Ad Hoc Networks 107 (2020) 102218

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

# Gait-Watch: A Gait-based context-aware authentication system for smart watch via sparse coding

Weitao Xu[a,*], Yiran Shen[b], Chengwen Luo[c], Jianqiang Li[c], Wei Li[d], Albert Y. Zomaya[d]

[a] Department of Computer Science, City University of Hong Kong, Hong Kong
[b] Data61, CSIRO, Australia
[c] School of Computer Science and Software Engineering, Shenzhen University, China
[d] School of Information Technologies, University of Sydney, Australia

A R T I C L E I N F O

*Article history:*
Received 9 January 2020
Accepted 20 May 2020
Available online 4 June 2020

*Keywords:*
Wearable devices
Authentication
Gait recognition
Sparse representation

A B S T R A C T

In recent years, wrist-worn smart devices such as smart wrist band and smart watch have pervaded our everyday life. Under this trend, the security issue of these wearable devices has received considerable attention as these devices usually store various private information. Conventional methods, however, do not provide a good user experience because they either depend on a secret PIN number input or require an explicit user authentication process. In this paper, we present Gait-watch, a context-aware authentication system for smart watch based on gait recognition. We address the problem of recognizing the user under various walking activities (e.g., walking normally, walking upstairs and walking with calling the phone), and propose a feature extraction method from gait signals to improve recognition accuracy. Extensive evaluations show that Gait-watchimproves recognition accuracy by up to 30.2% by leveraging the activity information, and can achieve 3.5% Equal Error Rate (EER). We also report a user study to demonstrate that Gait-watchcan accurately authenticate the user in real-world scenarios and require low system cost.



## 1. Introduction

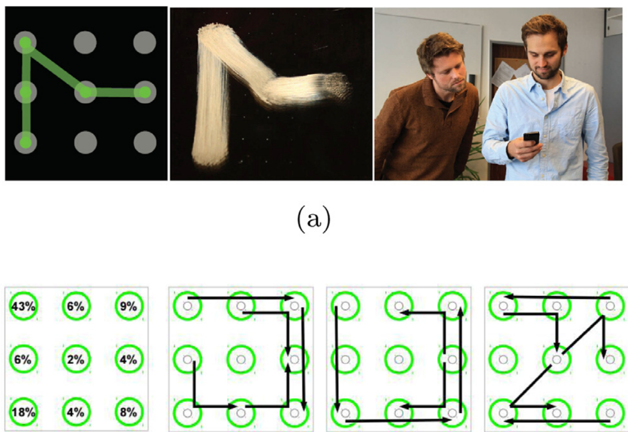
With recent advances in embedded computing technologies, smart wearable devices such as Apple Watch and Fitbit wristband, have become increasingly popular and play significant roles in our daily lives. With the pervasiveness of these devices, security is becoming crucial over time as they accumulate a large variety of sensitive data about the users. In particular, with sensors embedded in smart watches, the collected sensory data can be explored for the understanding of user's physical and mental health states. For instance, the accelerometer information collected by the smart watch can be mined to uncover user's daily life activities [1].

Traditional authentication methods such as passwords do not offer good user experience because of the need for keyboard. A mostly common deployed method in current smart watches is the so called Unlock Pattern scheme as in Fig 1(a). In this scheme, the user is presented a $3 \times 3$ grid and the secret (password) of a user is a drawing on that grid (i.e., a sequence of lines connecting the dots). During enrollment, a user has to choose a pattern and during the authentication phase, he has to recall his pattern and draw it on the screen. However, the pattern-based unlocking scheme has three major weaknesses. First, they are susceptible to smudge attacks, where imposters extract sensitive information from recent user input by using the smudges left by fingers on touch screens. Recent studies have shown that finger smudges (i.e., oily residues) of a legitimate user left on touch screens can be used to infer pattern [2]. In addition, they are susceptible to shoulder surfing attacks. Smart watches are often used in public settings (such as subway stations, schools, and cafeterias) where shoulder surfing often happens either purposely or inadvertently, and patterns are easy to spy [3]. Third, despite of numerous possibilities in pattern selection, researchers have found that there is a high bias in the pattern selection process, e.g., people often pick the top left corner as a starting point and prefer straight lines in their pattern. The results in [4] indicate that the security offered by this scheme is less than the security of only three digit randomly-assigned PINs for guessing.

Motivated by the above issues, we aim to develop an unobtrusive, continuous, and implicit authentication system for smart watch based on gait recognition. Biometric gait recognition refers to verifying or identifying persons by their walking style. Extensive studies from psychology and biometrics have demonstrated that biometric gait contains distinctive patterns that can be used for

* Corresponding author.
*E-mail addresses:* weitaoxu@cityu.edu.hk (W. Xu), yiran.shen@csiro.au (Y. Shen), chengwen@szu.edu.cn (C. Luo), lijq@szu.edu.cn (J. Li), weiwilson.li@sydney.edu.au (W. Li), albert.zomaya@sydney.edu.au (A.Y. Zomaya).

https://doi.org/10.1016/j.adhoc.2020.102218
1570-8705/




(a)

(b)

**Fig. 1.** Android pattern lock authentication system [4].

security purposes [5–7]. Gait based authentication offers several advantages over traditional authentication system. For instance, it is non-intrusive, does not require explicit authentication process, and provides continuous authentication while walking. On the other hand, gait, as a biometric trait, is hard to be forged and replicated. An imposter can observe how the genuine user walks but still have difficulty in replicating the walking patterns.

Although gait-based recognition has been well explored in the literature, there remains several challenges on smart watch. Firstly, due to the high freedom of arm, people could walk in a variety of ways (e.g., walk normally or walk with calling the phone). The large majority of existing studies on accelerometer-based gait recognition have used a very restrictive experimental setup where the performance evaluation was conducted on a dataset collected from a controlled laboratory environment and the participants are asked to walk normally. As the pervasiveness of smart watches in the wild, there is a need for robust and efficient authentication system in a realistic environment. Besides, the smart watches have limited energy and moderate computing power, thus to improve energy efficiency and reduce computational cost is a crucial task for authentication system on resource-constrained smart watches. Although deep learning-based technology can achieve high recognition accuracy [8,9], it is computationally expensive and usually used in cloud computing architecture which depends on reliable wireless connection.

In this paper, we propose a context aware gait-based authentication system on smart watches. Specifically, we implement an activity detector in the system to deal with different activities, and then identification is performed on corresponding training dictionaries according to the output of activity detector. Moreover, inspired by the success of Scale Invariant Feature Transform (SIFT) [10] for images feature extraction, we propose a novel feature extraction method to represent the characteristics of gait signals. The method is able to extract discriminative features of gait dynamics by searching extrema in the scale space of gait acceleration signals. We further apply sparse coding scheme and a novel probabilistic sparse representation classification to improve the recognition accuracy. To the best of our knowledge, this is the first work for gait-based authentication using commercial smart watches. The main contributions of this paper are threefold:

- We propose a context-aware authentication system in which the authentication system infers and leverage activity information during authentication. Evaluation results show that compared to the traditional methods which do not take the activity into consideration, the improvement of recognition accuracy can be up to 30.2%.
- We propose a novel feature extraction method which can extract robust and discriminative features from gait signals. We further employ a sparse coding scheme to build the training model based on the features extracted from each user. Then the identity is recognized based on a novel probabilistic sparse representation classification (PSRC). Evaluation results show that Gait-watchis able to achieve 97.3% recognition accuracy and 3.5% EER.
- We implement Gait-watchon Samsung smart watch, and conduct a user study to evaluate the performance in real world environments. The results show that Gait-watchcan authenticate the genuine user with 95.3% true positive rate. We also report the system overhead to demonstrate the feasibility on contemporary smart watches.

Partial and preliminary results of this paper have appeared in our previous work [11]. Comparing to the conference version, this paper contains significant new contributions which are listed as follows.

1. We present a novel feature extraction method which can extract robust and discriminative features from gait signals (Section 4.2).
2. We employ a sparse coding scheme to build subject-specific training model based on the features extracted from each user. The authentication is then performed on the training model by leveraging the activity information (Section 4.3.1).
3. We apply a novel probabilistic fusion model to further improve recognition accuracy (Section 4.3.3).
4. Compared to previous conference version, we extend the dataset from 20 subjects to 36 subjects. Then we re-do all the experiments and add several new experiments to evaluate the performance of the new system. Evaluation results show that Gait-watch can achieve $17-20\%$ higher accuracy than two competing gait recognition systems. Moreover, it is able to achieve 97.3% recognition accuracy which is 4.6% better than the previous conference paper (Section 5). We add a new subsection named validity analysis (Section 6.4) to discuss the potential threats in the design and the methods of our study. Finally, we significantly extend related work and discuss more recent works in terms of gait recognition and feature extraction in activity classification (Section 2).

The rest of this paper is organized as follows. Section 2 discusses the related work. We provide an overview of Gait-watchin Section 3 and detail the system architecture in Section 4. In Section 5, we evaluate the performance of Gait-watchon datasets. We then implement the system on smart watches and conduct user study to evaluate the system in Section 6. Finally, Section 7 concludes the paper.

## 2. Related work

In this section, we discuss the related work in three aspects: gait recognition, applications of SRC and feature extraction for activity classification.

**Gait Recognition.** Gait recognition has been well studied in the literature. From the way how gait is collected, gait recognition can be categorized into three groups: vision based, radio signal based, floor sensor based, and wearable sensor based. In vision based gait recognition system, gait is captured from a remote distance using video-camera. Then, video/image processing techniques are employed to extract gait features for further recognition. A large portion in the literature belong to this category [12,13]. Recent works also explore the feasibility of using radio signal to identify people [14,15]. In floor sensor based gait recognition, sensors (e.g., force plates), which are usually installed under the floor, are used for capturing gait features, such as ground reaction force (GRF) [16,17] or heel-to-toe ratio [18].

Compared with vision-based and other non-accelerometer based gait measurements, acceleration can reflect the dynamics of gait more directly and faithfully. For instance, accelerometer based gait recognition do not suffer from the existing problems for vision-based methods, like occlusions, clutter, and viewpoint changes. Existing works of wearable sensor based gait recognition are mainly based on the use of body-worn accelerometers. The first work of accelerometer based gait recognition is proposed by Ailisto et al. [19] and further developed by Gafurov et al. [20]. In the initial stages, dedicated accelerometers were used and worn on different body positions, such as lower leg [20], waist [19], hip [21], hip pocket, chest pocket and hand [22]. With the prevailing of smartphone, researchers have proposed several gait-based authentication systems by utilizing the built-in accelerometer [5,6,23,24]. For example, in [23], the researchers have proposed an unobtrusive gait verification system for mobile phones. In [24], the authors studied the impact of different phone locations such as hand and pocket. In recent years, researchers start to use emerging energy harvester to achieve gait recognition [25]. In addition, the unique gait has been exploited for key generation to protect user's personal devices [26–28].

Recently, there are several papers study gait-based authentication system for wrist-worn devices [29–32]. Compared to smartphones, wrist-worn devices have several advantages because users almost always wear their watch in the same location and orientation. The authors of [29] studied smartwatch-based biometric gait recognition. However, they only use simple statistical features such as mean and stand deviation and traditional classifiers such as random forest and Naive Bayes. [30] focuses on studying the performance of same-day, mixed-day and cross-day experiments.

Although various feature selection and classification methods have been proposed, most of the existing studies assume that the user is walking normally except [33]. In [33], the researcher aim to identify users based on the way during their multiple activities include walking, jogging, climb up stairs, and climb down stairs. However, the recognition accuracy is relatively low (70% − 90%). To the best of our knowledge, this is the first work for gait-based authentication system that considers various activities. To overcome the challenge of various activities, we propose a context-aware authentication system and a *sparse fusion* method to improve the recognition accuracy.

**Applications of SRC.** SRC is an emerging classification method and has been widely used in recognition tasks of sensor areas. Several papers have exploited the sparsity of multiple measurements to improve the system performance. [34] used CS to compress GPS signals and exploits the information of various propagation paths to improve the SNR of GPS signals. In [35], the researchers proposed opti-SRC by optimizing the random matrix used in SRC to increase the performance of face recognition system in smartphones. In [36], the authors improved face recognition accuracy by fusing several channel state information (CSI) vectors. The authors of [37] improve the performance of SRC by using a weighted sparse neighborhood-preserving projections. The authors in [38] improve recognition accuracy by exploiting the sparse representation of several face images from different views. In [39], the authors developed an acoustic classification system on wireless sensor networks by applying SRC to improve the recognition accuracy.

**Features Extraction for Activity Classification.** With the prevalence of wearable sensors and devices, activity classification using embedded sensors has drawn more and more attention. The flow chart of sensor-based activity recognition is similar, and usually include signal pre-processing, feature extraction, and classification using machine learning methods [40]. Traditionally, different types of features are combined together to achieve high accuracy such as time-domain features, frequency-domain features and heuristic features [41,42]. The adequate combination of features is a crucial task as the classification accuracy highly depends on a good representation. Inspired by the feature extraction algorithms in computation vision community [10], we present a novel feature extraction method that can extract discriminative features for each subject. Similar to SIFT which is robust to image scale and rotation, the proposed method is robust to changes of gait velocity and magnitude. The localisation of the feature points in [43] and our method are all based on SIFT, but the description of the features are different: the descriptors of our method are calculated from the gradient while the method of [43] calculates descriptors directly from the $\sigma$-scale of the temporal signal. Another issue in activity recognition is that activity classification results are sensitive to sensors' orientations. To address this issue, researchers either use orientation-independent features or use signal transformation to counter orientation changes. Such problem does not apply to smart watch as smart watch is always worn on one of user's wrist. However, user may perform different activities due to the high freedom of arm. In this paper, we address the problem of gait recognition under different activities by building independent training models for each activity.

## 3. System overview

In this section, we will describe the architecture in details. As shown in Fig 2, the flow chart of Gait-watchconsists of the following components.

***Offline Training.*** During the offline dictionary training phase, gait signals are denoised via a moving average filter. After denoising, feature extraction algorithm is applied on the filtered signal to obtain a cluster of feature descriptors. Then the feature descriptors are used to build a training dictionary $A$ via sparse coding scheme. After obtaining $A$, the column reduction algorithm [39] is applied to obtain a optimized training dictionary $\bar{A}$. Then the training dictionary $\bar{A}$ is used in the classifier as explained in Section 4.3.3.

***Online Processing***

**Walking Detector** To save energy consumption and extend battery life, Gait-watchis only activated when the user is walking. Therefore, a walking detector is firstly applied to indicate whether the subject is walking or not. Once Gait-watchdetects the user is walking, the accelerometer data will be processed further.

**Activity Classifier** In a realistic environment, the user can perform different activities during walking, which poses a major challenge for gait based recognition on smart watch. To deal with this issue, Gait-watchadopts an activity classifier to infer the specific activity of the user and leverages the activity information while performing recognition.

**Feature Extraction** After walking detector and activity classifier, feature extraction is applied to obtain a cluster of feature descriptors from the test signal (accelerometer data). The feature descriptors can effectively represent the walking pattern of the genuine user and are discriminative between different people. After this step, we will obtain a cluster of feature descriptors.

**Probabilistic Sparse Representation Classification (PSRC)** Now both the training dictionary $\bar{A}$ and the feature descriptors $y_i$ are passed to the classifier. The $\ell_1$ classifier first finds the sparse coefficient vector $x_i$. Then the vectors of different gait cycles are fused based on a novel *probabilistic fusion* model, and the fused sparse vector is used to calculate the residuals and confidence level. Finally, the confidence level is used to recognize whether the walker is the genuine user or imposter.

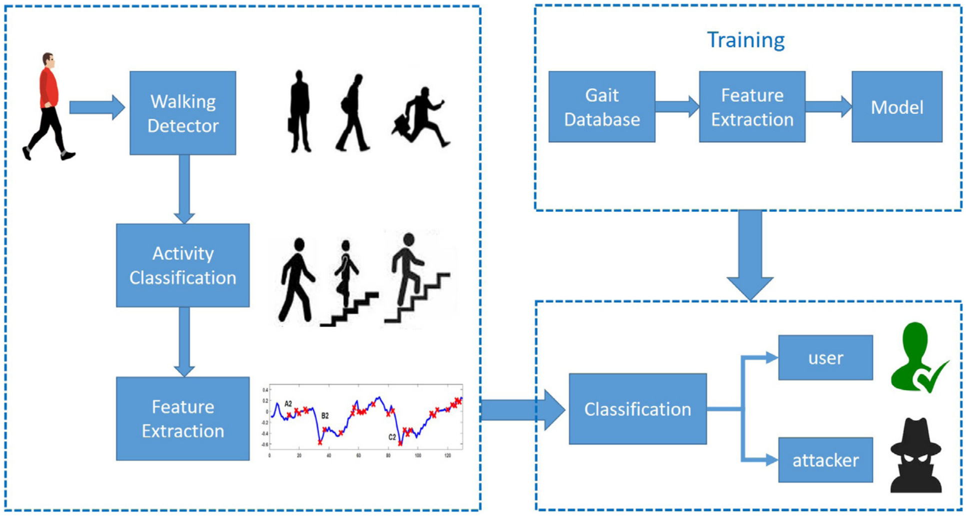



**Fig. 2.** System architecture..

## 4. System design

In this section, we detail the design of signal pre-processing, offline dictionary training, and classification in turn.

### 4.1. Signal pre-processing

#### 4.1.1. Walking detector

As the first step, walking detector is essential to avoid applying expensive recognition algorithms during periods of non-motion. The raw accelerometer signals along three axises of smart watch fluctuates greatly and irregularly due to the changing direction of smart watches and arbitrary body movements. However, we observed that the acceleration along gravity direction exhibits regular patterns because of the repetitive nature of walk. Fig. 3 depicts the acceleration values along gravity of different activities. We can see that the acceleration along gravity shows a rhythmic pattern because of the repetitive nature of walk. The intuition is that the smart watch bounces maximally along the gravity dimension, and the bounce-peaks correspond well with the heel strikes. Based on this observation, we apply the method in [44] on acceleration along gravity direction to detect whether the user is walking or not. It is worth mentioning that the repetitive nature of walk was first utilized in NASC [44] to count steps on unconstrained smartphones. We borrow their idea and make some improvements. We give a description of the improved walking detection method in Algorithm 1.

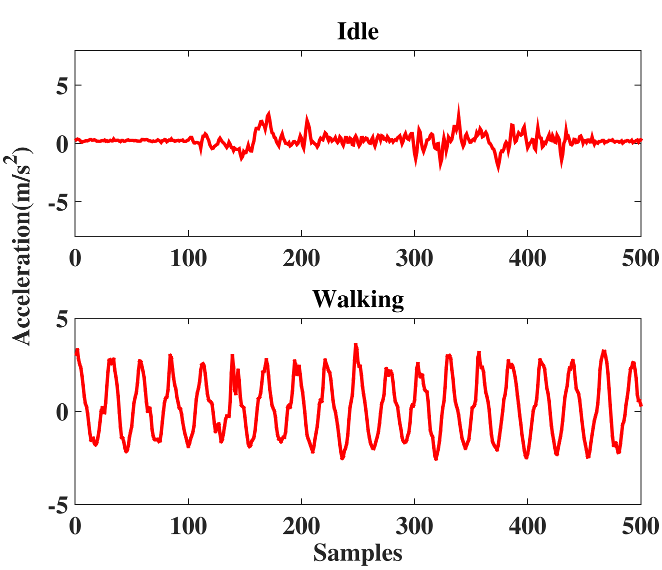



**Fig. 3.** Acceleration along gravity.

**Algorithm 1:** Walking Detection.

**Input:** acceleration samples $a(i)(1 \le i \le N_{acc})$, lag $\tau$;
**Initialization:** search window $(\tau_{\min}, \tau_{\max}) = (40, 65)$, walking frequencies $(f_{\min}, f_{\max}) = (1.2, 2.3)$, auto-correlation threshold $R_{th}$=0.7, spectral energy threshold $F_{th} = 400$, *STATUS=NOT WALKING*;
**for** $i = 1 : N_{acc}$ **do**
  **for** $\tau = \tau_{\min} : \tau_{\max}$ **do**
    $R(i, \tau)$=Auto-correlation$(i, \tau)$;
  **end**
  $R_{\max}(i) = max\ R(i, \tau)$;
  **if** $R_{\max}(i) \ge R_{th}$ **then**
    $F_f = STFT(a(j))\ (1 \le j \le N)$;
    $F_{walk} = \sum_{f=f_{\min}}^{f=f_{\max}} F_f$;
    **if** $F_{walk} \ge F_{th}$ **then**
      *STATUS*=WALKING;
      $\tau_{opti} = arg\,max_{\tau}\ R(i, \tau)$;
      $\tau_{\min} = \tau_{opti} - 15$, $\tau_{\max} = \tau_{opti} + 15$;
    **end**
  **end**
**end**
**Output:** *STATUS*

The initial status of the user is set as *NOT WALKING* (include *IDLE* and *RUNNING*). Given an accelerometer signal sequence $a(t)(t = 1, 2, \cdots, N_{acc})$, we compute the normalized auto-correlation of $a(t)$ for lag $\tau$ at sample $i$ as [44]:

$$R(i, \tau) = \frac{\sum_{k=0}^{k=\tau-1} \left[ \begin{array}{c} (a(i+k) - \mu(i, \tau)) \\ (a(i+k+\tau) - \mu(i+\tau, \tau)) \end{array} \right]}{\tau \delta(i, \tau) \delta(i+\tau, \tau)} \tag{1}$$

where $\mu(k, \tau)$ and $\delta(k, \tau)$ represent the mean and standard deviation of the samples. The auto-correlation $R(i, \tau)$ measures the similarity between acceleration signals as a function of the time lag $\tau$. $R(i, \tau)$ is in the range $[-1, 1]$ and should be close to 1 when

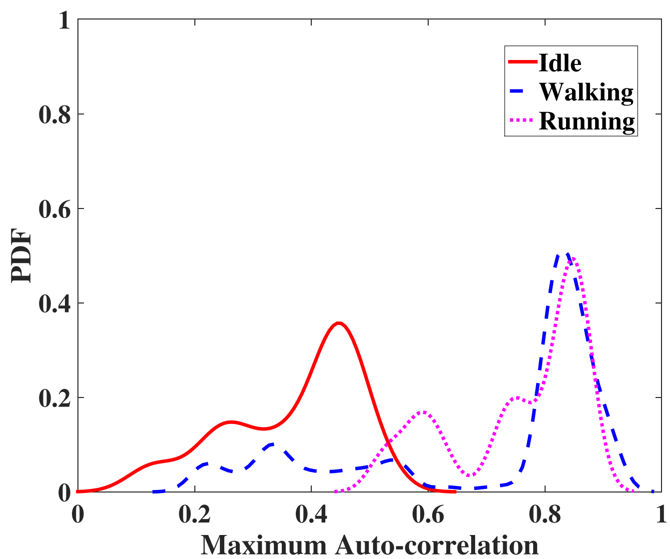



**Fig. 4.** Distribution of maximum auto-correlation of different activities.

**Table 1**
Normal activities during walking.

| With arm swing | Without arm swing |
|---|---|
| normal walk | walk with texting the phone |
| walk upstairs | walk with calling the phone |
| walk downstairs | walk with hand in jacket pocket |
| | walk with hand in pant pocket |

**Table 2**
List of features used for activity classification.

| Feature | Abbreviation | Description |
|---|---|---|
| mean | mean | The average of a window of samples |
| maximum | max | The maximum value in a window of samples |
| minimum | min | The minimum value in a window of samples |
| variance | var | Measures the amount of variation or dispersion from the mean |
| standard deviation | std | The square root of the variance |
| ratio | ratio | The ratio of three accelerometer axes X/Y, X/Z, Y/Z |

a subject is walking and $\tau$ is equal to the cycle of walking pattern. As the period of the subject's walking is unknown, we vary $\tau$ between $\tau_{\min}$ and $\tau_{\max}$ to find the $\tau_{opti}$ which makes the auto-correlation maximum (i.e., $\tau_{opti}$ is the period of walking):

$$R_{max}(i) = max_{\tau=\tau_{min}}^{\tau=\tau_{max}} R(i, \tau) \quad (2)$$

As mentioned in our previous conference paper [11], normal gait duration lies in 0.8-1.3s which produces 40–65 samples (the sampling rate of accelerometer is 50Hz). Therefore, the initial value of $(\tau_{\min}, \tau_{\max})$ is set to (40,65). Once the period of the subject's walking pattern is found in the first few steps (i.e., $\tau_{opti}$), the search window is reduced to $(\tau_{opti} - 15, \tau_{opti} + 15)$.

Fig. 4 depicts the distribution of $R_{\max}(m)$ for *IDLE, WALKING* and *RUNNING*, we can see that the probability that the person is *IDLE* is extremely low when $\psi(t) \geq 0.7$. However, we noticed that the maximum auto-correlation values is similar between *WALKING* and *RUNNING* as both of them are repetitive activities. We notice that the major difference between *WALKING* and *RUNNING* is the frequency. In order to distinguish different repetitive activities, we apply Short Term Fourier Transform (STFT) on the acceleration samples and calculate the spectral energy $F_{walk}$ (i.e., the magnitude of the STFT coefficients) at typical walking frequencies $(f_{\min}, f_{\max})$, we label the subject as walking if $F_{walk}$ is greater than a predefined threshold $F_{th}$ (400 in our system). Note that this method can also be used to detect other repetitive activities such as running by simply changing frequencies $(f_{\min}, f_{\max})$ to typical running frequency ranges; however, in this paper we consider walking only. The threshold in this Algorithm are set empirically. We evaluate the performance of the system by changing the threshold from the minimum value to the maximum value, then choose the one that achieves the highest recognition accuracy.

#### 4.1.2. Activity classifier

After walking detection, we use an activity classifier to detect how the user is walking. In this work, we focus on 7 of the most common activities while people are walking and divide them into two categories as shown in Table 1. The first category is walking with arm swing which include normal walk, walk upstairs and walk downstairs. The second category is walking without arm swing which include walk with texting the phone, calling the phone, and hand in jacket/pant pocket.

The real time data from an accelerometer contains much noise that needs to be filtered out before using it for activity recognition. The moving average filter is a simple low pass filter commonly used for regulating noisy signal. As the length of the filter increases, the smoothness of the output increases, whereas the sharp modulations in the data are made increasingly blunt. In our experiment we find that an order of 3 can produce good results. Thus a moving average filter of order 3 is applied for noise removal. After noise reduction, continuous sensor data is segmented into 2s sliding windows with 50% overlap. The window size of 2s is chosen to balance between classification accuracy and latency as discussed in Section 5.2. The overlap in sliding window is used to capture changes or transitions around the window limits. For each window, we extract a number of features as listed in Table 2. Accelerometer-based activity recognition has been well studied in the literature [40,42,45]. Previous studies used different kinds of features from both time and frequency domain representations of signals for activity recognition [40]. For computation efficiency, we choose some commonly used light-weight time domain features in the system. Therefore, each window is represented as a feature vector of length 18. These features are then used to train the classifier. As the evaluation in Section 5.2, we choose k-NN classifier as it achieves higher recognition accuracy than SVM and Decision Tree.

### 4.2. Feature extraction

The feature extraction approach presented in this paper is motivated by SIFT which is a commonly used feature extraction method for image matching and object recognition [10]. However, SIFT only works for 2D image and does not apply to 1 dimension signal. In this paper, we present a novel feature extraction method for one-dimensional gait signal.

The input of the proposed feature extraction approach is the segmented gait signal. The first stage is to identify locations and scales that can be repeatably assigned under different step cycles of the same subject. It has been shown that under a variety of reasonable assumptions the only scale-space kernel is the Gaussian function [10]. Therefore, the scale space of a gait signal is defined as a function $L(t, \delta)$, that is produced from the convolution of a variable-scale Gaussian $G(t, \delta)$ with an input gait signal $a(t)$:

$$L(t, \delta) = G_\delta(t, \delta) * a(t) \quad (3)$$

where $*$ is the convolution operation and $G_\delta(t, \delta)$ is the zero-mean Gaussian function with variance $\delta^2$:

$$G_\delta(t,\delta) = \frac{1}{\sqrt{2\pi}} exp\left(\frac{-t^2}{2\delta^2}\right) \tag{4}$$

To detect stable keypoint locations in scale space, we compute $D(t, \delta)$ which is the difference of two nearby scales separated by a constant multiplicative factor $\nu$:

$$\begin{aligned} D^\nu(t,\delta) &= (G(t,\nu\delta) - G(t,\delta)) * a(t) \\ &= L(t,\nu\delta) - L(t,\delta \end{aligned} \tag{5}$$

Then the DoG responses of gait signal $a(t)$ is represented by:

$$E(t,\delta) = (a * D_\delta^\nu)(t) \tag{6}$$

Now we aim to localize the keypoints by finding the extrema of $E(t, \delta)$. Instead of searching $E(t, \delta)$ continuously, we find extrema in a $\kappa$-layer pyramid by defining a discrete series:

$$E[t,i] = E(t, \nu^{i-1}\delta_0) \quad \text{for} \quad i = 1, 2, \cdots, \kappa \tag{7}$$

where $t$ is the sampling timestamp, $i$ is the layer index and $\delta_0$ is the base scale. To detect the local maxima and minima, each point in $E(t, i)$ is compared to its eight neighbors. It is selected only when it is larger than all of these neighbors or smaller than all of them. Any extremum in $E(t, i)$ is viewed as a keypoint and represented by a descriptor $\Psi(t, \delta)$. To accurately localize stable extrema, those extrema with small $\|E[t, i]\|$ which means low contrast will be rejected. Finally, the descriptor for each keypoint is obtained by calculating the gradient of a vector which contains $\rho$ points uniformly sampled around $t$:

$$\Psi(t,\delta) = \nabla\left(\frac{(\upsilon_1, \upsilon_2, \cdots, \upsilon_\rho)}{\|\upsilon_1, \upsilon_2, \cdots, \upsilon_\rho\|_2}\right)$$

$$where \quad \upsilon_i = (a * G_\delta)\left(t + i - \frac{\rho+1}{2}\right) \tag{8}$$

The descriptors are invariant to amplitude changes caused by varying walking speed. A change to the amplitude means each signal value will be multiplied by the same constant, so this change can be canceled by vector normalization. A speed change will not affect the gradient values, as they are computed from differences. Fig. 5 illustrates two gait signal series from two different subjects and their corresponding descriptors. We can see that the descriptors extracted from the same signal have similar patterns; however, the descriptors extracted from different signals are distinctive. In Gaitwatch, we empirically choose the length of the feature vector to be 18 (i.e., $q = 18$). The length of the feature vector is a trade-off between recognition accuracy and resource consumption. A longer feature vector will usually improve recognition accuracy but require more processing time at the same time. We find that after the length is larger than 18, the accuracy improvement diminishes.

### 4.3. Classification

#### 4.3.1. Dictionary construction

After feature extraction, the feature descriptors obtained from training data are used to construct the training dictionary. Recent research shows that learning a dictionary by fitting a set of overcomplete basis vectors to a collection of training samples can generate more compact and informative representation from given data and achieve better recognition accuracy [46]. We construct the training dictionary by sparse dictionary learning technique (also called sparse coding). In particular, we first learn one single dictionary for each subject, which is formed by a set of basis vectors learned by solving a sparse optimization problem. Then we construct the full dictionary by concatenating single dictionaries together.

Note that because descriptors obtained from training data and test data are vectors, we will also refer to them as training vectors and test vectors. Suppose we have $K$ classes indexed by $i = 1, \ldots, K$ and each class $i$ contains $N$ training examples which are denoted as $S_i = \{s_1, s_2, s_3, \ldots, s_N\}$, where $S_i$ contains the extracted feature vectors from subject $i$. Each training example is assumed to be a column vector with $q$ elements (i.e., feature dimension). For class $k$, we aim to find an overcomplete dictionary matrix $A_k \in \mathbb{R}^{q\times N}$ over which a test vector has a sparse representation $X_k = \{x_1, x_2, \ldots, x_{N_i}\}$. After that, the raw training examples $S_i$ can be linearly expressed by $n_k$ vectors in $A_k$ where $n_k \ll N$. The optimization problem of training a dictionary can be formulated as:

$$\arg\min_{A_k, X_k} \|S_k - A_k X_k\|_2^2 \quad \text{subject to } \|x_i\|_0 \le n_k \tag{9}$$

There are several dictionary learning algorithms that can be used to train a dictionary such as Method of Optimal Directions (MOD) [47], K-Singular Value Decomposition (K-SVD) [46] and Nonnegative Matrix Factorization (NMF) [48]. In this study, we

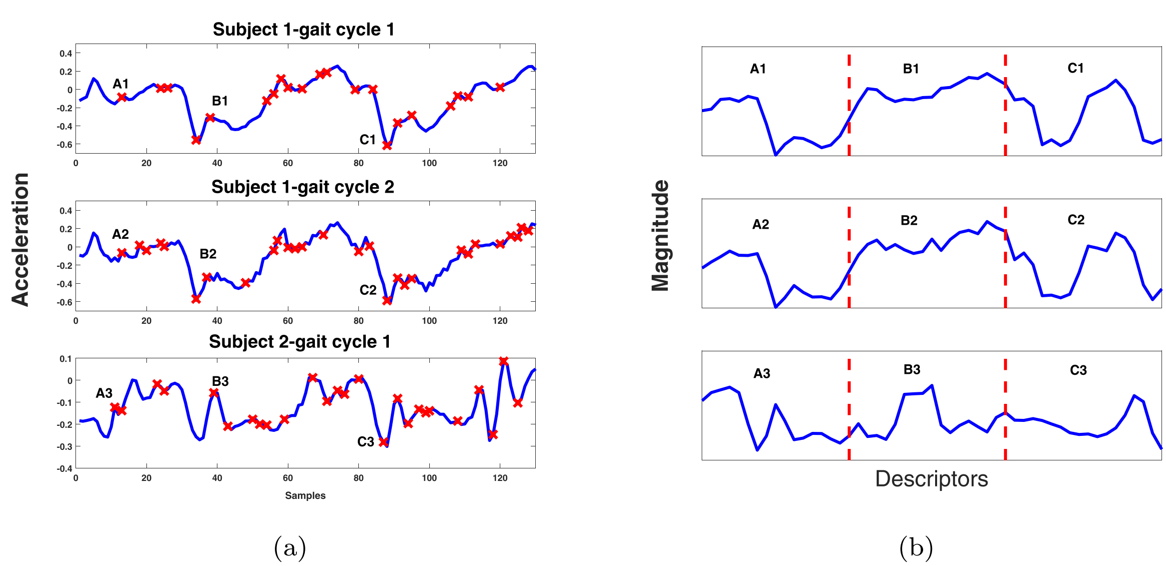



**Fig. 5.** Feature extraction: (a) The red crosses represent the locations of key points. (b) Extracted descriptors for the three keypoints in the left figure. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

choose K-SVD because it is efficient, flexible and works in conjunction with any pursuit algorithms. The K-SVD algorithm involves two stages: firstly, $A_k$ is kept fixed and the coefficient matrix $X_k$ is optimized by orthogonal matching pursuit (OMP) algorithm. Then the dictionary $A_k$ is updated using the calculated $X_k$. The process repeats until some stopping criterion (typically a fixed number of iterations) is achieved. The dictionary learning algorithm is detailed in Algorithm 2. After constructing a dictionary for each subject, we concatenate single dictionaries together to form the initial training dictionary $A = [A_1, A_2, \cdots, A_K]$.

**Algorithm 2:** Subject-Specific Dictionary Learning.

```
Input: Training samples S = {s_1, s_2, s_3, ..., s_N}, initial
dictionary A_0 ∈ R^{q×N}, target sparsity τ;
Output: Dictionary A and sparse coefficients matrix X;
Initialization: set dictionary A = A_0;
while != stopping criteria do
    x_i = argmin_x ||s_i − x||_2^2   s.t. ∀i   ||Ax||_0 ≤ τ;
    for j = 1, ..., N do
        J = {indices of the columns of X orthogonal to w_j (j-th
        column of A)};
        w_j = argmin_w ||w^T A_J||_2^2   s.t. ||w||_2 = 1;
        A(j-th row) = w_j^T;
    end
end
```

*4.3.2. Column reduction*

According to the formation of $\ell_1-$Homotopy, the computational complexity is $O(\tau^3 + \tau q(N \cdot K))$, where $\tau$ is the sparsity of the solution ($\tau \ll N \cdot K$), $q$ is the number of equations, and $N \cdot K$ is the number of unknowns, i.e., the number of columns in the training dictionary (please refer to [49] for more details). We can see that the computation of $\ell_1$ optimization is also proportional to the number of columns ($N \cdot K$) in the dictionary $A$. The gait cycles in the same class are highly correlated and lead to intra class redundancy. To reduce the intra class redundancy in the dictionary while retaining the most informative columns, we apply the columns reduction approach [39] to improve the efficiency and obtain an optimised dictionary $\bar{A}$.

*4.3.3. PSRC*

SRC proposed in [50] aims to solve the classification problem of one test vector. To overcome this limitation, we present a novel probabilistic fusion model which fuses the information from multiple consecutive gait cycles to further improve recognition accuracy.

Suppose we have obtained a series of gait signal whose length is $Ts$, and we have acquired a set of $M$ descriptors $Y = \{y_1, y_2, \ldots, y_M\}$ from this signal. Following the single test vector approach described in [50], we can obtain a set of estimated coefficients vectors $\hat{X} = \{\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_M\}$ by solving the $\ell_1$ optimization problem for each feature descriptor. Then we calculate the residual for each feature descriptor as [50] and obtain $\Re = \{r_1, r_2, \ldots, r_M\}$. The probability of the $m$-th test descriptor belonging to the $i$-th class is defined $p(\phi = i|y_m)$ where $\phi$ is used to denote the identity of $y_m$. Taking the elements of $Y$ as independent observations, the probability of all $M$ descriptors belonging to $i$-th class can be denoted by $p(\phi = i|Y)$.

As discussed in [50], the magnitude of $r_i$ represents the similarity between the test sample and $i$-th subject. With this knowledge, we use the $\ell_1$-norm of the residual $r_i$ to define the posterior probability of $m = i$ given $y_m$ as follows:

$$p(\phi = i|y_m) = \frac{exp(-\lambda\|r_i\|_1)}{\sum_{j=1}^{M} exp(-\lambda\|r_j\|_1)} \in [0, 1] \tag{10}$$

where $\lambda$ is a constant parameter. We vary the value of $\lambda$ from 0 to 1 with an increment of 0.1 and find that it can achieve the highest accuracy when $\lambda$=0.3. Therefore, we choose $\lambda$=0.3 in Gait-watch.

For the $i$-th subject, we define $\theta_i$ as

$$\theta_i = \sum_{y \in Y} ln\ p(\phi = i|y) \tag{11}$$

Since we have no prior knowledge of $y$, it should normally follow a uniform distribution over $1, 2, \ldots, M$, say $p(\phi = i) = 1/M$. We can obtain the probability of all $M$ gait cycles belonging to $i$-th class $p(\phi = i|Y)$ as follows:

$$p(\phi = i|Y) = \frac{exp(\phi_i)}{\sum_{j=1}^{M} exp(\phi_j)} \in [0, 1] \tag{12}$$

With the knowledge of $p(\phi = i|Y)$, the final classification result is obtained by finding the maximum posterior probability:

$$Identity = \max_i p(\phi = i|Y) \tag{13}$$

To identify whether the walker is the genuine user or imposter, we can make decision based on a threshold as:

$$p \begin{cases} \geq C & \text{genuine user} \\ < C & \text{imposter} \end{cases} \tag{14}$$

where $C$ is a threshold we set empirically. An appropriate threshold can be chosen by data-driven approach to make the recognition system robust to imposters.

## 5. Evaluation

*5.1. Goals, metrics and methodology*

In this section, we evaluate the performance of Gait-watchvia simulation. The goals of the simulation are threefold: 1) to evaluate the performance of activity classifier; 2) whether Gait-watchimproves recognition accuracy by leveraging the activity information; 3) whether Gait-watchachieves high accuracy in differentiating imposters and genuine users.

In this paper, we focus on the following metrics:

- **Recognition accuracy**: it represents the percentage of correct classifications which can be calculated as the percentage of the total number of tests that resulted in correct classifications.
- **False match rate (FMR)**: it is the measure of probability that the authentication system incorrectly accepts the access request by an imposter.
- **False rejection rate (FRR)**: it is the measure of the probability that the authentication system incorrectly rejects the access requests from the genuine users.

In general, FMR relates to the security of the system, while FRR to the usability. An interesting point in the Decision Error Tradeoff (DET) curve is the Equal Error Rate (EER) where FMR=FRR. For instance, an EER of 5% means that out of 100 genuine trials 5 is incorrectly rejected, and out of 100 impostor trials 5 are wrongfully accepted.

*Data Collection* To evaluate the performance of Gait-watch, we collected data (accelerometer data) with Samsung smart watch in both controlled environment and real-world environment[1].

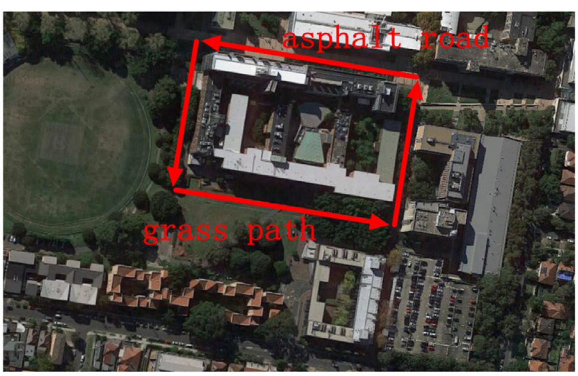
We developed a sensor recording application on smart watch. The application collects the sensor data continuously after the participant launches it. During data collection phase, the participants wore the smart watch at one of their wrists at will, and were asked to perform the activities listed in Table 1 with equal proportion

[1] Ethical approval for carrying out this experiment has been granted by the corresponding organization.

**Table 3**
Summary of gait dataset.

| Property | Value |
|---|---|
| Subjects | 36 (20 males,16 females) |
| Age | 22–43 |
| Height | 157–181 cm |
| Weight | 48–86 kg |
| Recording session | 2 |
| Time interval between two sessions | 1–2 months |
| Length of each record | 5–10 minutes |
| Sample frequency | 100 Hz |

(a) Indoor experiment




(b) Outdoor experiment

**Fig. 6.** Illustration of data collection.

to avoid class imbalance. For each activity, the participants were requested to walk 3 minutes at their normal speed. Each activity contains approximately $1.3 \times 10^6$ samples. The data collection is performed in several environments (indoor and outdoor) in order to capture different terrains. There were 36 volunteers (20 males and 16 females) participating in data acquisition. More details of the gait dataset are summarized in Table 3. An illustration of indoor environment and outdoor environment is shown in Fig 6(a) and Fig 6(b). The terrain of the chosen outdoor environment varies including plain, grass and asphalt. Each volunteer participated in two data collection sessions that was separated by 1–2 months. We will refer to these sessions as *session 1* and *session 2* respectively. A participant might dress in different ways (e.g. different clothes and shoes) for the two sessions. Based on the above description, the dataset is close to a realistic environment as it includes the natural gait changes over time and different environments (indoor and outdoor). In total, we obtained about 17 hours of data to evaluate the performance of Gait-watch.

As our goal is to recognize different subjects when the user performs different activities. We organize the dataset in two different forms for different purposes: activity dataset and gait dataset. The activity dataset is organized based on the activity labels and used to evaluate the performance of activity classifier. The activity dataset is organised based on the subject and used to evaluate the accuracy of gait recognition. In the following evaluations, we perform 10-fold cross-validation on the collected dataset and plot the results of the average values. Specifically, we randomly split the dataset into 10 folds with equal size. Then, each fold is retained as the validation data for testing the classifier, and the remaining 9 folds are used as training data. The cross-validation process is then repeated 10 times, with each of the 10 folds used exactly once as the testing data.

## 5.2. Performance of activity classifier

In this section, we evaluate the recognition accuracy of different classifiers based on activity dataset. Then, we demonstrate that Gait-watchimproves recognition accuracy by leveraging the activity information.

### 5.2.1. Comparison of different activity classifiers

We evaluate the recognition accuracy of three mostly used classifier in online activity recognition: k-NN, SVM and Decision Tree [42]. We first evaluate the accuracy of k-NN by varying $k$ (the number of nearest neighbor) from 3 to 50. As shown in Fig 7(a), k-NN achieves best accuracy when $k = 3$. After $k$ is determined, we evaluate the accuracy of different methods by varying window size from 0.5s to 2.5s. As we can see from Fig 7(b), k-NN achieves the classification accuracy by up to 98.6% when window size is chosen as 2s, and shows higher accuracy than SVM and Decision Tree at all feasible window size. The window size determines how many features can be extracted. A longer window means more features can be extracted and thus can achieve high accuracy. Therefore, from Fig. 7(b) we can see the accuracy of 2s window is higher than that of 1s and 0.5s. However, a longer window will sacrifice user experience because it requires users to walk more steps. Based on this experiment, we choose k-NN as activity classifier, the window size and $k$ value is chosen as 2s and 3 respectively. It should be noted that k-NN is used to detect the activity of the user rather than verifying the identity of the user.

### 5.2.2. Improvement of recognition accuracy by using activity classifier

Traditional gait-based recognition system assumes the user walks normally. In order to deal with the problem of various activities, a common method is to train a large dictionary by assembling gait features from all activities. We term this method as *conventional method*, and compare it with context-aware Gait-watch. For *conventional method*, we use all gait features of each subject to form the training dictionary. Regarding Gait-watch, we first perform activity detector to check the user's activity. Based on the result of activity classifier, user recognition is performed based on the appropriate training data. For example, if the test gait signal is labeled with walking upstairs, we use gait features from walking upstairs to form the training dictionary only. We evaluate the recognition accuracy of Gait-watchby extracting features from different length of gait signals. The more gait data we collect, the higher accuracy will achieve as we can extract more features. As shown in Fig 7(c), we can observe a significant accuracy improvement of Gait-watchover *conventional method*, the improvement can be up to 30.2% when $T = 2.5s$. The improvement diminishes after $T > 2.5s$. The results suggest that it is critical to know the user's activity for authentication, and the proposed context-aware authentication system improves recognition accuracy significantly by leveraging the activity information. Additionally, for comparison pur-

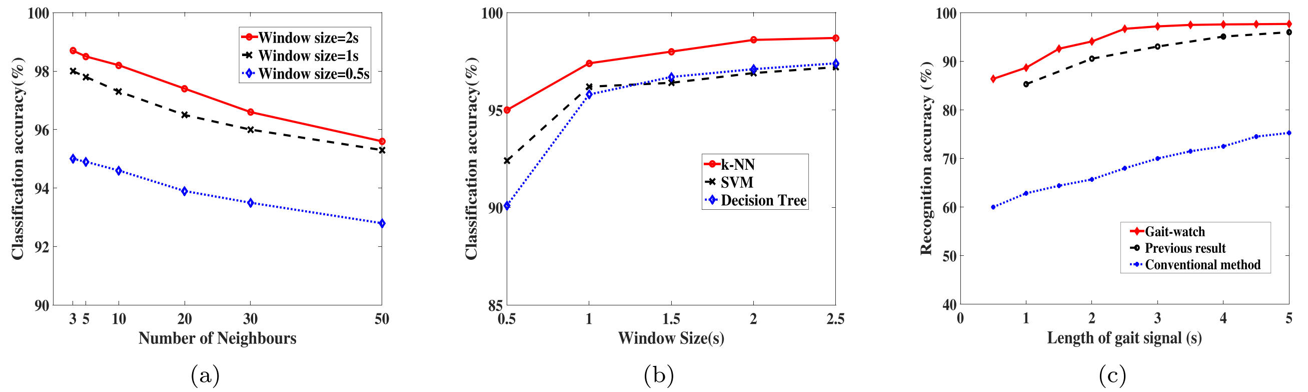



**Fig. 7.** Evaluation results of activity classification: (a) accuracy of different number of neighbors. (b) comparison of different methods. (c) improvement of recognition accuracy by using activity classifier.

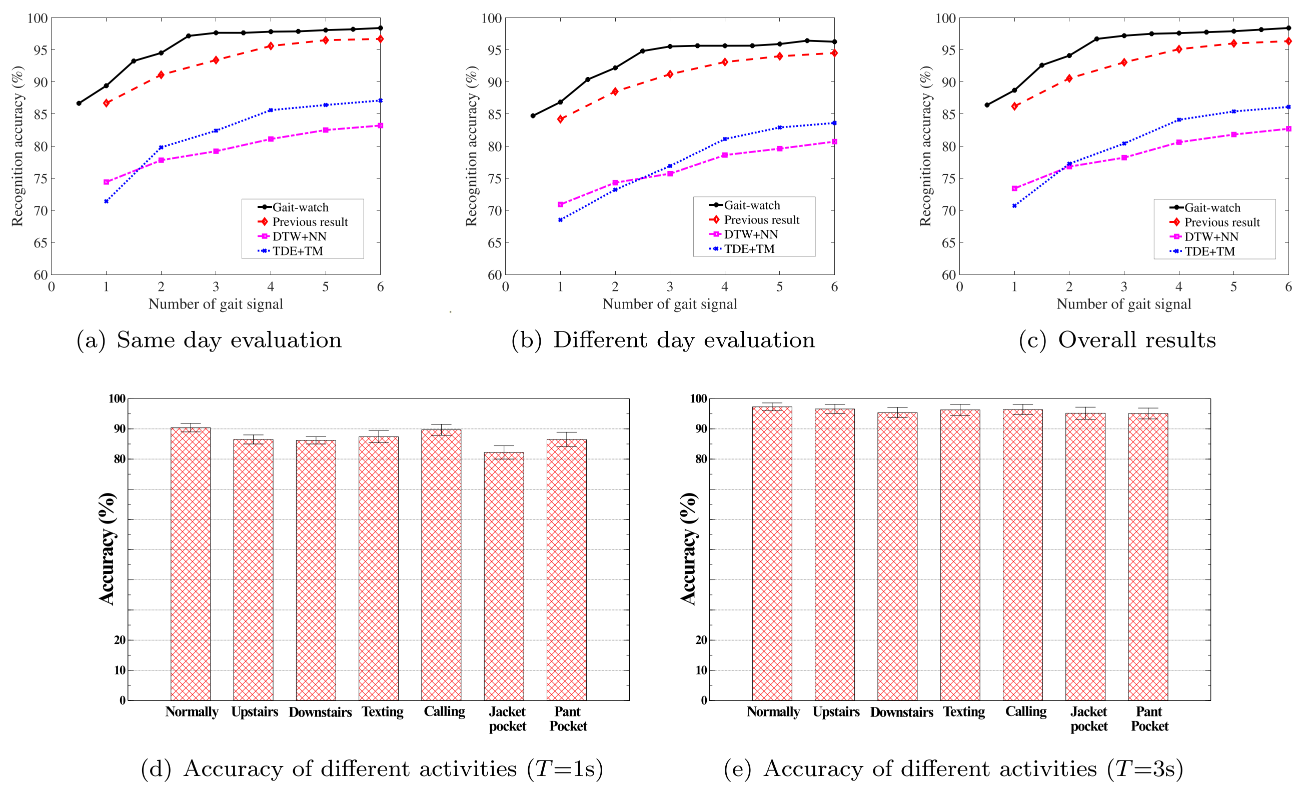



**Fig. 8.** Evaluation of activity classification.

pose, Fig 7(c) also shows the results of our previous conference paper [11], we can see that the improvement varies from 2% to 6.5%. The improvement originates from two aspects. First, the dictionary in this paper is trained by dictionary learning rather than simply assembling different features together. Learning a dictionary can generate more compact and informative representation from given data and achieve better recognition accuracy. Second, the feature extraction method used in this paper can extract robust and discriminative features from gait signals. To take a further look at the results, we plot the mean accuracy of each subject as well as 95% confidence level in Fig. 9. We can see that the mean accuracy of different subjects are similar. The error bar can be used to indicate the worst case of each subject. The results demonstrate that our system can recognize different subjects with high accuracy.

*5.2.3. Comparison with other gait recognition methods*

We compare Gait-watchwith several state-of-the-art gait recognition methods, namely, Dynamic Time Warping with Nearest Neighborhood (DTW+NN) [51] and Time-Delay Embeddings with Template Matching (TDE+TM) [52]. We perform evaluation in two categories: same day evaluation and different days evaluation. Same day evaluation means the training set and test set are chosen from the sessions of the same day while different days evaluation chooses the sessions of different days. We vary the number of gait cycles $M$ from 1 to 6 and calculate the recognition accuracy of different methods. As our method is not based on extracting gait cycles, we can extract features from gait signals with any length. We calculate the accuracy of our method with 0.5 gait cycle, 1 gait cycle,..., 6 gait cycles.

Fig 8 (a) presents the evaluation results. From the same day evaluation, we can see that Gait-watchachieves the best recognition accuracy. Fig 8(b) plots the recognition accuracy from different days evaluation. The accuracy of all the gait recognition methods is slightly lower than the same day evaluation as the different days evaluation tends to produce more dynamics between the training

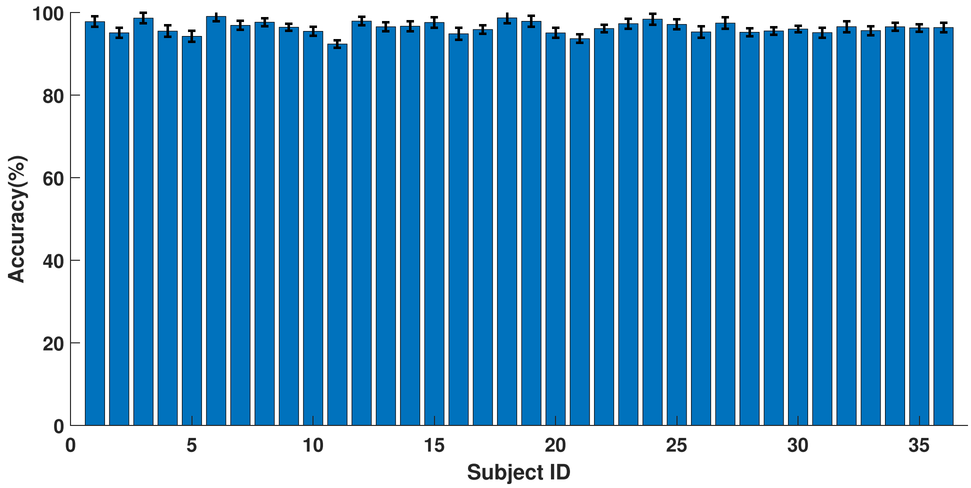
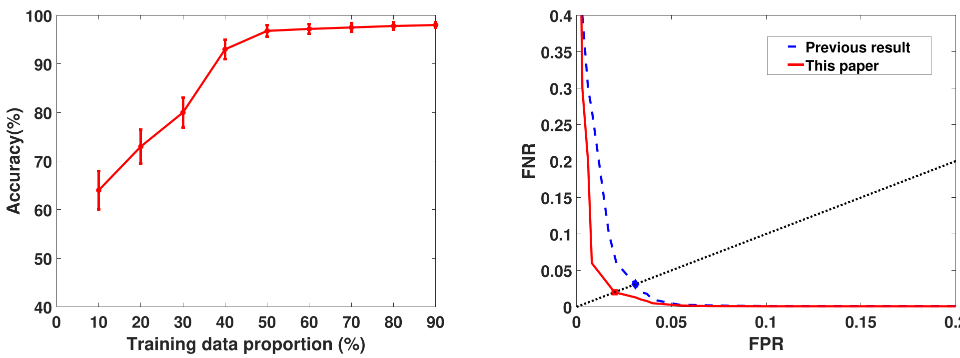



**Fig. 9.** Accuracy of each subject.




(a) Impact of training data size.
(b) Robustness to attackers.

**Fig. 10.** Evaluation results.

data and test data. The observations from the same day evaluation still hold true in the different days evaluation. The overall recognition accuracy of different methods are shown in Fig 8(c). We can see that Gait-watchis 17% better than TDE+TM and 20% better than DTW+NN when $M = 3$. We also notice that the recognition accuracy of Gait-watchincreases when more gait cycles are fused for recognition, this is intuitive as more information can be obtained by fusing multiple gait cycles together. The overall recognition accuracy of Gait-watchis over 95% when 3 or more consecutive gait cycles are used. However, for our previous conference version, it requires at least 5 gait cycles to achieve over 95% accuracy. We define the number of gait cycles as a user or application defined parameter; more accurate recognition will be achieved when the user would like to make more efforts. The setting of more gait cycles indicates more time it will take before collecting sufficient gait cycles to authenticate the walker.

#### 5.2.4. Impact of different activities

We now evaluate the recognition accuracy under different activities to explore the impact of activity on Gait-watch. For each activity, we use the features extracted from the activity to perform 10-fold cross-validation. From the results in Fig 8(d), we can see that walking normally shows better performance than other activities if we use only one gait cycle for recognition. Walking with hand in jacket/pant pocket performs the worst because the subjects wore different clothes at different days during data collection phase. The results indicate that, if we do have control over the users, walking normally is a good activity to use for identification purposes. However, when we fuse several gait cycles to perform recognition, different activities show comparable recognition accuracy as shown in Fig 8(f). This result demonstrates the advantage of the proposed sparse fusion method. Also, it indicates that Gait-watchcan recognize the user under various activities with a high accuracy.

#### 5.2.5. Impact of different training dataset size

Next, we evaluate the accuracy of the proposed system under different sizes of training dataset. In this experiment, we use different proportions of the whole dataset for training, and use the left dataset for testing. The proportion increases from 10% to 90% with an increment of 10%. For example, the proportion of 10% means we use 10% of the dataset for training, and use the left dataset for testing. From the results in Fig. 10(a), we can see that the accuracy of our methods becomes relatively stable after 50% of the dataset is used for training. The accuracy drops quickly when we use less data for training. This is because the dictionary-based method requires a lot of data to learn the representation of the raw signal.

#### 5.2.6. Robustness against attackers

As mentioned previously, an attacker can imposter the genuine user to gain access to the smartwatch. To evaluate the robustness of Gait-watchagainst the imposter attack scenario, we group the 36 subjects into 18 pairs. Each subject was told to mimic his/her partner's walking style and try to imitate him or her. Firstly, one participant of the pair acted as an attacker, the other one as a target, and then the roles were exchanged. The genders of the attacker and the target were the same. They observed the walking style of the target visually, which can be easily done in a real-life situation as gait cannot be hidden. Every attacker made 5 active impostor attempts. As an authentication system, Gait-watchis sensitive to false acceptance rate. Therefore, we set $T = 5s$ to reduce false positive rate. We vary the confidence threshold $C$ and plot DET curve in Fig 10(b). The black dash line stands for the possible points where FMR is equal to FRR. The crossover (marked as a square) of the black dash line and the red FMR-FRR curve stands for the location of the EER. We notice that EER of Gait-watchis as low as 3.5%, which means out of 100 impostor trials only 3.5 are wrongfully accepted. Additionally, we plot the results of our conference paper for comparison purpose. We can see that this paper further reduces EER from 5.6% to 3.5%. The setting of more steps indicates more time it will take before collecting sufficient step cycles to detect the suspect. Therefore, we define the number of steps as a user-defined parameter: more accurate authentication will be achieved if the users would like to pay more efforts on the walking.

## 6. User study in the wild

Our final study aims to evaluate the performance of Gait-watchin real world scenarios.

### 6.1. System implementation

The prototype of Gait-watchis implemented on Samsung Gear Live Smart Watch. The CPU is a Qualcomm Snapdragon at 1.2GHz and the operating system is Android wear. The efficient implement of $\ell_1$ optimization algorithm $\ell_1$-*Homotopy* [49] is used. To reduce the expected response time, we implement Gait-watchin multiple threads. 6 threads are allocated for Gait-watchaccording to the system setting. One of the threads is responsible for the step detection. The rest of the threads are in idle before the sensor data of gait cycles are received. For each time a new step is detected, the corresponding sensor data of the step cycle is passed to activate a new thread. After all the sparse coefficients vectors are obtained by dynamic sparse representation, the authentication decision is determined by comparing the confidence level with a pre-defined parameter $C$ as Eq 14. The time duration to collect gait signal is $T = 3s$ and the confidence level threshold is $C = 0.37$.

### 6.2. User study

We recruited 6 users (3 males and 3 females) for this study. The users wore the smart watch for 2 hours per day for three different days. We asked the users to launch Gait-watchafter they put the smart watch on and end it after they take it off. Gait-watchworks continuously after it is launched, and logs the authentication decision after the required length of gait signals are collected.

In practice, Gait-watchcan run in a power saving mode. It starts as a background service when the user puts the smart watch on, and performs authentication when it detects the wearer is walking. If the wearer is accepted as genuine user, Gait-watchstops working and the wearer will be regarded as the owner continuously without additional authentication before the take-it-off action is detected. This is due to the fact that the smart watch must be on the same subject's wrist before a take-it-off activity is detected. The put-it-on and take-it-off activities can be detected by the take-on sensor in smart watch.

### 6.3. Experimental results

Fig 11 shows the results of the user study. We notice that the total number of authentication attempts varied between 44 to 86 per day. Of all the 1016 detected authentications, Gait-watchauthenticates the genuine users correctly by 967 times, i.e., the True Positive Rate (TPR) achieves up to 95.2%. The results demonstrate the effectiveness of Gait-watchin authenticating users in real world scenarios.

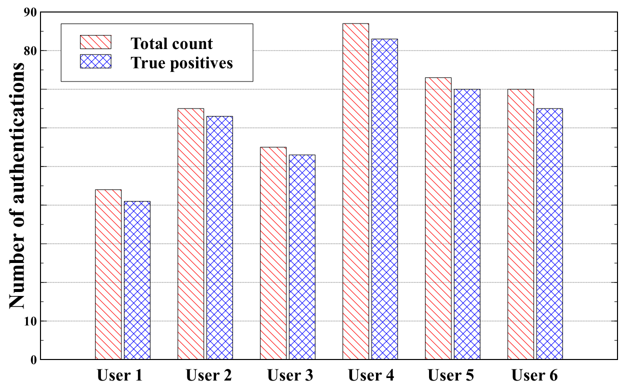
Table 4 shows the resource consumption of Gait-watchmeasured on Samsung Smart Watch. The computation time is obtained from the console of the Eclipse development environment and averaged by the results from 30 authentication attempts. The memory usage is measured by PowerTutor App (it was also used in [35]). The computation time of the five stages in the pipeline: walking detection, activity detection, segmentation and interpolation, classification take an average time of 120, 80, 4, 346 and 270 ms, respectively. When the whole pipeline is fully engaged, Gait-watchtakes about 2.5s to collect the required gait data and can respond in approximately 820 ms. Besides, it consumes 603.6 mJ only. The memory requirement to execute Gait-watchis modest and varies between 40–49 MB. Although the approaches in this work are more complex than our previous paper [11], we notice that the computation time and energy consumption are very close. This is due to the following two reasons. First, compared to [11] the system in this paper is based on feature extraction. Thus it does not require some signal processing steps in [11] such as gait cycle segmentation, interpolation and unusual cycle deletion. More importantly, as the accuracy of this system is higher than our previous system, the required signal length to achieve similar accuracy is reduced. For example, in [11], it requires 6 gait cycles (about 4.8-5.5s) to achieve 96% accuracy. However, the system proposed in this paper only requires less than 3s. Therefore, the total processing time is comparable. It is worth mentioning that we also measure the power consumption when the Samsung watch is idle which is 168mW.

The user study results show that Gait-watchcan recognize the user accurately in real world scenarios and require low system cost. Therefore, it is feasible to implement Gait-watchon off-the-shelf smart watches. However, we are aware that gait-based authentication system cannot provide a absolutely secure way to protect the data in smart watch. We imagine in the future to have many levels of security (e.g., two factor authentication) that tradeoff usability and accessibility given the risk imposed. For instance, accessing a user's bank account clearly requires higher security than retrieving the time of the next meeting and occurs less frequently, hence requiring additional authentication scheme (e.g., PIN) after Gait-watchgrants user the access to use the smart watch.

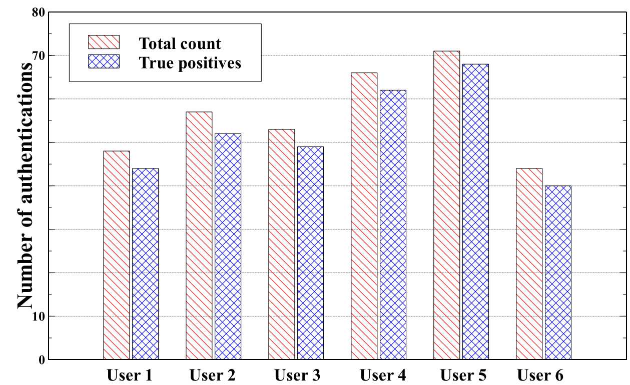
We also need to note that another common strategy is offloading computationally intensive operations from mobile devices to local or remote server to reduce computation burden on resource constrained devices. However, as mentioned in [38], the usability of the cloud-based recognition systems relies on the wireless connectivity. It is difficult to ensure the smartwatch can always be connected to the server. Therefore, the *in-situ* approaches are preferable considering the relatively high cost of wireless transmission and the inconvenience of relying on wireless connections. In the user study, the app is running in continuous authentication mode which drains the battery of smart watch quickly. This is the reason why our user study only lasts for 2 hours per day. However, as mentioned above, Gait-watchonly needs to perform one authentication per day because it is worn on the same user after the user puts it on. Based on the results in Table 4, the energy consumption of one authentication is negligible compared to the whole battery life of smart watch.

### 6.4. Validity analysis

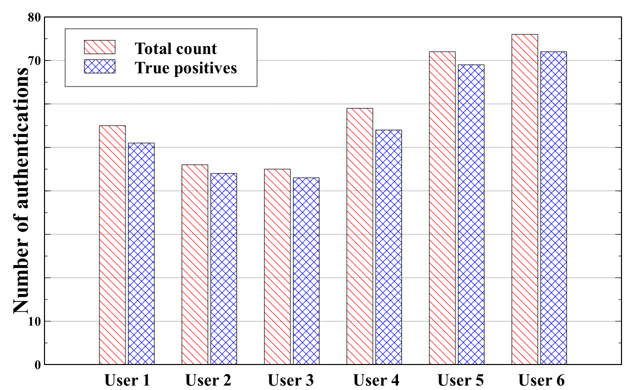
In this section, we discuss threats to the validity of our study. In the data collection and user study, the participants are randomly chosen and they are from different countries with different height,




(a) Day 1




(b) Day 2




(c) Day 3

**Fig. 11.** Evaluation results of user study: total count means the total number of authentication attempts user performed and true positive means the number of accepted authentications.

**Table 4**
System overhead.

| | Computation time (ms) | Energy consumption (mJ) |
|---|---|---|
| Walking detection | 120 | 121 |
| Activity classification | 80 | 20.6 |
| Signal processing | 4 | 6 |
| Feature extraction | 346 | 289 |
| Classification | 270 | 167 |
| Total | 820 ms | 603.6 mJ |
| Memory usage | 40–49 MB | |

weight and genders. Moreover, we repeat the data collection and user study in different days during which they may wear different clothes and shoes. Therefore, this study has high internal validity. However, the external validity is relatively low because it lacks some generalizability. First, the participants are students or faculty members working at our university. Second, although the data is collected from both indoor and outdoor environments, the indoor environment only includes one office and the outdoor environment only includes one walking path. It is unclear how the system performs for other groups of people such as labour workers, and in other environments which have different terrains. This is the limitation of our current work, and we will mitigate this threat in our future work by collecting more data from different groups of people and different environments.

## 7. Conclusion

Gait recognition on mobile devices is a hot research topic in recent years. In this work, we address the problem of recognizing the user under different activities. In the proposed Gait-watch, we first present a feature extraction technique that can extract discriminative features from user's gait signal. Then we employ sparse coding to build different training dictionaries for different activities for the genuine user. Extensive evaluation results show that Gait-watchimproves recognition accuracy greatly by leveraging the activity information and can recognize the user accurately in real world scenarios. We also perform a user study to demonstrate the feasibility of Gait-watchfor commercial smart watches. Evaluation results show Gait-watchcan authenticate the user in real world applications with high accuracy and require low system cost. To sum up, Gait-watchprovides an unobtrusive, continuous authentication way for smart watch users.

## Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgements

The work described in this paper was fully supported by a grant from City University of Hong Kong (Project No. 7200642).

## Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.adhoc.2020.102218.

## References


[1] M. Rabbi, S. Ali, T. Choudhury, E. Berke, Passive and in-situ assessment of mental and physical well-being using mobile sensors, in: Ubicomp' 2011, ACM, 2011, pp. 385–394.
[2] A.J. Aviv, K. Gibson, E. Mossop, M. Blaze, J.M. Smith, Smudge attacks on smartphone touch screens., WOOT 10 (2010) 1–7.
[3] F. Schaub, R. Deyhle, M. Weber, Password entry usability and shoulder surfing susceptibility on different smartphone platforms, in: Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia, ACM, 2012, p. 13.
[4] S. Uellenbeck, M. Dürmuth, C. Wolf, T. Holz, Quantifying the security of graphical passwords: The case of android unlock patterns, in: Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security, ACM, 2013, pp. 161–172.
[5] C. Nickel, T. Wirtl, C. Busch, Authentication of smartphone users based on the way they walk using k-nn algorithm, in: IIH-MSP' 2012, IEEE, 2012, pp. 16–20.
[6] Y. Ren, Y. Chen, M.C. Chuah, J. Yang, Smartphone based user verification leveraging gait recognition for mobile healthcare systems, in: Secon' 2013, IEEE, 2013, pp. 149–157.
[7] M. Alizadeh, S. Abolfazli, M. Zamani, S. Baharun, K. Sakurai, Authentication in mobile cloud computing: a survey, J. Netw. Comput. Appl. 61 (2016) 59–80.
[8] O.M. Parkhi, A. Vedaldi, A. Zisserman, et al., Deep face recognition., in: bmvc, 1, 2015, p. 6.
[9] T. Wolf, M. Babaee, G. Rigoll, Multi-view gait recognition using 3D convolutional neural networks, in: 2016 IEEE International Conference on Image Processing (ICIP), IEEE, 2016, pp. 4165–4169.
[10] D.G. Lowe, Object recognition from local scale-invariant features, in: The proceedings of the seventh IEEE international conference on Computer vision (ICCV), 2, Ieee, 1999, pp. 1150–1157.
[11] W. Xu, Y. Shen, Y. Zhang, N. Bergmann, W. Hu, Gait-watch: a context-aware authentication system for smart watch based on gait recognition, in: IOTDI, ACM, 2017, pp. 59–70.
[12] J. Han, B. Bhanu, Individual recognition using gait energy image, Pattern Anal. Mach. Intell. IEEE Trans. on 28 (2) (2006) 316–322.
[13] S.A. Shaikh, J.R. Rabaiotti, Characteristic trade-offs in designing large-scale biometric-based identity management systems, J. Netw. Comput. Appl. 33 (3) (2010) 342–351.
[14] K. Tian, J. Li, X. Yang, A novel method of micro-doppler parameter extraction for human monitoring terahertz radar network, Ad Hoc Netw. 58 (2017) 222–230.
[15] Y. Zeng, P.H. Pathak, P. Mohapatra, Wiwho: wifi-based person identification in smart spaces, in: Proceedings of the 15th International Conference on Information Processing in Sensor Networks, IEEE Press, 2016, p. 4.
[16] R.J. Orr, G.D. Abowd, The smart floor: a mechanism for natural user identification and tracking, in: Proc. of CHI, ACM, 2000, pp. 275–276.
[17] K.A. Sidek, V. Mai, I. Khalil, Data mining in mobile ecg based biometric identification, J. Netw. Comput. Appl. 44 (2014) 83–91.
[18] L. Middleton, A. Buss, A. Bazin, M.S. Nixon, et al., A floor sensor system for gait recognition, in: Automatic Identification Advanced Technologies, 2005. Fourth IEEE Workshop on, IEEE, 2005, pp. 171–176.
[19] H.J. Ailisto, M. Lindholm, J. Mantyjarvi, E. Vildjiounaite, S.-M. Makela, Identifying people from gait pattern with accelerometers, in: Defense and Security, International Society for Optics and Photonics, 2005, pp. 7–14.
[20] D. Gafurov, K. Helkala, T. Søndrol, Biometric gait authentication using accelerometer sensor, J. Comput. (Taipei) 1 (7) (2006) 51–59.
[21] D. Gafurov, E. Snekkenes, T.E. Buvarp, Robustness of biometric gait authentication against impersonation attack, in: On the Move to Meaningful Internet Systems 2006: OTM 2006 Workshops, Springer, 2006, pp. 479–488.
[22] E. Vildjiounaite, S.-M. Mäkelä, M. Lindholm, R. Riihimäki, V. Kyllönen, J. Mäntyjärvi, H. Ailisto, Unobtrusive Multimodal Biometrics for Ensuring Privacy and Information Security with Personal Devices, in: Pervasive Computing, Springer, 2006, pp. 187–201.
[23] H. Lu, J. Huang, T. Saha, L. Nachman, Unobtrusive gait verification for mobile phones, in: ISWC' 2014, ACM, 2014, pp. 91–98.
[24] A. Primo, V.V. Phoha, R. Kumar, A. Serwadda, Context-aware active authentication using smartphone accelerometer measurements, in: CVPRW' 2014, IEEE, 2014, pp. 98–105.
[25] W. Xu, G. Lan, Q. Lin, S. Khalifa, N. Bergmann, M. Hassan, W. Hu, Keh-gait: towards a mobile healthcare user authentication system by kinetic energy harvesting, in: NDSS' 2017.
[26] W. Xu, G. Revadigar, C. Luo, N. Bergmann, W. Hu, Walkie-talkie: motion-assisted automatic key generation for secure on-body device communication, in: IPSN' 2016, IEEE, 2016, pp. 1–12.
[27] W. Xu, C. Javali, G. Revadigar, C. Luo, N. Bergmann, W. Hu, Gait-key: a gait-based shared secret key generation protocol for wearable devices, ACM Trans. Sensor Netw. (TOSN) 13 (2017).
[28] G. Margelis, X. Fafoutis, G. Oikonomou, R. Piechocki, T. Tryfonas, P. Thomas, Efficient dct-based secret key generation for the internet of things, Ad Hoc Netw. 92 (2019) 101744.
[29] A.H. Johnston, G.M. Weiss, Smartwatch-based biometric gait recognition, in: 2015 IEEE 7th International Conference on Biometrics Theory, Applications and Systems (BTAS), IEEE, 2015, pp. 1–6.
[30] G. Cola, M. Avvenuti, F. Musso, A. Vecchio, Gait-based authentication using a wrist-worn device, in: Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services, ACM, 2016, pp. 208–217.
[31] N. Al-Naffakh, N. Clarke, F. Li, P. Haskell-Dowland, Unobtrusive gait recognition using smartwatches, in: 2017 International Conference of the Biometrics Special Interest Group (BIOSIG), IEEE, 2017, pp. 1–5.
[32] B. Shrestha, M. Mohamed, N. Saxena, Walk-unlock: zero-interaction authentication protected with multi-modal gait biometrics, arXiv Preprint arXiv:1605.00766 (2016).
[33] J.R. Kwapisz, G.M. Weiss, S. Moore, et al., Cell phone-based biometric identification, in: BTAS' 2010, IEEE, 2010, pp. 1–7.
[34] P.K. Misra, W. Hu, Y. Jin, J. Liu, A. Souza de Paula, N. Wirstrom, T. Voigt, Energy efficient gps acquisition with sparse-gps, in: IPSN' 2014, IEEE Press, 2014, pp. 155–166.
[35] Y. Shen, W. Hu, M. Yang, B. Wei, S. Lucey, C.T. Chou, Face recognition on smartphones via optimised sparse representation classification, in: IPSN '14, IEEE Press, 2014, pp. 237–248.
[36] B. Wei, W. Hu, M. Yang, C.T. Chou, Radio-based device-free activity recognition with radio frequency interference, in: IPSN' 2015, ACM, 2015, pp. 154–165.
[37] Y. Wang, H. Zhang, F. Yang, A weighted sparse neighbourhood-preserving projections for face recognition, IETE J. Res. 63 (3) (2017) 358–367.
[38] W. Xu, Y. Shen, N. Bergmann, W. Hu, Sensor-assisted face recognition system on smart glass via multi-view sparse representation classification, in: IPSN' 2016, IEEE, 2016, pp. 1–12.
[39] B. Wei, M. Yang, Y. Shen, R. Rana, C.T. Chou, W. Hu, Real-time classification via sparse representation in acoustic sensor networks, in: Sensys' 2013, ACM, 2013, p. 21.
[40] M. Shoaib, S. Bosch, O.D. Incel, H. Scholten, P.J. Havinga, A survey of online activity recognition using mobile phones, Sensors 15 (1) (2015) 2059–2085.
[41] Y.-S. Lee, S.-B. Cho, Activity recognition using hierarchical hidden Markov models on a smartphone with 3d accelerometer, in: International Conference on Hybrid Artificial Intelligence Systems, Springer, 2011, pp. 460–467.
[42] A. Mannini, A.M. Sabatini, Machine learning methods for classifying human physical activity from on-body accelerometers, Sensors 10 (2) (2010) 1154–1175.
[43] Y. Zhang, G. Pan, K. Jia, M. Lu, Y. Wang, Z. Wu, Accelerometer-based gait recognition by sparse representation of signature points with clusters, IEEE Trans. Cybern. 45 (9) (2015) 1864–1875.
[44] A. Rai, K.K. Chintalapudi, V.N. Padmanabhan, R. Sen, Zee: zero-effort crowdsourcing for indoor localization, in: Mobicom' 2012, ACM, 2012, pp. 293–304.
[45] M. Kose, O.D. Incel, C. Ersoy, Online human activity recognition on smart phones, in: Workshop on Mobile Sensing: From Smartphones and Wearables to Big Data, 2012, pp. 11–15.
[46] M. Aharon, M. Elad, A. Bruckstein, K-Svd: an algorithm for designing overcomplete dictionaries for sparse representation, IEEE Trans. Signal Process. 54 (11) (2006) 4311–4322.
[47] K. Engan, S.O. Aase, J.H. Husøy, Multi-frame compression: theory and design, Signal Process. 80 (10) (2000) 2121–2140.
[48] D.D. Lee, H.S. Seung, Algorithms for non-negative matrix factorization, in: Advances in neural information processing systems, 2001, pp. 556–562.
[49] D. Donoho, Y. Tsaig, Fast solution of $\ell_1$ -Norm minimization problems when the solution may be sparse, IEEE Trans. Inf. Theory 54 (11) (2008) 4789–4812.
[50] J. Wright, A. Yang, A. Ganesh, S. Sastry, Y. Ma, Robust face recognition via sparse representation, PAMI (2009) 210–227.
[51] M.B. Crouse, K. Chen, H. Kung, Gait recognition using encodings with flexible similarity measures (2014).
[52] J. Frank, S. Mannor, D. Precup, Activity and gait recognition with time-delay embeddings., AAAI, Citeseer, 2010.


**Weitao Xu** is an Assistant Professor at the Department of Computer Science at City University of Hong Kong. Before that, he was a Postdoctoral Research Associate at the School of Computer Science and Engineering (CSE) at UNSW from June 2017 to August 2019. He obtained his PhD degree from the University of Queensland in 2017 (advised by Prof. Neil Bergmann and Dr. Wen Hu). He received his B.E. degree in Communication Engineering and M.E. degree in Communication and Information System (advised by Prof. Dongfeng Yuan) both from the School of Information Science and Engineering, Shandong University (SDU), China, in 2010 and 2013, respectively. His research areas include mobile computing, sensor network and IoT.

**Yiran Shen** received the BE degree in communication engineering from Shandong University, China, and the PhD degree in computer science and engineering from the University of New South Wales. He published regularly at top-tier conferences and journals like ACM SenSys, the IEEE/ACM Information Processing in Sensor Networks, IEEE Ubicomp, IEEE Percom, the IEEE Transactions on Mobile Computing, etc. His current research interests include wearable/ mobile computing, wireless sensor networks and applications of compressive sensing. He is a member of the IEEE.

**Chengwen Luo** received the PhD degree from the School of Computing, National University of Singapore (NUS), Singapore. He was a postdoctoral researcher in computer science and engineering at the University of New South Wales (UNSW), Australia. He is currently an assistant professor in the College of Computer Science and Software Engineering, Shenzhen University, China. He has authored or co-authored research papers in top venues of mobile computing and wireless sensor network, such as ACM SenSys and ACM/IEEE IPSN. His research interests include mobile and pervasive computing, indoor localization, wireless sensor network, and security aspects of Internet of Things.

**Jianqiang Li** is the deputy dean of the School of Computer Software of Shenzhen University, the executive director of the Institute of Network and Information Security, and the deputy director of the Mobile Internet Application Middleware Technology Engineering Laboratory of Guangdong Province. In 2016, he was selected as a member the “Liyuan Youqing” training plan of Shenzhen University. Born in 1980, he obtained a bachelor’s degree in engineering and a doctorate in engineering from South China University of Technology in 2003 and 2008 respectively, and he started teaching at Shenzhen University in August 2008. He has been engaged in research work on artificial intelligence, robotics, internet of things, and mobile medical for many years. And He has hosted and completed three projects of the National Natural Science Foundation of China, including one key project, one for face and youth project. He had published more than 40 papers in the fields of robotics, internet of things, hybrid systems, mobile medicine, optimization control and artificial intelligence, and more than thirty papers have been received in the three major indexes. He has successfully applied for ten national patents and eight software copyrights.

**Wei Li** received the Ph.D. degree from the School of Information Technologies, The University of Sydney, Sydney, Australia, where he is currently a Research Associate with the Centre for Distributed and High Performance Computing. His research interests include Internet of Things, edge computing, sustainable computing, task scheduling, and energy efficiency and optimization. He was the recipient of four IEEE or ACM conference Best Paper Awards. He has received the IEEE TCSC Award for Excellence in Scalable Computing for Early Career Researchers (2018) and the IEEE Outstanding Leadership Award (2018). He is a Senior Member of the IEEE Computer Society and a member of the ACM

**Albert Y. Zomaya** is the Chair Professor of high performance computing and networking with the School of Information Technologies, The University of Sydney, Sydney, Australia, and he also serves as the Director with the Centre for Distributed and High Performance Computing. He has authored and coauthored more than 600 scientific papers and articles and is author, coauthor, or Editor of more than 20 books. He is the Founding Editor in Chief of the IEEE TRANSACTIONS ON SUSTAINABLE COMPUTING and serves as an Associate Editor for more than 20 leading journals. He served as an Editor-inChief for the IEEE TRANSACTIONS ON COMPUTERS (2011–2014). Prof. Zomaya was the recipient of the IEEE Technical Committee on Parallel Processing Outstanding Service Award (2011), the IEEE Technical Committee on Scalable Computing Medal for Excellence in Scalable Computing (2011), and the IEEE Computer Society Technical Achievement Award (2014). He is a Chartered Engineer, a fellow of AAAS, IEEE, and IET. His research interests include the areas of parallel and distributed computing and complex systems.