International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | Impact Factor: 7.488 |

|| Volume 8, Issue 5, May 2020 ||

IoT AWS Based Real-Time Eye Tracking using Gaze-Based Pin Entry for Password Authentication

Aishwarya N[1], Neha H[2], Neha P Patil[3], Pavithra P[4], Preethi Y R[5]

UG Student, Department of ISE, Sapthagiri College of Engineering, Bengaluru, Karnataka, India[1,2,3,4]

Assistant Professor, Department of ISE, Sapthagiri College of Engineering, Bengaluru, Karnataka, India[5]

ABSTRACT: Personal Identification Numbers play a vital role in our day-to-day life. PINs are used for user authentication and security purposes. Physically entering PINs may lead to vulnerabilities such as shoulder surfing and thermal tracking. The purpose of this project is password authentication using gaze-based pin entry, which avoids any kind of vulnerabilities. Gaze-based pin entry refers to finding eye location and tracking eye centre over time. Laptop camera is used as smart camera in our project which is used for data-processing. This project is a real-time application for gaze-based PIN entry by detecting eye for PIN identification. Applications include homeland security applications, ATM machine PIN entry, etc.

KEYWORDS: Gaze-based pin entry, Eye detection, Eye tracking, NIVision builder, LabVIEW, IoT, AWS

I. INTRODUCTION

The use of Personal Identification Numbers is a common user authentication method for many applications. PINs are used in ATM, approving electronic transactions, unlocking personal devices and opening doors. Perfect password authentication is still a challenging task in many financial applications. According to European ATM Security, ATM frauds are increased by 27% in 2018 when compared to 2017. The fact that the authorized user enters his PIN manually in public places, it may lead to shoulder surfing and thermal tracking, as the person standing close to that user may read his PIN very easily. To avoid such vulnerabilities, our project concentrates on gaze-based PIN entry for password authentication. This is achieved by real-time eye detection and tracking, using NI Vision Builder and LabVIEW techniques. Eye-detection algorithm is used for detecting eye location in a face through laptop camera. Eye centre is tracked and stored in the form of co-ordinates. This tracking of eye centre involves consideration of few important parameters such as different angles of the face, head motion, eye occlusions, distances between the user's eye and laptop camera. Noncontact PIN based authentication adds a layer of security to physical PIN entries.

II. LITERATURE SURVEY

A. *Internet of things*

Internet of Things (IOT) is a very common term nowadays. It's not a second internet; rather it's anetwork of devices that are connected to the Internet that is used every day to search Google, upload images and connect with friends. It's a network of products that are connected to the Internet, thus they have their own IP address and can connect to each other to automate simple tasks. However, IOT is still in its infancy. It has not been completely developed and is fragmented. For the IOT to be fully realized all devices need to be able to connect to each other, regardless of what company manufactured the product or which companies have business relationships with each other. In technical terms, the Internet of Things (IOT) is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. IOT is the future of technology that can make our lives more efficient, starting from the most mundane, everyday events to big, world changing ones.

B. *NumPy*

NumPy is an open source library available in Python that aids in mathematical, scientific, engineering, and data science programming. It works perfectly well for multi-dimensional arrays and matrices multiplication. NumPy is a programming language that deals with multi-dimensional arrays and matrices. On top of the arrays and matrices, NumPy supports a large number of mathematical operations.

NumPy is memory efficiency, meaning it can handle the vast amount of data more accessible than any other library. Besides, NumPy is very convenient to work with, especially for matrix multiplication and reshaping. On top of that, NumPy is fast. In fact, TensorFlow and Scikit learn to use NumPy array to compute the matrix multiplication in the back end.

C.*Open CV*

Object Detection using Haar feature-based cascade classifiers is an effective object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001. It is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images.

III. REAL-TIME EYE DETECTION SYSTEM

A.*LAPTOP CAMERA*

In this particular application, the viewer gazes at the digits of a digital key pad (Fig. 1). The camera captures the movement of the pupils and locates the eye centre, if the eye template matches, then it allows the user to enter the pin.

1	2	3
4	5	6
7	8	9

Fig. 1. Digital keypad for eye gazing and tracking

The eye detection algorithm processes about seven frames per second. The camera frame rate specification is not very stringent for this application, since the user pauses over each digit for several seconds. Typical 33 fps or slower cameras could also be used for gaze-based password identification.

B.*PERSONAL IDENTIFICATION NUMBER ENTRY*

Gaze-based PIN entry involves the user entering the PIN code by looking at the digital key pad (Fig. 1). The user stares at each digit of the PIN for a few seconds, and sequentially moves to the next digit with his/her eyes. When user is viewing the digits in the key pad the laptop camera captures the image of the partial face in sequential frames, computes the eye centre location using image processing algorithms, and records the Cartesian coordinates which represents the eye centre in an on-board spreadsheet file, with the associated image frame number. The eye tracking is stopped when PIN entry is completed.

C.*EYE TRACKING ALGORITHM*

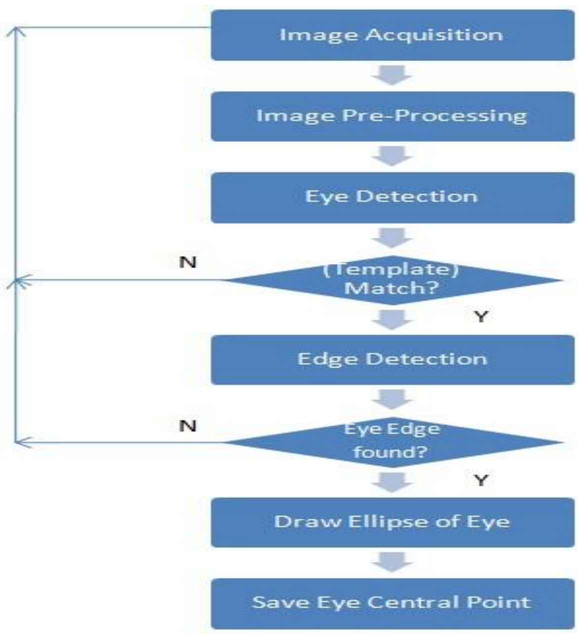

Fig. 2. Flowchart of the Real-time Eye Tracking Algorithm

Step 1: Image Acquisition: The raw image captured automatically by the laptop camera.
Step 2: Image Pre-processing: The acquired raw image is converted to a grey-scale image.
Step 3: Eye Detection: Initially the user's eye is detected using template matching. In template matching, a template from the S is compared with the given image using a matching metric. The matching metric provides similarity between two templates. This similarity is converted into a numerical value as a score of the template match.
Step 4: Edge Detection: If the eye is detected, new Region of interest (ROI) which covers only the eye is extracted to reduce the processing area. An edge detection technique is applied to the new ROI to find points around the ellipse or circle of the eye.
Step 5: If at least 3 points (for circle) or 4 points (for ellipse) are found, the circle or the ellipse of the eye will be drawn. Otherwise, the current frame is skipped over with no match.
Step 6: If the ellipse/circle of the eye is drawn the coordinates of the centre is calculated in the camera's processor and saved to spreadsheet for future reference. These coordinates are computed as the centre of the rectangle bounding the detected eye.

D.*EYE DETECTION ALGORITHM*

Eye detection begins with training the algorithm. This is obtained by capturing a single frame of the user's eye and saving it as the eye template. The process is then followed by eye tracking algorithm, which captures the images of the user's face and detect the position of the eye. If the matching is accomplished using the saved template and scanned image. The coordinate system obtains the coordinates of the reported eye location, to allow capturing eyes at various angles. From the best matched eye image, iris of the eye is detected using circle or oval matching via edge detection. The iris centre is then mathematically computed as the centre of the circle or oval, and recorded in a spreadsheet on board the camera as the eye centre. The LabVIEW block diagram for eye detection and tracking is shown in (Fig. 3).

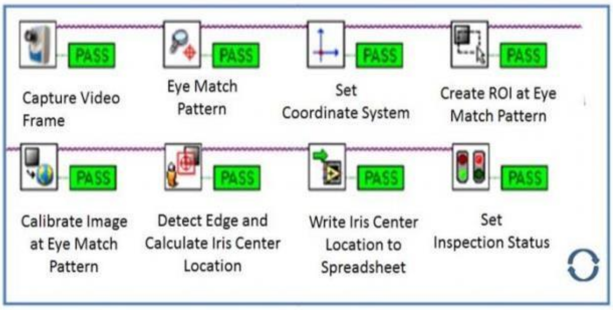

Fig. 3. LabVIEW block diagram for eye detection and tracking

Step 1: A real-time eye image is captured using a laptop camera interfaced to LabVIEW.
Step 2: The eye is matched with a pre-stored eye pattern (only the best match is reported).
Step 3: The coordinate system and origin are aligned with the captured eye image.
Step 4: The Region of Interest is set to the matching eye area to reduce the area that is processed.
Step 5: The image is calibrated to convert pixel coordinates to real-world coordinates.
Step 6: An ellipse or circle is drawn based on the points extracted from the Region of Interest. Also, the tracking line is drawn from a list of points that represent the center of the eye from successive frames.
Step 7: The coordinates of the center of the eye are stored in a spreadsheet.
Step 8: The inspection status is shown through pass (green) or fail (red) indicators.

E.*GAZE-BASED PIN IDENTIFICATION*

For PIN identification, firstly the eye centre co-ordinates (horizontal and vertical) must be plotted on a 2D spreadsheet, then the data points are grouped using clustering. In this process, the gaze digits are obtained but not in the order which they are gazed. To obtain the order of the entered digits, a 3D connected graph is plotted.

F.*Block Diagram*

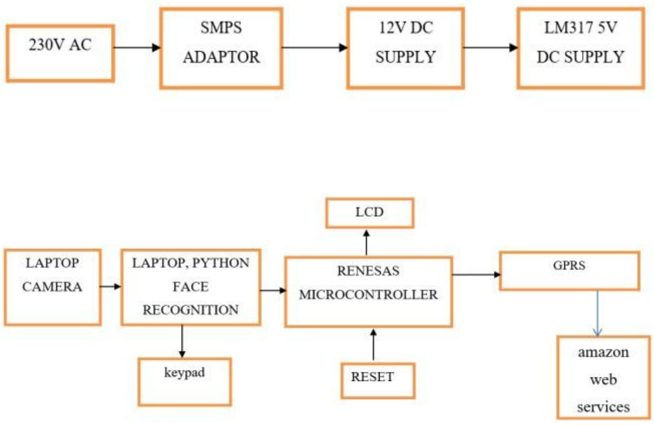

Fig. 4. Block diagram

R5F100LEA microcontroller from Renesas RL78 series which is a 16-bit microcontroller is used to implement this project. Microcontroller acts as the heart of this project, which controls the whole system. It contains of Flash ROM 64KB, RAM 4KB and Data Flash 4KB, and it has High speed on-chip oscillator, Self-reprogrammable under software control, 58 GPIO's, 3 UART's, Simplified I2C, 10-bit resolution ADC, 28 Interrupt Sources, ISP programming support, etc. The Renesas microcontroller is the heart of the project it is programmed such that it keeps on commanding and controlling the complete action through peripherals connected.

IV. RESULTS

A.*EYE TRACKING RESULTS*

Table 1 shows us the sample spreadsheet which illustrating the data recorded during eye tracking for the four-digit PIN code, 4-4-2-1. The table also shows us the frame numbers that are analysed. The horizontal and vertical coordinates of the eye centre location is also showed in the table.

FRAME NUMBER	X-AXIS (pixels)	YAXIS (pixels)
0	4411	1169
5	4412	1170
11	4415	1170
16	4415	1169
22	4415	1171
27	4404	1171
33	4402	1171
35	4421	1173
38	4021	1179
43	4022	1179
55	4022	1176
50	4014	1175
65	4411	1175
71	4412	1177
76	4419	1178

Table.1. Sample Eye Tracking Output

B.*EYE DETECTION RESULTS*

The results of eye detection is noted. After the process of eye detection, iris centre is computed, and later it is recorded as the eye location. Later the data is been saved for the further process.

C.*PIN IDENTIFICATION AND AUTHENTICATION RESULTS*

The (Fig. 5) shows us the scatter plot of four-digit PIN with a repeated digit is demonstrated. We note that some of the detected eye locations and eye detections which appears as a single point on the scatter plot, but can easily be distinguished in the temporal or sequential plots of eye coordinates that is xaxis and y-axis.
In (Fig. 6) the user tries the entry of the four-digit pin for four times to demonstrate the repeatability of the process. This is a sequential graph demonstration. The connected scatter plot identifies the order of the eye gazed PIN for authentication.

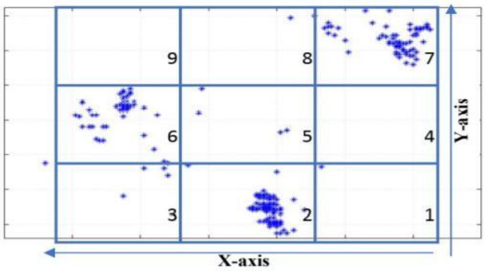

Fig. 5. Scatter plot of tracked eye location during 4-digit PIN entry with a repeat digit 2. Overlaid keypad is flipped about horizontal and vertical axes.

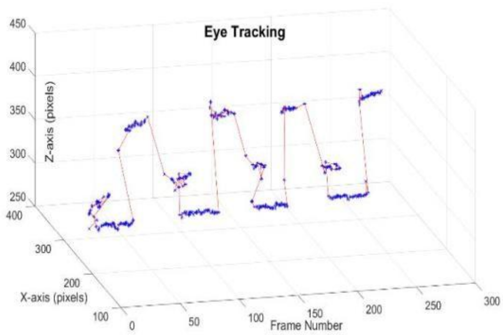

Fig. 6. Sequential eye tracking for identifying gaze-based PIN. The user repeated the four-digit PIN entry four times to demonstrate repeatability.

D.*PASSWORD AUTHENTICATION USING AWS*

Table. 2 shows data stored in amazon cloud. It contains user details like their names, password, time and date of recent access.

Time	Date	User	Password	Status
10:12:04	01-05-2019	xyz	*****	OK
11:16:08	02-05-2019	abc	*****	None
04:17:04	03-05-2019	pqr	*****	OK

Table.2. AWS data Storage

V. CONCLUSION

In this paper we discussed every possible ways of the password authentication by the use of real time eye detection and tracking. The laptop camera-based eye tracking system has been used for many applications for gaze-based pin entry and identification. The discussed paper's system works completely with 4-digit PIN entry and it has been tested successfully. The graph plots will show us the Stray data points in the scatter plots are generally associated with transitional movement of the eyes between digits. The stability of the user's eye gaze will affect the accuracy of the detected pins. Our model is resistant to shoulder attack, smudge attack, and many other attacks that infer a user's password based on the hand input information.

VI. FUTURE ENHANCEMENT

Future work includes allowing the algorithm of our model for the real time applications to use for all the one-time more than four-digit PIN entry system. It can also be extended to character and digit combination password entry. Some of the further updates can also be done for their requirements which are more robust algorithms to improve the performance of the system under varying conditions. In addition to this it can also be implemented in mobiles, smart-cameras, etc. Experimental data collection can also be improved by normalizing the distance from the camera to the tester.

REFERENCES

1. R. Revathy and R. Bama, "Advanced Safe PIN-Entry Against Human Shoulder-Surfing," *IOSR Journal of Computer Engineering*, vol 17, issue 4, ver. II, pp. 9-15, July-Aug. 2015. (Available: http://www.iosrjournals.org/iosr-jce/papers/Vol17-issue4/Version-
2. 2/B017420915.pdf)
3. J. Weaver, K. Mock and B. Hoanca, "Gaze-Based Password
4. Authentication through Automatic Clustering of Gaze Points," *Proc. 2011 IEEE Conf. on Systems, Man and Cybernetics*, Oct. 2011.
5. (DOI: 10.1109/ICSMC.2011.6084072)
6. "ATM Fraud, ATM Black Box Attacks Spread Across Europe", *European ATM Security Team (E.A.S.T.),online, posted 11 April 2017. (*Available: https://www.european-atm-security.eu/tag/atmfraud/)
7. Mehrube MEHRUBEOGLU, *Senior Member*, *IEEE*, and Vuong Nguyen, *Member*, *IEEE* ,Texas A&M University-Corpus Christi, Corpus Christi, Texas, USA ruby.mehrubeoglu@tamucc.edu, vnguyen4@islander.tamucc.edu
8. M. Mehrübeoglu, H. T. Bui and L. McLauchlan, "Real-time iris tracking with a smart camera," *Proc. SPIE 7871*, *787104*, 2011. (DOI:10.1117/12.872668)
9. M. Mehrubeoglu, L. M. Pham, H. T. Le, M. Ramchander, and D. Ryu,
10. "Real-time eye tracking using a smart camera," *Proc. 2011 IEEE*
11. *Applied Imagery Pattern Recognition Workshop (AIPR '11),* pp. 1-7, 2011. (DOI: 10.1109/AIPR.2011.6176373)
12. M. Mehrubeoglu, E. Ortlieb, L. McLauchlan, L. M. Pham, "Capturing reading patterns through a real-time smart camera iris tracking system," Proc. SPIE, vol. 8437, id. 843705, 2012. (DOI: 10.1117/12.922875)
13. Manu Kumar, Tal Garfinkel, Dan Boneh, Terry Winograd Stanford University Gates Building 353 Serra Mall, Stanford, CA +1.650.725.3722 {sneaker, talg, dabo, winograd}@cs.stanford.edu