 35113-35123, 2019.
- [17] L. Yang, G. Yang, Y. Yin, and X. Xi, "Finger vein recognition with anatomy structure analysis", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 8, pp. 1892-1905, Aug. 2018.
- [18] F. Radenović, G. Toliás, and O. Chum, "Fine-tuning CNN image retrieval with no human annotation", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1655-1668, Jun. 2018.
- [19] Junying Zeng, Fan Wang, Jianxiang Deng, Chuanbo Qin, Yikui Zhai, Junying Gan, "Finger vein verification algorithm based on fully convolution neural network" *IEEE Access*, April 1, 2020.
- [20] R. Das, E. Piciucco, E. Maiorana and P. Campisi, "convolutional neural network for finger vein based biometrics identification", *IEEE Trans. Inf. Forensics Security*, vol.14 pp.360-373, 2019.