Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

# Online Learning and Saliency Effects on CNN-based Gait Recognizers

Hozaifa H. Daoud[1] hdaoud@azhar.edu.eg
Abdurrahman A. Nasr[1*] anasr@azhar.edu.eg
Mohamed M. Ezz[2] ezz.mohamed@azhar.edu.eg
Mohamed Z. Abdulmaged[1] mzaki14@azhar.edu.eg

[1]System and Computer Engineering Dep., Al-Azhar University, Cairo-Egypt
[2] College of Computer and Information Sciences, Jouf University, Sakaka 72314, Saudi Arabia

**ABSTRACT**

**Authentication through gait analysis offers a reliable and an easy-to-use alternative to common authentication methods. This paper presents a novel gait recognizer that exploits online learning in Convolutional Neural Network, CNN. The features which make that algorithm promising are its high recognition accuracy and low computational cost, in addition to its adaptability, flexibility and applicability. Also, in a parallel line the effect of saliency as a means to generate global features is examined.**

**In this paper, the inertial measures (instead of visual data) are utilized for person authentication. Thus the smartphone inertial sensors are used to continuously assess whether the mobile is actually in the hands of the right owner or not. Three different approaches (saliency detection, offline, and online learning) have been proposed, examined, and implemented. The last two of these approaches are based on the use of convolutional neural networks to shift the measured values of the sensors into a new vector that can be classified more reliably, while the first approach is based on the use of saliency detection algorithms to get the most salient regions of the gait. The models of these approaches are carried out and various experiments on such models are conducted. The results of these experiments were promising and showed the applicability of gait recognition to provide implicit continuous authentication, specially, when online learning is relied upon, since the identification accuracy reaches 98.7%.**

Keywords: **continuous authentication, gait recognition, saliency, convolutional neural network, online learning.**

Date of Submission : 14, Nov 2022 Date of Acceptance: 20, Dec 2022

## 1. INTRODUCTION

The person's walking style is a unique behavioral characteristic, which is known as gait. In fact researchers have conducted a large body of gait studies and analyses, however, when machine learning recognition approaches are employed it can be easily discovered that many essential questions are still unanswered in that area. Those questions include the following.

1- Which input data to the recognizer is more impressive, whether visual or inertial?
2- What features, whether global or local are more effective for continuous authentication applications?
3- Should deep learning be relied upon, what is the training approach that can provide the most accurate results?
4- What is the proposed deep learning architecture that can serve efficiently as a real time gait recognizer?
5- What are the practical constraints for the hardware implementation of gait recognition algorithms?

This paper aims at answering those questions in a trust worthy way so that gait analysts can use it reliably for continuous authentication. Here, the first question is answered by examining two types of data, as input to the recognizer, where one is visual while the second is inertial, Figure 1. Moreover, the inertial data is statistically analyzed to reveal whether or not the gyroscopic readings are depending on the accelerometer measures. The second question is answered by making use of a saliency detection procedure to obtain the gait global features and a layered convolutional neural network to yield the corresponding local features of the motion. The third question is answered by considering both offline and online learning methods. The fourth question is answered logically by making use of circuits and systems engineering constraints to obtain a neural network architecture that is most suitable for the underlying learning algorithm. The last question is answered by exploiting an Odroid-XU4/GPU kit to implement the proposed gait recognizer. With these questions as well as the results of previous related works are in mind we arranged the execution of this research.

Since gait recognition is widely known as the most important non-contactable, non-invasive biometric identification technology, which is i.e. hard to imitate. It has been extensively acknowledged by researchers as a biometric invariant that can be used

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

for authentication purposes via recognizing individuals based on their style of walking.

This paper is organized as follows. Section 2 presents the related work while section 3 discusses the underlying CNN architecture. Section 4 explains saliency as a technique for providing global features as well as presenting both offline and online learning methods. Section 5 focuses on the software and hardware implementation of the proposed gait recognizer. Section 6 discusses the results and section 7 concludes the paper.

## 2. RELATED WORK

In this section attention is focused on smartphone based gait recognition researches and frameworks. In 2016, a framework for smartphone continuous authentication using a set of behavioral biometric features is introduced in[1] , it is hand movement, orientation, and grasp (HMOG). This framework uses inertial sensors such as (accelerometer, gyroscope, and magnetometer) and it is tested from three viewpoints (continuous authentication, biometric key generation, and energy consumption) to capture the movements and orientation when a user taps on the screen to demand a new data. Such approach achieves a reasonable result when it is used to continuously authenticate smartphone users because it improves the performance of user behavior represented by dynamic features.

Lee et. al.[2] proposed an authentication system for implicit continuous authentication on smartphones based on user behavioral characteristics. They used the built-in accelerometer and gyroscopes of smartphones and smartwatches. They combined the smartphone and smartwatch sensors data to enhance the authentication accuracy. They also propose novel context-based authentication models to differentiate the legitimate smartphone owner versus imposters. They used different algorithms to achieve authentication from which the kernel ridge regression, KRR was the best one with 98.1% accuracy and less than 2.4% additional battery consumption.[2]

In [3], the authors focused on using a smartphone to identify its user. A smartphone accelerometer, gyroscope, and magnetometer were used to collect data from the individuals. Each individual performed six different physical activities including walking, sitting, standing, running, walking upstairs, and walking downstairs for 3-5 minutes. Their results showed that the superiority of the Bayesian network classifier with accuracy of 94.57%.

In [4], unlike the traditional methods of gait recognition, the authors studied the continuous authentication in the wild. The smartphone inertial sensors (accelerometer and gyroscope) are used to collect the data from the individuals freely and the information of when, where, and how the user walks was unknown. Deep learning techniques were used in order to gain high performance. A hybrid method was proposed to combine the deep convolutional neural network and deep recurrent neural network for robust inertial gait feature representation, Then a CNN with one dimension kernels was used to transform the input time series into convolutional feature maps, which were then carefully rearranged as time-series feature maps and fed into an LSTM for gait feature extraction. They showed that the features extracted were discriminative for authentication. They also discovered that gyroscope data is less efficient than accelerometer data but combining the two types of data enhance the performance.

A framework for gait-based user identification and verification using smartphone sensor data was tackled in [5].It utilizes some embedded smartphone sensors such as gyroscope and accelerometer to identify the user gait characteristics. This framework relies on the biometric specificity of human activity traits like walking, sitting… etc.

In [6], Smartphone-based gait authentication is evaluated under realistic circumstances by applying different types of attack. The authors developed a mobile application to collect users' gait data using a built-in tri-axial accelerometer located on a smartphone. They built their own dataset with 35 volunteers putting the smartphone on the trousers' pocket while walking to collect the accelerometer data. The gait data analyzed then the identity of each individual was established. They tested their work against the zero-effort attack which gained an equal error rate of 13%. They also tested their work against live impersonation attacks.

Mario et. al.[7] proposed a system that explores an individual's specific motion pattern and decides whether he is the smartphone owner or fraudulent. They used H-MOG dataset. It is a public and available dataset that contains gait data for 100 different volunteers. The used smartphone sensors were accelerometer, gyroscope, and magnetometer. They used Siamese CNN for extracting the learning features. After that, they trained a one-class SVM model with a radial basis function kernel. Their results were promising as the verification accuracy reached 97.8%.

The critical problems of walking detection and step counting have been tackled by [8]. In their prosed algorithm, the authors depend on short time Fourier analysis to obtain a convenient frequency representation for the person steps.

Also, some gait recognition researches [9] and [10], exploited the statistical power of information fusion to enhance the neural network based gait recognizers. Accordingly, the recognition accuracy is improved using support vector machine, SVM [9] and convolutional neural networks [10].

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

Wan et. al. [11] studied a detailed survey on gait recognition by making use of video, passing through floor sensors, and finished by accelerometer and gyroscope sensors. They also examined challenges and vulnerabilities in this field and proposed a set of future research directions.

## 3. CNN ARCHITECTURE

There are various types of convolutional neural networks, CNN[12]from which the underling CNN is adopted not only for its popularity but also for its suitability for gait recognition tasks. That CNN consists mainly of two parts, the first part is the convolutional part which is responsible for handling the various features of the input. The second part is the fully connected layer which is responsible for class prediction. The CNN is composed of several of layers, namely, convolutional, pooling (down sampling), flattening, and fully connected layers that will be explained below. Figure 1presents a CNN architecture in which the input maybe either visual or inertial. Actually, the visual input is not considered here while the readings of both accelerometer and gyroscope are casted in two vectors V1 and V2 respectively.

Their covariance matrix Cov(V1,V2) is computed by:

Cov(V1,V2) = E{(V1-E{V1})(V2-E{V2})T} , and $E\{V\} = \frac{1}{M}\sum_1^M v_i$ represents the expected value(the mean) in which M is the number of readings. Such value of Cov(V1,V2) tends to be negligibly small indicating that the two measures are independent. That observation opposes the assumptions given in[4].

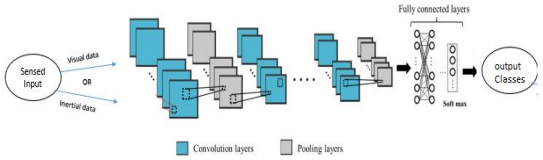



Figure 1 Proposed CNN architecture

### CONVOLUTIONAL LAYER

A convolution is a linear operation that involves the multiplication of a set of weights with the input, as in ordinary neural network. It is designed for two-dimensional input, the multiplication is performed between an array of input data and a two-dimensional array of weights, called a filter or a kernel. A convolutional layer contains a set of filters, the parameters of these filters need to be learned. The height and weight of the filters are smaller than those of the input volume. Filter goes through the width and height of the input then the computation of dot products between the input and filter are performed at each spatial position aiming at generating feature map. As an example, Figure 2 shows the feature map after finishing the convolutional operation.

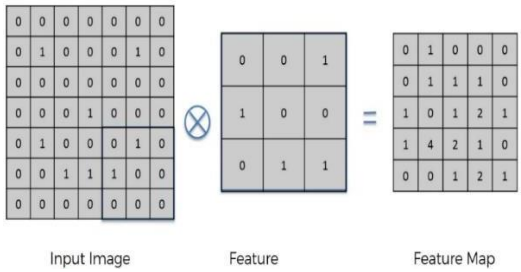



Figure 2 Feature map after convolution operation

### POOLING LAYER

The pooling layer is generally used to reduce the dimensions of data and computations, it is also called down sampling. Generally, down sampling is achieved by combining the output of neurons at one layer into a single neuron in the next layer. The pooling layer operates upon each feature map separately to create a new set of the same number of pooled feature maps. Pooling involves selecting a pooling operation, much like a filter to be applied to feature maps. The size of the pooling operation or filter is smaller than the size of the feature map; specifically, it is almost always 2×2 pixels applied with a stride of 2 pixels. This means that the pooling layer would often decrease the size of each feature map by a factor of 2, decreasing the number of pixels or values to one-quarter of the size of each feature map. For instance, if we have a pooling layer that is added to 6x6 (36 pixels), the resulting feature map will be a 3x3 pooled feature map (9 pixels). There are two common functions for calculating the pooling, average pooling, and max pooling. In average pooling, the average value of the region is calculated. Alternatively, in max pooling, the maximum value of the region is taken. Figure 3shows the pooled feature map (in case of max pooling).

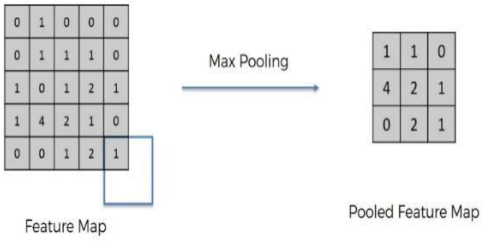



Figure 3 Pooled feature map

### FLATTENING LAYER

It is considered the last stage of CNN. Flattening operation from its name is to flatten our pooled feature map into a column. Just take the numbers row by row, and put them into this one long column. The

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

purpose of that step is that we want to later input this into the fully connected, (FC),layer for further processing. Figure 4 shows the flattening operation.

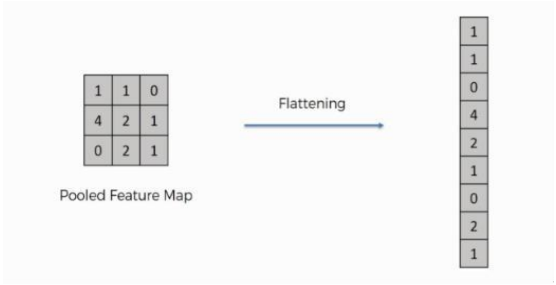



Figure 4 Flattening operation

**FULLY CONNECTED LAYER**

The last layer of CNN is the fully connected layer. In this layer, each input is connected to every output by a learnable weight. It is simply, a feed forward multi-layer neural network, its objective is to take the results of the convolutional part and use them for classification. Figure 5 shows a fully connected layer in which 10 classes are discriminated.

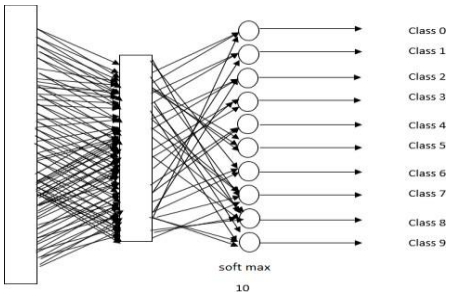



Figure 5 Fully connected, FC, layer

## 4. PROPOSED RECOGNITION TECHNIQUES

In this section three gait recognition techniques are explained, namely, saliency detection, offline and online learning models.

### 4.1 SALIENCY

Saliency refers to unique features (pixels, resolution, etc.) of the image in the context of visual processing. These unique features represent visually the most attractive locations in an image. The Saliency map depicts them topographically. So, the goal of the saliency map is to change the image representation into another representation. This new representation should be more meaningful and much easier to analyze. There are two different approaches for saliency detection, the first one is bottom-up and the other is top-down.

- Bottom-Up Approach

The bottom-up approach is the core of visual saliency, stimulus-driven signal that announces "this location is sufficiently different from its surroundings to be worthy of the attention". This bottom-up deployment of attention towards salient locations can be strongly modulated or even sometimes overridden by top-down, user-driven factors [13][14].

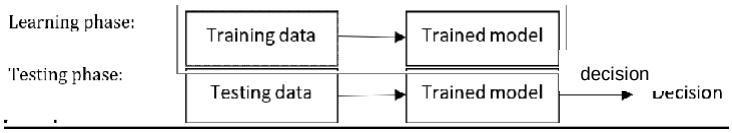
- Top-Down Approach

On the other hand, if one is looking through a child's toy bin for a red plastic dragon, amidst plastic objects of many vivid colors, no one color may be especially salient until the top-down desire to find the red object renders all red objects, whether dragons or not.

### 4.2 CONVOLUTION

As per the learning techniques, either offline or online, exploit transfer learning to provide the ultimate model. In this case, the CNN is pertained by a small subset of dataset persons to produce a fundamental model. Consequently, that model is transferred, as such, to the current problem of continuous authentication. *Figure 6* depicts the difference between offline and online learning approaches. The offline learning consists of two separate phases one for learning (offline) and the second for testing (real time). However, in online learning both learning and identification are carried out online based on an evolving model responsive to the actual real time gait of the underlying person.

**Offline Learning**




**Online Learning**

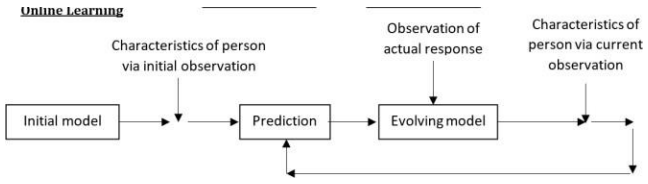



Figure 6 Difference between offline and online learning

#### 4.2.1 OFFLINE LEARNING

Such algorithm is designed to learn by example using a class of supervised learning. Before the training phase of the supervised learning algorithm, the data used for training will be formed as an input and output paired with it, Figure 6. During the training phase, the algorithms try to find a pattern in the training data that match the desired output. After the training phase, the

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

algorithms will take out of sight inputs and decide which input the new input will be classified according to prior training data, Figure 6. Here, batch learning is used in which the network is learnt on the entire training dataset at once in a series of epochs. Each epoch consists of one forward pass and one backpropagation pass. The true gradient can be computed by calculating the gradient value of each training case independently, then summing the resultant vectors together, this is known as full batch learning. In offline learning, it is required to train the predictor offline until reaches the acceptable accuracy, after that the final model is moved to the production stage.

### 4.2.2 ONLINE LEARNING

Online learning is the opposite technique of batch learning (offline learning). It is a common technique in machine learning. In online learning, data is fed to the predictor in sequential order which means we can update the model after every new instance. It is used when data does not fit into memory. It is also used when it is needed for the algorithm to adapt to new patterns in the data dynamically. It is also used when data is a function of time as stock prices. The algorithms of online learning may be prone to catastrophic interference, a problem that can be addressed by incremental learning approaches. However, there are many advantages for online learning for example it is data efficient and adaptable. Online learning is data efficient because the data no longer required once it has been consumed. Technically, this means you don't have to store your data. It is adaptable because it makes no assumption about the distribution of your data. Algorithms of online machine learning refine the models continuously. This happens by processing new data in real time then training the system to adapt to changing patterns and associations in the data[15].

Noteworthy to mention that online learning is a method of training in which data is available in a sequential order and is used to adapt the underlying model for new gait patterns. Typically learning one example at a time will take more steps to reach the same accuracy of offline learning, therefore the learning model is changed from the 2-phase model to the online evolving model, as shown in Figure 6. In algorithm 1, a sequence is applied to decide which learning point is visited in the present step. Accordingly, the model weights are updated as indicated by the algorithm

**Online learning algorithm pseudocode:**

Algorithm 1

1. Input: $w_0$
2. Output: $w_f$ (weights of the final model)
3. Execution:
4. Begin
5. Initialize: $w_t = w_0$
6. for $t$ = 1, 2, . . . , T do (where T is class x instances)
7. receive an incoming instance: $x_t \in X$ ;
8. predict the class label: $\hat{y}_t = sgn(f(x_t, w_t)) \in Y$;
9. //there exists 2 classes and 2 class labels {0,1}and $f(x_t, w_t) = \sum_1^T w * x$
10. reveal the true class label from the environment: $y_t \in Y$ ;
11. compute the suffered loss:

$l(w_t; (x_t, y_t))$;

12. $l(w_t; (x_t, y_t)) = -\sum_{i=1}^{2} b_i \log(p_i)$
$= -[b \log(p) + (1 - b) \log(1 - p)]$
13. // $b_i$takes a value 0 or 1 and pi is the softmax probability for i[th] class
14. If $(w_t; (x_t, y_t)) > 0$, $w_{t+1} \leftarrow w_t + \Delta(w_t; (x_t, y_t))$
15. $\Delta(w_t; (x_t, y_t)) = \eta \delta_t y_t$,
16. // where η is thelearning rate
17. // thus the learner updated the underlying model
18. end for
19. end

## 6. IMPLEMENTATION

### 6.1 SOFTWARE IMPLEMENTATION

#### 6.1.1 SETUP

Google Colab has been used as integrated development environment (IDE) with K80 GPU and 12 GB memory. It runs entirely in the cloud and provides access to powerful computing resources [16] . Upon loading the dataset, the batch size is 100, and the training epochs were set to 10. The programming language used is Python with machine learning libraries, including Keras and Tensorflow. All of these components are running on a Linux operating system. Consequently, The H-MOG dataset is employed for the underlying experiments. It has been collected and made available for public download in 2015. The dataset is described in details in [17] . The data was collected using the same smartphone model (Samsung Galaxy S4). It has been obtained using 10

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

Samsung Galaxy S4. That data was collected over multiple days, and the same user might have received a different device during each visit. These smart phones recorded accelerometer and gyroscope readings with a sampling rate of 100 Hz. Data is collected from 100 volunteers (53 male, 47 female). Each volunteer performed 24 sessions. Each session took from 5 to 15 minutes. All sessions were divided into 3 categories, 8 reading sessions, 8 writing sessions, and 8 map navigation sessions. Each volunteer contribution is about 2 to 6 hours of behavior traits. They recorded 9 categories of data from which we focused on the accelerometer and gyroscope readings. The total size of the dataset is 30 GB where the map navigation sessions only is extracted. Figure 7 illustrates the offline and online training phases that were executed on the collected data

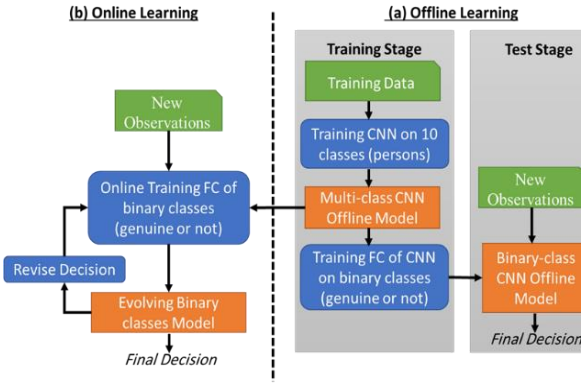



Figure 7 Components of software implementation

**EXPERIMENTS**

In what follows, the terms local and global features are extensively used, therefore, it is convenient to informally define them. Here, local features have their regions of interest in limited local areas and they are located at the feature maps (as output from convolutional layers). On the other hand, global features are the most salient features in the whole image (not part by part). Thus, they attract the viewer attention regardless of the image content, thus, if a gait energy image is considered, then the global feature is the corresponding silhouette while if the image is the spectral representation of person motion, then the global feature is the saliency map of the underlying frequency curves. The experiments that have been conducted can be categorized to three approaches, namely, saliency detection, offline learning and online learning. The first approach is saliency detection, the second approach is the offline learning, and the third approach is the online leaning, Figure 8. The first approach is the saliency detection. It depends on extracting the global features (saliency map) from the input then classifying it using a fully connected artificial neural network. The second and third approaches rely on CNN in the first stage and then use a fully connected layer in the second stage as shown in Figure 8.

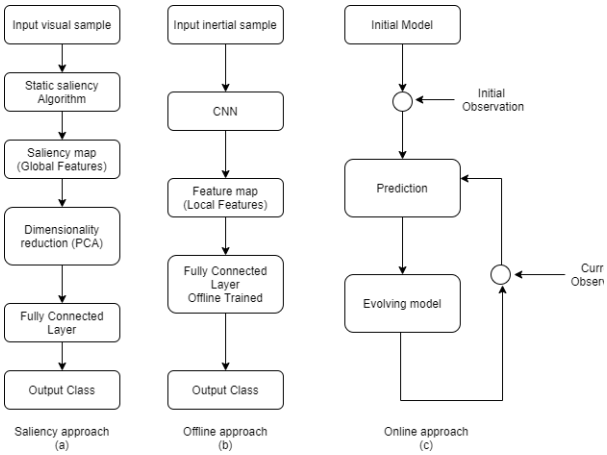



Figure 8 Proposed approaches

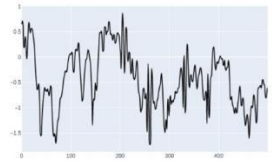
### 4.2.3 SALIENCY APPROACH IMPLEMENTATION

As shown in Figure 8-a, the input is fed to the static saliency detection algorithm[18] which is responsible for extracting the global features from the input samples, represented by the saliency map. For dimensionality reduction that saliency map is passed through a principle component analysis, PCA algorithm[19]to get the most important 512 features in the map. Consequently, the fully connected layers are trained using the output vector of the PCA algorithm. To facilitate a meaningful comparison with the other two approaches, only 512 values are adopted from the saliency map as global features that have been applied to the fully connected layer in order to obtain the output class.

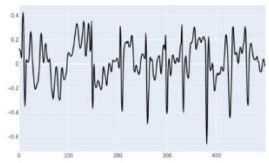
#### 4.2.3.1 DATASET PREPROCESSING

The dataset preprocessing, as such, implies converting every sample into an image that contains 6 curves. These curves are three accelerometer readings in X, Y, and Z directions as well as three gyroscope readings in X, Y, and Z directions, respectively. Figure 9 shows the curve of each space direction alone while Figure 10 shows the combination of one data sample.

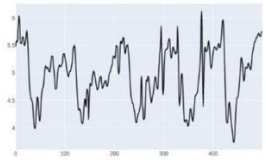
Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290



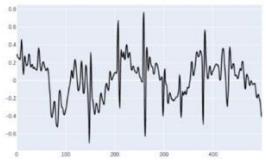
Acc_X



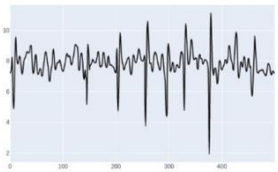
Gyr_X



Acc_Y



Gyr_Y



Acc_Z

Gyr_Z

Figure 9 The curve of each column in a data sample

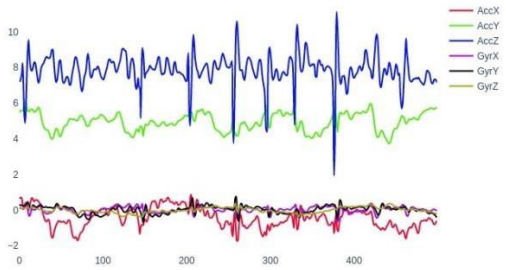



Figure 10 Image of one data sample

#### 4.2.3.2 Saliency Model

After getting the visual representation instead of the inertial data, that model has been implemented by making use of the static saliency algorithm of OpenCV library [20]. The saliency map of Figure 10 is illustrated in Figure 11.

11. Consequently, by making use of the PCA algorithm, the most important global features are achieved.

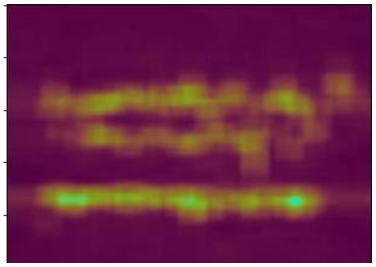


Figure 11 Saliency map

The confusion matrix resulted from this approach is shown in Figure 12.

| | |
|---|---|
| TP = 538 | FP = 85 |
| FN = 360 | TN = 252 |

Figure 12 Saliency model confusion matrix

Also, the following results are obtained.

- Precision: 0.665.
- Recall: 0.625.
- F1-score: 0.605.

By plotting the relation between the true positive rate (TPR) and the false positive rate (FPR), the area under the curve is obtained where AUC: 0.645, Figure 13. Also, Figure 14 shows the testing accuracy at resolution 64x64 pixels: 76.9%

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

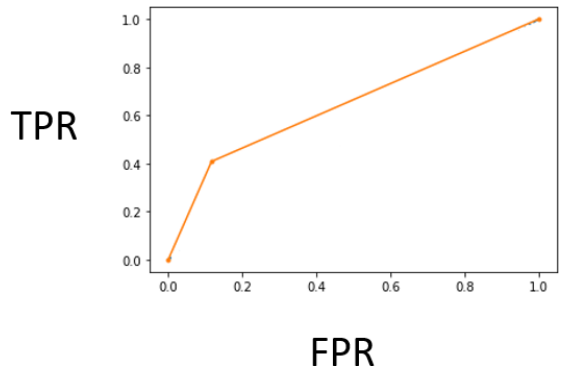



Figure 13 AUC curve for saliency approach

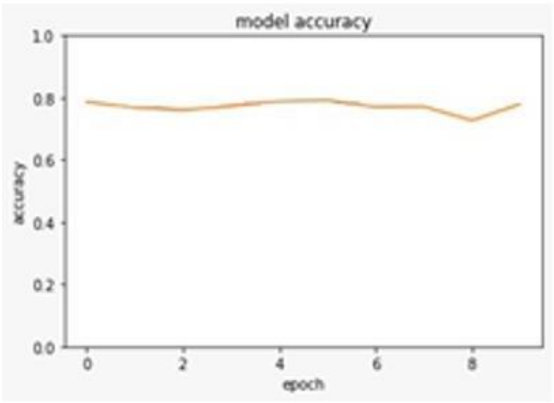



Figure 14 Accuracy of saliency model

Actually, [8]has made use of Fourier transform for walking representation and detection, however, its results are not discussed here but delayed to the comparative study of

Table 1 for convenience.

#### 4.2.4 OFFLINE APPROACH IMPLEMENTATION

In the offline learning model, Figure 8-b,the input sample is the inertial data, represented by a matrix of 500 rows and 6 columns. The six columns are the accelerometer readings in the X, Y and Z coordinates and Gyroscope readings in X, Y and Z coordinates, respectively. The frequency of both accelerometer and gyroscope is 100 Hz, thus, each sample takes 5 seconds. The second block in the offline approach is the CNN. It is used in order to get the feature map that represents the input sample. We started building the CNN with simple architecture. The simplest CNN architecture consists of input layer, one convolutional layer, one max pooling layer then the flattening layer. Input layer accepts a matrix of 500x6, while the convolutional layer consists of 32 parallel filters. The result of the convolutional layer will has a dimension of 500x6x32. After that, a max pooling is applied to the output of the convolutional layer, which is reduced to 250x3x32. Then a flattening layer is used to pass the data to the fully connected layer. That fully connected layer accepts 24000 features then reduces them to 512 features, in order to determine the class of the underlying features. In order to enhance the accuracy, new convolutional layers, max pooling layers are added and the accuracy is checked. When the architecture became 5 convolutional layers, 5 max pooling layers and one flatting layer, Figure 15, the network accuracy does not increase but saturated at 98.2%.

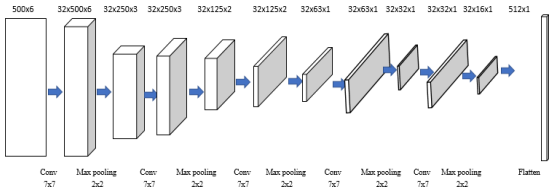



Figure 15 Last CNN proposed

Here, a pre-trained deep convolutional neural network is utilized as a basic architecture for our gait recognizer. It should be noticed that the underlying CNN is pre-trained using our data (not external data or models) for gait recognition i.e. the input is a matrix of (500x6) representing the underlying person and the output is its feature map after extracting the interesting features. Then, transfer learning is applied to adapt the underlying CNN to our gait recognition dataset. It is applied by freezing the whole CNN and retrain the fully connected layer on the target classes. After applying transfer learning we moved to binary classification in the fully connected, FC, layer in order to afford continuous authentication. The two classes were a new class from the dataset and the other class was collected from the 10 classes that the CNN trained on it before. The samples obtained from the target is considered a positive class while the samples collected from all other classes are negative. The total samples of the positive class are 3086, therefore, another 3086 negative samples are adopted. The data is splitted into two parts, 80% of it for training and the remaining 20% for testing.

The confusion matrix resulted from this approach is as follows: the true positive samples are 617, false positive samples are 11 samples, false negative samples are 11 samples, and true negative samples are 596 samples. Figure 16 describes the offline model confusion matrix.

| | |
|---|---|
| TP = 617 | FP = 11 |
| FN = 11 | TN = 596 |

Figure 16The confusion matrix of the offline model

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

Also, the following results are obtained.

- Accuracy: 98.2 %.
- Precision: 0.98.
- Recall: 0.98.
- F1-score: 0.98.
- Figure 17 shows the area under the curve, AUC: 0.982, compared with that of [4].

Also, a detailed comparison with[7]is given in

Table 1. The authors of [7] have utilized the HMOG dataset and used a Siamese neural network with four convolutional layers to extract the feature map. Then a PCA algorithm is applied to achieve dimensionality reduction. Then the PCA output is passed to one class support vector machine for classification. They chose different window sizes starting from 0.5 to 2 seconds. The true positive samples are 231683, false positive samples are 3217 samples, false negative samples are 7047 samples, and true negative samples are 227853 samples. In terms of accuracy, the proposed offline model is better than [7] as its accuracy varies from 95.8% to 96.3 depending on window size while the proposed offline model accuracy reaches 98.2 %. In terms of AUC, the AUC curve obtained by the proposed offline model is better than that of [4],Figure 17.

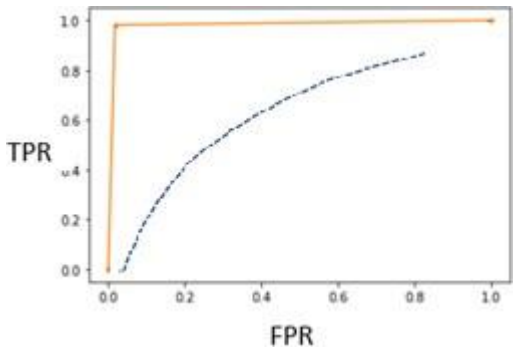



Figure 17 ROC curve of our offline model vs ROC curve of [4] model

### 4.2.5 ONLINE APPROACH IMPLEMENTATION

The main difference between the offline and online approach is that the fully connected layer of the online approach is trained online, Figure *8* Proposed approaches Figure 8-c. This means that the parameters and details of CNN in the online approach totally the same as the CNN of the offline approach. So we will not cover the CNN again in this section, we will cover only the online training of the fully connected layer then we will measure the performance of the online model. The total number of samples for each class was 3086. We used 80% of these samples for training and the rest 20% used for testing. In online training, there are no patches as the training takes place for each sample alone. The most important thing that distinguishes this method and makes it more reliable and applicable in real life is continuity. Continuity means the model is trained continuously after each new sample. This sample is classified by the model. After classification, the model retrains itself using this sample. Upon applying online learning the training took 49 seconds for 80% of our samples, then we tested it using the remaining 20% of the samples. The accuracy of the model was 98.7%. Figure 18 shows the accuracy of the online approach.

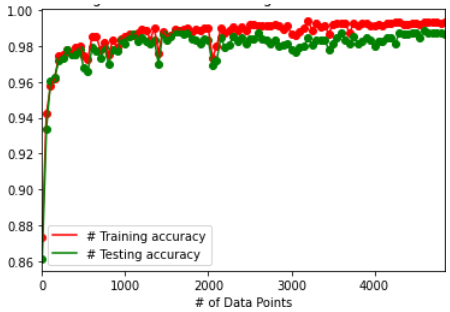



Figure 18 Online approach accuracy

The confusion matrix of this approach was as follow: The true positive samples are 608, false positive samples are 20 samples, false negative samples are 6 samples, and true negative samples are 601 samples. The confusion matrix obtained by the online model is also clearly better than the one obtained by[7].Figure 19shows the confusion matrix obtained by the online model.

| | |
|---|---|
| TP = 608 | FP = 20 |
| FN = 6 | TN = 601 |

Figure 19 Online model confusion matrix

Also, the following results are obtained.

- Accuracy: 98.7 %.
- Precision: 0.98.
- Recall: 0.98.
- F1-score: 0.98.
- Figure 20 shows the area under the curve, AUC: 0.979.

It is noted that the AUC curve obtained by online learning was also better than the offline one we proposed and also better than[4]. Figure 20

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

shows the AUC obtained by the online model.

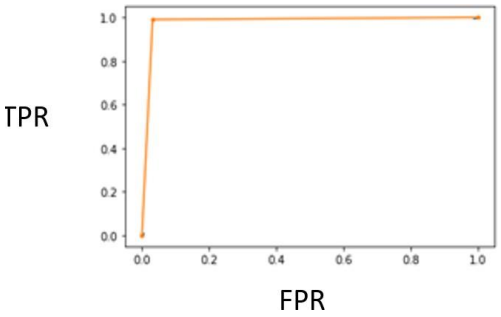



Figure 20 AUC curve for online model

### 4.3 HARDWARE IMPLEMENTATION

Since the hardware implementation is complementary to the software algorithms then Odroid XU4 has been chosen for hardware implementation as it affords a cost effective configuration. Odroid XU4 is a low-cost single board computer with credit card-sized device that plugs into a computer monitor or TV and uses a regular keyboard and mouse. Offering open source support, the board can run a variety of Linux distributions, including the latest Ubuntu 20.04 and Android 4.4 KitKat. Figure 21 shows Odroid XU4, its specifications includes Samsung Exynos5422 ARM® Cortex™-A15 Quad 2.0GHz/Cortex™-A7 Quad 1.4GHz as a CPU. It also include a 3D Accelerator of Mali™-T628 MP6 OpenGL ES 3.1 / 3.0 / 2.0 / 1.1 and OpenCL 1.2 Full profile, 2 GB RAM. There is also wireless LAN and Bluetooth Low Energy, Ethernet, 3 USB 2 ports, Full-size HDMI, micro SD port for storing data and loading the operating system, and its size is 83 x 58 x 20 mm[21], [23]. We used odroid 4.14.127 Linux distribution as an operating system. This distribution is specifically made for odroid boards. Odroid XU4 took 76 seconds to finish the training of the online approach and 70 seconds for the offline approach. However, it took 240 seconds to finish the training for the saliency approach. In terms of accuracy, the accuracy was 98.1% and 96% for the online and the offline approaches respectively. However, it dropped to 75.7% in the case of the saliency approach.

Figure 21Odroid XU4

Table 1 Comparison of gait recognition results.

| Approach / Metric | Online Learning | Offline Learning | | | Data Fusion |
|---|---|---|---|---|---|
| | [ours] | [ours] | [7] | [4] | [10] |
| Accuracy | 98.7% | 0.98.2% | 96.3% | 0.97% | 0.97% |
| Precision | 0.98 | 0.98 | 0.986 | N/A | N/A |
| Recall | 0.98 | 0.98 | 0.970 | N/A | N/A |
| F1-score | 0.98 | 0.98 | 0.98 | N/A | N/A |
| AUC | 0.979 | 0.989 | N/A | 0.96 | N/A |

Eventually,

Table 1 is constructed to summarize the results of our experiments and to compare them with [4],[7],[8], [9], and [10]. Actually, these research works are adopted because of the following results.

[4]presents a similar approach to ours but the CNN based architecture that has been used is complicated by recurrent sub networks represented by RNN and LSTM. In [7] the authors, also, have relied upon a deep learning approach in which Siamese Network and one class support vector machine have been exploited to achieve gait recognition.[8] used short time Fourier transform to obtain a spectral representation for the inertial measurements. [9] and [10] depend upon information fusion to improve the performance of the gait recognizer. Thus in

Table 1the following comments might be stated for clarification.

- For saliency detection, the inertial data, represented by the accelerometer and gyroscope reading is converted to visual representation in order to be entered properly to the open CV static saliency detection algorithm [20]. Accordingly, Figure 9 and Figure 10 show the frequency variations in X, Y, and Z axes of acceleration and angular motion, respectively. The corresponding results are reported in the
- Table 1as metric values, for convenience.
- It is obvious that the global features measured by the visual salient changes are less discriminative, consequently achieve less accurate results than the local features measured by the variation in acceleration and angular motion. The reason is clear, the inherent features obtained from the motion itself might be more specific, interpretive and descriptive than the corresponding motion appearance expressed by the motion spectral image.
- [4] and [10] use the same approach that has been used here, namely, deep learning. However, the comparison emphasizes that the architectural complications in [4] and the statistical complexity

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

[10] do not achieve better accuracy than that of online learning.

## 5. DISCUSSION

Since human gait is unobtrusive and difficult to conceal, it has been recommended by numerous researchers as a reliable biometric for continuous authentication. As those researchers are coming from different origins, different results are obtained, some of them are consistent and some of them are not. A comparison of our work with these results points out the following.

i. It has been found out that online learning, due to its adaptability, flexibility and less complexity is a key factor for robustness and accuracy increase.
ii. The confusion matrices, depicted here, indicate that our proposed CNN can be used successfully for both authentication and identification. On the contrary, the suggested deep learning architecture of [4] is confined to continuous authentication only also, the authors of [4] have chosen a combination of CNN+LSTM as a preferable architecture for gait recognizers. Actually, that architecture is complex and our results,
iii. Table 1, indicated that, although the LSTM part adds tremendous complications, it has negligible impact on the accuracy improvement.
iv. The use of RNN and LSTM neural networks is not recommended since they have feedback passes that principally slow down the speed of the recognition process. Also, for those who plan to achieve a hardware accelerator for their software recognizer design all feedback loops have to be excluded since they lead to hardware complications caused by the corresponding sequential circuits required to represent the recognizer states.
v. The results have been obtained from global features using saliency are consistent with that of [22], [24].Despite the fact that our saliency detection and gait energy image approaches exploit global features to perform identification, however, one should be careful that the gait energy image global features represent the person silhouette while the global features of the saliency detection algorithm are the limb linear acceleration and the foot angular motion providing a motion spectral image. Although, each set of features is global in a different sense both are used in different applications in order to utilize their ability to extract gait features from the visual input representation.
vi. In practice, two questions may arise against the online approach:
    1) What if the limb of the authenticated person is injured? and
    2) What if the smartphone is abused by a person who is not the authenticated one?

The first problem is solved by identifying the authenticated person not only by his normal motion but also by his abnormal and injured-like gait. While the second problem is solved by making use of the underlying transfer learning. In this case, the pertained network is transferred to a new target aiming at the recognition of the authenticated person only and only that person.

## 6. CONCLUSION

This paper has presented the essential results of a research on gait recognition using a deep learning approach. It combines both theoretical foundations and practical performance in an empirical approach that can be useful for gait analysts. The conclusion of our work is pointed out in the following.

1- Deep online learning can be employed efficiently to achieve gait recognition for continuous authentication. Actually the experimental results put forward online learning as a training scheme for future CNN-based gait recognizers because it is adaptable, reliable, accurate and easy to apply.
2- For continuous authentication, inertial data could achieve impressive results. The accelerometer and gyroscope readings are not only orthogonal but also they can express the gait details more accurately than visual data.
3- The use of local features (via a convolutional approach) is superior to using global features (via a saliency detection approach) for person identification using gait recognition.
4- The experimental results have emphasized that the accuracy of saliency detection is comparable only with that of offline learning. The reason is clear, in both cases they suffer from a serious inflexibility since their resultant models are difficult to change or to adapt with motion changes.
5- The use of the cheap and simple Odroid XU4 GPU kit has confirmed the hardware applicability of the proposed online gait recognition algorithm for real time applications.

Actually, for security applications, the use of CNN whether based on online or offline learning is considerably effective for gait features extraction. It offers a preferable approach for continuous authentication, since, the application data either, inertial or visual can be reliably used.

## 7. COMPLIANCE WITH ETHICAL STANDARDS


- Funding: no funding available
- Conflict of Interest
    1. First author: no conflict of interest
    2. Second author: no conflict of interest
    3. Third author: no conflict of interest
    4. Fourth author: no conflict of interest


## REFERENCES

[1] Z. Sitova, J. Sedenka, and Q. Yang, “HMOG: New Behavioral Biometric Features for

Volume: 14 Issue: 04 Pages: 5523-5534 (2023) ISSN: 0975-0290

Continuous Authentication of Smartphone Users," IEEE Trans. Inf. Forensics Secur., vol. 11, no. 5, pp. 877–892, 2016, doi: 10.1109/TIFS.2015.2506542.

[2] W. H. Lee and R. B. Lee, "Implicit Smartphone User Authentication with Sensors and Contextual Machine Learning," *Proc. - 47th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Networks, DSN 2017*, pp. 297–308, 2017, doi: 10.1109/DSN.2017.24.

[3] M. Ehatisham-Ul-Haq, M. A. Azam, U. Naeem, S. U. Rehman, and A. Khalid, "Identifying Smartphone Users based on their Activity Patterns via Mobile Sensing," *Procedia Comput. Sci.*, vol. 113, pp. 202–209, 2017, doi: 10.1016/j.procs.2017.08.349.

[4] Q. Zou, Y. Wang, Q. Wang, Y. Zhao, and Q. Li, "Deep Learning-Based Gait Recognition Using Smartphones in the Wild," *IEEE Trans. Inf. Forensics Secur.*, pp. 3197–3212, 2020, doi: 10.1109/tifs.2020.2985628.

[5] R. Damaševičius, R. Maskeliunas, A. Venčkauskas, and M. Woźniak, "Smartphone user identity verification using gait characteristics," *Symmetry (Basel).*, vol. 8, no. 10, 2016, doi: 10.3390/sym8100100.

[6] M. Muaaz and R. Mayrhofer, "Smartphone-Based Gait Recognition: From Authentication to Imitation," *IEEE Trans. Mob. Comput.*, vol. 16, no. 11, pp. 3209–3221, 2017, doi: 10.1109/TMC.2017.2686855.

[7] M. P. Centeno, Y. Guan, and A. van Moorsel, "Mobile based continuous authentication using deep features," *EMDL 2018 - Proc. 2018 Int. Work. Embed. Mob. Deep Learn.*, pp. 19–24, 2018, doi: 10.1145/3212725.3212732.

[8] X. Kang, B. Huang, and G. Qi, "A novel walking detection and step counting algorithm using unconstrained smartphones," *Sensors (Switzerland)*, vol. 18, no. 1, 2018, doi: 10.3390/s18010297.

[9] Z. He, "Accelerometer based gesture recognition using fusion features and SVM," *J. Softw.*, vol. 6, no. 6, pp. 1042–1049, 2011, doi: 10.4304/jsw.6.6.1042-1049.

[10] O. Dehzangi, M. Taherisadr, and R. ChangalVala, "IMU-based gait recognition using convolutional neural networks and multi-sensor fusion," *Sensors (Switzerland)*, vol. 17, no. 12, 2017, doi: 10.3390/s17122735.

[11] C. Wan, L. Wang, and V. V. Phoha, "A survey on gait recognition," *ACM Comput. Surv.*, vol. 51, no. 5, 2018, doi: 10.1145/3230633.

[12] Z. Li, W. Yang, S. Peng, and F. Liu, "A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects," Apr. 2020.

[13] R. Desimone and J. Duncan, "Neural mechanisms of selective visual attention," *Annu. Rev. Neurosci.*, vol. 18, no. February 1995, pp. 193–222, 1995, doi: 10.1146/annurev.ne.18.030195.001205.

[14] L. Itti and C. Koch, "Computational modelling of visual attention," *Nat. Rev. Neurosci.*, vol. 2, no. 3, pp. 194–203, 2001, doi: 10.1038/35058500.

[15] J. Hurwitz and D. Kirsch, *Machine Learning For Dummies®, IBM Limited Edition*. New York: John Wiley & Sons, Inc., 2018.

[16] E. Bisong, *Building Machine Learning and Deep Learning Models on Google Cloud Platform*. Berkeley, CA: Apress, 2019.

[17] Q. Yang, G. Zhou, and Z. Sitová, "A multimodal data set for evaluating continuous authentication performance in smartphones," *SenSys 2014 - Proc. 12th ACM Conf. Embed. Networked Sens. Syst.*, no. 1, pp. 358–359, 2014, doi: 10.1145/2668332.2668366.

[18] M.-M. Cheng, Z. Zhang, W.-Y. Lin, and P. Torr, "BING: Binarized Normed Gradients for Objectness Estimation at 300fps," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 3286–3293, doi: 10.1109/CVPR.2014.414.

[19] I. Jolliffe, "Principal Component Analysis," in *International Encyclopedia of Statistical Science*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 1094–1096.

[20] X. Hou and L. Zhang, "Saliency Detection: A Spectral Residual Approach," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2007, pp. 1–8, doi: 10.1109/CVPR.2007.383267.

[21] J. Ivković, A. Veljović, B. Ranđelović, and V. Veljović, "ODROID-XU4 as a desktop PC and microcontroller development boards alternative," *Tech. Informatics Educ.*, no. May, pp. 439–444, 2016.

[22] J. Han and B. Bhanu, "Individual recognition using gait energy image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 2, pp. 316–322, 2006, doi: 10.1109/TPAMI.2006.38.

[23] Özbay, Erdal, and Feyza Altunbey Özbay. "A CNN Framework for Classification of Melanoma and Benign Lesions on Dermatoscopic Skin Images." International Journal of Advanced Networking and Applications 13.2 (2021): 4874-4883..

[24] Mubassıra, Masiath, and Amit Kumar Das. "Implementation of Recurrent Neural Network with Language Model for Automatic Articulation Identification System in Bangla." International Journal of Advanced Networking and Applications 12.6 (2021): 4800-4808..