International Journal of Latest Trends in Engineering and Technology
Vol.(8)Issue(1), pp.643-649
DOI: http://dx.doi.org/10.21172/1.81.084
e-ISSN:2278-621X

BEHAVIORAL BIOMETRIC AUTHENTICATION USING LEAP MOTION SENSOR

Prof. Saritha L.R[1], Diana Thomas[2], Neema Mohandas[3], Pooja Ramnath[4]

Abstract- In this project, we intend to provide a solution for authentication of files or applications in Laptops or PCs. The Leap Motion sensor will be used for uniquely identifying the genuine user and grant him access to the system. The different existing methods of authentication have their own pros and cons. The Leap sensor is a camera based sensor which will capture the hand geometry and gesture of the user and store it. The gesture decided by the user at first, known as the Leap password, will also vary from person to person thus ensuring that the stored details are unique for every person. This will help the authorized user to access his system without the fear of an intruder cracking his password or gaining access by using the same gesture. The sensor will check the details of the person accessing the system such as dimensions of the palm and finger. If they match with the one stored by the sensor, the user is granted access to the system. This system will thus ensure that the files are not tampered or accessed by any other person except for the owner of the PC or laptop.
Keywords – Authentication, Leap Motion Sensor, Leap password.

I. INTRODUCTION

The Leap Motion sensor is a camera based sensing device that captures gestures and motion data from the user as input to a computing device. It consists of optical sensors and infrared light which help in detecting hand gestures and positions for human-computer interaction. The Leap sensor tracks a user's hand in three-dimensional space and a temporal resolution of 120 frames per second using the infrared sensors. The sensor captures the movements of hands of a user and hence, used widely in gaming applications. This sensor which analyses details of the hands can be used for authentication purpose. The user's unique signature, his finger length and palm details can be used to make a Leap password. This will allow only the user whose details are captured by the sensor to access files in PC. All these details are used for verification purposes in conjunction, classifiers. Thus, this type of 3D hand gesture motion capture is based on the two factors: hand geometry and hand gestures.

II. PROPOSED ALGORITHM

The Leap Motion Sensor will be used to capture the user's hand action or gestures above the camera and translate them into 3D input using LED and infrared cameras within the sensor. For obtaining the behavioral motion data, the normalized fingertip position will be collected in three dimensional space along with the magnitude of velocity and the direction of the fingertip.

The user will be asked to simulate his Signature above the sensor at his natural speed thereby allowing the sensor to capture the necessary information. The captured gestures and the input given by the user will be compared to the template which is already present in the system. The user will be given access only if the input matches with the stored template thus providing authentication.

[1] *Department of Information Technology Engineering SIES Graduate School of Technology, Nerul, Maharashtra, India*
[2] *Department of Information Technology Engineering SIES Graduate School of Technology, Nerul, Maharashtra, India*
[3] *Department of Information Technology Engineering SIES Graduate School of Technology, Nerul, Maharashtra, India*
[4] *Department of Information Technology Engineering SIES Graduate School of Technology, Nerul, Maharashtra, India*

Block Diagram:
The two main phases incorporated in the leap motion sensor for the authentication process are:
1) User enrolment

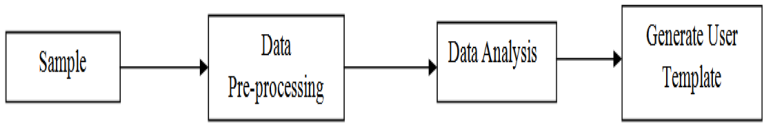

Fig.1. User Enrolment

The diagram given above depicts the first stage of the user that is User enrolment.
The user's hand actions are first captured by the leap motion sensor. The biometric data derived from derived from the user's hand. The data pre-processing occurs wherein by storing the starting position and the offsetting the signature position by the starting position .Data analysis occurs with the help of the DTW (Dynamic time warping algorithm). The initial user template is generated for further authentication purposes.

2) Verification and Authentication

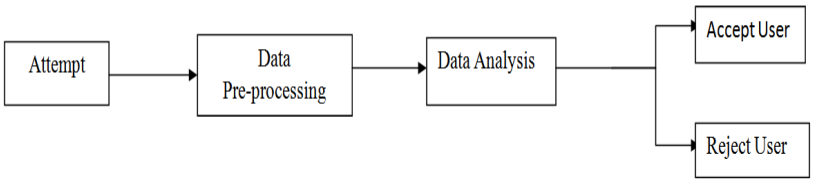

Fig.2. Verification and Authentication

The diagram above depicts the verification and authentication phase.

In this phase when the user tries to access into the system the second time (that is, the first time after his biometric data has been captured). The user's data then is pre-processed and the unwanted frames are eliminated (frames containing multiple hands or those outside the authentication level or zone). Data analysis occurs with the requisite algorithm which is in this case the Dynamic Time Warping algorithm. Now, depending on the result of this authentication process the user is verified and seen whether it is the authenticated user trying to access the system or whether an unauthorized user is trying to gain access. If the user is an authorized one, that is, the biometric and behavioral features match then the user is provided access else he is rejected access by the system.

DFD level 0:

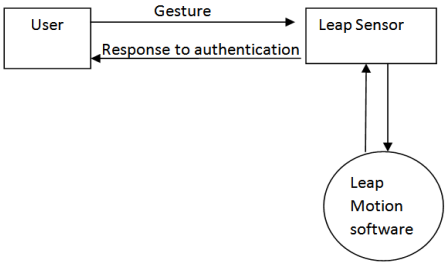

Fig.3. DFD level 0

The initial phase shows that the user inputs his gesture which is captured by the leap motion sensor. The Leap motion sensor forwards it further to the leap motion software for further processing. The result of this software is provided back the leap motion sensor and the user is then provided with the response for its authentication.

DFD level 1:

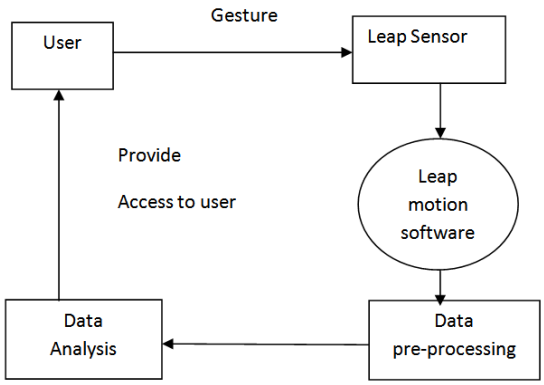

Fig.4. DFD level 1

The initial phase shows that the user inputs his gesture which is captured by the leap motion sensor. This Leap motion sensor forwards it further to the leap motion software for further processing. The Leap motion software performs acquisition of data where the Leap motion SDK helps in obtaining useful information from the captured data and helps in avoiding the unwanted frames like if the user slightly moves his hand from the confidence zone or the frames that contains no hand or multiple hands etc. The next process is data pre-processing. Due to the size of the interaction space above the sensor, the user can simulate his or her signature at different positions each time in front of the user, to avoid this we capture the signatures position relative to the starting position. The data is then analyzed by the Dynamic Time Warping algorithm where a threshold is set and the required parameters (biometric data and behavioral data)of the hand are calculated and access is provided to the users depending on the results of verification.

Flow chart:

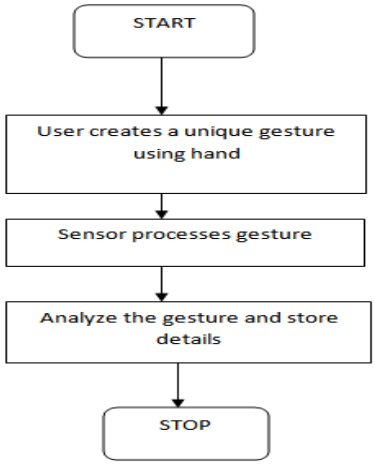

Fig.5. Flow chart for first access to the Leap motion System

The first flow chart depicts that initially the user places his hand in front of the Leap motion sensor and the sensor captures the users hand actions above the camera and translates them to 3 D input using two infrared camera and infrared LED within the sensor.

When activated, the sensor constantly captures 200 frames per second. Here, in the Leap Motion Sensor the biometric data derived from the user's hand and the behavioral motion is generated when the user signs his signature using his hand in front of the sensor. The leap motion Sensor then analysis this data and stores details.

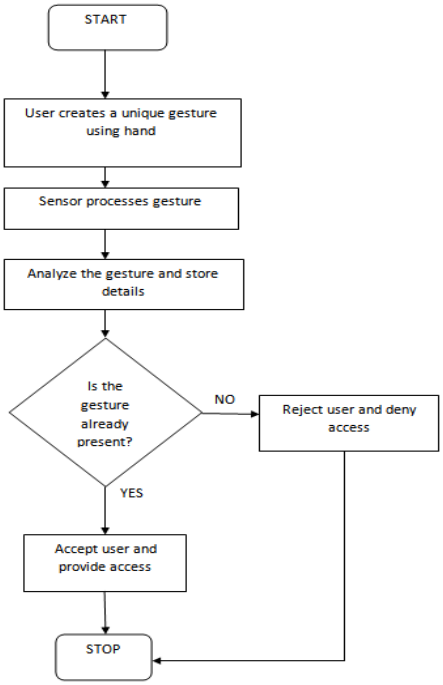

Fig.6. Flow chart for further access to the Leap motion System

The second flow chart shows that the user creates a unique gesture using his hand and the sensor analysis this hand gesture and eliminates the unnecessary frames using Leap Motion SDK.

The leap motion controller also recognizes four basic gestures out of the box. Each of these gestures has inherent properties such as the time to complete the gesture. Clockwise from top left, the gestures are circles, swipe, screen tap and key tap.

Fig.7. Leap sensor

Fig.8. Capturing the gesture

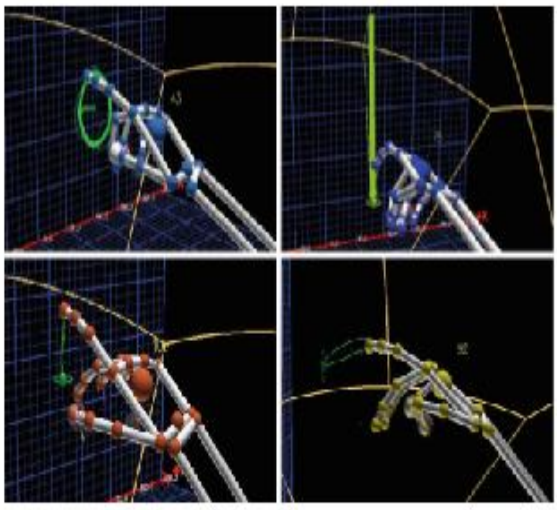

Fig.9. Gesture Recognition

The criteria involved in the SDK include:
1) Confidence :
During the data collection process,the user may move his or her hand slightly out of the confidence detection zone of the sensor.These inaccurate frames can be eliminated by filtering out the frames where the confidence of the frames is less than say a threahold of 0.20.
hand.confidence()>=0.20

2)Hand detection:
Assuming the user will be using only one hand to verify hand biometrics and simulate a signature, we discard unneccessary data by fitering out the frames based on hand detection. Since the leap motion sensor constantly captures frames, we can filter out those frames where no hand is detected or multiple hands are detected using the following condition:
Frame.hands().count()==1

3)Finger numbert detection in biometrics:
Assuming the user will be stretching all five fingers for the hand biometric collection, we can filter out the frames with 4 or less detected fingers with the following condition:
Hand.fingers().extended().count==5

In Data Preprocessing due to the size of the interaction space above the sensor, the user may simulate his or her signature at different positions each time. In order to accomodate this, we offset the signature by its starting position to minimize the differences. By storing the starting position and offsetting the signature position by its starting position, we can capture the signatures position relative to its starting position by using the following code:
startX= frame.interactionBox().normalizePoint(frontmostFingerFinger.tipPosition(),false).getX();
startY= frame.interactionBox().normalizePoint(frontmostFingerFinger.tipPosition(),false).getY();
startZ= frame.interactionBox().normalizePoint(frontmostFinger.tipPosition(),false).getZ();
Then the data is anlysed by checking the data already stored by the software present in the leap motion sensor.
Based on this data if the biometrics of the user match then the user is given access into the system else he is rejected and not given access to the system.

After a user's hand biometrics and behavioral signature are captured several algorithms can be used. Among them the Dynamic Time Warping (DTW) algorithm is the most efficient.
The Dynamic Time Warping algorithm calculates the difference (or distance) between two datasets:dtw_c(a,b) where a represents the first dataset and b represents the second dataset.

The advantages of using DTW algorithm are two-fold:
1) Any type of numerical data can be applied to numerical data can be applied to DTW, allowing the input not to be limited to X/Y/Z positions but to include more such as a magnitude of the velocity and the directions pitch, roll and yaw.

2) The number of frames collected in each trial can vary instead of being constant, tolerating the slight variation in the number of frames captured in each trial.

To use DTW in user authentication, a threshold is needed to distinguish a true user (distances smaller than the threshold) from imposters. The threshold should accommodate slight changes in scale, rotation, speed, direction and position between genuine signatures dynamic.
Dynamic Time Warping (DTW) is an effective algorithm based on dynamic programming. By treating the input as a time series of 3D (three dimensional) positions, DTW algorithms can be used to recognize characters. The task of identifying characters in a time series requires data to test and train on. Their first goal is to have a similarity search algorithm.

The similarity search will sweep across the data time series, checking every subsequence against the candidate and returning the best match. Both candidates and all subsequence are z-normalized in the process.
The dynamic time warping algorithm is used as a similarity metric between vectors. It is a generalization of the Euclidean distance metric but chooses the closest point within a certain time window, rather than creating a one-to-one mapping of points.

III. Experiment and Result

Leap Signature	Parameters	Probability
Circle Gesture	Hand type, fingers details, palm position, wrist position, radius, clockwise/anticlockwise, angle	60%
Swipe Gesture	Hand type, fingers details, palm position, wrist position, swipe direction, speed	70%
Screen Tap Gesture	Hand type, fingers details, palm position, wrist position, screen tap direction	90%
Key Tap Gesture	Hand type, fingers details, palm position, wrist position, key tap direction	85%

Table 1: Expected Output

If the parameters mentioned above are captured and matches the stored template in the leap software, the user will be identified as genuine and granted access.

IV.CONCLUSION

The leap motion sensor takes authentication procedures to a whole new level. Additionally, it is a highly secure, reliable, compact and feasible system. Authentication using Leap motion sensor is not only innovative idea but also a sustainable one. Hence, there will be lesser intrusion into the file with maximum authenticity, confidentiality and integrity.

REFERENCES

[1] Grady Xiao, MariofannaMilanova and MengjunXie, "Secure Behavioural Biometric Authentication with Leap Motion", Johns Hopkins University, ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7473528-27,April.2016

[2] Alexander Chan, TziporaHalevi, "Leap Motion Controller for Authentication via Hand Geometry and Gestures" Hunter College high school, New York, USA, -www.springer.com/cda/content/document/cda.../9783319203751-c2.pdf, 2-7,August.2015

[3] AmanChahar, ShivangiYadav and Ishan Nigam, "A Leap Password based Verification System", IIIT-Delhi, New Delhi, India, 2015

[4] Saritha LR, "Energy Efficient Routing Protocols for wireless Sensor Networks- A Review" SIES Graduate School of Technology, Navi Mumbai, India, ICERAT, 2016

[5] Jayash Kumar Sharma; Rajeev Gupta; Vinay Kumar Pathak, "Numeral Gesture Recognition Using Leap Motion Sensor", Rajasthan Tech. University, Kota, IEEE Conference Publications, India,2015

[6] Rajesh B. Mapari; Govind Kharat, "Real time human pose recognition using leap motion sensor", Department of Electronics & Telecommunication Engg., Anuradha Engineering College, Chikhli, India, IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), 2015

[7] Alina Delia Calin, "Gesture Recognition on Kinect Time Series Data Using Dynamic Time Warping and Hidden Markov Models", Department of Computer Science Babeș-Bolyai University Cluj-Napoca, Romania,2015

[8]T. Arici, S. Celebi, A. S. A. and T. T. Temiz, "Robust Gesture Recognition Using Feature Pre-Processing and Weighted Dynamic Time Warping," Multimedia Tools and Applications, October 2014, Vol. 72, Issue 3, pp. 3045-3062.

[9]J. M. Carmona and J. Climent, "A Performance Evaluation of HMM and DTW for Gesture Recognition," Proceedings of the 17th Iberoamerican Congress on Pattern Recognition (CIARP), Buenos Aires, Argentina, Sept, 2012, pp. 236-243.

[10] John D'souza "Adaptive dynamic time warping for recognition of natural gestures" IBISC laboratory, University of Evry, France, LIMIARF laboratory, University of Mohammed V Agdal, Morocco,2015.