Leap Motion Controller for Authentication via Hand Geometry and Gestures

Alexander Chan[1](✉), Tzipora Halevi[2], and Nasir Memon[2]

[1] Hunter College High School, New York, NY, USA
alexanderchan97@gmail.com
[2] New York University Polytechnic School of Engineering, Brooklyn, NY, USA

Abstract. The Leap Motion controller is a consumer gesture sensor aimed to augment a user's interactive experience with their computer. Using infrared sensors, it is able to collect data about the position and motions of a user's hands. This data allows the Leap to be used as an authentication device. This study explores the possibility of performing both login as well as continuous authentication using the Leap Motion device. The work includes classification of static data gathered by the Leap Motion using trained classifiers, with over 99 % accuracy. In addition, data was recorded from the users while utilizing the Leap Motion to read and navigate through Wikipedia pages. A template was created using the user attributes that were found to have the highest merit. The algorithm found when matching the template to the users newly collected data, the authentication provided an accuracy of over 98 %, and an equal error rate of 0.8 % even for a small number of attributes. This study demonstrates that the Leap Motion can indeed by used successfully to both authenticate users at login as well as while performing continuous activities. As the Leap Motion is an inexpensive device, this raises the potential of using its data in the future for authentication instead of traditional keyboard passwords.

Keywords: Security · Biometrics · Authentication

1 Introduction

With the introduction of "motion capture" devices to the consumer market, users have experienced many new ways to interact with computer systems. One of the first of such systems was the Nintendo Wii, which utilized an infrared emitter in the form of a "sensor bar" and an IR receiver in a separate remote. Users moved the remote relative to the IR receiver in order to interact with items on-screen. One significant deficiency of this technology, however, is that the Wii senses only the remote and its movements, rather than the person himself.

As this and similar motion capture technology has progressed, systems have become smaller and more self-contained. Most recently, a gesture sensor called the "Leap Motion controller" (Fig. 1) was released. It uses both optical sensors and infrared light to detect a user's hands. The controller sits flat on a table, with

T. Tryfonas and I. Askoxylakis (Eds.): HAS 2015, LNCS 9190, pp. 13–22, 2015.
DOI: 10.1007/978-3-319-20376-8_2

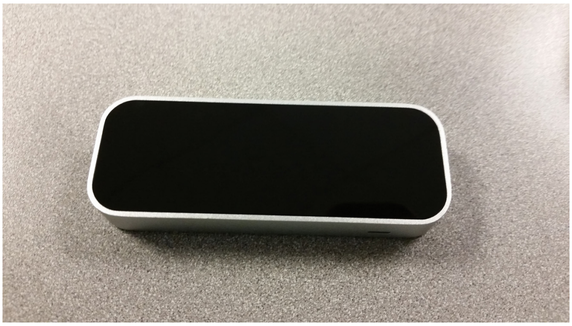

Fig. 1. The Leap Motion controller is capable of collecting physical and gesture properties of a user's hands.

the sensors pointing up, creating a range of detection in the shape of an inverse pyramid extending up to 600 mm and encompassing a field of view of approximately 150 degrees. The system interprets data from the sensors alongside an internal model of a human hand to determine the position and orientation of a user's hand.

This type of 3D hand gesture motion capture allows for biometric authentication based on two different factors: hand geometry and hand gestures. A large body of literature addresses both hand geometry and hand gesture based authentication, but very little considers gestures in the third dimension, or 3D motion capture devices. In addition to these two types of biometric features, the Leap Motion can also monitor human interactions with a computer, so that it can authenticate both statically, that is, only when access is requested, and dynamically, or continuously, in which identity is continuously verified during device use.

In this study, we assess the feasibility of a 3D gesture device, the Leap Motion controller, for static and continuous authentication based on hand geometry and gestures. Section 2 elaborates on related work and Sect. 3 discusses the features and specification of the Leap Motion controller. Section 4 describes our approach, Sect. 5 presents our results, and Sect. 6 concludes.

2 Related Work

Hand geometry has long been known to be unique among individuals and has already been used as a biometric [7]. Commercial systems, such as the Schlage HandKey, require a user to place his hand around a series of pegs, while a camera takes an image. Such systems measure a set of 30 typical features. That include only physical geometries, and not other physical identifiers such as fingerprints, palm prints, or other markings on the hand.

In addition to simple hand geometry, gesture behavior has been explored on touchscreens and touchpads in both static and continuous contexts [2,5,8,9]. Angulo determined that in a static context, the time taken in drawing a pattern lock contains enough unique information for authentication. Training required drawing at least 250 keystrokes, which makes this method less applicable in a real life situation [5]. Sae-Bae showed that touch gestures are unique to each person, and can authenticate users [9]. In a continuous context, it was also shown that gestures meant for interacting with a device can also authenticate [8]. A Hidden Markov model was shown to be able to ease training while still supporting a high accuracy [2]. In addition, it was shown that a random forest classifier provided the highest identification accuracy, which is supported by other studies regarding gesture data [1]. We apply this work to a 3D gesture device, using a random forest classifier.

3 Leap Motion Controller

The Leap Motion controller is a consumer 3D gesture sensor. Using both optical sensors and infrared light, it detects hand gestures and positions for a novel method of human-computer interaction. Its detection range is in the shape of an inverse pyramid, extending up to 600 mm with a field of view of approximately 150 degrees. Beginning with the v2 Leap software, authentication both receives and interprets data from sensors and compares such data to an internal model of a human hand in order to determine accurately the position and orientation of a user's hand.

The Leap controller generates, on a frame by frame basis, information regarding a user's hands as well as information pertaining to already recognized gestures. A frame typically contains the position of objects, and in the case of a hand, the frame also contains physical properties such as the width and length of the hand and arm as well as the width and length of each digit and the four bones associated with each digit. In addition to these properties, the Leap recognizes certain movement patterns as "gestures". The Leap records gestures for each finger. There are four currently recognized gestures: a circle, a swipe, a key tap, and a screen tap. A circle gesture is simply a single finger drawing a circle; a swipe is a long linear movement of a finger; a key tap is a finger rotating slightly downwards and back up; and a screen tap is a finger moving forward and backward quickly. Figure 2 shows all four gestures in the diagnostic visualizer, with path tracking for visibility. These four gestures, moreover, have their own properties, such as speed.

There has been at least one study analyzing the reliability of the data generated by the Leap controller, and many others analyzing the use of the controller in practical applications such as sign language recognition [3,4,6]. Using an industrial robot capable of providing an accuracy of less than 0.2 mm, meaning it is capable of moving an object in increments of no more than 0.2 mm, Weichert et al. have shown that the Leap has an overall detection accuracy of 0.7 mm, and that the size of "pointable objects", such as fingers, does not affect the accuracy. This high accuracy is complemented by the "hand confidence", a value ranging from 0 to 1 based on the correlation between the observed data and an internal hand model. However, Jakus et al. demonstrate that while static objects are reliably tracked, moving objects are tracked less accurately. Potter et al. also mention that occlusion of parts of the hand negatively affects tracking.

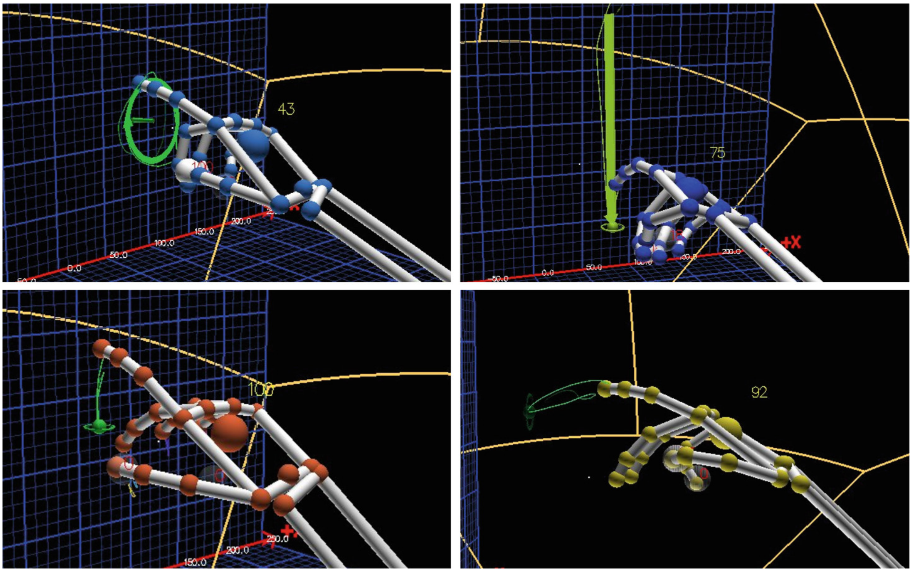

Fig. 2. The Leap Motion controller recognizes four basic gestures out of box. Each of these gestures has inherent properties, such as the time to complete the gesture. Clockwise, from top left, the gestures are: circle, swipe, screen tap, and key tap

Many of these features, such as the hand confidence and the internal hand model, were introduced with v2 of the Leap Software, which also includes skeletal tracking. However, previous research generally used v1, which suffered from poor tracking, especially when part of a hand was occluded. As a result, many previous conclusions of the Leap Motion controller's inconsistency with extensive practical applications might not apply to the updated software.

4 Data Collection

This study is composed of two parts. The first part mimicked a static authentication scenario, where the user would use the Leap for the purpose of login-in authentication. The second analyzed a continuous authentication scenario, where the user performs basic actions using the Leap, i.e. reading web pages online (without performing any explicit user authentication), while simultaneously being authenticated.

4.1 Static Authentication

In this section of the work, 16 participants provided data in two identical sessions. They were asked to place both hands above the Leap Motion controller for 25 s. During this time, only frames where the confidence value of each hand was above 0.9 were recorded. Once the confidence reached this value, the participant was then asked to draw a circle with one finger. The two sessions ensured repeatability of the data.

107 attributes were stored at a rate of 55 frames per second. Three attributes are gesture properties, pertaining to the circle gesture, and the rest are physical properties, consisting of the width and length of both hands and arms, all digits, and the four bones associated with each digit (metacarpal, proximal, intermediate, and distal). For the thumb, the metacarpal bone has length 0. The gesture properties include the radius of the circle, the duration, in seconds, of the gesture, and the acceleration of the finger ($\frac{2\pi r}{t}$, where r is the radius and t is the duration of the gesture).

4.2 Continuous Authentication

For continuous authentication, 10 participants used a program enabling interaction with the operating system through the Leap Motion controller. To move the mouse cursor, users would move their right index finger. Clicking involved a "key tap" gesture with the left hand or a "screen tap" gesture with the right hand, and scrolling involved either a "circle" gesture with the left hand or a "swipe" gesture with the right hand. Subjects were given a few minutes to familiarize themselves with the various controls. Each user was told to navigate from one random Wikipedia page to another, only through clicking links.

Data for this part involved 135 attributes. In addition to the physical properties recorded (the same as those recorded in the previous part), gesture properties were also recorded. These included the orientation of each hand, the speed of each hand and finger, the "grab" and "pinch" strengths of each hand, and various properties pertaining to the four prerecognized gestures.

5 Authentication Algorithms

Two algorithms were implemented for authentication. In the first one, classifiers were used on all of the attributes. In the second algorithm, a template matching approach was adopted to explore the possibility of identifying user's data based on a small number of attributes.

5.1 Classification

A random forest classifier was decided upon for the classification step. It was shown that for such behavioral data sets, random forest classification is superior to other algorithm types, such as linear and kernel density estimation classifiers [1]. For example, a naive Bayes classifier fails, due to high correlation between certain features.

The Waikato Environment for Knowledge Analysis (WEKA) was used for classification. A random forest classifier was used on both data sets. The model was built using a 10-fold cross validation, that is, the data were partitioned into 10 subsamples, and each was used as a test set, while the remainder was used as a training set.

In addition to classification, an attribute selector, in conjunction with the ranker search method, was used to identify which attributes contribute most to the classification. The algorithm tests each attribute and returns a value that indicates how important it is for classification (the higher the value, the more important the feature is).

5.2 Template Matching

For data collected from the second part of the work, further analysis was done to determine the Equal Error Rate (EER), which is the value at which the False Acceptance (FAR) and False Reject Rate (FRR) are equal. First, the data was reduced to a sample size of 4500 instances per user. This data set was then split in half such that 2250 data values were used to build the template for the user, and the rest were used to test the template. Two attributes from the training data were selected to build a template of each user - these attributes were selected based on their relative importance. The template consisted of the weighted averages of these two values.

Every data value from the testing set was compared against the template. A threshold was chosen, and if the distance between the data value and the template value was smaller than this threshold, it was considered an accept. Otherwise, it was considered a reject. If a data value not belonging to that user's template was accepted, the false acceptance rate increased. If a data value belonging to that user's template was rejected, the false rejection rate increased.

A range of thresholds, beginning at 0 % and ending at 100 % (percentage of maximum possible distance), were tested. The intersection of these two rates was calculated to be the EER. Following are the formulas for these algorithms.

Template creation for user i, when b attribute vectors are used:

$$T_i = Profile(user_i) = average(v_i^1 \ldots v_i^b)$$

Authentication for user i when given attribute vector v_i and a threshold τ:

$$auth(T_i, v_i) = 1 \quad if \quad |T_i - v_i| > \tau, \quad 0 \quad otherwise$$

The distance that is measured is the Euclidean distance.

6 Results

During classification, the identification rate, which measures the percentage of correctly classified instances, was used as the main indicator of performance. The closer the identification rate was to 100 %, the better the classifier performed. In addition to this metric, WEKA provides more specific information, including the true positive rate, false positive rate, precision, recall, F-Measure, and ROC area.

The true positive rate, or the recall rate, is the ratio of true positives to total positives, while the false positive rate, or false acceptance rate (FAR), is the ratio of false positives to total negatives. Higher TP/recall rates and lower FP rates imply more accurate classifications. The precision, or positive predictive value, is the ratio of true positives to classified positives. F-Measure combines both precision and recall, and is calculated using $\frac{2pr}{p+r}$ where p is the precision and r is the recall. The closer the F-Measure is to 1, the more accurate the classifier is. Finally, the ROC area is the area under a graph of the FP rate versus the TP rate. It represents the probability that the classification algorithm will rank a random positive higher than a random negative. Like the F-Measure, the closer the ROC Area is to 1, the more accurate the classifier.

6.1 Static

Classification accuracy for static authentication using the random forest algorithm was 99.9667 %. Out of 12,006 instances, only 4 were incorrectly classified. The detailed classification is in Fig. 3.

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
Weighted Average	1	0	1	1	1	1

Fig. 3. Detailed accuracy by class for continuous authentication

Average merit	Feature description
3.765 ± 0	All physical properties
0.478 ± 0.006	Radius of circle gesture
0.134 ± 0.002	Acceleration of hand during gesture
0.127 ± 0.002	Duration of circle gesture
0 ± 0	Length of left and right thumb metacarpal bones

Fig. 4. Calculated average merit for each feature in classification. Merit is a relative measure, and so the higher its score, the more weight it is given during classification. Every single physical property, with the exception of the metacarpal bone length, has a higher merit than the other gesture properties.

All but 5 attributes had the same average merit of 3.765 ± 0. The only attributes that did not have a significant weight were the three gesture properties (radius of the circle, duration of the gesture, and acceleration of the hand), and the lengths of the metacarpal bones in both thumbs. A list of selected attributes and their average merit is presented in Fig. 4.

6.2 Continuous

For continuous authentication, classification accuracy was 98.3862 %. The weighted average of the classification is in Fig. 5.

As in static authentication, physical properties have a higher average merit than gesture properties. Figure 6 shows a list of all attributes and their average merit.

A template was constructed for each of the 10 users. Then each individual data value was tested against this template - if the test value was within a certain threshold of the weighted mean of the values, it was considered a success. Each training value was tested against each user template, allowing for both determination of false accept (when an illegitimate user was authenticated) and false reject (when the true user was rejected) rates.

Two features were chosen based on relative weights. The width of the right and left hand were compared against. A plot of the FAR and FRR is shown in Fig. 7.

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area
Weighted Average	0.984	0.001	0.986	0.984	0.984	1

Fig. 5. Detailed accuracy by class for continuous authentication

Average merit	**Feature description**
2.547 ± 0.001	Physical properties of right hand
1.074 ± 0.002	Physical properties of left hand
0.554 ± 0.001	Number of hands
0.537 ± 0.002	Roll of the right hand
0.452 ± 0.001	Pinch strength of the right hand
0.442 ± 0.002	Yaw of the right hand
0.386 ± 0.001	Pitch of the right hand
0.320 ± 0.001	Yaw of the left hand
0.178 ± 0.001	Roll of the left hand
0.150 ± 0.001	Grab strength of the right hand
0.144 ± 0.001	Pitch of the left hand
0.111 ± 0.001	Pinch strength of the left hand
0.100 ± 0.001	Speed of the right hand and right fingers
0.077 ± 0.001	Grab strength of the left hand
0.050 ± 0.001	Speed of the left hand and left fingers
0.014 ± 0.002	Radius of the circle gesture
0.001 ± 0	Speed of the circle gesture
0 ± 0	Key tap, swipe, and screentap gesture properties
0 ± 0	Length of left and right thumb metacarpal bones

Fig. 6. Average merit for all features for the continuous data. As with the merits calculated in a static context, physical features have a higher merit than all other gesture properties, such as finger speed and hand rotation.

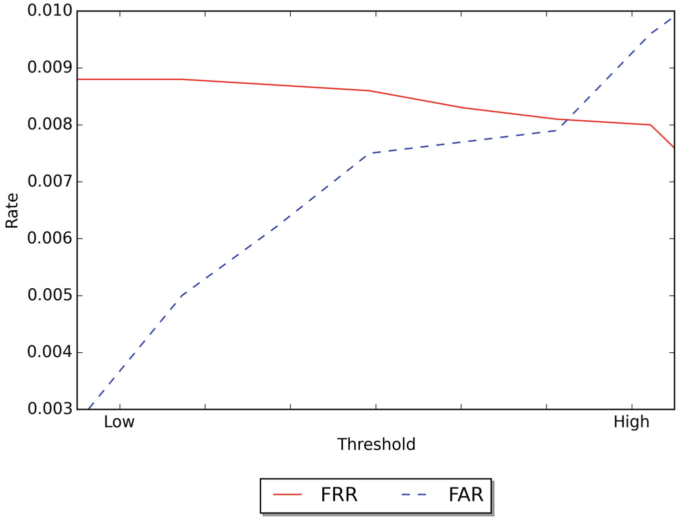

Fig. 7. A plot of error rates as the threshold changes. As the threshold increases, more values are accepted, which increases the false accept rate, but decrease the false reject rate. The reverse is also true for as the threshold decreases. The EER is the point at which the FAR and FRR are equal. Biometric systems aim for the lowest possible EER. Here, the EER is around 0.8 %.

7 Discussions and Conclusions

Based on results of the classification, it is obvious that the Leap Motion, in conjunction with a random forest classifier, is able to distinguish individuals with high accuracy. The ability of the Leap Motion controller to reliably collect data at all times, even in cases where parts of the hand were occluded, shows the improvement of the Leap Motion's software. Therefore, many earlier claims of poor recognition are likely no longer valid. While more rigorous testing must be done, it is safe to assume that the Leap is more capable of complex tasks such as sign-language or handwriting recognition.

In terms of authentication, it is not gestures that matter, but instead simple hand geometry. This is the case for both static, where the only gesture is a circle, and continuous authentication, where there was a greater number of gesture attributes. Because of the greater significance physical properties have, it can be concluded that classification based on physical properties alone is sufficient. Removing gesture attributes will increase the efficiency of the classification while preserving the accuracy, improving performance in a real world scenario.

In both parts, the lengths of the thumb metacarpal bones had no significance whatsoever. This is explained by the fact that the Leap API creates a thumb metacarpal bone, although one does not exist. It assigns this bone a length of 0, and so the length of this bone was the same for all participants. Not only do the lengths of the thumb metacarpals have no weight, but the properties of the key tap, swipe, and screentap gestures also have no weight. This is most likely due to the fact that such gestures did not occur often enough to be reliably used in identification.

Template matching revealed an EER of 0.8 %, while only using two attributes. This EER is lower than most fingerprint recognition algorithms, with most having an EER of between 2 % and 4 % [10]. Using more attributes may lower the EER even further, providing for heightened security, but at the cost of computational efficiency. As testing against only two attributes reveals such a low EER, it is impractical in a real-world scenario to use more attributes.

Future work will include focusing specifically on the four recognized gestures, ignoring hand geometry to either confirm or deny the ineffectiveness of these gestures in authentication. In addition, it might be possible to introduce a "knowledge" component to gesture authentication, that is, to have each individual user authenticate using a unique gesture, while simultaneously tracking of hand biometrics. Such a two-factor system could dramatically increase security.

References

1. Chan, A., Halevi, T., Memon, N.: Touchpad input for continuous biometric authentication. In: De Decker, B., Zúquete, A. (eds.) CMS 2014. LNCS, vol. 8735, pp. 86–91. Springer, Heidelberg (2014)
2. Roy, A., Halevi, T., Memon, N.: An HMM-based behavior modeling approach for continuous mobile authentication. In: IEEE Conference on ICASSP, pp. 3789–3793 (2014)
3. Weichert, F., Bachmann, D., Rudak, B., Fisseler, D.: Analysis of the accuracy and robustness of the leap motion controller. Sensors **13**, 6380–6393 (2013). doi:10.3390/s130506380
4. Jakus, G., Guna, J., Tomažič, S., Sodnik, J.: Evaluation of Leap motion controller with a high precision optical tracking system. In: Kurosu, M. (ed.) HCI 2014, Part II. LNCS, vol. 8511, pp. 254–263. Springer, Heidelberg (2014)
5. Angulo, J., Wästlund, E.: Exploring touch-screen biometrics for user identification on smart phones. Priv. Identity Manage. Life **375**, 130–143 (2012)
6. Potter, L.E., Araullo, J., Carter, L.: The Leap Motion controller: A view on sign language. Griffith University (2013)
7. Bača, M., Grd, P., Fotak, T.: Basic principles and trends in hand geometry and hand shape biometrics. In: Dr. Jucheng Yan (ed.) New Trends and Developments in Biometrics. InTech (2012). doi:10.5772/51912
8. Frank, M., Biedert, R., Ma, E., Martinovic, I., Song, D.: Touchalytics: on the applicability of touchscreen input as a behavioral biometric for continuous authentication. IEEE Trans. Inf. Forensics Secur. **8**(1), 136–148 (2012)
9. Sae-Bae, N., Ahmed, K., Isbister, K., Memon, N.: Biometric-rich gestures: a novel approach to authentication on multitouch devices. In: Conference on Human Factors in Computing Systems, pp. 977–986 (2012)
10. Cappelli, R., Maio, D., Maltoni, D., Wayman, J., Jain, A.: Performance evaluation of fingerprint verification systems. IEEE Trans. Pattern Anal. Mach. Intell. **28**(1), 3–18 (2006)
11. Bulatov, Y., Jambawalikar, S., Kumar, P., Sethia, S.: Hand Recognition System Using Geometric Classifiers, DIMACS Workshop on Computational Geometry (2002)

http://www.springer.com/978-3-319-20375-1

Human Aspects of Information Security, Privacy, and Trust
Third International Conference, HAS 2015, Held as Part of
HCI International 2015, Los Angeles, CA, USA, August 2-7,
2015. Proceedings
Tryfonas, T.; Askoxylakis, I. (Eds.)
2015, XVIII, 719 p. 206 illus., Softcover
ISBN: 978-3-319-20375-1