data

MDPI

Data Descriptor

BrainRun: A Behavioral Biometrics Dataset towards Continuous Implicit Authentication

Michail D. Papamichail [1,*], Kyriakos C. Chatzidimitriou [1], Thomas Karanikiotis [1], Napoleon-Christos I. Oikonomou [1], Andreas L. Symeonidis [1] and Sashi K. Saripalle [2]

[1] Department of Electrical and Computer Engineering, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece; kyrcha@issel.ee.auth.gr (K.C.C.); thomas.karanikiotis@issel.ee.auth.gr (T.K.); napoleon.oikonomou@issel.ee.auth.gr (N.-C.I.O.); asymeon@eng.auth.gr (A.L.S.)

[2] ZOLOZ, Kansas City, MO 64108, USA; sashi@zoloz.com

* Correspondence: mpapamic@issel.ee.auth.gr; Tel.: +30-2310-996-349

Received: 22 March 2019; Accepted: 29 April 2019; Published: 1 May 2019

Abstract: The widespread use of smartphones has dictated a new paradigm, where mobile applications are the primary channel for dealing with day-to-day tasks. This paradigm is full of sensitive information, making security of utmost importance. To that end, and given the traditional authentication techniques (passwords and/or unlock patterns) which have become ineffective, several research efforts are targeted towards biometrics security, while more advanced techniques are considering continuous implicit authentication on the basis of behavioral biometrics. However, most studies in this direction are performed "in vitro" resulting in small-scale experimentation. In this context, and in an effort to create a solid information basis upon which continuous authentication models can be built, we employ the real-world application "BrainRun", a brain-training game aiming at boosting cognitive skills of individuals. BrainRun embeds a gestures capturing tool, so that the different types of gestures that describe the swiping behavior of users are recorded and thus can be modeled. Upon releasing the application at both the "Google Play Store" and "Apple App Store", we construct a dataset containing gestures and sensors data for more than 2000 different users and devices. The dataset is distributed under the CC0 license and can be found at the EU Zenodo repository.

Dataset: https://doi.org/10.5281/zenodo.2598135

Dataset License: CC0

Keywords: continuous implicit authentication; mobile security; behavioral biometrics

1. Summary

The continuously increasing penetration of smartphones into every aspect of everyday life is more than evident. According to recent studies [1], the number of mobile phone users is almost exponentially increasing and is expected to exceed 5 billion within 2019. This aggressive penetration of smartphones into our daily lives as a primary way not only for entertainment and socializing, but also for working, has constituted a new paradigm, where more and more mobile applications are released for services provision purposes.

Obviously, the extensive use of mobile applications and the wide exchange of data over the web raise several security and privacy concerns. These concerns mainly originate from the fact that mobile devices (where information is stored) can be easily stolen and as a result access to confidential data can be compromised. In an attempt to come up against these security threats, many state-of-the-practice approaches involve authentication mechanisms such as passwords, unlock patterns, or even face or

fingerprint recognition [2,3]. The use of these techniques, however, has been proved to be inefficient in some cases, firstly because they add overhead (thus users store passwords and pins), and secondly because they perform one-time authentication (if you are authenticated once, you get full access to the service/application functionality). This is reflected by the fact that almost four out of ten users (42%) do not use the lock mechanism of their devices in an effort to simplify their interaction with them [4]. On top of the above, even in the cases where passwords are used, the data exposure concern is not sufficiently prevented, as the users tend to select simple and thus easy-to-type but also easy-to-guess passwords.

In an attempt to overcome the aforementioned issues, many research efforts are directed towards continuous implicit authentication on the basis of behavioral biometrics [2,5,6]. The main idea behind this approach is to take advantage of data that originate from the continuous interaction of the user with the mobile device, generate a number of features that uniquely model the user's interaction, and discriminate him/her among others [7,8]. Such a methodology enables passive authentication, as it relies on touch data that are already produced by the users without requiring any authentication-specific action. In fact, there are several studies in the context of continuous implicit authentication that focus on the capturing of gestures (touch sequences) [9,10]. These studies, however, appear to have two major limitations. First of all, they use a controlled laboratory-environment application with certain functionalities and limited predefined actions from the end-users [11], which comes in contrast to many real-world daily scenarios, where the users freely navigate through applications, in order to consume the functionality offered. Additionally, they are confined with a ground truth that involves a limited number of users (subjects) and thus are restricted to certain modelling scenarios. These limitations constitute a significant threat to validity.

In the context of this work and in an effort to overcome the above limitations, we employ the educational game "BrainRun"[1], available at both the "Google Play Store"[2] and "Apple App Store"[3], in order to collect gestures and sensors data from many different users/devices. The data collection methodology along with the design of the application as a series of mini-games each involving a unique setup enable capturing the behaviour of end-users under different usage scenarios. The collected data are included in the provided dataset, which contains the raw information regarding the gestures made by each user, meta-data regarding the user and the games played, and the raw data from the mobile device sensors.

2. Data Description

The data gathered from "BrainRun" are split in three major parts:

1. **Gestures**

   The gestures data contain the raw information regarding the coordinates of the screen points involved in every tap and swipe of all registered users. The data are organized in documents (.json files), each containing the information regarding a certain captured gesture (tap or swipe). In an effort to provide customizable dataset creation abilities, each document also contains information regarding the device, the name of the application screen, and the duration of the gesture.

2. **Users/Devices/Games**

   The data collected from users and games belong to three main categories: (a) information regarding the registered users, (b) information regarding the registered devices, and (c) information regarding all the games played. The information included in these categories facilitates the dataset creation procedure as it enables the retrieval of various statistics that can provide valuable insight to

[1] http://brainrun.issel.ee.auth.gr/.
[2] https://play.google.com/store/apps/details?id=gr.auth.ee.issel.brainrun.
[3] https://itunes.apple.com/us/app/brain-run-issel/id1413132878?ls=1&mt=8.

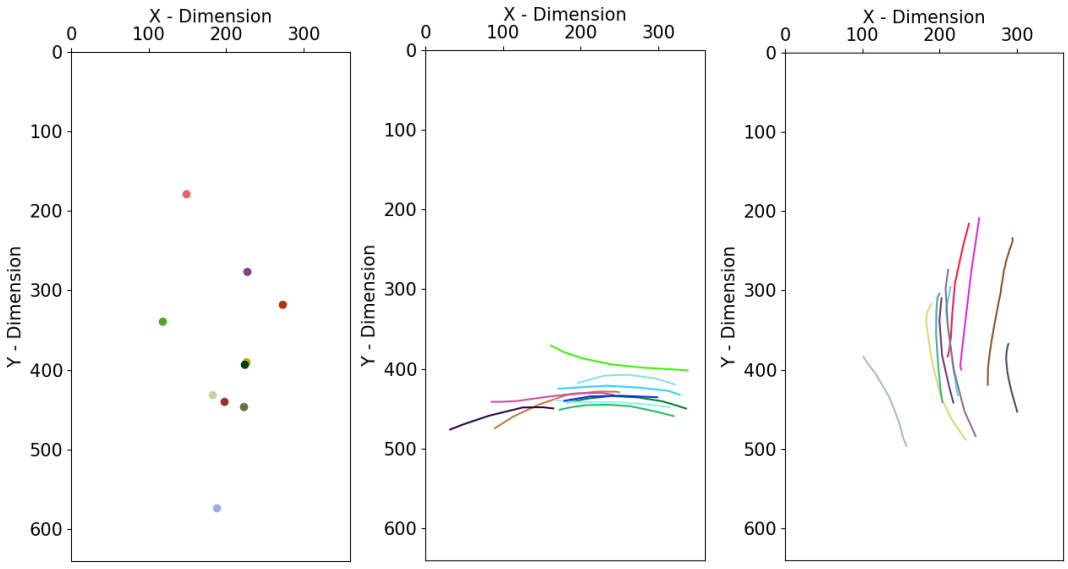
the modelling procedure (i.e., the investigation of the impact of the experience or the game type—speed-based/concentration-based—on the swiping behavior, the change of behavior among different devices, etc.).

3. **Sensors**

The raw measurements extracted from the mobile device sensors (accelerometer, gyroscope, magnetometer, and device motion sensor) of the BrainRun registered users. The sensors data can be used as an additional modelling parameter towards continuous implicit authentication as they enable capturing the behavior of end-users in terms of the way they interact with the device itself (i.e., holding position, shaking events, moving behavior, etc.).

2.1. Gestures

The gestures data capturing information reflects the way users interact with their mobile devices while playing the BrainRun game. There are two types of gestures: (a) *taps*, each referring to a single certain point touched in the screen, and (b) *swipes*, each referring to the activation of a series of screen points in a continuous hand movement. The provided dataset contains 3.11 M gestures collected from 2218 different users and 2418 different devices. The percentage of taps among the collected gestures is around 79%. Figure 1 illustrates a series of sample gestures (taps, horizontal swipes, and vertical swipes) for a certain user playing BrainRun in a mobile device with dimensions 380 × 640.



(**a**) Taps (**b**) Horizontal swipes (**c**) Vertical swipes

Figure 1. Visualization of sample gestures for a certain user.

Table 1 depicts the attributes stored for each data instance. This way one can link the gesture to a registered user (through the device identification number) and obtain information regarding the certain screen of the application and the timeframe when the gesture took place. The *data* attribute contains the core information regarding the gesture in the form of a list of objects, each referring to a single point touched in the screen during the hand movement. While taps contain only one list item, the number of list items in the case of swipes depends on the duration of the gesture. The sampling rate is 15–20 ms. Table 2 presents the attributes of each data list item, while Figure 2 along with Table 3 depict two different swipes along with the coordinates of the respective data points.

Table 1. Attributes of gestures data.

Attribute	Description
type	The type of the gesture (tap or swipe)
session_id	The id of the current session (if the application restarts, a different session id is generated)
device_id	The identification number of the device in which the gesture was made
t_start	The timestamp when the gesture started
t_stop	The timestamp when the gesture ended
screen	The identifier of the screen in which the gesture was made (i.e., ReactonGame-1.1.4)
data	A list containing all the data points involved in the gesture

Table 2. Attributes of data item.

Attribute	Description
moveX	The latest screen horizontal coordinate of the recently-moved touch
moveY	The latest screen vertical coordinate of the recently-moved touch
x0	The initial screen's horizontal coordinate when the gesture started
y0	The initial screen's vertical coordinate when the gesture started
dx	The accumulated horizontal distance of the gesture since the gesture started
dy	The accumulated vertical distance of the gesture since the gesture started
vx	The current horizontal velocity of the gesture
vy	The current vertical velocity of the gesture

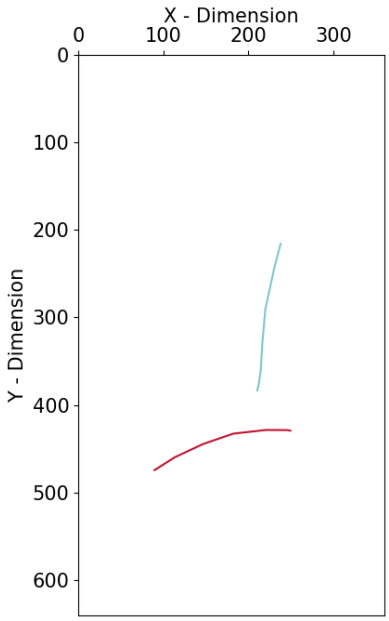


Figure 2. Swipes of a user.

Figure 3 illustrates the histogram regarding the number of points sampled by the mobile phone for every swipe included in the dataset. It is clear that the values regarding short swipes (less than 10 samples) follow a normal distribution, while 6 samples swipes exhibit the highest frequency. It is worth noting that there is a high number of gestures (>80 k) of length higher than 10 points.

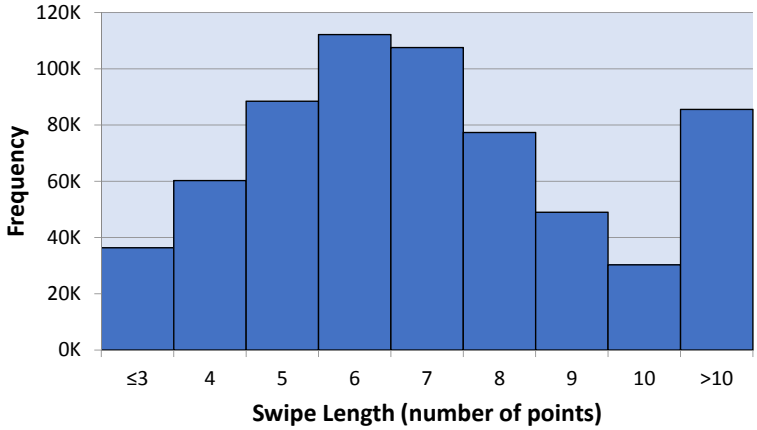


Figure 3. The number of points sampled at each swipe.

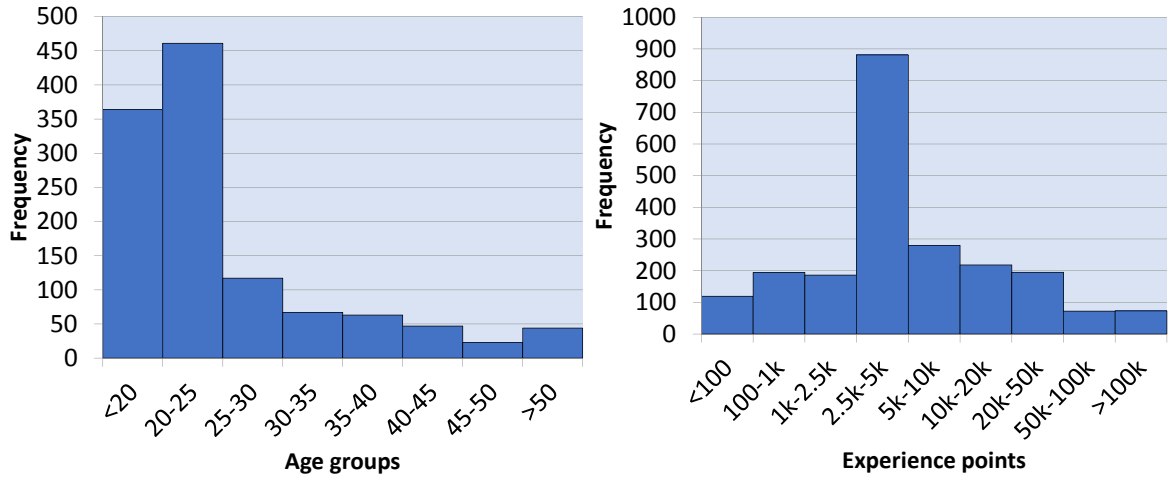
Table 3. Points sampled from swipes.

Swipe	Point	X-Dimension	Y-Dimension
horizontal	0	249.43	429.43
	1	246.10	428.76
	2	220.92	428.58
	3	182.14	432.82
	4	146.41	444.80
	5	113.36	459.86
	6	91.03	473.64
	7	89.58	474.40
vertical	0	210.47	383.45
	1	212.32	375.71
	2	214.58	360.12
	3	216.45	329.21
	4	219.95	290.73
	5	230.66	242.85
	6	237.94	215.87

2.2. Users

The available dataset contains 2218 registered users, i.e., users that have played at least one game. According to the gathered data, 60% of the users are men, 26% are women, while the rest (14%) have chosen not to reveal their gender. In addition, the vast majority of the users (almost 95%) have only used one specific device to play the game. Figure 4a depicts the distribution of the age groups among the registered users, while Figure 4b refers to the experience points[4] gained by completing the games. It is worth noting that both distributions reveal that the BrainRun registered users cover a wide range of usage scenarios, which facilitates the modeling procedure and strengthens the generalization ability of the continuous implicit authentication strategies.

[4] An experience point is a unit of measurement to quantify a player's progress through the game. Experience points are awarded for completing a single game according to the user's performance. The experience points of every player define the overall leaderboard of the game.



(**a**) Distribution of the age groups

(**b**) Distribution of the experience points

Figure 4. Distribution of (**a**) age groups and (**b**) experience points regarding the BrainRun registered users.

Table 4 depicts the attributes collected for users; apart from the player id, mainly gender and age attributes, game progress, and user statistics are collected.

Table 4. Attributes of the users collection.

Attribute	Description
user_id	The unique _id of the documents in the collection
gender	The registered user's gender
player_id	The registered user's unique player id
username	The registered user's selected username
age	The age of the registered user
last_login	The last timestamp when the registered user was active
xp	The experience points the registered user has earned
statistics	The statistics of the registered user

2.3. Devices

The third collection of the provided dataset contains information on the devices that the users have used to access the BrainRun application. In total, 2418 different devices were used (almost one device per user as described before), while the most used devices are "Redmi Note" smartphones. Most of the devices have 360 × 640 width and height dimensions and the vast majority of them run an Android operating system (nearly 90% of the devices versus 10% of iOS software devices).

Table 5 depicts the attributes included in the devices' collection. Using the aforementioned attributes one can link the device both to a registered user through the user identification number and to the games completed in the specific device through the device identification number, as well as obtain information regarding the dimensions (through the width and height attributes) and the operating system (through the os attribute) of the device.

Table 5. Attributes of the devices collection.

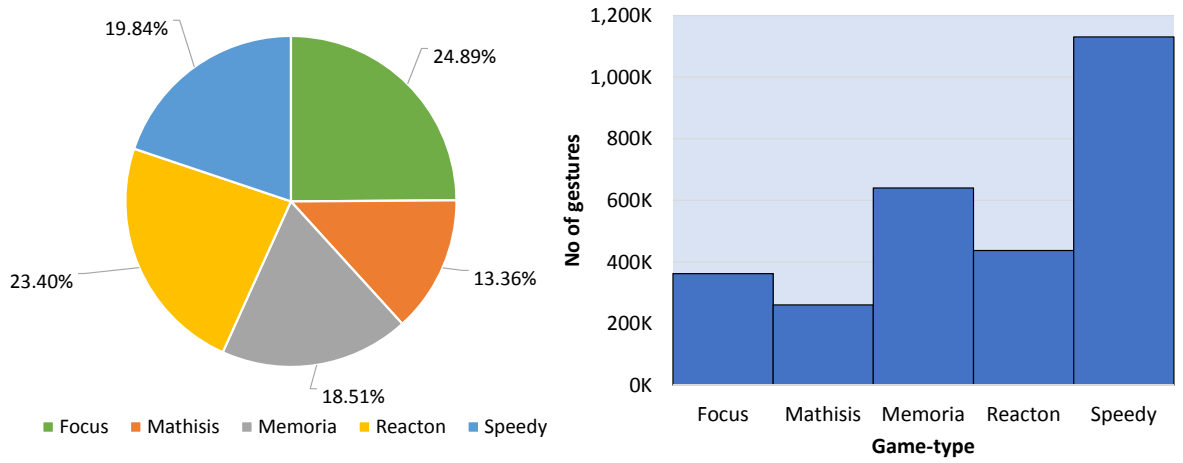
Attribute	Description
user_id	The _id of the collection users
device_id	The device id (as provided by the device os)
width	The width of the device in px
height	The height of the device in px
os	The operating system of the device (iOS/Android)

2.4. Games

BrainRun includes five different game-types, which are called "Focus", "Mathisis", "Memoria", "Reacton", and "Speedy", and their main goal is the collection of users' information. The provided dataset includes a collection that describes the games that the different users have played. The collection includes the 106,805 different games that the users have completed. Each game-type is specifically designed to collect different kind of hand gestures, such as taps, horizontal swipes, vertical swipes, swipes and taps combined, etc. The games' stages and difficulties are further discussed in Section 3.

Figure 5a presents the percentage of playing time for each game-type. Figure 5b depicts the number of gestures collected at each game-type. It is worth noticing that the game-type "Speedy", while is one of the least played games, has far more collected gestures than the rest of the game-types, and, at the same time, the most played game ("Focus") contains very few gestures, which is explained by the fact that the number of gestures needed to complete one game differs significantly between the game-types.

Table 6 depicts the attributes included in the game collection; information regarding the statistics of the game are collected (such as the number of correct or wrong answers, the game-type, and the timestamps of the beginning and the end of the game). Additionally, a link to the player and device id is provided.



(**a**) Percentage of playing time

(**b**) Number of gestures per game-type

Figure 5. The percentage of playing time and number of gestures for each game-type.

Table 6. Attributes of the games collection.

Attribute	Description
corrects_number	The number of correct answers in the game from the user
wrongs_number	The number of wrong answers in the game from the user
user_id	The _id of the user who played the game
device_id	The device_id of the device in which the game was played
stage	The stage of the game completed
base	The base (supplementary of stage) of the game completed
game_type	The game_type (Mathisis, Memoria, Reacton, Speedy, Focus)
game	The game number (1 to 4)
stars	The number of stars the user earned
t_start	The timestamp when the game begun
t_stop	The timestamp when the game ended
experience_earned	The experience points earned for this game

2.5. Sensors

The sensors dataset is a collection of the raw data coming from some mobile sensors that are activated during the user's interaction with the application and the games. The data are collected with a sampling rate of 100 ms and are saved to files with an appropriate format for their distinction:

{player_id}_{timestamp}.json

It is worth mentioning that the users are able to deactivate/turn off the sensors' data collection, in order to improve the performance of the application, reduce the data exchange rate and improve battery life.

The data coming from the sensors act as a supplementary material to the gestures' data, as they can be used together towards continuous implicit authentication. In many previous approaches, sensors' data constitute a major part of the features used, while there are some approaches in which the sensors' data are the only features used towards phone and device security. In [12], the phone orientation along with the finger's orientation were used as extra features, in order to better model the behavior of an individual and train a classifier that can continuously discriminate every user that uses one device. On the other hand, in [13] there are used only sensors' data coming from accelerometer, gyroscope and magnetometer sensors, in an attempt to define the behavioral pattern of each user, while the mobile phone is used. Finally, in [14], various sensors from the mobile phone and a wearable IoT device are used to identify an individual. All the approaches mentioned above conclude that the use of some sensors could definitely boost the achieved performance of a continuous user authentication system, along with the use of user's gestures.

Data from the **accelerometer, gyroscope, magnetometer, and deviceMotion sensors** are collected.

2.5.1. Accelerometer

The accelerometer sensor measures the acceleration of the specific device in the three axes. The sensor's measurements help figure out how fast the device is moving and which direction it is pointing at. Table 7 depicts the data stored from the accelerometer sensor.

Table 7. Accelerometer sensor's data.

Data	Description
acc_x	The acceleration of the device in the x-axis
acc_y	The acceleration of the device in the y-axis
acc_z	The acceleration of the device in the z-axis
screen	The screen in which the measurement was taken from

2.5.2. Gyroscope

The gyroscope acts as a complementary sensor to the accelerometer, figuring out the general position and behavior of the device. Gyroscope specifically measures the degree of rotation around the three axes. Table 8 depicts the data stored from the gyroscope sensor.

Table 8. Gyroscope sensor's data.

Data	Description
rot_x	The rotation of the device around the x-axis
rot_y	The rotation of the device around the y-axis
rot_z	The rotation of the device around the z-axis
screen	The screen in which the measurement was taken from

2.5.3. Magnetometer

The magnetometer defines the exact device position, orientation and direction. This sensor measures the magnetic field, in order to specify where the north is according to the device. Table 9 depicts the data stored from the magnetometer sensor.

Table 9. Magnetometer sensor's data.

Data	Description
mag_x	The strength and polarity of the magnetic field along the x-axis
mag_y	The strength and polarity of the magnetic field along the y-axis
mag_z	The strength and polarity of the magnetic field along the z-axis
screen	The screen in which the measurement was taken from

At this point, it worths noting that in the actual .json files included in the dataset, the data attributes for the accelerometer, gyroscope and magnetometer are defined as x, y, and z. The use of the prefixes acc, rot and mag was only for demonstration purposes (the prefixes not needed in the .json files as the attributes are nested inside the accelerometer, gyroscope, and magnetometer descriptors).

2.5.4. DeviceMotion

The data from sensors that are provided include also a package of measurements coming from various sensors and collected by a library, called DeviceMotion, of the React Native framework, which was used to develop the application (more details in Appendix A). These measurements are used complementary with the aforementioned sensors to describe in a more precise way the device's general status. The values returned from the DeviceMotion library are slightly different from the respective ones from the other sensors data. Table 10 depicts the data collected by the DeviceMotion library.

Table 10. DeviceMotion library's data.

Data	Description
orientation	Device orientation based on screen rotation (0 for portrait, 90 for right landscape, 180 for upside down, and −90 for left landscape)
acceleration	Device acceleration on the three axis as an object with x, y, z keys
rotationRate	Rotation rates of the device around each of its axes as an object with alpha, beta, gamma keys where alpha is around Z axis, beta for X axis, and gamma for Y axis
accelerationIncludingGravity	Device acceleration with the effect of gravity on the three axis as an object with x, y, z keys
rotation	Device's orientation in space as an object with alpha, beta, gamma keys where alpha is for rotation around Z axis, beta for X axis rotation, and gamma for Y axis rotation
screen	The screen in which the measurement was taken from

2.6. Data Format

As already discussed, the full dataset is divided into three major parts: (i) the gestures' data, (ii) the users and games' data, and (iii) the sensors' data. All data are saved in json format. The sensors' data are saved with the appropriate title. The datasets of gestures, users, devices and games are provided in different files in order to be easy to handle and use. The files could be also easily loaded into the MongoDB environment for further querying and processing.

3. Methods

As already mentioned, the generated dataset aims at providing a solid basis upon which continuous authentication models can be built. Towards this direction and in an effort to refrain

from the limitations of current approaches, we employed "*BrainRun*". Through BrainRun we are able to attract a wide range of different users and collect different types of gestures (vertical, horizontal, speed-based, accuracy-based, etc.) that appropriately describe swiping and tapping behavior, thus facilitating robust user modelling strategies.

In an effort to cover a wide range of usage scenarios, BrainRun includes 5 different stages, with 2 sub-stages (called bases) of increasing difficulty each (i.e., stage 1 includes the bases 1 and 2, which are referred as 1.1 and 1.2). Each sub-stage contains 4 game-types, each with 4 different difficulty levels (i.e., the sub-stage 1.2 contains 4 levels regarding the game type "*mathisis*"). As a result, the complete BrainRun framework includes 160 different usage scenarios ($5\,stages * 2\,subStages * 4\,gameTypes * 4\,levels = 160$). The game format was carefully selected in order to attract users with different background and level of expertise along with enabling their long-term active involvement. A user who completes a single level earns up to three stars based on his/her performance (evaluated as a combination of both wrong answers rate and speed of action). These stars help the user unlock the next levels of the game and thus progress to more challenging stages. More info and explanatory game screens can be found in Appendix A.

The main target of BrainRun involves the collection of biometrics data from a large number of users using different devices who will play on a regular basis. The application was actively promoted, especially in the university community where the authors have direct access. Additionally, in an effort to further motivate the existing users and boost the publicity of BrainRun for attracting new users, three big tournaments were organized, with a duration of 3 to 4 weeks each. The rules of each tournament were carefully designed in order to serve the data collection needs. Towards this direction, the first two tournaments followed a sum-based scoring scheme where users gained more points by playing more games, while the third tournament followed a max-based scoring scheme where users were directed towards improving their previous score in order to gain more points. In addition, trying to increase motivation and thus the enrolled users' number and engagement, the best players of each tournament (5 up to 8) were given prizes (smartwatches, giftcards, etc.). The results showed that the tournaments were crucial for boosting the publicity of BrainRun and attracting more users. It is worth noting that in the second tournament (*November tournament*) there were more than 400 enrolled users who played more than 60,000 games and made more than 1 million gestures.

The constructed dataset, following the aforementioned strategy, was gathered exclusively for research purposes in the domain of Continuous Implicit Authentication (CIA). As stated in the Terms of Services[5], the collected data are absolutely anonymous and there is no way that anyone could back-track the application use and/or gestures and possibly find out the real person owning a specific game profile, unless the username used is indeed the person's name, or can be linked to one. But this is entirely up to the user. The only data gathered that could be used to identify an individual are:

1. Username
2. Age
3. Gender
4. Technical details regarding the type of the mobile device

From the aforementioned data, the only mandatory is the username, while the rest are optional. Regarding the technical specifications of the mobile device, the application only stores the screen dimensions, the operating system and the device id. Last but not least, the application respects all the legal data protection regulations and, in particular, the EU General Data Protection Regulation 2016/679[6].

[5] http://brainrun.issel.ee.auth.gr/tos.html.
[6] http://data.europa.eu/eli/reg/2016/679/oj.

As previously stated, the provided dataset could constitute the main object towards Continuous Implicit Authentication (CIA) approaches. The main approach towards CIA adopts the "One-vs-All" strategy. In this strategy, a model is trained to recognize the gestures of what we call the original user and then tested upon the gestures made both from the original user, but also using available gestures from other users (considered as attackers). The original user should be left uninterrupted by the model, while the other users should be restricted. Usually, such models are built using Outlier Detection algorithms (for example Isolation Forests) or 1-class classifiers like 1-class Support Vector Machines. Certainly, the provided dataset could also be used in different strategies found in CIA approaches, such as the "One-vs-One" strategy, comparing two different users, one against the other, the "One-vs-Group" approach, in which the above-mentioned "One-vs-All" strategy is applied to a subgroup of the available users and the multi-class classification approach, in which a classifier has to recognize the "owner" of a gesture from a pool of preselected users.

Additionally, the raw data of every gesture are stored in the dataset and provide valuable information about the behavior of the user. From the raw data one can calculate most of the features proposed in the relevant approaches [12]. Table 11 depicts some of the most used derived features regarding tap data, while Table 12 contains derived features regarding swipes.

Table 11. Features referring to taps.

Feature Name	Feature Description
Horizontal Position	Calculated distance from the left edge of the screen
Vertical Position	Calculated distance from the top edge of the screen
Descriptive Statistics for Position Data	Maximum, minimum and median values of taps position in vertical and horizontal direction

Table 12. Features referring to swipes.

Feature Name	Feature Description
Duration	The time duration of the swipe
Horizontal Length	The total length of the swipe in the horizontal axis
Vertical Length	The total length of the swipe in the vertical axis
Average Velocity	The average velocity of the movement
Horizontal Trace Variance	Calculated deviation of the swipe movement from the straight horizontal line
Vertical Trace Variance	Calculated deviation of the swipe movement from the straight vertical line
Angle	The calculated angle of movement during the swipe

4. User Notes

Figure 6 illustrates the provided dataset[7], which is composed of two main parts. The first contains the sensors data, which are in .json format and thus can be easily read and processed by any tool that provides json parsing capabilities. As shown in the figure, the name of each .json file contains information regrading the user along with the timeframe the data collection took place (the timeframe is given in unix timestamp in milliseconds).

[7] Available at Zenodo repository: https://doi.org/10.5281/zenodo.2598135.

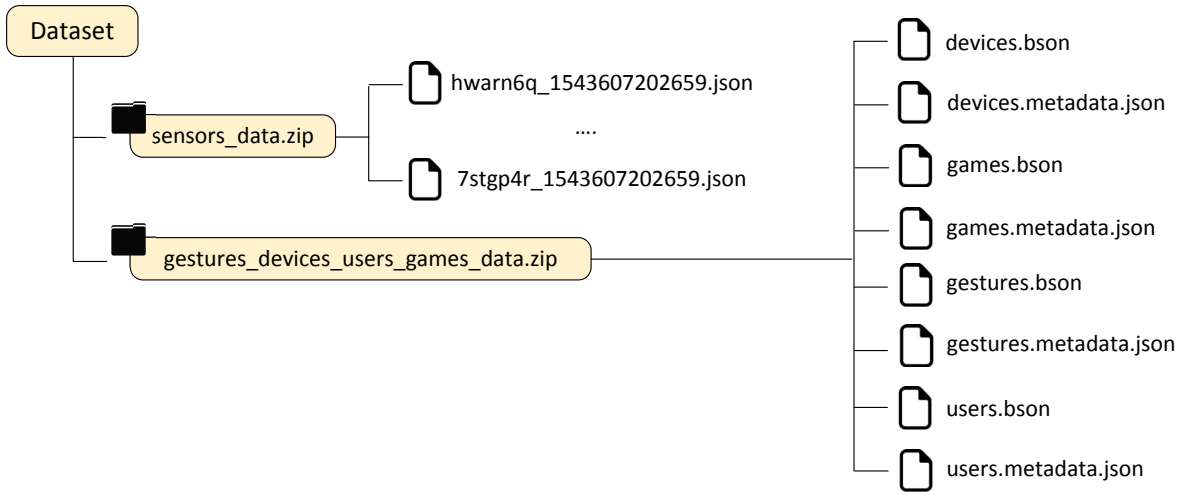


Figure 6. Overview of the provided dataset.

The second part contains the data regarding the registered users, the devices, the games, and the gestures (as described in the previous section). The data were exported from a MongoDB database system in a .bson format (binary-encoded serialization of JSON documents). We chose to use MongoDB for data storage purposes given the fact that it is a document database which ensures scalability and offers advanced storing, retrieving and querying abilities. In order to use the provided dataset, one can follow the next simple steps:

1. ***Install MongoDB***

   MongoDB is open source and available for all major operating systems. Intallation instructions can be found at the following link[8].
2. ***Download dataset***

   Download the dataset (hosted at Zenodo repository) and save it into a certain directory in your hard drive.
3. ***Extract data***

   Extract *gestures_devices_users_data.zip* into a certain directory in your hard drive.
4. ***Import dataset into MongoDB instance***

   Once having extracted the *gestures_devices_users_data.zip*, you can simply import it in to the local MongoDB instance using the following command:

   mongorestore −−**db** database_name dataset_directory/

   The *database_name* can be any name of your choice, while the *dataset_directory* refers to the path of the folder where you extracted the data.

The link between the sensors data and the MongoDB data is the player_id, which is included in the users MongoDB collection.

Author Contributions: Conceptualization, M.D.P., K.C.C., and A.L.S.; methodology, M.D.P., K.C.C., N.-C.I.O., T.K., and A.L.S.; software, M.D.P., N.-C.I.O. and T.K.; validation, M.D.P., K.C.C., A.L.S, and T.K.; formal analysis, M.D.P., K.C.C., and T.K.; investigation, M.D.P., K.C.C., T.K., and A.L.S.; resources, A.L.S.; data curation, M.D.P., N.-C.I.O.; writing—original draft, T.K.; writing—review and editing, M.D.P., K.C.C., A.L.S., and S.K.S; visualization, M.D.P., K.C.C., and T.K.; supervision, K.C.C., A.L.S., and S.K.S.; project administration, M.D.P., K.C.C., A.L.S., and S.K.S.; funding acquisition, A.L.S., and S.K.S.

[8] https://www.mongodb.com/download-center/community.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Brain Run App Overview

The main data collection mechanism, as already described before, is implemented through the BrainRun mobile application for both Android and iOS operating systems. The application was developed using the React Native framework[9] and the data collection of the gestures and the sensors' data was accomplished using the Pan Responder[10] and the DeviceMotion[11] libraries. The homepage displayed to the user when the application opens, along with the available choices are shown in Figure A1.

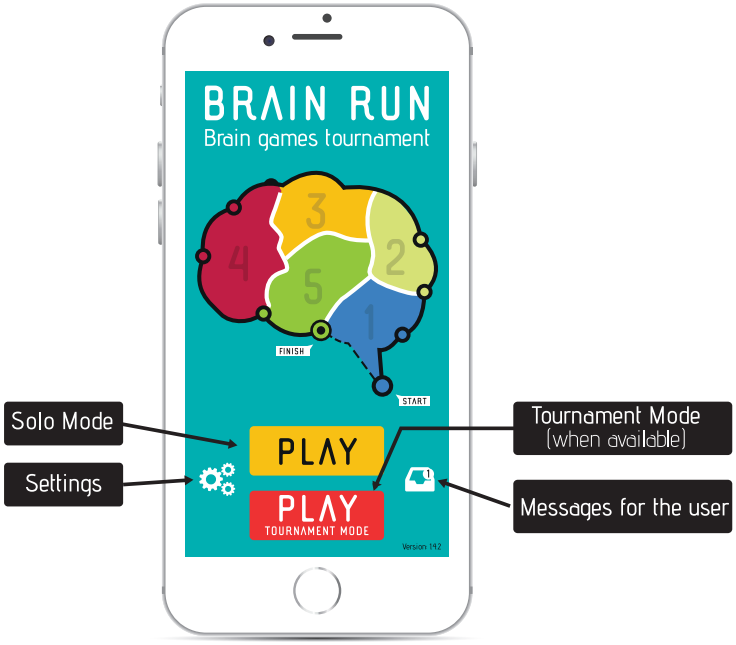


Figure A1. The homepage of BrainRun.

BrainRun consists of 5 different game-types that test the memory, the reaction, and the mathematical skills of users. The game types are described in the following sections.

Appendix A.1. Focus

This game targets at the memory skills of users. Initially, the user is presented with a set of different shapes and/or numbers. After a small time frame, the set disappears and the game starts; the user is presented with a shape/number and is asked to recall if the shape was in the initial set. In order to answer, the user has to swipe up or down, depending on whether the presented shape existed in the initial set. This game type targets at gathering vertical swipes. As the games progresses, the questions get more challenging; the initial set has not only more objects, but also objects with different colours. Figure A2 illustrates two sample screens of the focus game. In this case given that the number -7 existed in the initial set of numbers, the user should swipe up to proceed to the next question.

[9] https://facebook.github.io/react-native/.
[10] https://facebook.github.io/react-native/docs/panresponder.
[11] https://docs.expo.io/versions/latest/sdk/devicemotion/.

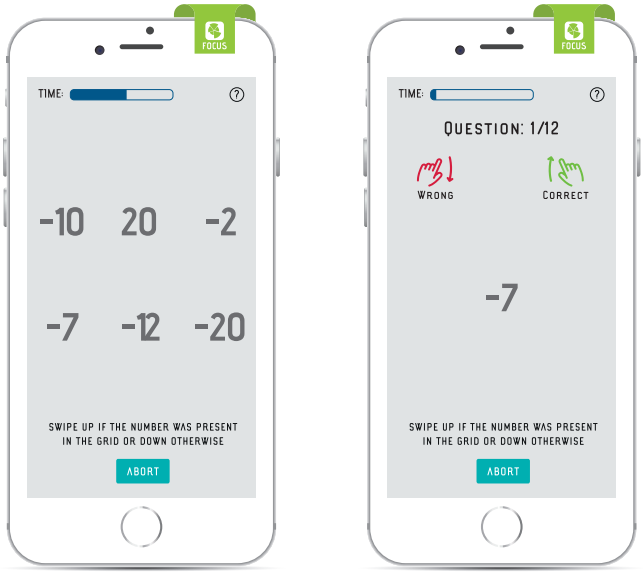


Figure A2. Focus example screens.

Appendix A.2. Mathisis

Mathisis involves solving small mathematical equations and targets at gathering horizontal swipes. The user is presented with an equation and is asked to swipe left or right, depending on whether the equation is wrong or correct. As the games progresses, the questions get more challenging. Figure A3 illustrates three sample screens of the mathisis game.

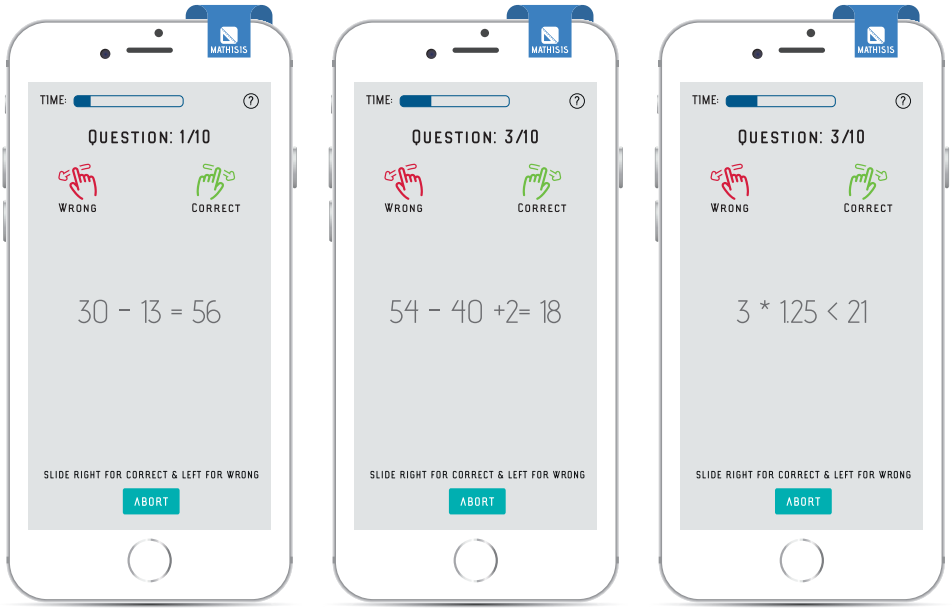


Figure A3. Mathisis example screens.

Appendix A.3. Memoria

Similar to focus, memoria is also a memory-testing game which targets at gathering taps. Memoria starts with a grid of white tiles. After a certain time period, some tiles change colour and the user is given a time frame to memorize the colour of each tile. After this time period, all the tiles of the grid turn white again and the user is prompted to select the tiles of a certain colour. As the games

progresses, the questions get more challenging; the initial grid not only gets bigger, but also the number of colours increases. Figure A4 illustrates three sample screens of the memoria game.

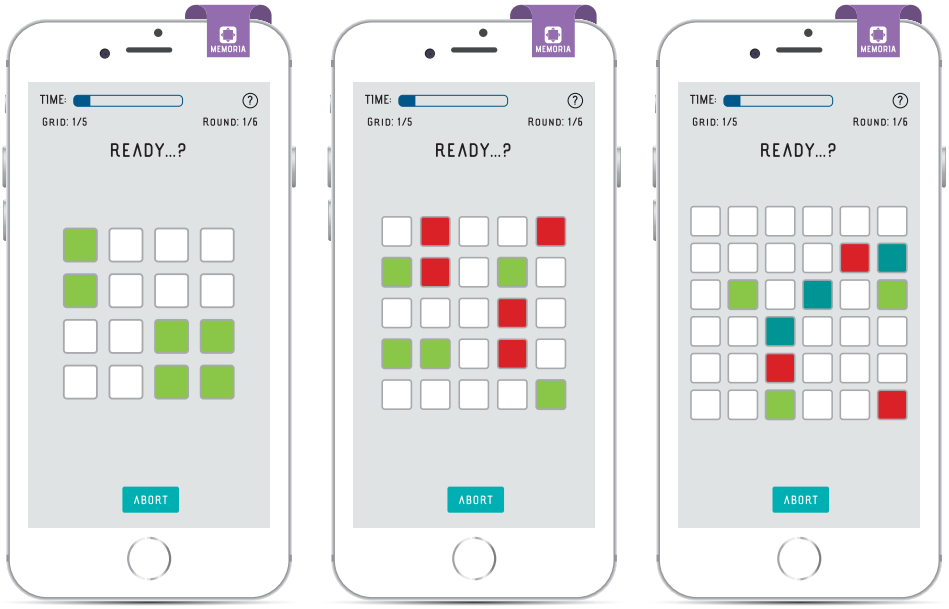


Figure A4. Memoria example screens.

Appendix A.4. Reacton

Reacton is a reaction-based game that combines both taps and swipes. The game includes a grid consisting of bulbs and switches. At random time intervals, one of those turns on (changes to colour different than the default grey). The goal of the game is to turn each one off, as quickly as possible. As the games progresses, it gets more challenging; the initial grid gets bigger, the given time window decreases, and more colours are added, including ones that the user must not turn off. That way, both user's taps and swipes are gathered. Figure A5 illustrates three sample screens of the reacton game.

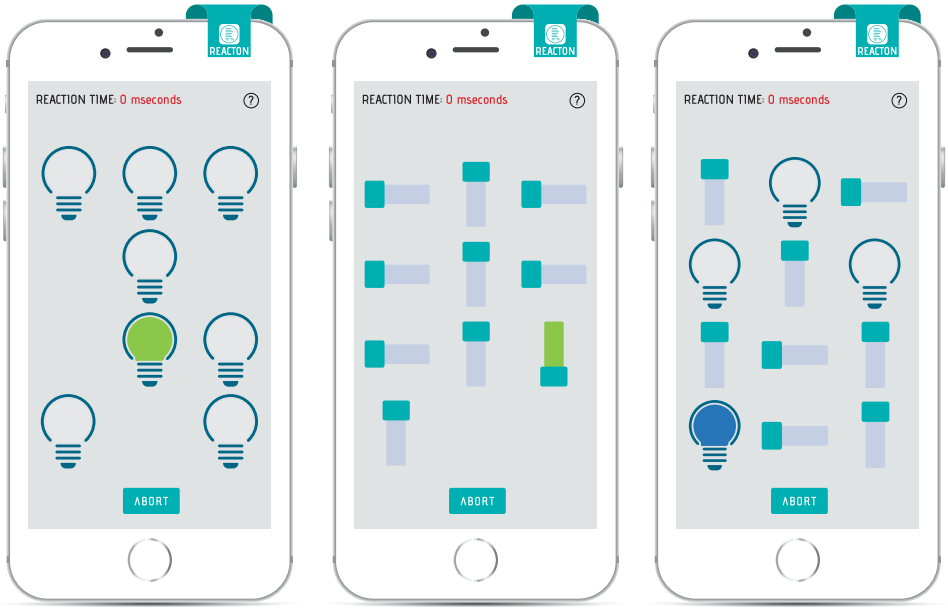


Figure A5. Reacton example screens.

Appendix A.5. Speedy

Speedy is all about speed and targets at collecting the taps of the registered users. This game starts with a grid of coloured rockets. Every rocket has one of two random colours and by pressing it the user can turn the one colour to the other. The goal of the game is to make every rocket the same colour, as quickly as possible. As the games progresses, it gets more challenging; the initial grid gets bigger, the given time window decreases, and obstacles are added in front of the rockets, which the user has to "break" to be able to change the color of the rocket. Figure A6 illustrates two sample screens of the reacton game.

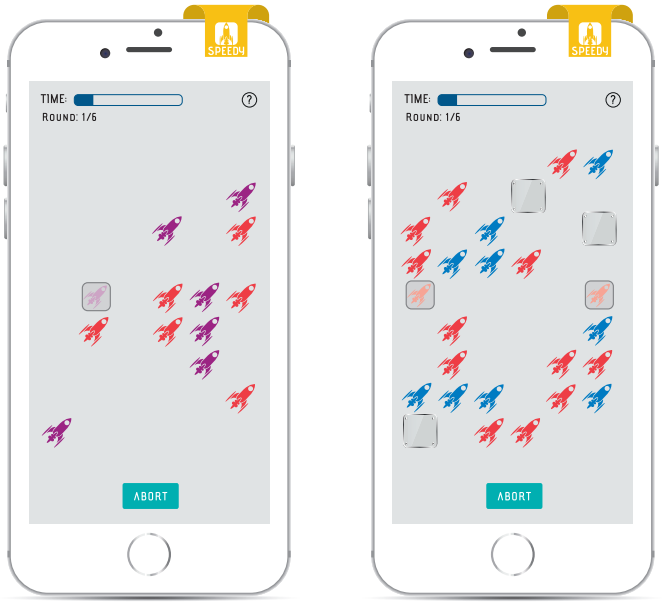


Figure A6. Speedy example screens.

References

1. Number of Mobile Phone Users. 2016. Available online: https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/ (accessed on April 19, 2019).
2. Alzubaidi, A.; Kalita, J. Authentication of Smartphone Users Using Behavioral Biometrics. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 1998–2026. [CrossRef]
3. Gong, N.Z.; Payer, M.; Moazzezi, R.; Frank, M. Towards Forgery-Resistant Touch-based Biometric Authentication on Mobile Devices. *CoRR* **2015**, abs/1506.02294.
4. Egelman, S.; Jain, S.; Portnoff, R.S.; Liao, K.; Consolvo, S.; Wagner, D. Are You Ready to Lock? Understanding User Motivations for Smartphone Locking Behaviors. In Proceedings of the 2014 ACM SIGSAC conference on Computer & communications security (CCS '14), New York, NY, USA, 3–7 Novermber 2014.
5. Yampolskiy, R.; Govindaraju, V. Behavioural biometrics: A survey and classification. *Int. J. Biom.* **2008**, *1*, doi:10.1504/IJBM.2008.018665. [CrossRef]
6. Feng, T.; Liu, Z.; Kwon, K.; Shi, W.; Carbunar, B.; Jiang, Y.; Nguyen, N. Continuous mobile authentication using touchscreen gestures. In Proceedings of the 2012 IEEE Conference on Technologies for Homeland Security (HST), Waltham, MA, USA, 13–15 November 2012; pp. 451–456.
7. Saevanee, H.; Clarke, N.; Furnell, S.; Biscione, V. Continuous user authentication using multi-modal biometrics. *Comput. Secur.* **2015**, *53*, 234–246. [CrossRef]
8. Neverova, N.; Wolf, C.; Lacey, G.; Fridman, L.; Chandra, D.; Barbello, B.; Taylor, G. Learning Human Identity From Motion Patterns. *IEEE Access* **2016**, *4*, 1810–1820. [CrossRef]

9. De Luca, A.; Hang, A.; Brudy, F.; Lindner, C.; Hussmann, H. 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, Austin, TX, USA, 5–10 May 2012; ACM: New York, NY, USA, 2012; pp. 987–996.
10. Mahbub, U.; Sarkar, S.; Patel, V.M.; Chellappa, R. Active user authentication for smartphones: A challenge data set and benchmark results. In Proceedings of the 2016 IEEE 8th International Conference on Biometrics Theory, Applications and Systems (BTAS), Niagara Falls, NY, USA, 6–9 September 2016; pp. 1–8.
11. Zheng, N.; Bai, K.; Huang, H.; Wang, H. You Are How You Touch: User Verification on Smartphones via Tapping Behaviors. In Proceedings of the 2014 IEEE 22nd International Conference on Network Protocols, Raleigh, NC, USA, 21–24 October 2014; pp. 221–232.
12. Frank, M.; Biedert, R.; Ma, E.; Martinovic, I.; Song, D. Touchalytics: On the Applicability of Touchscreen Input as a Behavioral Biometric for Continuous Authentication. *CoRR* **2012**, abs/1207.6231. [CrossRef]
13. Ehatisham-ul Haq, M.; Awais Azam, M.; Naeem, U.; Amin, Y.; Loo, J. Continuous Authentication of Smartphone Users Based on Activity Pattern Recognition Using Passive Mobile Sensing. *J. Netw. Comput. Appl.* **2018**, *109*, 24–35. [CrossRef]
14. Lee, W.; Lee, R.B. Sensor-Based Implicit Authentication of Smartphone Users. In Proceedings of the 2017 47th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Denver, CO, USA, 26–29 June 2017; pp. 309–320.