Continuous user authentication in mobile device and IoT environment

Liu, Allen Yang

2020

Liu, A. Y. (2020). Continuous user authentication in mobile device and IoT environment.
Master's thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/143089>

<https://doi.org/10.32657/10356/143089>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0
International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 21:18:07 SGT

CONTINUOUS USER AUTHENTICATION IN MOBILE DEVICE AND IOT ENVIRONMENT

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Master of Engineering

ALLEN LIU YANG

School of Computer Science and Engineering
Nanyang Technological University
yliu051@e.ntu.edu.sg

2020

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

19 Jan 2020

Liu Yang

.....
Date

.....
Liu Yang

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare that there is no plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

19 Jan 2020

.....
Date

.....
Ng Wee Keong

Authorship Attribution Statement

This thesis **does not** contain any materials from papers published in peer-reviewed journals or from papers accepted at conferences in which I am listed as an author.

19 Jan 2020

.....
Date

.....
Liu Yang

Acknowledgments

I would like to first of all express my deepest thankfulness to Professor Ng Wee Keong, for his great and illuminating guidance and advice during my whole academic period. He has been willingly supporting me in terms of time, effort and expense for my research projects. I really have gained a lot of benefits from what he gave me in terms of spiritual and academic knowledge.

Secondly, I would also want to send my sincere gratitude to school of computing, for their generous support of all those advanced and well-developed facilities and opportunities over the past three years.

Last but not least, I gratefully acknowledge the support of Nanyang Technological University with guidance and supervision for my study and research work.

Table of Contents

Statement of Originality	2
Supervisor Declaration Statement	3
Authorship Attribution Statement	4
Acknowledgments	5
Summary	12
Keywords.....	13
<i>I. Introduction.....</i>	<i>14</i>
A. Motivation.....	14
1) Disadvantages of existing authentication	14
a) One time access control.....	14
b) Risk of being hacked.....	14
c) Inconvenience & Intrusiveness.....	14
2) Continuous authentication	15
B. Objective	15
1) Continuous authentication	15
2) Continuous authentication advantages	15
a) Continuous access control	15
b) Low risk of being hacked.....	15
c) Convenience & Non-intrusiveness.....	16
C. Challenges in continuous authentication	16
1) Continuous authentication is expected to run in near-real-time.	16
2) We should select features in a context-aware and complementary manner.	16

3) It's not necessary to root the devices for continuous authentication.....	16
4) Evaluating continuous authentication scheme in an uncontrolled environment is required.....	17
5) The mechanism should be able to recognize mimicry attacks.....	17
D. Scope	17
E. Contributions.....	18
F. Report organization	18
II. Related work	19
A. Conventional authentication approaches.....	19
1) Overview	19
2) Approaches.....	19
a) Secret knowledge-based approach	19
Personal identification (PIN), password and passphrase	19
Cognitive knowledge question.....	20
Patterns and graphical passwords	20
b) Token-based approach.....	20
3) Summary.....	21
B. Continuous authentication	21
1) Overview	21
2) Approaches.....	21
a) Keystroke Dynamics	21
b) Gait Pattern for User Identification.....	22
c) Use Patterns	22
3) Weakness of the approaches by other researchers	23
III. Continuous authentication on mobile device.....	24
A. Approach.....	24

B.	Experiment	25
1)	Data collection	26
2)	Data augmentation	29
3)	Implementation & Feature extraction	31
a)	Package importing.....	31
b)	Data loading.....	31
c)	The combination of training and testing dataset	32
d)	The feature datatype conversion	32
e)	Dummy variables generation for categorical features.....	32
f)	Missing values	32
g)	Normalization	33
h)	Restoring the training and testing dataset by splitting the data frame.....	34
4)	Classification.....	35
a)	Performance	35
b)	Random forest.....	36
c)	Ensemble learning	38
5)	Adversarial model	39
C.	Result	39
D.	Application development	39
IV.	<i>Contextual authentication in iot environment.....</i>	42
A.	Approach.....	42
B.	Sensor system and dataset.....	43
C.	Algorithm and performance	44
1)	Center vectors	45
2)	Weights.....	45
3)	SOM map grid size & Basis function width	46

D. Results.....	48
E. Transfer learning.....	48
1) Advantage of transfer learning.....	48
2) Implementation analysis	48
V. Conclusion & Future work.....	50
VI. References	51

List of Figures

Figure 1: MainActivity.java	26
Figure 2: Variables for the touch event values.....	26
Figure 3: Variables for the sensor values.....	27
Figure 4: Data for storing in the SQLite database.....	27
Figure 5: SQLite database creation	28
Figure 6: Data collection activities for test subjects.....	28
Figure 7: Touch patterns of test subjects	29
Figure 8: Training and decision process.....	35
Figure 9: Random forest classifier	38
Figure 10: Ensemble learning.....	38
Figure 11: Fetching data from firebase.....	40
Figure 12: Calling the machine learning models.....	40
Figure 13: Machine learning testing	41
Figure 14: Security alert.....	41
Figure 15: Authentication process	43

List of Tables

Table 1: Report organization	18
Table 2: Weakness of the approaches by other researchers	23
Table 3: Package and functionality	31
Table 4: Accuracy and speed performance of machine learning models	36
Table 5: Parameter tuning	47
Table 6: Performance of Radial Basis Function neural network and Self- Organizing Map neural network	47

Summary

The conventional authentication methods, such as password authentication, fingerprint recognition and face recognition, have many drawbacks and disadvantages. The conventional authentication methods are snapshot and one-time authentication with a very high risk of being hacked. In the early days, the hackers can cause \$2,300 in fraud losses every hour by phishing or other hacking methods. The password leak and account takeover are serious security problems in financial industry.

We aim to create continuous authentication methods in mobile devices and IoT environment which allows user to authenticate without the burden of credential requests. Thus, creating a secure, user-friendly experience for the user. We have proposed two methods to authenticate users in two different scenarios, which are mobile device environment and IoT environment. Both of the two methods are proposed to authenticate user in the background by learning their behaviours, and building their unique profiles. And machine learning techniques are applied for information discovery from the behaviour data to make authentication decisions.

Keywords

Machine learning; Continuous authentication; Sensors; Smartphone; Seamless authentication; Implicit authentication; Internet of things; IoT; Big data; Transfer learning; Adversarial attack; User profile

I. INTRODUCTION

A. *Motivation*

Authentication is the key step to ensure security for all kinds of transactions. In order to make authorization to function well, which in turn maintain integrity and confidentiality, we must achieve accurate authentication of various users. Fundamentally, an authentication method can be classified into one of the three categories: something you know, something you have and something you are. 'Something you know' refers to the knowledge of owner that can be used for authentication. 'Something you have' refers to the physical stuff. 'Something you are' mainly refers to the biometrics related to human characteristics.

Here is some examples for each authentication category.

- Something you know: Password, Security question
- Something you have: Access card, Key, Hardware token
- Something you are: Fingerprint, Iris, Palmprint, Face

1) *Disadvantages of existing authentication*

While many forms of authentication exist such as numerical and alphabetical passwords, one-time password (OTP), graphical pattern passwords, hardware tokens and biometric authentication (face, fingerprint and iris recognition), they all have various disadvantages.

a) *One time access control*

All the existing authentication mechanisms provide traditional one-time access control. After the one-time access control is passed, there is no more authentication for the data access. Thus, if the intruder gets through the one time authentication, he will be able to get all the data in the system.

b) *Risk of being hacked*

In addition to it, existing authentication mechanisms have the risk of being hacked owing to the vulnerability to various attacks such as key-loggers, shoulder surfing attack, phishing scams.

c) *Inconvenience & Intrusiveness*

Nowadays, mobile devices are quickly gaining in popularity, more and more people are relying on mobile devices, particularly mobile phones, as their primary source of Internet access for basic operations. Mobile phone use is composed of short and frequent sessions, and it is very inconvenient for users to perform intrusive authentication for every short session.

We can notice that people start to use hardware token as part of the authentication process to have higher assurance. This practice is the mainstream within enterprises

and is slowly entering the consumer market. However, this method still has relatively higher expense and worse usability.

2) *Continuous authentication*

Rapid increase of online applications and services requires a more convenient and reliable method of authentication. Continuous authentication is the desirable authentication method which is able to overcome the abovementioned problems.

B. *Objective*

The objective of this research project is to develop methods of recognizing and validating the identity of users based on behaviour analysis, so that the security and usability limitations of current authentication can be solved [5].

The authentication methods based on behaviour analysis are known as continuous authentication. We are stimulated to develop the continuous authentication mechanisms which can recognize a user by using unique patterns of device use that are discovered from the data collected from the device user without requiring explicit actions.

1) *Continuous authentication*

For general authentication needs, continuous authentication can play a role as a second factor and supplements password authentication to have higher assurance in a non-intrusive manner. And it can be applied for fraud detection to provide more protection for business transactions such as online purchases [5].

IoT “Internet of Things” devices are on a rise. IoT is predicted to have billions of connected devices in near future and just about every appliance and device will be connected to a network. IoT devices are able to capture a great number of behaviour data about users in the IoT environment. Thus, the behaviour-based authentication methods can also be applied in the IoT environment.

2) *Continuous authentication advantages*

a) *Continuous access control*

Continuous authentication collects the data and makes authentication decisions continuously instead of providing a one-time access control.

b) *Low risk of being hacked*

Since continuous authentication is collecting the user behavioural data continuously, even intruder can pass some access control, it doesn't mean the intruder can keep accessing all the data continuously. The access can be banned anytime if the system observes any strange behaviour.

c) Convenience & Non-intrusiveness

Continuous authentication does not require users to perform any explicit actions. All the operations are performed at the background. Thus, continuous authentication is a non-intrusive method.

C. Challenges in continuous authentication

In this challenge section, we will focus on how we are going to carry out data collection and the continuous authentication plan based on the massive data we have collected. And next will be focus on how to perform the implicit authentication on Android devices.

1) Continuous authentication is expected to run in near-real-time.

Among all the running timing, the slowest process is running training scenarios, which can take more than 60 seconds sometimes. However, this will not be a huge concern, since running training scenarios is one-time usage, and can be achieved via background environment. The accuracy of continuous authentication can be quite high, so this method can be a competent alternative as non-primary authentication. Last but not least, the continuous authentication has set up mechanism to prevent usage by other people instead of the owner, and this mechanism can successfully detect out any abnormality like misuse in a short time.

2) We should select features in a context-aware and complementary manner.

From all the data we collected, it is necessary to be extra cautious to determine what behavioral features will be selected and what context should be assigned. Usually, in order to categorize the data from onboard sensors better, the user's actual situation, e.g., what is the app that the user is using, whether the user is still making movements [45]. However, these information may not be the suitable tool to distinguish the information. It is not suggested to use single source of behavioral data to construct continuous authentication scheme.

3) It's not necessary to root the devices for continuous authentication.

One concern from Android system is that it prohibits those background applications to collect any input information since collection of personal information may have security problems. In this case, the root authentication is needed for those continuous authentication plans that is dependent on implicit information to collect the input data. However, for those continuous authentication plans not relying on implicit input event, no root authentication is needed [6]. However, when talking about individual apps, those implicit input information can be captured and gathered from those apps and even don't need any further permissions. In this way, the protection of continuous authentication plan is executed at app level, not at device level. Take an instance, if the continuous authentication plans are tied within their apps instead of device, they will make sure those shared data can be protected in a good confidential level. By comparing with bundle continuous authentication at device system level, the restrictions will be compromised if applying continuous authentication at app level, and it also has a bigger space for improvement and development. So every time where

there is any latest continuous authentication scheme proposed, it is needed to take all those relevant potential problems into account.

4) *Evaluating continuous authentication scheme in an uncontrolled environment is required*

Sometimes, those continuous authentication scheme proposals may come up with unrealistic evaluations, which requires repeating tasks multiple times in lab condition to product information and data. However, once these continuous authentication schemes are put into real-world practice, those evaluations that are executed in the lab condition may be not applicable and thus will influence the scheme's result and performance. It is proved that the error rates between real-world datasets and data sets generated in lab condition and published in papers vary in a significant extent, where error rate of real world is much higher. Commonality is observed among a variety of continuous authentication schemes with published datasets availability [7].

To elaborate even further, there is a possibility that the people other than the owner of a smartphone device without guardianship can successfully log on to the device and get access to the personal information and data of the owner. So the location filtering feature may no longer be useful. Among all the device users, the percentage of concerning about their devices may be approachable to their family members and friends is also high. Some features like contacts and locations may find it not useful nor helpful enough inside the continuous authentication schemes [7]. In this way, to take all those device users' concern about confidentiality that may be encountered in real world into consideration becomes indispensable to grant them a better sense of security.

5) *The mechanism should be able to recognize mimicry attacks*

Besides inherent behaviour conflicts demonstrated previously, there also exists the collisions between keystroke and swipes which have been trained on purpose. There are two argument points we would like to state. The first is that continuous authentication is more suitable to be applied as a secondary authentication approach that can help cover the function of primary authentication approaches. The second argument is that the behaviour features should be collected from a variety of sources instead of individual sources [50]. Since by doing this, the challenges of devices trying to imitate human features can be increased greatly, and the level of complicity will also increase, which can cause greater confusion to the behavioural models.

D. *Scope*

- In the first phase of our research, I designed the continuous authentication on the mobile device using touch data and data of built-in sensors.
- After I got the reasonable results, I proposed another continuous authentication method in the IoT environment using sensor data generated from the wireless sensor network.

E. Contributions

I have improved the performance of continuous authentication by applying machine learning on a combination of the touch data and sensor data. We have used machine learning extensively in our research and machine learning has helped us increase the authentication accuracy to a great extent.

The continuous authentication method that I proposed for IoT environment is named as contextual authentication. This method gives out a very good design based on the sensor data in the IoT sensor network. Machine learning is used in this method for making authentication decisions as well. However, the big IoT data usually has various feature spaces, so transfer learning is of great significance to the IoT research, because transfer learning is able to learn from data belonging to different feature spaces.

F. Report organization

The organization of this report is stated clearly in the following table

Section	Content
Introduction	<ul style="list-style-type: none">• Motivation• Objective• Challenges• Scope• Contributions• Report organization
Related work	Literature review: <ul style="list-style-type: none">• Conventional authentication approaches• Continuous authentication approaches
Continuous authentication on mobile device	<ul style="list-style-type: none">• Approach• Experiment• Result• Application development
Contextual authentication in IoT environment	<ul style="list-style-type: none">• Approach• Sensor system and dataset• Algorithm and performance• Transfer learning
Conclusion & Future work	<ul style="list-style-type: none">• A brief summary of the research work• Discussion about the future work on continuous authentication

Table 1: Report organization

II. RELATED WORK

Nowadays there is a continuously increasing tendency for individuals, businesses and governments to carry out activities, programs or projects via internet and a variety of internet-capable electronic devices. However, the maturity of cybersecurity seems to fail to catch up with the popularity of online activities, making these online activities exposed and vulnerable to cybercrimes. In this way, it has become a top priority to ensure that user is granted to approach to devices legally.

I have reviewed the literature about both conventional authentication approaches and continuous authentication approaches.

A. *Conventional authentication approaches*

1) *Overview*

Here the concept of authentication is the first defense wall intended to guarantee that only authenticated user will be accessible to the device. However, by tracing back, the authentication still can induce a variety of feasibility and security relevant issues. Those conventional authentication's functions can only perform in a preliminary level. And re-authentication execution may act in an invasive way.

2) *Approaches*

a) *Secret knowledge-based approach*

The so called secret knowledge-based approach mainly refer to any secret which is only supposed to be owned and known to authenticated users, which includes PIN, password, answer to a question, or characteristic pattern. These secrets may be generated by the system itself initially and can be later set or updated by the supposed authenticated users. Therefore, the secret is shared between the device authentication system and the user. In order to get access to the device, the information provided by the user and the information stored inside the device system must match exactly. In this case, the result will only be two conditions, either device will allow user to access if information match exactly, or device will decline user to access if information is not matched exactly. The nature of this secret knowledge-based approach will need human to memorize the secret exactly disregarding of the secret's complicity, length and irregularity [31]. What's worse, the secret can be passed or figure out by other un-authenticated users, since it is not an inherent nature of the owner.

Personal identification (PIN), password and passphrase

PIN secret is known as the most easily set knowledge-based authentication technique. It is commonly used in mobile devices to unlock their smartphone, or to authenticate the permission of subscribing and buying SIM card via credit cards. The length of the PIN will normally vary from 4 to 8 numbers merely. On one aspect, PIN is easy to

remember and execute, so on the other aspect, it is not safe as other people may easily figure out your PIN. Another secret approach, password, normally will be longer and more complicated. Password is normally identified as a useful way as an authentication method if set and utilize it properly.

Cognitive knowledge question

This method usually will ask user to set a variety of questions with defined answer first. Then those sets of questions and answers will be stored inside the authentication system. Those questions are closely related to personal information and experience, like your mother's surname, your born city and hobby, etc, aiming to relieve the burden for users to memorize the irregular numbers and characters compared to password. However, the limitation from this method is that it is lack of confidentiality, since personal information can be dug out through a variety social media unconsciously and thus it is not hard to figure out the correct answer by other people.

Therefore, this authentication obviously is not qualified as a sole and determining method as authentication system. Normally, in order to ensure better confidentiality, multiple cognitive knowledge questions instead of a single question method is utilized, or the cognitive knowledge question method is acted as a secondary authentication method in combination with other authentication method like password etc [9]. By using these two methods, the level of privacy and safety indeed increases, but this also insert heavier burden to users since they need longer time to compose multiple cognitive questions and every time when accessing, will need longer time to figure out all these answers. So this method is most suitable as a secondary authentication that can be used to strengthen the security level of alternative secret-knowledge methods. It is also very suitable to be used as the gating factor for user to reset their password if they forget their old password, or it can also appear and put into use when users are locked due to multiple attempts to log in to the device.

Patterns and graphical passwords

There are some discussions over the corner to suggest another way that can avoid or minimize the limitations of PIN, password and passphrase, which are in form of text and words [9]. One way is to use patterns or graphical passwords. This is based on the theory that human brain is better at memorizing information in terms of pattern or graphs compared with texts and words. And this results in the emergence of graphical password. In details, the graphical password is a predefined pattern by the owner based on a nine (3*3) dots grid. It is suggested that the graphical password will be more convenient for user to memorize and extract from brain compared to alphanumeric password. Also this can reduce the burden for the user since it is less cognitive and more easy to remember. This graphical password system is commonly used in Android devices to make sure secret can be remembered in an exact way.

b) Token-based approach

The last method has emerged in order to mitigate the downside of those secret knowledge-based approach mentioned previously. And it is called token-based approach. This is a widely used method including logical access, physical access to services and systems. The token-based approach can be classified as two kinds: Hardware and software tokens if taking their appearance and necessity of another

physical devices [33]. The hardware token will introduce another physical device. And this is commonly seen in bank system. The software token doesn't need to provide an extra device. Instead, a one-time-password may be sent to the user's phone via SMS, or certain apps is needed to be installed where can receive a similar verification code like a one-time-password.

3) *Summary*

Among all the conventional authentication methods, they indeed helped to increase the level of security. However, all the extra information or so called secret and extra hardware device or software apps have exerted burden to the user, and the level is still preliminary. By using these conventional method, it will not be feasible to re-authenticate the user regularly because of its invasive nature. Thus, there is an emerging need to invent a more advanced, burden-free and confidential authentication approach.

B. Continuous authentication

1) *Overview*

A few continuous authentication(CA) mechanisms have been proposed by researchers which employ different behavioural features and provide reasonable authentication accuracy, including keystroke authentication, gait pattern identification, phone event scoring and touch data based authentication [3][4]. Through the literature review, we have evaluated the CA schemes in terms of: accuracy, processing time and detection speed, processing complexity, privacy preserving [3][42]. While these CA frameworks work in principle, it is critical to understand from these research works which schemes have better performance and compatibility under different conditions. In view of the accuracy and privacy protection, we chose to use touch data generated by user interaction with the mobile device touchscreen to build an authentication framework. And we also proposed a Contextual Authentication method in IoT environment.

2) *Approaches*

a) Keystroke Dynamics

The concept of keystroke dynamics is a promising detecting system to recognize a unique user on keyboard-based device [2]. It used the data collected based on user's typing characteristics, and this can be achieved by asking user to type an appointed sentence, or by taking samples of a user's typing characteristics when they are typing on keyboard [12][13]. This keystroke dynamics method to identify one user since each individual user will have different typing patterns, can be traced back to a very long time period. This concept, categorized as a behaviour biometric, was first come up by Spillane in 1975 [10]. From then on, more and more relevant research have been conducted by different researchers. There are researches to verify if it is trustworthy to be functioned as an identifier on both computer and electronic mobile devices. Initially, the method to check whether one user's keystroke characteristics will be identical to the database stored previously is by using statistical categorizers [17]. But now the most updated way is to use neural networks. The neural networks is known to show less number of characters needed to recognize a unique user.

Nevertheless, it is still suspecting whether keystroke dynamics on its own is sufficient to recognize a unique user, even though this keystroke dynamics have quite amount of information to allow access for identify authentication. The biggest advantage of keystroke dynamics lies in that typing is the action that most users will do in order to use the devices, e.g. searching a web site, posting a moment etc. So there is no extra effort from the keystroke dynamic system to collect information elsewhere [11]. So it is highly promising that by integrating with other biometric features, e.g. voice system, into one integrated system, this integrated system will be able to provide a comprehensive set of features or patterns to recognize and thus authenticate an unique user.

b) Gait Pattern for User Identification

By reviewing multiple research journals, a lot of authors have suggested that gait patterns can be a new way to identify smartphone users. Frank et al [14] have allowed their database system and carry out method to be approachable to publics. In this way, it will be easier to verify their performing results by regenerate again. The more exciting thing is that they claimed that compared to other gait pattern based topic, they can produce higher precise result greatly. In Frank's proposal, there is a so-called time-delay embedding method to figure out the gait. This method mainly focus on by observing the system taken over time, the state of a not-known dynamical system can be rebuilt up. First of all, the time-delay embedding method can obtain the first hand features, then these first hand features will be applied with noise-reduction activity by the usage of principal component analysis (PCA) [14] based on training data's short embedding system. The principal component analysis can convert from the time-delay embedding region to a model region with lower dimension. After all these have been done, an ANN classifier will utilize the acquired features.

c) Use Patterns

Use pattern is a concept that is used to gather information based on the interaction between user and computing system, including mobile device, laptop, and desktop computer. These series of information will as a whole generating the uniqueness of a user, and thus can be used to identify different user biometrically. Examples of use patterns can be interpreted as what web sites a user usually visited, to and from whom the user makes a call or send SMS, what is the most frequent type of music that the user listen to, and what are the APPs that the user normally downloaded [44]. All these information provides a very detailed and oriented description of a user, which should be more than sufficient to lock down on one user. Because the likelihood of two people performing exactly same actions in the mobile device as mentioned above is very low.

Behavioural biometrics is also aroused high interested in use pattern study. Clarke et al has come up with the concept as 'service utilization', from this concept, it may be likely to figure out what is the attitude that the user treat to authenticate themselves on mobile devices. However, this service utilization hasn't been tried in a currently working system [15]. Although it is quite straightforward that by monitoring how a user users his or her mobile devices, there will be clear clue to recognize and define one user, this research is not well-studied, especially when discussing about mobile device aspect. It is still promising that if we integrate other identifying systems including keystroke dynamics and voice detectors into one entity, the error rate can be further

decreased and the result will be more accurate since we can collect an even more comprehensive set of personal data to define one user.

3) *Weakness of the approaches by other researchers*

The existing continuous authentication methods have various weaknesses so the researchers in this field are still supposed to come up with new techniques to make continuous authentication feasible and applicable in real life.

Approaches	Weakness
Keystroke Dynamics	Privacy violation
Gait Pattern for User Identification	Low accuracy
Use Patterns	Long latency

Table 2: Weakness of the approaches by other researchers

III. CONTINUOUS AUTHENTICATION ON MOBILE DEVICE

A. *Approach*

This approach is designed to collect data silently and transparently at the background of mobile devices. In order to protect the privacy of the user, we store the collected data on the mobile device instead of transmitting to the server. There are various data sources that can be used for authentication, such as application usage information, keystroke logs, browser history, short messages, etc [41]. In our approach, we select the touch data generated by touchscreen when user interacts with the smartphone. Different users have different touch behaviour and patterns, so we design the approach to investigate the uniqueness of user touch data [16][40]. In addition to it, we also include sensor data of the mobile device in our research because a smartphone has a collection of various sensors. Specifically, we select 3 sensors which are accelerometer, gyroscope sensor and orientation sensor. We choose these 3 sensors because they can provide accurate information about the movement, vibration and rotation of the smartphone. In our work, we designed and conducted data collection experiment with test subjects to collect the touch data and sensor data. Once we have the required data, we are supposed to do data pre-processing, data normalization and feature engineering to generate high quality data for machine learning algorithms. Then, a classification algorithm is implemented to learn the user behaviour and make authentication decisions. Ideally, this approach can be non-intrusive because the algorithm verifies whether the current user is owner or guest in the background.

B. Experiment

1) *Data collection*

In the experiment, we developed an Android application as data collector, because Android system provides powerful API about touch data and sensor data. Here are the details about the Android application for data collection. I have modified the MainActivity.java file so that the data collection will be triggered once there is touch event happening.

```
@Override
public boolean onTouch(View v, MotionEvent event) {
    hideKeyboard(false);
    collectionData.getCollectionData(this,event);
    Log.v("ltl","ontouch -----");
    return mGestureDetector.onTouchEvent(event);
}
```

Figure 1: MainActivity.java

I have created a set of variables to store the data values. In the end, all the collected data have been stored in the SQLite database in the Android phone.

```
x = event.getX();
y = event.getY();
xRaw = event.getRawX();
yRaw = event.getRawY();
pressure = event.getPressure();
areaCover = event.getSize();
pointerCount = event.getPointerCount();
actionCode = event.getAction();
actionCodeMasked = event.getActionMasked();
actionIndex = event.getActionIndex();
```

Figure 2: Variables for the touch event values

```

@Override
public void onSensorChanged(SensorEvent sensorEvent) {
    if(sensorEvent.sensor.getType()==Sensor.TYPE_ACCELEROMETER){
        accValues=sensorEvent.values.clone();
        gx = accValues[0];
        gy = accValues[1];
        gz = accValues[2];
    }

    if(sensorEvent.sensor.getType()==Sensor.TYPE_ORIENTATION){
        orientationValues=sensorEvent.values.clone();
        azimuth = orientationValues[0];
        pitch = orientationValues[1];
        poll = orientationValues[2];
    }

    if(sensorEvent.sensor.getType()==Sensor.TYPE_GYROSCOPE){
        gyroscopeValues=sensorEvent.values.clone();
        sx = gyroscopeValues[0];
        sy = gyroscopeValues[1];
        sz = gyroscopeValues[2];
    }
}

```

Figure 3: Variables for the sensor values

```

ContentValues contentValues = new ContentValues();
contentValues.put("timeStamp",timeStamp);
contentValues.put("x",x);
contentValues.put("y",y);
contentValues.put("xRaw",xRaw);
contentValues.put("yRaw",yRaw);
contentValues.put("pressure",pressure);

contentValues.put("areaCover",areaCover);
contentValues.put("actionCode",actionCode);
contentValues.put("actionCodeMasked",actionCodeMasked);
contentValues.put("actionIndex",actionIndex);

contentValues.put("pointerCount",pointerCount);
contentValues.put("fingerOrientation",fingerOrientation);
contentValues.put("screenOrientationStr",screenOrientationStr);
contentValues.put("azimuth",azimuth);
contentValues.put("pitch",pitch);
contentValues.put("poll",poll);
contentValues.put("gx",gx);
contentValues.put("gy",gy);
contentValues.put("gz",gz);
contentValues.put("sx",sx);
contentValues.put("sy",sy);
contentValues.put("sz",sz);

db.insert("collection",null,contentValues);
new DBHelper(context).openDatabase();

```

Figure 4: Data for storing in the SQLite database

The codes below show how the SQLite database is created.

```
public class CollectionDataStoreSQLite extends SQLiteOpenHelper {  
    private static final String DBNAME = "collection_data.db";  
    private static final int VERSION = 11;  
    public CollectionDataStoreSQLite(Context context) {  
        super(context, DBNAME, null, VERSION);  
    }  
  
    @Override  
    public void onCreate(SQLiteDatabase sqLiteDatabase) {  
        String sql = "create table collection(timestamp float PRIMARY KEY,x float,y float,xRaw float,yRaw float,pressure float," +  
            "areaCover float,pointerCount float,actionCode float,actionCodeMasked float,actionIndex float,fingerOrientation float," +  
            "azimuth float,pitch float,poll float,gx float,gy float,gz float,sx float,sy float,sz float);";  
        sqLiteDatabase.execSQL(sql);  
    }  
  
    @Override  
    public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int ii) {  
    }  
}
```

Figure 5: SQLite database creation

And I have designed a list of reading comprehension quizzes, typing tests and games in the Android application for test subjects. Test subjects can try to play with some of the games in the application to participate in the data collection for our research.

Figure 6: Data collection activities for test subjects

Finally, we recruited 10 test subjects to take part in the data collection and the duration of experiment session per person is 50 minutes. In each session, the test subject is free to play with the games within the application. Finally, we collected 7000 - 9000 data points for each test subject.

We have visualized the touch data patterns of test subjects using touch coordinates and timestamp. Some visualization examples are shown below. The different colours of touch points represent the timestamps.

Figure 7: Touch patterns of test subjects

2) *Data augmentation*

Currently, the most important method for extracting value from a large amount of data is machine learning. And machine learning has a very high requirement on the quality and scale of training data in order to achieve a high effectiveness. Nevertheless, for most of the researchers and engineers, it's very challenging to have the access to a great amount of training data in the training process of machine learning [24]. Thus, many researchers and scientists have used data augmentation techniques to increase the size of machine learning datasets. However, not all the augmentation techniques are able to generate data with high quality. At the very beginning, the researchers are expected to perform a comprehensive evaluation on the training data. Sometimes, there is too much noise in the training dataset and noisy data must be separated out automatically.

In my research work, I use machine learning classification algorithms to recognize the true user and perform the authentication in the background automatically. The raw data may not be able to meet the requirement of machine learning models. Thus, we always need to do a lot of feature engineering and data pre-processing work before the classification, such as feature datatype conversion, dummy variable generation, missing value processing and data normalization. For machine learning, the amount of data plays an very important role in the classification process. The more data we have, the higher accuracy we can achieve. Cross validation is usually applied to check the quality of training data. And it's always necessary to check the data pattern and perform the comparison for data distribution [25].

It has been viewed that data augmentation can inject prior insights on the constant properties of the data against some transformations. Data augmentation can help with machine learning on many aspects, including overfitting prevention, covering the unexplored input space, and the enhancement of model generalization ability

[25]. For computer vision research, we know the data labels won't be changed by the minor adjustments such as warping, scaling, jittering, cropping and rotating because they have a high chance to happen in real-world observations. But the sensor data is different from the image data, it's usually not easy to intuitively recognize the label-preserving transformations for sensor data.

Differences in sensor placement between test subjects can introduce the label-constant variability of sensor data. For example, a different sensor placement can change the sign of sensor data values with the constant labels. Thus, we can apply random rotations to the data that we have to perform the data augmentation and simulation of various sensor placements.

The temporal location of events in the window can introduce the variability as well, such as tremor. In the window, we can't get meaningful information from the location of the observed symptom because the segmentation of window is random and the size of window is fixed. Consequently, the window and event location can be perturbed to perform the data augmentation [24].

There are various methods to make the temporal location of events in the window get perturbed arbitrarily. We can use time-warping to perturb the temporal location. For time-warping, the time intervals between samples are distorted, so the temporal locations will be changed in this way.

In image processing, researchers like to use image cropping. We can use a similar cropping technique to reduce the event location dependency. An event free region can be captured by this cropping technique and the label may also be changed. Besides, we can get a convergence at a sliding window method with random sizes of stride if the cropping is with arbitrary locations over epochs.

Permutation is one of the easy ways to achieve our aim. Firstly, we are supposed to perform the segmentation on the data equally, usually into within 5 segments, in order to perturb the data location in the window. Then, the segments can be permuted arbitrarily to form the new window.

The labels may not be altered if there are tiny changes in magnitude. In a window, scaling (scale) can change the data magnitude through the multiplication by an arbitrary scalar. In contrast with that, magnitude-warping (MagW) is able to convolve the data window using a smooth curve oscillating around 1 and the sample magnitude can be changed with this method [20]. What's more, researchers like to use jittering (Jitter) to do the simulation for additive sensor noise. The above-mentioned techniques for data augmentation are able to enhance the robustness against additive noise and achieve a better performance.

We have introduced a few techniques for sensor data augmentation in this chapter, including rotating, time-warping, cropping, permutating, scaling, magnitude-warping and jittering. The machine learning model can have a better performance if the augmentation is applied in a proper manner. These approaches are very generic and are easily adaptable to various big data domains [24].

3) *Implementation & Feature extraction*

We have done data pre-processing on the collected data, such as data normalization, noise point deletion, etc. And we extracted a number of features from the raw data, including touch position coordinates, touch area size, finger pressure, timestamp, pointer count, action code, screen orientation, 3 sensor values of accelerometer, 3 sensor values of orientation sensor and 3 sensor values of gyroscope sensor. In addition, we have also derived the swipe features based on the action code and other features.

a) *Package importing*

Python is the language that has been used for this project. There are a variety of packages needed for this project, which are matplotlib, Pandas, Sklearn and Numpy. Matplotlib can be used in the area of plot and visualization; Pandas can be used to pre-process data; Sklearn is for machine learning algorithms and Numpy is for linear algebra. As we can see, we need to use different packages which have multiple functions to solve distinct problems.

Package name	Functionality
Matplotlib	Visualization and Plotting
Numpy	Linear algebra
Sklearn	Machine learning models
Pandas	Data pre-processing

Table 3: Package and functionality

b) *Data loading*

With the application of 'read_csv' function from Pandas, the data in the training and testing dataset can be extracted. The parameter 'index_col' is used to indicate which column can be used to label rows for the 2 data frames. So in this way, normally the columns that is able to label the row will be the first column with an index value of 0 in Data Frame. This special column does not have significant insights that can be used to build up the model for machine learning, and It is just used to specify the data records.

c) The combination of training and testing dataset

While we are doing data mining, the feature engineering must be the same for both training and testing dataset. Thus, testing and training datasets can have exactly same input variables. In this way, during the initial stage of the feature engineering, the 'load data' step will generate 2 data frames, which need to be combined. Whereas prior to the combination, the target variable from training data frame need to be dropped out and saved separately. As a result, once the 2 data frames have been combined together, another data frame can be generated where both training and testing data records can be found inside.

d) The feature datatype conversion

In the dataset, there are two kinds of features, one is continuous, and the other is categorical. As for categorical feature, it cannot be in the numerical form, and we need to change the numerical feature to string instead. The function 'value_counts()' can be applied to figure out how many data records are there for each category of the feature.

e) Dummy variables generation for categorical features

In order to build the machine learning models in this project, sklearn package is introduced and applied. One drawback regarding to sklearn is that it is not fit with categorical features. In this way, if dataset has categorical features, then a variety of dummy variables are needed to substitute categorical features. The corresponding relationship between dummy variable and category of categorical feature is one to one. By using the 'get_dummies()' function from pandas, the categorical variables can be substituted with dummy variables. Once the 'get_dummies()' has been performed, there will be additional columns generated compared to the initial data frame. Hence, this function implementation produces more features. So in this way, if there exists more than two categories in one feature, there will be only one dummy variable as 1 and the other dummy variable as 0. If the feature has missing values, they will set 0 for all the dummy variables.

f) Missing values

As we discussed in the last sector, since sklearn package does not accept categorical features, so we need to make sure all the categorical variables have been set to dummy variables, either 0 or 1. And the missing value in categorical feature also needs to be set to dummy variables and value should be kept as 0. However, besides missing values inside categorical feature, there are also missing values in continuous variables. 'NA' will be used to indicate those missing values, whereas sklearn does not accept 'NA'. In this way those missing values which are shown as 'NA' need to be converted to numerical values.

A variety of methods can be adopted to deal with those missing values.

- First is by simply treating the missing values as nothing.
- Second is by deleting those missing values inside the dataset.
- The third way, which is the most optimized way, is to guess and replace the missing value with real value through proper analysis. Recommended analysis method can be either based on past knowledge and experience, or based on mathematic statistic method, including mean value of the variable, most frequent appeared values, and also can be the reference of values that has similar condition as missing values.

In this project, we adopt the method by filling in the mean values of variable for the missing values. This method is carried out in the below procedures: first is to figure out in each feature, how many missing values are there. Then we calculate for each feature, what is the mean value. Next, the 'fillna()' function is used to substitute the missing values with the mean values for each feature. In this way, all the missing values will no longer exist since they have been replaced with the mean value.

g) Normalization

Once completed the actions mentioned above, now the data frame will no longer have any categorical values or missing values. Instead, numerical values is the only existing form. Apart from this, with the purpose of predicting result more precisely, data normalization will be needed.

Data normalization, also named as standardization, is to reduce the range and scale of the data features. Normally to perform data normalization, three methods are adopted [49].

- First is min-max normalization
- Second is Z-score normalization
- And third is normalization by decimal scaling

In this project, z-score normalization method is used. This method is implemented by using the old value to deduct by the feature mean value, then followed with dividing the obtained value by feature standard deviation. I prefer to use z-score normalization because z-score normalization takes into account both the mean value and the variability in a set of raw data, with the data range preserved. The mathematical formula for z-score normalization is shown as below [49]:

$$v_{new} = \frac{v_{old} - \mu_{old}}{\sigma_{old}} \quad \rightarrow \quad \mu_{new} = 0, \text{ and } \sigma_{new} = 1$$

As we previously discussed, the columns that contain dummy variables will only be 0 or 1. And these columns will not implement z-score normalization. We will only focus on the initial features which are continuous instead of categorical to

implement z-score normalization. To distinguish categorical features and continuous features, data type will be used. The data type of categorical feature is 'object'. So in this way we only need to filter out those features with data type of 'object', then we will only implement z-score normalization on continuous features.

Once finished z-score normalization, it is recommended to draw out the histogram in order to check the data range after z-score normalization. As what we can observe from the plot, it is clearly observed that range of the variables are centred around 0, and only few data points will have high leverage.

h) Restoring the training and testing dataset by splitting the data frame

By tracing back to the initial stage of feature engineering, the training and testing data frames are combined to one data frame. Now we have already finished the feature engineering process, and next step is to use training dataset to train machine learning model. In order to make sure there is a good fitting for the modelling, the originally combined data frames will need to split out.

4) *Classification*

Figure 8: Training and decision process

a) *Performance*

We have built 3 classification models on the data which are kNN, SVM and Random forest. Then, we run the 3 models on the workstation and the performance of the 3 models are shown in the table below

As shown in the table, the random forest algorithm has achieved a higher accuracy compared with kNN and SVM.

	kNN	SVM	Random forest
Classification accuracy	70%	78%	85%
Authentication accuracy with threshold 75%	52%	97%	99%
True rejection rate with threshold 75%	58%	94%	99%
Training time	60s	40s	40s
Inference time	Within 3s	Within 3s	Within 3s

Table 4: Accuracy and speed performance of machine learning models

b) Random forest

Here we need to introduce the concept of random forest, and it is a well-known and commonly used machine learning model. The reason why it is so popular lies in below statements.

- First of all it is one of machine learning algorithms that can achieve the highest accuracy.
- Second reason is that it is very efficient in processing on big scale dataset.
- Last but not least, while this model is being processed, it is able to produce an unbiased estimation on the error for the testing result.

In random forests, the general technique of bootstrap aggregating or bagging have been applied to tree learners through the training algorithm. Suppose we have a training set $X = x_1, \dots, x_n$ with $Y = y_1, \dots, y_n$, bagging would select a sample randomly and repeatedly (B times) with the training set's replacement and fits trees to these samples [22]:

For $b = 1, \dots, B$:

1. Sample n training examples from X, Y with replacement; call these X_b, Y_b .
2. Train a classification tree f_b on X_b, Y_b .

After the model is well trained, we can make the predictions for testing samples x' by taking the majority vote in the case of classification trees or by averaging the results from all the regression trees on x' :

$$\hat{f} = \frac{1}{B} \sum_{b=1}^B f_b(x')$$

The abovementioned process tells us the original bagging model for trees. The random forest has a slight difference from that because tree learning algorithm used by random forest is modified, and in the learning process, it selects a subset of the features randomly at each candidate split [21].

Below list out the procedures to perform the random forest algorithm [18].

- First step is to extract n_{tree} bootstrap samples from original data.
- Second step will have some modifications compared to the traditional method. Treat each bootstrap sample as a unit, and grow unpruned tree on top of it, either regression or classification. Different from traditional method, which consider all predictor as the range to choose the best split, this time we will need to choose samples extracted from the predictors randomly first, then followed by choose the best split in the range of those randomly selected samples instead of all predictors. Bagging is a special case of random forest, where the number of randomly selected samples equals to the number of all predictors.
- Third step is to combine all the n_{tree} trees prediction together for prediction. Normally for classification, the aggregation method is by most frequently appeared votes, whereby for regression tree is the mean value.

Apart from it, on the basis of training data, error rate estimation can also be acquired by using the methods below [19][20].

- First of all, when processing bootstrap iteration, the bootstrap sample along with the tree grown can be used to predict whether the data is in the bootstrap sample or not. If not, the data will be called as 'out-of-bag', also called as OOB data.
- Secondly, assemble OOB predictions by aggregation. Normally there will be 36% likelihood that OOB will appear, and this will be needed to take into consideration. The result calculated out is called out-of-bag estimation of the error rate.

The random forest classifier is used and this can be found from `sklearn.ensemble`. At this execution step, the difficulty pending to be solved is how to set the `max_features` parameter. This is because varying the value of `max_feature` parameters can have an impact on the testing error rate. One of the most important thing is to find the good-fit `max_feature` parameter that can yield highest testing accuracy. The testing accuracy, however, cannot be obtained in the training stage. Thus the concept of cross validation is introduced to have a rough estimation on testing error rate. The function `cross_val_score` from `sk-learn` is implemented to perform cross validation.

In this project, a variety of `max_features` parameter values are listed out, including 0.01, 0.1, 0.3, 0.5, 0.7, 0.9 and 0.99. And we set the `n_estimators` as 200, meaning that the random forest will have 200 trees in total. In order to obtain the testing error rate for each `max_features` value, a for loop function will be performed. On the training dataset, a 5-fold cross validation is implemented every time one loop is

executed. And we use a list called 'test_scores' to store each testing score. By plotting out the testing score versus max_features, it is clearly shown that when max_features values is 0.7, then it will yield lowest testing error rate and the best performance. Therefore, in this model, 0.7 is the optimal candidate for the best max_features value that yield highest testing accuracy rate.

```
### 2.3 Random Forest Classifier

max_features = [.01, .1, .3, .5, .7, .9, .99]
test_scores = []
for max_feat in max_features:
    clf = RandomForestClassifier(n_estimators=200, max_features=max_feat)
    # 5-fold cross validation
    test_score = cross_val_score(clf, X_train, y_train, cv=5)
    test_scores.append(np.mean(test_score))
print(test_scores)
```

Figure 9: Random forest classifier

c) Ensemble learning

It's suggested to use ensemble learning for machine learning prediction in order to increase the prediction accuracy. Ensemble learning aggregates the results of various base algorithms to get the final results. Thus, the selection of appropriate base models has a great importance. And aggregation methods are different for regression and classification.

Figure 10: Ensemble learning

However, for continuous user authentication, the devices will have a high risk of being hacked if the processing is slow and response comes late. Thus, we always try to simplify the algorithm to reduce the processing time. Application of ensemble learning will increase the algorithm complexity, so we haven't used ensemble learning to combine the results of different base algorithms.

5) *Adversarial model*

In the experiment, we have also set up the adversarial model to perform hacking. In this step, we asked the test subject to imitate the touch behaviour and phone handling behaviour of another test subject when playing with games in the application [43]. However, the algorithm is still able to tell the hacker from owner with a reasonable accuracy.

C. Result

Based on the classification algorithm performances, we can see all the 3 models have achieved a reasonable accuracy and the Random Forest has the highest accuracy of 85%. And the processing time of the algorithms is ranging from 40s to 60s. With higher performance machine, the processing speed can be significantly faster. Besides, if we can have a larger amount of user data, the algorithm accuracy can also be higher. Hence, this authentication approach can keep the mobile device more secure as a supplement to the traditional authentication method and help with the fraud detection on mobile device without any additional hardware.

D. Application development

The above-mentioned continuous authentication method on mobile device works well in the experimental environment. However, there are still challenges to overcome if we want to deploy this continuous authentication mechanism on the real mobile device. In order to prove our design is workable in the real-world environment, I have developed more applications to test our design.

I have used firebase SDK to upload the real-time data collected by the Android application to the firebase server, and the firebase server is set to be accessible. Then I wrote scripts to get the collected data from firebase database. Every time the number of data points in firebase reaches 200, the data points will be fetched and deleted from firebase server. The fetched data is stored in the local server for machine learning processing.


```

if result:
    print("The numebr of data records:", len(result.keys()))

    while (len(result.keys()) >= 200):
        features = list(result.values())
        # print(features)
        firebase.delete('/', None)

        import csv

        keys = features[0].keys()
        with open('testFirebase.csv', 'w') as output_file:
            # clear the existing content in csv file
            output_file.truncate()
            dict_writer = csv.DictWriter(output_file, keys)
            dict_writer.writeheader()
            dict_writer.writerows(features)

```

Figure 11: Fetching data from firebase

Every time the data is fetched from firebase and stored in the local server, machine learning model will be called to make prediction on the classification of the collected data.

```

print('Testing with Machine Learning...')
import machineLearning

machineLearning.main()

```

Figure 12: Calling the machine learning models

The feature engineering has been performed and the machine learning models have been well trained before it is used for prediction and testing. And the prediction can be done within 3 seconds.

```

rf = RandomForestClassifier(n_estimators=200, max_features=.7)
rf.fit(X_train, y_train)
y_rf = rf.predict(X_test)
y_final = y_rf
print(y_final)
#print(sum(y_final), len(y_final))
#print(sum(y_final)/len(y_final))

submission_df = pd.DataFrame(data={'Id': test_df.index, 'label': y_final})
submission_df.head(10)
submission_df.to_csv('submisison.csv', index=False)
print('Testing done')

if sum(y_final) / len(y_final) < 0.7:
    import sendEmail
    sendEmail.main()

```

Figure 13: Machine learning testing

After the prediction is done, the result of prediction will be compared with the threshold which is configured beforehand. If the user is classified as an intruder, a security alert will be sent out through email. If the user is classified as the device owner, the user will be allowed to continue to use the device without any security alert.

```

def main():
    import smtplib

    server = smtplib.SMTP('smtp.gmail.com', 587)
    server.starttls()
    server.login("demox@gmail.com", "audemo")

    msg = "Security alert!!!"
    server.sendmail("demox@gmail.com", "demox@gmail.com", msg)
    server.quit()
    print('Security alert has been sent out.')

if __name__ == "__main__":
    main()

```

Figure 14: Security alert

IV. CONTEXTUAL AUTHENTICATION IN IOT ENVIRONMENT

A. *Approach*

Authentication is the frontline countermeasure of ensuring only the authorized user is granted access; however, it has historically suffered from a range of cybercrimes related to the security and usability of the approaches. Therefore, it is apparent that a more convenient and secure user authentication solution is vital.

The existing context engine analyses a user's logon attributes based on a set of parameters such as geographical location, IP address, time of day, and device ID, and generates a context assurance level. If a security policy is met, users are granted access. If a security policy is not met, the user may be asked to step up security by providing an additional authentication factor.

Contextual Authentication collects a set of suitable IoT device data from a user's physical environment, generate unique fingerprints which can be used to authenticate users in applications. To my knowledge, there is no similar methods using "What's around you" factor and IoT devices for authentication.

IoT devices (such as smart home appliances) which are around and possibly connected to the user's device form a unique profile that varies through time and space as the user goes about his/her day. By using this profile along with the variables, it is possible to create a fingerprint of a user's environment as seen through a networked device. That profile may serve as a factor of authentication, adding "What's around you" to the common factors of "What you know" (passwords) and "What you have" (physical tokens).

From there, behaviour pattern of user can be built using a machine learning algorithm that could reside in user's mobile device or in a remote computing device. An overall unique fingerprint (or profile) of the user can be formed to distinguish himself from others. Hence, a unique identity of individual can be built up according to his lifestyle and interaction with IoT devices.

Figure 15: Authentication process

B. Sensor system and dataset

In the next step, some open-source sensor datasets have been used to conduct the experiment for this approach and neural network algorithms have been used for classification. The dataset we use here is the ADL (Activities of daily living) dataset of wireless sensor network.

Firstly, we will be going to explain the concept of sensor systems, with the application of existing systems. The wireless sensor network system is composed of multiple wireless network nodes where different sensors are connected. In the ADL system, for example, there are sensors that include passive infrared (PIR), which is used to sense any movement in a confined region; mercury contacts, which can detect how is the object moving; reed switches, which can calculate the cupboards and doors' open-close states; float sensors, which are used to detect whether the toilet have been flushed or not; pressure mats, which can figure out whether someone is sleeping on bed, or sitting down on a sofa [27]. To sum up, sensor throughput is binary, and it can be symbol of a specific space that can indicate how the activities been performed.

In the ADL system, there are three houses in total to keep record of data in the sensor system. House A has three room, house B has two room, and house C has two stories. Three houses' display varies significantly, e.g. house A has separate room for shower and toilet, house B has same room for shower and toilet, and house C has two toilets room. Apart from this, residence also different from each house to each house, in house A, there is a male of 26-year-old; in house B, there is a male of 28-year-old, whereas in house C, there is a male of 57-year-old [27]. The floorplans among house A, house B and house C are also different.

In this case, these three people have been asked to refer to the eight activities of daily living list to indicate their behaviour. The list of activities of daily living (ADLs) is a standard reference to monitor the health conditions for the elderly [27]. A wireless Bluetooth headset is used to indicate activities performed in house A and B. In this way, the residence can take down the starting and ending point of an activity that they performed. In house C however, a handwritten blog instead of wireless Bluetooth headset is used to indicate different activities [27]. These activities among all three houses are the same so they can compare apple to apple among each other.

C. Algorithm and performance

The objective of artificial neural network is to get the generalizations of math models of biology neural systems. And artificial neural network has been widely used in various industrial and academic areas, such as face recognition, machine translation, speech recognition, autonomous driving, fraud detection, bioinformatics, etc.

In this part, we have used multiple neural network models and techniques, including Radial Basis Function neural network and Self-Organizing Map neural network, to perform a binary-class pattern classification. Radial Basis Function neural network is a neural network relevant to multi-layer feed-forward neural network. Radial Basis Function neural network is inspired by biological neural system, hidden layer neurons are locally tuned and output layer is composed of linear neurons [37][38][39]. We are supposed to use the training data to train the RBF neural network in order to get the optimized parameters for the classifier. After the RBF neural network is well trained, we use the model to predict the classification labels of testing data.

The proposed RBF neural network has 20 neurons in the hidden layer and 1 neuron in the output layer. Actually there is no generic way to determine the best number of neurons and number of layers for a neural network. Other reasonable numbers are also workable, but experiments should be done to testify the performance in each setting. And we have chosen the Self-Organizing Map neural network to find the centre vectors for hidden layer neurons. In addition, linear least square estimation algorithm is used to determine the weights from hidden layer to output layer. The parameters that we need to find include centre vectors, weights, SOM map grid size (width and height) and basis function width [34][35][36].

1) *Center vectors*

The input space may not be covered well if the centres are randomly selected. Thus, it is advised to choose prototypes of samples as neuron centres and the centre locations of basis functions can be moved in a self-organized manner. The self-organized section of the learning process places the RBF function centres in the regions of input space where significant data is presented. As a result, we use Self-Organizing Map neural network to find the centre vectors. And we use som_make function to initialize and train the SOM neural network. The parameter 'msize' is a vector that specifies the map grid width and height. The generated centre vectors are stored in the array 'centres'. The array 'centres' is a 2 dimensional array with 20 rows.

2) *Weights*

The weights from input layer neurons to hidden layer neurons in RBF neural network are all set to 1, so we only need to find the weights from hidden layer to output layer. We use linear least square estimation algorithm to find the weights. Since the centre vectors have been known, the principle for finding weights is to find the weights which can minimize the loss function. In the equation, $d(k)$ is the target value and $f[x(k)]$ is the neural network output value.

$$J = \sum_{k=1}^N \{f[x(k)] - d(k)\}^2$$

The $f[x(k)]$ is defined as follows. We use Gaussian function as our radial basis function.

$$f[\mathbf{x}(k)] = \sum_{j=1}^m w_j \exp\left(-\frac{\|\mathbf{x}(k) - \mathbf{c}_j\|^2}{2\sigma^2}\right) = \sum_{j=1}^m w_j o_j(k)$$

3) *SOM map grid size & Basis function width*

As abovementioned, we have 20 neurons in the hidden layer. Thus, we can have various SOM map grids, such as 1 row 20 columns, 2 row 10 column, 4 row 5 columns, etc. And we also need to find the optimal value for basis function width.

In order to have a goodfit, we ought to choose the map grid and basis function width that generate highest testing accuracy. We cannot choose these parameters by optimizing training accuracy because it may lead to overfitting.

We use cross validation method to estimate the testing accuracy because testing data label is not available. In the cross validation, we use Leave-One-Out-Cross-Validation and randomly select one of the training data samples to hold out for the evaluation set and logical index vectors are returned for cross validation

```
[train,test] = crossvalind('HoldOut',size(data_train,1),1/330);  
xtrain = data_train(train,:);  
xtest = data_train(test,:);  
t_train = label_train(train,:);  
t_test = label_train(test,:);
```

Because cross validation randomly selects the evaluation set, the estimated testing accuracy is not a constant. Therefore, we run the cross validation within a loop and calculate the average testing accuracy.

We have tried the algorithm with various map grid sizes and basis function widths, different parameters generate different testing accuracy estimation. And we need to choose the parameters with the highest testing accuracy. Based on the cross validation results, we choose vector [10, 2] as SOM map grid size and 40 as the square of basis function width.

The details of parameter tuning are shown in the table below. The combination of [10,2] and 40 has achieved the highest testing accuracy, which is 0.901. Other parameter values have generated lower testing accuracies.

SOM map grid size	Square of basis function width	Iterations	Evaluation set size	Average testing accuracy
[10,2]	40	1000	1	0.901
[5,4]	40	1000	1	0.888
[4,5]	40	1000	1	0.884
[2,10]	40	1000	1	0.871
[10,2]	500	1000	1	0.873
[10,2]	250	1000	1	0.885
[10,2]	100	1000	1	0.876
[10,2]	25	1000	1	0.894
[10,2]	5	1000	1	0.79
[5,4]	100	1000	1	0.896

Table 5: Parameter tuning

The performance table of Radial Basis Function neural network and Self-Organizing Map neural network is shown below.

	RBF + SOM
Classification accuracy	90%
Authentication accuracy with threshold 80%	98%
True rejection rate with threshold 80%	99%
Training time	95s
Inference time	Within 3s

Table 6: Performance of Radial Basis Function neural network and Self-Organizing Map neural network

D. Results

From the algorithm performance table above (table 6), we can see that RBF and SOM algorithms have achieved a reasonable accuracy of 90%. The training time of the algorithms is around 95s, and the inference time is within 3s. However, we are using open-source sensor datasets to conduct the experiments, so the setting here is quite different from the real world environment.

E. Transfer learning

1) Advantage of transfer learning

All the data records are required to have the same feature space in traditional machine learning. However, this requirement is not applicable to every IoT system. Thus, transfer learning is of great significance to the IoT relevant research, because transfer learning is able to learn from data belonging to different feature spaces [23][48].

Transfer learning gets classification model parameters by encompassing training data from classification tasks which are relevant but different. There are two terms which are different from each other, one is source task, and the other one is target task. Source task is the process where training data is extracted, whereas the latter is the classification task, this task is what we are truly interested in [23]. In the early stage work of transfer learning, the main focus is on multi-task learning. In this case, compared with learning the tasks one by one, multi-task learning can produce a more desirable performance.

Let's take the newsgroup classification task as an example. The goal of this task is to figure out which newsgroup any specific document should be classified to [29]. First of all, we need to clarify whether this document originates from a space newsgroup or a hardware newsgroup. The performance will improve in a great extent if other newsgroups' training data is also included, e.g. baseball, motorcycles and religion, etc. The reason lies in that other newsgroups can contain information about the word co-occurrence. For example, Word 'moon' and 'rocket' often appear along with one another. In this case, if the word 'moon' appears in the space newsgroup, even if the word 'rocket' did not appear, the classifier can still obtain the information that the word 'rocket' is a word inside space newsgroup. This is because in other datasets, these two words usually appear co-currently.

2) Implementation analysis

How to implement transfer learning in the best way is still a hot research topic now. By using a prior distribution on model parameters, the implementation will cope well with the probabilistic models. This is because prior could give us an initial estimation of the model parameters for the target task and prior is learned from source task. Since the amount of training information can be obtained gradually, transfer learning will thus rely less on the prior [26]. In this way, an instinct system is formed to balance the influence of training data and prior distribution when model parameters are learned.

If we want to put transfer learning into recognition of the activity and authentication, every individual classification task corresponds to a house, where we can conduct recognition. One thing to take note is that we need to identify the difference between target house and source house. Similar to target task and source task, target house barely has any training data, whereas source house contains a large dataset which has been labelled. Every single house uses the same ADL list, however the sensor system is not the same for different houses. There exists two issues to be dealt with in order to apply transfer learning from source houses to target house. First problem is that since different houses will have different display and arrangement, how can we figure out the disparities existing among different sensor networks? Second problem is that how the model parameters can be learned so that we can take the residence's behaviour differences into consideration [30].

In order to solve the first problem, first thing is to take the house's different feature space into consideration. The structure of each individual's sensor network will be different since each house's display is different. Therefore, the feature space is different. For example, house 1 may use separate room for toilet and bathroom, whereas house 2 may put toilet and bathroom as a single room. Thus, the quantity and function of the sensors in use will definitely be different. In order to solve this issue, a common feature space based on certain mapping technique needs to be introduced to make the feature space applicable to all houses. In this mapping technique, we use the features called meta features [32], which is to describe the actual feature property. Every single sensor can be interpreted with various meta features. Taking a sensor installed in the microwave as an example, there may be one meta feature indicating that the sensor is placed inside the kitchen, whereas the other meta feature may indicate that the sensor is installed to the heating machine.

Regarding the second problem, it mainly focuses on the behaviour disparities among residents. In actual situation, same activity labels are utilized for every individual house, but how activities are actually carried out may still vary from each other. For instance, one person usually eats toast bread for breakfast, whereas another person may have preference to eat cereal. In this case, both person eat breakfast, which is the same label, but what and how they eat breakfast are different from each other. So in order to deal with those differences, different parameters are compulsory to make the model identify relevant activities. So in the mentioned case, each house should have a distinctive model, in which it has its own parameters to define this model. From the source houses, the system can learn the prior distribution, which can be applicable to build up the initial set of values for the model parameters in target house. If there are special behaviours existing, the model parameters can be changed by referring to the target house's training data, which can be unlabelled or labelled.

In the future work, more experiments are expected to justify the usefulness of transfer learning. And we hope to have a contextual IoT environment built up in the public, then we can run our experiments in the real-world environment.

V. CONCLUSION & FUTURE WORK

We have proposed 2 continuous authentication methods in the paper. And it has been proved that continuous authentication can reduce fraud by preventing account abuse and stolen credentials. Thus, when password is stolen or mobile device with easy to crack log-in password is hacked, continuous authentication can help detect the anomaly and control the device access accordingly. What's more, continuous authentication can identify abnormal behaviours of company employee to detect crime and combat insider attack using the sensor data.

In future, there is more work to do for the researchers in this area before the continuous authentication can be fully applied into the current authentication system. The multi-user recognition is one of the most important challenge for continuous authentication, because more than 1 user are granted the access in many authentication systems. It's necessary to recognize all the granted users individually when they try to get into the system. What's more, we must take effort to reduce the detection time of continuous authentication system, so we can perform compression for neural network to reduce the latency.

VI. REFERENCES

- [1] Frank, Jordan, Shie Mannor, and Doina Precup. "Activity and gait recognition with time-delay embeddings." *AAAI*. 2010.
- [2] Feng, T., Zhao, X., Carbunar, B., Shi, W.: Continuous mobile authentication using virtual key typing biometrics. In: *TrustCom*. IEEE (2013)
- [3] Shi, E., Niu, Y., Jakobsson, M., Chow, R.: Implicit authentication through learning user behavior. In: Burmester, M., Tsudik, G., Magliveras, S., IliÅLc, I. (eds.) *ISC 2010*. LNCS, vol. 6531, pp. 99–113. Springer, Heidelberg (2011)
- [4] 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 TIFS* 8(1) (2013)
- [5] Shi, Elaine, et al. "Implicit Authentication through Learning User Behavior." *ISC*. Vol. 6531. 2010.
- [6] Khan, Hassan, Aaron Atwater, and Urs Hengartner. "Itus: an implicit authentication framework for android." *Proceedings of the 20th annual international conference on Mobile computing and networking*. ACM, 2014.
- [7] Feng, Tao, et al. "Tips: Context-aware implicit user identification using touch screen in uncontrolled environments." *Proceedings of the 15th Workshop on Mobile Computing Systems and Applications*. ACM, 2014.

- [8] Amin, Reham, Tarek Gaber, and Ghada ElTaweel. "Implicit authentication system for smartphones users based on touch data." *Intelligent Data Analysis and Applications*. Springer, Cham, 2015. 251-262.
- [9] Al Abdulwahid, Abdulwahid, et al. "Continuous and transparent multimodal authentication: reviewing the state of the art." *Cluster Computing* 19.1 (2016): 455-474.
- [10] Spillane, R.: Keyboard Apparatus for Personal Identification. Technical Report 17, IBM Technical Disclosure Bulletin (1975)
- [11] Buchoux, A., Clarke, N.L.: Deployment of Keystroke Analysis on a Smartphone. In: Proceedings of the 6th Australian Information Security Management Conference, Perth, Western Australia, pp. 40–47. SECAU - Security Research Centre (2008)
- [12] Ahmed, A.A.E., Traore, I., Almulhem, A.: Digital Fingerprinting Based on Keystroke Dynamics. In: Proceedings of the Second International Symposium on Human Aspects of Information Security & Assurance (HAISA 2008), Plymouth, UK, pp. 94–104 (July 2008)
- [13] Cho, S., Han, C., Han, D.H., Kim, H.-I.: Web based Keystroke Dynamics Identity Verification Using Neural Network. *Journal of Organizational Computing and Electronic Commerce* 10(4), 295–307 (2000)
- [14] Frank, J., Mannor, S., Precup, D.: Activity and gait recognition with time-delay embeddings. In: AAAI (2010)

- [15] Clarke, N.L., Furnell, S.M., Reynolds, P.L.: Biometric Authentication for Mobile Devices. In: Proceedings of the 3rd Australian Information Warfare and Security Conference 2002, pp. 61–69 (2002)
- [16] Ala AA, Azizah AM (2014) Touch gesture authentication framework for touch screen mobile devices. *J Theor Appl Inf Technol* 62(2)
- [17] Jain L, Monaco JV, Coakley MJ, Tappert CC. Passcode keystroke biometric performance on smartphone touchscreens is superior to that on hardware keyboards. *Int J Res Comput Appl Inf Technol* 2014;2:29–33
- [18] Liaw, Andy, and Matthew Wiener. "Classification and regression by randomForest." *R news* 2, no. 3 (2002): 18-22.
- [19] Strobl, Carolin, Anne-Laure Boulesteix, Achim Zeileis, and Torsten Hothorn. "Bias in random forest variable importance measures: Illustrations, sources and a solution." *BMC bioinformatics* 8, no. 1 (2007): 25.
- [20] Breiman, Leo. "Random forests." *Machine learning* 45, no. 1 (2001): 5-32.
- [21] Ho, Tin Kam (2002). "A Data Complexity Analysis of Comparative Advantages of Decision Forest Constructors". *Pattern Analysis and Applications*. 5(2): 102–112. doi:10.1007/s100440200009.
- [22] En.wikipedia.org. (2019). Random forest. [online] Available at: https://en.wikipedia.org/wiki/Random_forest [Accessed 13 Aug. 2019].
- [23] Pan, Sinno Jialin, and Qiang Yang. "A survey on transfer learning." *IEEE Transactions on knowledge and data engineering* 22, no. 10 (2009): 1345-1359.

- [24] Um, Terry Taewoong, Franz Michael Josef Pfister, Daniel Pichler, Satoshi Endo, Muriel Lang, Sandra Hirche, Urban Fietzek, and Dana Kulić. "Data augmentation of wearable sensor data for parkinson's disease monitoring using convolutional neural networks." arXiv preprint arXiv:1706.00527 (2017).
- [25] Ding, Junhua, XinChuan Li, and Venkat N. Gudivada. "Augmentation and evaluation of training data for deep learning." In 2017 IEEE International Conference on Big Data (Big Data), pp. 2603-2611. IEEE, 2017.
- [26] Van Kasteren, T. L. M., Gwenn Englebienne, and Ben JA Kröse. "Transferring knowledge of activity recognition across sensor networks." In International Conference on Pervasive Computing, pp. 283-300. Springer, Berlin, Heidelberg, 2010.
- [27] S. Katz. Assessing self-maintenance: Activities of daily living, mobility, and instrumental activities of daily living. J. Am. Geriatrics Soc., 31(12):721-726, 1983.
- [28] Donald J. Patterson, Dieter Fox, Henry A. Kautz, and Matthai Philipose. Finegrained activity recognition by aggregating abstract object usage. In ISWC, pages 44-51. IEEE Computer Society, 2005.
- [29] Wen Yuan Dai, Gui-Rong Xue, Qiang Yang, and Yong Yu. Transferring naive bayes classifiers for text classification. In AAAI, pages 540-545, 2007.
- [30] Rajat Raina, Andrew Y. Ng, and Daphne Koller. Constructing informative priors using transfer learning. In ICML '06: Proceedings of the 23rd

international conference on Machine learning, pages 713{720, New York, NY, USA, 2006. ACM Press.

- [31] Irakleous, I., S. M. Furnell, P. S. Dowland, and Maria Papadaki. "An experimental comparison of secret-based user authentication technologies." *Information Management & Computer Security* 10, no. 3 (2002): 100-108.
- [32] Tim van Kasteren, Gwenn Englebienne, and Ben Krose. Recognizing activities in multiple contexts using transfer learning. In *Proceedings of the AAAI Fall Symposium on AI in Eldercare: New Solutions to Old Problems*. AAAI Press, 2008. ISBN=978-1-57735-394-2.
- [33] Tan W, Hsu J, Pinn F, inventors; Citicorp Developement Center Inc, assignee. Method and system for token-based authentication. United States patent application US 09/792,785. 2001 Nov 29.
- [34] Kohonen, T., 1997, June. Exploration of very large databases by self-organizing maps. In *Proceedings of International Conference on Neural Networks (ICNN'97)* (Vol. 1, pp. PL1-PL6). IEEE.
- [35] Kohonen, Teuvo, Erkki Oja, Olli Simula, Ari Visa, and Jari Kangas. "Engineering applications of the self-organizing map." *Proceedings of the IEEE* 84, no. 10 (1996): 1358-1384.
- [36] Vesanto, J., Himberg, J., Alhoniemi, E. and Parhankangas, J., 1999, November. Self-organizing map in Matlab: the SOM Toolbox. In *Proceedings of the Matlab DSP conference* (Vol. 99, pp. 16-17).

- [37] Park, Jooyoung, and Irwin W. Sandberg. "Universal approximation using radial-basis-function networks." *Neural computation* 3, no. 2 (1991): 246-257.
- [38] Musavi, Mohamad T., Wahid Ahmed, Khue Hiang Chan, Kathleen B. Faris, and Donald M. Hummels. "On the training of radial basis function classifiers." *Neural networks* 5, no. 4 (1992): 595-603.
- [39] Schaback, Robert. "Error estimates and condition numbers for radial basis function interpolation." *Advances in Computational Mathematics* 3, no. 3 (1995): 251-264.
- [40] De Luca, Alexander, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. "Touch me once and i know it's you!: implicit authentication based on touch screen patterns." In *proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 987-996. ACM, 2012.
- [41] Sitová, Zdeňka, Jaroslav Šeděnka, Qing Yang, Ge Peng, Gang Zhou, Paolo Gasti, and Kiran S. Balagani. "HMOG: New behavioral biometric features for continuous authentication of smartphone users." *IEEE Transactions on Information Forensics and Security* 11, no. 5 (2015): 877-892.
- [42] Matchett, N.D. and Kehoe, B.D., Matchett Noel D and Kehoe Brian D, 1993. Continuous biometric authentication matrix. U.S. Patent 5,229,764.
- [43] Jakobsson, Markus, Elaine Shi, Philippe Golle, and Richard Chow. "Implicit authentication for mobile devices." In *Proceedings of the 4th USENIX conference on Hot topics in security*, pp. 9-9. 2009.

- [44] Khan, Hassan, and Urs Hengartner. "Towards application-centric implicit authentication on smartphones." In Proceedings of the 15th Workshop on Mobile Computing Systems and Applications, p. 10. ACM, 2014.
- [45] Barbello, Brandon. "Continuous user authentication on mobile devices: Recent progress and remaining challenges." (2016).
- [46] Liu, Jie, et al. "Optimal combined intrusion detection and biometric-based continuous authentication in high security mobile ad hoc networks." IEEE transactions on wireless communications 8.2 (2009): 806-815.
- [47] Feng, Tao, Ziyi Liu, Kyeong-An Kwon, Weidong Shi, Bogdan Carbutar, Yifei Jiang, and Nhung Nguyen. "Continuous mobile authentication using touchscreen gestures." In 2012 IEEE Conference on Technologies for Homeland Security (HST), pp. 451-456. IEEE, 2012.
- [48] Sharaf-Dabbagh, Yaman, and Walid Saad. "On the authentication of devices in the Internet of Things." In 2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM), pp. 1-3. IEEE, 2016.
- [49] Patro, S., and Kishore Kumar Sahu. "Normalization: A preprocessing stage." arXiv preprint arXiv:1503.06462 (2015).
- [50] Khan, H., Hengartner, U. and Vogel, D., 2016, June. Targeted mimicry attacks on touch input based implicit authentication schemes. In Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services(pp. 387-398). ACM.